



INSTITUTO POLITÉCNICO NACIONAL  
ESCUELA SUPERIOR DE CÓMPUTO

ESCOM

*Trabajo de titulación*

“Sistema web para la evaluación automática  
de programas de Karel el Robot.”

*Que para cumplir con la opción de titulación curricular en la carrera de:*

“Ingeniería en Sistemas Computacionales”

*Presentan*

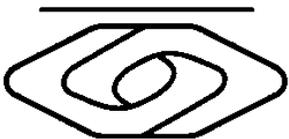
Larissa Arcos Reyes

Miguel Ángel Martínez Reyes

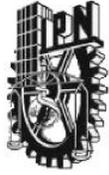
Juan Carlos Pérez Alvarez

*Director*

M. en C. *Edgardo Adrián Franco Martínez*



ESCOM



**Instituto Politécnico Nacional**  
**Escuela Superior de Computo**



**Registro de titulación:**  
ISCCR050-2013A003/2015

**Fecha:**  
24 de abril de 2015

Documento Técnico

**“Sistema web para la evaluación automática de programas de Karel el Robot”**

Presentan:

**Larissa Arcos Reyes<sup>1</sup>**

**Miguel Ángel Martínez Reyes<sup>2</sup>**

**Juan Carlos Pérez Alvarez<sup>3</sup>**

Director

**M. en C Edgardo Adrián Franco Martínez**

**Resumen:**

Los evaluadores automáticos son de gran utilidad dentro de las competencias de informática, debido a que proveen una plataforma para la recopilación y evaluación imparcial de las soluciones desarrolladas por los competidores. Actualmente existen diversos sitios web que tienen disponibles evaluadores automáticos para los lenguajes de programación C, Java, entre otros.

En los entrenamientos impartidos para la Olimpiada Mexicana de Informática (OMI) y en las evaluaciones realizadas, el uso de una herramienta como esta es imprescindible, debido a que facilita la enseñanza de la lógica de programación, para que los futuros participantes puedan encontrar soluciones a problemas cada vez más complejos.

La propuesta desarrollada consiste en un nuevo sistema web, que mejora y añade características al sistema que actualmente se utiliza con este fin dentro de la Olimpiada Mexicana de Informática, implementando un clasificador de los problemas, una plataforma capaz de administrar concursos y evaluar las soluciones, disponible para cualquier persona interesada en adquirir o mejorar sus habilidades en programación.

**Palabras clave:** Sistema web, Karel el robot, Software educativo, Evaluador de programas

---

<sup>1</sup> ar.larissa@outlook.com

<sup>2</sup> miickeyreyez@gmail.com

<sup>3</sup> j.perez@outlook.com

## *Advertencia*

*"Este documento contiene información desarrollada por la Escuela Superior de Cómputo del Instituto Politécnico Nacional a partir de datos y documentos con derecho de propiedad y por lo tanto su uso quedar restringido a las aplicaciones que explícitamente se convengan."*

La aplicación no convenida exime a la escuela su responsabilidad técnica y da lugar a las consecuencias legales que para tal efecto se determinen.

Información adicional sobre este reporte técnico podrá obtenerse en: En La Subdirección Académica de la Escuela Superior de Cómputo del Instituto Politécnico Nacional, situada en Av. Juan de Dios Bátiz s/n Teléfono: 57296000 Extensión 52000

## Agradecimientos.

Quiero agradecer a mis padres Rogelio Arcos y Alba Reyes, a mi hermana Yesenia por apoyarme y, hacer sacrificios para que yo pudiera sacar mis sueños adelante. Por creer y confiar en mí, por los valores y enseñanzas que me inculcaron ya que sin su apoyo nada de esto sería posible.

Al Instituto Politécnico Nacional y a la Escuela Superior de Cómputo por brindarme las herramientas necesarias para alcanzar mis metas.

A Juan Carlos y Miguel Ángel porque a pesar de las dificultades conseguimos nuestro objetivo, por los momentos felices y los momentos tristes, por todos esos días de trabajo. Gracias por dejarme trabajar con ustedes y por compartir metas conmigo.

Al profesor Edgardo Franco por enseñarme que la mayor grandeza viene de la humildad, ya que no sólo es excelente profesor sino un gran ser humano. Por su apoyo, comprensión y dirección durante el desarrollo de este trabajo.

A Enrique Lira, Luis Humberto Ríos, Carlos Jiménez, Javier Negrete, Ethan Jiménez y Amilkhael Chavez por ser mis amigos, mis compañeros de equipo, por estar conmigo durante mi estancia en ESCOM. Gracias por los consejos, las críticas, los abrazos, los regaños. Porque de ustedes aprendí muchas cosas, la más importante el valor de la amistad.

**Larissa Arcos Reyes**

Agradezco a mis padres, así como mi hermano, por estar ahí desde siempre, apoyándome, por ser el soporte para lograr esta meta tan importante, por creer en mí, porqué a pesar de todo lo que hemos vivido ya sea para bien o para mal hemos aprendido a salir juntos como familia, gracias por todo, no tengo palabras para agradecerles lo que han hecho por mí, los amo.

A mi abuelo que siempre ha estado a mi lado a pesar de ya no continuar aquí, este logro también es tuyo, no imaginas cuanto me hubiera gustado que estuvieras aquí en este momento tan importante para mí, gracias por tantas enseñanzas que me diste a lo largo de la vida.

A mis amigos, por estar ahí en las situaciones tanto buenas y malas, no solo en cuestiones académicas, por sus consejos, apoyo y enseñanzas, son una parte importante en mi vida, gracias por estar siempre ahí presentes a pesar de todo.

Al Instituto Politécnico Nacional así como a la ESCOM por proporcionarme los medios necesarios para alcanzar el éxito a lo largo de estos 7 años de estudio dentro de la institución, orgullosamente Politécnico.

Y finalmente mis compañeros de trabajo terminal, así como mi director, que sin ellos nada de esto hubiera sido posible, gracias por esta convivencia que nos permitió aprender tanto cosas nuevas, gracias por compartir esta experiencia personal tan importante, les deseo éxito en todo aquello que decidan emprender en la vida.

**Miguel Angel Martinez Reyes**

Agradezco a mis padres, Juan y Laura, por haberme dado siempre su apoyo incondicional en todo lo que quise intentar, su amor, y esas palabras de aliento que me impulsaron a lograr muchos objetivos. Gracias.

A mi hermana Ana Laura, mi abuelita Socorro, y a toda mi familia que siempre estuvo ahí para apoyarme, darme ánimos, e impulsarme en todos los proyectos en los que me involucre. Sin ustedes nada de esto hubiera sido posible.

A Miguel Ángel, por ser mi amigo incondicional. A Larissa, que a pesar de los problemas, logramos sacar adelante este gran proyecto. Gracias a los dos por su apoyo y dedicación, fue un gran honor para mí haber trabajado con ustedes.

Al M. en C. Edgardo Adrián Franco, por ser un gran profesor y una gran persona. Por su preocupación en la formación integral de sus alumnos, por su apoyo en el Club de Algoritmia y en este Trabajo Terminal.

Al Instituto Politécnico Nacional, mi segundo hogar desde hace siete años. Hogar en el que pude llevar a cabo lo que me apasiona.

**Juan Carlos Pérez Alvarez**

## Índice

Índice de Tablas.....	12
Índice de Imágenes.....	14
Introducción.....	15
Antecedentes .....	15
Estado del arte.....	18
Problemática.....	20
Objetivos.....	20
Objetivos Particulares.....	20
Justificación.....	20
Marco Teórico.....	21
Programación .....	21
Importancia de la programación.....	21
¿Por qué aprender a programar? .....	22
Técnicas para aprender a programar.....	23
Karel, el robot.....	24
Ejemplo de un programa de Karel.....	26
Metodología.....	29
Patrón Modelo Vista Controlador.....	29
Análisis de factibilidad.....	31
Factibilidad operativa.....	31
Factibilidad técnica.....	35
Tecnologías que operan del lado del cliente.....	36
Tecnologías que operan del lado del servidor.....	37
Gestores de base de datos .....	38
Otras tecnologías.....	39
Requerimientos técnicos .....	40
Factibilidad económica.....	41
Análisis de Riesgos.....	43
Riesgos Tecnológicos.....	43

Riesgos Organizacionales.....	44
Riesgos de estimación.....	44
Plan de contingencia.....	45
MÓDULO DE GESTIÓN DE USUARIOS.....	46
<i>DOCUMENTO DE REQUISITOS</i> .....	46
Resumen Ejecutivo.....	46
Metas y no metas.....	46
Requerimientos funcionales.....	47
Requerimientos no funcionales .....	47
Dependencias.....	47
Requisitos detallados del modulo .....	47
Riesgos y problemas detectados .....	49
Ediciones al documento .....	49
Apéndice.....	49
Definiciones y acrónimos .....	49
MÓDULO DE GESTIÓN DE CONCURSOS .....	50
<i>DOCUMENTO DE REQUISITOS</i> .....	50
Resumen Ejecutivo.....	50
Metas y no metas.....	50
Requerimientos funcionales.....	50
Requerimientos no funcionales .....	52
Dependencias.....	52
Requisitos detallados del módulo .....	52
Riesgos y problemas detectados .....	53
Ediciones al documento .....	53
Apéndice.....	53
Definiciones y acrónimos .....	53
MÓDULO DE INTERACCIÓN CON REDES SOCIALES .....	54
<i>DOCUMENTO DE REQUISITOS</i> .....	54
Resumen Ejecutivo.....	54

Metas y no metas.....	54
<i>Metas</i> .....	54
<i>No metas</i> .....	54
Requerimientos funcionales.....	55
Requerimientos no funcionales .....	55
Dependencias.....	55
Requisitos detallados del modulo .....	55
Riesgos y problemas detectados .....	56
Ediciones al documento .....	56
Apéndice.....	56
Definiciones y acrónimos .....	56
MÓDULO DEL EVALUADOR.....	57
<i>DOCUMENTO DE REQUISITOS</i> .....	57
Resumen Ejecutivo.....	57
Metas y no metas.....	57
Requerimientos funcionales.....	57
Requerimientos no funcionales .....	57
Dependencias.....	58
Requisitos detallados del modulo .....	58
Riesgos y problemas detectados .....	59
Ediciones al documento .....	59
Apéndice.....	59
Definiciones y acrónimos .....	59
Lista de errores del compilador .....	59
MODULO DEL CLASIFICADOR DE PROBLEMAS.....	62
<i>DOCUMENTO DE REQUISITOS</i> .....	62
Resumen Ejecutivo.....	62
Metas y no metas.....	62
Requerimientos funcionales.....	62
Requerimientos no funcionales .....	63

Dependencias.....	63
Requisitos detallados del modulo .....	63
Riesgos y problemas detectados .....	64
Ediciones al documento .....	64
Apéndice.....	64
Definiciones y acrónimos .....	64
DOCUMENTO DE DIAGRAMAS DEL SISTEMA.....	65
Diagrama de arquitectura.....	65
Diagrama de base de datos .....	67
Diagramas de casos de uso .....	68
Módulo Gestión de Usuarios .....	68
Módulo Interacción con Redes Sociales.....	68
Módulo Gestión de Concursos.....	69
Módulo Gestión de Problemas .....	69
Diagramas de secuencia.....	70
Crear nuevo concurso.....	71
Inicio de sesión.....	72
Evaluar Problema.....	73
Diagrama de flujo de datos.....	74
MÓDULO DEL EVALUADOR.....	76
ESPECIFICACIONES DE DISEÑO DEL DESARROLLO.....	76
Resumen del módulo .....	76
Descripción de la arquitectura .....	76
Detalles del diseño .....	77
La clase KarelJudge.....	77
Windows Communication Foundation.....	78
Realización de peticiones .....	80
Respuesta del servicio web.....	80
Pruebas .....	80
Primer feedback.....	80

Resultados.....	81
Ediciones al documento .....	81
Apéndice.....	82
Definiciones y acrónimos .....	82
MÓDULO DEL CLASIFICADOR DE PROBLEMAS.....	83
ESPECIFICACIONES DE DISEÑO DEL DESARROLLO.....	83
Resumen del módulo .....	83
Descripción de la arquitectura .....	83
Detalles del diseño .....	84
Migración de los datos del sitio web anterior.....	84
Pruebas .....	85
Primer feedback.....	85
Resultados.....	85
Ediciones al documento .....	85
MÓDULO GESTIÓN DE USUARIOS .....	86
ESPECIFICACIONES DE DISEÑO DEL DESARROLLO.....	86
Resumen del módulo .....	86
Descripción de la arquitectura .....	86
Detalles del diseño .....	87
Pruebas .....	87
Primer feedback.....	87
Resultados.....	88
Segundo Feedback.....	88
Resultados.....	89
Ediciones al documento .....	89
MÓDULO DE INTERACCIÓN CON REDES SOCIALES .....	90
ESPECIFICACIONES DE DISEÑO DEL DESARROLLO.....	90
Resumen del módulo .....	90
Descripción de la arquitectura .....	90
Detalles del diseño .....	91

La clase ExternalLoginConfirmationViewModel .....	91
La clase ExternalLoginViewModel.....	91
Primer feedback.....	92
Resultados.....	92
MÓDULO GESTIÓN DE CONCURSOS.....	93
ESPECIFICACIONES DE DISEÑO DEL DESARROLLO.....	93
Resumen del módulo .....	93
Descripción de la arquitectura .....	93
Detalles del diseño .....	94
Pruebas .....	94
Primer feedback .....	95
Resultados.....	95
Ediciones al documento .....	95
Riesgos mitigados y planes de contingencia .....	96
Cambio del gestor de base de datos. ....	96
Problemas con los módulos existentes. ....	102
Problemas en la migración.....	102
Pruebas y Resultados.....	104
Módulo del Evaluador.....	105
Condiciones.....	106
Resultados.....	107
Módulo Gestión de Usuarios.....	108
Condiciones.....	108
<i>Resultados</i> .....	109
Módulo Gestión de problemas.....	109
Condiciones.....	110
Resultados.....	110
Módulo Gestión de Concursos.....	110
Condiciones.....	111
Resultados.....	111

Conclusiones .....	112
Glosario .....	113
Referencias generales .....	113
Anexo A: Manual de Usuario.....	117
Inicio de sesión .....	117
Registro.....	128
Problemas .....	142
Listas .....	157
Concursos .....	166
Asesorías.....	184
Información adicional.....	187
Anexo B: Manual de Instalación.....	193
Requerimientos necesarios .....	193
Proceso de instalación .....	193
Anexo C: Términos y condiciones del sitio .....	198

## Índice de Tablas

Tabla 1- Compración entre Karelotitlán y la propuesta.....	17
Tabla 2-Comparación del actual Karelotitlán y la propuesta del presente protocolo. ....	18
Tabla 3-Sistemas Web con características compartidas con la propuesta.....	19
Tabla 4- Instrucciones de Karel el Robot.....	25
Tabla 5-Condiciones detectadas por Karel el robot en Pascal.....	26
Tabla 6-Condiciones detectadas por Karel el robot en Java.....	26
Tabla 7-Pesos de los Puntos Objeto con base al modelo de Boehm.....	32
Tabla 8-Valores del ratio de productividad .....	32
Tabla 9-Análisis de tipos objeto con base a cantidad y peso.....	32
Tabla 10- División de actividades.....	33
Tabla 11- Perfiles .....	34
Tabla 12-Tecnologías que operan del lado del cliente .....	36
Tabla 13-Tecnologías que operan del lado del servidor .....	37
Tabla 14-Gestor de base de datos.....	38
Tabla 15-Servicios y Equipo.....	41
Tabla 16-Software para el desarrollo del proyecto .....	42
Tabla 17- Sueldos.....	42
Tabla 18-Costo Total del Proyecto.....	42
Tabla 19-Riesgos Tecnológicos.....	43
Tabla 20-Riesgos Organizacionales.....	44
Tabla 21-Riesgos de estimación .....	44
Tabla 22-Plan de contingencia .....	45
Tabla 23-Requerimientos funcionales módulo gestión de usuarios.....	47
Tabla 24-Requerimientos no funcionales módulo gestión de usuarios.....	47
Tabla 25-Dependencias módulo gestión de usuarios .....	47
Tabla 26-Requerimientos funcionales módulo gestión de concursos .....	50
Tabla 27-Requerimientos no funcionales módulo gestión de concursos.....	52
Tabla 28-Dependencias módulo gestión de concursos.....	52
Tabla 29-Requerimientos funcionales módulo interacción con redes sociales.....	55
Tabla 30-Requerimientos no funcionales módulo interacción con redes sociales .....	55
Tabla 31- Dependencias módulo interacción con redes sociales .....	55
Tabla 32-Requirimientos funcionales módulo evaluador .....	57
Tabla 33-Requerimientos no funcionales módulo evaluador.....	58
Tabla 34-Dependencias módulo evaluador.....	58
Tabla 35-Requerimientos funcionales módulo gestión de problemas.....	62
Tabla 36-Requerimientos no funcionales módulo gestión de problemas .....	63
Tabla 37-Descripción de clases .....	77

Tabla 38-Detalles de la evaluación .....	78
Tabla 39-Primer feedback módulo evaluador .....	81
Tabla 40-Resultados primer feedback módulo evaluador.....	81
Tabla 41-Descripción de las clases del módulo gestión de problemas.....	84
Tabla 42-Feedback del módulo gestión de problemas .....	85
Tabla 43-Resultados del primer feedback del módulo gestión de problemas.....	85
Tabla 44-Primer feedback módulo gestión de usuarios.....	88
Tabla 45-Resultado primer feedback módulo gestión de usuarios .....	88
Tabla 46-Segundo Feedback módulo gestión de usuarios.....	89
Tabla 47-Resultado segundo feedback módulo gestión de usuarios.....	89
Tabla 48-Primer feedback módulo interacción con redes sociales .....	92
Tabla 49-Resultado primer feedback módulo gestión de usuarios .....	92
Tabla 50-Feedback del módulo gestión de concursos .....	95
Tabla 51-Resultados del primer feedback del módulo gestión de problemas.....	95
Tabla 52- Descripción de la clase EvaluadorTest.....	105
Tabla 53- Descripción de la clase UsuarioControllerTest.....	108
Tabla 54- Descripción de la clase ProblemaControllerTest .....	109
Tabla 55-Descripción de la clase ConcursoControllerTest.....	111

## Índice de Imágenes.

Imagen 1: Diagrama del proceso actual para subir una solución a Karelotitlán .....	16
Imagen 2: Modelo de enseñanza de la programación.....	24
Imagen 3: El mundo de Karel .....	25
Imagen 4: Mundo de ejemplo .....	27
Imagen 5: Solución del mundo de ejemplo .....	27
Imagen 6: Diagrama de Arquitectura.....	66
Imagen 7: Diagrama de base de datos.....	67
Imagen 8: Casos de uso-Gestión de usuarios.....	68
Imagen 9: Casos de uso: Redes sociales.....	68
Imagen 10: Casos de uso- Gestion de concursos .....	69
Imagen 11: Casos de uso: Gestión de problemas .....	69
Imagen 12: Secuencia .....	71
Imagen 13: Secuencia Inicio de sesión.....	72
Imagen 14: Evaluar problema .....	73
Imagen 15: Diagrama de clases-Módulo evaluador.....	76
Imagen 16: Diagrama de clases: Módulo evaluador .....	84
Imagen 17: Clases del módulo Usuarios.....	86
Imagen 18-Clase EvaluadorTest.....	105
Imagen 19: Clase UsuarioControllerTest.....	108
Imagen 20: Clase ProblemaControllerTest.....	109
Imagen 21: Clase ConcursoControllerTest.....	110

# TRABAJO TERMINAL 2013-A003

## Sistema web para la evaluación automática de programas de Karel el Robot.

### Introducción

Los evaluadores automáticos son de gran utilidad dentro de las competencias de informática, debido a que proveen una plataforma para la recopilación y evaluación imparcial de las soluciones desarrolladas por los competidores. Actualmente existen diversos sitios web que tienen disponibles evaluadores automáticos para los lenguajes de programación C, Java, entre otros.

En los entrenamientos impartidos para la Olimpiada Mexicana de Informática (OMI) y en las evaluaciones realizadas, el uso de una herramienta como esta es imprescindible, debido a que facilita la enseñanza de la lógica de programación, para que los futuros participantes puedan encontrar soluciones a problemas cada vez más complejos.

La propuesta desarrollada consiste en un nuevo sistema web, que mejora y añade características al sistema que actualmente se utiliza con este fin dentro de la Olimpiada Mexicana de Informática, implementando un clasificador de los problemas, una plataforma capaz de administrar concursos y evaluar las soluciones, disponible para cualquier persona interesada en adquirir o mejorar sus habilidades en programación.

### Antecedentes

La Olimpiada Mexicana de Informática (OMI) es un concurso a nivel nacional para jóvenes de secundaria y preparatoria con facilidad para resolver problemas prácticos mediante la lógica y el uso de herramientas computacionales [1], que busca promover el desarrollo tecnológico en México y encontrar a los mejores programadores, para que posteriormente puedan competir a nivel internacional en la Olimpiada Internacional de Informática (IOI) [2].

Los participantes de la OMI son seleccionados a partir de su desempeño en los exámenes que presenten. En las primeras etapas los exámenes son de lógica matemática y razonamiento, posteriormente se realiza un examen práctico utilizando el simulador de "Karel el robot" y por último un examen en el lenguaje de programación C, C++ o Pascal.

Actualmente existe en México un sitio de entrenamiento para el lenguaje Karel llamado "Karelotitlán" [3], el cual es una sencilla aplicación web donde cualquier persona puede

registrarse, encontrar un listado de problemas y comenzar a subir la solución a cualquiera estos. Esta aplicación a veces genera conflictos debido a no fue diseñada pensando en las necesidades de los usuarios ni en el crecimiento de la aplicación.

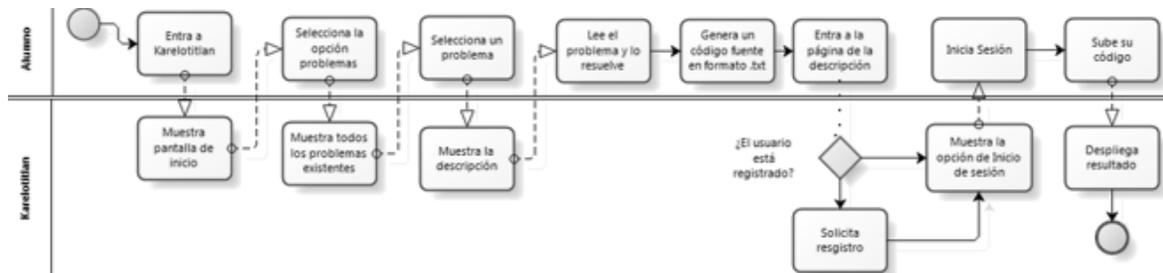


Imagen 1: Diagrama del proceso actual para subir una solución a Karelotitlán

La arquitectura de Karelotitlán está constituida de manera que el servidor web y el evaluador se localizan en el mismo servidor, por lo tanto, éste recibe peticiones sin necesidad de que el usuario envíe soluciones y al acceder a él múltiples usuarios el servicio se pone lento o deja de responder.

Karelotitlán cuenta con las siguientes funcionalidades

Karelotitlán	Propuesta
Visualizar los problemas sin necesidad de registrarse en el sistema	Se permitirá visualizar sólo las listas públicas.
Acceder a una lista única de problemas	Permitirá la creación de varias listas de problemas y elegir que usuarios pueden visualizarla.
Visualizar los datos de los usuarios registrados si se ha iniciado sesión.	Permitirá visualizar los datos de los usuarios registrados si se ha iniciado sesión.
Descargar el simulador y el evaluador de la página.	Contendrá una sección donde se permita la descarga del simulador, evaluador y tutoriales.
Acceder a una lista de preguntas frecuentes.	Contará con una sección de preguntas frecuentes.
No cuenta con un módulo de gestión de usuarios	Permitirá a los usuarios, modificar los datos de su cuenta y darse de baja definitiva del sistema
La carga de problemas se realiza desde la base de datos.	Contará con un módulo con interfaz gráfica que permita gestionar los problemas del

Tabla 1- Comparación entre Karelotitlán y la propuesta

Tiene registrados un poco más de 15 000 usuarios, sin embargo no todos ellos suben soluciones de forma periódica, algunos incluso llevan inactivos varios años. El sistema busca evitar los usuarios inactivos.

A pesar de que ha servido como sitio de entrenamiento por varios años, presenta algunos problemas los cuales se muestran en la tabla 2 junto con la solución planteada.

Sitio actual: Karelotitlán	Solución propuesta
Lento, desperdicio de recursos, a veces se cae el sistema.	Optimizar tiempos de respuesta y carga del servidor (12k usuarios registrados, picos de 150 usuarios simultáneos)
Una sola lista de problemas	Crear listas de problemas con enfoques específicos
No permite hospedar concursos propios	Una plataforma para hospedar concursos con problemas del sitio o personalizados
Problemas clasificados por el administrador	Clasificar los problemas de manera automática
Página web y evaluador juntos	Sistema escalable y distribuido
No hay relación con otros sitios	Integración con redes sociales

Tabla 2-Comparación del actual Karelotitlán y la propuesta del presente protocolo.

## Estado del arte

De acuerdo a la investigación realizada no ha sido desarrollado en la ESCOM un trabajo terminal similar al que proponemos; por lo que se realiza una comparación con algunos sitios Web de características similares a la propuesta, sin embargo ninguno de ellos está orientado a lenguaje Karel, tales sitios se muestran en la Tabla 2.

A pesar de que no existe otro sitio, además de Karelotitlán, se pueden encontrar evaluadores automáticos para lenguajes de programación como C o Java, con una gran variedad de problemas, que son empleados en competencias como el ACM ICPC, la IOI, entre otras.

Sitios de entrenamiento en programación.	Características
<b>Teddy Online Judge</b>	<ul style="list-style-type: none"> <li>-Creado por el Instituto Tecnológico de Celaya</li> <li>-Soporta las soluciones a los problemas en cualquiera de estos lenguajes: C/C++, Java, Python, PHP, VisualBasic.NET, C# o Perl.</li> <li>-Soporta la creación de concursos.</li> <li>-Genera estadísticas semanales de los problemas resueltos.</li> <li>-Necesitas conocimientos de programación para resolver los problemas.</li> </ul>
<b>CodeChef</b>	<ul style="list-style-type: none"> <li>-Vinculado con redes sociales</li> <li>-Soporta soluciones en 48 lenguajes de programación distintos: C, C++, Java, C#, etc.</li> <li>-Crea concursos en línea cada mes.</li> <li>-Separa los problemas de práctica por complejidad</li> <li>-Necesitas conocimientos de programación para resolver los problemas.</li> </ul>
<b>BOCA Online Contest Administrator</b>	<ul style="list-style-type: none"> <li>- Utilizado en el concurso ACM ICPC en Latinoamérica</li> <li>- Permite hospedar concursos en línea y en diferentes sedes, unificando los resultados</li> <li>- Permite el uso de lenguaje C, C++ y Java</li> <li>- No cuenta con una sección de práctica o entrenamiento</li> </ul>

Tabla 3-Sistemas Web con características compartidas con la propuesta.

## **Problemática.**

El sitio web actual no fue diseñado para soportar la cantidad actual de usuarios, lo que ocasiona que el sistema no funcione de manera correcta. En cuanto a la clasificación de los problemas ésta es realizada por el administrador y no siempre se realiza de una manera objetiva. Además actualmente la OMI no cuenta con un sistema que le permita realizar las evaluaciones en la etapa eliminatoria, esto retarda los procesos de selección de los competidores.

## **Objetivos.**

Desarrollar un sistema web para la gestión y publicación de problemas a desarrollar bajo el lenguaje Karel. Así como un módulo para la gestión de usuarios, la evaluación automática de los programas resueltos por estos y la administración de concursos virtuales.

### **Objetivos Particulares.**

1. Desarrollo de un módulo web que gestione diversos tipos de usuarios según las necesidades de entrenamiento actuales de la OMI.
2. Desarrollo de un módulo web que permita publicar y clasificar de manera simple los problemas propuestos para implementar su solución en lenguaje Karel.
3. Implementación de un evaluador para programas en Karel automatizado.
4. Desarrollo de un módulo para administrar concursos de programación virtuales.
5. Propuesta y prototipo de infraestructura para la integración en un único sistema Web.

## **Justificación.**

Actualmente el sitio web "Karelotitlán" es fundamental en el entrenamiento de los jóvenes que desean competir en la OMI, tanto para aquellos que aprenden de manera autodidacta, como aquellos que toman alguno de los cursos impartidos por el comité estatal.

El mejorar las características de este sistema a través de la solución que se plantea, beneficiara a todos aquellos jóvenes que deseen aprender a utilizar este lenguaje, no solo para la competencia de la OMI, sino también para todos aquellos interesados en mejorar sus habilidades de programación, brindando un sistema fiable, y con un contenido mejor organizado, que ayude a los usuarios a identificar los problemas que son aptos para cada uno.

Además, actualmente no existe un sistema que ayude a la organización de los exámenes realizados por el comité estatal en las fases eliminatorias de la OMI, por lo que ésta solución también proveerá de los recursos necesarios para poder organizar un concurso y evaluarlo de manera automática.

Para éste proyecto de Trabajo Terminal se cuenta con el apoyo del Lic. Martin Ibarra Romero, líder de la Olimpiada de Informática del Distrito Federal, con la finalidad de que el sistema que se va a desarrollar sustituya al sitio actual, para el beneficio de los jóvenes que participen en ediciones futuras de la OMI.

## Marco Teórico

### Programación

La programación es un proceso propio del código fuente de un programa computacional, el cual incluye las siguientes etapas:

- Diseño: en esta etapa se realiza la identificación del problema y el planteamiento de la solución, elección de algoritmos, lenguaje de programación así como de técnicas propias de otras áreas distintas a la computación (matemáticas, biología, física, lógica, etc.).
- Codificación: una vez definida la solución al problema, ésta se escribe en el lenguaje de programación para que puede ser interpretada por la computadora.
- Depuración: una vez codificada la solución se deben identificar y corregir los errores de programación que se generaron al codificar.
- Mantenimiento: ya que se corrigen los errores se realiza un monitoreo al código para que no aparezcan nuevos errores y que el funcionamiento del programa siempre sea el esperado.

La programación está regida por un conjunto de reglas y expresiones que ayudan a la computadora a entender instrucciones.

### Importancia de la programación

*"Todo el mundo en este país debería aprender como programar una computadora... porque eso es lo que te enseña como pensar"*

- Steve Jobs

La programación facilita el aprendizaje de conceptos informáticos como iteraciones, condicionales y eventos, además de ganar experiencia en conceptos matemáticos como sistemas coordenados, variables y números al azar, requiere de una lógica distinta al pensamiento humano natural además de conocimientos de diferentes áreas a la computación y también de la sintaxis del lenguaje de programación que se va a emplear.

La programación va más allá de saber un lenguaje, sino se tiene lógica para expresar una idea será imposible dar solución a un problema.

Aprender código ayuda a la resolución de problemas, más allá del manejo de algoritmos se combinan muchas de las habilidades de aprendizaje del siglo XXI, la cuales serán críticas para el éxito en el futuro: pensamiento creativo, claridad en la comunicación, análisis sistemático, colaboración efectiva, diseño reiterativo y aprendizaje continuo.

Mitchel Resnik (creador de Scratch, programa gratuito para programar juegos y tarjetas interactivas) habla sobre la importancia de desarrollar fluidez con el uso de tecnología.

Fluidez es saber expresarse a través del lenguaje, para ser fluido con la tecnología, se debe aprender no solo cómo interactuar con la computadora sino también cómo crear con ella, es decir aprender a ser persistentes y a razonar sistemáticamente. [4]

Es importante por esa razón aprender a programar, los problemas que existen y que a futuro existirán tendrán soluciones en el código, es por eso mismo que grandes países están considerando la idea de implementar el *"learn to coding"* o enseñar a programar desde la educación básica.

¿Por qué aprender a programar?

*"La programación expande la mente y ayuda a pensar de una manera algorítmica, es decir, de una forma mucho más ordenada. Sabiendo un poco de programación básica, puede ayudar a las personas a mejorar y automatizar tareas que realizan en sus trabajos de la vida cotidiana."*

Vivimos en un mundo donde la tecnología y la computación están por todas partes la programación es una habilidad que prácticamente garantizará el conseguir un trabajo, todos aquellos que no desarrollen dichas habilidades quedarán fuera del mercado laboral en algunos muchos años, actualmente en la mayoría de los países pasa un fenómeno que los economistas llaman desempleo estructural: los puestos de trabajo están disponibles, pero la fuerza de trabajo no está capacitada para esos trabajos. [5]

En una reciente investigación llevada a cabo por la revista Forbes se dio a conocer unos resultados que muestran que en el mundo el mejor lugar para invertir su dinero está en los desarrolladores de software; nos estamos enfrentando a una economía que depende cada vez más en las habilidades técnicas duras, no las habilidades suaves, para la productividad.

Es por eso que la importancia de poseer habilidades de programación va tomando cada vez más sentido, existen compañías educativas como Code Academy cuyo fin es la enseñanza de la programación en diferentes lenguajes de programación o Treehouse que se encarga en el desarrollo de aplicaciones móviles con Ruby, Rails y PHP; en Estados Unidos existen varias iniciativas todas enfocadas en la enseñanza de la programación de los niños, como Scratch (aplicación para la animación y creación de videojuegos), Lego WeDo (aplicación para la enseñanza de la robótica utilizando piezas LEGO) y la iniciativa code.org [6] que trabaja por ampliar los procesos de formación en informática y programación, con el propósito de hacerlos parte del plan de estudios en las escuelas

desde nivel elemental; de tal manera que aprender código se convierta en un área como biología, física, química o álgebra.

Existen otros países que han gastado grandes cantidades de dinero en ordenadores eficientes y potentes sabiendo que no son un simple gasto, sino una inversión de futuro, es decir, herramientas tecnológicas fundamentales para la educación en las escuelas y para que los alumnos desde edades más tempranas se familiaricen con ellas y aprendan a dominarlas.

Estonia es de los principales países pioneros que están entrando de lleno al mundo digital donde todos los trámites se pueden realizar mediante internet, en Estonia ya se ha aprobado un plan para enseñar programación en las escuelas, dentro de la enseñanza primaria, así mismo los profesores están recibiendo formación sobre cómo enfocar y qué conocimientos impartir en esta asignatura, mientras que el material se ha ido elaborando en los últimos meses.

Está previsto que para el próximo ciclo escolar comiencen las clases con un primer curso, que se irá ampliando progresivamente.

Por el momento se trata de un programa piloto limitado a un grupo de colegios, pero el plan es extenderlo al resto de centros poco a poco.

En Reino Unido también están prestando atención a esta nueva arista de la educación, por el momento existe una iniciativa para aprender desarrollo como actividad extraescolar y se pretende que el 25% de las escuelas de primaria se hayan asociado al plan en el año 2014.

Aprender a programar y a descomponer un programa general en otros más pequeños y abordables o localizar errores y repeticiones son algunas de las capacidades que son impulsadas, actualmente, las matemáticas son una necesidad, al mezclar ambas ciencias se va desarrollando no sólo la comprensión para entender estas materias sino el interés por las mismas así como el impulso de la creatividad, que es propulsada por el reto de crear un programa que funcione e incluso que se diferencie de otros ya existentes.

#### Técnicas para aprender a programar

Existen métodos de enseñanza que se fundamentan en algún paradigma de programación específico, como el paradigma funcional, imperativo o imperativo con aporte a la orientación de objetos, dentro de un paradigma determinado se visualizan varios enfoques para la enseñanza, algunos se basan en la enseñanza del lenguajes en particular utilizando la sintaxis y la semántica, otros emplean un lenguaje algorítmico lo bastante general para permitir la traducción a cualquier lenguaje de programación.

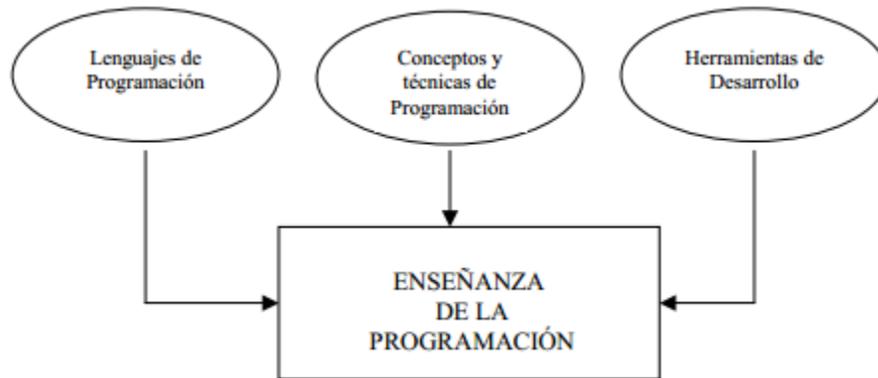


Imagen 2: Modelo de enseñanza de la programación

El primero de estos enfoques tiene el inconveniente de ligar los conceptos básicos a un lenguaje determinado, el cual tiene características y especificaciones propias, donde los conceptos generales se pierden de vista. [7]

Karel, el robot.

“Karel” es un lenguaje de programación estructurado, utilizado para la enseñanza de algoritmos. Como lenguaje de programación no tiene un fin práctico, pues su única finalidad es el desarrollo de la lógica, así como la introducción a la programación para jóvenes.

Fue introducido por Richard Pattis en 1981, representado como un robot simple que vive en un mundo simple. Karel surgió como parte de un proyecto del Departamento de Ciencias de la Computación en la Universidad Carnegie Mellon, el cual buscaba la creación de un ambiente gráfico para manipular las instrucciones de un robot [8]

Este lenguaje fue adaptado para enseñar a los participantes de la OMI a entender cómo funciona la lógica de los lenguajes de programación y sus principios básicos, tales como sentencias de control, comparaciones, búsquedas, ordenamientos y recursividad, lo que permite a los aprendices poder más tarde implementar estos principios en lenguajes formales. Permite el seguimiento de una secuencia de pasos y la elección de los pasos necesarios para alcanzar un objetivo, ya que los alumnos sin bases de programación no son capaces de expresar la solución de un problema en instrucciones para una computadora.

“Karel el robot” es la implementación de un robot virtual representado por una flecha; se simula el viaje a través de su mundo, que es una cuadrícula que representa calles y avenidas (cien avenidas y cien calles), paredes (que pueden ser recorridas a menos que estén bloqueadas), así como con una mochila para guardar los “beepers” o zumbadores que puede recoger con ayuda de su brazo mecánico, éstos son utilizados para contar o marcar el camino.

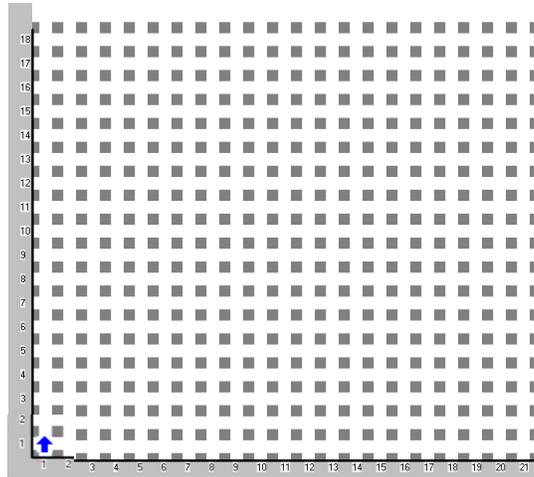


Imagen 3: El mundo de Karel

La única forma que se tiene para interactuar con Karel es a través de una serie de instrucciones, previamente definidas, las cuales puede comprender y que pueden o no generar errores dependiendo de las condiciones del mundo en el cual se desarrollen.

Pascal	Java	Descripción
<b>avanza</b>	<b>move()</b>	Avanza una calle o avenida en la orientación actual de Karel. Condición: no debe existir un muro enfrente al momento de su ejecución.
<b>recoge-zumbador</b>	<b>pickbeeper()</b>	Activa el brazo mecánico de Karel y le indica que debe recoger el zumbador de la posición actual. Condición: debe existir un zumbador al momento de su ejecución
<b>deja-zumbador</b>	<b>putbeeper()</b>	Activa el brazo robot de Karel y le indica que debe tomar un zumbador de su mochila y colocarlo en la posición actual. Condición: Debe haber zumbadores en la mochila al momento de su ejecución
<b>gira-izquierda</b>	<b>turnleft()</b>	Karel gira sobre su propio eje un ángulo de 90° hacia la izquierda.
<b>apagate</b>	<b>turnoff()</b>	Indica a Karel que detenga el funcionamiento de sus sensores y su brazo mecánico para finalizar la actividad.

Tabla 4- Instrucciones de Karel el Robot

Karel permite a los alumnos definir condiciones, existe una lista predefinida de las condiciones que Karel puede identificar, las cuales se muestran en la tabla 4, estas devuelven un valor booleano y pueden auxiliarse de operaciones lógicas.

frente-libre	junto-a-zumbador	orientado-al-este
rente-bloqueado	no-junto-a-zumbador	orientado-al-oeste
izquierda-libre	algun-zumbador-en-la-mochila	no-orientado-al-norte
izquierda-bloqueada	ningun-zumbador-en-la-mochila	no-orientado-al-sur
derecha-libre	orientado-al-norte	no-orientado-al-este
derecha-bloqueada	orientado-al-sur	no-orientado-al-oeste

Tabla 5-Condiciones detectadas por Karel el robot en Pascal

frontIsClear	nextToABeeper	facingEast
frontIsBlocked	notNextToABeeper	facingWest
leftIsClear	anyBeepersInBeeperBag	notFacingNorth
leftIsBlocked	noBeepersInBeeperBag	notFacingSouth
rightIsClear	facingNorth	notFacingEast
rightIsBlocked	facingSouth	notFacingWest

Tabla 6-Condiciones detectadas por Karel el robot en Java

Ejemplo de un programa de Karel.

### **Amontonar zumbadores 9a OMI, Morelia 2004**

#### **Descripción**

Por su naturaleza, Karel es un robot muy ordenado y le gusta tener todos sus zumbadores acomodados en montones.

Tras un fin de semana de juegos, Karel tiene su cuarto lleno de zumbadores regados en columnas, por lo que ahora deberá juntar los zumbadores de cada columna y hacer un montón con ellos en la base de la misma.

#### **Problema**

Tu tarea consiste en escribir un programa que ayude a Karel a amontonar los zumbadores de cada columna en un montón en la base de la misma.

#### **Consideraciones**

- Inicialmente, Karel se encuentra en la posición (1,1) orientado hacia el norte.
- El cuarto de Karel se encuentra lleno de columnas de zumbadores como se muestra en el ejemplo.
- Las columnas de zumbadores pueden tener cualquier altura desde 1 hasta 100 zumbadores, y se suceden una a la otra sin dejar espacios.

- Al encontrar la primera columna vacía, Karel sabrá que ha llegado al final del cuarto.
- Karel no tiene zumbadores en la mochila.
- La cantidad de zumbadores en cada casilla es desconocida.
- No importa la posición ni la orientación final de Karel

### Ejemplo

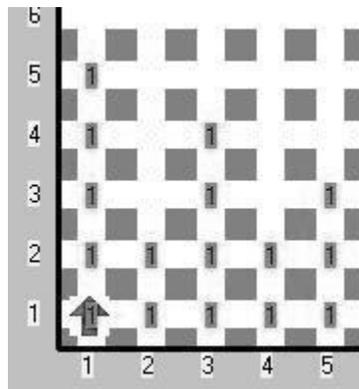


Imagen 4: Mundo de ejemplo

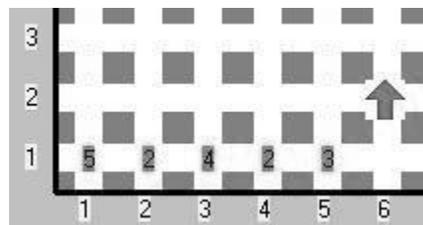


Imagen 5: Solución del mundo de ejemplo

### Agradecimiento

Se agradece al Comité Olímpico Mexicano de Informática el permiso para publicar este problema en nuestro sitio; que fue aplicado en el examen nacional de la 9a OMI, celebrada en la ciudad de Morelia, Michoacán en el año 2004.

## Solución en Java:

```
class program {
program() {

    while (nextToABeeper)
    {
        while(nextToABeeper)
        {
            while(nextToABeeper)
                pickbeeper ();
            if(frontIsClear)
                move ();
        }
        turnleft ();
        turnleft ();
        while(frontIsClear)
            move ();
        while(anyBeeperInBeeperBag)
            putbeeper ();

        turnleft ();

        if(frontIsClear)
        {
            move ();
            turnleft ();
        }
        else
        {
            turnoff ();
        }

    }
    turnoff ();
}
}
```

## Solucion en Pascal:

```
iniciar-programa
  inicia-ejecucion
    mientras junto-a-zumbador hacer inicio
      mientras frente-libre y junto-a-zumbador hacer avanza;
      mientras junto-a-zumbador hacer coge-zumbador;
      repetir 2 veces gira-izquierda;
      avanza;
      mientras frente-libre hacer inicio
        si junto-a-zumbador entonces inicio
          mientras junto-a-zumbador hacer coge-zumbador;
          fin;
          avanza;
      fin;
      mientras algun-zumbador-en-la-mochila hacer deja-zumbador;
      gira-izquierda;
      avanza;
      gira-izquierda;
    fin;
  apagate;
  termina-ejecucion
finalizar-programa
```

## Metodología.

Debido a que es posible priorizar cada módulo del sistema, se tiene planeado utilizar la metodología SCRUM, que es una metodología ágil de desarrollo. Esta metodología requiere trabajo duro y constante porque no se basa en el seguimiento de un plan, sino en la adaptación continua a las circunstancias de la evolución del proyecto. Las evoluciones son controladas por medio de un Sprint, el cual es un ciclo de desarrollo básico en el cual se debe realizar un incremento en la funcionalidad del sistema.

SCRUM se utiliza principalmente para proyectos web, debido a la constante adaptación de elementos que estos requieren y la versatilidad que presenta la metodología al momento de hacer integraciones de otros módulos. Se utiliza para resolver situaciones en que no se está entregando al cliente lo que necesita, cuando las entregas se alargan demasiado, los costes se disparan o la calidad no es aceptable, cuando es necesario identificar y solucionar ineficiencias sistemáticamente o cuando se quiere trabajar utilizando un proceso especializado en el desarrollo de producto.

La duración de cada Sprint se ha fijado en 30 días, ya que se considera el tiempo adecuado para poder volver a generar especificaciones.

Considerando un equipo de 3 integrantes, se decide separar el sistema completo en módulos fáciles de integrar. La integración de cada módulo se obtendrá a partir del análisis y diseño de todo el sistema, para así crear una arquitectura de fácil integración.

Cada Sprint considera una semana para realizar el análisis y correcciones necesarias para conseguir el objetivo del Sprint. Se realizan pruebas del código generado durante el Sprint y la integración con otros módulos existentes. Una vez implementados todos los módulos se llevan a cabo las pruebas generales del sistema.

### Patrón Modelo Vista Controlador

Es un patrón de diseño de software verdaderamente probado que convierte una aplicación en un paquete modular fácil de mantener, y mejora la rapidez del desarrollo.

Se utilizará el patrón Modelo Vista Controlador apoyándonos en el framework ASP.NET MVC 4 para poder tener un control de los datos que los usuarios ven y separar la lógica del negocio de las interfaces. La separación de las tareas de tu aplicación en modelos, vistas y controladores hace que su aplicación sea además muy fácil de entender. Las nuevas características se añaden fácilmente, y agregar cosas nuevas a código viejo se hace muy sencillo. El diseño modular también permite a los desarrolladores y los diseñadores trabajar simultáneamente, incluyendo la capacidad de hacer prototipos rápidos. La separación también permite a los desarrolladores hacer cambios en una parte de la aplicación sin afectar a los demás.

El modelo representa la parte de la aplicación que implementa la lógica de negocio. Esto significa que es responsable de la recuperación de datos, convirtiéndolo en conceptos significativos para la aplicación, así como su procesamiento, validación, asociación y cualquier otra tarea relativa a la manipulación de dichos datos.

La vista hace una presentación de los datos del modelo. Estando separada de los objetos del modelo, es responsable del uso de la información de la cual dispone para producir cualquier interfaz de presentación de cualquier petición que se presente.

La capa del controlador gestiona las peticiones de los usuarios. Es responsable de responder la información solicitada con la ayuda tanto del modelo como de la vista. Los controladores pueden ser vistos como administradores, cuidando de que todos los recursos necesarios para completar una tarea se deleguen a los trabajadores más adecuados. Espera peticiones de los clientes, comprueba su validez de acuerdo a las normas de autenticación o autorización, delega la búsqueda de datos al modelo, y selecciona la tipo de respuesta más adecuado según las preferencias del cliente, finalmente, delega este proceso de presentación a la capa de la Vista.

## Análisis de factibilidad.

### Factibilidad operativa.

El análisis de factibilidad operativa del proyecto se ha realizado con base a la técnica algorítmica conocida como: Modelo COCOMO II<sup>4</sup>, el cual permite hacer una estimación de las personas necesarias para el proyecto con base a las interfaces de usuario, mensajes del sistema y funciones a programar, estos elementos son denominados "Puntos Objeto" y con base a ellos se realiza la estimación.

La fórmula para realizar la estimación es la siguiente:

$$PM = \frac{NOP}{PROD} \quad \text{donde: } PM = \frac{\text{Persona}}{\text{Mes}}$$

Donde:

**NOP** (Nuevos puntos objeto): Tamaño del nuevo software a desarrollar expresado en OP (Puntos Objeto) y se calcula de la siguiente manera:

$$NOP = OP * (100 - \%reuso)/100)$$

**OP** (Puntos Objeto): Componentes del software a desarrollar a cada uno se le asigna un peso, este fue determinado con base a tablas predefinidas en el modelo de Boehm<sup>5</sup>

**% reuso**: Porcentaje de reuso que se espera lograr en el proyecto, el modelo sugiere tomar este en un rango de 35 a 50%, ya que según sus estadísticas son los valores más comunes de reuso en un proyecto de tecnologías web.

**PROD**: Ratio de productividad, definen la capacidad o experiencia de los desarrolladores así como la madurez ICASE<sup>6</sup>. Se definen su valores en la Tabla 5.

Tipo de Objeto	Peso de la complejidad		
	Básico	Intermedio	Avanzado
<b>Pantalla</b>	2	4	6
<b>Informe</b>	4	6	8
<b>Programación</b>	0	8	10

<sup>4</sup> Constructive Cost Model: Permite realizar estimaciones en función del tamaño del software, y de un conjunto de factores de costo y de escala

<sup>5</sup> Modelo de calidad basado en características como la usabilidad, mantenibilidad y confiabilidad del sistema

<sup>6</sup> Herramientas que engloban todo el proceso de desarrollo software, desde análisis hasta implementación.

Tabla 7-Pesos de los Puntos Objeto con base al modelo de Boehm

Capacidad y experiencia de los desarrolladores	Muy baja	Baja	Nominal	Alta	Muy alta
<b>Capacidad y madurez ICASE</b>	Muy baja	Baja	Nominal	Alta	Muy alta
<b>Ratio de productividad</b>	4	7	13	25	50

Tabla 8-Valores del ratio de productividad

Se realiza entonces una estimación de los puntos objeto del sistema en todas las complejidades y se toman los pesos asignados a cada punto objeto para obtener el total de puntos objeto del sistema, la estimación se muestra en la Tabla 6.

Tipo de objeto	Básico		Intermedio		Avanzado		Total
	Cantidad	Peso	Cantidad	Peso	Cantidad	Peso	
<b>Pantalla</b>	1	2	3	12	1	6	17
<b>Informe</b>	1	4	1	6	1	8	18
<b>Programación</b>			2	16	4	20	36
<b>Total</b>	2	6	6	34	6	34	71

Tabla 9-Análisis de tipos objeto con base a cantidad y peso

Como se puede observar la estimación arroja un total de 71 puntos objeto, de modo que al sustituir los datos en la fórmula para determinar los nuevos puntos objeto obtenemos lo siguiente:

$$NOP = 71 * \frac{100 - 50}{100} = 35.5$$

Con dicho dato obtenemos que el aproximado de personas mes necesarias para que el proyecto sea factible al sustituirlo en la fórmula inicial, se tomó un ratio de productividad normal para todos los desarrolladores debido a que es lo que modelo sugiere.

$$PM = \frac{35.5}{13} = 2.73$$

Como dato final obtenemos que para realizar el proyecto se necesita de aproximadamente 3 personas por mes, en dicho cálculo se ha considerado la programación como parte de un objeto del sistema. Se trabajará siguiendo un ciclo de desarrollo concurrente, es decir que cada persona realizará una actividad que permita la construcción y modificación del sistema en forma continua a través de la entrega de elementos funcionales que se irán entregando una vez validados y probados de forma individual. Se realizó una separación de actividades con base a los perfiles del desarrollador, dicha separación se presenta a continuación:

Elemento	Actividad
<b>Persona 1</b>	Implementación del módulo de gestión de usuarios Implementación del módulo gestión de concursos.
<b>Persona 2</b>	Implementación del evaluador de soluciones en Karel. Implementación del clasificador de problemas.
<b>Persona 3</b>	Desarrollo e implementación de la interfaz gráfica Implementación de canales de redes sociales a la interfaz gráfica

Tabla 10- División de actividades

Las actividades como la planeación, análisis y diseño han sido consideradas pero no se realiza la división por persona dado que serán realizadas de forma concurrente.

Cada actividad tiene un objetivo y requiere de un perfil de usuario específico como se puede observar en la tabla 11.

Actividad	Descripción	Perfil del desarrollador
<b>Planeación</b>	Realizar las tareas necesarias para definir tiempos, recursos y la planificación del proyecto.	<ul style="list-style-type: none"> <li>- Habilidad para la elaboración y control de planes</li> </ul>
<b>Análisis</b>	Actividades relacionadas con la comprensión y solución del problema, definir funciones del software y sus medios de interacción.	<ul style="list-style-type: none"> <li>- Conocimientos de ingeniería de software</li> <li>- Comprender el funcionamiento del software con base a rendimiento y operación</li> <li>- Criterio en la toma de decisiones</li> </ul>
<b>Diseño</b>	Proceso centrado a la consolidación de los elementos obtenidos en el análisis, basado en la arquitectura del sistema, interfaces y desarrollo procedimental del sistema.	<ul style="list-style-type: none"> <li>- Conocimientos de lenguaje UML</li> <li>- Conocimientos de arquitectura de software</li> <li>- Conocimientos de Interacción Humano-Computadora y Usabilidad para el diseño de interfaces</li> <li>- Conocimientos de bases de datos</li> </ul>
<b>Implementación del módulo de gestión de usuarios</b>	Módulo que permite realizar la administración de los usuarios del sistema, así como una asignación de permisos según su tipo.	<ul style="list-style-type: none"> <li>- Conocimiento de tecnologías de .NET</li> <li>- Conocimiento de bases de datos</li> <li>- Conocimiento en seguridad</li> </ul>
<b>Implementación del módulo de gestión de concursos</b>	Módulo que permite administrar los concursos y habilitarlos en determinado periodo de tiempo, así como mostrar los resultados al término de estos.	<ul style="list-style-type: none"> <li>- Conocimiento de tecnologías de .NET</li> <li>- Conocimiento de bases de datos</li> <li>- Manejo de sistemas distribuidos</li> </ul>
<b>Implementación del evaluador de soluciones en Karel</b>	Módulo que permite evaluar los problemas en lenguaje Karel ya sea en tiempo real o dejándolos en espera.	<ul style="list-style-type: none"> <li>- Conocimiento de tecnologías de .NET</li> <li>- Conocimiento de bases de datos</li> </ul>
<b>Implementación del clasificador de problemas</b>	Módulo que permite clasificar un problema con base al número de usuarios que lo han resuelto y las técnicas de programación necesarias para resolverlo.	<ul style="list-style-type: none"> <li>- Conocimiento de tecnologías de .NET</li> <li>- Conocimiento de bases de datos</li> <li>- Estructuras de datos</li> </ul>
<b>Desarrollo e implementación de la interfaz gráfica</b>	Módulo que manipulara todas las interfaces del sistema con el fin de homogenizarlas.	<ul style="list-style-type: none"> <li>- Manejo de HTML con hoja de estilos CSS.</li> </ul>
<b>Implementación de canales de redes sociales a la interfaz gráfica</b>	Módulo que permitirá la vinculación con distintas redes sociales	<ul style="list-style-type: none"> <li>- Conocimiento de tecnologías de .NET</li> <li>- Conocimiento de la API de Facebook</li> <li>- Conocimiento de la API de Twitter</li> </ul>

Tabla 11- Perfiles

### Factibilidad técnica.

Se realizó un análisis de las tecnologías web más usadas para la realización de sitios similares a la propuesta, se hizo un comparativo de las ventajas y desventajas de cada uno de las tecnologías que operan del lado del cliente para ayudar a elegir la que ofreciera mayores ventajas.

## Tecnologías que operan del lado del cliente

Tecnología	Características	Ventajas	Desventajas	Precio
<b>HTML 5</b>	Lenguaje de marcado de hipertexto para la creación de sitios web. Lenguaje descriptivo. Lenguaje estructurado.	Es el lenguaje de marcado en desarrollo reconocido por la mayoría de los navegadores así como dispositivos móviles. Cuenta con editores web que facilitan el código. Permite colocar archivos multimedia de manera más sencilla. Implementación de etiquetas que permiten mejorar la distribución de contenidos. Mejoras en los formularios. Nuevos tipos de datos (eMail, number, url, datetime ...). Facilidades para validar el contenido sin necesidad de Javascript. Despliegue rápido de contenido ya que el código es muy simple. Diseños adaptables a diferentes dispositivos. Permite múltiples Javascripts corriendo en paralelo en una misma página.	HTML es solo un lenguaje de marcado que da una apariencia lineal, necesita ser complementado con otras tecnologías para dar funcionalidad y diseño. La visualización puede ser distinta para cada navegador. Algunos símbolos pueden visualizarse o no, dependiendo de la codificación.	Libre
<b>JavaScript</b>	Lenguaje de scripting ejecutado en el lado del cliente. Lenguaje de programación interpretado, orientado a objetos y prototipos, imperativo, débilmente tipado y dinámico. Existe una variante para lenguajes de servidor.	Lenguaje de scripting sencillo y fiable. Los scripts tienen capacidades limitadas por seguridad al ser un lenguaje que se ejecuta del lado del cliente. Propiedades y valores pueden ser creados, modificados y eliminados en tiempo de ejecución. Puede ser evaluado en tiempo de ejecución. Soporta el uso de expresiones regulares. Biblioteca Modernizr.	Código visible para el usuario. Depende del entorno para poder ser ejecutado el script. El código debe descargarse completamente. Se puede ejecutar código malicioso del lado del cliente que permita explotar la vulnerabilidad de seguridad en las aplicaciones, navegadores o el sistema operativo.	Libre

Tabla 12-Tecnologías que operan del lado del cliente

## Tecnologías que operan del lado del servidor

Tecnología	Características	Ventajas	Desventajas	Precio
<b>JAVA Server Pages</b>	Lenguaje utilizado para la elaboración de sitios dinámicos a través del lenguaje Java. Implementa el paradigma orientado a objetos. Cuenta con las características de Java. Es un lenguaje multiplataforma. Generación de contenido dinámico.	Tiene un buen desempeño y es eficiente dado a que se compila. Es apto para la creación de clases que manejen lógica de negocios. Presenta la misma portabilidad de Java. El manejo de sesiones y la conexión a base de datos es más eficiente. Es multiplataforma ya que se implementa en una máquina virtual. Genera el HTML al ejecutarse. Integridad con los módulos de Java. Permite separar la parte dinámica de la estática en las páginas web.	Para ejecutar, es necesario contar con un servidor compatible con contenedores servlet. Tiempo de carga de interfaz es superior que a la de una aplicación web. Interfaz web limitada.	Libre
<b>PHP</b>	Orientado a aplicaciones web que interactúan con bases de datos. Seguridad y confiabilidad al ser lenguaje de servidor. Conectividad con MySQL y PostgreSQL. Permite implementar técnicas POO. Manejo de excepciones.	Similar al lenguaje C. Se puede utilizar directamente en el código fuente del HTML sin necesidad de llamar a un archivo externo. Compatibilidad con la mayoría de los servidores y sistemas operativos. Compatibilidad con el protocolo HTTP. Crear funciones grandes en pocas líneas de código. Calculo de operaciones matemáticas mucho más rápido que ASP.	Puede presentar ineficiencia si el número de peticiones es elevado. No existe una separación por capas lo que puede hacer el código ilegible.	Software Libre
<b>ASP .NET 4.5</b>	Entorno de construcción, desarrollo y ejecución de servicios web así como otras aplicaciones. Es una plataforma de ejecución intermedia mediante una máquina virtual. Basada en el paradigma de programación orientado a objetos. Cuenta con soporte multilinguaje.	Soporta aplicaciones grandes y complejas. Compatibilidad con el patrón MVC. Controles de usuario y personalizados. Respuestas rápidas de servidor. Ofrece un buen nivel de seguridad. Procesamiento rápido de peticiones. Compilación del código lo que proporciona un rendimiento mayor. Creación de formularios sin necesidad de Web Forms. Controles de usuario y personalizados. Compatibilidad con los nuevos tipos de formulario HTML5.	Funciona solo con el sistema operativo Windows. Consumo elevado de recursos. Mantenimiento elevado si se desarrolla en varios lenguajes.	Comercial

Tabla 13-Tecnologías que operan del lado del servidor

## Gestores de base de datos

Gestor	Características	Ventajas	Desventajas	Precio
<b>SQL Server</b>	<p>Basado en el modelo relacional.</p> <p>Soporte de transacciones.</p> <p>Soporte de procedimientos almacenados.</p> <p>Permite el uso de comandos DDL y DML gráficamente.</p> <p>Permite trabajar de manera cliente-servidor.</p> <p>Permite la administración de servidores.</p>	<p>Proporciona escalabilidad, estabilidad y seguridad.</p> <p>Modo de uso cliente-servidor es demasiado útil.</p> <p>Mejor utilización del CPU.</p> <p>Soporte de la interfaz de acceso OLE DB y ADO.</p>	<p>Funciona simplemente en plataforma Windows.</p> <p>Memoria limitada en sistema de 32 bits.</p> <p>La compresión de datos solo está disponible en la versión Enterprise.</p>	Comercial
<b>MySQL</b>	<p>Basado en el modelo relacional.</p> <p>Amplio subconjunto del lenguaje SQL.</p> <p>Disponibilidad multiplataforma.</p> <p>Posibilidad de selección de mecanismos de almacenamiento.</p> <p>Cuenta con transacciones y llaves foráneas.</p> <p>Ofrece seguridad en la conectividad.</p> <p>Permite la búsqueda e indexación en los campos de textos.</p> <p>Soporta escalabilidad.</p>	<p>Funciona bien en complemento con PHP.</p> <p>Multiplataforma.</p> <p>Joins muy rápidos usando un multi-join de un paso optimizado.</p> <p>Velocidad al realizar operaciones lo que le hace tener un buen rendimiento.</p> <p>Bajo costo en requerimientos.</p> <p>Baja probabilidad de corromper datos.</p>	<p>Problemas de integridad en entornos de alta concurrencia.</p> <p>No es intuitivo.</p>	Open Source

Tabla 14-Gestor de base de datos

## Otras tecnologías

Tecnología	Características	Ventajas	Desventajas	Precio
<b>CSS3</b>	Lenguaje de marcado para establecer el diseño de un documento HTML. Soporte de una alta gama de colores.	Permite dar diseño a un documento HTML sin necesidad de modificar todo el documento. Nuevas propiedades para el aspecto gráfico. Permite lograr efectos sin uso de otros programas o editores gráficos. Puede establecer distintos estilos para diferentes dispositivos. Efectos de transición sin necesidad de JavaScript.	Puede darse el caso que la visualización varíe dependiendo del navegador.	Libre
<b>JQuery</b>	Eventos. Presenta animación y efectos. Permite manipular CSS. Soporte de extensiones.	Permite obtener información del navegador. Compatibilidad con todos los navegadores. Permite crear páginas dinámicas. Permite agregar plugins fácilmente previamente desarrollados.	Constante actualizaciones hacen que siempre sea necesario actualizar.	Software Libre

Una vez realizado el análisis de tecnologías se optó por realizar el sistema en .NET, una de las razones que el compilador actual del lenguaje Karel está hecho en dicho lenguaje de modo que integrarlo facilitará la realización del sistema. Se listan a continuación otras ventajas de .NET

- Uso de Visual Studio como IDE, que cuenta con las mejores herramientas para el desarrollo de software.
- Permite el uso de varios lenguajes, sobre la plataforma de .NET
- La versión anterior está desarrollada en esta plataforma
- Cuenta con un framework fácil de utilizar para la implementación del patrón Modelo-Vista-Controlador

#### Requerimientos técnicos

##### **Requisitos de procesador**

1.6 GHz más rápido

##### **Requisitos de sistema operativo**

Windows 7 SP1 (x86 y x64), Windows 8 (x86 y x64), Windows Server 2008 R2 SP1 (x64), Windows Server 2012 (x64)

##### **Requisitos de memoria**

1 GB (1.5 GB si se ejecuta en una máquina virtual)

##### **Requisitos de espacio en disco duro**

10 GB, a 5400 RPM

##### **Requisitos de tarjeta de video**

Tarjeta de video capaz de ejecutar DirectX 9 y con una resolución de 1024 x 768 o mayor

Se cuenta con cuatro equipos que reúnen las características necesarias para instalar Visual Studio en su versión profesional, por lo que no es necesario realizar una inversión en equipo nuevo y el desarrollo del proyecto en la parte técnica es factible.

## Factibilidad económica.

El análisis de factibilidad económica fue realizado considerando que el presente proyecto es con fines educativos, sin embargo se ha determinado el costo con el que se distribuiría en el mercado si en determinado momento se decide utilizar el proyecto con fines de lucro.

En la tabla 15 se muestran los servicios necesarios para el desarrollo del trabajo terminal. Las especificaciones de los equipos son basadas en los requerimientos del entorno de desarrollo a utilizar y los precios son los correspondientes a un Sprint de trabajo, en la última columna se muestra el costo total al final de los 10 Sprints de trabajo arrojando un total de **\$25,371.90M.N.**

Insumos	Características	Precio	Total
<b>Servicio de Internet</b>	<ul style="list-style-type: none"> <li>• Conexión de 12Mbps</li> </ul>	\$399.00	\$3990.00
<b>Equipo de Cómputo</b>	<ul style="list-style-type: none"> <li>• Procesador a 1.6 GHz</li> <li>• Windows 7 SP1 (x86 y x64), Windows 8 (x86 y x64), Windows Server 2008 R2 SP1 (x64), Windows Server 2012 (x64)</li> <li>• Tarjeta de video capaz de ejecutar DirectX 9 y con una resolución de 1024 x 768 o mayor</li> </ul>	\$13037.00 \$10350.00	\$23407.00
<b>Electricidad</b>	<ul style="list-style-type: none"> <li>• Servicio 1 A para localidades con temperatura media mínima en verano de 25 grados centígrados</li> </ul>	\$415.98	\$2079.90
	•		<b>\$25,371.90M.N.</b>

Tabla 15-Servicios y Equipo

La tabla 16 muestra el software que se utilizara en el desarrollo, el cual fue seleccionado con anterioridad, se incluye el costo de las licencias de uso con un total de **\$13,184.00 M.N.**

Plataforma	Precio
<b>Visual Studio Professional 2012</b>	\$10585.00
<b>Microsoft Office 2010</b>	\$2599.00
<b>Skydrive</b>	- NA

Tabla 16-Software para el desarrollo del proyecto

Se evaluaron los sueldos de un programador en la tecnología .NET de distintos sitios con experiencia en el desarrollo de este tipo de aplicaciones y se obtuvo un promedio para determinar el sueldo de los desarrolladores, considerando que no se contará con prestaciones de ningún tipo y que el tipo de contrato sería por proyecto los sueldos se determinan en la siguiente tabla:

Ocupación	Sueldo	Total
Analista Programador .NET	<b>\$28,000.00</b>	<b>\$84,000.00</b>

Tabla 17- Sueldos

Costo total del proyecto:

Concepto	Costo
<b>Insumos</b>	\$25,371.90 M.N.
<b>Software</b>	\$13,184.00 M.N.
<b>Sueldos</b>	\$84,000.00 M.N.
<b>Total:</b>	<b>\$122,530.90 M.N.</b>

Tabla 18-Costo Total del Proyecto

El trabajo terminal es factible económicamente considerando que no se comprará ninguna licencia ni equipo adicional al que ya se cuenta. Los costos de electricidad e internet serán absorbidos por la Escuela Superior de Cómputo ya que el trabajo será desarrollado dentro de sus instalaciones.

## Análisis de Riesgos

Determinados las funcionalidades del sistema y las tecnologías a emplear se llevó a cabo un análisis de las posibles amenazas o eventos no deseados y el impacto que éstos podrían tener en el sistema.

A cada uno de los riesgos determinados se les asigno una probabilidad.

- Baja: Riesgo que puede no ocurrir pero es necesario considerar
- Media: Riesgo que puede o no ocurrir pero debe considerarse
- Alta: Riesgo que ocurrirá y se debe tener presente siempre

También se les asigno un impacto con base a las repercusiones que tendría en la culminación del proyecto.

- Insignificante: No afecta en la culminación del proyecto
- Tolerable: Afecta al proyecto, pero éste puede terminarse.
- Serio: Afecta al proyecto de forma que no podría terminarse si ocurriera.
- Grave: Afecta directamente la culminación del proyecto.

### Riesgos Tecnológicos.

Este tipo de riesgo tiene que ver con las tecnologías elegidas y con el hardware necesario para el desarrollo del sistema.

Nombre	Descripción	Probabilidad	Impacto
<b>Tecnología errónea</b>	La elección de las tecnologías a emplear no ayuda a cumplir con los requerimientos del sistema	Media	Tolerable
<b>Fallas de hardware</b>	El hardware con el que se llevara a cabo el desarrollo tiene fallas en su funcionamiento.	Baja	Serio
<b>Pérdida de versionamiento</b>	Pérdida de la última versión del proyecto	Baja	Insignificante

Tabla 19-Riesgos Tecnológicos

## Riesgos Organizacionales.

Este tipo de riesgos está ligado al equipo de trabajo y la forma en la que éste se vincule con el proyecto.

Nombre	Descripción	Probabilidad	Impacto
<b>Desintegración</b>	Desintegración del equipo ya sea por la salida de uno o varios integrantes.	Baja	Serio
<b>Falta de comunicación</b>	El equipo no logra tener la comunicación adecuada para llevar a cabo la planeación de actividades.	Media	Tolerable
<b>Distribución inadecuada</b>	Actividades a realizar mal distribuidas a cada miembro del equipo	Media	Tolerable

Tabla 20-

Riesgos Organizacionales

## Riesgos de estimación.

Riesgos ligados a las estimaciones realizadas de personal, costos y alcance del proyecto.

Nombre	Descripción	Probabilidad	Impacto
<b>Personas insuficientes</b>	Error al estimar el número de personas y no tener la capacidad de llevar a cabo el desarrollo del sistema en el tiempo definido	Media	Tolerable
<b>Alcance</b>	El alcance del proyecto no corresponde al tiempo de desarrollo con el que se cuenta	Baja	Tolerable

Tabla 21-

Riesgos de estimación

## Plan de contingencia

Se desarrollan actividades para supervisar que el riesgo no ocurra y en caso de ocurrir se formulan actividades a realizar para minimizar su impacto en el desarrollo del proyecto.

Riesgo	Supervisión	Contingencia
<b>Tecnología errónea</b>	<ul style="list-style-type: none"> <li>- Revisión constante de los requerimientos</li> <li>- Revisión de las actualizaciones tecnológicas</li> </ul>	<ul style="list-style-type: none"> <li>- Búsqueda de herramientas tecnológicas compatibles con las elegidas</li> </ul>
<b>Fallas de hardware</b>	<ul style="list-style-type: none"> <li>- Revisión del hardware</li> <li>- Contar con hardware de apoyo</li> </ul>	<ul style="list-style-type: none"> <li>- Reemplazo del hardware</li> </ul>
<b>Pérdida de versionamiento</b>	<ul style="list-style-type: none"> <li>- Mantener copias de seguridad</li> <li>- Mantener las versiones anteriores</li> </ul>	<ul style="list-style-type: none"> <li>- Análisis del impacto de la Pérdida</li> <li>- Darle prioridad a los avances</li> </ul>
<b>Desintegración</b>	<ul style="list-style-type: none"> <li>- Registro de actividades que realiza cada miembro del equipo</li> <li>- Monitoreo del cumplimiento de actividades asignadas</li> </ul>	<ul style="list-style-type: none"> <li>- Ajustar los requerimientos</li> <li>- Minimizar alcances</li> <li>- Replantear objetivos</li> </ul>
<b>Falta de comunicación</b>	<ul style="list-style-type: none"> <li>- Reuniones constantes</li> <li>- Asistir a las revisiones</li> <li>- Preguntar por los avances</li> </ul>	<ul style="list-style-type: none"> <li>- Establecer calendarios</li> </ul>
<b>Distribución inadecuada</b>	<ul style="list-style-type: none"> <li>- Establecer las actividades cada Sprint</li> <li>- Establecer responsabilidades a cada miembro del equipo</li> </ul>	<ul style="list-style-type: none"> <li>- Redefinir actividades</li> <li>- Reasignar responsabilidades</li> <li>- Replantear tiempos de entrega</li> </ul>
<b>Personas insuficientes</b>	<ul style="list-style-type: none"> <li>- Tener presente los objetivos</li> <li>- Evitar la sobre carga de actividades a una persona</li> </ul>	<ul style="list-style-type: none"> <li>- Replantear los alcances y requerimientos</li> <li>- Redefinir los módulos del proyecto</li> </ul>
<b>Alcance</b>	<ul style="list-style-type: none"> <li>- Establecer prioridades y tiempos de entrega</li> <li>- Mantener una visión del Sprint y el estado en el que se encuentra (inicio, revisión, corrección, termino)</li> </ul>	<ul style="list-style-type: none"> <li>- Restructurar los alcances y adaptarlos a los tiempos</li> <li>- Priorizar las actividades y cambiar el orden</li> </ul>

Tabla 22-Plan de contingencia

# TRABAJO TERMINAL 2013-A003

Sistema web para la evaluación automática de programas de Karel el Robot.

## MÓDULO DE GESTIÓN DE USUARIOS

### *DOCUMENTO DE REQUISITOS*

#### Resumen Ejecutivo

Para el proyecto se requiere un módulo de gestión de usuarios que sea capaz de administrar (realizar altas, bajas, cambios y consultas) a todos usuarios que harán uso del sistema. El módulo debe contar con una interfaz gráfica que esté disponible a través de internet, donde se podrán realizar las operaciones necesarias para la administración de los usuarios.

#### Metas y no metas

##### Metas

- Gestionar todos los usuarios de la plataforma.
- Relacionar usuarios de tipo asesor con usuarios de tipo alumno.
- Proveer de un mecanismo para comprobar que los usuarios son personas reales.
- Proveer de los mecanismos de seguridad necesarios para evitar que usuarios no autorizados modifiquen la información.

##### No metas

- Detectar por algún medio, que no sea validación de correo electrónico, que los usuarios registrados son personas reales.
- Compartir la información de los usuarios registrados a otros servicios o aplicaciones.

## Requerimientos funcionales

#	Requerimiento	Prioridad
1	Los usuarios podrán registrarse a través de un formulario.	P1
2	Los usuarios podrán actualizar su información personal en cualquier momento.	P1
3	Los usuarios podrán dar de baja su cuenta en cualquier momento.	P1
4	Únicamente los usuarios de tipo asesor podrán registrar usuarios de tipo competidor como sus alumnos.	P2
5	Los usuarios únicamente podrán tener solo un tipo, que puede ser Administrador, Asesor, Alumno o Invitado.	P1
6	Los usuarios deberán de confirmar su identidad para poder utilizar el sistema.	P2
7	Los usuarios deberán aceptar los términos y condiciones del sistema para poder registrarse en el sistema.	P3
8	Los usuarios podrán ver información pública de los demás competidores.	P2
9	Los usuarios que olviden su contraseña podrán recuperarla a través de un formulario para recuperar contraseñas.	P2
10	Los usuarios no podrán almacenar información personal sensible en el sistema.	P3

Tabla 23-Requerimientos funcionales módulo gestión de usuarios

## Requerimientos no funcionales

#	Requerimiento	Prioridad
1	Los usuarios deberán de confirmar su identidad proporcionando un correo electrónico valido, el cual deberán de confirmar.	P2
2	Establecer un mecanismo de seguridad que proteja la información de los usuarios de accesos no autorizados.	P1

Tabla 24-Requerimientos no funcionales módulo gestión de usuarios

## Dependencias

Modulo	Tipo de dependencia	Detalles
Interacción con redes sociales	Sucesor	Los usuarios podrán complementar su información personal con a través de información proveniente de redes sociales.

Tabla 25-Dependencias módulo gestión de usuarios

## Requisitos detallados del modulo

1. Los usuarios podrán registrarse a través de un formulario.

Una vez registrados se les solicitara a los usuarios que ingresen la siguiente información:

- Nombre
- Fecha de nacimiento
- Escuela/Institución
- Grado escolar
- Estado donde vive
- Tipo de cuenta, que puede ser Alumno, Asesor o Invitado.
- Correo electrónico

- Nombre de usuario
- Contraseña
- Género
- Fotografía

*4. Únicamente los usuarios de tipo asesor podrán registrar usuarios de tipo competidor como sus alumnos.*

Los usuarios de tipo asesor podrán tener asociados uno o más usuarios de tipo alumno, esto con el fin de llevar un control del progreso de sus alumnos.

*5. Los usuarios únicamente podrán tener solo un tipo, que puede ser Administrador, Asesor, Alumno o Invitado.*

Los usuarios del sistema deberán pertenecer a solo una categoría, que puede ser Administrador, Asesor, Alumno o Invitado. Los usuarios de tipo Administrador podrán modificar los datos del resto de los usuarios e inhabilitar cuentas.

*6. Los usuarios deberán de confirmar su identidad para poder utilizar el sistema.*

Los usuarios deberán proporcionar un correo electrónico válido, para poder confirmar su identidad en el sistema. Al registrarse en el sistema, el usuario recibirá un correo con un vínculo para poder confirmar la cuenta. El usuario tendrá 24 horas a partir del momento de registro para llevar a cabo esta operación. Únicamente se podrá asociar una cuenta de correo electrónico por usuario.

*9. Los usuarios que olviden su contraseña podrán recuperarla a través de un formulario para recuperar contraseñas.*

Los usuarios que no recuerden los datos para poder iniciar sesión en el sistema, podrán recuperarlos proporcionando la dirección de correo electrónico que ingresaron al momento de registrarse. Al realizar este proceso, los usuarios recibirán un correo electrónico donde conocerán su nombre de usuario y podrán indicar una nueva contraseña.

*10. Los usuarios no podrán almacenar información personal sensible en el sistema.*

Se considera información personal sensible:

- Dirección completa de los usuarios
- Nivel socioeconómico
- Religión
- Raza
- Preferencia sexual
- Teléfono

## Riesgos y problemas detectados

1. Registro de la misma persona más de 1 vez en el sistema.
2. Registro de información falsa o inapropiada en el sistema.

## Ediciones al documento

Fecha	Autor	Cambios
18/8/2013	Juan Carlos	Creación del borrador del documento
25/8/2013	Juan Carlos	Redacción de RF.
18/9/2013	Juan Carlos	Especificación de los RF
13/4/2014	Larissa	Títulos a las tablas

## Apéndice

### Definiciones y acrónimos

**CAPTCHA:** siglas de *Completely Automated Public Turing test to tell Computers and Humans Apart* (Prueba de Turing pública y automática para diferenciar máquinas y humanos). Este test es controlado por una máquina, en lugar de por un humano como en la Prueba de Turing, por ello consiste en una prueba de Turing inversa [9]

# TRABAJO TERMINAL 2013-A003

## Sistema web para la evaluación automática de programas de Karel el Robot.

### MÓDULO DE GESTIÓN DE CONCURSOS

#### DOCUMENTO DE REQUISITOS

##### Resumen Ejecutivo

Para el proyecto se requiere un módulo de gestión de concursos que sea capaz de administrar (realizar altas, bajas, cambios y consultas) los concursos que serán hospedados en el sistema. Los concursos constan de un conjunto de problemas y una duración determinada. En los concursos podrán participar los usuarios de tipo alumno y asesor que cumplan con los criterios establecidos al momento de crear el concurso.

##### Metas y no metas

###### Metas

- Proveer de un sistema capaz de hospedar concursos.
- Proveer, en caso de ser necesario, un sistema de evaluación en tiempo real durante la duración del concurso.
- Personalizar concursos, para que se adapten a las necesidades de los usuarios.

###### No metas

- Proporcionar los medios para la evaluación de problemas de manera manual

##### Requerimientos funcionales

#	Requerimiento	Prioridad
1	Los usuarios podrán crear concursos en el sistema, con problemas previamente cargados en el sistema.	P1
2	Los usuarios que creen concursos podrán configurar las características de tiempo, problemas, y usuarios que podrán registrarse en el concurso.	P1
3	Los usuarios que cumplan con los requisitos establecidos al momento de configurar el concurso podrán registrarse.	P1
4	El sistema podrá evaluar los problemas en tiempo real.	P1
5	El sistema podrá desplegar los resultados del concurso en tiempo real.	P1
6	El sistema podrá almacenar de manera permanente los resultados de todos los concursos.	P2

Tabla 26-Requerimientos funcionales módulo gestión de concursos



## Requerimientos no funcionales

#	Requerimiento	Prioridad
1	El sistema deberá soportar al menos, un número de conexiones simultaneas igual a la suma de participantes en los concursos activos en un momento dado.	P1
2	El sistema deberá de contar con los mecanismos de seguridad necesarios para evitar alteraciones no autorizadas a los concursos, principalmente, cuando estos están en progreso.	P1

Tabla 27-Requerimientos no funcionales módulo gestión de concursos

## Dependencias

Modulo	Tipo de dependencia	Detalles
Evaluador	Predecesor	Se requiere de evaluador para poder utilizar este módulo.
Gestión de usuarios	Predecesor	Se requiere de usuarios que puedan ingresar al sistema

Tabla 28-Dependencias módulo gestión de concursos

## Requisitos detallados del módulo

1. Los usuarios podrán crear concursos en el sistema, con problemas previamente cargados en el sistema.

Los usuarios de tipo administrador y asesor podrán hospedar concursos en el sistema, con problemas previamente cargados en él.

2. Los usuarios que creen concursos podrán configurar las características de tiempo, problemas, y usuarios que podrán registrarse en el concurso.

Los usuarios que hospeden concursos en el sistema, podrán configurar las siguientes opciones:

- a. Fecha y hora de inicio del concurso
- b. Duración en minutos del concurso
- c. Listado de problemas que conformara el concurso
- d. Los usuarios que podrán participar, por:
  - a. Grado escolar
  - b. Estado
  - c. Asesor asignado
- e. Cuanto tiempo antes de finalizar el concurso se congelara la tabla de resultados
- f. Si se contarán penalizaciones por envíos fallidos
- g. Si el tiempo es un criterio de desempate

- h. Si los problemas tendrán una evaluación de 0 a 100, o únicamente Correcto e Incorrecto.

*3. Los usuarios que cumplan con los requisitos establecidos al momento de configurar el concurso podrán registrarse.*

Los usuarios tendrán que confirmar su participación en los concursos, opción que será habilitada siempre y cuando el usuario cumpla con los requisitos que el creador del concurso defina.

#### Riesgos y problemas detectados

1. Soportar una gran cantidad de conexiones simultaneas

#### Ediciones al documento

Fecha	Autor	Cambios
17/8/2013	Juan Carlos	Creación del borrador del documento
17/9/2013	Juan Carlos	Definición de RF
05/4/2014	Larissa	Cambios al formato del documento
13/4/2013	Larissa	Rótulos a las tablas

#### Apéndice

##### Definiciones y acrónimos

Tabla de resultados es aquella donde se despliega la información de los usuarios que participan en un concurso, donde se puede observar:

- Nombre del usuario
- Numero de problemas resueltos
- Tiempo que tardo en resolverlos
- Numero de intentos fallidos por problema

# TRABAJO TERMINAL 2013-A003

Sistema web para la evaluación automática de programas de Karel el Robot.

## MÓDULO DE INTERACCIÓN CON REDES SOCIALES

### *DOCUMENTO DE REQUISITOS*

#### Resumen Ejecutivo

Para el proyecto se requiere un módulo de interacción con redes sociales que sea capaz de llevar a cabo el inicio de sesión mediante alguna red social, ya sea Facebook o Twitter, así como permitir publicar los resultados en alguna red social (Twitter, Facebook) de los problemas que han sido resueltos con éxito (cuando una solución a un problema es correcta, el sistema debe permitir realizar la publicación), así como de los concursos que serán hospedados en el sistema.

#### Metas y no metas

##### *Metas*

- Permitir al usuario acceder al sistema mediante su perfil de alguna red social.
- Permitir al usuario publicar la resolución de un problema en sus redes sociales.
- Generar un sitio web más dinámico para la experiencia del usuario mediante la interacción del sistema con las redes sociales.
- Dar a conocer el sistema a otros usuarios que utilizan las redes sociales.

##### *No metas*

- Detectar SPAM

## Requerimientos funcionales

#	Requerimiento	Prioridad
1	Los usuarios podrán iniciar sesión mediante alguna red social.	P1
2	Los usuarios podrán publicar la resolución de un problema en alguna red social, siempre y cuando la evaluación de éste sea correcta.	P1
3	Los usuarios podrán elegir en que red social será publicado.	P2
4	Los usuarios deben permitir establecer conexión con la red social.	P2
5	Los concursos podrán ser compartidos en alguna red social.	P1

Tabla 29-Requerimientos funcionales módulo interacción con redes sociales

## Requerimientos no funcionales

#	Requerimiento	Prioridad
1	El sistema deberá realizar la menor cantidad de peticiones a servidores externos, con la finalidad de no impactar el rendimiento del sistema.	P3

Tabla 30-Requerimientos no funcionales módulo interacción con redes sociales

## Dependencias

Modulo	Tipo de dependencia	Detalles
Gestión de usuarios	Predecesor	Requiere de la creación de un perfil en el sistema.
Evaluador de soluciones en Karel	Predecesor	Requiere una solución correcta del problema.
Clasificación de problemas	Predecesor	Requiere la solución de un problema de las diferentes listas de problemas.
Gestión de concursos	Predecesor	Requiere la solución de un problema que pertenece al concurso.

Tabla 31- Dependencias módulo interacción con redes sociales

## Requisitos detallados del modulo

*1.Los usuarios podrán iniciar sesión mediante alguna red social.*

Las redes sociales permitidas para iniciar sesión en el sitio son Facebook y Twitter y el usuario podrá elegir mediante cual de ellas accederá, para esto será necesario que el usuario permita vincular el sistema a sus perfiles sociales.

*2.Los usuarios podrán publicar la resolución de un problema en alguna red social siempre y cuando la evaluación de éste sea correcta.*

El sitio permite llevar a cabo publicaciones en perfiles sociales de la resolución de un problema, para eso es necesario que el problema haya sido resuelto correctamente.

Si se elige Facebook la publicación será compartida desde la aplicación Karel que permitirá compartir el vínculo web del sistema así como el problema que se ha resuelto satisfactoriamente. Si se elige Twitter se creará un tweet con el nombre del problema que ha sido resuelto así como el vínculo web del sistema, el tweet será sometido a los límites de caracteres de dicha red social.

5. Los concursos podrán ser compartidos en alguna red social.

Los concursos alojados en el sistema, podrán ser compartidos mediante las redes sociales, con el fin de crear difusión tanto del concurso así como de Karel.

## Riesgos y problemas detectados

### Suplantación de identidades

1. Ingreso de datos falsos o inapropiados al sistema
2. Riesgo de SPAM generado por las redes sociales

### Ediciones al documento

Fecha	Autor	Cambios
22/8/2013	Miguel Ángel	Creación del borrador del documento
09/9/2013	Larissa	Detallar Requerimientos
19/9/2013	Larissa	Definiciones y acrónimos
05/4/2014	Larissa	Formato del documento
13/4/2014	Larissa	Rótulos a las tablas

## Apéndice

### Definiciones y acrónimos

**Facebook:** Herramienta social que conecta a la gente con sus amigos y otras personas que trabajan, estudian y viven en su entorno. [10]

**Perfil:** Página personalizada que el usuario configura a su manera y decide que es lo que otros usuarios pueden visualizar.

**SPAM:** Mensajes no solicitados, no deseados o de remitente no conocido [11]

**Red Social:** Servicios basados en Web que permiten: construir un perfil público o semi-público dentro de un sistema limitado; articular una lista de otros usuarios con los que comparten una conexión; y ver y recorrer su lista de las conexiones y las hechas por otros dentro del sistema [12]

**Twitter:** Red Social que permite enviar mensajes de texto plano de corta longitud, con un máximo de 140 caracteres, llamados tweets, que se muestran en la página principal del usuario. [13]

# TRABAJO TERMINAL 2013-A003

Sistema web para la evaluación automática de programas de Karel el Robot.

## MÓDULO DEL EVALUADOR

### DOCUMENTO DE REQUISITOS

#### Resumen Ejecutivo

Para el proyecto se requiere un módulo que pueda evaluar de manera inmediata las soluciones a los problemas, proporcionando los métodos necesarios para poder solicitar una evaluación y obtener los resultados de esta. Este módulo no contara con interfaz gráfica.

#### Metas y no metas

##### Metas

- Proveer de un mecanismo para la evaluación de problemas
- Obtener los resultados de la evaluación del problema
- Identificar la razón por la cual la solución no es correcta

##### No metas

- Identificar problemas en la lógica de programación

#### Requerimientos funcionales

#	Requerimiento	Prioridad
1	El sistema podrá evaluar, bajo petición del sistema, una solución a un problema.	P1
2	El evaluador podrá determinar si la solución es correcta, o cual es la razón por la cual no lo es.	P1
3	El evaluador podrá tener una cola de espera para evaluar soluciones.	P1

Tabla 32-Requirimientos funcionales módulo evaluador

#### Requerimientos no funcionales

#	Requerimiento	Prioridad
1	El evaluador deberá de determinar una respuesta en el menor tiempo posible, para evitar la acumulación de soluciones en la cola de espera.	P1

2	Establecer un mecanismo de verificación de los archivos de evaluación, para evitar la modificación de estos de manera no autorizada.	P2
---	--	----

Tabla 33-Requerimientos no funcionales módulo evaluador

### Dependencias

Este	Tipo de dependencia	Detalles
Concursos	Sucesor	Se requiere del evaluador para la evaluación de soluciones dentro de un concurso.
Clasificación de problemas	Sucesor	Se requiere del evaluador para la evaluación de soluciones dentro de práctica.

Tabla 34-Dependencias módulo evaluador

### Requisitos detallados del modulo

2. *El evaluador podrá determinar si la solución es correcta, o cual es la razón por la cual no lo es.*

El evaluador determinara una de las siguientes respuestas a las soluciones procesadas:

1. Correcto: Cuando la solución al problema supera la prueba de manera correcta.
2. Incorrecto: Cuando la solución al problema no supera la prueba de manera correcta.
3. Tiempo excedido: Cuando la solución al problema realiza demasiadas instrucciones.
4. Error desconocido: Cuando hubo un problema al evaluar la solución.

3. *El evaluador podrá tener una cola de espera para evaluar soluciones.*

El evaluador contara con una cola de espera para evaluar las soluciones una a una. El evaluador no reconocerá el origen de la solución, únicamente se encargara de determinar una respuesta.

*NF2 Establecer un mecanismo de verificación de los archivos de evaluación, para evitar la modificación de estos de manera no autorizada.*

El evaluador contara con un mecanismo de verificación de los casos de prueba, validando la autenticidad de los archivos a haciendo uso de una firma digital empleando una función hash.

## Riesgos y problemas detectados

1. Soportar una gran cantidad de conexiones simultaneas

## Ediciones al documento

Fecha	Autor	Cambios
17/8/2013	Juan Carlos	Creación del borrador del documento
24/9/2013	Juan Carlos	Definición de RF
2/11/2013	Juan Carlos	Actualización de RNF

## Apéndice

### Definiciones y acrónimos

**Solución:** Código escrito por un usuario del sistema con la intención de resolver un problema.

**Instrucción:** Una instrucción es una acción realizada por el robot, por ejemplo: Moverse, dar vuelta, etc.

### Lista de errores del compilador

#### Mensajes de la compilación

- "Comentario sin terminar.",
- "Cadena sin terminar.",
- "Se detectó un carácter inválido.",
- "Error en el encabezado del problema.",
- "Se esperaba un FINALIZAR-PROGRAMA al final del bloque del programa.",
- "Se encontraron instrucciones después del fin de programa.",
- "Palabra reservada encontrada como nombre de función o parámetro.",
- "Se esperaba un identificador.",
- "Se esperaba la palabra ""COMO"" o el parámetro de la función.",
- "Se esperaba """)"".",
- "Se esperaba la palabra ""COMO"".",
- "Se esperaba ; ",
- "El identificador especificado ya se está usando como nombre de otro procedimiento.",
- "Se esperaba un procedimiento o comando.",
- "Se esperaba ""( "".",
- "Se encontró un identificador no definido.",
- "Se esperaba una expresión entera.",
- "Se esperaba una función booleana.",
- "Se esperaba ""ENTONCES"".",

- "Se esperaba ""HACER"".",
- "Se esperaba ""VECES"".",
- "Se esperaba ""FIN"".",
- "Se esperaba un procedimiento.",
- "Procedimiento no definido.",
- "Se esperaba un parámetro en la llamada al procedimiento.",
- "Se esperaba ""INICIA-EJECUCION"".",
- "Se esperaba ""TERMINA-EJECUCION"".",
- "Error al definir la clase principal del programa.",
- "Se esperaba un identificador como nombre de la clase principal",
- "No se esperaba un parámetro en la llamada al procedimiento.",
- "Se esperaba {.",
- "Se esperaba }.",
- "Se esperaba método principal del programa.",
- "El método principal no puede llevar ""void"" o ""define"".",
- "Se encontró un ""else"" sin ""if"".",
- "Se encontró un ""sino"" sin ""si"".",
- "El identificador especificado ya se está usando como nombre de otro prototipo.",
- "La definición de la instrucción difiere a la de su prototipo." y
- "No se encontró la definición de la instrucción para el prototipo."

#### Mensajes de la ejecución

- "Karel quiso pasar a través de una pared.",
- "Karel quiso dejar un zumbador pero ya no tenía.",
- "Karel quiso recoger un zumbador pero no había."

#### Mensajes de evaluación

- "OK",
- "Se superó el máximo de líneas de código permitidas.",
- "Se superó el número máximo de instrucciones avanza (move).",
- "Se superó el número máximo de instrucciones gira-izquierda (turnleft).",
- "Se superó el número máximo de instrucciones coge-zumbador (pickbeeper).",
- "Se superó el número máximo de instrucciones deja-zumbador (putbeeper).",
- "Se superó el número máximo de zumbadores en la mochila.",
- "Se superó el número máximo de zumbadores en una posición.",
- "La posición de Karel es incorrecta.",
- "La orientación de Karel es incorrecta.",
- "El número de zumbadores en una posición es incorrecta."



# TRABAJO TERMINAL 2013-A003

Sistema web para la evaluación automática de programas de Karel el Robot.

## MODULO DEL CLASIFICADOR DE PROBLEMAS

### *DOCUMENTO DE REQUISITOS*

#### Resumen Ejecutivo

Para el proyecto se requiere un módulo que pueda almacenar de manera ordenada los problemas con los que el sistema cuenta. Estos problemas podrán ser desplegados de acuerdo a una serie de criterios definidos por el usuario. Además, este módulo será el encargado de recibir la solución de un usuario para determinar si es correcta o no, usando el evaluador.

#### Metas y no metas

##### Metas

- Proveer de una clasificación de problemas de acuerdo a diferentes factores como dificultad, tema, etc.
- Proveer de un sistema para agrupar problemas.
- Recibir soluciones a los problemas del sistema.

#### Requerimientos funcionales

#	Requerimiento	Prioridad
1	El sistema podrá clasificar problemas.	P1
2	Los usuarios podrán crear listas de problemas, que podrán ser públicas o privadas.	P1
3	Los usuarios de tipo asesor o administrador podrán subir nuevos problemas al sistema.	P1
4	Los usuarios podrán solicitar la evaluación de una solución a un problema en el sistema.	P1
5	El sistema podrá proporcionar información acerca del número de soluciones correctas a un problema, y el porcentaje de soluciones correctas.	P2

Tabla 35-Requerimientos funcionales módulo gestión de problemas

## Requerimientos no funcionales

#	Requerimiento	Prioridad
1	La interfaz gráfica deberá ser fácil de utilizar para los usuarios	P1

Tabla 36-Requerimientos no funcionales módulo gestión de problemas

## Dependencias

Este	Tipo de dependencia	Detalles
Evaluador	Predecesor	Se requiere del evaluador para verificar las soluciones a los problemas.
Gestión de usuarios	Predecesor	Se requiere del módulo de gestión de usuarios para poder desplegar información de acuerdo al usuario.

## Requisitos detallados del modulo

1. *El sistema podrá clasificar problemas.*

El sistema proporcionara los métodos para clasificar los problemas de acuerdo a los siguientes criterios:

- Número de usuarios que han intentado resolver el problema
- Número de usuarios que han resuelto el problema correctamente
- Porcentaje de usuarios que han resuelto el problema correctamente

2. *Los usuarios podrán crear listas de problemas, que podrán ser públicas o privadas.*

Todos los usuarios podrán crear listas personalizadas de problemas, que podrán ser públicas (otras personas puedan ver las listas que han creado y los problemas que las conforman) o privadas (únicamente el usuario que creo la lista puede verla).

3. *Los usuarios de tipo asesor o administrador podrán subir nuevos problemas al sistema.*

Los usuarios de tipo asesor o administrador podrán subir nuevos problemas al sistema. Al subir un nuevo problema, este podrá ser público o privado. Si el problema es público, se agregara al listado general de problemas, donde los demás usuarios podrán verlo e intentar resolverlo, si es privado, el problema únicamente podrá ser visto por el usuario

que lo subió y ser usado por el para algún concurso. Al momento de usarlo en algún concurso, el problema únicamente será visible para los participantes de este.

Para subir un problema se requiere:

1. Definir un nombre para el problema.
2. Proporcionar una descripción del problema, con ejemplos.
3. Definir los casos de prueba, a través de archivos en formato KEC de Karel.
4. Definir si el problema será público o privado.

*4. Los usuarios podrán solicitar la evaluación de una solución a un problema en el sistema.*

La pantalla donde se describe el problema contendrá una sección donde los usuarios podrán enviar una solución al sistema para solicitar su evaluación. Este deberá ser un archivo de texto en formato TXT con el código en Java o Pascal de la solución implementada por el usuario. El sistema deberá mostrar los resultados de la evaluación.

#### Riesgos y problemas detectados

1. Problemas de rendimiento al crear demasiadas listas.

#### Ediciones al documento

Fecha	Autor	Cambios
17/8/2013	Juan Carlos	Creación del borrador del documento
24/9/2013	Juan Carlos	Definición de RF
26/9/2013	Juan Carlos	Detalles de RF
2/11/2013	Juan Carlos	Actualización del RF 1
25/2/2014	Juan Carlos	Actualización del RF 1
05/4/2014	Larissa	Formato del documento

#### Apéndice

##### Definiciones y acrónimos

**Solución:** Código escrito por un usuario del sistema con la intención de resolver un problema.

# TRABAJO TERMINAL 2013-A003

Sistema web para la evaluación automática de programas de Karel el Robot.

## DOCUMENTO DE DIAGRAMAS DEL SISTEMA

### Diagrama de arquitectura

Los diagramas de arquitectura de software son útiles para modelar sistemas. Este diagrama por bloques divide el sistema en varios subsistemas de software, donde cada capa se basa en la anterior. En este esquema, la capa más alta utiliza servicios definidos por la inferior y la última es inconsciente de la superior. Además, normalmente cada capa oculta las capas inferiores de las superiores.

Es útil trabajar un sistema en capas, debido a que:

- Se puede entender una capa como un todo, sin considerar las otras.
- Las capas se pueden sustituir con implementaciones alternativas de los mismos servicios básicos
- Se minimizan dependencias entre capas.
- Las capas posibilitan la estandarización de servicios
- Luego de tener una capa construida, puede ser utilizada por muchos servicios de mayor nivel.

A continuación se muestra el diagrama de la arquitectura de nuestro sistema.

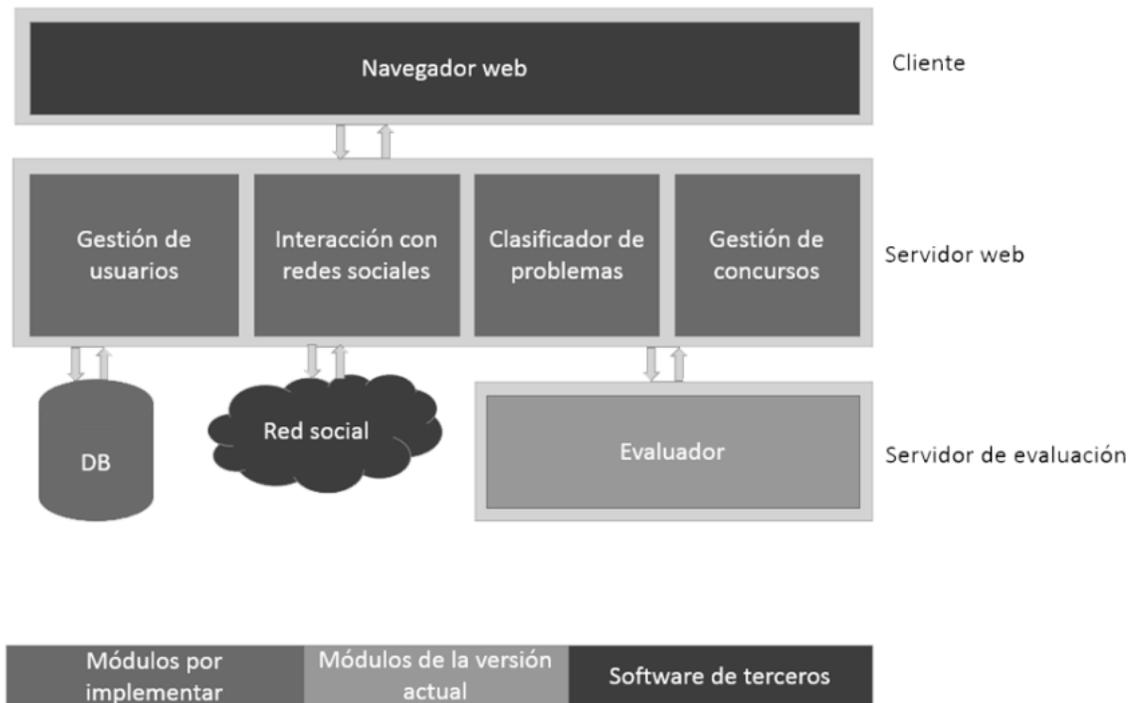


Imagen 6: Diagrama de Arquitectura

## Diagrama de base de datos.

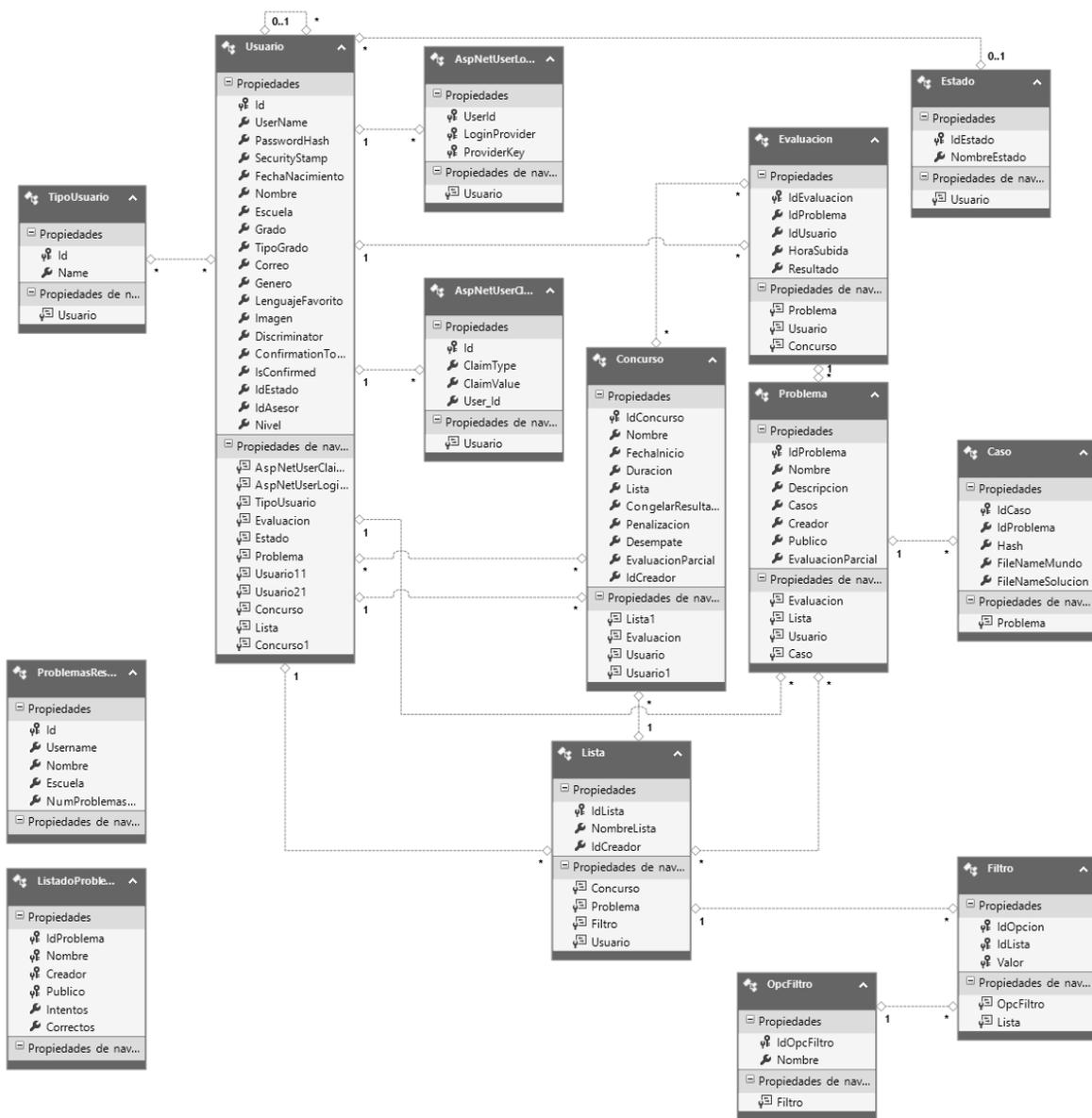


Imagen 7: Diagrama de base de datos

## Diagramas de casos de uso

Es la representación gráfica de cómo debería interactuar el sistema con el usuario o con otro sistema para lograr un objetivo específico. El caso de uso es la representación gráfica de cada proceso desde la perspectiva usuario, lo que se espera del sistema en cuanto a funcionalidad.

### Módulo Gestión de Usuarios.

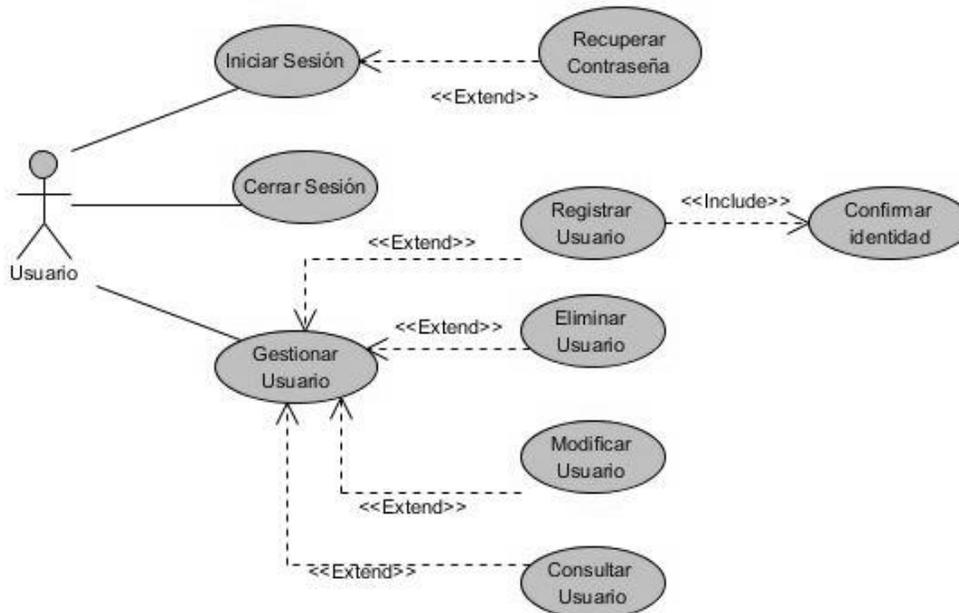


Imagen 8: Casos de uso-Gestión de usuarios

### Módulo Interacción con Redes Sociales.

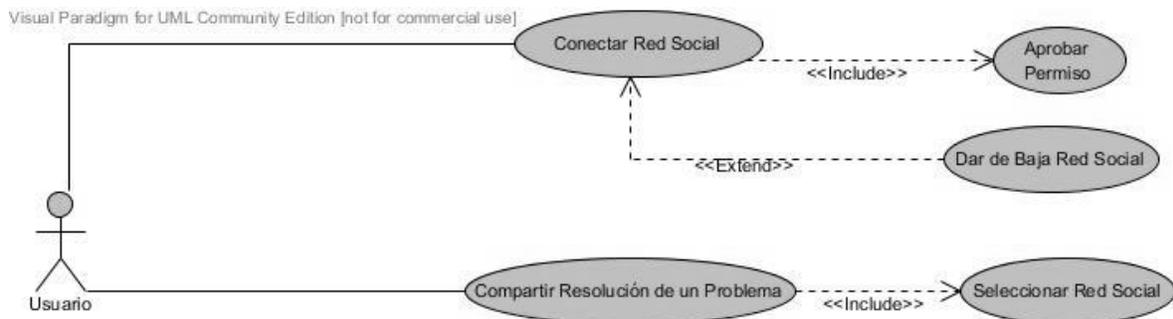


Imagen 9: Casos de uso: Redes sociales

## Módulo Gestión de Concursos

Visual Paradigm for UML Community Edition [not for commercial use]

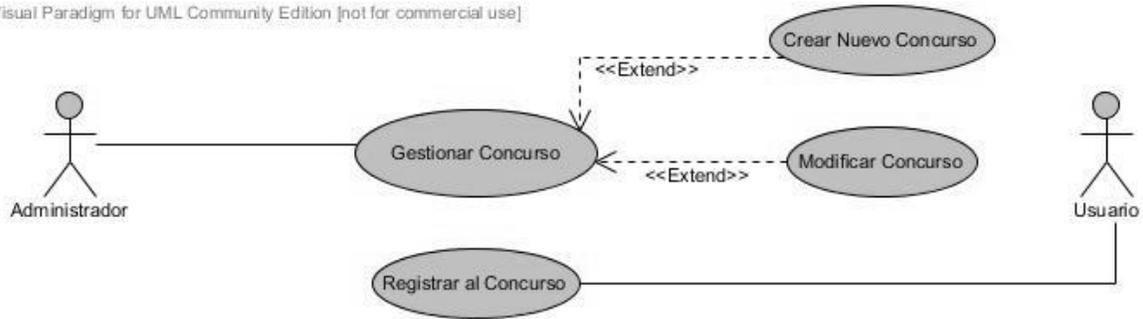


Imagen 10: Casos de uso- Gestion de concursos

## Módulo Gestión de Problemas

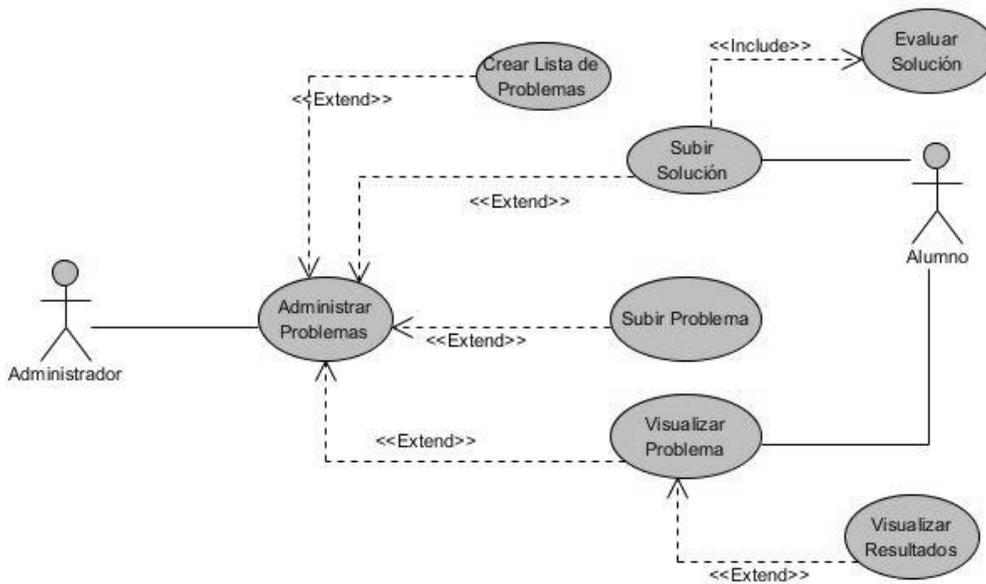


Imagen 11: Casos de uso: Gestión de problemas

Diagramas de secuencia.

Los diagramas de secuencia muestran una interacción ordenada según la secuencia temporal de eventos. Estos diagramas muestran la interacción entre los objetos que tiene lugar a través del intercambio de mensajes, en particular, muestran los objetos participantes en la interacción y los mensajes que se intercambian ordenadas según la secuencia de tiempo.

La metodología empleada en este proyecto indica que sólo se deben realizar los diagramas de secuencia de los componentes principales o de aquellos casos de uso que requieran de un procedimiento muy específico.

*Crear un nuevo concurso:* Muestra la secuencia que el usuario debe seguir para cargar un concurso nuevo, es importante debido a que es uno de los módulos que no existen en el sistema actual y es necesario llevar un seguimiento de su compartamiento, así como la vinculación con el resto de los módulos.

*Inicio de sesión:* Muestra la secuencia para iniciar sesión en el sistema, ya que este inicio se realiza utilizando proveedores externos se debe tener un control de los datos que se envían.

*Evaluar Problema:* Remarca el objetivo principal del sistema llevando a cabo la secuencia para vincular nuestra propuesta con la parte ya implementada.

# Crear nuevo concurso.

sd CrearConcurso

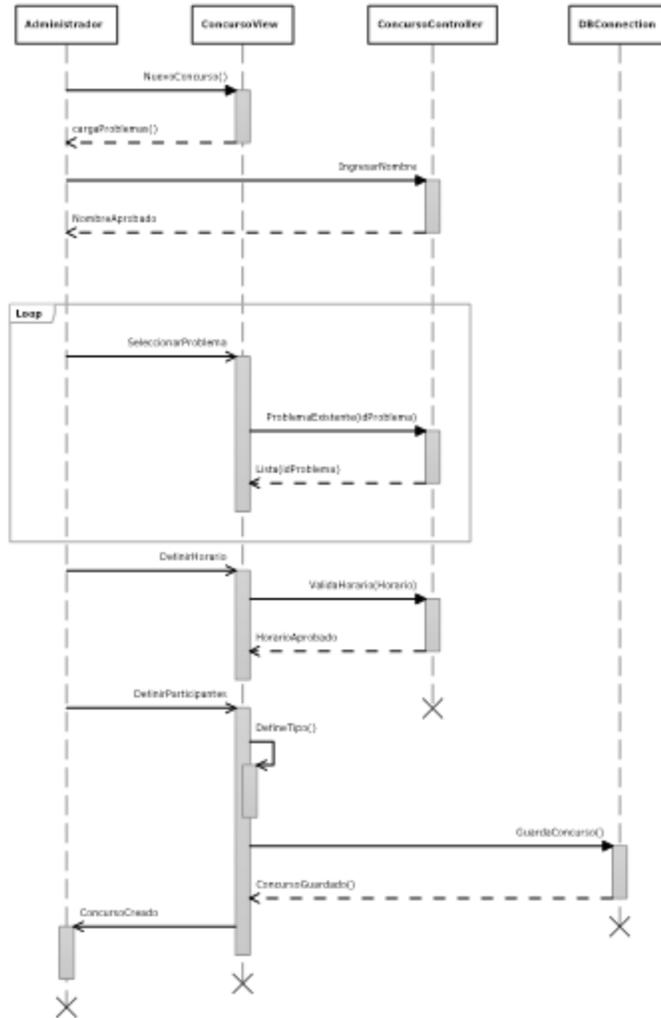


Imagen 12: Secuencia

# Inicio de sesión

sd InicioSesion

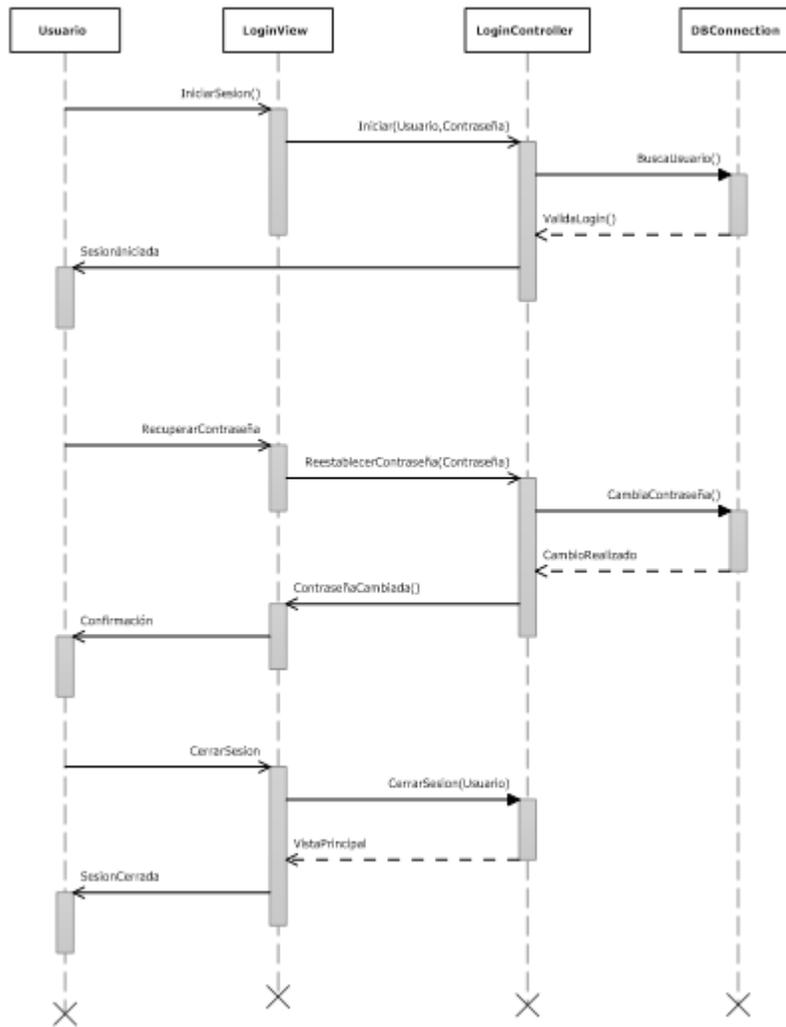


Imagen 13: Secuencia Inicio de sesión

# Evaluar Problema.

sd Evaluador

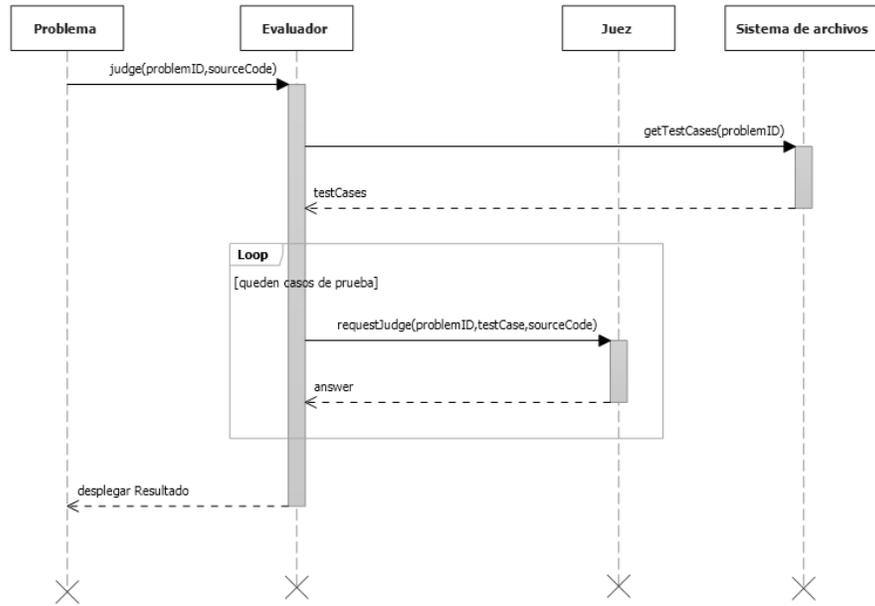


Imagen 14: Evaluar problema

## Diagrama de flujo de datos

Los diagramas de flujo de datos son usados comúnmente para representar de manera gráfica un sistema, descomponiendo este en partes y mostrando cuales son o no susceptibles a amenazas.

Los "límites de confianza" (líneas rojas punteadas) son la parte más subjetiva del diagrama. Representan los límites entre elementos de confianza y no confianza, que ayudan a identificar el origen de amenazas potenciales para el sistema.

Para mitigar posibles amenazas de seguridad que se pudieran presentar en el sistema, se empleó la metodología SDL de Microsoft<sup>7</sup> para identificar los riesgos a los que el sistema está expuesto.

Esta metodología sirve para identificar cinco amenazas principales:

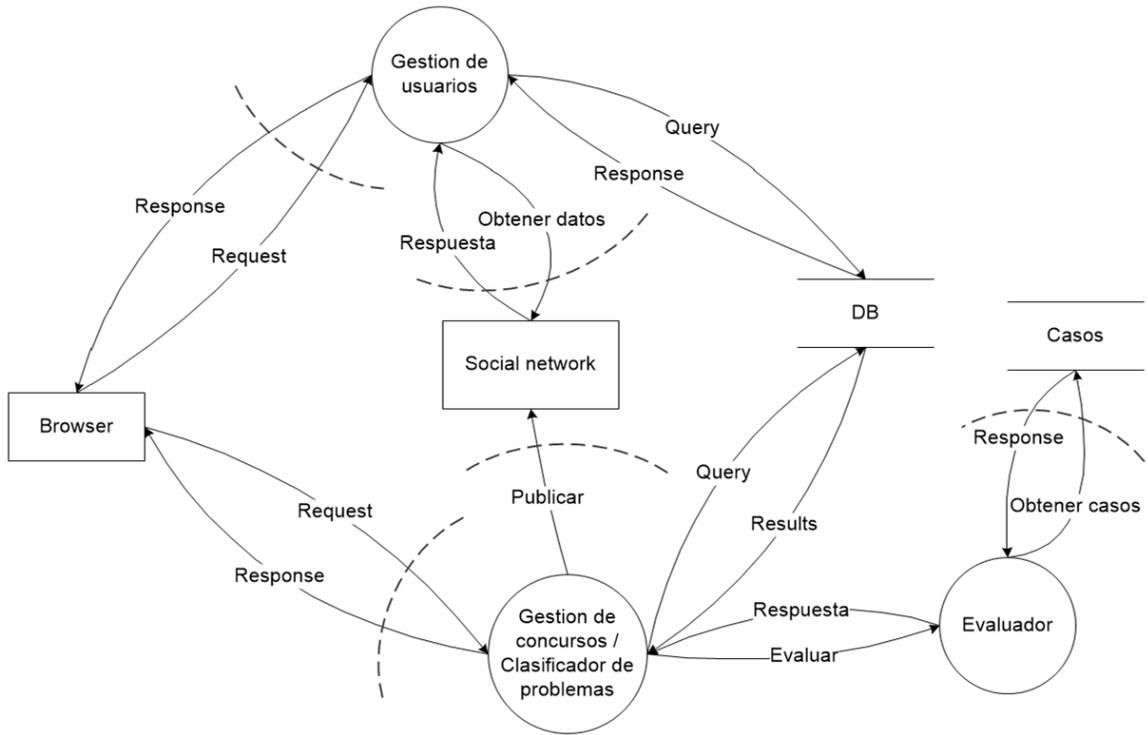
- Suplantación de identidad.
- Alteración de la información.
- Repudio.
- Revelación de información.
- Negación de servicio.

En el diagrama de flujo de datos del sistema se identificaron los siguientes riesgos potenciales:

- La comunicación entre el evaluador y el sistema de archivos del sistema operativo donde resida el sistema. Esto es debido a que se debe validar la autenticidad de los archivos de casos de prueba. Se solucionara a través de una firma digital, como se especificó en el documento de requerimientos del módulo Evaluador.
- La comunicación entre el sistema y los sitios de redes sociales. Se deberá hacer uso de las API oficiales de Facebook y Twitter para disminuir los riesgos de revelación de información y alteración de la información.
- Toda la comunicación entre el cliente y el servidor deberá ser tratada como hostil, por lo que se deberá de validar y limpiar toda la información que el cliente envíe al servidor.

---

<sup>7</sup> <http://msdn.microsoft.com/en-us/magazine/dd347831.aspx>



# TRABAJO TERMINAL 2013-A003

Sistema web para la evaluación automática de programas de Karel el Robot.

## MÓDULO DEL EVALUADOR

### ESPECIFICACIONES DE DISEÑO DEL DESARROLLO

#### Resumen del módulo

En este módulo se realizan las evaluaciones a través de un servicio web, ya que se requiere que el proceso de evaluación este lógicamente separado del sistema web. En este documento se describe la arquitectura del servicio web y la manera de realizarle peticiones.

#### Descripción de la arquitectura

Las clases que forman parte de este módulo son:

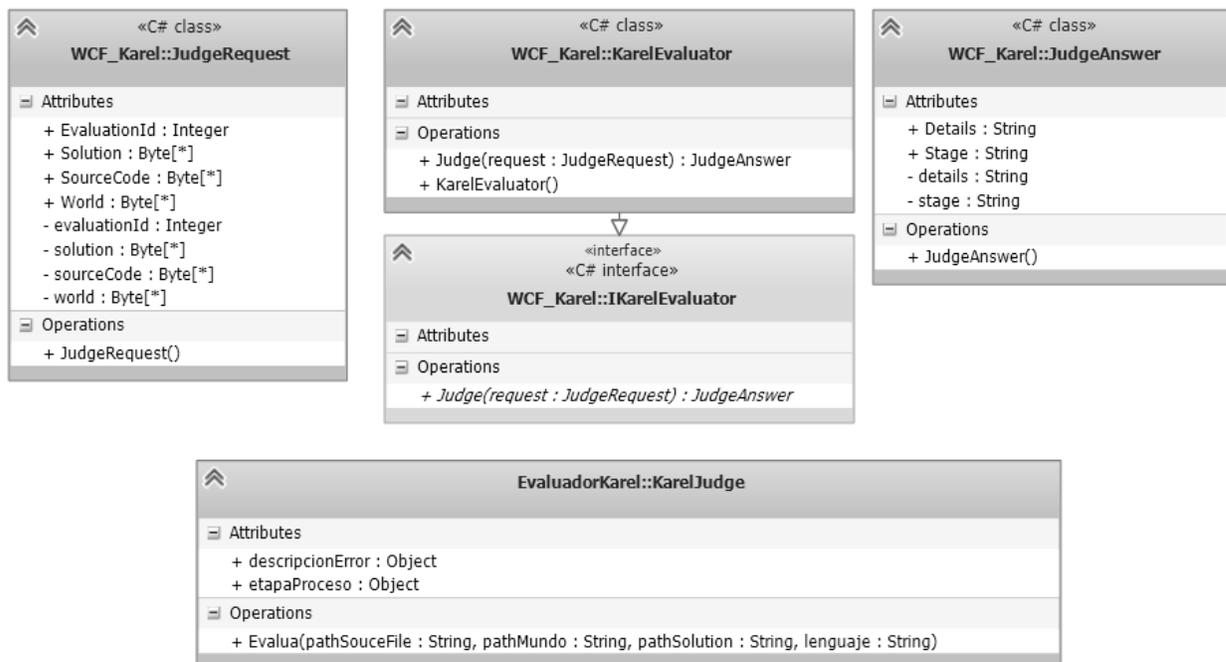


Imagen 15: Diagrama de clases-Módulo evaluador

Clases	Descripción
<b>WCF_Karel::JudgeRequest</b>	Clase que almacena la información relacionada con una petición al servicio web. Los objetos de esta clase almacenaran el código fuente, el mundo de prueba y la solución al problema, para poder determinar si el problema fue correctamente resuelto.
<b>WCF_Karel::JudgeResponse</b>	Clase que almacena la información relacionada a la evaluación. Contiene una descripción de la evaluación, que se compone de dos elementos: Etapa de evaluación alcanzada y detalles de la evaluación.
<b>WCF_Karel::Evaluator</b>	Clase principal del servicio web, que expone el método Judge que es el método a través del cual se solicitan las evaluaciones.
<b>EvaluadorKarel::KarelJudge</b>	Clase provista por el Comité Mexicano de Informática, que contiene el compilador y evaluador de los programas de Karel.

Tabla 37-Descripción de clases

## Detalles del diseño

### La clase KarelJudge

Esta clase contiene el evaluador oficial que utiliza el Comité Mexicano de Informática durante las competencias de la Olimpiada Mexicana de Informática. La clase está escrita en el lenguaje de programación Visual Basic .NET, pero gracias a la plataforma de .NET puede ser utilizada dentro del servicio web que está escrito en el lenguaje C#.

La clase contiene el método:

```
Public Sub Evalua(pathSouceFile As String,
                 pathMundo As String,
                 pathSolution As String,
                 lenguaje as String)
```

Que es el método que realiza la evaluación, asignando en las variables:

```
Public etapaProceso
Public descripcionError
```

El resultado de la evaluación. Los valores que pueden tomar la variable `etapaProceso` son:

Valor	Significado
<b>Compilación</b>	La evaluación se ha detenido en la etapa de Compilación. Esto generalmente es debido a que el programa no compila correctamente.
<b>Ejecución</b>	La evaluación se ha detenido en la etapa de Evaluación. Esto generalmente es debido a errores en la ejecución (Karel intento atravesar una pared, tomar un beeper donde no existía alguno, etc.)
<b>Evaluación</b>	Se termina la evaluación, y se determinó si la respuesta es correcta o no.

Tabla 38-Detalles de la evaluación

Los valores que pueden tomar la variable `descripcionError` son los especificados en el documento de requisitos de este módulo.

Únicamente la combinación de valores *Evaluación* y *OK* resultan en un veredicto de Correcto al momento de evaluar el programa. Cualquier otra combinación, se considera que el programa no resuelve el problema.

#### Windows Communication Foundation

Windows Communication Foundation (WCF) es un framework para construir aplicaciones orientadas a servicios. Al utilizar WCF, se pueden realizar peticiones de manera asíncrona o síncrona de un extremo de un servicio a otro. Estos extremos pueden ser parte de un servicio disponible de manera permanente hospedada en IIS, o bien, puede ser un servicio que corre sobre una aplicación. Un extremo del servicio puede ser el cliente de un servicio que solicita datos de otro extremo del servicio.

WCF soporta diferentes especificaciones acorde a las funciones que tiene que llevar a cabo, principalmente:

- Envío de mensajes: SOAP (*Simple Object Access Protocol*) es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. Utiliza un protocolo para realizar el envío de mensajes, como puede ser HTTP.
- Metadatos: Web Services Description Language (WSDL), es un formato que se utiliza para describir servicios Web. WSDL describe la interfaz pública a los servicios Web. Está basado en XML y describe la forma de comunicación, es decir, los

requisitos del protocolo y los formatos de los mensajes necesarios para interactuar con los servicios listados en su catálogo. Las operaciones y mensajes que soporta se describen en abstracto y se ligan después al protocolo concreto de red y al formato del mensaje.

A continuación se muestra el archivo WSDL del servicio web del evaluador:

```
<wsdl:definitions xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
xmlns:tns="http://tempuri.org/"
xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
xmlns:wsx="http://schemas.xmlsoap.org/ws/2004/09/mex"
xmlns:wsap="http://schemas.xmlsoap.org/ws/2004/08/addressing/policy"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
xmlns:msc="http://schemas.microsoft.com/ws/2005/12/wsdl/contract"
xmlns:wsp="http://www.w3.org/ns/ws-policy"
xmlns:wsa10="http://www.w3.org/2005/08/addressing"
xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata" name="KarelEvaluator"
targetNamespace="http://tempuri.org/">
<wsdl:types>
<xsd:schema targetNamespace="http://tempuri.org/Imports">
<xsd:import schemaLocation="http://localhost:8080/evaluador?xsd=xsd0"
namespace="http://tempuri.org/">
<xsd:import schemaLocation="http://localhost:8080/evaluador?xsd=xsd1"
namespace="http://schemas.microsoft.com/2003/10/Serialization/">
<xsd:import schemaLocation="http://localhost:8080/evaluador?xsd=xsd2"
namespace="http://schemas.datacontract.org/2004/07/WCF_Karel"/>
</xsd:schema>
</wsdl:types>
<wsdl:message name="IKarelEvaluator_Judge_InputMessage">
<wsdl:part name="parameters" element="tns:Judge"/>
</wsdl:message>
<wsdl:message name="IKarelEvaluator_Judge_OutputMessage">
<wsdl:part name="parameters" element="tns:JudgeResponse"/>
</wsdl:message>
<wsdl:portType name="IKarelEvaluator">
<wsdl:operation name="Judge">
<wsdl:input wsam:Action="http://tempuri.org/IKarelEvaluator/Judge"
message="tns:IKarelEvaluator_Judge_InputMessage"/>
<wsdl:output wsam:Action="http://tempuri.org/IKarelEvaluator/JudgeResponse"
message="tns:IKarelEvaluator_Judge_OutputMessage"/>
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="BasicHttpBinding_IKarelEvaluator" type="tns:IKarelEvaluator">
<soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
<wsdl:operation name="Judge">
<soap:operation soapAction="http://tempuri.org/IKarelEvaluator/Judge"
style="document"/>
<wsdl:input>
<soap:body use="literal"/>
</wsdl:input>
<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="KarelEvaluator">
```

```

<wsdl:port name="BasicHttpBinding_IKarelEvaluator"
binding="tns:BasicHttpBinding_IKarelEvaluator">
<soap:address location="http://localhost:8080/evaluador"/>
</wsdl:port>
</wsdl:service>
</wsdl:definitions>

```

El servicio web que procesa las peticiones de evaluación está desarrollado sobre esta plataforma.

### Realización de peticiones

Se envía la petición invocando al método `Judge`. Se puede acceder a este método una vez que se tienen los datos de la interfaz, a través del archivo WSDL. Se envía como parámetro un objeto de tipo `JudgeRequest` que contiene los elementos necesarios para realizar dicha evaluación.

Una vez realizada la petición, en el proceso que corre el servicio web, los despachadores son los responsables de extraer los mensajes entrantes de los canales destinados a este propósito, traduciros en llamadas a métodos, y del envío del resultado a quien realizo la petición.

### Respuesta del servicio web

Las peticiones realizadas al servicio web se procesan y retornan un archivo XML que tiene la siguiente estructura:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header />
  <s:Body>
    <JudgeResponse xmlns="http://tempuri.org/">
      <JudgeResult
        xmlns:a="http://schemas.datacontract.org/2004/07/WCF_Karel"
        xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Details>
          Error al definir la clase principal del programa.
        </a:Details>
        <a:Stage>Compilacion</a:Stage>
      </JudgeResult>
    </JudgeResponse>
  </s:Body>
</s:Envelope>

```

Este archivo XML es una versión serializada de la clase **JudgeResponse**, que será tomada por el sistema web para desplegar la información al usuario y almacenarla en la base de datos.

### Pruebas

#### Primer feedback

#	Requerimiento	Observación	Estatus
1	El sistema podrá evaluar, bajo petición del sistema, una solución a un problema.		Finalizado
2	El evaluador podrá determinar si la solución es correcta, o cual es la razón por la cual no lo es.		Finalizado
3	El evaluador podrá tener una cola de espera para evaluar soluciones.	Threadpooling de WCF, configuración de una instancia con múltiple concurrencia.	Finalizado
4	El evaluador deberá de determinar una respuesta en el menor tiempo posible, para evitar la acumulación de soluciones en la cola de espera.	Tiempo de evaluación de un problema de nivel nacional en menos de 250 ms. (Rombo, OMI 2010)	Finalizado.
5	Establecer un mecanismo de verificación de los archivos de evaluación, para evitar la modificación de estos de manera no autorizada.	Comparación del hash del archivo que se cargó en el sistema contra el que está almacenado en el sistema de archivos del sistema operativo.	Finalizado

Tabla 39-Primer feedback módulo evaluador

## Resultados

Requerimientos finalizados	Requerimientos en construcción	Requerimientos por implementar
5/5	0/5	0/5

Tabla 40-Resultados primer feedback módulo evaluador

## Avance: 100%

### Ediciones al documento

Fecha	Autor	Cambios
6/1/2014	Juan Carlos	Creación del documento
25/2/2014	Juan Carlos	Se agrega información acerca de WCF
21/4/14	Larissa	
23/4/14	Juan Carlos	Observaciones de los requerimientos.

## Apéndice

### Definiciones y acrónimos

**Aplicaciones orientadas a servicios (Service-oriented applications, SOA).** La arquitectura orientada a servicios es el uso de servicios web para enviar y recibir datos. Los servicios tienen la ventaja general de estar débilmente acoplados entre una aplicación y otra. Una relación de acoplamiento débil implica que cualquier cliente creado en cualquier plataforma puede conectar con cualquier servicio siempre y cuando se cumplan los contratos esenciales.

# TRABAJO TERMINAL 2013-A003

Sistema web para la evaluación automática de programas de Karel el Robot.

## MÓDULO DEL CLASIFICADOR DE PROBLEMAS

### ESPECIFICACIONES DE DISEÑO DEL DESARROLLO

#### Resumen del módulo

En este módulo se realizarán las acciones necesarias para poder desplegar al usuario los problemas de una manera ordenada, de acuerdo a una serie de criterios definidos por el usuario, o bien, ya establecidos por el sistema. Además, este módulo es el que tiene comunicación con el módulo de evaluación para determinar si es correcta o no una respuesta.

#### Descripción de la arquitectura

Las clases que forman parte de este módulo son:

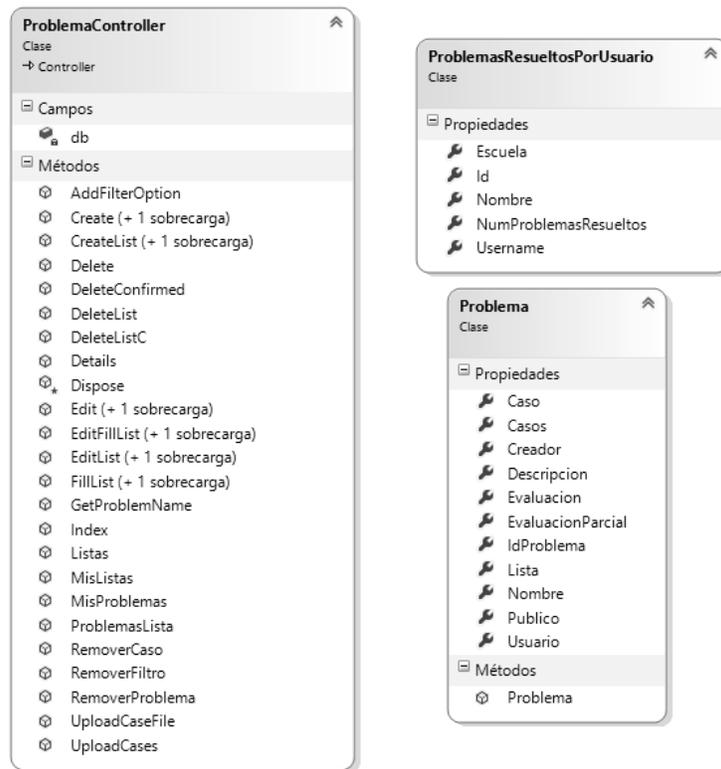


Imagen 16: Diagrama de clases: Módulo evaluador

Clases	Descripción
<b>ProblemaController</b>	Permite realizar las altas, bajas, consultas y cambios de los problemas del sistema, además de permitir a los usuarios la generación de listas de problemas específicos a las cuales se les puede agregar un filtro para definir quién puede acceder a ellas.

Tabla 41-Descripción de las clases del módulo gestión de problemas

### Detalles del diseño

#### Migración de los datos del sitio web anterior

Este módulo es el único que retomara la información del sitio web anterior, por lo que se retomara la base de datos de 125 problemas aproximadamente para poder ser utilizadas en este sistema.

#### Consideraciones:

- Evitar la pérdida de datos.

- Minimizar los errores comunes limpiando los datos antes de realizar la migración.
- Acceder de forma nativa a casi todos los tipos de datos.
- Evaluar, corregir y supervisar la calidad de los datos, incluyendo su limpieza, estandarización y eliminación de duplicados.

## Pruebas

### Primer feedback

#	Requerimiento	Observación	Estatus
1	El sistema podrá clasificar problemas.	Aún no se implementan las métricas para la clasificación de los problemas	Finalizado
2	Los usuarios podrán crear listas de problemas, que podrán ser públicas o privadas.		Finalizado
3	Los usuarios de tipo asesor o administrador podrán subir nuevos problemas al sistema.	Aún no se implementa la parte que permita a los usuarios subir imágenes para indicar los ejemplos	Finalizado Con detalles.
4	Los usuarios podrán solicitar la evaluación de una solución a un problema en el sistema.		Finalizado. Sin errores.
5	El sistema podrá proporcionar información acerca del número de soluciones correctas a un problema, y el porcentaje de soluciones correctas.		Terminado

Tabla 42-Feedback del módulo gestión de problemas

### Resultados

Requerimientos finalizados	Requerimientos en construcción	Requerimientos por implementar
5/5	5/5	5/5

Tabla 43-Resultados del primer feedback del módulo gestión de problemas

**Avance: 100%**

### Ediciones al documento

Fecha	Autor	Cambios
25/2/2014	Juan Carlos	Creación del documento
22/4/2014	Larissa	Observaciones del primer feedback

# TRABAJO TERMINAL 2013-A003

Sistema web para la evaluación automática de programas de Karel el Robot.

## MÓDULO GESTIÓN DE USUARIOS

### ESPECIFICACIONES DE DISEÑO DEL DESARROLLO

#### Resumen del módulo

En este módulo se llevaran a cabo todas las actividades relacionadas con la gestión de los usuarios del sistema, es decir, llevar a cabo altas, bajas, cambios y consultas de las cuentas de usuario. Proporcionar los mecanismos necesarios para la validación de las cuentas y el acceso sólo a usuarios autorizados en el resto del sistema.

#### Descripción de la arquitectura

Este módulo se encuentra implementado dentro del controlador UsuarioController apoyándose de la clase AccountController :

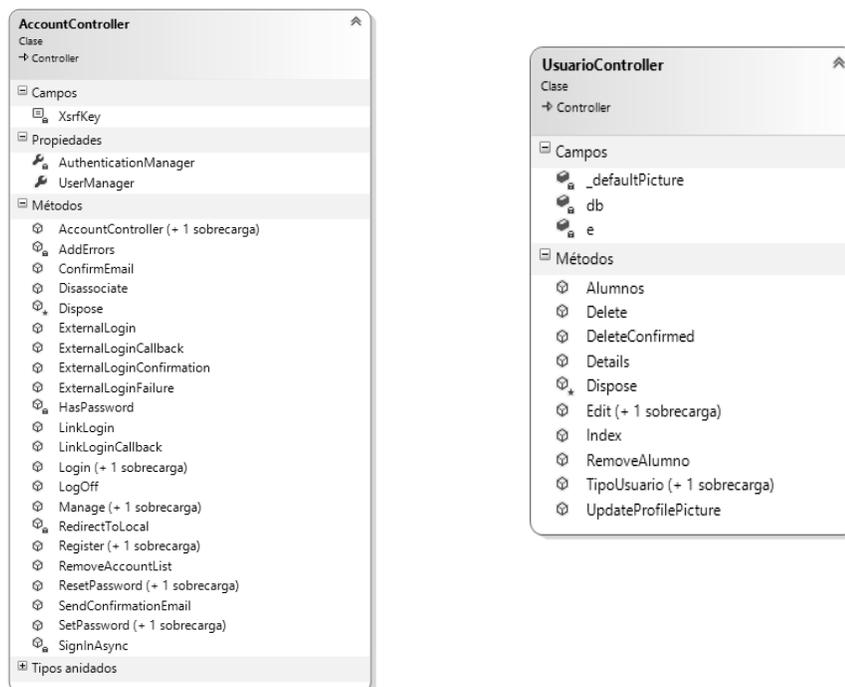


Imagen 17: Clases del módulo Usuarios

Métodos	Descripción
<b>Alumnos</b>	Muestra la lista de alumnos de un asesor.
<b>Delete</b>	Muestra una vista donde se confirma la eliminación de la cuenta
<b>DeleteConfirmed</b>	Si la eliminación es confirmada elimina al usuario
<b>Details</b>	Devuelve los detalles del perfil del usuario seleccionado
<b>Dispose</b>	Permite a un control de servidor para realizar final limpieza antes de que se libere de memoria.
<b>Edit</b>	Permite la edición del perfil del usuario que ha iniciado sesión
<b>Index</b>	Devuelve una lista de todos los usuarios existentes en el sistema así como los problemas que ha resuelto.
<b>RemoveAlumno</b>	Elimina un alumno de la lista del asesor
<b>TipoUsuario</b>	Permite el cambio entre tipos de usuarios, así como la asignación de un asesor a los alumnos.
<b>UpdateProfilePicture</b>	Permite subir una imagen para el perfil del usuario en formato jpg y la convierte en un arreglo de bytes para almacenarlo en la base de datos

### Detalles del diseño

El inicio de sesión se realiza con los datos proporcionados por las clases generadas por el framework MVC 4 de ASP. NET, de modo que la edición del nombre de usuario no se permite y en este módulo no se realiza el cambio de contraseñas. Se apoya de la clase AccountController para realizar los registros de proveedores externos, estos métodos forman parte del módulo de las redes sociales.

### Pruebas

#### Primer feedback

#	Requerimiento	Observación	Estatus
1	Los usuarios podrán registrarse a través de un formulario.	Algunas alertas aparecen en español y otras en inglés. Es necesario homogenizar las interfaces	Terminado Con detalles de interfaz
2	Los usuarios podrán actualizar su información personal en cualquier momento.	Problemas si no selecciona una imagen	Terminado Errores de validación
3	Los usuarios podrán dar de baja su cuenta en cualquier momento.	No esta implementado	Por realizar
4	Los usuarios de tipo alumno podrán seleccionar su asesor		En construcción

5	Los usuarios únicamente podrán tener solo un tipo, que puede ser Administrador, Asesor, Alumno o Invitado.	Implementar esta opción sólo para el administrador	En construcción
6	Los usuarios deberán de confirmar su identidad para poder utilizar el sistema.	No esta implementado	Por realizar
7	Los usuarios deberán aceptar los términos y condiciones del sistema para poder registrarse en el sistema.	No existen los términos y condiciones, permite el registro ingresando los datos	Por realizar
8	Los usuarios podrán ver información pública de los demás competidores.		Terminado Sin errores
9	Los usuarios que olviden su contraseña podrán recuperarla a través de un formulario para recuperar contraseñas.	Esta implementado pero la opción no aparece en el formulario	Terminado Con errores ya que no se muestra
10	Los usuarios no podrán almacenar información personal sensible en el sistema.	En ningún momento el sistema solicita esta información.	Terminado Sin errores

Tabla 44-Primer feedback módulo gestión de usuarios

### Resultados

Requerimientos finalizados	Requerimientos en construcción	Requerimientos por implementar
5/10	3/10	2/10

Tabla 45-Resultado primer feedback módulo gestión de usuarios

**Avance: 65%**

### Segundo Feedback.

#	Requerimiento	Observación	Estatus
1	Los usuarios podrán registrarse a través de un formulario.	Poner validación del formato del correo electrónico	Terminado
2	Los usuarios podrán actualizar su información personal en cualquier momento.		Terminado
3	Los usuarios podrán dar de baja su cuenta en cualquier momento.		Terminado
4	Los usuarios de tipo alumno podrán seleccionar su asesor.		Terminado
5	Los usuarios únicamente podrán tener solo un tipo, que puede ser Administrador, Asesor, Alumno o Invitado.		Terminado
6	Los usuarios deberán de confirmar su identidad para poder utilizar el sistema.		Terminado

7	Los usuarios deberán aceptar los términos y condiciones del sistema para poder registrarse en el sistema.	No existen los términos y condiciones, permite el registro ingresando los datos	Por realizar
8	Los usuarios podrán ver información pública de los demás competidores.		Terminado
9	Los usuarios que olviden su contraseña podrán recuperarla a través de un formulario para recuperar contraseñas.		Terminado
10	Los usuarios no podrán almacenar información personal sensible en el sistema.	En ningún momento el sistema solicita esta información.	Terminado

Tabla 46-Segundo Feedback módulo gestión de usuarios

### Resultados

Requerimientos finalizados	Requerimientos en construcción	Requerimientos por implementar
10/10	0/10	0/10

Tabla 47-Resultado segundo feedback módulo gestión de usuarios

**Avance: 100%**

### Ediciones al documento

Fecha	Autor	Cambios
25/2/2014	Larissa	Creación del documento
13/4/2014	Larissa	Observaciones del primer feedback
22/4/2014	Larissa	Observaciones del segundo feedback y actualización de las clases.

# TRABAJO TERMINAL 2013-A003

Sistema web para la evaluación automática de programas de Karel el Robot.

## MÓDULO DE INTERACCIÓN CON REDES SOCIALES

### ESPECIFICACIONES DE DISEÑO DEL DESARROLLO

#### Resumen del módulo

En este módulo se permitirá realizar la publicación en perfiles sociales del usuario de la correcta solución a un problema así como permitir el acceso al sistema mediante alguna red social, las redes sociales que serán implementadas son Twitter y Facebook.

#### Descripción de la arquitectura

Las clases que forman parte de este módulo son:



Tabla 1-Diagramas de clases del módulo Redes Sociales

Clases	Descripción
<b>ExternalLoginConfirmationViewModel</b>	Clase permite confirmar que la conexión a la red social ha sido exitosa, redirecciona a la vista para llevar a cabo el registro mediante una red social.
<b>ExternalLoginViewModel</b>	Clase por defecto de MVC 4 en ASP.NET que permite llevar a cabo el inicio de sesión mediante un proveedor externo.

## Detalles del diseño

### La clase ExternalLoginConfirmationViewModel

Esta clase permite llevar a cabo el inicio de sesión en el sistema mediante un proveedor externo, cuando la conexión con la red social es exitosa, se redirecciona a la página de registro mediante una red social, donde se solicita al usuario el nombre de usuario para el sistema (por defecto este método toma el nombre de usuario de la red social) así como el tipo de usuario dentro del sistema.

### La clase ExternalLoginViewModel

Esta clase esta implementada por defecto en ASP.NET con MVC 4, donde obtiene la url de dicha red a la que se desea hacer el inicio de sesión, para poder utilizar los métodos es necesario tanto en Twitter como Facebook obtener el ID de la aplicación así como la llave secreta y configurarlas como proveedor externo en el archivo Startup del sistema.

\*Para la publicación de redes sociales cuando el problema ha sido resuelto satisfactoriamente se necesita solo un script que se encuentra en la vista del evaluador, estos códigos son proporcionados por las API's de Twitter y Facebook que sugieren sean incrustados en la sección dónde se va a hacer uso, solamente es necesario asignar los parámetros de la publicación.

## Primer feedback

#	Requerimiento	Observación	Estatus
1	Los usuarios podrán iniciar sesión mediante alguna red social.	Hasta el momento ha sido posible acceder mediante redes sociales, todas las conexiones han sido exitosas.	Terminado
2	Los usuarios podrán publicar la resolución de un problema en alguna red social, siempre y cuando la evaluación de éste sea correcta.	En los problemas que no se evalúan por casos funciona correctamente, falta realizar la prueba por casos.	Terminado
3	Los usuarios podrán elegir en que red social será publicado.	Cuando se resuelve un problema, se muestran los botones de compartir en Facebook y Twitter.	Terminado
4	Los usuarios deben permitir establecer conexión con la red social.	Esta parte no es propia del sistema, pero es necesaria para poder publicar en los perfiles sociales del usuario, cuando se desea publicar en redes sociales Facebook y Twitter solicitan permisos al usuario para poder publicar contenido.	Terminado
5	Los concursos podrán ser compartidos en alguna red social.	No está implementado	Terminado

Tabla 48-Primer feedback módulo interacción con redes sociales

## Resultados

Requerimientos finalizados	Requerimientos en construcción	Requerimientos por implementar
5/5	5/5	5/5

Tabla 49-Resultado primer feedback módulo gestión de usuarios

## Avance: 100%

Fecha	Autor	Cambios
25/2/2014	Miguel	Creación del documento
19/4/2014	Miguel	Observaciones del primer feedback

# TRABAJO TERMINAL 2013-A003

Sistema web para la evaluación automática de programas de Karel el Robot.

**MÓDULO GESTIÓN DE CONCURSOS.**

## ESPECIFICACIONES DE DISEÑO DEL DESARROLLO

### Resumen del módulo

Éste módulo es capaz de administrar (realizar altas, bajas, cambios y consultas) los concursos que serán hospedados en el sistema. Los concursos constan de un conjunto de problemas y una duración determinada. En los concursos podrán participar los usuarios de tipo alumno y asesor que cumplan con los criterios establecidos al momento de crear el concurso. Así mismo este módulo permite visualizar los resultados del concurso.

### Descripción de la arquitectura

Las clases que forman parte de este módulo son:

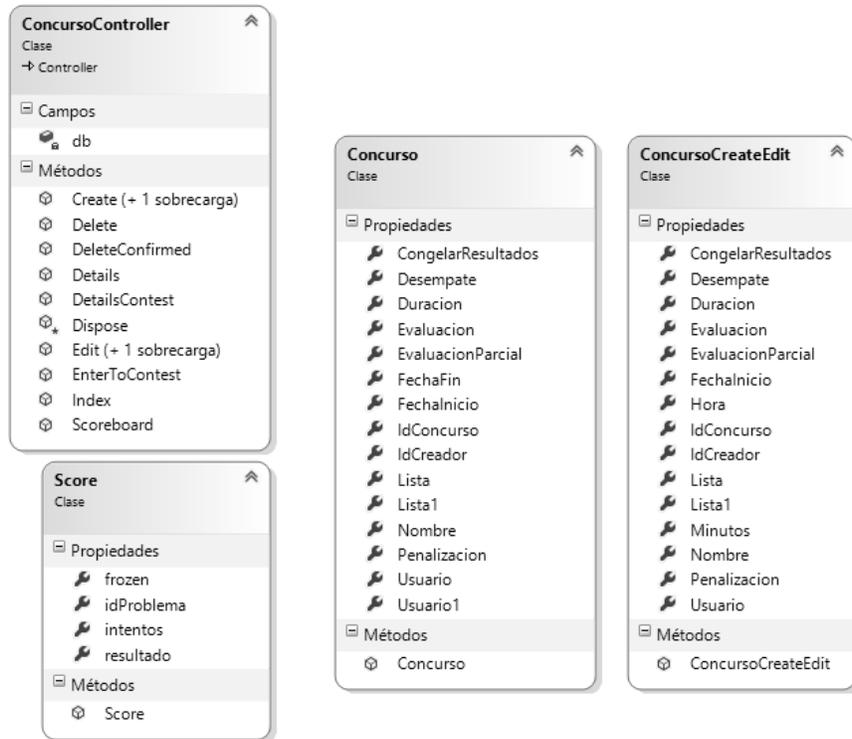


Ilustración 1-Diagrama de clases del módulo gestión de concursos

Clases	Descripción
<b>ConcursoController</b>	Es el controlador que permite el acceso a las vistas del módulo, así como las funcionalidades propias del módulo.
<b>Concurso</b>	Modelo definido des la base de datos para poder almacenar un concurso.
<b>ConcursoCreateEdit</b>	Modelo creado para la creación y Edición de un concurso
<b>Score</b>	Modelo del Scoreboard que permite traer todos los elementos de la base de datos.

Ilustración 2-Descripción de clases del módulo gestión de concursos.

Detalles del diseño

Pruebas

### Primer feedback

#	Requerimiento	Observación	Estatus
1	Los usuarios podrán crear concursos en el sistema, con problemas previamente cargados en el sistema.		Terminado
2	Los usuarios que creen concursos podrán configurar las características de tiempo, problemas, y usuarios que podrán registrarse en el concurso.		Terminado
3	Los usuarios que cumplan con los requisitos establecidos al momento de configurar el concurso podrán registrarse.		Terminado
4	El sistema podrá evaluar los problemas en tiempo real.	Excepto que se caiga el servidor ya que de ese modo se guardan en una cola respetando su tiempo de envío.	
5	El sistema podrá desplegar los resultados del concurso en tiempo real.		Terminado
6	El sistema podrá almacenar de manera permanente los resultados de todos los concursos.		Terminado

Tabla 50-Feedback del módulo gestión de concursos

### Resultados

Requerimientos finalizados	Requerimientos en construcción	Requerimientos por implementar
6/6	0/6	0/6

Tabla 51-Resultados del primer feedback del módulo gestión de problemas

**Avance: 100%**

### Ediciones al documento

Fecha	Autor	Cambios
25/2/2014	Juan Carlos	Creación del documento
22/4/2014	Larissa	Observaciones del primer feedback

# TRABAJO TERMINAL 2013-A003

## Sistema web para la evaluación automática de programas

### Riesgos mitigados y planes de contingencia

Cambio del gestor de base de datos.

Durante el desarrollo del proyecto se presentó el siguiente riesgo: *La elección de las tecnologías a emplear no ayuda a cumplir con los requerimientos del sistema.* Mismo que estaba contemplado dentro de los riesgos iniciales y nos llevó a cambiar el gestor de base de datos a de MySQL a SQL Server, esto debido a que no existe un conector del Framework ASP.NET MVC 5 con Entity Framework 6 para MySQL. La tecnología del framework es prácticamente nueva y no existe una forma de conectar estas tecnologías por el momento. Este riesgo nos ocasiono un atraso de dos sprints ya que dentro de la documentación no indica dicha incompatibilidad, fue hasta después de la realización de diversas pruebas que se optó por realizar la migración de la base de datos implementada en MySQL a SQL Server.

*Base de datos de MySQL:*

```
DROP DATABASE IF EXISTS karel;

CREATE DATABASE karel;
USE karel;

DROP TABLE IF EXISTS usuario;
CREATE TABLE usuario (
    idUsuario varchar(20) NOT NULL,
    nombre varchar(50) NOT NULL,
    fechaNacimiento date NOT NULL,
    escuela varchar(100) NULL,
    gradoNum int NOT NULL,
    gradoUnit varchar(12) NOT NULL,
    estado int NOT NULL,
    tipo int NOT NULL,
    correo varchar(50) NOT NULL,
    pass varchar(32) NOT NULL,
    genero char(1) NOT NULL,
    lenguaje varchar(6) NOT NULL,
    pregunta int NOT NULL,
    respuesta varchar(40) NOT NULL,
    tokenfb varchar(100),
    tokentw varchar(100),
    PRIMARY KEY(idUsuario)
```

```

);

DROP TABLE IF EXISTS concurso;
CREATE TABLE concurso(
    idConcurso int NOT NULL AUTO_INCREMENT,
    nombre varchar(70) NOT NULL,
    fechaInicio date NOT NULL,
    duracion int NOT NULL,
    lista int,
    filtro int,
    congelarResultados int,
    penalizacion boolean NOT NULL,
    desempate boolean NOT NULL,
    tipoEvaluacion int NOT NULL,
    PRIMARY KEY(idConcurso)

);

DROP TABLE IF EXISTS problema;
CREATE TABLE problema(
    idProblema int NOT NULL AUTO_INCREMENT,
    nombre varchar(50) NOT NULL,
    descripcion text NOT NULL,
    casos int,
    tipo boolean NOT NULL,
    PRIMARY KEY (idProblema)
);

DROP TABLE IF EXISTS evaluacion;
CREATE TABLE evaluacion(
    idEvaluacion int NOT NULL AUTO_INCREMENT,
    idProblema int NOT NULL,
    idUsuario varchar(20) NOT NULL,
    horaSubida date NOT NULL,
    resultado varchar(30) NOT NULL,
    PRIMARY KEY(idEvaluacion)
);

DROP TABLE IF EXISTS resultadoConcurso;
CREATE TABLE resultadoConcurso(
    idConcurso int NOT NULL,
    idEvaluacion int NOT NULL,
    PRIMARY KEY(idConcurso, idEvaluacion)
);

DROP TABLE IF EXISTS asesorAlumno;
CREATE TABLE asesorAlumno(
    idAlumno varchar(20) NOT NULL,
    idAsesor varchar(20) NOT NULL,
    PRIMARY KEY(idAlumno, idAsesor)
);

DROP TABLE IF EXISTS listaProblemas;
CREATE TABLE listaProblemas(
    idLista int NOT NULL AUTO_INCREMENT,
    idProblema int NOT NULL,
    PRIMARY KEY(idLista, idProblema)
);

```

```

);

DROP TABLE IF EXISTS estado;
CREATE TABLE estado(
    idEstado int NOT NULL AUTO_INCREMENT,
    nombreEdo varchar(20),
    PRIMARY KEY(idEstado)
);

DROP TABLE IF EXISTS tipoUsuario;
CREATE TABLE tipoUsuario(
    idTipo int NOT NULL AUTO_INCREMENT,
    nombre varchar(15),
    PRIMARY KEY(idTipo)
);

DROP TABLE IF EXISTS pregunta;
CREATE TABLE pregunta(
    idPregunta int NOT NULL AUTO_INCREMENT,
    nombre varchar(120),
    PRIMARY KEY(idPregunta)
);

INSERT INTO tipoUsuario
VALUES (null, 'Alumno'), (null, 'Asesor'), (null, 'Administrador');

DROP PROCEDURE IF EXISTS agregaEstado;
delimiter :)
CREATE PROCEDURE agregaEstado(in edo varchar(20))
BEGIN
    INSERT INTO estado (nombreEdo) VALUES (edo);
END :)
delimiter ;
call agregaEstado('Aguascalientes');
call agregaEstado('Baja California');
call agregaEstado('Baja California Sur');
call agregaEstado('Campeche');
call agregaEstado('Coahuila');
call agregaEstado('Colima');
call agregaEstado('Chiapas');
call agregaEstado('Chihuahua');
call agregaEstado('Distrito Federal');
call agregaEstado('Durango');
call agregaEstado('Guanajuato');
call agregaEstado('Guerrero');
call agregaEstado('Hidalgo');
call agregaEstado('Jalisco');
call agregaEstado('Mico');
call agregaEstado('Michoac');
call agregaEstado('Morelos');
call agregaEstado('Nayarit');
call agregaEstado('Nuevo Le');
call agregaEstado('Oaxaca');
call agregaEstado('Puebla');
call agregaEstado('Querro');
call agregaEstado('Quintana Roo');
call agregaEstado('San Luis Potos');

```

```

call agregaEstado('Sinaloa');
call agregaEstado('Sonora');
call agregaEstado('Tabasco');
call agregaEstado('Tamaulipas');
call agregaEstado('Tlaxcala');
call agregaEstado('Veracruz');
call agregaEstado('Yucatán');
call agregaEstado('Zacatecas');

```

### Base de datos en SQL Server:

```
USE [Karel_DataBase];
```

```

CREATE TABLE [dbo].[Estado] (
    [IdEstado] INT IDENTITY (1, 1) NOT NULL,
    [NombreEstado] NVARCHAR (20) NULL,
    PRIMARY KEY ([IdEstado] ASC)
);

```

```

CREATE TABLE [dbo].[Problema] (
    [IdProblema] INT IDENTITY (1000, 1) NOT NULL,
    [Nombre] VARCHAR(50) NOT NULL,
    [Descripcion] TEXT NOT NULL,
    [Casos] INT NULL DEFAULT NULL,
    [EvaluacionParcial] BIT NOT NULL,
    [Creador] NVARCHAR(128) NOT NULL,
    [Publico] BIT NOT NULL,
    PRIMARY KEY ([IdProblema] ASC),
    CONSTRAINT [FK_dbo.Creador] FOREIGN KEY ([Creador])
        REFERENCES [dbo].[Usuario] ([Id])
);

```

```

CREATE TABLE [dbo].[Lista] (
    [IdLista] INT IDENTITY (1, 1) NOT NULL,
    [NombreLista] VARCHAR(100) NOT NULL,
    [IdCreador] NVARCHAR(128) NOT NULL,
    PRIMARY KEY ([IdLista] ASC),
    CONSTRAINT [FK_dbo.CreadorLista] FOREIGN KEY ([IdCreador])
        REFERENCES [dbo].[Usuario] ([Id]) ON DELETE CASCADE
);

```

```

CREATE TABLE [dbo].[ListaProblemas] (
    [IdLista] INT NOT NULL,
    [IdProblema] INT NOT NULL,
    PRIMARY KEY ([idLista],[idProblema]),
    CONSTRAINT [FK_dbo.ListaProblemas.Lista] FOREIGN KEY ([idLista])
        REFERENCES [dbo].[Lista] ([IdLista]) ON DELETE CASCADE,
    CONSTRAINT [FK_dbo.ListaProblemas.Problemas] FOREIGN KEY
        ([IdProblema])
        REFERENCES [dbo].[Problema] ([IdProblema]) ON DELETE CASCADE
);

```

```

CREATE TABLE [dbo].[Concurso] (
    [IdConcurso] INT IDENTITY (1, 1) NOT NULL,
    [Nombre] VARCHAR(70) NOT NULL,
    [FechaInicio] DATETIME NOT NULL,

```

```

    [Duracion] INT NOT NULL,
    [Lista] INT NOT NULL,
    [CongelarResultados] INT NULL,
    [Penalizacion] INT NOT NULL,
    [Desempate] BIT NOT NULL,
    [EvaluacionParcial] BIT NOT NULL,
    [IdCreador] NVARCHAR(128) NOT NULL,
    PRIMARY KEY ([IdConcurso]),
    CONSTRAINT [FK_dbo.Concurso.Lista] FOREIGN KEY ([Lista])
        REFERENCES [dbo].[Lista] (IdLista),
    CONSTRAINT [FK_dbo.CreadorConcurso] FOREIGN KEY ([IdCreador])
        REFERENCES [dbo].[Usuario] (Id)
);

CREATE TABLE [dbo].[Evaluacion] (
    [IdEvaluacion] INT IDENTITY (1, 1) NOT NULL,
    [IdProblema] INT NOT NULL,
    [IdUsuario] NVARCHAR(128) NOT NULL,
    [HoraSubida] DATETIME NOT NULL,
    [Resultado] INT NOT NULL,
    PRIMARY KEY ([IdEvaluacion] ASC),
    CONSTRAINT [FK_dbo.Evaluacion.Problemas] FOREIGN KEY ([IdProblema])
        REFERENCES [dbo].[Problema] (IdProblema),
    CONSTRAINT [FK_dbo.Evaluacion.Usuario] FOREIGN KEY ([IdUsuario])
        REFERENCES [dbo].[Usuario] (Id)
);

CREATE TABLE [dbo].[ResultadoConcurso] (
    [IdConcurso] INT NOT NULL,
    [IdEvaluacion] INT NOT NULL,
    PRIMARY KEY ([IdConcurso], [IdEvaluacion]),
    CONSTRAINT [FK_dbo.Resultado.Concurso] FOREIGN KEY ([IdConcurso])
        REFERENCES [dbo].[Concurso] (IdConcurso),
    CONSTRAINT [FK_dbo.Resultado.Evaluacion] FOREIGN KEY ([IdEvaluacion])
        REFERENCES [dbo].[Evaluacion] (IdEvaluacion)
);

CREATE TABLE [dbo].[Participante] (
    [IdConcurso] INT NOT NULL,
    [Participante] NVARCHAR(128) NOT NULL,
    PRIMARY KEY ([IdConcurso], [Participante]),
    CONSTRAINT [FK_dbo.ParticipanteConcurso] FOREIGN KEY (IdConcurso)
        REFERENCES [dbo].[Concurso] (IdConcurso),
    CONSTRAINT [FK_dbo.ParticipanteUsuario] FOREIGN KEY (Participante)
        REFERENCES [dbo].[Usuario] (Id)
);

CREATE TABLE [dbo].[OpcFiltro] (
    [IdOpcFiltro] INT IDENTITY (1, 1) NOT NULL,
    [Nombre] VARCHAR(60) NULL,
    PRIMARY KEY ([IdOpcFiltro] ASC));

CREATE TABLE [dbo].[Filtro] (
    [IdLista] INT NOT NULL,
    [IdOpcion] INT NOT NULL,
    [Valor] VARCHAR (100) NOT NULL,

```

```

PRIMARY KEY ([IdLista], [IdOpcion], [Valor]),
CONSTRAINT [FK_dbo.Filtro.Lista] FOREIGN KEY ([IdLista])
REFERENCES [dbo].[Lista] (IdLista) ON DELETE CASCADE,
CONSTRAINT [FK_dbo.Filtro.Opcion] FOREIGN KEY ([IdOpcion])
REFERENCES [dbo].[OpcFiltro] (IdOpcFiltro)
);

CREATE TABLE [dbo].[Caso](
[IdCaso] INT IDENTITY (1, 1) NOT NULL,
[IdProblema] INT NOT NULL,
[Hash] VARCHAR(MAX) NOT NULL,
[FileNameMundo] VARCHAR(MAX) NOT NULL,
[FileNameSolucion] VARCHAR(MAX) NOT NULL,
PRIMARY KEY([IdCaso]),
CONSTRAINT [FK_dbo.CasoProblema] FOREIGN KEY ([IdProblema])
REFERENCES [dbo].[Problema] (IdProblema) ON DELETE CASCADE
);

ALTER TABLE [dbo].[Usuario] ADD [IdAsesor] NVARCHAR(128);

ALTER TABLE [dbo].[Usuario] ADD CONSTRAINT
[FK_dbo.Asesor] FOREIGN KEY ([IdAsesor])
REFERENCES [dbo].[Usuario]([Id]) ;

ALTER TABLE [dbo].[Usuario] ADD CONSTRAINT
[FK_dbo.UsuarioEstado] FOREIGN KEY ([IdEstado])
REFERENCES [dbo].[Estado]([IdEstado]) ;

INSERT INTO OpcFiltro
(Nombre)
VALUES
('Estado'),
('Asesor'),
('Grado');

INSERT INTO Estado
(NombreEstado)
VALUES
('Aguascalientes'),
('Baja California'),
('Baja California Sur'),
('Campeche'),
('Coahuila'),
('Colima'),
('Chiapas'),
('Chihuahua'),
('Distrito Federal'),
('Durango'),
('Guanajuato'),
('Guerrero'),
('Hidalgo'),
('Jalisco'),
('México'),
('Michoacán'),
('Morelos'),
('Nayarit'),
('Nuevo León>),

```

```
('Oaxaca'),  
( 'Puebla'),  
( 'Querétaro'),  
( 'Quintana Roo'),  
( 'San Luis Potosí'  
( 'Sinaloa'),  
( 'Sonora'),  
( 'Tabasco'),  
( 'Tamaulipas'),  
( 'Tlaxcala'),  
( 'Veracruz'),  
( 'Yucatán'),  
( 'Zacatecas');
```

La base de datos nos permite generar con ayuda de Entity Framework los modelos del sistema, los cuales definen los tipos de datos utilizados en los controladores. Está basado en el mapeo objeto-relacional (más conocido por su nombre en inglés, Object-Relational mapping ORM) es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y la utilización de una base de datos relacional, utilizando un motor de persistencia. En la práctica esto crea una base de datos orientada a objetos virtual, sobre la base de datos relacional. Esto posibilita el uso de las características propias de la orientación a objetos y seguridad al momento de almacenar los datos de forma persistente.

#### Problemas con los módulos existentes.

El proyecto contaba con un módulo existente: el evaluador; el cual se pidió se respetara en su totalidad, este módulo está implementado en lenguaje Visual Basic, para lo cual fue necesario realizar un Windows Communication Foundation (WCF) es un marco de trabajo para la creación de aplicaciones orientadas a servicios. Con WCF, es posible enviar datos como mensajes asíncronos de un extremo de servicio a otro. Un extremo de servicio puede formar parte de un servicio disponible continuamente hospedado por IIS, o puede ser un servicio hospedado en una aplicación. Un extremo puede ser un cliente de un servicio que solicita datos de un extremo de servicio. Los mensajes pueden ser tan simples como un carácter o una palabra que se envía como XML, o tan complejos como una secuencia de datos binarios.

#### Problemas en la migración.

El comité de la olimpiada se comprometió a entregarnos los sets de problemas para poder llevar a cabo la migración de los problemas existentes, sin embargo no nos entregó dicho set, por lo que la migración no pudo llevarse a cabo en el tiempo y la forma planeada lo cual retrasó la implementación de la clasificación ya que no contamos con los problemas suficientes ni con los casos de prueba oficiales. Algunos de estos problemas son soluciones

oficiales a olimpiadas pasadas y requieren de autorización para su uso. De modo que tuvimos que realizar las pruebas con problemas limitados. Esperamos tener el set de problemas con casos y soluciones oficiales en el momento en que se implante el sistema para el uso de la OMI.

# TRABAJO TERMINAL 2013-A003

Sistema web para la evaluación automática de programas de Karel el Robot.

## Pruebas y Resultados.

Para cada módulo del sistema se elaboró un set de pruebas unitarias, las cuales se apoyaron en la clase UnitTesting de ASP.NET. Una prueba unitaria es un método que prueba una unidad de código. Al hablar de una unidad de código nos referimos a un requerimiento y opera bajo las siguientes características:

- Prueba solamente pequeñas cantidades de código: Solamente prueba el código del requerimiento específico.
- Se aísla de otro código y de otros desarrolladores: prueba exclusivamente el código relacionado con el requerimiento y no interfiere con el trabajo hecho por otros desarrolladores.
- Solamente se prueban los puntos finales públicos: Esto principalmente porque los disparadores de los métodos privados son métodos públicos por lo tanto se abarca el código de los métodos privados dentro de las pruebas.
- Los resultados son automatizados: Cuando ejecutamos las pruebas lo podemos hacer de forma individual o de forma grupal. Estas pruebas las hace el motor de prueba y los resultados de los mismos deben de ser precisos con respecto a cada prueba unitaria desarrollada
- Repetible y predecible: No importa el orden y las veces que se repita la prueba, el resultado siempre debe de ser el mismo.
- Son rápidos de desarrollar: Difícilmente una prueba unitaria deba de tomar más de cinco minutos en su desarrollo.

Módulo del Evaluador.

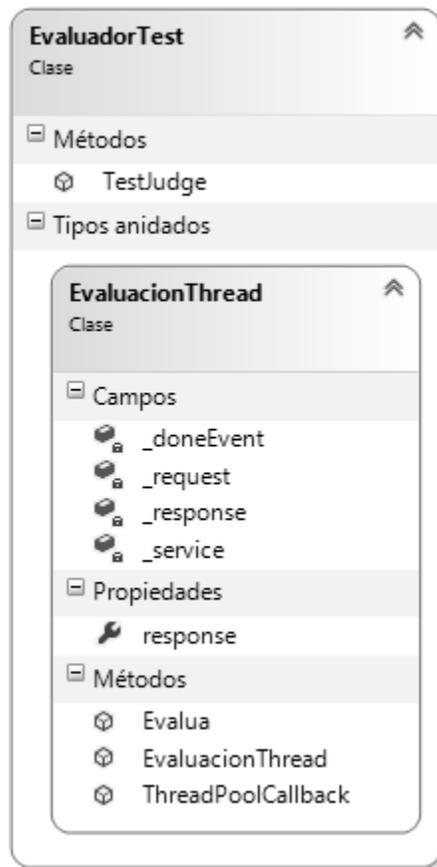


Imagen 18-Clase EvaluadorTest

Método	Descripción
<b>Evalua</b>	Realiza un request al web service del evaluador. Regresa la vista detalles en caso de ejecutarse correctamente.
<b>EvaluacionThread</b>	Es el constructor por defecto de la evaluación tiene como atributos los siguientes elementos: <code>_service</code> , <code>_request</code> y <code>_doneEvent</code> .
<b>ThreadPoolCallback</b>	Realiza el llamado a la función y la guarda como argumento para que en cuanto el hilo esté listo se le envíe.
<b>TestJudge</b>	Envía la solución de un problema y es probada en un determinado número de casos, bajo las siguientes condiciones: <ol style="list-style-type: none"> <li>1. El hash de los casos debe ser el mismo que en el momento en que éstos se subieron al sistema.</li> <li>2. Debe ejecutarse en menos de 200 milisegundos</li> <li>3. La respuesta a todos los casos debe ser correcta</li> </ol>

Tabla 52- Descripción de la clase EvaluadorTest

Condiciones.

*Problema:* Rombos. 15a OMI, Mérida 2010

*Descripción:*

A Karel le gustan los rombos y desea dibujar uno. Para ello cuenta con un número de zumbadores en su mochila y un mundo cuadrado de longitud impar sin zumbadores ni paredes interiores.

La cantidad de zumbadores que tiene es tal que alcanzan de manera exacta para que usándolos todos en montones de 1 zumbador Karel pueda rellenar las 4 esquinas del mundo de modo que quede un rombo sin zumbadores en el centro. Para visualizar mejor a lo que nos referimos, te recomendamos ver las figuras de ejemplo.

### **Problema**

Ayuda a Karel a colocar TODOS los zumbadores de su mochila en el mundo para que obtenga su rombo.

### **Consideraciones**

- Karel inicia en esquina inferior izquierda del mundo con orientación norte.
- Karel inicia con un número múltiplo de 4 de zumbadores en su mochila.
- El mundo inicial no tiene ningún zumbador ni pared interior y siempre será de longitud impar.
- El espacio sin zumbadores que representa el rombo debe quedar centrado.
- No es obligatorio que el rombo toque las paredes exteriores del mundo.
- En ninguna posición Karel deberá dejar más de 1 zumbador
- Karel siempre debe quedar con 0 zumbadores en la mochila.
- No importa la posición ni la orientación final de Karel, sólo los lugares donde dejaste zumbadores.

*Número de casos:* 10

*Lenguaje:* Pascal

*Solución:*

```
iniciar-programa
  define-nueva-instruccion mediaVuelta como inicio
    gira-izquierda;
    gira-izquierda;
  fin;

  define-nueva-instruccion giraDerecha como inicio
    mediaVuelta;
```

```

    gira-izquierda;
fin;

define-nueva-instruccion avanzaEscalon como inicio
  giraDerecha;
  avanza;
  gira-izquierda;
  si frente-libre y no-junto-a-zumbador entonces inicio
    avanza;
    si junto-a-zumbador entonces inicio
      repetir 2 veces inicio
        gira-izquierda;
        avanza;
      fin;
    gira-izquierda;
  fin;
fin
sino inicio
  gira-izquierda;
  avanza;
  mediaVuelta;
fin;
fin;

define-nueva-instruccion avanzaHastaEsquina como inicio
  avanza;
  mientras frente-libre y no-junto-a-zumbador hacer avanza;
  si junto-a-zumbador entonces inicio
    mediaVuelta;
    avanza;
    mediaVuelta;
    si junto-a-zumbador entonces coge-zumbador;
  fin;
fin;

inicia-ejecucion
  mientras algun-zumbador-en-la-mochila hacer inicio
    avanzaHastaEsquina;
    mientras no-junto-a-zumbador hacer inicio
      deja-zumbador;
      avanzaEscalon;
    fin;
  fin;
  apagate;
termina-ejecucion
finalizar-programa

```

## Resultados.

La prueba unitaria fue ejecutada en 10 ocasiones y en cada ejecución demostró resultados consistentes, es decir si ninguno de los casos era alterado superaba la prueba de otro modo la prueba fallaba, en cada ejecución cumplió con el tiempo de ejecución estipulado y la ejecución era correcta cuando se superaban todos los casos, porque el módulo cumplió con las condiciones estipuladas.

## Módulo Gestión de Usuarios.

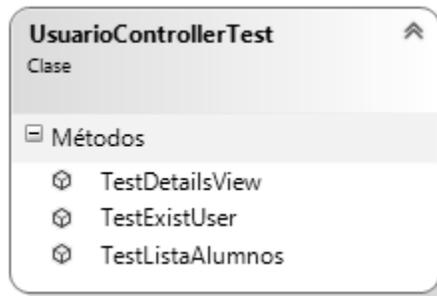


Imagen 19: Clase UsuarioControllerTest

Método	Descripción
<b>TestDetailsView</b>	Muestra la vista de detalles de un determinado usuario bajo las siguientes condiciones: <ol style="list-style-type: none"><li>1. Si el usuario entra a su propio perfil activa la opción de edición.</li><li>2. Si el usuario entra a otro perfil solo puede visualizar la información</li></ol>
<b>TestExistUser</b>	Dado un identificador de usuario devuelve el perfil del mismo bajo las siguientes condiciones: <ol style="list-style-type: none"><li>1. Si el usuario no existe regresa a la pantalla de inicio.</li><li>2. Si el usuario existe devuelve el perfil</li></ol>
<b>TestListaAlumnos</b>	Regresa la lista de alumnos de un asesor bajo las siguientes condiciones: <ol style="list-style-type: none"><li>1. Si el usuario no es de tipo asesor, esta vista no existe y se mantiene en la vista actual</li><li>2. Si el asesor no tiene alumnos devuelve la vista vacía</li><li>3. Si el asesor tiene alumnos devuelve la vista con la lista correspondiente.</li></ol>

Tabla 53- Descripción de la clase UsuarioControllerTest

### Condiciones.

#### *Usuario TestViewDetails :*

- Identificador: 276eb27d-ea92-4860-9860-e552cdd803ac
- Nombre: Ethan Adrián Jiménez Vargas
- Rol: Asesor

#### *Usuario TestExistUser:*

- Identificador: 673ef484-eb2e-48d2-9a48-0db3097a8e7f

- Nombre: Alondra Reyes Carranza
- Rol: Alumno

### Resultados

La prueba fue ejecutada 10 veces y en cada ejecución era aprobada. Para el primer método el usuario entraba a su propio perfil. En el segundo método la prueba era aprobada siempre que no se eliminara al usuario del sistema ya que una de las condiciones era la existencia del mismo. Para el tercer método se mostraba la lista de usuarios del asesor indicado. De modo que las pruebas del módulo cumplieron con las condiciones estipuladas.

Módulo Gestión de problemas.

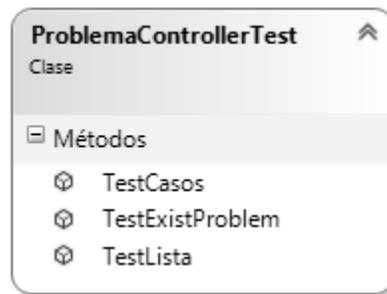


Imagen 20: Clase ProblemaControllerTest

Método	Descripción
<b>TestCasos</b>	Dado el identificador de un problema, muestra la vista para agregar casos, así como sus casos ya existentes. <ol style="list-style-type: none"> <li>1. El problema debe existir para poder mostrar los casos de prueba</li> </ol>
<b>TestExistProblem</b>	Dado el identificador de un problema, muestra su vista de detalles, es decir su descripción y la opción de subir la solución. <ol style="list-style-type: none"> <li>1. El problema debe existir para mostrar la vista</li> <li>2. Si el problema no existe regresa al inicio.</li> </ol>
<b>TestLista</b>	Dado el identificador de una lista devuelve los problemas que la conforman siempre que el usuario tenga acceso. <ol style="list-style-type: none"> <li>1. Si la lista no tiene problemas devuelve la lista vacía</li> <li>2. Si la lista no existe regresa al inicio</li> </ol>

Tabla 54- Descripción de la clase ProblemaControllerTest

### Condiciones.

#### Problema existente:

- Identificador: 1001
- Nombre: Amontonar Zumbadores
- Número de casos: 5

#### Lista existente:

- Identificador: 2
- Nombre: Lista OMI D.F.
- Filtro: Distrito Federal
- Problemas: 1002,1004,1020, 1034,1035

### Resultados.

Las pruebas se ejecutaron 10 veces, en cada ejecución la prueba se superaba siempre y cuando el problema, la lista y los casos siguieran existiendo, de otro modo la prueba fallaba. LA prueba fue ejecutada en un tiempo menor a 250 milisegundos, de modo que el sistema realiza búsquedas y las muestra de forma rápida y sin errores.

### Módulo Gestión de Concursos.

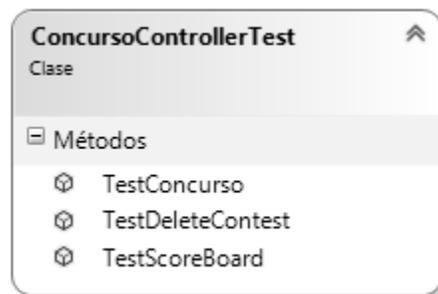


Imagen 21: Clase ConcursoControllerTest

<b>Método</b>	<b>Descripción</b>
<b>TestConcurso</b>	Muestra la vista de detalles de un concurso dado su identificador. <ol style="list-style-type: none"> <li>1. Si no existe se queda en la página de inicio.</li> <li>2. Muestra los problemas sólo si ya inicio el concurso</li> <li>3. Si el concurso finalizo muestra los problemas y los resultados.</li> </ol>
<b>TestDeleteContest</b>	Dado un identificador, elimina todos sus datos y las referencias a los problemas <ol style="list-style-type: none"> <li>1. Si el problema no existe se queda en la página de inicio.</li> <li>2. Si el concurso existe no lo vuelve a mostrar.</li> </ol>
<b>TestScoreBoard</b>	Dado el identificar de un concurso muestra los resultados <ol style="list-style-type: none"> <li>1. Cambia dinámicamente</li> <li>2. Si el concurso finalizo mantiene resultados estáticos.</li> </ol>

Tabla 55-Descripción de la clase ConcursoControllerTest

Condiciones.

Concurso

- Identificador: 2
- Nombre: Concurso OMI D.F.
- Inicio: 02/05/2014 10:30 a.m.
- Lista: Lista OMI D.F.

Resultados.

Las pruebas se ejecutaron 10 veces en cada ejecución las pruebas fueron aceptadas hasta el momento en el que el concurso fue eliminado, ya que a partir de ese momento ya no había datos que cargar. El tiempo ejecución del Scoreboard es de 15 milisegundos, lo cual indica que el procesamiento de datos dinámicos es rápido.

## Conclusiones.

El objetivo principal de éste trabajo terminal era desarrollar un sistema web para la gestión y publicación de problemas a desarrollar bajo el lenguaje Karel. Así como un módulo para la gestión de usuarios, la evaluación automática de los programas resueltos por estos y la administración de concursos virtuales. Este objetivo se quería lograr debido a las deficiencias presentadas por el sitio que actualmente utiliza la Olimpiada Mexicana de Informática y a la importancia que el lenguaje Karel tiene para los participantes, ya que en México no existe un sitio que permita la creación de concursos mediante lenguaje Karel, lo cual dificulta el proceso de selección de los participantes en diferentes etapas del proceso, pero ahora, el sistema implementado proporcionará los medios necesarios para llevar a cabo dichos procesos.

Una vez finalizado el desarrollo del sistema, se realizaron diversas pruebas mediante el uso de herramientas proporcionadas por ASP.NET, con lo cual podemos garantizar su funcionamiento ante un gran número de usuarios concurrentes. Esto, por una parte, es debido a la implementación de un servicio web para la evaluación de los problemas, en el cual se optimizó al máximo las características de multiprocesamiento de los equipos actuales y permite la posibilidad de distribuir la carga de trabajo en diferentes equipos.

En el sistema actual, se cuenta únicamente con una lista de problemas, los cuales tienen una clasificación de acuerdo al criterio de su autor, que no siempre es objetiva, por lo que el sistema desarrollado proporciona los medios para realizar una clasificación basada en estadísticas reales, así como la posibilidad de generar listas personalizadas de problemas.

Además, al ofrecer la opción de comunicar el sitio con redes sociales, ofrecemos a la OMI la posibilidad de darse a conocer a un mayor número de usuarios, fomentando el aprendizaje de la programación de manera sencilla.

Todas estas funcionalidades, en conjunto, permitirán a los aspirantes a participar en la OMI, y en general, a las personas interesadas en mejorar su lógica de programación, a contar con los recursos necesarios para su preparación y superación, tanto profesional como personal.

## Glosario

**ALGORITMO** Conjunto ordenado y finito de operaciones que permite hallar la solución de un problema. Método y notación en las distintas fórmulas del cálculo. El algoritmo constituye un método para resolver un problema mediante una secuencia de pasos a seguir.

**AUTENTICAR** Autorizar o legalizar alguna cosa: en este caso hace referencia a permitir al usuario visualizar cierta sección del menú.

**BACKLOG** La lista de objetivos/requisitos priorizada representa la visión y expectativas del cliente respecto a los objetivos y entregas del producto o proyecto. El cliente es el responsable de crear y gestionar la lista.

**COMI** Abreviación -Comité de la Olimpiada Mexicana de Informática.

**COMPILADOR** Un compilador es un programa informático que traduce un programa escrito en un lenguaje de programación a otro lenguaje de programación.

**CONTROLADOR** Responde a eventos (usualmente acciones del usuario) e invoca peticiones al 'modelo' cuando se hace alguna solicitud sobre la información (por ejemplo, editar un documento o un registro en una base de datos). También puede enviar comandos a su 'vista' asociada si se solicita un cambio en la forma en que se presenta el 'modelo'.

**ENTITY FRAMEWORK** Entity Framework (EF) es un asignador objeto-relacional que permite a los desarrolladores de .NET trabajar con datos relacionales usando objetos específicos del dominio. Elimina la necesidad de la mayor parte del código de acceso a datos que los desarrolladores suelen tener que escribir.

**EVALUADOR** Abreviación -que evalúa algo en sus aspectos técnicos, en este caso se evalúan programas codificados por los alumnos en lenguaje karel.

IOI Abreviación -Del inglés International Olympiad in Informatics.

KAREL Es un lenguaje de programación para programar a un robot (Karel) mediante instrucciones sencillas y bien estructuradas, fáciles de entender y utilizar sin la necesidad de manejar algún otro lenguaje de programación. Karel vive en su propio mundo, formado por calles, avenidas y bardas por las que deberá trasladarse para recoger trompos, guardarlos y llevarlos a otro lugar, dependiendo de cuál sea su objetivo.

MODELO Es la representación de la información con la cual el sistema opera, por lo tanto gestiona todos los accesos a dicha información, tanto consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación.

PERFORMANCE En los sistemas informáticos el performance se refiere al tiempo de respuesta que se obtiene cuando varios usuarios interactúan de forma concurrente.

PROCESO Un proceso es una secuencia de pasos dispuesta con algún tipo de lógica que se enfoca en lograr algún resultado específico.

SCOREBOARD Se refiere al listado de puntuación de los participantes de un concurso.

SCRUM SCRUM es un modelo de referencia que define un conjunto de prácticas y roles, y que puede tomarse como punto de partida para definir el proceso de desarrollo que se ejecutará durante un proyecto.

SIMULADOR Un simulador es un aparato, por lo general informático, que permite la reproducción de un sistema. Los simuladores reproducen sensaciones y experiencias que en la realidad pueden llegar a suceder.

SPRINT Ciclo básico de desarrollo que permite alcanzar el cumplimiento de un objetivo a corto plazo.

**TEST** Investigaciones empíricas y técnicas cuyo objetivo es proporcionar información objetiva e independiente sobre la calidad del producto a la parte interesada o stakeholder. Es una actividad más en el proceso de control de calidad.

**VISTA** Presenta el 'modelo' (información y lógica de negocio) en un formato adecuado para interactuar (usualmente la interfaz de usuario) por tanto requiere de dicho 'modelo' la información que debe representar como salida.

**WEBSERVICE** Un servicio web (en inglés, Web Service o Web services) es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos.

**ZUMBADOR** Un zumbador es una forma de marca que Karel puede escuchar sólo cuando se encuentra en la misma esquina que el zumbador.

## Referencias generales.

- [1] «Olimpiada de Informática,» 2004. [En línea]. Available: [www.olimpiadadeinformatica.org.mx/inf](http://www.olimpiadadeinformatica.org.mx/inf). [Último acceso: 20 Diciembre 2012].
- [2] «International Olympiad in Informatics,» [En línea]. Available: <http://www.ioinformatics.org/>. [Último acceso: enero 2012].
- [3] M. I. Romero, «Karel titlán,» cmirg, 2006. [En línea]. Available: <http://www.cmirg.com/kareltitlan/Pantallas/Inicio.aspx>. [Último acceso: 20 Diciembre 2012].
- [4] M. Resnick, «Video Ted,» 2012. [En línea]. Available: [video.ted.com/talk/podcast/2012X/None/MitchResnick\\_2012X.mp4](http://video.ted.com/talk/podcast/2012X/None/MitchResnick_2012X.mp4). [Último acceso: julio 2013].
- [5] E. Duarte, «Capacity Academy,» IT Academy, 2012. [En línea]. Available: <http://blog.capacityacademy.com/2012/05/25/por-que-aprender-a-programar/>. [Último acceso: Julio 2013].
- [6] L. Chilcott, Dirección, *What Most Schools Don't Teach - Short Film*. [Película]. U.S.A.: Code.org, 2011.
- [7] A. F. Szpiniak, «Revista Teyet,» 2010. [En línea]. Available: [http://teyet-revista.info.unlp.edu.ar/files/No1/09\\_Ensenanza\\_de\\_la\\_programacion.pdf](http://teyet-revista.info.unlp.edu.ar/files/No1/09_Ensenanza_de_la_programacion.pdf). [Último acceso: 2013].
- [8] R. Pattis, «Karel the Robot,» de *Karel the Robot: A Gentle Introduction to the Art of Programming with Pascal*, New York, John Wiley & Sons, Inc, 1981, p. 106.
- [9] J. Sutherland, «jeffsutherland.com,» [En línea]. Available: <http://jeffsutherland.com/ScrumPapers.pdf>. [Último acceso: 7 Enero 2012].
- [10] «Biblioteca Digital,» 2002. [En línea]. Available: <http://bibliotecadigital.conevyt.org.mx/colecciones/documentos/somece2002/Grupo4/Oviedo.pdf>.

# TRABAJO TERMINAL 2013-A003

Sistema web para la evaluación automática de programas de Karel el Robot.

## Anexo A: Manual de Usuario

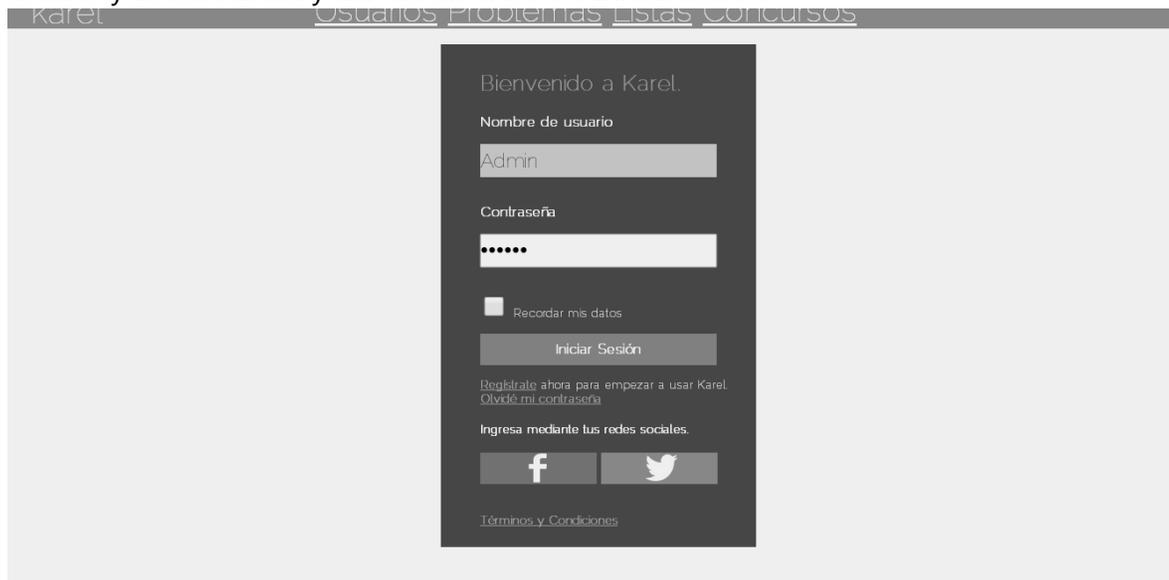
### Inicio de sesión

#### 1. Inicio de sesión mediante una cuenta de Karel

1. En la página principal, en la parte superior derecha acceder al vínculo 'Iniciar Sesión' para poder ingresar a la ventana de acceso del sistema.



- Ingresar los datos correspondientes al formulario, en este caso el nombre de usuario y la contraseña y dar clic en el botón 'Iniciar Sesión'.



The screenshot shows a login form for the Karel website. At the top, there is a navigation bar with the Karel logo and links for 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. The login form itself is a dark grey box with the following elements: a greeting 'Bienvenido a Karel.', a 'Nombre de usuario' field containing 'Admin', a 'Contraseña' field with masked characters, a 'Recordar mis datos' checkbox, an 'Iniciar Sesión' button, a registration prompt 'Regístrate ahora para empezar a usar Karel. Olvidé mi contraseña', social media links for Facebook and Twitter, and a 'Términos y Condiciones' link.

- El sistema redirecciona a la página de inicio una vez que ha sido exitoso el acceso al sistema y en la parte superior derecha se puede visualizar el nombre del usuario así como la opción 'Salir' para finalizar la sesión.



### 1.1 Error en el inicio de sesión – No se han enviado datos al servidor

1. Cuando el usuario no ha escrito ningún dato, el sistema notifica el error.

The screenshot shows the Karel login interface. At the top left, there is a logo with "[ ]" above "karel". At the top right, there is a link for "Iniciar sesión". Below the logo, there are navigation links: "Usuarios", "Problemas", "Listas", and "Concursos". The main content area is a dark grey box with the following elements:

- Greeting: "Bienvenido a Karel."
- Form field: "Nombre de usuario" with an empty input box.
- Error message: "El campo Nombre de usuario es obligatorio."
- Form field: "Contraseña" with an empty input box.
- Error message: "El campo Contraseña es obligatorio."
- Checkbox: "Recordar mis datos" (unchecked).
- Button: "Iniciar Sesión"
- Text: "Regístrate ahora para empezar a usar Karel. [Olvídate mi contraseña](#)"
- Text: "Ingresa mediante tus redes sociales."
- Social media icons: Facebook and Twitter.

### 1.2 Error en el inicio de sesión – Algún dato proporcionado es incorrecto

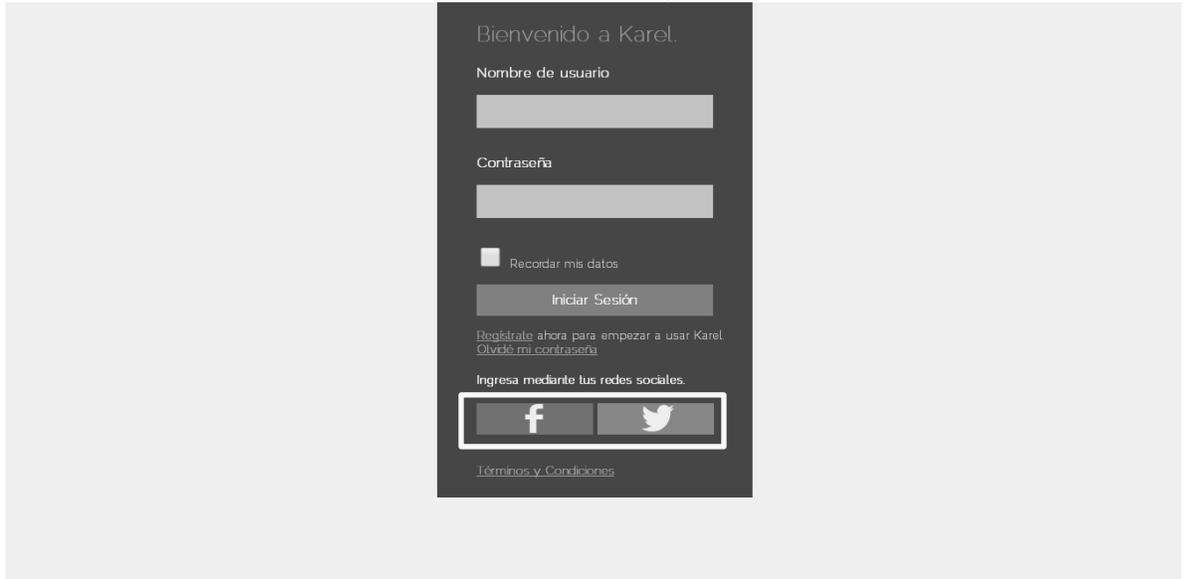
1. Cuando el usuario ha proporcionado un dato incorrecto, el sistema notifica el error.

The screenshot shows the Karel login interface with an error message. The layout is identical to the previous screenshot, but with the following changes:

- Error message: "Error al iniciar sesión, usuario o contraseña incorrecto."
- Form field: "Nombre de usuario" with the text "Admin" entered.
- Form field: "Contraseña" with an empty input box.
- Button: "Iniciar Sesión"
- Text: "Regístrate ahora para empezar a usar Karel. [Olvídate mi contraseña](#)"
- Text: "Ingresa mediante tus redes sociales."
- Social media icons: Facebook and Twitter.

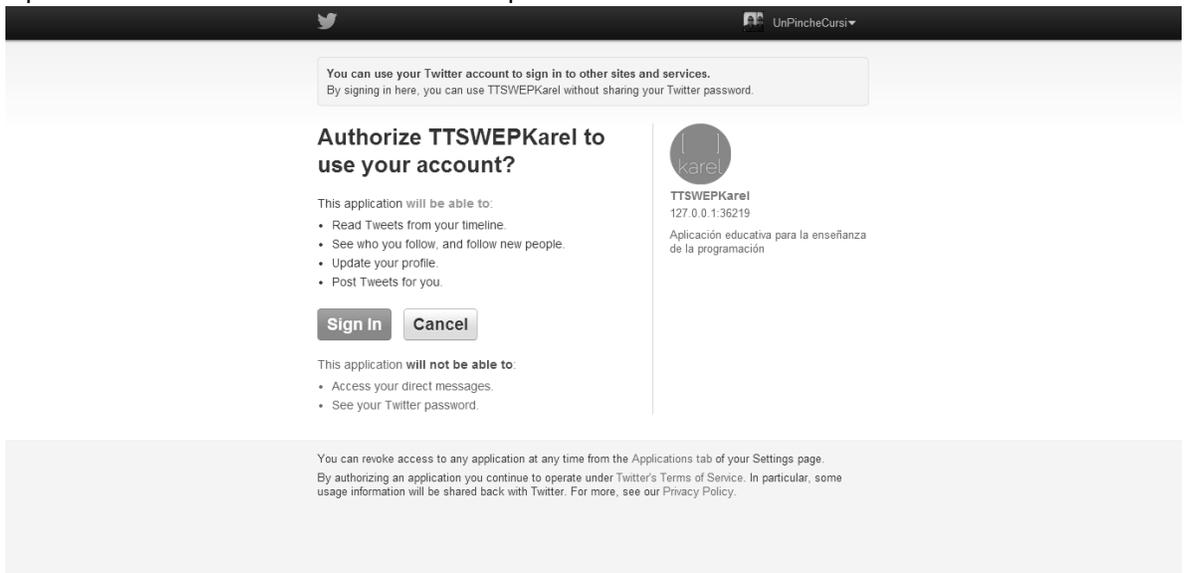
## 2. Inicio de sesión mediante un proveedor externo

1. Dar clic en el botón de Facebook o Twitter en la página de inicio de sesión.



### 2.1 Ingreso al sistema mediante Twitter

1. Aprobar la autorización de vinculación para el uso de la cuenta dentro del sistema.



- Una vez aprobada la vinculación, se es redirigido al sistema y éste muestra una página que lleva a cabo el registro mediante dicha red, el nombre de usuario es tomado directamente de la cuenta de Twitter y solamente se elige el tipo de cuenta.



The screenshot shows a dark-themed registration form titled "Asocia tu cuenta de Twitter." The form contains the following elements:

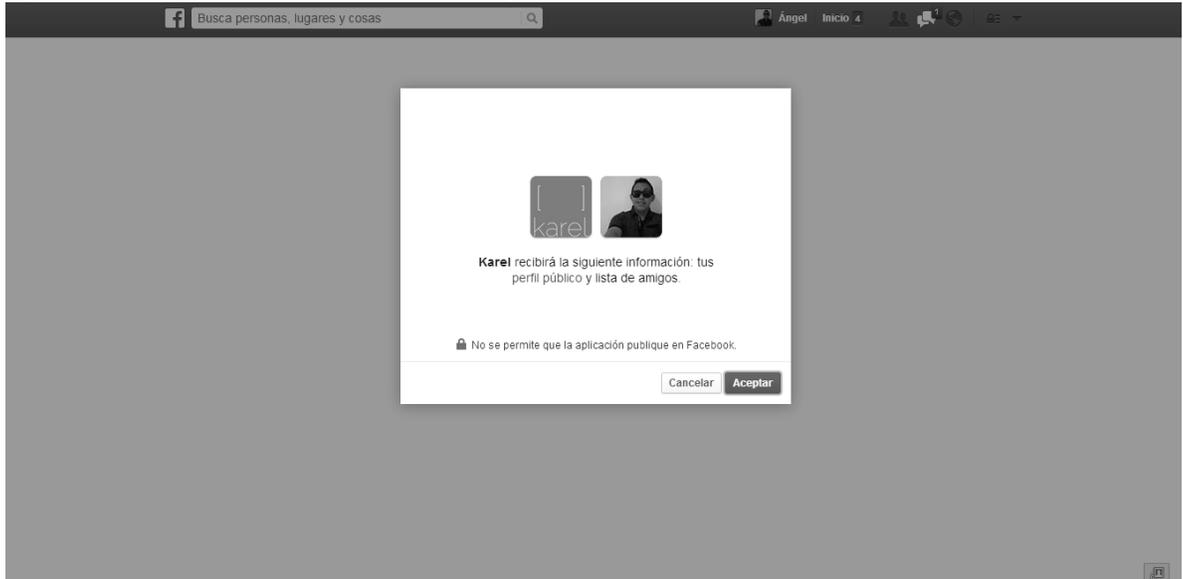
- Header: "karel" logo and navigation links: "Usuarios", "Problemas", "Listas", "Concursos". A "Iniciar sesión" link is in the top right.
- Message: "Se ha asociado tu cuenta de **Twitter** correctamente. Por favor ingresa un nombre de usuario para Karel."
- Field: "Nombre de usuario" with the value "UnPincheCursi".
- Field: "Tipo de usuario" with a dropdown menu showing "Alumno".
- Button: "Registrarse".

- El sistema redirecciona a la página de inicio una vez que ha sido exitoso el acceso al sistema y en la parte superior derecha se puede visualizar el nombre del usuario así como la opción 'Salir' para finalizar la sesión.



## 2.2 Ingreso al sistema mediante Facebook

1. Aprobar la autorización de vinculación para el uso de la cuenta dentro del sistema.



*Cuando se inicia sesión a través de una red social por primera vez, el registro a esa cuenta es llevado a cabo dentro del sistema.*

2. Una vez aprobada la vinculación, se es redirigido al sistema y éste muestra una página que lleva a cabo el registro mediante dicha red, el nombre de usuario es tomado directamente de la cuenta de Facebook y solamente se elige el tipo de cuenta.



3. El sistema redirecciona a la página de inicio una vez que ha sido exitoso el acceso al sistema y en la parte superior derecha se puede visualizar el nombre del usuario así como la opción 'Salir' para finalizar la sesión.



### 3. Restaurar contraseña

1. En la página de Inicio de Sesión, dar clic al vínculo 'Olvidé mi contraseña'.



2. Ingresar los campos que solicita el sistema, en este caso son el nombre de usuario y el correo electrónico registrado para el posterior envío de un correo electrónico.

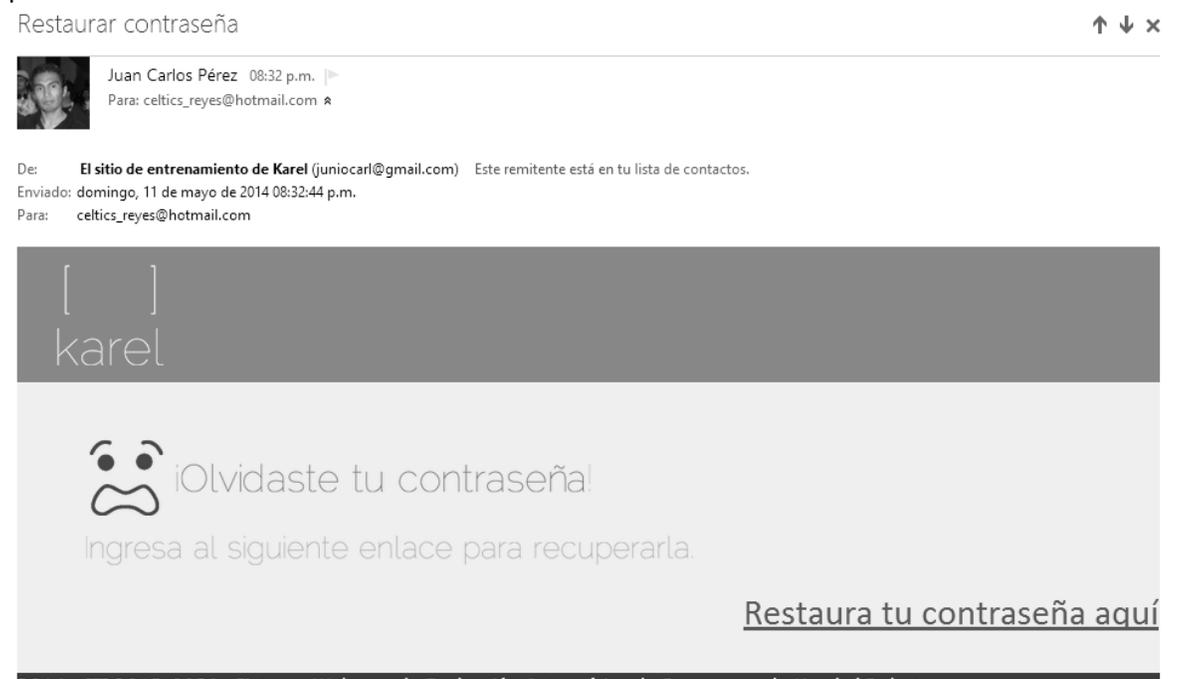


3. El sistema notifica que el vínculo ha sido enviado al correo del usuario.



The screenshot shows the Karel website's password reset interface. At the top, there is a navigation bar with the Karel logo on the left and links for 'Usuarios', 'Problemas', 'Listas', and 'Concursos' in the center. A 'Iniciar sesión' link is located in the top right corner. The main content area features a dark grey box with the following text: 'Restaurar Contraseña.', 'El vínculo ha sido enviado.', 'Este método no funciona si te registraste mediante Facebook o Twitter y no creaste una contraseña local.', and 'Se te enviará un enlace para que puedas generar una nueva contraseña.' Below this text are two input fields labeled 'Usuario' and 'Correo electrónico', followed by a button labeled 'Enviar enlace a mi correo'.

4. En la bandeja de entrada del proveedor de correo se notifica el vínculo que permite llevar a cabo la restauración de la contraseña.



5. El vínculo direcciona a una página que permite generar una nueva, en el cual será necesario ingresar la nueva contraseña y confirmarla.

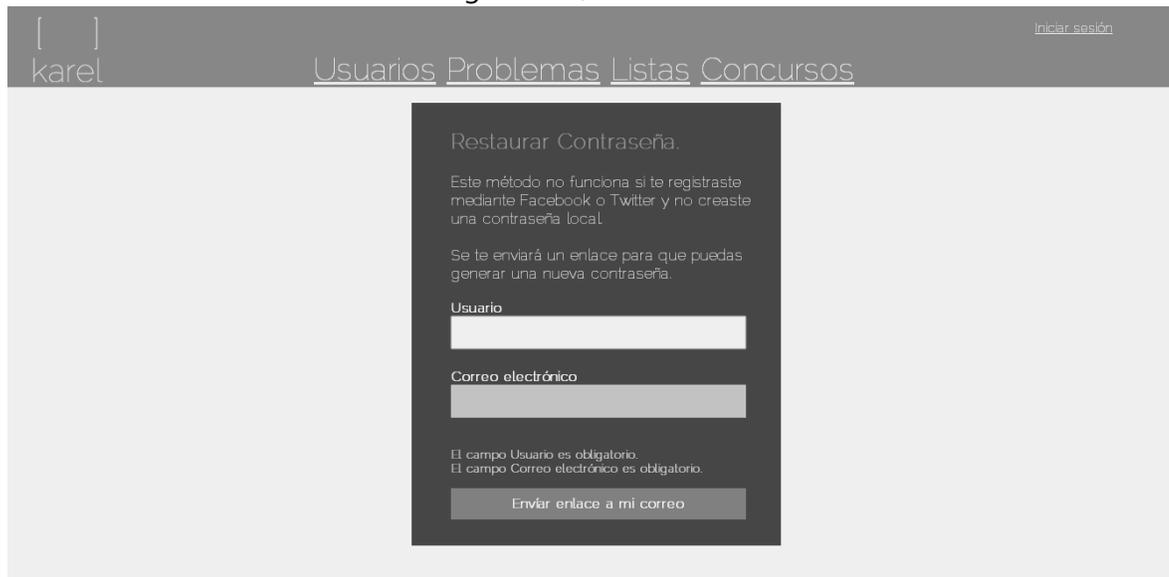
The screenshot shows a web page with a dark header. On the left, there is a logo consisting of a square with a vertical line and the text 'karel'. On the right, there is a link 'Iniciar sesión'. In the center of the header, there are navigation links: 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. The main content area is light gray and contains a dark gray modal box. The modal box has the title 'Establecer Nueva Contraseña.' and the subtitle 'Establecer contraseña'. It contains two input fields: 'Contraseña' and 'Confirmar contraseña', both with masked characters (dots). Below the fields is a button labeled 'Establecer contraseña'.

6. El sistema redirecciona a la página de inicio de sesión, para poder acceder al sistema ahora con la nueva contraseña.

The screenshot shows a web page with a dark header. On the left, there is a logo consisting of a square with a vertical line and the text 'karel'. On the right, there is a link 'Iniciar sesión'. In the center of the header, there are navigation links: 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. The main content area is light gray and contains a dark gray modal box. The modal box has the title 'Bienvenido a Karel.' and the subtitle 'Nombre de usuario'. It contains two input fields: 'Nombre de usuario' and 'Contraseña'. Below the fields is a checkbox labeled 'Recordar mis datos' and a button labeled 'Iniciar Sesión'. Below the button, there is a link 'Regístrate ahora para empezar a usar Karel.' and a link 'Olvidé mi contraseña'. At the bottom, there is a section 'Ingresa mediante tus redes sociales.' with icons for Facebook and Twitter. At the very bottom, there is a link 'Términos y Condiciones'.

### 3.1 Error en la restauración de contraseña – No se han enviado datos al servidor

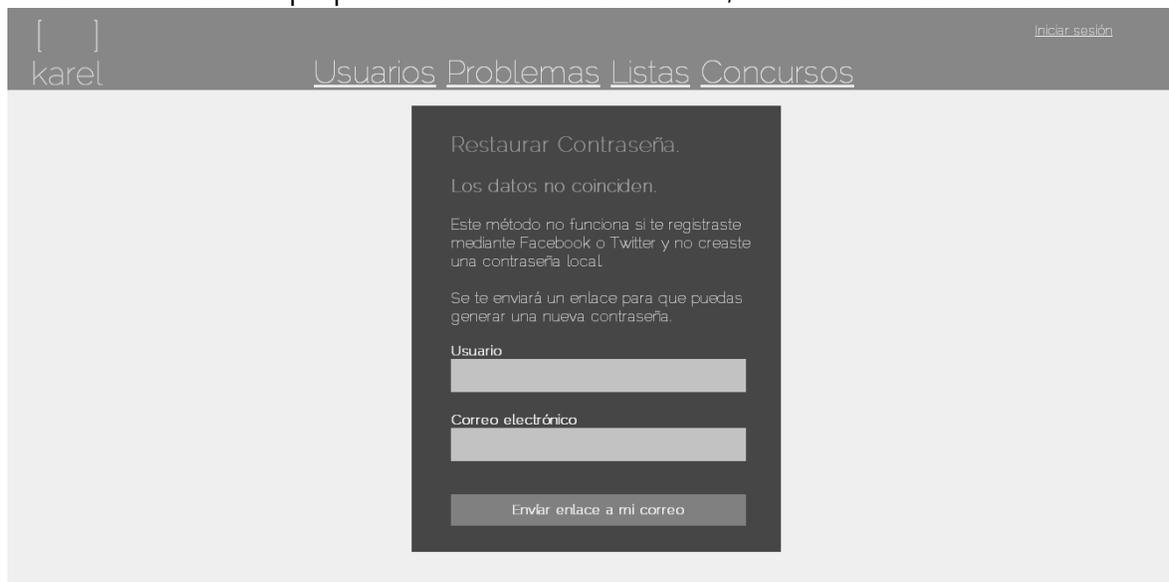
1. Cuando el usuario no ha escrito ningún dato, el sistema notifica el error.



The screenshot shows a web application interface with a dark header. On the left, there is a logo consisting of two brackets [ ] above the text 'karel'. On the right, there is a link 'Iniciar sesión'. In the center of the header, there are navigation links: 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. Below the header, a dark modal box is centered on the page. The modal has the title 'Restaurar Contraseña.' and contains the following text: 'Este método no funciona si te registraste mediante Facebook o Twitter y no creaste una contraseña local.' and 'Se te enviará un enlace para que puedas generar una nueva contraseña.' Below this text are two input fields: 'Usuario' and 'Correo electrónico'. At the bottom of the modal, there is a button labeled 'Enviar enlace a mi correo'. The background of the page is light gray.

### 3.2 Error en la restauración de contraseña – Algún dato proporcionado es incorrecto

1. Cuando el usuario ha proporcionado un dato incorrecto, el sistema notifica el error.



The screenshot shows the same web application interface as in 3.1. The dark modal box now displays an error message: 'Los datos no coinciden.' Below this message, the same text and instructions are present: 'Este método no funciona si te registraste mediante Facebook o Twitter y no creaste una contraseña local.' and 'Se te enviará un enlace para que puedas generar una nueva contraseña.' The input fields for 'Usuario' and 'Correo electrónico' are visible, and the 'Enviar enlace a mi correo' button is at the bottom. The background remains light gray.

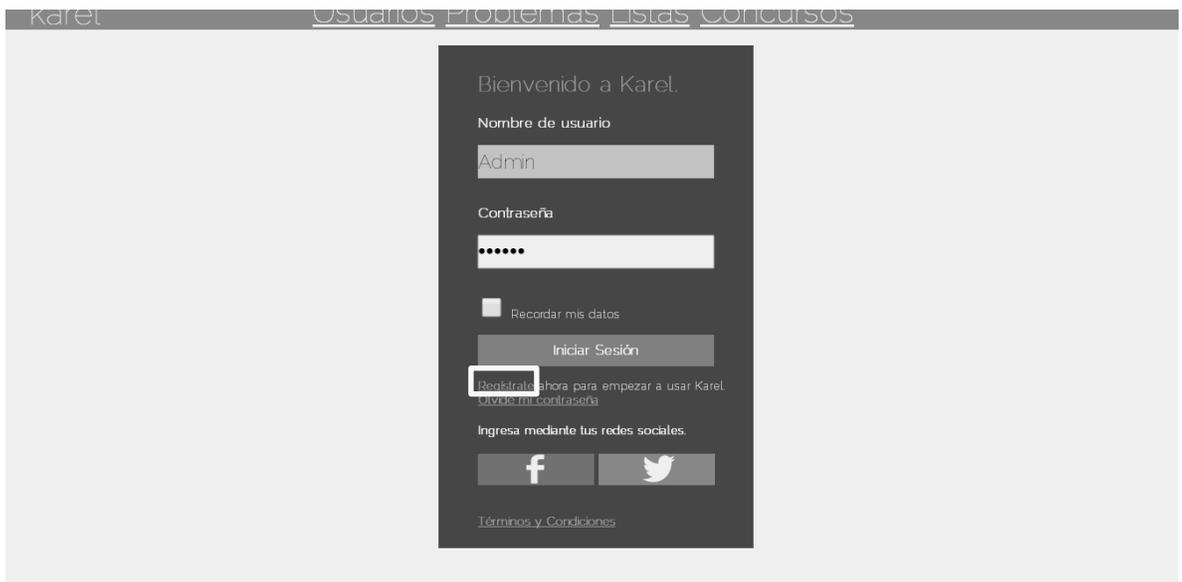
## Registro

### 1. Registro de usuario

1. En la página principal, dar clic en el link del tercer círculo (Color azul) 'Regístrate'.



2. O a través de la página de inicio de sesión dar clic en el vínculo 'Regístrate ahora para empezar a usar Karel.'.



3. El sistema muestra un formulario con los campos necesarios para el registro, una vez completado dar clic al botón 'Regístrate'.

The screenshot shows the registration page of the Karel system. The header includes the Karel logo, navigation links for 'Usuarios', 'Problemas', 'Listas', and 'Concursos', and a 'Iniciar sesión' link. The main content area features a dark registration form with the following fields: 'Usuario' (filled with 'Angel'), 'Correo electrónico' (filled with 'beltics\_reyes@hotmail.com'), 'Contraseña' (masked with dots), 'Confirmar contraseña' (masked with dots), and 'Tipo de cuenta' (a dropdown menu set to 'Asesor'). A 'Regístrate' button is located at the bottom of the form and is highlighted with a white border.

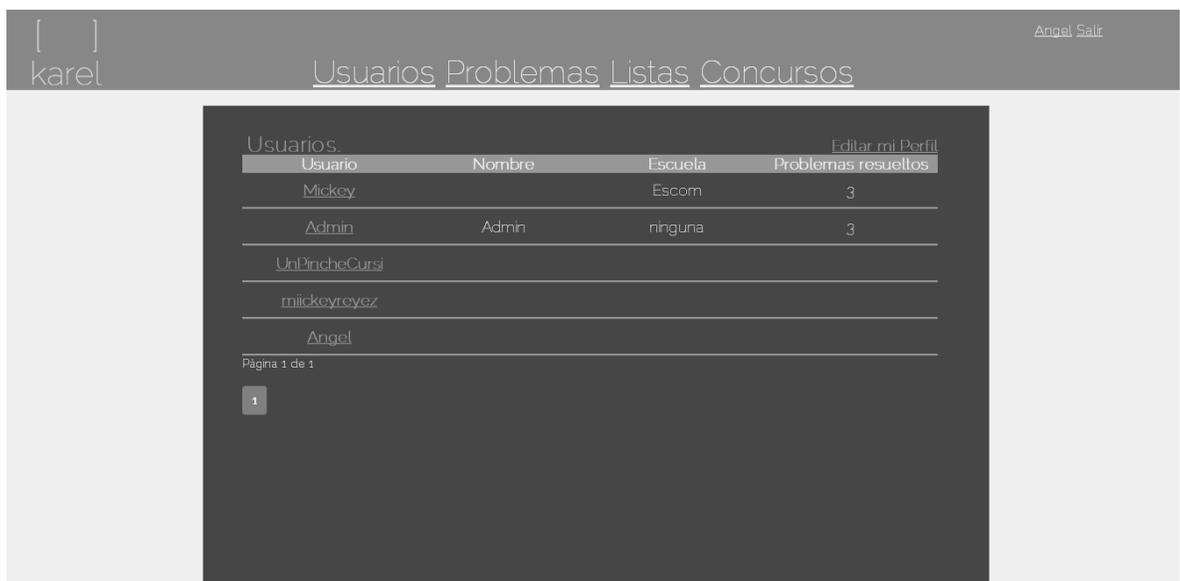
4. El sistema redirecciona al perfil del usuario e inicia sesión automáticamente, además muestra la leyenda 'Confirma primero tu cuenta'.

The screenshot shows the user profile page in the Karel system. The header includes the Karel logo, navigation links for 'Usuarios', 'Problemas', 'Listas', and 'Concursos', and a 'Angel Salir' link. A dark banner at the top of the page reads 'Confirma primero tu cuenta'. The main content area is divided into two columns. The left column, titled 'Editar perfil', contains a 'Volver' link and the following fields: 'Usuario: Angel', 'Nombre', 'Fecha de nacimiento' (01/01/0001), 'Estado' (Aguascalientes), 'Escuela', 'Nivel' (Primaria), 'Grado' (0), 'Duración del grado' (Trimestre), 'Correo electrónico' (beltics\_reyes@hotmail.com), and 'Género' (Masculino). The right column contains three sections: 'Actualizar imagen de perfil' with a 'Subir imagen' button, 'Tipo de cuenta' with a 'Configurar' link, and 'Eliminar perfil' with an 'Eliminar' link.

5. Para finalizar el registro, es necesario confirmar la cuenta a través del vínculo 'Confirmar tu cuenta aquí' del mensaje enviado al correo electrónico proporcionado.



6. Una vez que se ha dado clic, se valida la cuenta y se dirige al sistema donde se muestra la lista de todos los usuarios que han sido registrados.



### 1.1 Error al crear cuenta – Nombre de usuario ya existente.

1. Cuando el usuario desea crear una cuenta con un nombre de usuario previamente registrado, el sistema notifica el error.



The screenshot shows the registration page of the Karel system. The header includes the Karel logo, a navigation menu with links for 'Usuarios', 'Problemas', 'Listas', and 'Concursos', and a link for 'Iniciar sesión'. The registration form is centered and contains the following fields and messages:

- Header: 'Registro. ¡Únete a Karel!'.
- 'Usuario' field: Contains 'Angel'. Below it, a message reads 'El nombre de usuario ya existe.'
- 'Correo electrónico' field: Contains 'mickeyreyez@gmail.com'.
- 'Contraseña' field: An empty password input field.
- 'Confirmar contraseña' field: An empty confirmation password input field.
- 'Tipo de cuenta' dropdown: Set to 'Alumno'.

### 1.2 Error al crear cuenta – Correo electrónico ya existente.

1. Cuando el usuario desea crear una cuenta con un correo electrónico previamente registrado, el sistema notifica el error.

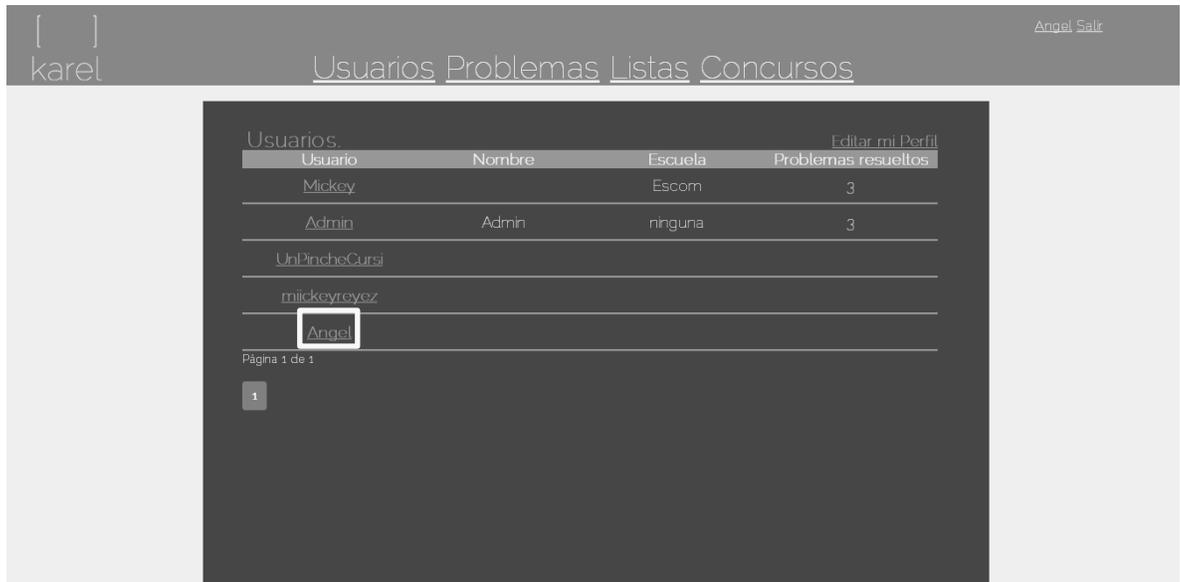


The screenshot shows the registration page of the Karel system, similar to the previous one. The header and navigation menu are identical. The registration form contains the following fields and messages:

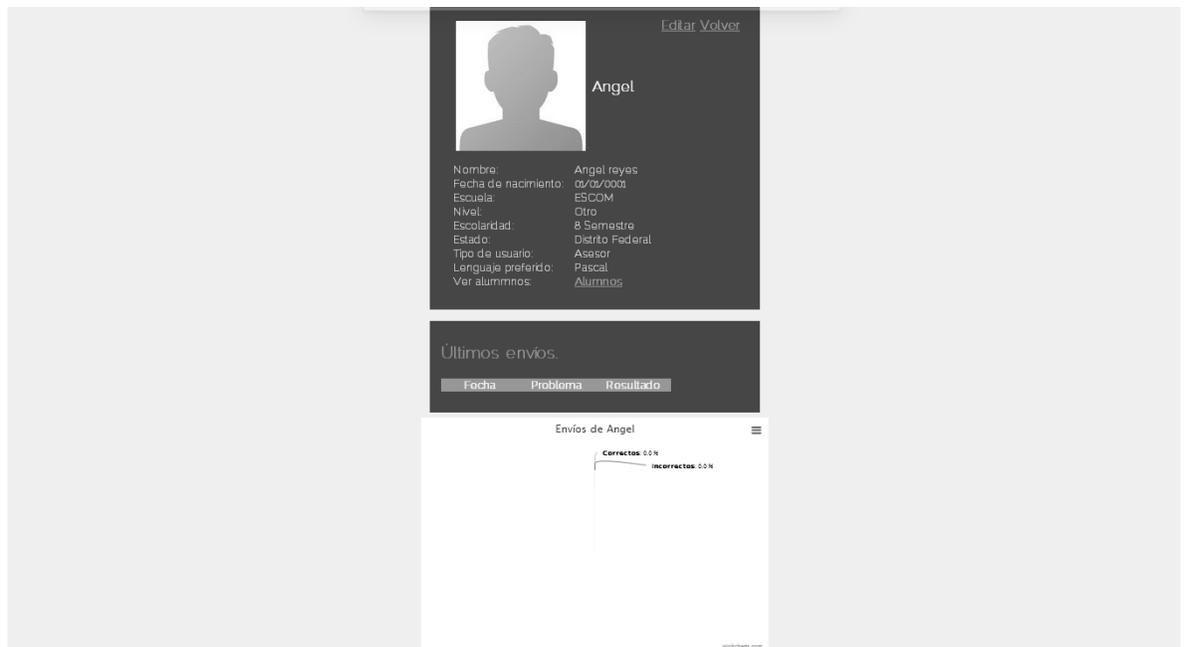
- Header: 'Registro. ¡Únete a Karel!'.
- 'Usuario' field: Contains 'AngelR'.
- 'Correo electrónico' field: Contains 'celtics\_reyes@hotmail.com'. Below it, a message reads 'El correo ya ha sido registrado.'
- 'Contraseña' field: An empty password input field.
- 'Confirmar contraseña' field: An empty confirmation password input field.
- 'Tipo de cuenta' dropdown: Set to 'Alumno'.

2. Visualizar perfil del usuario, últimos envíos y estadísticas de los envíos.

1. En la lista de usuarios, dar clic en el nombre del usuario.



2. El sistema mostrará el perfil del usuario con la información correspondiente, los últimos envíos así como las estadísticas de los envíos.



### 3.1 Error al actualizar información – Campos nulos o datos inválidos

1. Cuando el usuario no ha escrito algún dato o el tipo de dato no coincide a lo requerido, el sistema notifica el error.

The screenshot displays a user profile editing interface. The left sidebar contains the following fields:

- Editar perfil** [Volver](#)
- Usuario: Angel
- Nombre**: [Empty text input]
- Fecha de nacimiento**: 01/01/0001
- Estado**: Aguascalientes
- Escuela**: [Empty text input]  
Campo obligatorio
- Nivel**: Primaria
- Grado**: 0  
El valor debe estar entre 1 y 10
- Duración del grado**: Trimestre
- Correo electrónico**: celtics\_reyeshotmail.com  
Formato inválido
- Género**: Masculino
- Lenguaje preferido**: Pascal

The right sidebar contains the following sections:

- Actualizar imagen de perfil**  
Puedes subir una imagen en formato jpg como fotografía de perfil.  
[Seleccionar Archivo](#) No se eligió archivo  
[Subir imagen](#)
- Tipo de cuenta** [Configurar](#)  
Puedes configurar el tipo de tu cuenta, cambiar entre alumno o asesor
- Eliminar perfil** [Eliminar](#)

#### 4. Actualizar foto de perfil del usuario

1. En la barra lateral derecha, en el recuadro 'Actualizar imagen de perfil', seleccionar la imagen a subir en formato .JPG y dar clic al botón 'Subir imagen'.



2. La imagen se actualiza y una vez cargada, se muestra en el recuadro.



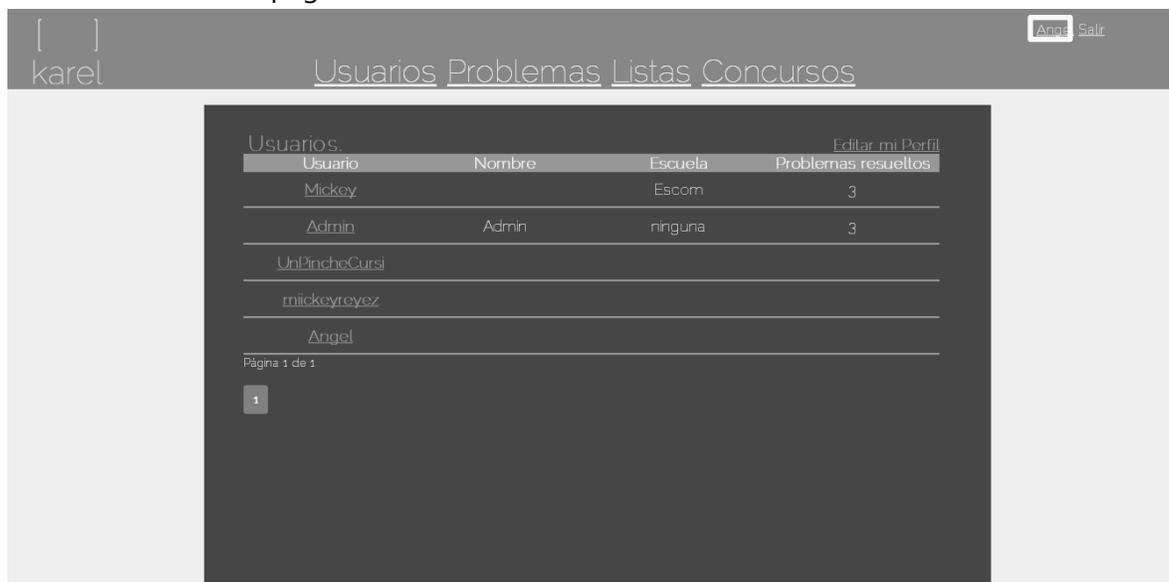
#### 4.1 Error al subir imagen – Formato inválido

1. Cuando se elige una foto para el perfil en un formato inválido, se notifica el error de formato incorrecto.



#### 5. Cambiar contraseña del usuario

1. En la cabecera de la página dar clic en el nombre del usuario.



2. El sistema mostrará un formulario donde se deberá agregar la contraseña actual, así como la nueva contraseña y su correspondiente confirmación, posteriormente dar clic en el botón 'Cambiar Contraseña'.

[ ]  
karel

Angel Salir

[Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

Cambiar Contraseña.

Angel

Contraseña actual  
.....

Nueva contraseña  
.....

Confirmar contraseña  
.....

Cambiar Contraseña

3. Una vez realizado el cambio, el sistema notifica que ha sido exitoso.

[ ]  
karel

Angel Salir

[Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

Cambiar Contraseña.

Tu contraseña ha sido cambiada.

Angel

Contraseña actual  
.....

Nueva contraseña  
.....

Confirmar contraseña  
.....

Cambiar Contraseña

### 5.1 Error al cambiar contraseña – Contraseña actual incorrecta

1. Cuando la contraseña actual es incorrecta, se notifica el error.



The screenshot shows a web application interface with a dark header. On the left, there is a user profile icon and the name 'karel'. On the right, there is a link 'Angel Salir'. The main navigation menu includes 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. A modal dialog titled 'Cambiar Contraseña.' is centered on the screen. It displays the user's name 'Angel' and three input fields: 'Contraseña actual', 'Nueva contraseña', and 'Confirmar contraseña'. Below the input fields, a red error message reads 'Contraseña actual incorrecta.' and a 'Cambiar Contraseña' button is visible at the bottom of the modal.

### 5.2 Error al cambiar contraseña – Contraseña nueva y su confirmación no coinciden.

1. Cuando la nueva contraseña y su confirmación no coinciden, se notifica el error.



The screenshot shows the same web application interface as in 5.1. The modal dialog titled 'Cambiar Contraseña.' is centered on the screen. It displays the user's name 'Angel' and three input fields: 'Contraseña actual', 'Nueva contraseña', and 'Confirmar contraseña'. The 'Contraseña actual' and 'Nueva contraseña' fields contain six dots, indicating masked text. The 'Confirmar contraseña' field also contains six dots. Below the input fields, a red error message reads 'Confirmar contraseña y Nueva contraseña no coinciden.' and a 'Cambiar Contraseña' button is visible at the bottom of the modal.

## 6. Configurar tipo de cuenta del usuario

1. En la barra lateral derecha, en el recuadro 'Tipo de cuenta', dar clic en el vínculo 'Configurar'.



Nombre  
Angel reyes

Fecha de nacimiento  
01/01/0001

Estado  
Distrito Federal

Escuela  
ESCOM

Nivel  
Otro

Grado  
8

Duración del grado  
Semestre

Correo electrónico  
celtics\_reyes@hotmail.com

Actualizar imagen de perfil  
Puedes subir una imagen en formato jpg como fotografía de perfil  
Seleccionar Archivo No se eligió archivo  
Subir imagen

Tipo de cuenta **Configurar**  
Puedes configurar el tipo de tu cuenta, cambiar entre alumno o asesor

Eliminar perfil Eliminar

2. El sistema muestra la ventana que permite configurar el tipo de cuenta, una vez que se elige entre asesor o alumno, se da clic en el botón 'Guardar cambios'.



[ ] Angel Salir

karel Usuarios Problemas Listas Concursos

Editar perfil Volver

Editar tipo de cuenta

Tipo de cuenta  
Asesor

Guardar cambios

3. El sistema redirecciona a la vista de editar perfil del usuario y notifica que los cambios han sido guardados correctamente.



## 7. Eliminar perfil del usuario

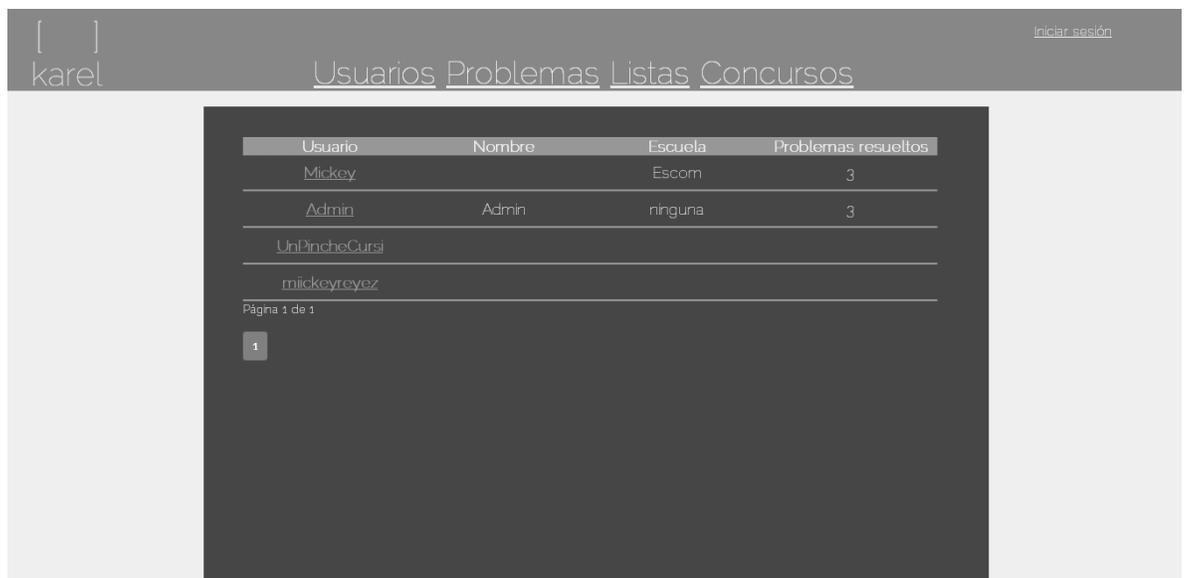
1. En la barra lateral derecha, en el recuadro 'Eliminar perfil', dar clic en el vínculo 'Eliminar'.



2. El sistema muestra la vista de eliminar perfil, dar clic al botón 'Eliminar'.

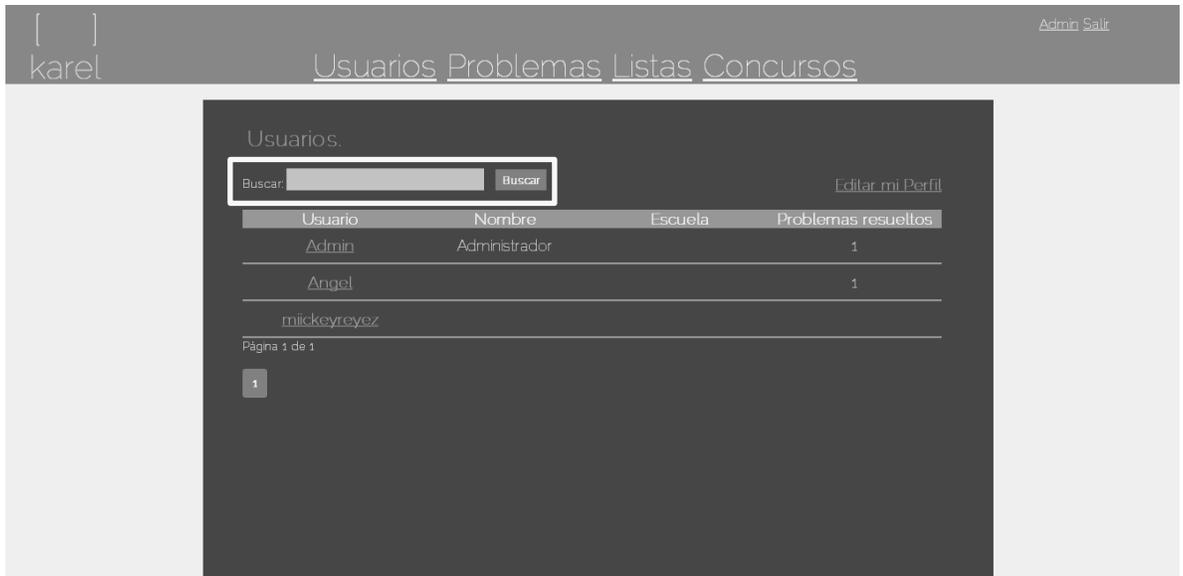


3. El sistema redirecciona a la lista de usuarios, se finaliza la sesión y la cuenta es borrada del sistema.

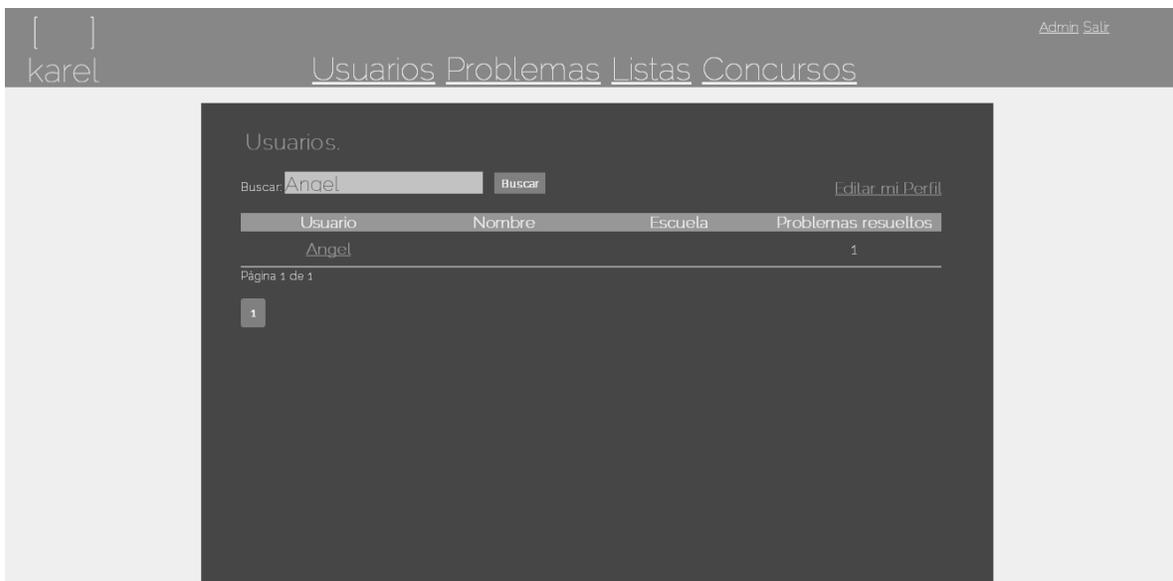


## 8. Buscar un usuario

1. En la página de Usuarios, escribir en el cuadro de texto el problema a buscar y dar clic al botón 'Buscar'.



2. El sistema mostrará los resultados similares al criterio de búsqueda.



## Problemas

### 1. Creación de un problema

1. En la página de problemas, dar clic en la opción 'Generar Problema'.



2. En el formulario de registro, se asocia un nombre al problema, la privacidad del problema (si el problema es privado, solo podrá ser visto en 'Mis Problemas'), el tipo de evaluación para el problema y se anexa una descripción, posteriormente se da clic en el botón 'Crear'.



3. Una vez creado el problema, se solicitarán los casos de prueba es decir, los mundos así como las soluciones asociadas a un problema.

[ ] Admin Salir  
karel Usuarios Problemas Listas Concursos

Agregar casos. [Volver](#)

Problema: Problema de prueba

Agrega un caso de prueba

Mundo:  
Seleccionar Archivo No se eligió archivo

Solución:  
Seleccionar Archivo No se eligió archivo

Subir caso de prueba

Mundo Solución

4. Se procede a asociar los casos de prueba al problema y una vez que se da clic en el botón 'Subir casos de prueba', el sistema notifica que se han anexado correctamente los casos de prueba (existe el botón eliminar, que remueve los casos de prueba).

[ ] Admin Salir  
karel Usuarios Problemas Listas Concursos

Agregar casos. [Volver](#)

Problema: Problema de prueba

Agrega un caso de prueba

Mundo:  
Seleccionar Archivo No se eligió archivo

Solución:  
Seleccionar Archivo No se eligió archivo

Subir caso de prueba

Mundo	Solución	
4.mdo	4.kec	<a href="#">Eliminar</a>

## 1.1 Error en la creación del problema – Generar problema vacío

1. Cuando el usuario no ha escrito un nombre al problema, el sistema notifica el error.

The screenshot shows the 'Nuevo Problema' form in a web application. The form has a dark background. At the top right, there is a 'Volver' link. The 'Nombre' field is empty, and a red error message below it reads 'El campo Nombre es obligatorio.' Below this, there are two checkboxes: 'Problema público' (checked) and 'Evaluación parcial' (checked). The 'Descripción' field is a rich text editor with a toolbar. The top navigation bar includes 'karel' and links for 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. The top right corner has 'Admin Salir'.

## 2. Detalles de un problema

1. En la página de problemas, dar clic en el nombre del problema a visualizar.

The screenshot shows the 'Problemas' page in a web application. At the top right, there is an 'Admin Salir' link. The navigation bar includes 'karel' and links for 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. The main content area has a dark background and contains the following table:

Problemas. [Generar Problema](#) [Mis Problemas](#)

Id	Nombre	Correctos	Intentos
1000	<a href="#">Problema 1</a>	5	8
1001	<a href="#">Problema 2</a>	2	2
1002	<a href="#">Problema 3</a>	1	1
1003	<a href="#">Problema 4</a>	0	0
1004	<a href="#">Problema 5</a>	0	0
1006	<a href="#">Problema 6</a>	0	0
1007	<a href="#">Problema de prueba</a>	0	0

Página 1 de 1

1

2. Posteriormente se abrirá una ventana donde se muestran los detalles del problema seleccionado.



### 3. Evaluación de un problema

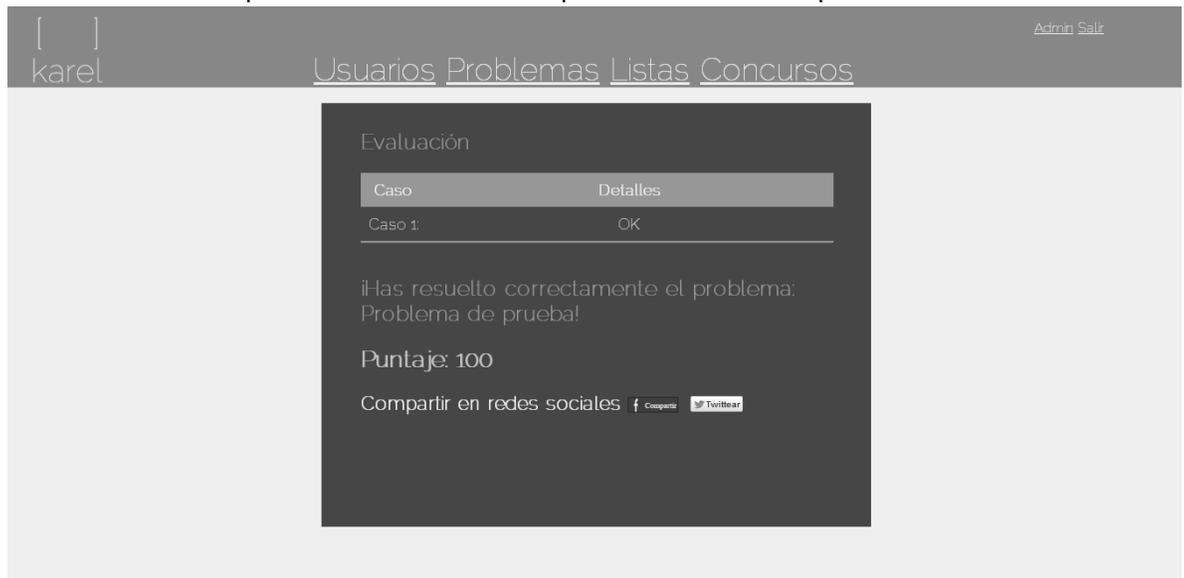
1. En la vista de detalles del problema, se puede enviar la solución a un problema en lenguaje Java y Pascal (el lenguaje por defecto seleccionado es Java).



- Una vez seleccionado el archivo que satisface las condiciones del caso de evaluación, se da clic al botón de '¡Evaluar!'.



- El sistema realiza la evaluación correspondiente al problema, una vez resuelto correctamente, se puede llevar a cabo una publicación en cualquier red social.



4. El sistema actualiza las veces que ha sido resuelto correctamente el problema.

Problemas. [Generar Problema](#) [Mis Problemas](#)

Id	Nombre	Correctos	Intentos
1000	Problema 1	5	8
1001	Problema 2	2	2
1002	Problema 3	1	1
1003	Problema 4	0	0
1004	Problema 5	0	0
1006	Problema 6	0	0
1007	Problema de prueba	2	3

Página 1 de 1

1

### 3.3 Error en la evaluación – No se ha resuelto el problema correctamente

1. Cuando la evaluación del problema ha sido incorrecta, el sistema notifica los detalles por la cual la evaluación ha sido considerada incorrecta y no se permite compartir la evaluación en alguna red social.

Evaluación

Caso	Detalles
Caso 1:	El número de zumbadores en una posición es incorrecta.

Puntaje: 0

2. El sistema actualiza las veces que ha sido intentado el problema.

Problemas. [Generar Problema](#) [Mis Problemas](#)

Id	Nombre	Correctos	Intentos
1000	Problema 1	5	8
1001	Problema 2	2	2
1002	Problema 3	1	1
1003	Problema 4	0	0
1004	Problema 5	0	0
1006	Problema 6	0	0
1007	Problema de prueba	2	3

Página 1 de 1

1

### 3.4 Error en la evaluación – No se ha iniciado sesión

1. Cuando se desea resolver un problema sin iniciar sesión, el sistema no deja llevar a cabo la evaluación.



#### 4. Visualizar mis problemas

1. En la página de problemas, dar clic en el vínculo 'Mis Problemas'.



The screenshot shows a web application interface. At the top left, there is a user profile icon and the name 'karel'. At the top right, there are links for 'Admin' and 'Salir'. Below the navigation bar, there are links for 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. The main content area is titled 'Problemas.' and contains a table with columns 'Id', 'Nombre', 'Correctos', and 'Intentos'. The table lists seven problems. A link 'Generar Problemas' is visible, and the link 'Mis Problemas' is highlighted with a red box. Below the table, there is a pagination indicator 'Página 1 de 1' and a button with the number '1'.

Id	Nombre	Correctos	Intentos
1000	Problema 1	5	8
1001	Problema 2	2	2
1002	Problema 3	1	1
1003	Problema 4	0	0
1004	Problema 5	0	0
1006	Problema 6	0	0
1007	Problema de prueba	2	3

2. El sistema redirecciona a una vista donde se muestran todos los problemas que han sido generados por el usuario.



The screenshot shows the same web application interface as the previous one. The main content area is titled 'Problemas.' and contains a table with columns 'IdProblema', 'Nombre', and 'Editar | Eliminar'. The table lists seven problems. The link 'Generar Problemas' is visible, and the link 'Problemas' is highlighted with a red box.

IdProblema	Nombre	Editar   Eliminar
1000	Problema 1	Editar   Eliminar
1001	Problema 2	Editar   Eliminar
1002	Problema 3	Editar   Eliminar
1003	Problema 4	Editar   Eliminar
1004	Problema 5	Editar   Eliminar
1005	Problema 6	Editar   Eliminar
1006	Problema 6	Editar   Eliminar
1007	Problema de prueba	Editar   Eliminar

## 5. Editar un problema

1. En la página de problemas, dar clic en el vínculo 'Mis Problemas'.

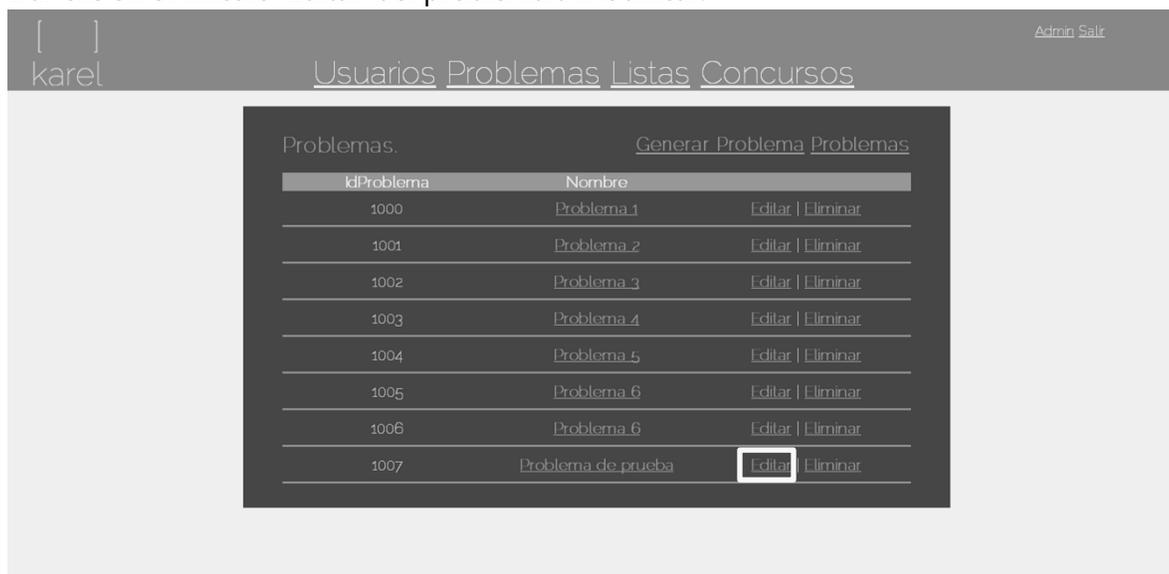


The screenshot shows a web interface with a header containing a user profile icon and the name 'karel', and navigation links for 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. The 'Problemas' link is active. Below the header, there are links for 'Problemas.', 'Generar Problemas', and 'Mis Problemas', with 'Mis Problemas' highlighted by a red box. A table lists several problems with columns for 'Id', 'Nombre', 'Correctos', and 'Intentos'. The table data is as follows:

Id	Nombre	Correctos	Intentos
1000	Problema 1	5	8
1001	Problema 2	2	2
1002	Problema 3	1	1
1003	Problema 4	0	0
1004	Problema 5	0	0
1006	Problema 6	0	0
1007	Problema de prueba	2	3

At the bottom of the table, it indicates 'Página 1 de 1' and a pagination control showing '1'.

2. Dar clic en el vínculo 'Editar' del problema a modificar.



The screenshot shows the same web interface as above. In this view, the 'Problemas' link is active, and the table lists the problems with 'Editar' and 'Eliminar' links for each. The 'Editar' link for the 'Problema de prueba' (ID 1007) is highlighted by a red box. The table data is as follows:

IdProblema	Nombre	Acciones
1000	Problema 1	Editar   Eliminar
1001	Problema 2	Editar   Eliminar
1002	Problema 3	Editar   Eliminar
1003	Problema 4	Editar   Eliminar
1004	Problema 5	Editar   Eliminar
1005	Problema 6	Editar   Eliminar
1006	Problema 6	Editar   Eliminar
1007	Problema de prueba	Editar   Eliminar

3. El sistema carga la información previa del problema, para poder ser modificada (editar casos funciona de la misma manera en que se sube un caso al crear un problema).



Editar. [Editar Casos](#) [Volver](#)

**Nombre**  
Problema de prueba

**Problema público**   
Si tu problema es privado, sólo tú podrás visualizarlo para utilizarlo en concursos.

**Evaluación parcial**   
La evaluación parcial se da como resultado proporcional en base a los casos proporcionados de otro modo se evalúa de como 'correcto' o 'incorrecto'.

**Descripción**  
Este problema es una prueba, es decir un problema ejemplo.

Guardar

4. Una vez que la información ha sido modificada, se da clic al botón 'Guardar' para almacenar la información que ha sido editada.



Editar. [Editar Casos](#) [Volver](#)

**Nombre**  
Problema de prueba

**Problema público**   
Si tu problema es privado, sólo tú podrás visualizarlo para utilizarlo en concursos.

**Evaluación parcial**   
La evaluación parcial se da como resultado proporcional en base a los casos proporcionados de otro modo se evalúa de como 'correcto' o 'incorrecto'.

**Descripción**  
Este problema es una prueba, es decir un problema ejemplo que ha sido modificado.

Guardar

5. El sistema redirecciona a las lista de problemas que existen actualmente.

[ ]  
karel

Admin Salir

Usuarios Problemas Listas Concursos

Problemas. [Generar Problema](#) [Mis Problemas](#)

Id	Nombre	Correctos	Intentos
1000	Problema 1	5	8
1001	Problema 2	2	2
1002	Problema 3	1	1
1003	Problema 4	0	0
1004	Problema 5	0	0
1006	Problema 6	0	0
1007	Mi Problema de prueba	2	3

Página 1 de 1

1

## 6. Eliminar un problema

1. En la página de problemas, dar clic en el vínculo 'Mis Problemas'.

[ ]  
karel

Admin Salir

Usuarios Problemas Listas Concursos

Problemas. [Generar Problema](#) [Mis Problemas](#)

Id	Nombre	Correctos	Intentos
1000	Problema 1	5	8
1001	Problema 2	2	2
1002	Problema 3	1	1
1003	Problema 4	0	0
1004	Problema 5	0	0
1006	Problema 6	0	0
1007	Problema de prueba	2	3

Página 1 de 1

1

2. Dar clic en el vínculo 'Eliminar' del problema a eliminar.

The screenshot shows a web application interface with a header containing the logo 'karel' and navigation links: 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. The user is logged in as 'Admin' and can 'Salir'. Below the header, there is a section titled 'Problemas.' with a link to 'Generar Problema' and another 'Problemas' link. A table lists several problems with columns for 'IdProblema', 'Nombre', and 'Editar | Eliminar'. The 'Eliminar' link for the last row, 'Problema de prueba', is highlighted with a red box.

IdProblema	Nombre	Editar   Eliminar
1000	Problema 1	Editar   Eliminar
1001	Problema 2	Editar   Eliminar
1002	Problema 3	Editar   Eliminar
1003	Problema 4	Editar   Eliminar
1004	Problema 5	Editar   Eliminar
1005	Problema 6	Editar   Eliminar
1006	Problema 6	Editar   Eliminar
1007	Problema de prueba	Editar   <b>Eliminar</b>

3. El sistema carga el nombre del problema con el fin de notificar el problema que será borrado, dar clic al botón 'Eliminar' para eliminar el problema permanentemente del sistema.

The screenshot shows a confirmation dialog box titled 'Eliminar.' with a 'Volver' link. The text asks: '¿Estás seguro que deseas eliminar este problema?' and displays the name of the problem to be deleted: 'Mi Problema de prueba'. At the bottom, there is a button labeled 'Eliminar' which is highlighted with a red box.

4. El sistema redirecciona al inicio de los problemas, donde se puede observar que el problema ha sido eliminado.

Problemas. [Generar Problema](#) [Mis Problemas](#)

Id	Nombre	Correctos	Intentos
1000	Problema 1	5	8
1001	Problema 2	2	2
1002	Problema 3	1	1
1003	Problema 4	0	0
1004	Problema 5	0	0
1006	Problema 6	0	0

Página 1 de 1

1

### 7. Ordenar un problema – Clasificador de problemas

5. En la página de problemas, se muestran las opciones Correctos e Intentos.

Problemas

Buscar

Ver evaluaciones

Id	Nombre	Correctos	Intentos	Porcentaje
1002	1	0	0	0.00
1004	Problema Uno	555	555	100.00

Página 1 de 1

1

6. Dar clic en la opción a ordenar, para que el sistema ordene los problemas según el criterio.

### Problemas

Buscar

[Ver evaluaciones](#)

Id	Nombre	Correctos	Intentos	Porcentaje
1004	Problema Uno	555	555	100.00
1002	1	0	0	0.00

Página 1 de 1

1

## 8. Buscar un problema

1. En la página de Problemas, escribir en el cuadro de texto el problema a buscar y dar clic al botón 'Buscar'.



The screenshot shows the 'Problemas' page of the 'karel' application. The header includes the 'karel' logo and navigation links for 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. A search bar is present with the text 'Buscar' and a 'Buscar' button. Below the search bar, there is a table with the following data:

Id	Nombre	Correctos	Intentos	Porcentaje
1002	1	0	0	0.00
1004	Problema Uno	555	555	100.00

Below the table, it indicates 'Página 1 de 1' and a page number '1'.

2. El sistema mostrará los resultados similares al criterio de búsqueda.



The screenshot shows the 'Problemas' page after a search for 'Uno'. The search bar now contains the text 'Uno' and the 'Buscar' button. The table below shows the search results:

Id	Nombre	Correctos	Intentos	Porcentaje
1004	Problema Uno	555	555	100.00

Below the table, it indicates 'Página 1 de 1' and a page number '1'.

## Listas

### 1. Creación de una lista

1. En la página de listas, dar clic en la opción 'Generar Lista'.



2. Posteriormente se asigna un nombre a la lista y da clic al botón 'Crear lista'.

The screenshot shows the 'Crear lista' form in the Karel system. The header includes the Karel logo and navigation links for 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. The form contains a 'Nombre' field with the value 'Lista de ejemplo' and a 'Crear lista' button.

3. El sistema redirecciona a una vista donde se agregan los problemas que serán parte de la lista a través de un id, así como los filtros a mostrar por entidad federativa y asesor (los problemas y los filtros pueden ser removidos en la vista), la lista es guardada directamente.

The screenshot shows the 'Crear lista' view in the Karel system. The header includes the Karel logo and navigation links for 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. The view contains a form for adding problems, a table of existing problems, and a section for visibility options.

**Crear lista**

El problema: 1007 Mi Problema de prueba

Agregar problema

Id	Nombre	
1000	Problema 1	Eliminar
1002	Problema 3	Eliminar
1003	Problema 4	Eliminar

Opciones de visibilidad.

Estado: Aguascalientes Agregar filtro

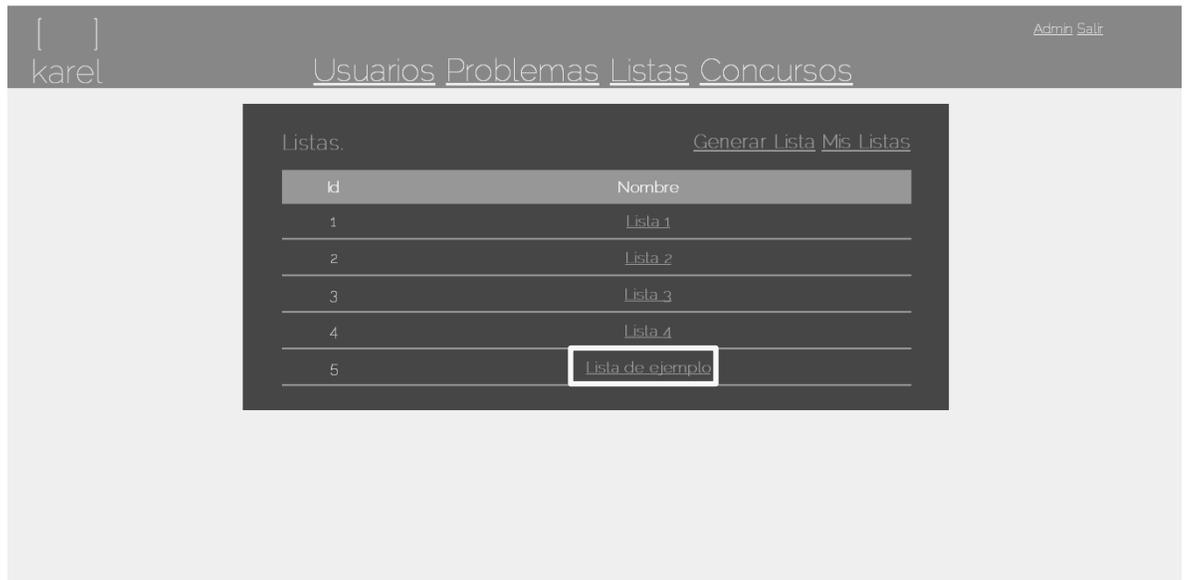
Asesor: Agregar filtro

Filtrado por: Valor

Estado	Detrío Federal	Eliminar

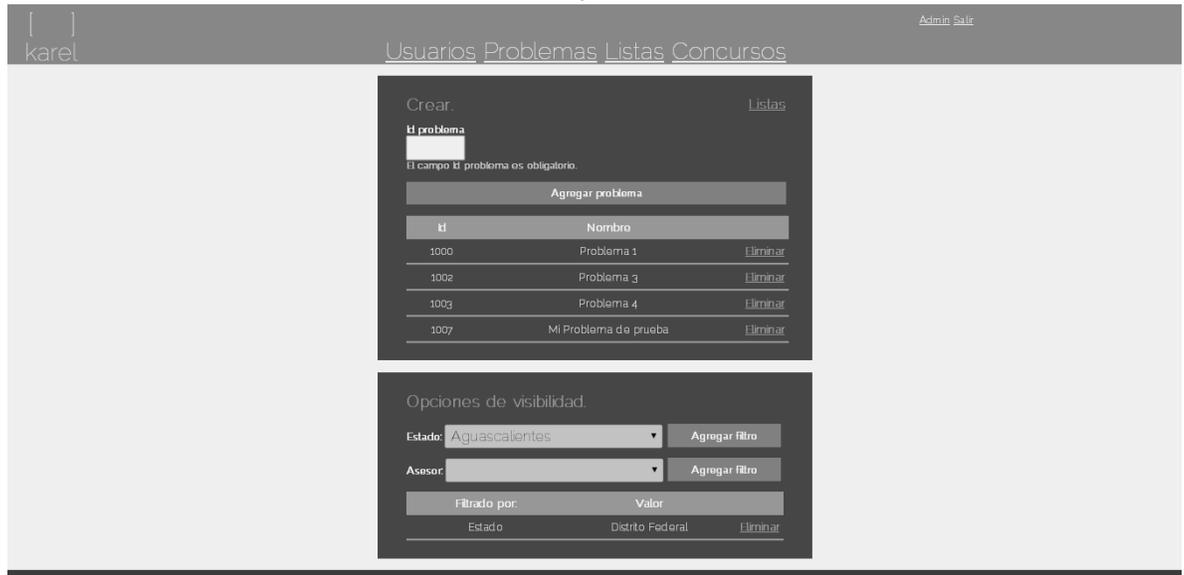
© 2014 - TT 2013-A003 Sistema Web para la Evaluación Automática de Programas de Karel el Robot

4. Al acceder a las listas del sistema, se puede observar que ha sido guardada la nueva lista.



### 1.1 Error al agregar problema la lista – No se ha asignado un id de problema a la lista

1. Cuando el usuario no ha escrito un id de un problema, el sistema notifica el error.



## 2. Detalles de una lista

1. Dar clic en el nombre de la lista de la cual se desea obtener los detalles.



[ ] Admin Salir

karel Usuarios Problemas Listas Concursos

Listas. [Generar Lista](#) [Mis Listas](#)

Id	Nombre
1	Lista 1
2	Lista 2
3	Lista 3
4	Lista 4
5	Lista de ejemplo

2. El sistema mostrará los problemas que han sido asociados a la lista.



[ ] Admin Salir

karel Usuarios Problemas Listas Concursos

Problemas. [Volver](#)

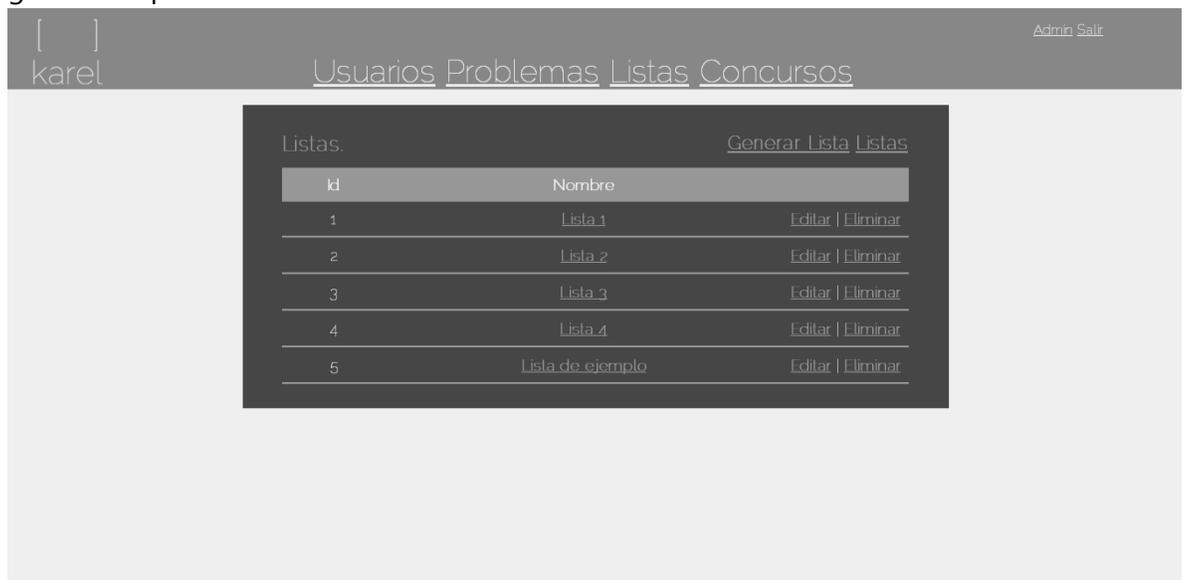
Id	Nombre
1000	Problema 1
1002	Problema 3
1003	Problema 4
1007	Mi Problema de prueba

### 3. Visualizar mis listas

1. En la página de listas, dar clic en el vínculo 'Mis Listas'.

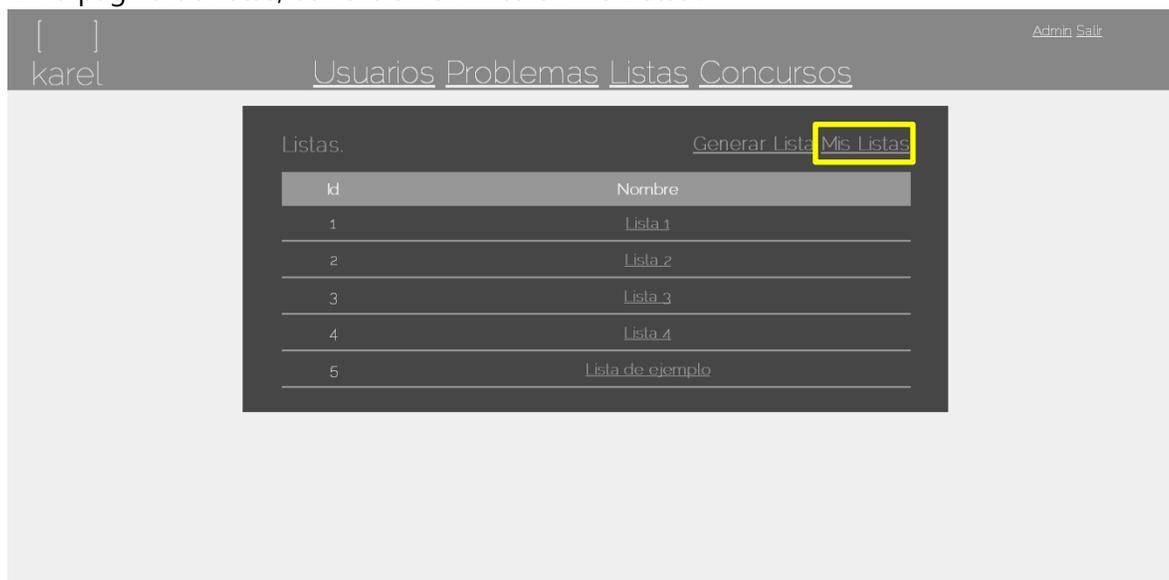


2. El sistema redirecciona a una vista donde se muestran todas las listas que han sido generadas por el usuario.

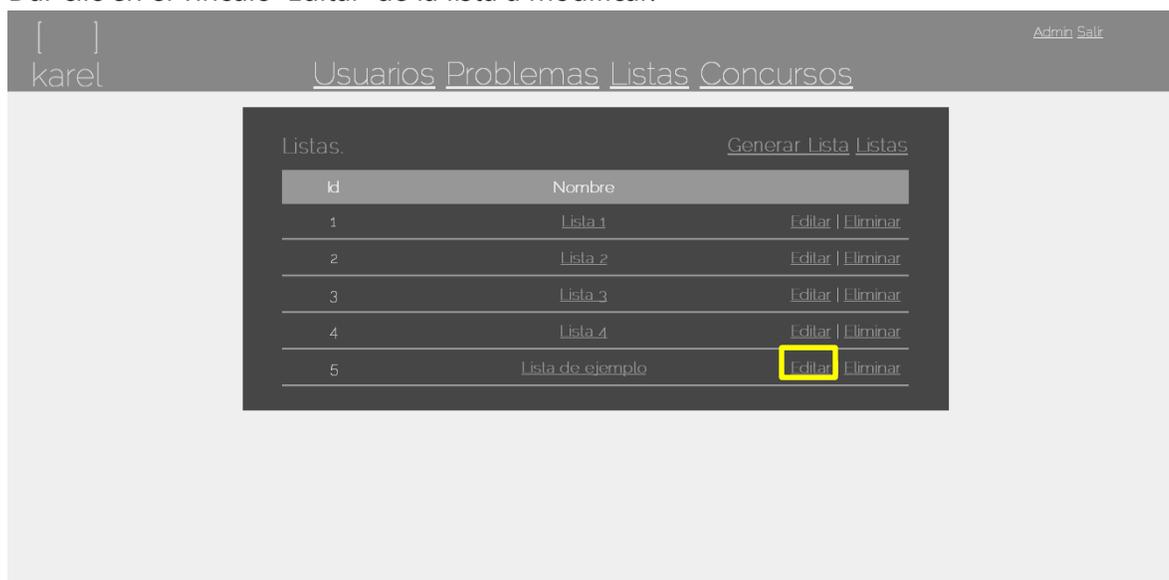


#### 4. Editar una lista

1. En la página de listas, dar clic en el vínculo 'Mis Listas'.



2. Dar clic en el vínculo 'Editar' de la lista a modificar.



3. El sistema carga el nombre previo de la lista, para poder ser modificado, al dar clic en el botón 'Guardar', se guarda la lista con el nombre nuevo.

[ ]  
karel

Admin Salir

Usuarios Problemas Listas Concursos

Editar. [Editar contenido](#) [Volver](#)

Nombre

Lista de ejemplo

Guardar

4. El contenido de la lista se modifica mediante el vínculo 'Editar contenido'.

[ ]  
karel

Admin Salir

Usuarios Problemas Listas Concursos

Editar. [Editar contenido](#) [Volver](#)

Nombre

Lista de ejemplo

Guardar

5. El contenido de la lista se modifica de la misma manera en que se creó, los problemas que contiene la lista, son previamente mostrados así como los filtros existentes.

The screenshot shows the 'Listas' management page. At the top, there is a navigation bar with 'karel' on the left and 'Admin Salir' on the right. The main navigation menu includes 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. The 'Listas' section is active.

Under the 'Crear.' heading, there is a form with a 'Id problema' input field and an 'Agregar problema' button. Below this is a table of existing problems:

Id	Nombre	
1000	Problema 1	<a href="#">Eliminar</a>
1002	Problema 3	<a href="#">Eliminar</a>
1003	Problema 4	<a href="#">Eliminar</a>
1007	Mi Problema de prueba	<a href="#">Eliminar</a>

Below the table, there are 'Opciones de visibilidad.' with two dropdown menus: 'Estado' (set to 'Aguascalientes') and 'Asesor'. Each dropdown has an 'Agregar filtro' button. At the bottom, there is a 'Filtrado por:' label and a 'Valor' input field.

At the very bottom of the page, there is a footer: '© 2014 - TT 2013-2003 Sistema Web para la Evaluación Automática de Programas de Karel el Robot'.

### 5. Eliminar una lista

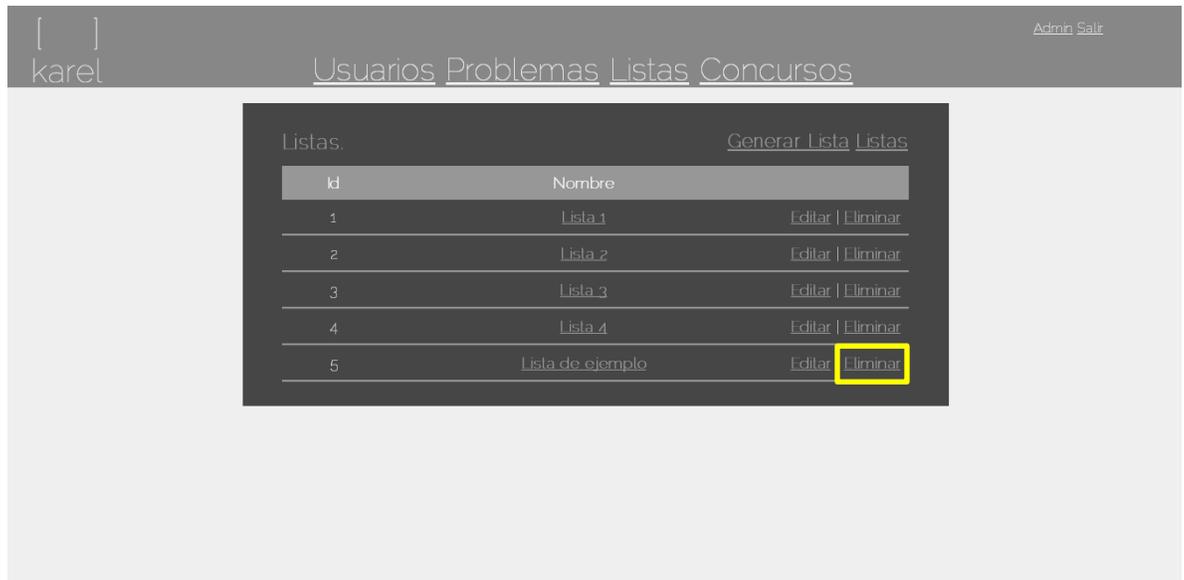
1. En la página de listas, dar clic en el vínculo 'Mis Listas'.

The screenshot shows the 'Listas' page. At the top, there is a navigation bar with 'karel' on the left and 'Admin Salir' on the right. The main navigation menu includes 'Usuarios', 'Problemas', 'Listas', and 'Concursos'. The 'Listas' section is active.

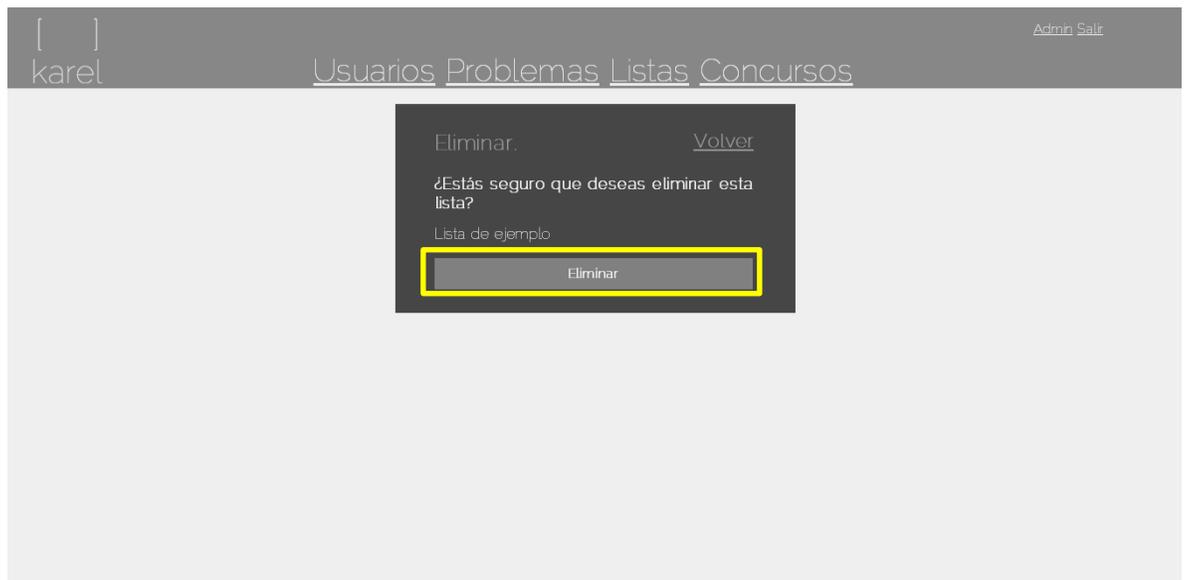
Under the 'Listas.' heading, there is a 'Generar Lista' button and a 'Mis Listas' link, which is highlighted with a yellow box. Below this is a table of existing lists:

Id	Nombre
1	Lista_1
2	Lista_2
3	Lista_3
4	Lista_4
5	Lista de ejemplo

2. Dar clic en el vínculo 'Eliminar' de la lista a eliminar.



3. El sistema carga el nombre de la lista con el fin de notificar la lista que será eliminada, dar clic al botón 'Eliminar' para eliminar la lista permanentemente del sistema.



4. El sistema redirecciona al inicio de las listas, donde se puede observar que la

[ ] Admin Salir  
karel Usuarios Problemas Listas Concursos

Listas. Generar Lista Mis Listas

Id	Nombre
1	Lista 1
2	Lista 2
3	Lista 3
4	Lista 4

lista ha sido eliminada.

## Concursos

### 1. Creación de un concurso

1. En la página de concursos, dar clic en la opción 'Crear Nuevo Concurso'.

[ ] Admin Salir  
karel Usuarios Problemas Listas Concursos

Concursos. **Crear Nuevo Concurso**

Id	Nombre	Fecha Inicio	Fecha Fin	
1	Concurso 1	02/05/2014 02:500 a. m.	02/05/2014 03:500 a. m.	Entrar   Editar   Eliminar
2	2	15/05/2014 01:000 a. m.	15/05/2014 02:000 a. m.	Entrar   Editar   Eliminar
3	Concurso de Prueba	07/05/2014 06:100 p. m.	07/05/2014 07:400 p. m.	Entrar   Editar   Eliminar

- Posteriormente se llena el formulario de creación del concurso, se asigna un nombre al concurso, una fecha de inicio y hora exacta, la duración del mismo, el tiempo para congelar el *ScoreBoard*, el tiempo de penalización, el criterio de desempate y el tipo de evaluación, una vez listo, se da clic al botón 'Crear'.

Crea concurso. [Volver](#)

Nombre del Concurso  
Ejemplo de concurso

Fecha de inicio del concurso  
19/05/2014

Seleccione la hora: 21 : 00

Duración del concurso en minutos  
30

Tiempo para congelar el ScoreBoard  
10  
Cantidad de minutos previos a la finalización del concurso que la tabla dejará de mostrar resultados.

Tiempo de penalización  
1  
Indique la cantidad de minutos que se sumaran al envío en caso de que sea incorrecto.

El tiempo es criterio de desempate

Evaluación Parcial

**Crear**

© 2014 - TT 2013-A003: Sistema Web para la Evaluación Automática de Programas de Karel el Robot

- Una vez que el concurso ha sido creado, el sistema muestra la ventana de selección de problemas para el concurso, donde se agrega el id del problema en el cuadro de texto y mediante agregar se agrega el problema a la lista, se pueden remover problemas directamente desde la lista, además de que se pueden agregar filtros al concurso.

[ ] karel [Admin](#) [Salir](#)

[Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

Crear. [Finalizar](#)

id problema

**Agregar problema**

id	Nombre
1002	1

[Eliminar](#)

Opciones de visibilidad.

Estado:  [Agregar filtro](#)

Asesor:  [Agregar filtro](#)

Filtrado por:  Valor

4. El sistema redirecciona a la vista que alberga todos los concursos existentes en el sitio, donde se puede visualizar el concurso que ha sido creado.

[ ] Admin Salir  
karel Usuarios Problemas Listas Concursos

Concursos. [Crear Nuevo Concurso](#)

Id	Nombre	Fecha Inicio	Fecha Fin	
1	Concurso 1	02/05/2014 02:500 a. m.	02/05/2014 03:500 a. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
2	2	15/05/2014 01:000 a. m.	15/05/2014 02:000 a. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
3	Concurso de Prueba	07/05/2014 06:1000 p. m.	07/05/2014 07:4000 p. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
4	Ejemplo de concurso	19/05/2014 12:0000 p. m.	19/05/2014 01:3000 p. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>

### 1.1 Error al crear concurso – No se han enviado datos al servidor

1. Cuando el usuario intenta crear un concurso vacío, el sistema notifica el error.

karel Usuarios Problemas Listas Concursos

Crea concurso. [Volver](#)

**Nombre del Concurso**  
Campo obligatorio

**Fecha de inicio del concurso**  
Campo obligatorio

Seleccione la hora: 01 : 00

**Duración del concurso en minutos**  
Campo obligatorio

**Lista**

**Tiempo para congelar el Scoreboard**

Cantidad de minutos previos a la finalización del concurso que la tabla dejará de mostrar resultados.

**Tiempo de penalización**  
Indique la cantidad de minutos que se sumaran al envío en caso de que sea incorrecto.  
Campo obligatorio

El tiempo es criterio de desempate

Evaluación Parcial

Crear

## 2. Editar un concurso

1. En la página de concursos, dar clic al vínculo 'Editar' del concurso que se desea modificar.



Concursos. [Crear Nuevo Concurso](#)

Id	Nombre	Fecha Inicio	Fecha Fin	
1	Concurso 1	02/05/2014 02:500 a. m.	02/05/2014 03:500 a. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
2	2	15/05/2014 01:0000 a. m.	15/05/2014 02:0000 a. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
3	Concurso de Prueba	07/05/2014 06:1000 p. m.	07/05/2014 07:4000 p. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
4	Ejemplo de concurso	19/05/2014 12:0000 p. m.	19/05/2014 01:3000 p. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>

2. El sistema carga previamente los datos del concurso, para poder ser modificado, al dar clic en el botón 'Guardar', se guardan los nuevos datos del concurso.



Editar. [Editar Problemas](#) [Volver](#)

Nombre del Concurso  
Ejemplo de concurso

Fecha de inicio del concurso  
19/05/2014

Seleccione la hora: 01 00

Duración del concurso en minutos  
30

Tiempo para congelar el ScoreBoard  
10

Tiempo de penalización  
1

El tiempo es criterio de desempate

Evaluación Parcial

IdCreador  
Admin

[Guardar](#)

© 2014 - TT 2013-A003- Sistema Web para la Evaluación Automática de Programas de Karel el Robot

3. Para editar los problemas dar clic al vínculo 'Editar Problemas'.

The screenshot shows a form titled 'Editar Problemas' with a yellow box around the title. The form contains the following fields and options:

- Nombre del Concurso: Ejemplo de concurso
- Fecha de inicio del concurso: 19/05/2014
- Selección de hora: 01 : 00
- Duración del concurso en minutos: 90
- Tiempo para congelar el ScoreBoard: 10
- Tiempo de penalización: 1
- El tiempo es criterio de desempate:
- Evaluación Parcial:
- IdCreador: Admin
- Botón: Guardar

At the bottom of the page, there is a footer: © 2014 - TT 2013-A003: Sistema Web para la Evaluación Automática de Programas de Karel el Robot

4. El sistema muestra los problemas asociados al concurso y mediante esa vista se pueden agregar o remover problemas y filtros, una vez terminado dar clic al vínculo 'Finalizar'.

The screenshot shows the 'Finalizar' view with a yellow box around the 'Finalizar' button. The page includes a navigation bar with 'karel' and 'Admin Salir'.

Navigation links: [Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

Form sections:

- Crear:** Includes a text input for 'Id problema', an 'Agregar problema' button, and a table with one row: 

Id	Nombre	Eliminar
1002	1	<a href="#">Eliminar</a>
- Opciones de visibilidad:** Includes dropdowns for 'Estado' (Aguascalientes) and 'Asesor', each with an 'Agregar filtro' button, and a 'Filtrado por: Valor' input field.

5. El sistema redirecciona a la lista de concursos.

Concursos. [Admin Salir](#)

[Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

[Crear Nuevo Concurso](#)

Id	Nombre	Fecha Inicio	Fecha Fin	
1	Concurso 1	02/05/2014 021500 a. m.	02/05/2014 031500 a. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
2	2	15/05/2014 010000 a. m.	15/05/2014 020000 a. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
3	Concurso de Prueba	07/05/2014 061000 p. m.	07/05/2014 074000 p. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
4	Ejemplo	19/05/2014 120000 p. m.	19/05/2014 013000 p. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>

### 3. Eliminar un concurso

1. En la página de concursos, dar clic al vínculo 'Eliminar' del concurso que se desea eliminar.

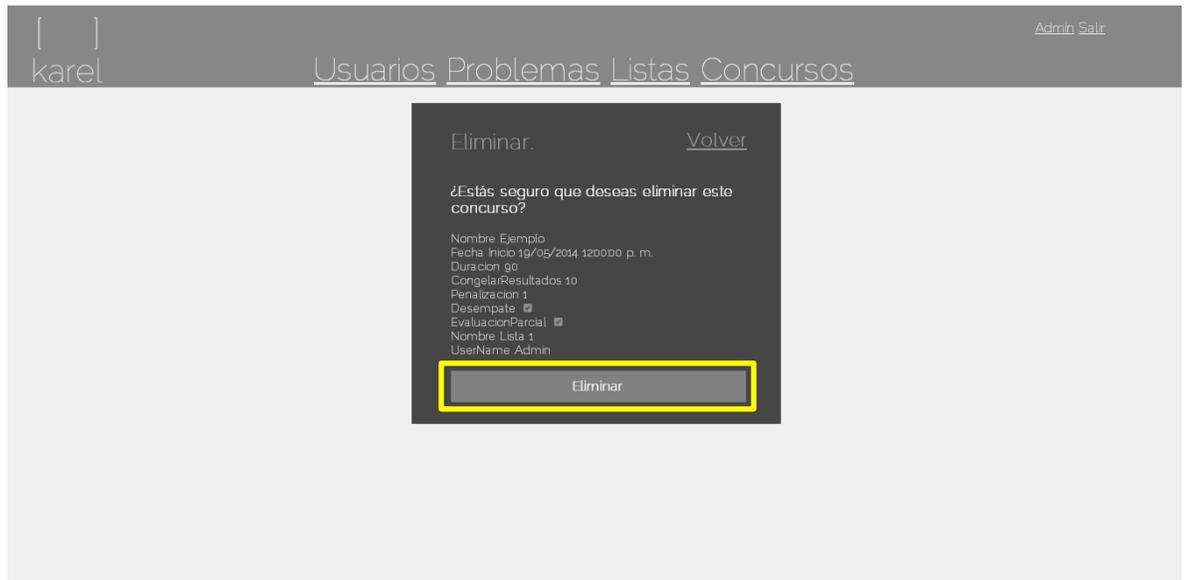
Concursos. [Admin Salir](#)

[Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

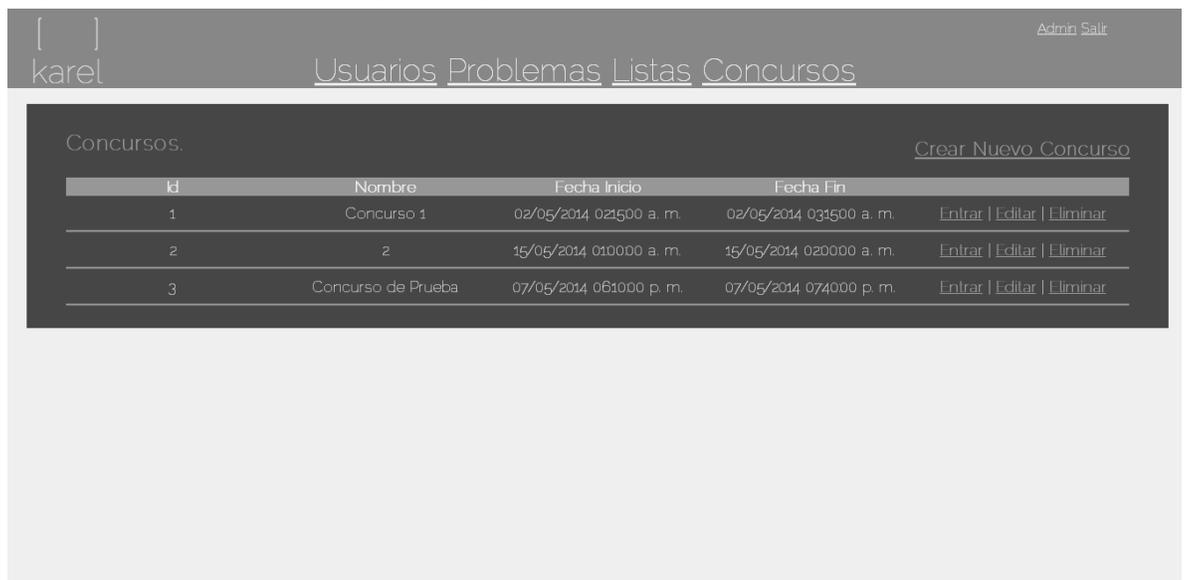
[Crear Nuevo Concurso](#)

Id	Nombre	Fecha Inicio	Fecha Fin	
1	Concurso 1	02/05/2014 021500 a. m.	02/05/2014 031500 a. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
2	2	15/05/2014 010000 a. m.	15/05/2014 020000 a. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
3	Concurso de Prueba	07/05/2014 061000 p. m.	07/05/2014 074000 p. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
4	Ejemplo	19/05/2014 120000 p. m.	19/05/2014 013000 p. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>

2. El sistema carga previamente los datos del concurso a ser eliminado, con el fin de notificar al usuario, al dar clic en el botón 'Eliminar', el concurso es borrado.



3. El sistema redirecciona a la lista de concursos.



#### 4. Participar a un concurso

1. En la página de concursos, dar clic al vínculo 'Entrar' del concurso que se desea participar.



Concursos. [Admin](#) [Salir](#)

[Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

[Crear Nuevo Concurso](#)

Id	Nombre	Fecha Inicio	Fecha Fin	
1	Concurso 1	02/05/2014 02:15:00 a. m.	02/05/2014 03:15:00 a. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
2	2	15/05/2014 01:00:00 a. m.	15/05/2014 02:00:00 a. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
3	Concurso de Prueba	07/05/2014 06:10:00 p. m.	07/05/2014 07:40:00 p. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>

2. El sistema mostrará los problemas asociados al correspondiente concurso, mostrando además el tiempo restante del concurso, para resolver un problema es necesario dar clic en el nombre del problema.



Concurso de Prueba. [Volver](#)

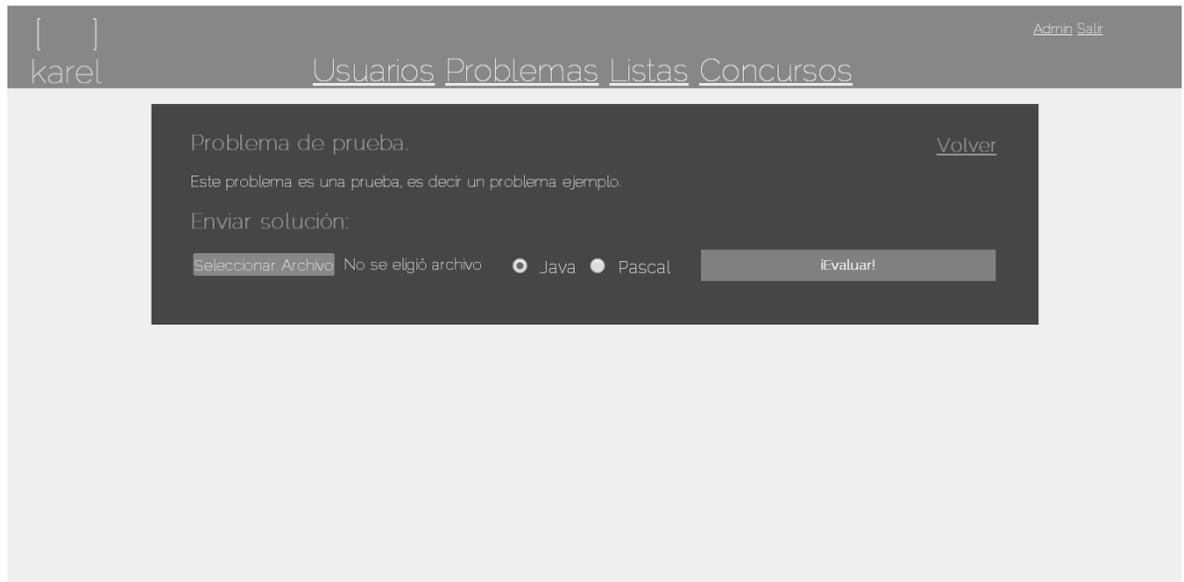
Quedan: 0 días, 1 hora, 37 minutos, 9 segundos.

Problemas. [Scoreboard](#)

Id	Nombre
1000	<a href="#">Problema 1</a>
1001	<a href="#">Problema 2</a>
1002	<a href="#">Problema 3</a>
1003	<a href="#">Problema 4</a>
1005	<a href="#">Problema 6</a>
1007	<a href="#">Problema de prueba</a>

Compartir [Twitter](#) [Compartir](#)

3. Posteriormente se abrirá una ventana donde se muestran los detalles del problema seleccionado.



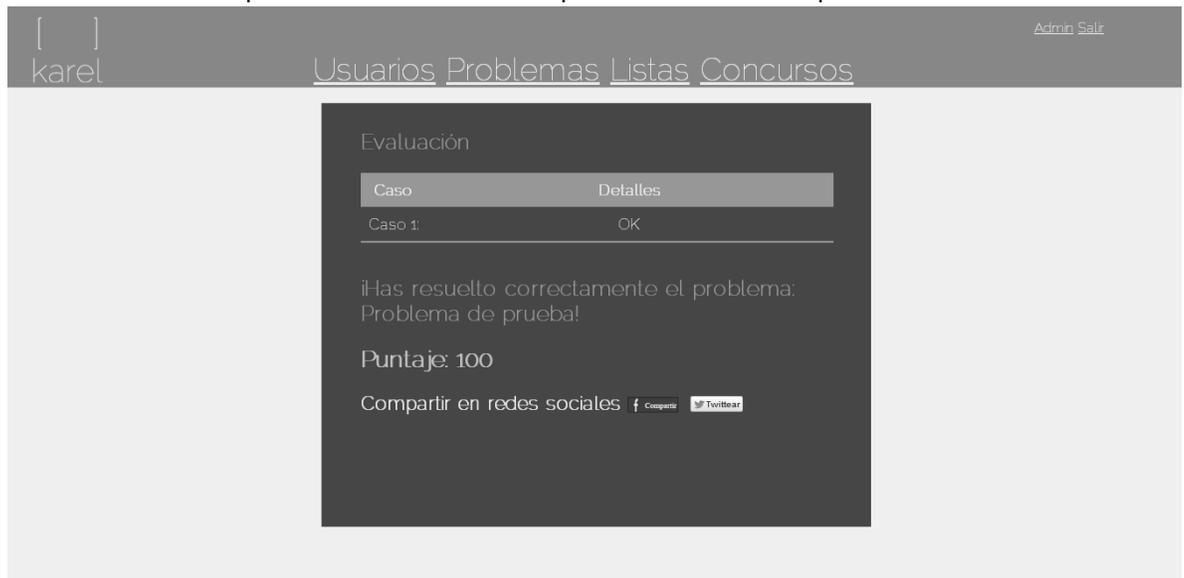
4. En la vista de detalles del problema, se puede enviar la solución a un problema en lenguaje Java y Pascal (el lenguaje por defecto seleccionado es Java).



5. Una vez seleccionado el archivo que satisface las condiciones del caso de evaluación, se da clic al botón de '¡Evaluar!'.

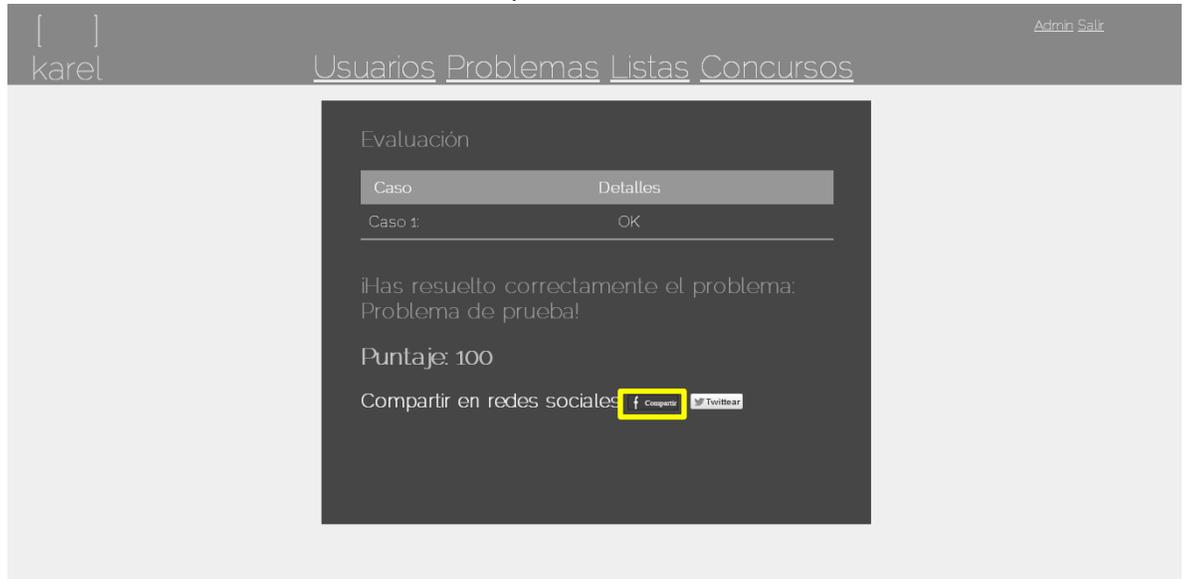


6. El sistema realiza la evaluación correspondiente al problema, una vez resuelto correctamente, se puede llevar a cabo una publicación en cualquier red social.

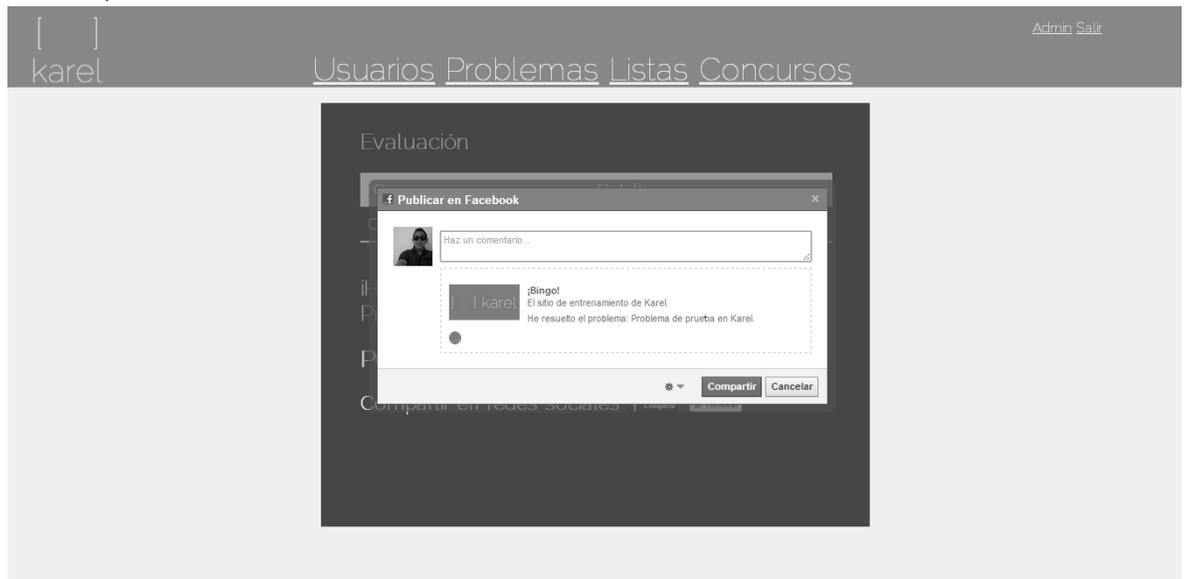


#### 4.1 Compartir contenido en Facebook – Resolución de un problema

1. Cuando la evaluación del problema ha sido correcta, el sistema permite compartir en el muro de Facebook del usuario, que ha resuelto un problema en Karel correctamente a través del botón 'Compartir' de Facebook.

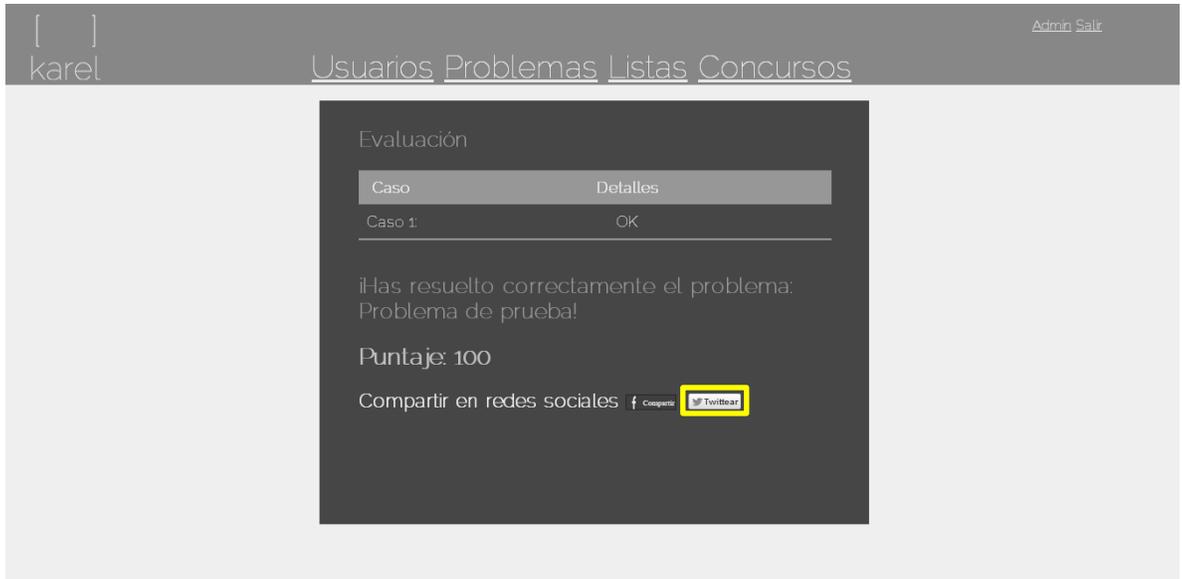


2. Al dar clic al botón, se abre una ventana emergente que permite hacer dicha publicación, la cual por defecto lleva el nombre del problema que ha sido resuelto (el enlace que es publicado en Facebook contiene un enlace a la dirección web del sistema).



## 4.2 Compartir contenido en Twitter – Resolución de un problema

1. Cuando la evaluación del problema ha sido correcta, el sistema permite compartir en la timeline de Twitter del usuario, que ha resuelto un problema en Karel correctamente a través del botón 'Twttear' de Twitter.

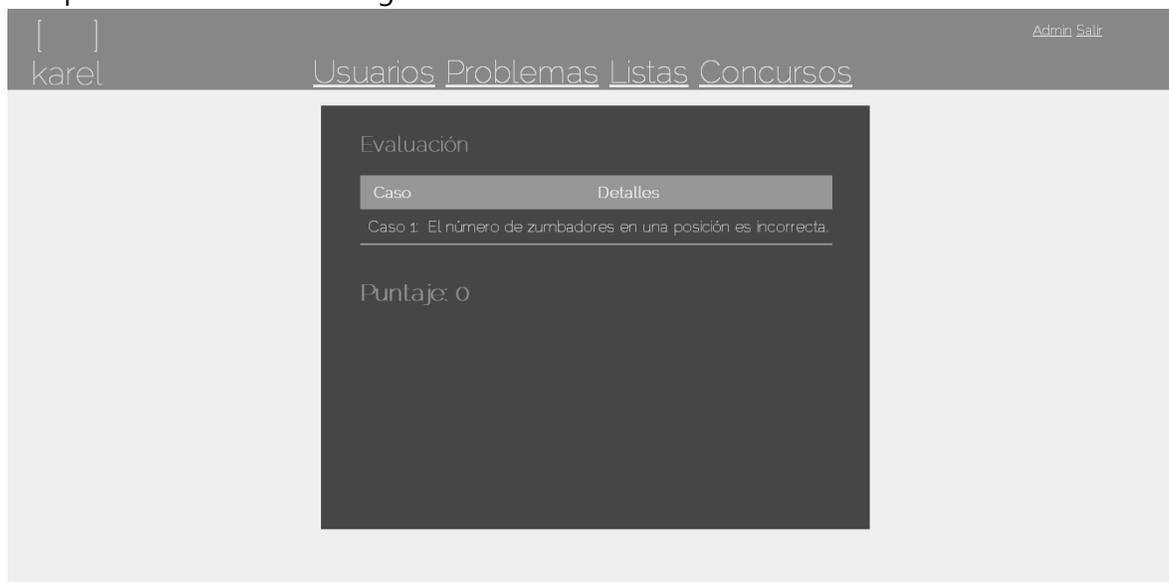


2. Al dar clic al botón, se abre una ventana emergente que permite hacer dicha publicación, la cual por defecto lleva el nombre del problema que ha sido resuelto, la dirección web del sistema y el hashtag '#Karel' en alusión a Karel el Robot.



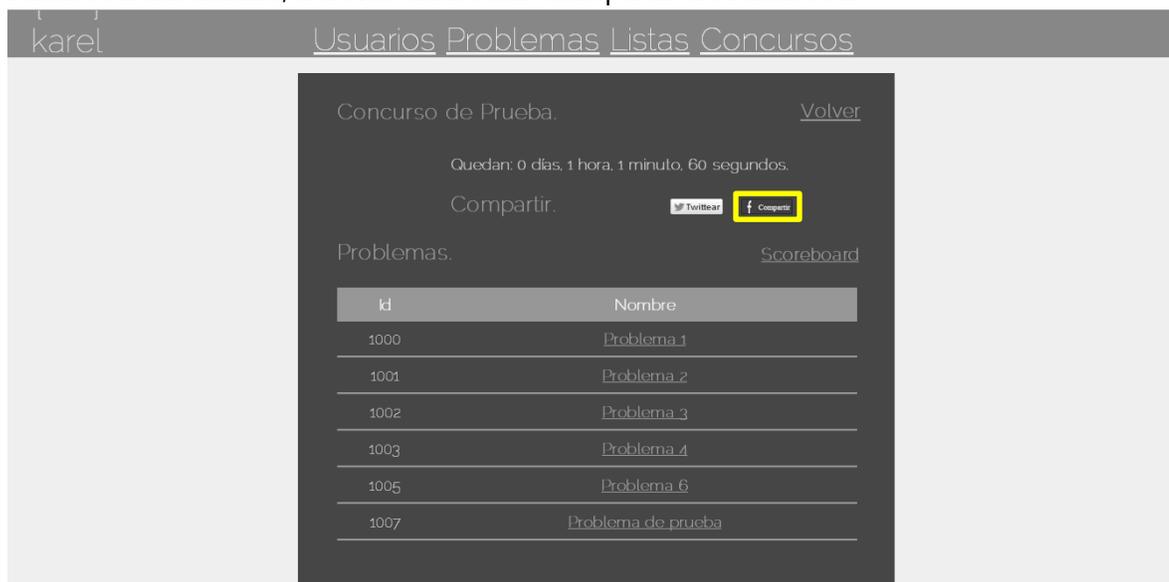
#### 4.3 Error en la evaluación – No se ha resuelto el problema correctamente

1. Cuando la evaluación del problema ha sido incorrecta, el sistema notifica los detalles por la cual la evaluación ha sido considerada incorrecta y no se permite compartir la evaluación en alguna red social.



#### 4.4 Compartir contenido en Facebook – Concurso

1. El sistema permite compartir los concursos existentes en Karel en el muro de Facebook del usuario, a través del botón 'Compartir' de Facebook.



- Al dar clic al botón, se abre una ventana emergente que permite hacer dicha publicación, la cual por defecto lleva el nombre del concurso (el enlace que es publicado en Facebook contiene un enlace a la dirección web del sistema).



#### 4.6 El concurso ha terminado

- Cuando el concurso ha terminado, el sistema muestra los problemas así como la notificación de que el concurso terminó.



2. Al visualizar un problema de un concurso, ya no se permite realizar la correspondiente evaluación.



#### 4.7 El concurso no ha comenzado

1. Cuando el concurso no ha comenzado, el sistema no muestra los problemas relacionados al concurso, evitando así evaluaciones anteriores a su inicio, únicamente muestra el tiempo faltante al comienzo del concurso.



## 5. Ver scoreboard

1. En la página de concursos, dar clic al vínculo 'Entrar' del concurso que se desea visualizar la tabla de resultados.



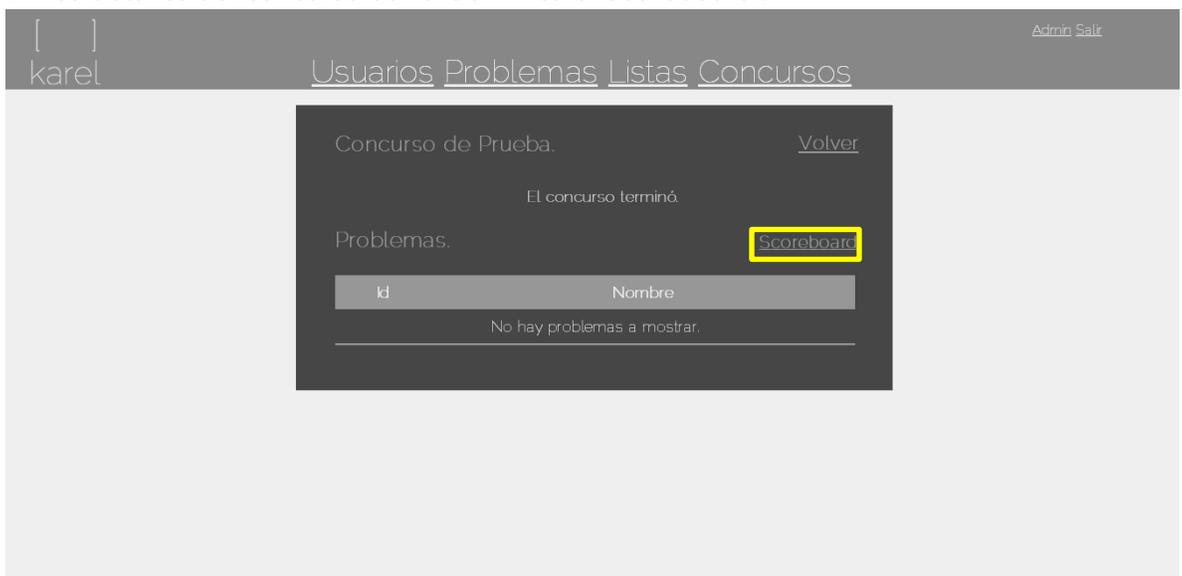
Concursos. [Admin](#) [Salir](#)

[Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

[Crear Nuevo Concurso](#)

id	Nombre	Fecha Inicio	Fecha Fin	
1	Concurso 1	02/05/2014 02:500 a. m.	02/05/2014 03:500 a. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
2	2	15/05/2014 01:000 a. m.	15/05/2014 02:000 a. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>
3	Concurso de Prueba	07/05/2014 06:1000 p. m.	07/05/2014 07:4000 p. m.	<a href="#">Entrar</a>   <a href="#">Editar</a>   <a href="#">Eliminar</a>

2. En los detalles del concurso dar clic al vínculo 'Scoreboard'.



Concurso de Prueba. [Volver](#)

El concurso terminó.

Problemas. [Scoreboard](#)

id	Nombre
No hay problemas a mostrar.	

3. El sistema mostrará la tabla de resultados del concurso del concurso.

Concurso de Prueba. [Admin](#) [Salir](#)

[karel](#) [Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

Concurso de Prueba. [Volver](#)

El concurso terminó.

Problemas.

Usuario	1000	1001	1002	1003	1005	1007	Score	Penalty
Mickey	100 (t)	100 (t)	100 (t)	0 (0)	0 (0)	0 (0)	300	0
Admin	100 (g)	100 (t)	0 (0)	0 (0)	0 (0)	100 (t)	300	4

Compartir.

### 5.1 Compartir contenido en Facebook – Scoreboard

1. La tabla de resultados se puede compartir en el muro de Facebook del usuario, a través del botón 'Compartir' de Facebook.

Concurso de Prueba. [Admin](#) [Salir](#)

[karel](#) [Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

Concurso de Prueba. [Volver](#)

Quedan: 0 días, 2 horas, 4 minutos, 46 segundos.

Problemas.

Usuario	1000	1001	1002	1003	1005	1007	Score	Penalty
Mickey	100 (t)	100 (t)	100 (t)	0 (0)	0 (0)	0 (0)	300	0
Admin	100 (g)	100 (t)	0 (0)	0 (0)	0 (0)	100 (t)	300	4

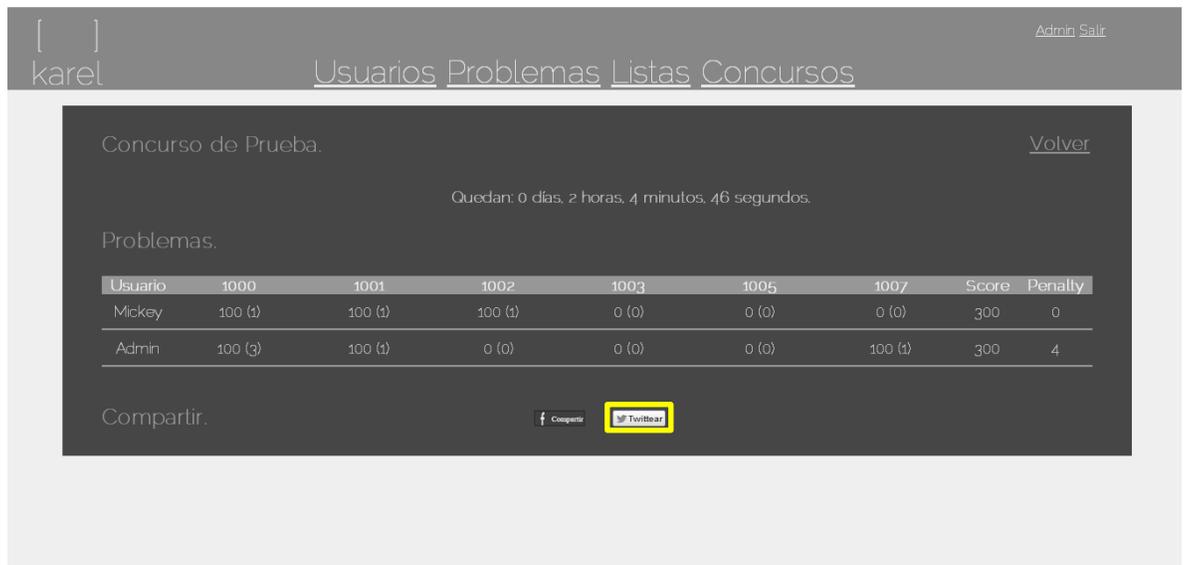
Compartir.

- Al dar clic al botón, se abre una ventana emergente que permite hacer dicha publicación, la cual por defecto lleva el nombre del concurso (el enlace que es publicado en Facebook contiene un enlace a la dirección web del sistema).



## 5.2 Compartir contenido en Twitter – Scoreboard

- La tabla de resultados se puede compartir en el timeline del usuario, a través del botón 'Twtitear' de Twitter.



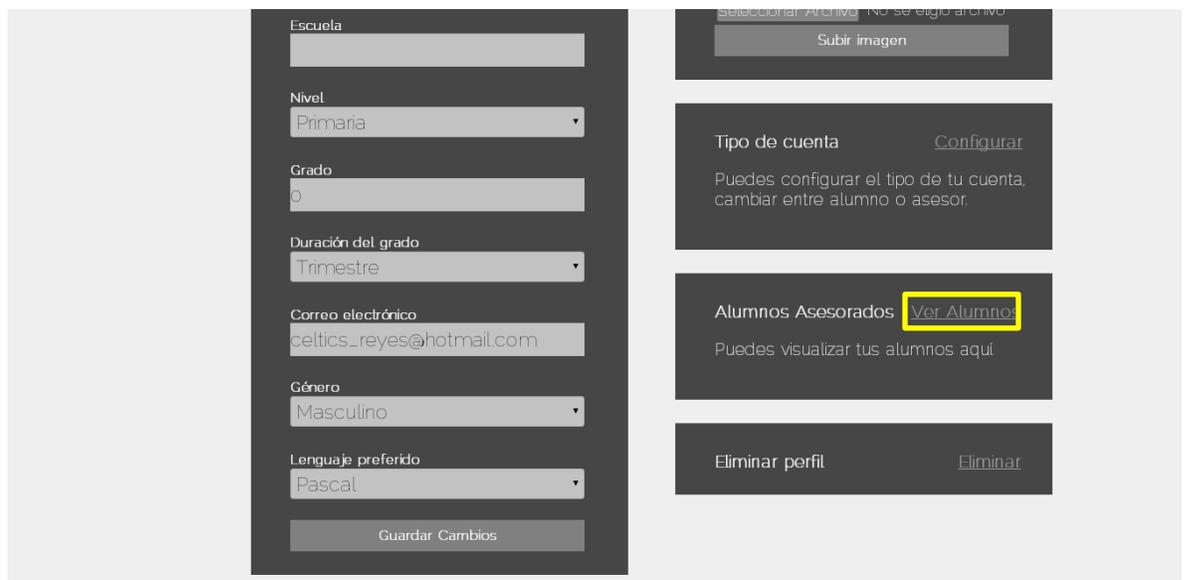
- Al dar clic al botón, se abre una ventana emergente que permite hacer dicha publicación, la cual por defecto lleva el nombre del concurso del cual se quiere compartir la tabla de resultados, la dirección web del sistema y el hashtag '#Karel' en alusión a Karel el Robot.



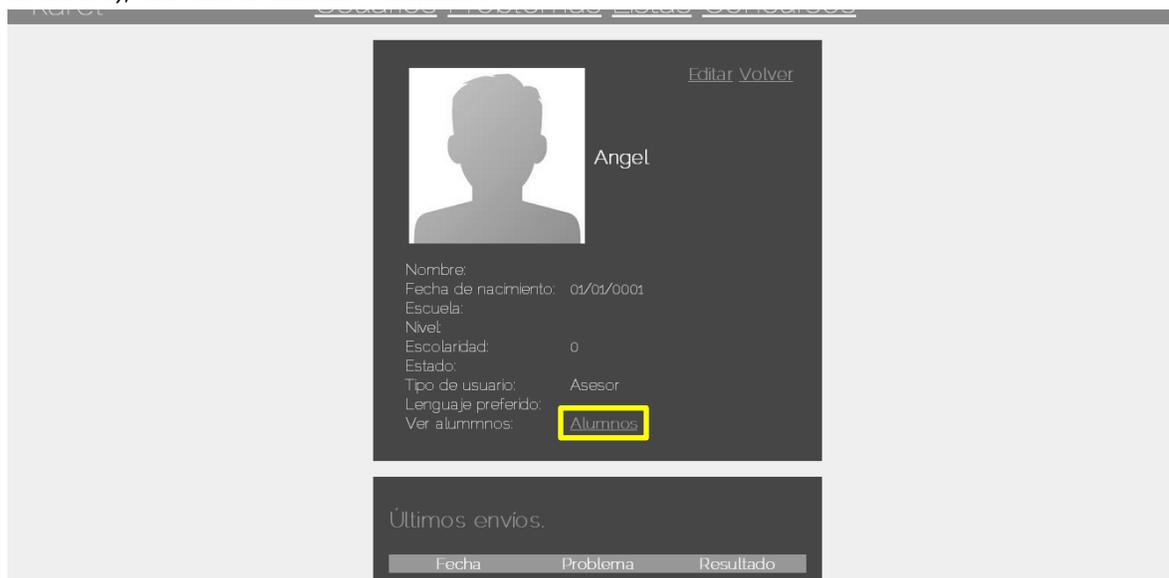
## Asesorías

### 1. Ver alumnos asesorados

- En la vista editar perfil dar clic al vínculo 'Ver Alumnos' del recuadro, alumnos asesorados.



- O en la vista detalles de perfil (al dar clic en el nombre del asesor en la lista de usuarios), dar clic al vínculo 'Alumnos'.



- El sistema mostrará los alumnos asesorados y ahí se podrán visualizar los alumnos asesorados, así como remover directamente.



## 2. Agregar asesor

1. En la ventana de editar perfil del usuario, dar clic al vínculo 'Configurar' del recuadro configurar tipo de cuenta.

Estado  
Aguascalientes

Escuela

Nivel  
Primaria

Grado

Duración del grado  
Trimestre

Correo electrónico

Género  
Masculino

Lenguaje preferido  
Pascal

Actualizar imagen de perfil  
Puedes subir una imagen en formato jpg como fotografía de perfil.  
[Seleccionar Archivo](#) No se eligió archivo  
[Subir imagen](#)

Tipo de cuenta [Configurar](#)  
Puedes configurar el tipo de tu cuenta, cambiar entre alumno o asesor.

Eliminar perfil [Eliminar](#)

2. Seleccionar el asesor que se desea agregar y dar clic al vínculo guardar.

karel [Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

Editar perfil [Volver](#)

Editar tipo de cuenta

Tipo de cuenta  
Alumno

Asesor  
Angel

[Guardar cambios](#)

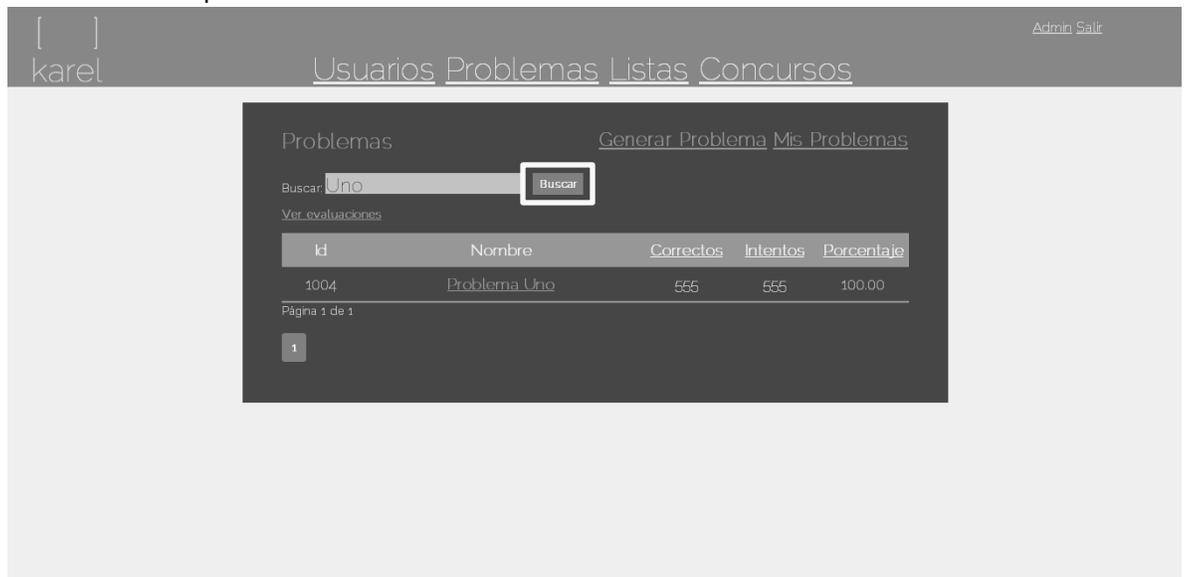
3. El sistema notifica que los cambios han sido guardados correctamente.



## Información adicional

### 1. Buscar problema

1. En la página de problemas, en el recuadro de búsqueda, escribir la consulta y dar clic en el botón 'Buscar', el sistema mostrará los resultados que concuerden con el criterio de búsqueda.



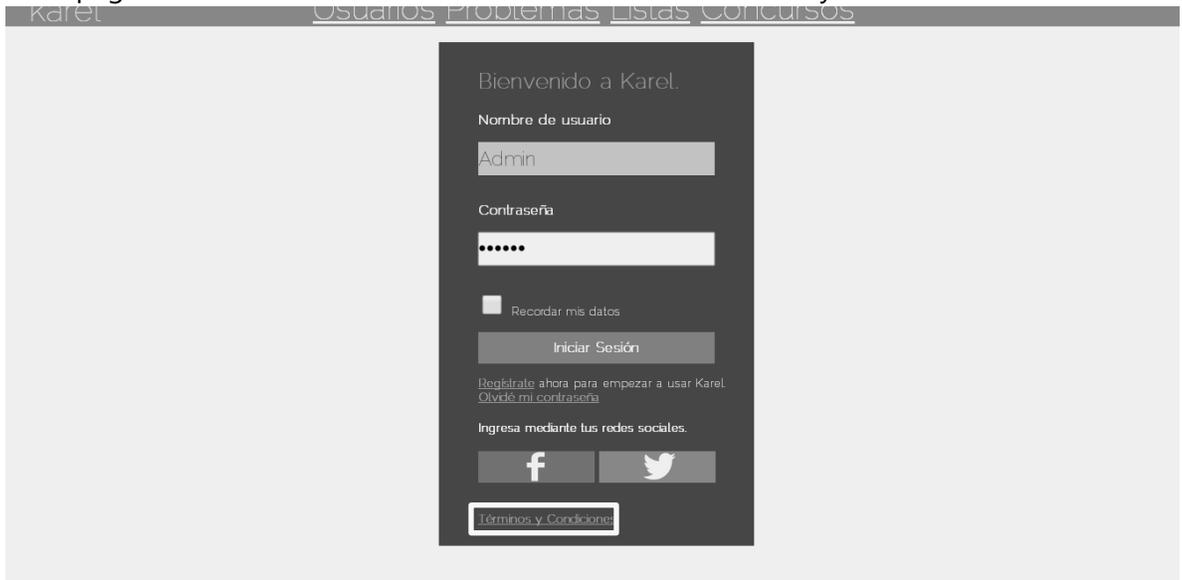
## 2. Buscar usuario

1. En la página de usuarios, en el recuadro de búsqueda, escribir la consulta y dar clic en el botón 'Buscar', el sistema mostrará los resultados que concuerden con el criterio de búsqueda.

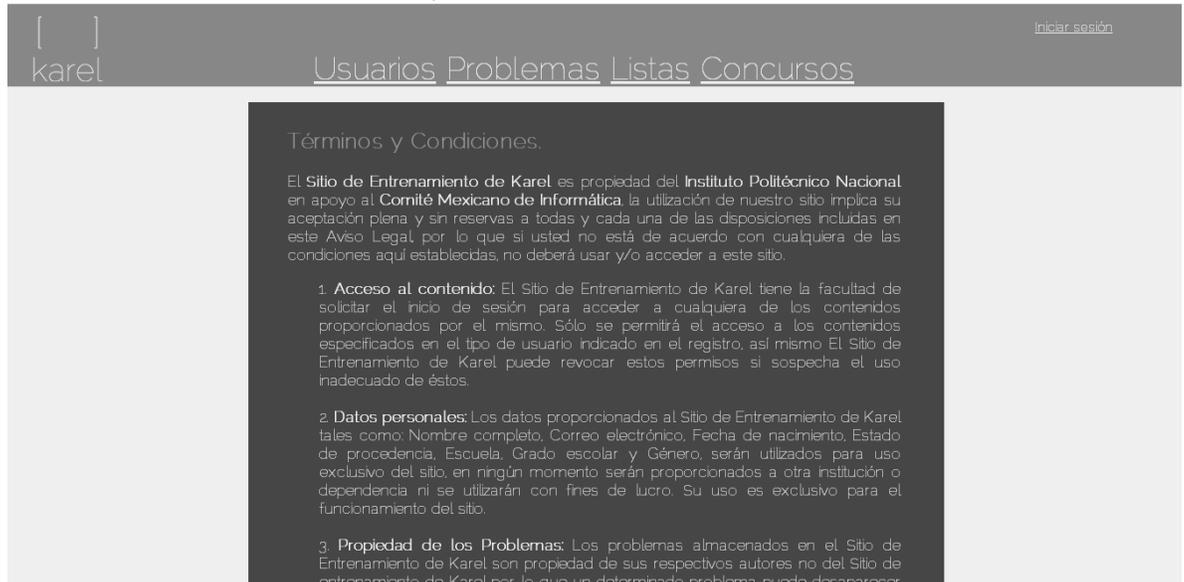


## 3. Ver términos y condiciones

1. En la página de inicio de sesión dar clic en el vínculo 'Términos y Condiciones'.



## 2. El sistema mostrará los términos y condiciones de Karel.



[ ]  
karel

Iniciar sesión

[Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

### Términos y Condiciones.

El Sitio de Entrenamiento de Karel es propiedad del Instituto Politécnico Nacional en apoyo al Comité Mexicano de Informática, la utilización de nuestro sitio implica su aceptación plena y sin reservas a todas y cada una de las disposiciones incluidas en este Aviso Legal, por lo que si usted no está de acuerdo con cualquiera de las condiciones aquí establecidas, no deberá usar y/o acceder a este sitio.

- 1. Acceso al contenido:** El Sitio de Entrenamiento de Karel tiene la facultad de solicitar el inicio de sesión para acceder a cualquiera de los contenidos proporcionados por el mismo. Sólo se permitirá el acceso a los contenidos especificados en el tipo de usuario indicado en el registro, así mismo El Sitio de Entrenamiento de Karel puede revocar estos permisos si sospecha el uso inadecuado de éstos.
- 2. Datos personales:** Los datos proporcionados al Sitio de Entrenamiento de Karel tales como: Nombre completo, Correo electrónico, Fecha de nacimiento, Estado de procedencia, Escuela, Grado escolar y Género, serán utilizados para uso exclusivo del sitio, en ningún momento serán proporcionados a otra institución o dependencia ni se utilizarán con fines de lucro. Su uso es exclusivo para el funcionamiento del sitio.
- 3. Propiedad de los Problemas:** Los problemas almacenados en el Sitio de Entrenamiento de Karel son propiedad de sus respectivos autores, no del Sitio de entrenamiento de Karel por lo que un determinado problema puede desaparecer.

## 4. Menú adicional de Karel

### 4.1. Círculo rojo

#### 1. El vínculo del círculo rojo, muestra una descripción de que es Karel.



[ ]  
karel

Iniciar sesión

[Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

# El sitio de entrenamiento de Karel.

APRENDE

¿QUÉ ES KAREL?

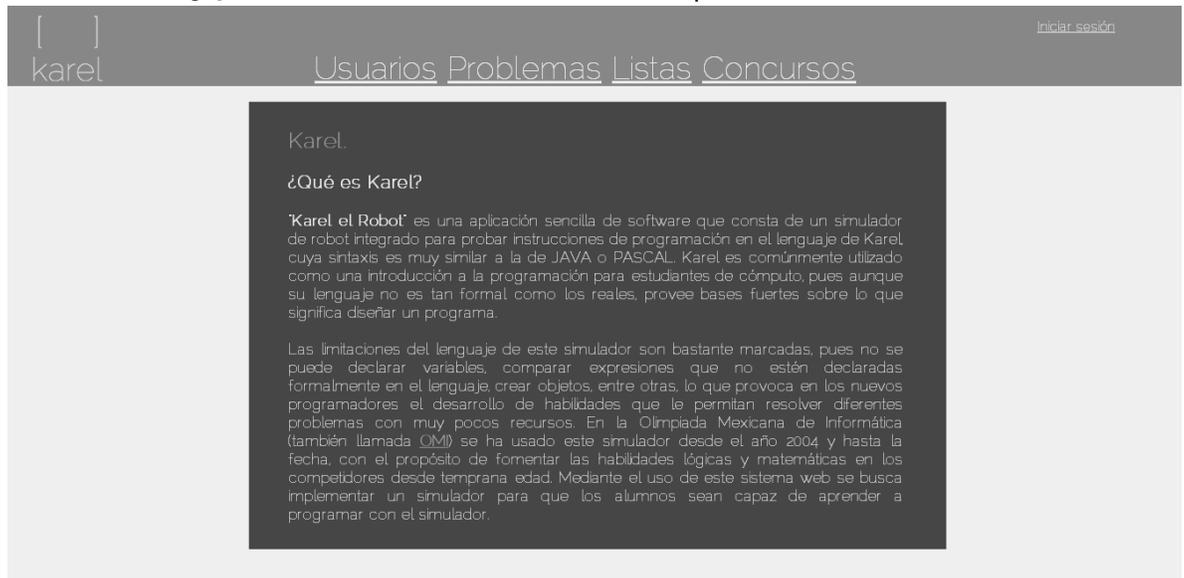
PROGRAMA

¿CÓMO FUNCIONA?

¡CONCURSA!

REGÍSTRATE

2. Al dar clic en '¿QUÉ ES KAREL?' se muestra una descripción breve de Karel.



[ ]  
karel

[Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

Iniciar sesión

Karel.

¿Qué es Karel?

"Karel el Robot" es una aplicación sencilla de software que consta de un simulador de robot integrado para probar instrucciones de programación en el lenguaje de Karel cuya sintaxis es muy similar a la de JAVA o PASCAL. Karel es comúnmente utilizado como una introducción a la programación para estudiantes de cómputo, pues aunque su lenguaje no es tan formal como los reales, provee bases fuertes sobre lo que significa diseñar un programa.

Las limitaciones del lenguaje de este simulador son bastante marcadas, pues no se puede declarar variables, comparar expresiones que no estén declaradas formalmente en el lenguaje, crear objetos, entre otras, lo que provoca en los nuevos programadores el desarrollo de habilidades que le permitan resolver diferentes problemas con muy pocos recursos. En la Olimpiada Mexicana de Informática (también llamada OMI) se ha usado este simulador desde el año 2004, y hasta la fecha, con el propósito de fomentar las habilidades lógicas y matemáticas en los competidores desde temprana edad. Mediante el uso de este sistema web se busca implementar un simulador para que los alumnos sean capaz de aprender a programar con el simulador.

#### 4.2. Círculo verde

1. El vínculo del círculo verde, muestra una descripción de como Karel.



[ ]  
karel

[Usuarios](#) [Problemas](#) [Listas](#) [Concursos](#)

Iniciar sesión

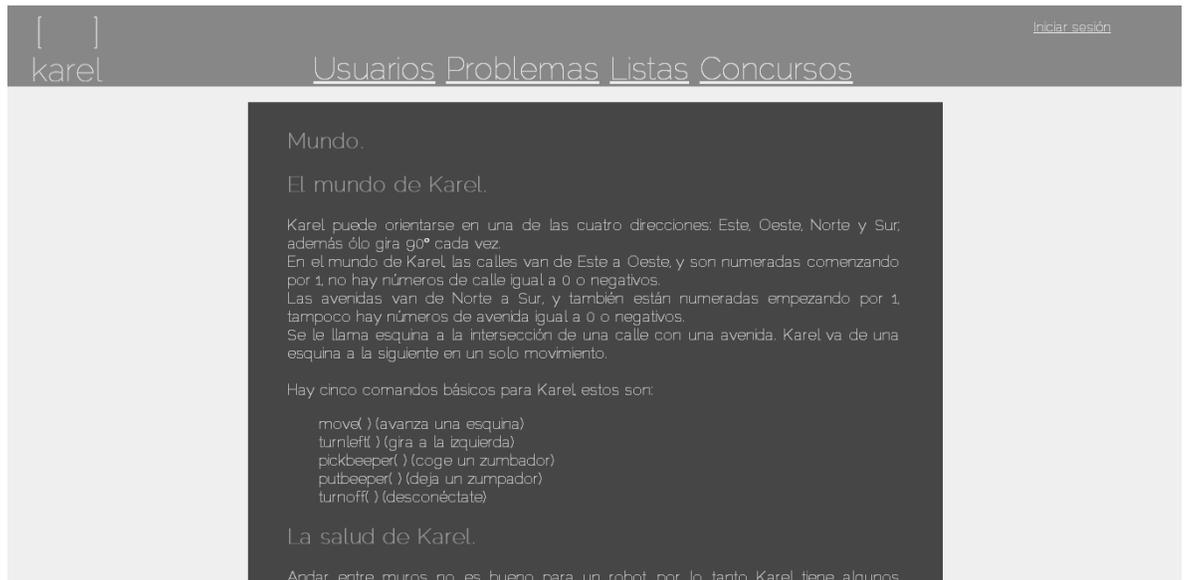
El sitio de entrenamiento de Karel.

APRENDE  
¿QUÉ ES KAREL?

PROGRAMA  
¿CÓMO FUNCIONA?

¡CONCURSA!  
REGÍSTRATE

2. Al dar clic en '¿CÓMO FUNCIONA?' se muestra una descripción breve de como funciona Karel.



#### 4.3. Círculo azul

1. El vínculo del círculo azul, muestra el formulario de registro del usuario.



2. Al dar clic en '¿Regístrate?' se muestra el formulario de registro de Karel.



The screenshot shows the Karel website's registration form. At the top left is the Karel logo, and at the top right is a link for "Iniciar sesión". The navigation menu includes "Usuarios", "Problemas", "Listas", and "Concursos". The registration form itself is a dark grey box with the following fields: "Registro. ¡Únete a Karel!", "Usuario" (text input), "Correo electrónico" (text input), "Contraseña" (password input), "Confirmar contraseña" (password input), and "Tipo de cuenta" (dropdown menu with "Alumno" selected). A "Regístrate" button is at the bottom.

#### 4.3.1- Círculo azul – Usuario ya registrado

1. Cuando el usuario intenta registrarse y ya está registrado, se puede compartir Karel en Facebook o Twitter.



The screenshot shows the Karel website's registration confirmation message. At the top left is the Karel logo, and at the top right is a link for "Admin Salt". The navigation menu includes "Usuarios", "Problemas", "Listas", and "Concursos". The confirmation message is a dark grey box with the following text: "Ya estás registrado!", "Pero puedes compartírnos en Facebook." with a Facebook share button, and "O en Twitter." with a Twitter share button.

# TRABAJO TERMINAL 2013-A003

Sistema web para la evaluación automática de programas de Karel el Robot.

## Anexo B: Manual de Instalación

Requerimientos necesarios.

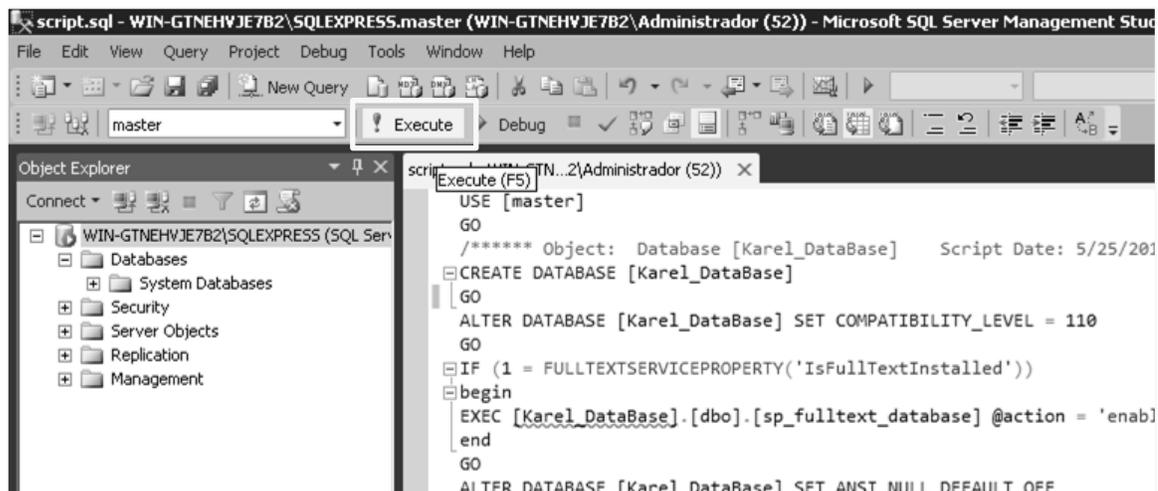
- Sistema operativo Windows Server 2008 o posterior.
- Servidor IIS con soporte para ASP.NET 4
- SQL Server 2008 o posterior.

Proceso de instalación.

Dentro de la carpeta Karel encontrara los siguientes archivos:

- Karel.sql: El script para crear la base de datos.
- Karel\_TT2013A003.zip: paquete de aplicación en donde se encuentran los códigos fuentes del sistema.

1. Abrir SQL Management , cargar el script de la base de datos y hacer clic en el icono  para generarla..

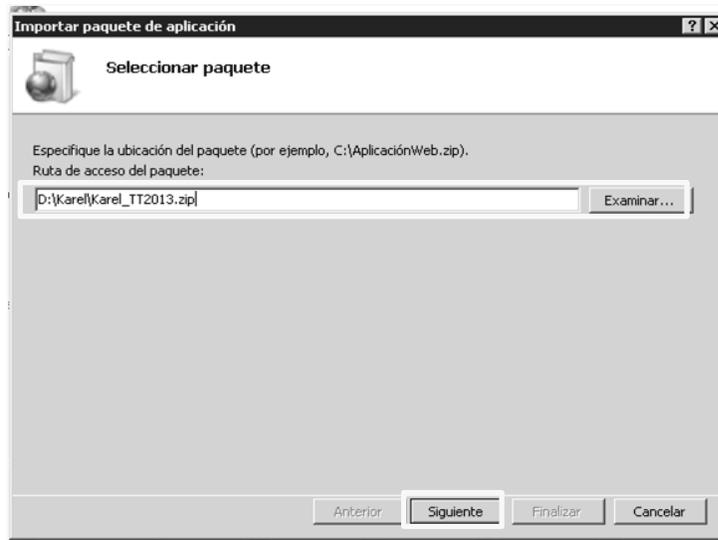


2. Abrir el Administrador de Internet Information Services.
3. En la página principal se selecciona la opción de agregar nuevo sitio.
4. En el campo nombre del sitio escribir "Karel"

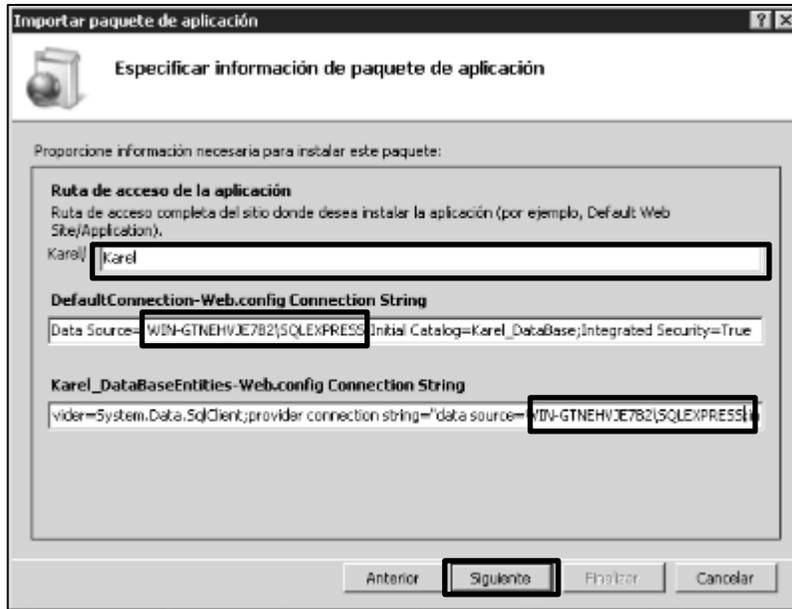
- Indicar la ruta física, para la cual se debe crear una carpeta con el nombre del sitio dentro de httpdocs.
- Definir el enlace como "http" y el puerto de ejecución como el "80".
- Hacer clic en el botón 'Aceptar'.



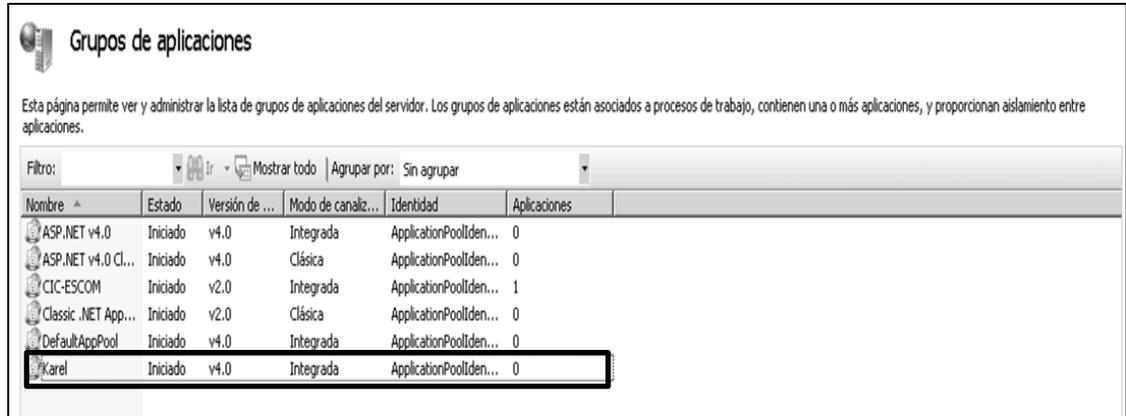
- Una vez creado el sitio de debe hacer clic sobre él y seleccionar la opción de importar el paquete de aplicación en donde se encuentran los códigos fuentes del sistema: "Karel\_TT2013.zip".
- Hacer clic en el botón 'Siguiente'.



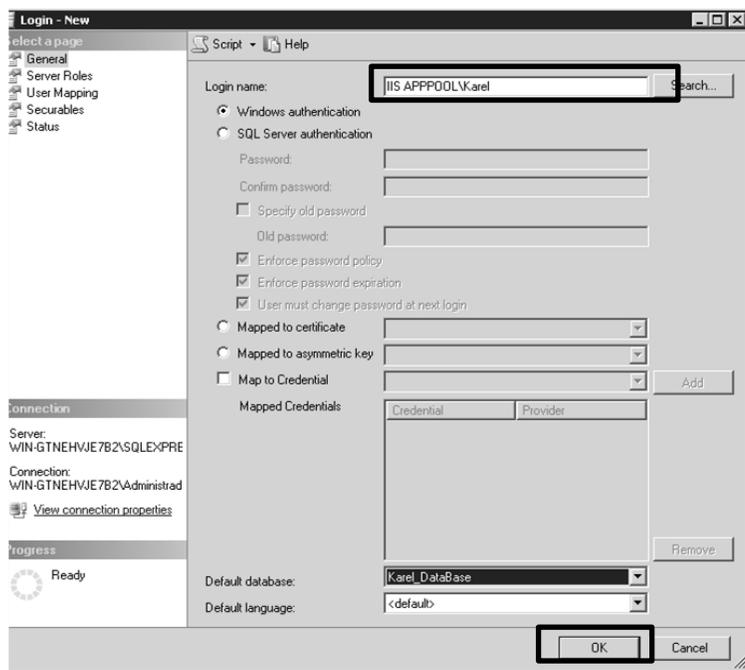
10. Indicar la ruta de acceso a la aplicación.
11. Cambiar las conexiones del web.config para indicar la conexión a la base de datos con el usuario correcto para acceder a SQL Server.



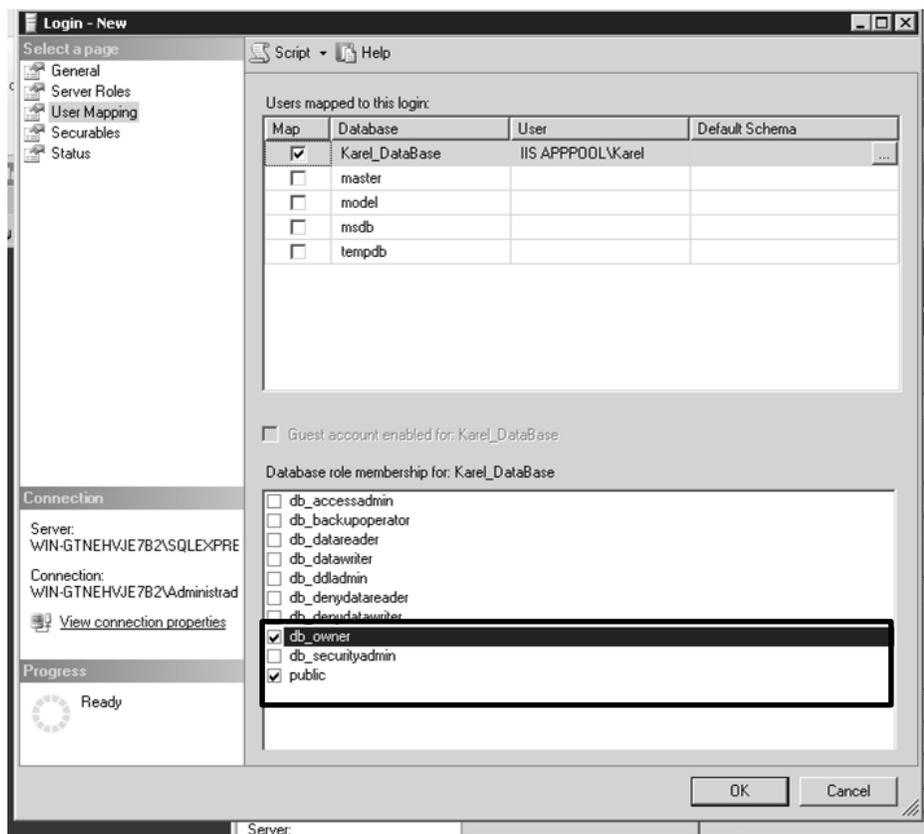
12. En la sección "Grupos de Aplicaciones" es necesario verificar que este configurado sobre ASP.NET 4.



13. Dentro de SQL Server es necesario generar el usuario IIS APPPOOL\Karel, indicando como base de datos a Karel\_Database



14. Asignarle permisos de usuario público.



15. Ingresar a la url del servidor y comprobar que el sitio funciona.

# TRABAJO TERMINAL 2013-A003

Sistema web para la evaluación automática de programas de Karel el Robot.

## Anexo C: Términos y condiciones del sitio

### **Términos y Condiciones del sitio de entrenamiento de Karel**

El **Sitio de Entrenamiento de Karel** es propiedad del **Instituto Politécnico Nacional** en apoyo al **Comité Mexicano de Informática**, la utilización de nuestro sitio implica su aceptación plena y sin reservas a todas y cada una de las disposiciones incluidas en este Aviso Legal, por lo que si usted no está de acuerdo con cualquiera de las condiciones aquí establecidas, no deberá usar y/o acceder a este sitio.

1. **Acceso al contenido:** El Sitio de Entrenamiento de Karel tiene la facultad de solicitar el inicio de sesión para acceder a cualquiera de los contenidos proporcionados por el mismo. Sólo se permitirá el acceso a los contenidos especificados en el tipo de usuario indicado en el registro, así mismo El Sitio de Entrenamiento de Karel puede revocar estos permisos si sospecha el uso inadecuado de éstos.

2. **Datos personales:** Los datos proporcionados al Sitio de Entrenamiento de Karel tales como: Nombre completo, Correo electrónico, Fecha de nacimiento, Estado de procedencia, Escuela, Grado escolar y Género, serán utilizados para uso exclusivo del sitio, en ningún momento serán proporcionados a otra institución o dependencia ni se utilizarán con fines de lucro. Su uso es exclusivo para el funcionamiento del sitio.

3. **Propiedad de los Problemas:** Los problemas almacenados en el Sitio de Entrenamiento de Karel pertenecen a sus respectivos autores no al Sitio de entrenamiento de Karel por lo que un determinado problema puede desaparecer en cualquier momento. El hacer uso de un problema en algún sitio ajeno implica dar crédito al autor del problema. Como autor de un problema le otorga al Sitio la facultad de publicarlo y utilizarlo de forma indefinida hasta que dicho problema sea eliminado del sistema.

4. **Disponibilidad del Sitio:** Aunque el Sitio de Entrenamiento de Karel pretende ofrecerle el mejor servicio posible, no puede garantizar que el Sitio esté libre de fallos. El acceso al sitio podrá ocasionalmente ser restringido para permitir reparaciones, mantenimiento o la introducción de nuevas instalaciones o nuevos servicios se intentará reanudar el acceso tan pronto como sea posible.

5. **Limitaciones:** No se podrá hacer uso del Sitio de Entrenamiento de Karel para los fines siguientes:

- a. transmitir materiales que fomenten conductas constitutivas de delito, que den lugar a responsabilidad civil o que, de cualquier otra forma, vulneren leyes reglamentos o códigos profesionales;
- b. obtener acceso no autorizado a otros sistemas de ordenador;
- c. interferir con el uso o el disfrute del Sitio por otra persona;
- d. incumplir cualesquiera normas legales relativas al uso de redes públicas de comunicaciones;
- e. interferir en redes o en sitios web conectados con el Sitio o interrumpirlos

6. **Sitios Web de Terceros:** Los permisos de sitios de terceros tales como Twitter y Facebook son responsabilidad del usuario al aceptar compartir el contenido del Sitio de Entrenamiento de Karel, en ningún momento se podrá acceder a la información del usuario almacenada en estos sitios, ya que solo proporcionan una forma de evitar el llenado del registro, la contraseña estará protegida por dichos servicios y el usuario tendrá la facultad de revocar estos permisos en el momento que lo desee.

7. **Terminaciones:** Usted acepta que El Sitio de Entrenamiento de Karel puede, bajo ciertas circunstancias y sin necesidad de notificación previa, cancelar y terminar inmediatamente su posibilidad de acceso al Sitio. Entre las causales de terminación se incluyen, sin limitarse:

- a. incumplimientos o violaciones a estos términos y condiciones;
- b. su solicitud (terminación de cuenta por requerimiento del Usuario);
- c. terminación o modificaciones sustanciales al Servicio (o cualquier parte del mismo);
- d. problemas técnicos o de seguridad inesperados;
- e. períodos de inactividad prolongados.

Asimismo, usted acepta que todas las causales de terminación con causa podrán ser invocadas por nosotros a nuestra sola discreción y que no seremos responsables frente a usted ni frente a ningún tercero por cualquier terminación de su cuenta, y las direcciones de correo electrónico asociada o acceso al Sitio.

Reservamos el derecho a modificar esta Declaración de Privacidad en cualquier momento. Es su deber verificar constantemente cualquier modificación Su uso continuado de este Sitio Web luego de la presentación de nuevos términos y condiciones implicará y significará su aceptación a las modificaciones introducidas.

*Gracias por hacer uso de Sitio de Entrenamiento de Karel.*