

UNIVERSIDAD DEL BÍO-BÍO
FACULTAD DE CIENCIAS EMPRESARIALES
DEPARTAMENTO DE SISTEMAS DE INFORMACIÓN



UNIVERSIDAD DEL BÍO-BÍO

**Generación Automática de Informes de Evaluación en
Lenguaje Natural de Proyectos de la asignatura Inteligencia
Artificial: Caso de Estudio - Algoritmos de Búsqueda en un
Espacio de Estados Mediante Descripción Lingüística de
Datos.**

Alumno:

Tomás Lermenda Senoceáin

Profesor guía:

Clemente Rubio-Manzano

Fecha:

28 de julio de 2017

Este proyecto se presenta para dar conformidad a los requisitos exigidos por la Universidad del Bío-Bío en el proceso de titulación para la carrera de Ingeniería Civil en Informática.

Resumen

Este proyecto se presenta para dar conformidad a los requisitos exigidos por la Universidad del Bío-Bío en el proceso de titulación para la carrera de Ingeniería Civil en Informática. El proyecto se titula Generación Automática de Informes de Evaluación en Lenguaje Natural de Proyectos de la asignatura Inteligencia Artificial: Caso de Estudio - Algoritmos de Búsqueda en un Espacio de Estados Mediante Descripción Lingüística de Datos. Es un proyecto de desarrollo de software que busca la creación de un sistema de generación automática de informes de evaluación para algoritmos de búsqueda en un espacio de estados empleando descripción lingüística de datos y generación automática de lenguaje natural.

Con la implementación de este proyecto se pueden evaluar los proyectos de la asignatura de Inteligencia Artificial de la carrera Ingeniería Civil en Informática de la Universidad del Bío-Bío automáticamente sin la necesidad de un exhaustiva evaluación lo cual permitirá seguir el progreso de los proyectos mientras se desarrollan y no depender de una evaluación final.

Índice General

1 INTRODUCCIÓN	8
2 DEFINICIÓN DE LA EMPRESA O INSTITUCIÓN	10
2.1 DESCRIPCIÓN DE LA EMPRESA	10
2.2 DESCRIPCIÓN DEL ÁREA DE ESTUDIO	12
2.3 DESCRIPCIÓN DE LA PROBLEMÁTICA	15
3 DEFINICIÓN PROYECTO	16
3.1 OBJETIVOS DEL PROYECTO	16
3.2 AMBIENTE DE INGENIERÍA DE SOFTWARE	16
3.3 DEFINICIONES, SIGLAS Y ABREVIACIONES	18
4 CONCEPTOS PRELIMINARES	19
4.1 CONCEPTOS CLAVE	19
4.2 ESTADO DEL ARTE	20
5 ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE	21
5.1 ALCANCES	21
5.2 OBJETIVO DEL SOFTWARE	21
5.3 DESCRIPCIÓN GLOBAL DEL PRODUCTO	22
5.3.1 INTERFAZ DE USUARIO	22
5.3.2 INTERFAZ DE HARDWARE	22
5.3.3 INTERFAZ SOFTWARE	22
5.3.4 INTERFACES DE COMUNICACIÓN	22
5.4 REQUERIMIENTOS ESPECÍFICOS	23
5.4.1 REQUERIMIENTOS FUNCIONALES DEL SISTEMA	23
5.4.2 INTERFACES EXTERNAS DE ENTRADA	24
5.4.3 INTERFACES EXTERNAS DE SALIDA	24
5.4.4 ATRIBUTOS DEL PRODUCTO	25
6 FACTIBILIDAD	26
6.1 FACTIBILIDAD TÉCNICA	26
6.2 FACTIBILIDAD OPERATIVA	27
6.3 FACTIBILIDAD ECONÓMICA	27
6.4 CONCLUSIÓN DE LA FACTIBILIDAD	30

7 ANÁLISIS	31
7.1 CASOS DE USO	31
7.1.1 ACTORES	31
7.1.2 DIAGRAMA DE CASOS DE USO Y DESCRIPCIÓN	31
7.1.3 ESPECIFICACIÓN DE LOS CASO DE USO	32
7.2 MATRIZ DE TRAZABILIDAD	35
7.3 MODELAMIENTO DE DATOS	36
8 DISEÑO	37
8.1 DISEÑO DE FÍSICO DE LA BASE DE DATOS	37
8.2 DISEÑO ARQUITECTÓNICO	38
8.3 DISEÑO INTERFAZ Y NAVEGACIÓN	39
9 DESARROLLO	40
9.1 INTRODUCCIÓN	40
9.2 VIDEOJUEGO: PROYECTO DEL PRIMER SEMESTRE DE INTELIGENCIA ARTIFICIAL	40
9.3 TRAZA DE EJECUCIÓN	43
9.4 EVALUACIÓN	43
10 PRUEBAS	48
10.1 ELEMENTOS DE PRUEBA	48
10.2 ESPECIFICACIÓN DE LAS PRUEBAS	48
10.3 RESPONSABLES DE LAS PRUEBAS	49
10.4 CALENDARIO DE PRUEBAS	49
10.5 DETALLE DE LAS PRUEBAS	49
10.6 CONCLUSIONES DE PRUEBA	52
11 PLAN DE CAPACITACIÓN Y ENTRENAMIENTO	53
12 PLAN DE IMPLANTACIÓN Y PUESTA EN MARCHA	54
13 RESUMEN ESFUERZO REQUERIDO	55
14 RESULTADOS	56
15 CONCLUSIONES	57
16 BIBLIOGRAFÍA	58
17 ANEXO: PLANIFICACIÓN INICIAL DEL PROYECTO	61
17.1.1 ESTIMACIÓN INICIAL DE TAMAÑO	62
18 ANEXO: DICCIONARIO DE DATOS DEL MODELO DE DATOS	64

19 ANEXO: MUESTRA DE RESÚMENES	68
20 ANEXO: TRAZAS DE EJECUCIÓN	71
21 ANEXO: APLICACIÓN WEB	74
22 ANEXO: MANUAL DE USUARIO	77

Índice Tablas

Tabla. 1. Requerimientos funcionales del sistema (informes)	23
Tabla. 2. Requerimientos funcionales del sistema (sitio web)	23
Tabla. 3. Requerimientos no funcionales del sistema	23
Tabla. 4. Interfaces externas de entrada	24
Tabla. 5. Interfaces externas de salida	24
Tabla. 6. Estimación puntos casos de uso	28
Tabla. 7. Matriz de trazabilidad	35
Tabla. 8. Conceptos del videojuego	41
Tabla. 9. Métricas de juego definidas	44
Tabla. 10. Percepciones definidas	45
Tabla. 11. Términos lingüísticos	47
Tabla. 12. Especificación de las pruebas	48
Tabla. 13. Detalle de las pruebas (registrar usuario)	49
Tabla. 14. Detalle de las pruebas (evaluar traza de ejecución)	51
Tabla. 15. Tabla de resumen de esfuerzo requerido	55
Tabla. 16. Estimación inicial de puntos casos de uso	62
Tabla. 17. Tablas diccionario de datos	64
Tabla. 18. Variables de la traza de ejecución	71

Índice Figuras

Fig. 1. Estructura organizativa de la Universidad del Bío-Bío	12
Fig. 2. Estructura organizativa del departamento de sistemas de información	14
Fig. 3. Diagrama de casos de uso	31
Fig. 4. Modelo Entidad-Relación	36
Fig. 5. Modelo físico	37
Fig. 6. Patrón de diseño arquitectónico del framework Yii2	38
Fig. 7. Diseño interfaz	39
Fig. 8. Imagen referencial de un proyecto de Inteligencia artificial	42
Fig. 9. Imágen referencial de un resumen generado a partir de una traza de ejecución	47
Fig. 10. Carta Gantt calendario de pruebas	49
Fig. 11. Carta Gantt calendario de entrenamiento	53
Fig. 12. Carta Gantt calendario de implantación y puesta en marcha	54
Fig. 13. Carta Gantt planificación inicial de proyecto	62
Fig. 14. Muestra de resúmenes	68
Fig. 15. Imágen referencial de una tabla añadida a un de proyecto	73
Fig. 16. Imágen referencial de una traza de ejecución de un proyecto	73
Fig. 17. Imágen referencial de la vista upload	75
Fig. 18. Imágen referencial de la vista assess	75
Fig. 19. Imágen referencial de la vista upload	76

1 INTRODUCCIÓN

La evaluación es una de las partes más importantes del proceso de enseñanza-aprendizaje ya que permite determinar si los estudiantes logran los objetivos de aprendizaje establecidos en la programación didáctica. Los resultados que se obtienen de la evaluación se entregan a los estudiantes con el objetivo de que éstos reciban una retroalimentación adecuada que les proporcionen información útil sobre su progreso y sobre cómo puede mejorar para alcanzar sus objetivos de aprendizaje.

En la asignatura de Inteligencia Artificial de la carrera de Ingeniería Civil en Informática de la Universidad del Bío-Bío se emplea una metodología de aprendizaje basada en proyectos. Los proyectos de la asignatura consisten en el diseño e implementación de algoritmos de búsqueda con el objetivo de dotar a los agentes (actores en un videojuego cuyos movimientos son desempeñados por un jugador virtual. Ver sección 9.2: Videojuego, página 40) de movimientos autónomos e inteligentes.

Actualmente la evaluación de los proyectos de la asignatura está limitada a una evaluación final al término del semestre de clases que se basa principalmente en la visualización del comportamiento de los agentes implementados en el proyecto perdiéndose detalles cualitativos y cuantitativos del desempeño.

La solución propuesta es un sistema de generación automática de informes de evaluación basado en el análisis de trazas de ejecución de los proyectos desarrollados por los estudiantes. El sistema permite generar una rápida evaluación del desempeño de los proyectos considerando métricas de calidad y de cantidad, proveyendo de un resumen de desempeño actual en cualquier momento del desarrollo.

Este documento presenta un proyecto que busca solucionar los problemas relacionados a la evaluación de estos proyectos.

El documento se estructura de la siguiente forma, en la sección 2 se describen la empresa y la problemática. En la sección 3 se definen los objetivos y el ambiente. En la sección 4 se tratan los conceptos claves y el estado del arte de la tecnología a desarrollar. En la sección 5 se especifican los requerimientos. En la sección 6 se trata el estudio de factibilidad. En la sección 7 se hace el análisis de casos de uso. En la sección 8 se

describe el diseño. En la sección 9 se describe el proceso de evaluación. En la sección 10 se describen las pruebas realizadas sobre el sistema. La sección 11 trata el proceso de entrenamiento. La sección 12 trata el plan de implantación. La sección 13 ofrece un breve resumen del esfuerzo requerido. La sección 14 ofrece algunos resultados preliminares del uso del sistema. La sección 15 ofrece conclusiones sobre el proyecto y el sistema. La sección 16 lista las referencias utilizadas en este documento. La sección 17 incluye la planificación inicial. La sección 18 incluye el diccionario de datos de la base de datos. La sección 19 presenta una muestra de resúmenes. La sección 20 explica las trazas de ejecución y la captura de datos. La sección 21 explica la aplicación web de forma sencilla. Finalmente 22 es un manual de usuario para nuevos usuarios, explica uso completo del sistema.

2 DEFINICIÓN DE LA EMPRESA O INSTITUCIÓN

2.1 Descripción de la empresa

Antecedentes generales de la Empresa

- nombre: Universidad del Bío-Bío
- RUT: 60.911.006-6
- Representante legal: Héctor Guillermo Gaete Feres
- dirección: Av. Collao N°1202 Casilla 5-C, Concepción, Región del Bío-Bío, Chile
- rubro: Educación
- servicios que ofrece: Otorga grados académicos y títulos profesionales

Entorno

La Universidad del Bío-Bío se encuentra en una región donde hay muchas universidades próximas, la mayoría ubicadas en Concepción y sus cercanías. Se destacan las universidades del consejo de rectores: Universidad de Concepción, Universidad Católica de la Santísima Concepción y Universidad Técnica Federico Santa María, pero también hay universidades privadas como la Universidad del Desarrollo, Universidad San Sebastián, entre otras, además están los múltiples institutos como: Instituto Profesional DuocUC, Instituto Profesional Virginio Gómez, entre otros. Todas estas instituciones cuentan con carreras relacionadas con el área informática que compiten directamente con la Universidad del Bío-Bío.

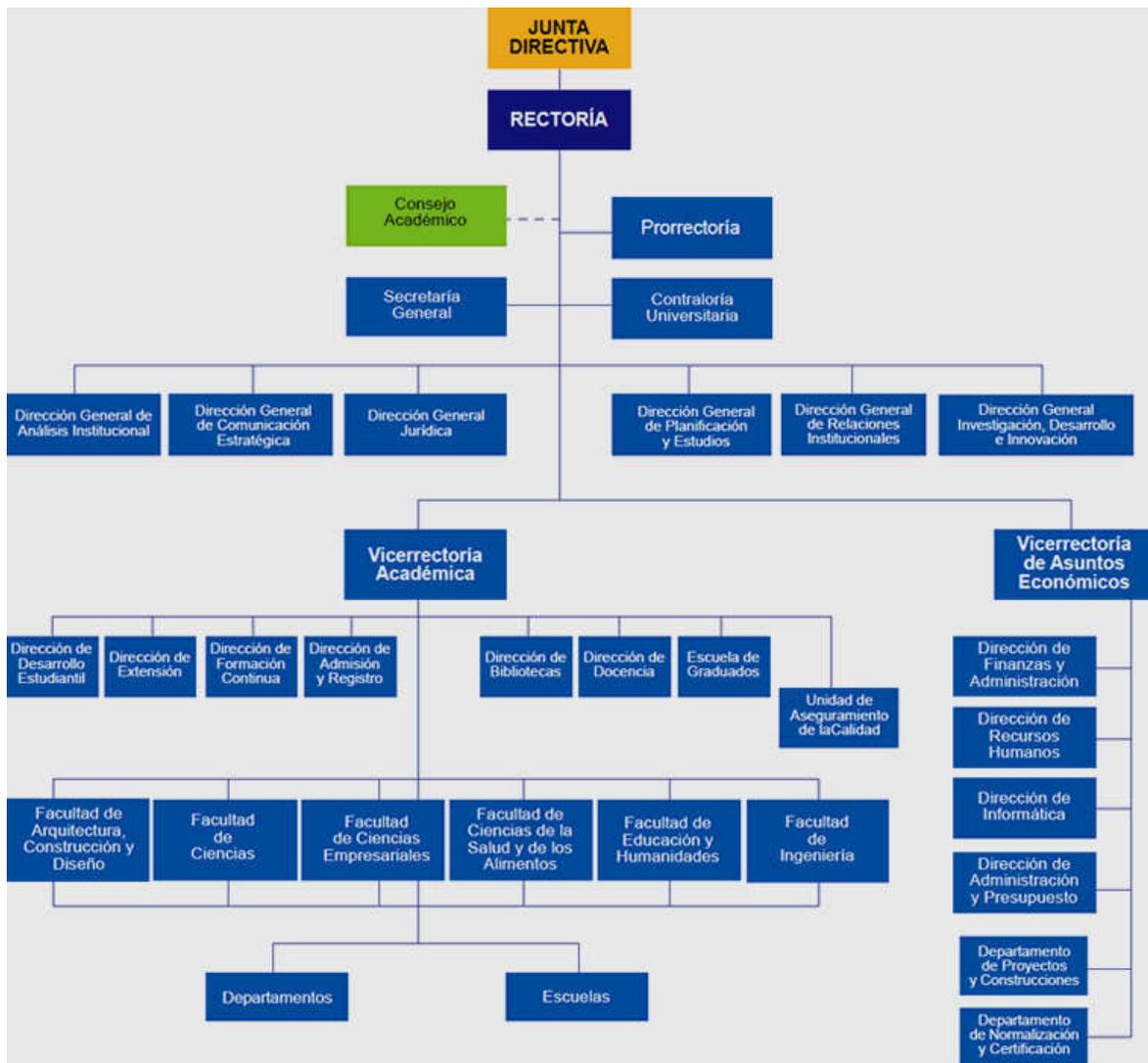
Visión

Ser reconocida a nivel nacional e internacional como una Universidad pública, responsable socialmente y regional que, comprometida con su rol estatal, desde la Región del Biobío, forma personas integrales de excelencia y aporta a través de su quehacer al desarrollo sustentable de la región y el país (Universidad del Bío-Bío, 2017).

Misión

La Universidad del Bío-Bío, a partir de su naturaleza pública, responsable socialmente y estatal, tiene por misión, desde la Región del Biobío, aportar a la sociedad con la formación de personas integrales, a través de una Educación Superior de excelencia. Comprometida con los desafíos de la región y del país, contribuye a la movilidad e integración social por medio de; la generación y transferencia de conocimiento avanzado, mediante la docencia de pregrado y postgrado de calidad, la investigación fundamental, aplicada y de desarrollo, la vinculación bidireccional con el medio, la formación continua y la extensión. Asimismo, impulsa el emprendimiento y la innovación, el fortalecimiento de la internacionalización y el desarrollo sustentable de sus actividades, basada en una cultura participativa centrada en el respeto a las personas (Universidad del Bío-Bío, 2017).

Estructura organizativa:



(Fig. 1. Organigrama. Estructura organizativa de la Universidad del Bío-Bío.)

2.2 Descripción del área de estudio

El área de estudio es el Departamento de Sistemas de Información de la Facultad de Ciencias Empresariales.

El profesor Clemente Rubio-Manzano es el profesor guía y cliente del proyecto. Es un académico Ingeniero Superior Informático, Máster en Tecnologías Informáticas Avanzadas y Doctor en Informática. También es director del grupo de investigación Software, Modelling and Science.

Estructura de Software, Modelling and Science

Director: Clemente Rubio-Manzano.

Integrantes DSI: Pedro Campos, Alejandra Segura, Christian Vidal-Castro.

Período: 2017-2018.

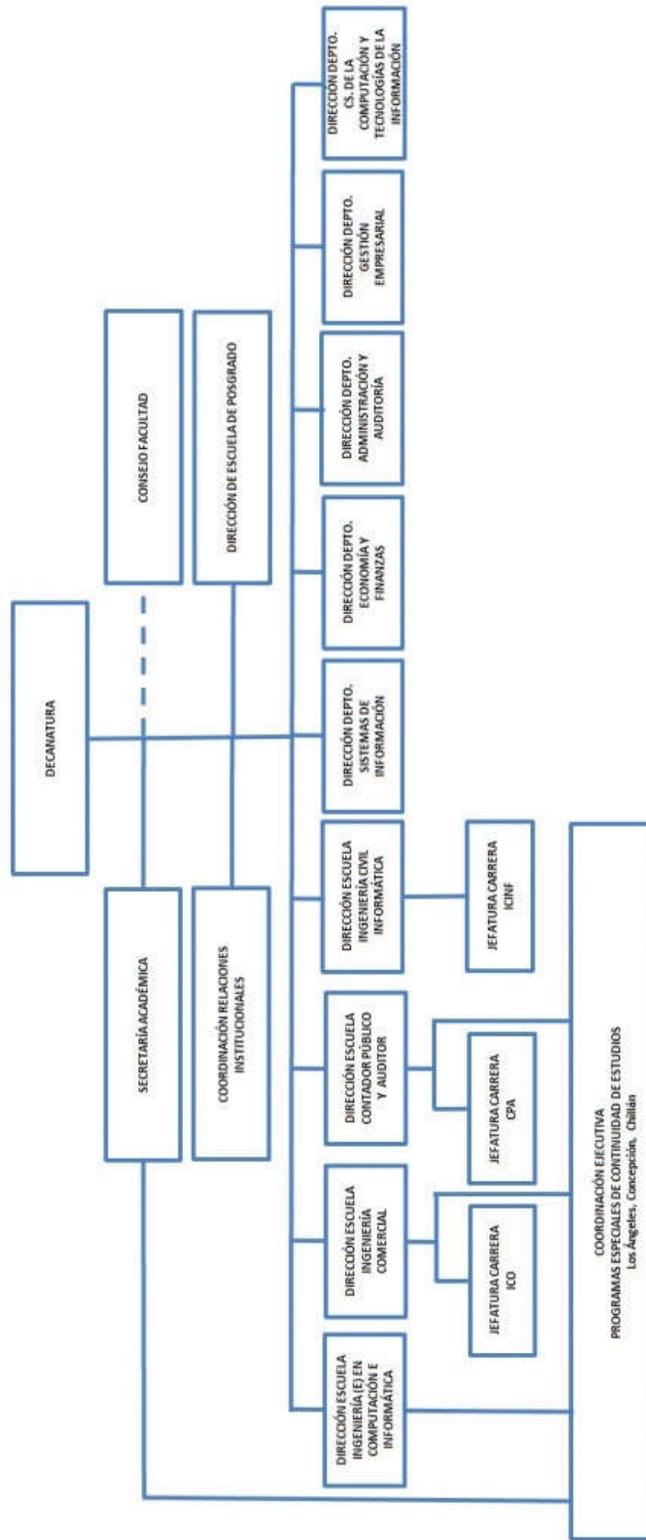
Visión

Ser un departamento reconocido a nivel nacional en las disciplinas de las Ciencias de la Computación y de la Informática, con vinculación nacional e internacional y excelencia en su quehacer académico, que aporta a la sociedad en los ámbitos de la docencia de pregrado y postgrado, generación de conocimiento mediante el desarrollo de la investigación y difusión de la disciplina (Departamento de Sistemas de Información, 2017).

Misión

Somos un departamento comprometido con la formación de excelencia de profesionales de pre y postgrado en las disciplinas de las Ciencias de la Computación y de la Informática; así como también con el desarrollo de la investigación para la generación de conocimiento y su difusión en la sociedad (Departamento de Sistemas de Información, 2017).

Estructura organizativa:



(Fig. 2. Estructura organizativa de la facultad de ciencias empresariales.)

2.3 Descripción de la problemática

En la asignatura de Inteligencia Artificial de la carrera de Ingeniería Civil en Informática de la Universidad del Bío-Bío se emplea una metodología de aprendizaje basada en proyectos. Los proyectos de la asignatura consisten en el diseño e implementación de algoritmos de búsqueda en un espacio de estados con el objetivo de dotar a agentes (Actores en un videojuego cuyos movimientos son desempeñados por un jugador virtual. Ver sección 9.2: Videojuego, página 40) de movimientos autónomos e inteligentes.

Actualmente la evaluación de los proyectos de la asignatura tiene notorias limitaciones, dado que los proyectos desarrollados por los estudiantes son complejos y generan una gran cantidad de datos, una evaluación detallada por parte del profesor no es posible, por el simple hecho de la cantidad de tiempo y esfuerzo que requeriría. La evaluación actual consiste en visualizar el comportamiento de los agentes implementados en el proyecto a través de la interfaz gráfica. Dadas las limitaciones de tiempo del curso la evaluación de los proyectos se realiza una vez al final del semestre, por lo que el feedback entregado a los estudiantes está limitado a la resolución de dudas específicas que los estudiantes tengan con respecto a problemas que perciban. Como el progreso no se evalúa si no hasta el final, el progreso de los estudiantes es desconocido durante el semestre, en otras palabras, los estudiantes desconocen que tan bien o mal va el desarrollo de su proyecto durante el semestre.

Las tareas realizadas por los agentes virtuales tienen detalles que no se pueden apreciar con la visualización o las trazas de ejecución clásicas. Una correcta evaluación de los algoritmos debería considerar métricas relacionadas con el comportamiento de los agentes, métricas que determinen la cantidad (de espacio y tiempo requerido) y la calidad (cuan buena ha sido la ejecución) de la ejecución de tal algoritmo.

3 DEFINICIÓN PROYECTO

3.1 Objetivos del proyecto

Objetivo general:

Diseño e implementación de un sistema de generación automática de informes de evaluación para algoritmos de búsqueda en un espacio de estados empleando descripción lingüística de datos y generación automática de lenguaje natural.

Objetivos específicos:

- Estado del arte sobre la generación automática de lenguaje natural y la descripción lingüística de datos.
- Estudiar la naturaleza y estructura del feedback en los proyectos de la asignatura Inteligencia Artificial.
- Identificar los requerimientos y establecer una alternativa de solución.
- Diseñar e implementar la solución.

3.2 Ambiente de Ingeniería de Software

Metodología:

La metodología a utilizar será una investigación a partir de la cual se llevará a cabo un desarrollo en cascada el cual se utilizará como caso de estudio. Se escogió el desarrollo en cascada puesto que los requerimientos son claros, están bien definidos y no cambiarán, la funcionalidad principal del sistema es la evaluación, al no tener muchas funcionalidades además de esa, se descartaron las metodologías iterativas/incrementales, puesto que no habrían incrementos que presentar.

Este método consiste en las siguientes etapas:

1. Elaborar estado del arte.
2. Análisis.
3. Diseño.
4. Implementación.
5. Pruebas.
6. Elaboración de resultados y conclusiones.
7. Documentación.

Estándar de documentación:

IEEE Std. 830-1998 (Software Engineering Standards Committee of the IEEE Computer Society, 1998). Para cumplir con el formato de la plantilla del proyecto de título de desarrollo de software de la carrera.

Herramientas de apoyo al desarrollo de software:

- NetBeans 8 y Sublime Text 3 como entornos de desarrollo integrado (IDE).
- Apache2 como servidor de aplicaciones.
- Java 8 y PHP 5.6 como lenguajes de programación.
- MySQL 5.6 como sistema de gestión de base de datos.
- phpMyAdmin 4.7 para administrar MySQL de forma visual.
- Yii2 como entorno de trabajo (Software Framework), extendido con el módulo Yii2-User para la administración de usuarios y mPDF para generar archivos PDF.

Herramientas de diseño:

- Draw.io para el diagrama de casos de uso.
- PowerDesigner para el modelo físico y modelo E-R.
- Wireframe.cc para el diseño de interfaz.

3.3 Definiciones, Siglas y Abreviaciones

RF	Requerimiento Funcional.
RNF	Requerimiento No Funcional.
DE	Datos de Entrada.
IS	Interfaz de Salida.
CU	Caso de Uso.
PU	Prueba Unitaria
CPU	Caso Prueba Unitaria
GLMP	Granular Linguistic Model of a Phenomenon

4 CONCEPTOS PRELIMINARES

4.1 Conceptos claves

- Informe: es una descripción, oral o escrita, de las características y circunstancias de un suceso o asunto.(Real Academia Española,2017a)
- Evaluar: es estimar los conocimientos, aptitudes y rendimiento de los alumnos.(Real Academia Española,2017b)
- Lenguaje natural: es un lengua que es nativa de un pueblo (Collins Dictionary,2017), se distingue de lenguajes artificiales o formales.
- Espacio de estados: es el conjunto de todas las configuraciones que un determinado problema y su entorno podrían lograr. A cada una de las configuraciones se le llama estado, y contiene información estática y dinámica. (University of Miami's Department of Computer Science,2017)

Un amplio rango de problemas pueden ser formulados como búsquedas, las cuales se pueden resolver utilizando algoritmos de búsqueda.

- Algoritmos de búsqueda en un espacio de estados: se refieren a algoritmos que resuelven problemas buscando los caminos posibles a una solución (estado objetivo).
- Aprendizaje Basado en Proyectos: es un método de enseñanza en el que los estudiantes obtienen conocimientos y habilidades trabajando durante un período de tiempo prolongado para investigar y responder a una pregunta, problema o desafío auténtico, atractivo y complejo.(Buck Institute for Education,2017)

4.2 Estado del arte

Las aplicaciones de generación automática de informes en materia de tecnología educativa (e-learning) que conocemos hasta el momento son la evaluación del aprendizaje del algoritmo de Dijkstra (Sánchez-Torrubia, Torres-Blanc & Trivino, 2012), la generación de informes sobre estudiantes y los factores que los afectan durante el semestre (Gkatzia, Hastie, Janarthanam & Lemon, 2013) y la generación de informes de la actividad de los estudiantes (Ramos-Soto, Lama, Vazquez-Barreiros, Bugarín, Mucientes & Barro, 2015). La generación automática de informes de evaluación no se ha aplicado a la ejecución de algoritmos de búsqueda en un espacio de estados.

El proyecto plantea un sistema de generación automática de informes de evaluación basado en el análisis de trazas de ejecución de los proyectos, los informes resultantes están escritos en lenguaje natural, permiten entregar feedback al estudiante sobre el desempeño de su proyecto, lo que está haciendo bien y lo que está haciendo mal, información suficiente para hacer recomendaciones sobre cómo se pueden mejorar los proyectos.

Considerando que no se tiene conocimiento de la existencia aplicaciones de aplicaciones de generación automática de informes en materia de tecnología educativa relacionadas con la ejecución de algoritmos de búsqueda en un espacio de estados, se establece que este proyecto es una novedosa aplicación de la generación automática de informes.

5 ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE

5.1 Alcances

El sistema genera automáticamente informes de evaluación en lenguaje natural (LN) basados en el análisis de trazas de ejecución de proyectos del primer semestre de 2017 de la asignatura de Inteligencia artificial de la carrera Ingeniería Civil en Informática de la Universidad del Bío-Bío. Permite subir trazas de ejecución para evaluarlas, entrega un informe a partir de estas y permite exportar el informe a un archivo PDF.

El sistema también permite el registro de usuarios para el uso de sus datos (nombre y RUT) en la cabecera del archivo PDF.

El usuario interactúa con el sistema a través de un sitio web por medio de un navegador web de escritorio.

5.2 Objetivo del software

Objetivo general:

Generar automáticamente informes de evaluación en lenguaje natural (LN) que evalúen el desempeño de proyectos del primer semestre de 2017 de la asignatura de Inteligencia artificial de la carrera Ingeniería Civil en Informática de la Universidad del Bío-Bío para entregar feedback sobre qué tan bueno o malo ha sido el desempeño del proyecto.

Objetivos específicos:

Generar automáticamente informes de evaluación en lenguaje natural (LN) de los proyectos del primer semestre de 2017 de la asignatura de Inteligencia artificial que consideren métricas de cantidad (memoria utilizada, iteraciones).

Generar automáticamente informes de evaluación en lenguaje natural (LN) de los proyectos del primer semestre de 2017 de la asignatura de Inteligencia artificial que consideren métricas de calidad (qué tan bueno es el agente de la inteligencia artificial capturando recompensas, ¿rápido?, ¿valiente?, etc.).

5.3 Descripción Global del Producto

5.3.1 Interfaz de usuario

El informe exportado a un archivo pdf (RF-03) debe mostrar (si se quiere) el RUT y el nombre completo del usuario en la parte superior del documento.

El formulario registro de usuarios (RF-04) debe incluir los campos de nombre completo y RUT.

El Sitio Web (RF-05) debe incluir el logo de SOMOS (SOftware-MOdeling-Science - UBB).

5.3.2 Interfaz De Hardware

El sistema no utilizará ningún periférico hardware en particular para funcionar.

5.3.3 Interfaz Software

El sistema no interactúa con ningún otro sistema de la empresa, por lo que actúa independientemente y no fue diseñado para integrarse con otros sistemas en el futuro puesto que no se han informado planes futuros para ello.

5.3.4 Interfaces de comunicación

El usuario interactúa con el sistema por medio un navegador web utilizando el protocolo HTTP.

5.4 Requerimientos Específicos

5.4.1 Requerimientos Funcionales del sistema

Informes:

Id	Nombre	Descripción
RF-01	Subir traza de ejecución	El sistema debe permitir la subida de trazas de ejecución para su evaluación
RF-02	Generar informes	El sistema debe generar automáticamente informes de evaluación en lenguaje natural que consideren métricas de cantidad y de calidad.
RF-03	Exportar informes	El sistema debe permitir exportar el informe de evaluación generado automáticamente a un archivo pdf.

(Tabla. 1. Requerimientos funcionales del sistema (informes).)

Sitio Web:

Id	Nombre	Descripción
RF-04	Registro de usuarios	El sistema debe permitir el registro de usuarios para que sus datos (nombre, RUT) aparezcan (o no) en el archivo pdf.

(Tabla. 2. Requerimientos funcionales del sistema (sitio web).)

Requerimientos no funcionales:

Id	Nombre	Descripción
RNF-01	Sitio Web	El sistema debe proveer un sitio web para la interacción con los usuarios.

(Tabla. 3. Requerimientos no funcionales del sistema.)

5.4.2 Interfaces externas de entrada

Identificador	Nombre del ítem.	Detalle de Datos contenidos en ítem
DE-01	Subir traza de ejecución	Archivo *.txt
DE-02	Registro de usuario	Correo electrónico, Nombre de usuario, Nombre completo, RUT, Contraseña
DE-03	Iniciar sesión	Nombre de usuario, Contraseña

(Tabla. 4. Interfaces externas de entrada.)

5.4.3 Interfaces externas de Salida

Identificador	Nombre del ítem.	Detalle de Datos contenidos en ítem	Medio Salida
IS-01	Informe de evaluación	Nombre, RUT	Archivo PDF

(Tabla. 5. Interfaces externas de salida.)

5.4.4 Atributos del producto

Portabilidad: el sistema será compatible y utilizable en los navegadores internet explorer, mozilla firefox, google chrome y opera.

Usabilidad: el sistema cuenta con una gran variedad de mensajes de error para distintas situaciones (si se suben archivos en un formato diferente a txt, o si el archivo txt no está formateado correctamente, si el rut a registrar es incorrecto, repetido, mal escrito, etc.) que advierten a los usuarios que están haciendo mal.

Seguridad: el sistema tiene un control de acceso a la funcionalidad más importante, subir archivo, y más allá de eso, todos los campos relativos a las interfaces de entrada tienen validaciones para impedir cualquier situación fuera de orden.

Perfil de usuario registrado: Solo un usuario registrado puede subir una traza de ejecución para evaluar.

6 FACTIBILIDAD

6.1 Factibilidad técnica.

Se cuenta con un servicio de hosting contratado previamente el cual cuenta con todo lo necesario para el desarrollo y operación del sistema de información, incluyendo un entorno de pruebas además del de producción.

A continuación se describen los entornos desarrollo, pruebas y producción:

Entorno de desarrollo:

- Sistema operativo Windows 10
- NetBeans 8
- Sublime Text 3
- Apache2
- Java 8
- PHP 5.6
- MySQL 5.6
- phpMyAdmin 4.7

Entorno de pruebas:

- Sistema operativo GNU/Linux
- Apache2
- PHP 5.6
- MySQL 5.6
- phpMyAdmin 4.7

Entorno de producción:

- Sistema operativo GNU/Linux
- Apache2
- PHP 5.6
- MySQL 5.6
- phpMyAdmin 4.7

6.2 Factibilidad operativa.

El sistema apoya el proceso de evaluación de los proyectos de la asignatura de Inteligencia artificial de la carrera Ingeniería Civil en Informática de la Universidad del Bío-Bío.

Permite que el profesor o los alumnos obtengan una rápida evaluación del desempeño de los proyectos por medio de un resumen en lenguaje natural que se genera automáticamente con solo subir un archivo de traza de ejecución al sistema.

La evaluación generada por el sistema no requiere intervención del profesor para generarse y el profesor no requiere ver el código de fuente o la ejecución del proyecto para interpretar el resumen generado.

La información contenida en el informe es suficiente para que el profesor pueda entregar feedback al estudiante sobre cómo mejorar su proyecto.

6.3 Factibilidad económica.

Considerando que el software a utilizar es gratuito y el hosting a utilizar ya se había contratado previamente, por lo que no añade ningún costo, se procede a estimar el valor de las horas hombre (HH) destinadas a este proyecto.

Estimación personal:

En base al estimado de esfuerzo requerido, se establece que la cantidad de horas a destinar al proyecto es de 520 HH. Se da un valor 0,3 UF a cada HH.

Calculando el valor de la siguiente forma:

$$\text{Costo total} = \text{Horas Hombre} * \text{Valor Hora Hombre}$$

$$156 \text{ UF} = 520 \text{ HH} * 0,3 \text{ UF}$$

En base a este cálculo y considerando que no hay otros cálculos se determina que es el costo total del sistema y representa un ahorro de 156 UF.

Estimación puntos casos de uso:

FACTOR DE PESO DE LOS ACTORES SIN AJUSTAR				
Tipo de Actor	Descripción	Factor de Peso	Nro Actores	Subtotal (UAW)
Simple	Otro Sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API, Application Programming Interface)	1	0	0
Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto	2	0	0
Complejo	Una persona que interactúa con el sistema mediante una interfaz	3	1	3
Total Actores (UAW)			1	3

FACTOR DE PESOS DE CASO DE USO SIN AJUSTAR				
Tipo de Caso de Uso	Descripción	Factor de Peso	Nro Casos de Uso	Subtotal (UUCW)
Simple	El caso de uso contiene de 1 a 3 transacciones	5	3	15
Medio	El caso de uso contiene de 4 a 7 transacciones	10	0	0
Complejo	El caso de uso contiene más de 8 transacciones	15	0	0
Total Use Cases (UUCW)			3	15

Puntos de Caso de Uso Sin Ajustar (UUCP)	18
---	-----------

FACTORES DE COMPLEJIDAD TÉCNICA (TCF)				
Factor	Descripción	Factor de Peso	Valor Asignado (0-5)	Subtotal
T1	Sistema distribuido	2	0	0
T2	Objetivos de performance o tiempo de respuesta	2	5	10
T3	Eficiencia del usuario	1	4	4
T4	Procesamiento interno complejo	1	5	5
T5	El código debe ser reutilizable	1	3	3
T6	Facilidad de instalación	0.5	2	1
T7	Facilidad de uso	0.5	5	2.5
T8	Portabilidad	2	5	10
T9	Facilidad de cambio	1	3	3
T10	Concurrencia	1	0	0
T11	Incluye objetivos especiales de seguridad	1	3	3
T12	Provee accesos directos a terceras partes	1	0	0
T13	Se requieren facilidades especiales de entrenamiento a usuarios	1	1	1
Total Factor				42.5

FACTORES DE AMBIENTE (EF)				
Factor	Descripción	Factor de Peso	Valor Asignado	Subtotal (EF)
E1	Familiaridad con el modelo del proyecto Utilizado (RUP, MSF, Métrica3)	1.5	3	4.5
E2	Experiencia en la aplicación (web, standalone, EAI, SOA, BPM, SOA-BPM)	0.5	2	1
E3	Experiencia en orientación a objetos (Proc, OO, Aspect)	1	3	3
E4	Lead analyst capability	0.5	3	1.5
E5	Motivación	1	5	5
E6	Estabilidad de los requerimientos	2	4	8
E7	Personal Part-time	-1	0	0
E8	Dificultad del lenguaje de programación (Java, C#, VB, SQL) <w -1,2>	-1	3	-3
Total Factor Ambiente (EF)				20

ESTIMANDO LOS UCPs		
Param	Description	Calculated Value
UUCP	Unadjusted UCPs	18
TCF	Technical Complexity Factor	1.025
ECF o EF	Environmental Complexity Factor	0.800
UCP	Total Adjusted UCPs	14.76
Capacity Factor (min 10, avg 20, max 40) hours/UCP		10

(Tabla. 6. Estimación puntos casos de uso.)

Calculando Horas Hombre:

Esfuerzo Horas Hombre = Puntos de Casos de Uso ajustados * Horas-Persona

$$295,2 = 14,76 * 20$$

Calculando costo total:

Costo total = Horas Hombre * Valor Hora Hombre

$$88,56 \text{ UF} = 295,2 \text{ HH} * 0,3 \text{ UF}$$

6.4 Conclusión de la factibilidad

Se puede apreciar una considerable diferencia entre la estimación personal y la estimación de puntos de casos de uso, donde la esta última tiene problemas para cuantificar las horas hombre requeridas cuando hay complejidades que no se reflejan en muchos casos de uso, como es el caso de este proyecto donde debe desarrollarse una nueva aplicación para una tecnología. Por lo que se preferencia la estimación personal por sobre los puntos de casos de uso, considerándose un ahorro significativo.

Tras considerar la factibilidad técnica, operativa y económica se establece que el proyecto es factible de realizar ya que no tiene costos de desarrollo e implementación y entrega considerables beneficios a los usuarios del sistema.

7 ANÁLISIS

7.1 Diagrama de casos de uso

7.1.1 Actores

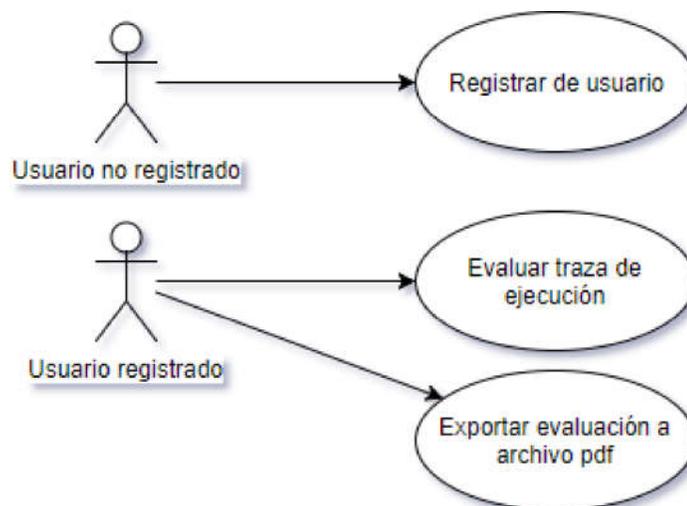
Usuario no registrado:

Corresponde a cualquier persona que acceda a al sitio web, no requiere ningún nivel de conocimientos técnicos. Más allá de acceder al sitio web la única interacción que tiene con el sistema es la de registrarse como usuario del sistema.

Usuario registrado:

Corresponde a cualquier persona que haya iniciado sesión como usuario registrado del sistema, se espera que sea un profesor o estudiante que esté cursando la asignatura de Inteligencia artificial de la carrera Ingeniería Civil en Informática de la Universidad del Bío-Bío y posea previamente una traza de ejecución válida. Puede subir trazas de ejecución para su evaluación y exportar las evaluaciones a un archivo pdf.

7.1.2 Casos de Uso y descripción



(Fig. 3. Diagrama de casos de uso.)

El diagrama de casos de uso representa las interacciones de los usuarios con el sistema, donde todos los usuarios, registrados o no, acceden al sistema de la misma forma.

Todo usuario accede al sistema a través de un computador conectado a internet, utilizando un navegador web, en el cual escriben la dirección:

<http://www.youractionsdefineyou.com/assess>

Desde ahí llegan a la página principal del sitio web del sistema y se aplican los casos de uso para las interacciones de los usuarios.

7.1.3 Especificación de los Casos de Uso

7.1.3.1 Caso de Uso: CU-01 Registrar usuario

- Actores: Usuario no registrado
- Objetivo: Registrar un usuario en la base de datos del sistema.
- Pre-Condiciones: El usuario no se ha registrado previamente y se encuentra en la página de registro de usuarios.
- Flujo de Eventos Básicos:

Al actor	El sistema
1. El usuario no registrado completa los campos de registro de usuario y entonces hace clic sobre el botón “registrarse”.	2. El sistema valida los campos rellenos por el usuario y lo registra en la base de datos como un nuevo usuario registrado. El sistema también envía un mensaje de confirmación al correo electrónico del usuario.
3. (Opcional) El usuario ahora registrado, revisa el mensaje de confirmación enviado a su correo electrónico y hace clic en el enlace de confirmación.	4. El sistema confirma la dirección de correo electrónico del usuario y cambia el estatus de la cuenta a confirmada.

- Flujo de Eventos Alternativo:

Al actor	El sistema
1.(a) El usuario no registrado completa los campos de registro con valores inválidos (símbolos o letras en el RUT, una dirección de correo mal escrita, etc.) y entonces hace clic sobre el botón “registrarse”.	5. El sistema muestra un mensaje de error específico para el error y el campo que se relleno con valores inválidos.
1.(b) El usuario no registrado completa los campos registro que utilizan valores únicos (nombre de usuario, correo electrónico, RUT) con valores ya existentes en la base de datos y entonces hace clic sobre el botón “registrarse”.	6. El sistema muestra un mensaje de debajo del campo específico diciendo que el valor ingresado ya ha sido utilizado.

- Post-Condiciones: La página se recarga, desplegando un mensaje advirtiendo el éxito o fracaso en el registro de usuario.

7.1.3.2 Caso de Uso: CU-02 Evaluar traza de ejecución

- Actores: Usuario registrado
- Objetivo: Generación automática de un informe de evaluación escrito en lenguaje natural a partir de un archivo de traza de ejecución.
- Pre-Condiciones: El usuario registrado ha iniciado sesión en el sistema y se encuentra en la página principal del sitio web.
- Flujo de Eventos Básicos:

Al actor	El sistema
1. El usuario registrado hace clic en el botón “examinar” ubicado debajo de “seleccionar archivo” (escrito en letras de color negro) y selecciona un archivo de traza de ejecución (*.txt) de su disco duro.	2. El sistema valida que el archivo seleccionado tenga la extensión *.txt, de tenerla, entonces el sistema cambia el color de seleccionar archivo a verde.

3. El usuario registrado hace clic en el botón “Subir”.	4. El sistema valida que el archivo de traza de ejecución esté formateado correctamente y entonces procede a realizar la evaluación del contenido de la traza de ejecución.
---	---

- Flujo de Eventos Alternativo:

Al actor	El sistema
1(a) El usuario registrado hace clic en el botón “examinar” ubicado debajo de “seleccionar archivo” (escrito en letras de color negro) y selecciona un archivo de su disco duro con una extensión distinta a *.txt.	5. El sistema muestra un mensaje en letras rojas debajo de “examinar” advirtiendo que solo se aceptan archivos con la extensión *.txt.
2(a) El usuario registrado hace clic en el botón “Subir”, pero el archivo *.txt seleccionado no se encuentra formateado como una traza de ejecución.	6. El sistema muestra un mensaje advirtiendo que el archivo subido no se encuentra en el formato correcto, que necesita ser verificado y vuelto a subir.

- Post-Condiciones: El usuario registrado es redirigido a una página que muestra los resultado del informe de evaluación escrito en lenguaje natural

7.1.3.3 Caso de Uso: CU-03 Exportar la evaluación a archivo pdf

- Actores: Usuario registrado
- Descripción: Exportar el informe de evaluación escrito en lenguaje natural generado automaticamente a un archivo pdf.
- Pre-Condiciones: El usuario registrado ha iniciado sesión en el sistema y se encuentra la página que muestra los resultados del informe de evaluación escrito en lenguaje natural.
- Flujo de Eventos Básicos:

Al actor	El sistema
1. El usuario registrado hace clic en el botón “PDF” al término del informe.	2. El sistema crea un archivo pdf a partir del contenido de los resultados del informe de evaluación.

- Flujo de Eventos Alternativo:

Al actor	El sistema

- Post-Condiciones: El usuario registrado es redirigido al documento pdf a través del navegador.

7.2 Matriz de trazabilidad

	CU-01	CU-02	CU-03
RF-01		x	
RF-02		x	
RF-03			x
RF-04	x		

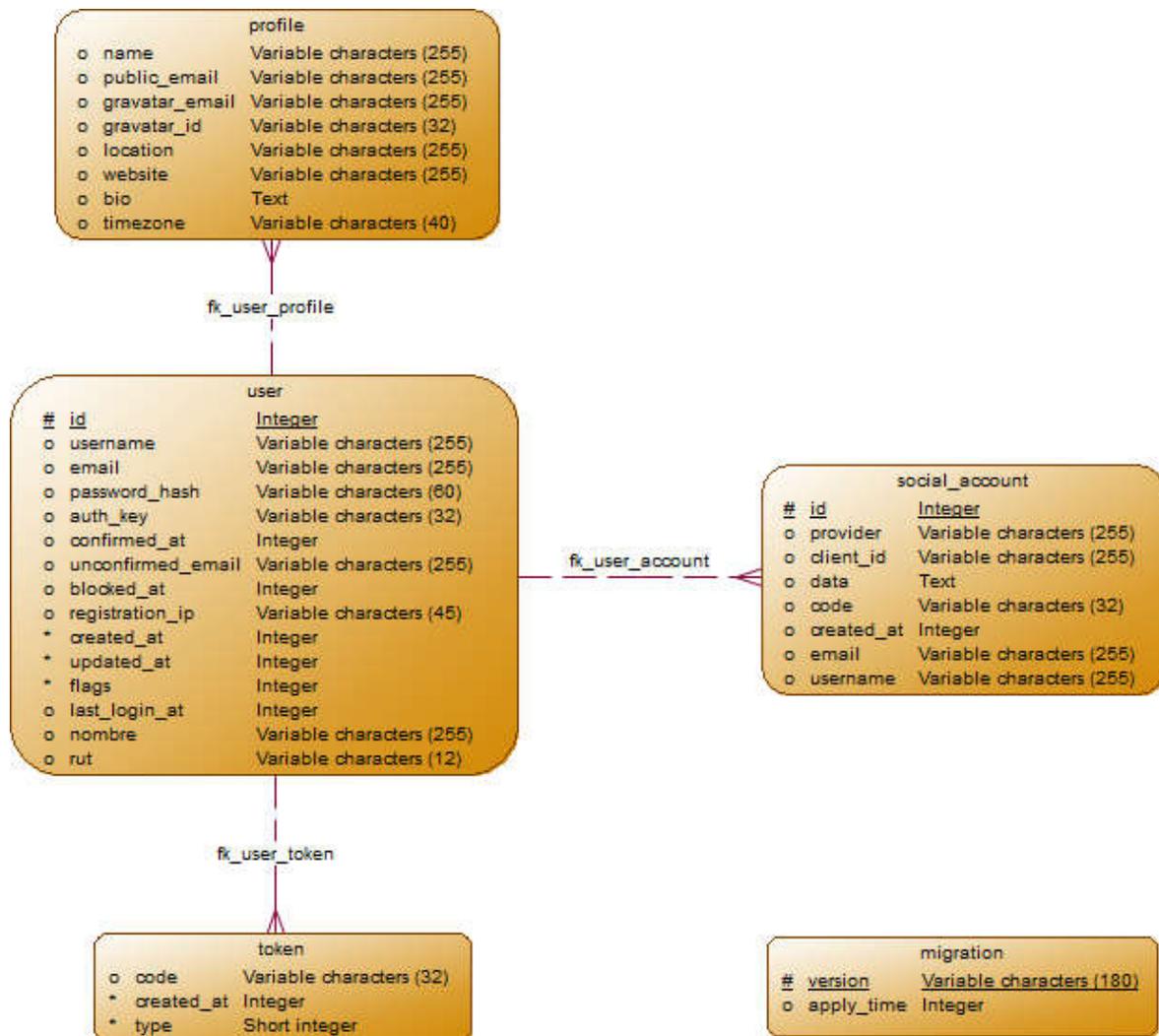
(Tabla. 7. Matriz de trazabilidad.)

7.3 Modelamiento de datos

El modelo entidad relación corresponde al modelo utilizado por el módulo Yii2-user del framework Yii2, el módulo se utiliza para la creación y control de cuentas de usuario del sistema, los datos nombre y RUT de la entidad user son usados en la cabecera del informe de evaluación en caso de ser completados.

El sistema solo utiliza las entidades user (para registrar los datos de un usuario) y token (para las confirmaciones de correo electrónico que vienen integradas con el módulo).

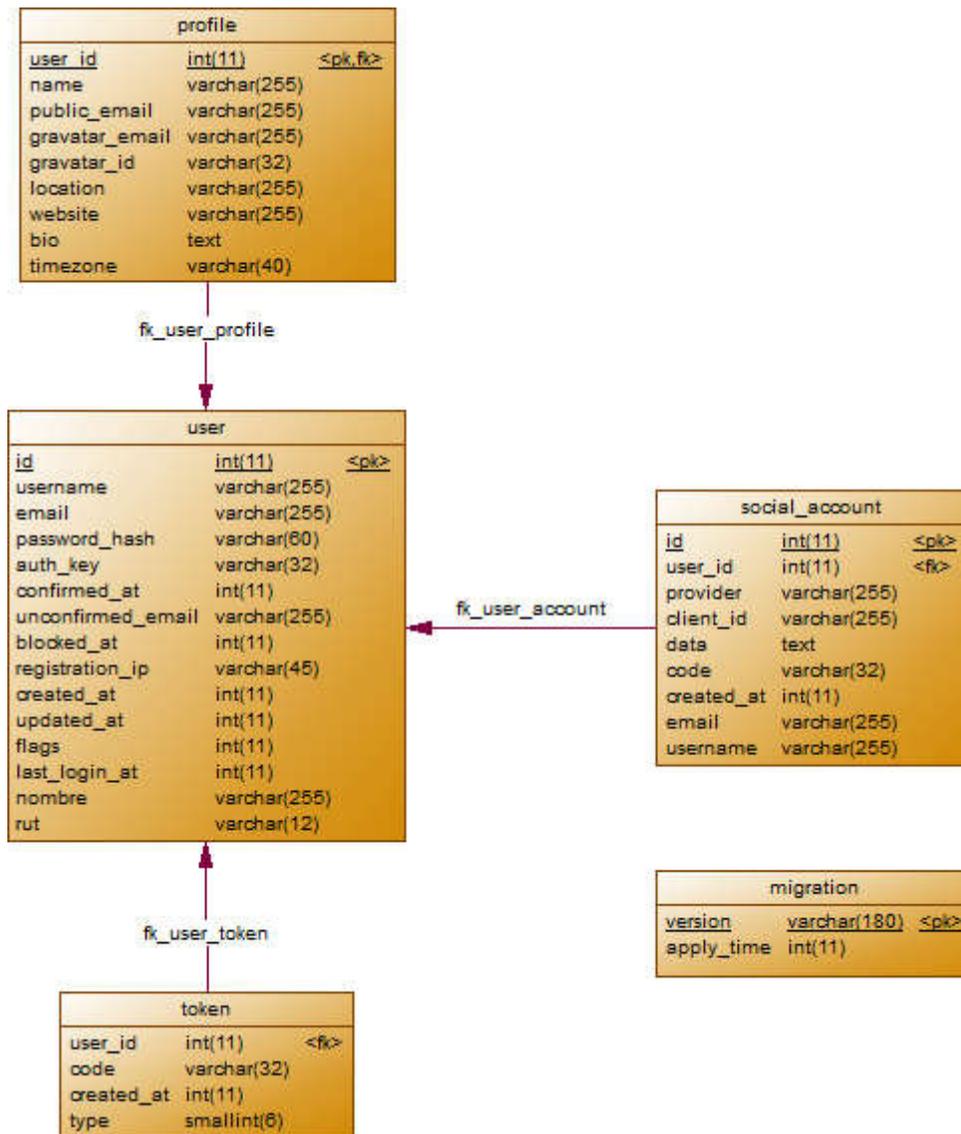
El proceso de evaluación no interactúa con la base de datos.



(Fig. 4. Modelo Entidad-Relación.)

8 DISEÑO

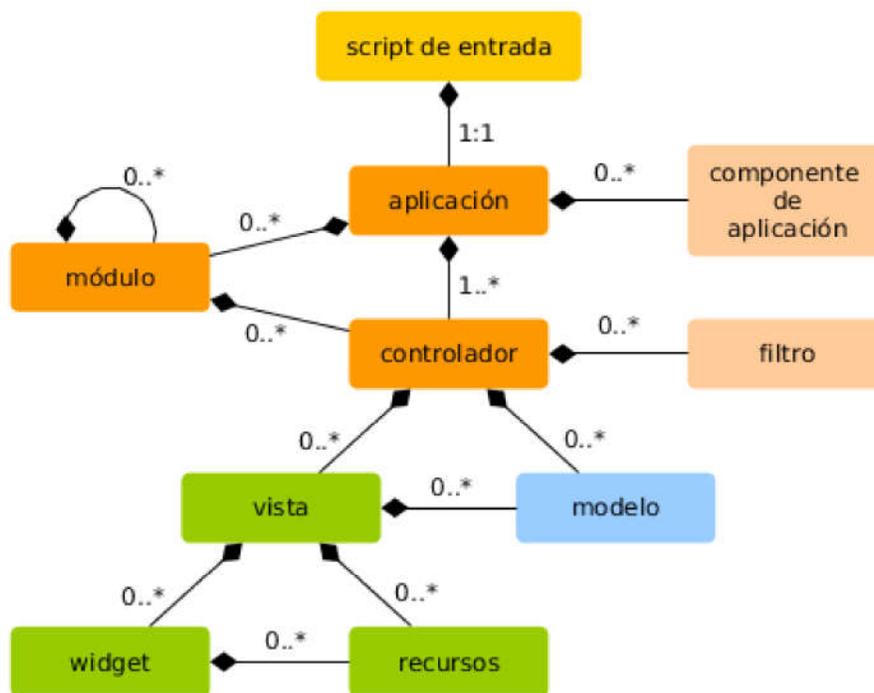
8.1 Diseño de Físico de la Base de datos



(Fig. 5. Modelo físico.)

8.2 Diseño arquitectónico

El funcionamiento completo del sistema se apoya en el framework Yii2, es por esto que la arquitectura del sistema es la correspondiente al patrón de diseño arquitectónico del framework Yii2:



(Fig. 6. patrón de diseño arquitectónico del framework Yii2.)

Como se describe en (Read the Docs Team, 2017) “las aplicaciones realizadas con Yii están organizadas de acuerdo al patrón de diseño modelo-vista-controlador (MVC). Los modelos representan datos, la lógica de negocios y sus reglas; las vistas son la representación de salida de los modelos; y finalmente, los controladores que toman datos de entrada y los convierten en instrucciones para los modelos y vistas.

Además de MVC, las aplicaciones Yii también tienen las siguientes entidades:

- **scripts de entrada:** Existen scripts PHP directamente accesibles a los usuarios finales. Son los responsables de comenzar el ciclo de manejo de una solicitud.
- **aplicaciones:** Son objetos accesibles globalmente que gestionan y coordinan los componentes de la aplicación con el fin de atender las diferentes solicitudes.

- componentes de la aplicación: Son los objetos registrados con la aplicación, y proporcionan varios servicios para cumplir las solicitudes.
- módulos: Son paquetes auto-contenidos los cuales por si solos poseen estructura MVC. Una aplicación puede estar organizada en términos de múltiples módulos.
- filtros: Representan el código que debe ser invocado antes y después de la ejecución de cada solicitud por los controladores.
- widgets: Son objetos que pueden ser embebidos en las Vistas. Pueden contener lógica del controlador y ser reutilizados en múltiples vistas.”

8.3 Diseño interfaz y navegación

Considerando que el diseño de la interfaz de usuario debe considerar un diseño estándar que será respetado en todas las pantallas. El siguiente diseño es consistente en todas las pantallas del sistema:



(Fig. 7. diseño interfaz.)

El diseño incluye un menú de navegación en la parte superior derecha de cada página (1), el contenido de cada página en el área central (2) y un pie de página (footer) que incluye el copyright del sistema (3) y el logo respectivo (4).

9 DESARROLLO

9.1 Introducción

El análisis y diseño del proyecto se basan estrictamente en los requerimientos funcionales, el sistema desarrollado al no tener muchos requerimientos o módulos, no se ve bien reflejado por estas etapas, para explicar la forma en la que se cumplieron los objetivos del proyecto es necesario explicar el contexto del proyecto, el videojuego del que se obtienen las trazas de ejecución, cómo se obtienen las trazas de ejecución. También es necesario explicar en qué consiste el proceso de evaluación, cómo se procesan los datos obtenidos de las trazas de ejecución y cómo se generan los resúmenes de los informes de evaluación.

Solo entendiendo el funcionamiento completo desde el videojuego a la creación del resumen es que se puede entender qué hace el sistema y cómo se cumplieron con su objetivo.

A continuación se describen el videojuego, la traza de ejecución y la evaluación.

9.2 Videojuego: Proyecto del primer semestre de la asignatura de Inteligencia Artificial de la carrera de Ingeniería Civil en Informática

Como se planteó en la descripción de la problemática, en la asignatura de Inteligencia Artificial de la carrera de Ingeniería Civil en Informática de la Universidad del Bío-Bío se emplea una metodología basada en proyectos. Ésta metodología se basa en el diseño e implementación de algoritmos de búsqueda en un espacio de estados con el objetivo de dotar a los agentes virtuales de movimientos autónomos e inteligentes.

Para describir el proyecto del primer semestre de la asignatura de inteligencia artificial de la carrera de ingeniería civil en informática primero deben introducirse algunos conceptos básicos como lo son:

Conceptos del videojuego:

Escenario	Normalmente representado por una matriz llena de objetos llamados celdas. Está conformado por 19 celdas de ancho y 20 celdas de alto. Representado por el cuerpo de la ventana de la aplicación.
Celdas	Las celdas son un objeto capaz de representar un elemento o un actor del videojuego. En conjunto forman una cuadrícula que representa el mundo del videojuego donde los actores interactúan. Las celdas pueden representar un jugador, un obstáculo, un camino, un adversario o una recompensa.
actor	Es una entidad que existe en el escenario del videojuego y que interactúa con los elementos del escenario y otros actores. Su interacción responde a las reglas del videojuego.
jugador	Es un actor cuyos movimientos son realizados por un algoritmo de búsqueda en un espacio de estados. Posee una cantidad de energía asociada.
obstáculo	Es un elemento representado por una celda sobre la cual un actor no se puede mover y por tanto debe rodear.
camino	Es un elemento representado por una celda sobre la cual un actor se puede mover.
adversario	Es un actor cuyos movimientos son realizados por un algoritmo de búsqueda en un espacio de estados.
recompensa	Es un elemento representado por una celda la cual es el objetivo principal de las búsquedas del jugador.
Energía	Es un número ubicado sobre el jugador cuyo valor inicial es 10. Cada 5 segundos se resta una unidad de energía, cada vez el jugador alcanza una recompensa se le suman 10 unidades de energía, si la energía es igual a 0 unidades, entonces el jugador se detiene.

(Tabla. 8. Conceptos del videojuego.)

Descripción del videojuego:

El videojuego es programado en Java y transcurre en un escenario en el que hay distribuidos muchos obstáculos y 4 recompensas, el resto de de las celdas se rellenan con caminos, estando todos los caminos conectados por al menos otro camino. Además de estos elementos también hay 4 actores, un jugador y 3 adversarios. El objetivo del jugador es capturar las 4 recompensas sin ser alcanzado por los adversarios. El jugador captura las recompensas ubicándose en la celda de la respectiva recompensa, el adversario alcanza al jugador ubicándose en la celda en la que se encuentra ubicado el jugador. Los actores solo pueden moverse a una celda contigua cada vez y hacen un movimiento cada 1 segundo. El videojuego termina cumple con su objetivo de capturar todas las recompensas o cuando el jugador es alcanzado por un adversario.

Se espera que el estudiante utilice una búsqueda informada para los movimientos del jugador y búsquedas en anchura no informada para los movimientos de los adversario.



(Fig. 8. Imagen referencial de un proyecto de Inteligencia artificial (Videojuego de Manuel Arias))

9.3 Traza de ejecución

Disponible sección 20 Anexo: Trazas de ejecución, en la página 71, al final de este documento.

9.4 Evaluación

El proceso de evaluación de trazas se basa en las principales estructuras de datos de la “Architecture for player behavior analysis and description in Computer Games” [arquitectura para el análisis y descripción del comportamiento del jugador en juegos de computadora], descrita en “Improving player experience in Computer Games by using players’ behavior analysis and linguistic descriptions” (Rubio-Manzano & Trivino, 2016).

Las estructuras descritas en el documento corresponden a Gameplay Metrics, GLMP y Report Template. Explicadas brevemente y a grandes rasgos a continuación:

Gameplay Metrics o métricas de juego se describen como métricas de juego que deben ser definidas por el diseñador por medio del análisis de las posibles medidas que describen las acciones de los jugadores durante las sesiones de juego. El objetivo es encontrar el mejor conjunto de métricas de juego para describir los eventos más relevantes ocurridos durante las sesiones de juego.

GLMP se describe como un modelo granular lingüístico de un fenómeno para el análisis y descripciones del comportamiento de los jugadores, organiza todas las percepciones relacionadas de forma similar a como los diseñadores o los jugadores organizan su experiencia en juegos de computadora por medio del lenguaje natural. En particular, el diseñador debe definir sus percepciones subjetivas como percepciones computacionales.

Report template o plantilla de informe se describe como un plantilla construida por el diseñador analizando el significado particular de cada expresión lingüística en tipos de situación específicos. Una plantilla de informe es definida considerando los requerimientos de la aplicación para entregar feedback a los jugadores sobre su conducta.

Las estructuras descritas fueron adaptadas para el proyecto y se implementaron de la siguiente forma:

Métricas de juego definidas:

Métrica	Descripción
Distancia(Jugador, Adversario)	Distancia entre el jugador y un adversario utilizando: $d = \sqrt{(x - x')^2 + (y - y')^2}$
Distancia(Jugador, Recompensa)	Distancia entre el jugador y la recompensa más cercana utilizando: $d = \sqrt{(x - x')^2 + (y - y')^2}$
Distancia(Adversario, Recompensa)	Distancia entre un adversario y la recompensa más cercana al jugador utilizando: $d = \sqrt{(x - x')^2 + (y - y')^2}$
Energía jugador	Cantidad de unidades de energía restantes del jugador.
Protección(Jugador, Adversario)	Cantidad de obstáculos entre el jugador y un adversario, considera un área rectangular formada a partir de las ubicaciones del jugador y el adversario ¹ .
Número de iteraciones	Cantidad de iteraciones realizadas por la aplicación durante la búsqueda que dio lugar al movimiento actual.
Memoria utilizada	Cantidad de memoria ocupada por los los estados de la búsqueda que dio lugar al movimiento actual.
Tiempo transcurrido	Tiempo transcurrido desde que los actores comienzan a moverse hasta el movimiento actual de un actor.

(Tabla. 10. Métricas de juego definidas.)

Las métricas utilizan los valores de las trazas de ejecución para las mediciones. Los valores obtenidos de las mediciones se traducen en términos lingüísticos que se utilizan para determinar las percepciones.

Percepciones definidas:

Actitud		
Actitud	Distancia(Jugador,Recompensa)	Distancia(Adversario,Recompensa)
Sabio	Pequeña	Normal
Valiente	Pequeña	Pequeña
Prudente	Normal	Pequeña
Pasivo	Grande	Grande

Situación			
Situación	Protección(Jugador,Adversario)	Distancia(Jugador,Adversario)	Energía jugador
Arriesgada	Media	Pequeña	Normal
Peligrosa	Baja	Pequeña	Normal
Segura	Media	Normal	Normal
Fácil	Baja	Normal	Normal
Peligrosa	Baja	Normal	Baja
Peligrosa	Media	Pequeña	Baja
Peligrosa	Media	Normal	Baja

Movimiento			
Movimiento	Distancia(Jugador,Recompensa)	Distancia(Jugador,Adversario)	Energía jugador
Bueno	Pequeña	Normal	Normal
Asustado	Normal	Normal	Normal
Kamikaze	Pequeña	Pequeña	Normal
Malo	Normal	Pequeña	Normal
Bueno	Pequeña	Pequeña	Baja

Habilidad			
Habilidad	Actitud	Movimiento	Tiempo
Experto	Sabio	Bueno	Poco
Intermedio	Valiente	Asustado	Razonable
Básico	Pasivo	Malo	Mucho
Principiante	Prudente	Asustado	Mucho

Capacidad			
Capacidad	Actitud	Movimiento	Situación
Habilidoso	Sabio	Bueno	Fácil
Poco habilidoso	Prudente	Bueno	Segura
Mejorable	Valiente	Malo	Peligrosa
Muy mejorable	Pasivo	Malo	Arriesgada

Recursos		
Recursos	Número de iteraciones	Memoria utilizada
Muy eficiente	Pocas	Baja
eficiente	Normal	Media
ineficiente	Normal	Alta
Muy ineficiente	Muchas	Alta

(Tabla. 11. Percepciones definidas: conjunto de tablas que comprende las reglas asociadas a las percepciones definidas, en cada tabla la primera fila corresponde a la percepción definida, la segunda fila divide las columnas subsiguientes en (de izquierda a derecha) los términos lingüísticos asociados a la percepción y los términos lingüísticos asociados a percepciones que la componen que deben cumplirse para que los primeros sean verdaderos)

Finalmente la plantilla de reporte se generó utilizando una gran cantidad de reglas if then las cuales actúan comparando términos lingüísticos de cantidad definidos a partir de la sumatorias de cada percepción.

Ejemplificado en esta tabla:

Término lingüístico de cantidad	Σ
Pocos	Entre cero y un tercio del total de la sumatoria.
Algunos	Entre un tercio y dos tercios del total de la sumatoria.
Muchos	Entre dos tercios y el total de la sumatoria.

(Tabla. 12. términos lingüísticos.)

17.897.862-4 - Tomás Lermanda

RESUMEN:

La inteligencia artificial ha demostrado una actitud principalmente sabia durante buena parte de la partida, aunque algunas veces fue valiente. Las situaciones experimentadas fueron arriesgadas en gran medida. La inteligencia artificial demostró ser capaz de realizar algunos movimientos kamikaze. El nivel de habilidad exhibido durante la partida fue principiante. La inteligencia artificial demostró ser poco habilidosa. El uso de recursos medido demuestra un funcionamiento que muchas veces es muy eficiente.

(Fig. 9. Imágen referencial de un resumen generado a partir de una traza de ejecución.)

10 PRUEBAS

10.1 Elementos de prueba

Los elementos de prueba seleccionados son:

Registrar usuario: Componente que incluye los campos de registro de usuarios y validadores para sus campos.

Evaluar traza de ejecución: Componente que incluye el campo de selección de una traza de ejecución y los validadores asociados a esta.

10.2 Especificación de las pruebas

Características a probar	Nivel de prueba	Objetivo de la Prueba	Enfoque para la definición de casos de prueba	Técnicas para la definición de casos de prueba	Actividades de prueba	Criterios de cumplimiento
Funcionalidad	Unitaria	Comprobar el correcto funcionamiento del registro	Caja negra	Particiones	El tester es un usuario no registrado en la página de registro	Comprobar que todos los casos de prueba resulten en su respectiva salida esperada
Funcionalidad	Unitaria	Comprobar la correcta evaluación de trazas de ejecución	Caja negra	Particiones	El tester es un usuario registrado que ha iniciado sesión y está en la página principal	Comprobar que todos los casos de prueba resulten en su respectiva salida esperada

(Tabla. 13. Especificación de las pruebas.)

10.3 Responsables de las pruebas

El encargado y responsable de realizar las distintas pruebas en el sistema de forma correcta es el desarrollador.

10.4 Calendario de pruebas



(Fig. 10. Carta Gantt calendario de pruebas.)

10.5 Detalle de las pruebas

10.5.1 PU-01 Registrar usuario

En este ítem se especifican:

- Configuración: La prueba no necesita ningún tipo de configuración de hardware, software, sistema operativo, o de comunicaciones en particular.
- Pre-condiciones: El usuario no ha registrado sus datos con anterioridad.

ID Caso de prueba	Características a probar	Datos de entrada					Salida esperada	Salida obtenida	éxito /fracaso	Observaciones
		Correo electrónico	Nombre de usuario	Nombre Completo	RUT	Contraseña				
CPU-01	Funcionalidad	abc@abc.a bc	nomb re	nombre c	12.312. 312-3	qwerty	Tu cuenta ha sido creada	Tu cuenta ha sido creada	éxito	sin observaciones
CPU-02	Funcionalidad	abc@abc	nomb re	nombre c	12.312. 312-3	qwerty	Correo electrónico no es una dirección de correo válida.	Correo electrónico no es una dirección de correo válida.	éxito	sin observaciones

CPU-03	Funcionalidad	abc@abc.a bc	nombre	nombre 123	12.312. 312-3	qwerty	Los nombres sólo pueden contener letras.	Los nombres sólo pueden contener letras.	éxito	sin observaciones
CPU-04	Funcionalidad	abc@abc.a bc	nombre	n@mbre	12.312. 312-3	qwerty	Los nombres sólo pueden contener letras.	Los nombres sólo pueden contener letras.	éxito	sin observaciones
CPU-05	Funcionalidad	abc@abc.a bc	nombre	nombre c	12.312. 312-2	qwerty	RUT incorrecto.	RUT incorrecto.	éxito	sin observaciones
CPU-06	Funcionalidad	abc@abc.a bc	nombre	nombre c	12.312. 312-a	qwerty	Contiene letras.	Contiene letras.	éxito	sin observaciones
CPU-07	Funcionalidad	abc@abc.a bc	nombre	nombre c	12.311. 312-3	qwerty	Debe escribirse de la forma XX.XXX.XXX-X	Debe escribirse de la forma XX.XXX.XXX-X	éxito	sin observaciones
CPU-08	Funcionalidad	abc@abc.a bc	nombre	nombre c	12.312. 312-31	qwerty	RUT debería contener como máximo 12 letras.	RUT debería contener como máximo 12 letras.	éxito	sin observaciones
CPU-09	Funcionalidad	abc@abc.a bc	nombre	nombre c	12.312. 32-3	qwerty	Debe escribirse de la forma XX.XXX.XXX-X	Debe escribirse de la forma XX.XXX.XXX-X	éxito	sin observaciones
CPU-10	Funcionalidad	abc@abc.a bc	nombre	nombre c	12.312. 312-3	qwerty	Contraseña debería contener al menos 6 letras.	Contraseña debería contener al menos 6 letras.	éxito	sin observaciones

(Tabla. 14. Detalle de las pruebas (registrar usuario).)

10.5.2 PU-02 Evaluar traza de ejecución

En este ítem se especifican:

- Configuración: La prueba no necesita ningún tipo de configuración de hardware, software, sistema operativo, o de comunicaciones en particular.
- Pre-condiciones: El usuario registrado ha iniciado sesión en el sistema.

ID Caso de prueba	Características a probar	Datos de entrada	Salida esperada	Salida obtenida	éxito/fracaso	Observaciones
		Archivo *.txt				
CPU-11	Funcionalidad	(archivo *.txt formateado correctamente como traza de ejecución)	El archivo es subido, evaluado y se genera un informe	El archivo es subido, evaluado y se genera un informe	éxito	sin observaciones
CPU-12	Funcionalidad	(archivo *.txt no formateado correctamente como traza de ejecución)	El contenido de la traza de ejecución no se encuentra en el formato correcto.	El contenido de la traza de ejecución no se encuentra en el formato correcto.	éxito	sin observaciones
CPU-13	Funcionalidad	(archivo de otro tipo renombrado con la extensión *.txt)	ERROR al subir el archivo.	ERROR al subir el archivo.	éxito	sin observaciones
CPU-14	Funcionalidad	(archivo con una extensión distinta a *.txt)	Sólo se aceptan archivos con las siguientes extensiones: txt	Sólo se aceptan archivos con las siguientes extensiones: txt	éxito	sin observaciones

(Tabla. 15. Detalle de las pruebas (evaluar traza de ejecución).)

10.6 Conclusiones de Prueba

El proceso de pruebas se hizo sobre las partes en las que se codificó, las pruebas realizadas se concentran en evaluar la correcta funcionalidad del sistema en los casos considerados más probables, de forma en que se asegure que los mensajes de error sean claros en las distintas situaciones y así evitar que un usuario se encuentre con un error que no entienda y no sepa qué hacer.

El sistema superó exitosamente todos los casos de prueba planificados.

11 PLAN DE CAPACITACIÓN Y ENTRENAMIENTO

El único usuario al momento de la implantación y puesta en marcha es el profesor Clemente Rubio, puesto que la asignatura de Inteligencia artificial del primer semestre de 2017 ya ha concluido y el profesor Clemente no la volverá a impartir hasta el próximo año.

Dada la simpleza del sistema, comparada con la del buscador google durante su diseño de interfaz, bastó con tan solo unas instrucciones simples de acceso enviadas por correo electrónico por parte del desarrollador para que el sistema fuera utilizado completamente. Puesto que se siguieron los requerimientos detalladamente no hubieron dudas sobre el funcionamiento.

Calendario:



(Fig. 11. Carta Gantt calendario de entrenamiento.)

Hay preparado un manual en caso de nuevos usuarios. El manual se encuentra disponible en la sección 22 Anexo: Manual de usuario, en la página 75, al final de este documento.

12 PLAN DE IMPLANTACIÓN Y PUESTA EN MARCHA

Puesto que no se desarrollaron múltiples módulos o funcionalidades se fijó la implantación y puesta en marcha en un mismo día, en el cual se instala la funcionalidad completa con la excepción de la plantilla del resumen de la evaluación la cual se integra en una fecha posterior.

Calendario:



(Fig. 12. Carta Gantt calendario de implantación y puesta en marcha.)

Previo al proceso de implantación en el entorno de producción (servidor del sistema), el sistema fue probado en los entornos de desarrollo y pruebas, siempre existiendo respaldos de todo lo desarrollado.

13 RESUMEN ESFUERZO REQUERIDO

Actividades/fases	N° Horas
Elaboración del estado del arte	24
Estudiar la naturaleza y estructura del feedback	60
Análisis	60
Diseño	60
Implementación	120
Pruebas	40
Elaboración de resultados y conclusiones	30
Documentación	126
TOTAL	520 aprox.

(Tabla. 16. Tabla de resumen de esfuerzo requerido.)

La estimación inicial de tamaño del proyecto se basó en el método de puntos de caso de uso, el cual resultó no ser un método efectivo para esta tarea, dado que la dificultad no estaba en la cantidad de casos de uso o el nivel de conocimientos del lenguaje de programación en particular, además la estimación inicial se basó en valores estándar que no reflejan de la mejor manera el esfuerzo requerido.

14 RESULTADOS

Durante este proyecto se desarrolló un sistema capaz de generar automáticamente informes en lenguaje natural a partir de trazas de ejecución de los proyectos del primer semestre de la asignatura de Inteligencia Artificial de la Carrera de Ingeniería Civil en Informática de la Universidad del Bío-Bío. Para propósitos de probar el correcto funcionamiento sistema más allá de las pruebas unitarias, se utilizó una muestra de trazas de ejecución de diez proyectos de estudiantes del primer semestre. Estas pruebas fueron de carácter subjetivo y se utilizaron para ajustar la correcta representación de las percepciones subjetivas del desarrollador. Durante las pruebas se estableció en base al juicio del desarrollador que las percepciones se verían mejor representadas si el proyecto de videojuego tiene un jugador más veloz que sus adversarios puesto que de otra forma las diferencias entre los proyectos aunque existentes y capturadas, no se harían muy notorias y requerirían un mejor entendimiento del proyecto. Esta conclusión se basó en la observación, a partir de las aparentes tendencias observadas. También es importante recalcar que las pruebas se hicieron sobre una muestra limitada y que para comprobar indiscutiblemente los resultados se requiere de una muestra más grande y pruebas especialmente elaboradas para el sistema, lo que estaba fuera de los objetivos y alcances de este proyecto.

Los diez resúmenes obtenido a partir de las diez muestras se pueden encontrar en la sección 19 Anexo: Muestra de resúmenes, en la página 68, al final de este documento.

15 CONCLUSIONES

Durante este proyecto se desarrolló, un sistema de generación automática de informes en lenguaje natural, capaz de evaluar trazas de ejecución de los proyectos del primer semestre de 2017 de la asignatura de Inteligencia Artificial de la carrera Ingeniería Civil en Informática de la Universidad del Bío-Bío. Las evaluaciones consideran métricas de calidad y cantidad por medio del uso de percepciones que consideran las particularidades de los proyectos del primer semestre de 2017.

El sistema implementado, es una herramienta capaz de evaluar trazas de ejecución en instantes, permite al profesor o los estudiantes obtener informes que evalúan el desempeño de un proyecto rápidamente, sin necesidad de visualizar el comportamiento de los agentes virtuales, observar el código o mirar los datos de la ejecución. A partir del informe se puede determinar el comportamiento de los agentes y la eficiencia de las búsquedas. Además se distinguen claramente las búsquedas informadas de las no informadas.

Si bien este proyecto apuntaba al desarrollo de un sistema de generación automática de informes en lenguaje natural de proyectos de la asignatura Inteligencia Artificial en general, el sistema creado solo funciona correctamente con los proyectos del primer semestre de 2017, limitando su uso posterior a que el proyecto del siguiente semestre sea idéntico o se realicen cambios al código de la evaluación que permitan implementar nuevas percepciones que se adapten a un nuevo proyecto.

Un trabajo futuro implicaría resolver el problema de la escalabilidad para permitir el uso del sistema en múltiples proyectos diferentes, ya sea ampliando el sistema o rediseñando el funcionamiento de la evaluación de forma que fácil de modificar y adaptar. Resolviendo este problema la utilidad y vida útil del sistema se podría extender indefinidamente.

Otro trabajo futuro podría ser la integración de un certificado SSL al sitio web del sistema para que los datos de registro e inicio de sesión estén encriptados y no sean vulnerables, puesto que esta medida de seguridad iba más allá de los alcances del proyecto y los datos no son seguros antes de llegar al sistema.

16 BIBLIOGRAFÍA

Universidad del Bío-Bío. 2017. "Visión y Misión". Ubiobio.cl.
<http://www.ubiobio.cl/w/m.php?id=27> (visitada Julio 24, 2017).

Universidad del Bío-Bío. 2017. "Organigrama". Ubiobio.cl.
http://ubiobio.cl/miweb/web2012.php?id_pagina=5152 (visitada Julio 24, 2017).

Departamento de Sistemas de Información. 2017. "Visión y Misión". Dsi.face.cl.
http://dsi.face.ubiobio.cl/?page_id=40 (visitada Julio 24, 2017).

Departamento de Sistemas de Información. 2017. "Estructura". Dsi.face.cl.
<http://webface.ubiobio.cl/wp-content/uploads/2015/11/organigrama-1024x440.jpg>
(visitada Julio 24, 2017).

Read the Docs Team. "Información general". Yii2-framework.readthedocs.io.
<https://yii2-framework.readthedocs.io/en/stable/guide-es/structure-overview/>
(visitada Julio 24, 2017).

PHP Documentation Group. 2017. "Uso básico". php.net.
<http://php.net/manual/es/session.examples.basic.php> (visitada Julio 25, 2017).

Real Academia Española. 2017a. "informe". dle.rae.es.
<http://dle.rae.es/?id=LYB2BS5|LYF57Ax> (visitada Septiembre 21, 2017).

Real Academia Española. 2017b. "evaluar". dle.rae.es. <http://dle.rae.es/?id=H8KIdC6>
(visitada Septiembre 21, 2017).

Collins Dictionary. 2017. "natural language". collinsdictionary.com. <https://www.collinsdictionary.com/dictionary/english/natural-language> (visitada Septiembre 21, 2017).

University of Miami's Department of Computer Science. 2017. "State Spaces". cs.miami.edu. <http://www.cs.miami.edu/home/geoff/Courses/COMP6210-10M/Content/StateSpaceSearch.shtml> (visitada Septiembre 21, 2017).

Buck Institute for Education. 2017. "What is Project Based Learning (PBL)". bie.org. http://www.bie.org/about/what_pbl (visitada Septiembre 21, 2017).

Sánchez-Torrubia, M. G., Torres-Blanc C. & Trivino G. 2012. An approach to automatic learning assessment based on the computational theory of perceptions. *Expert Systems with Applications* Vol. 39, Issue 15, 12177–12191p.

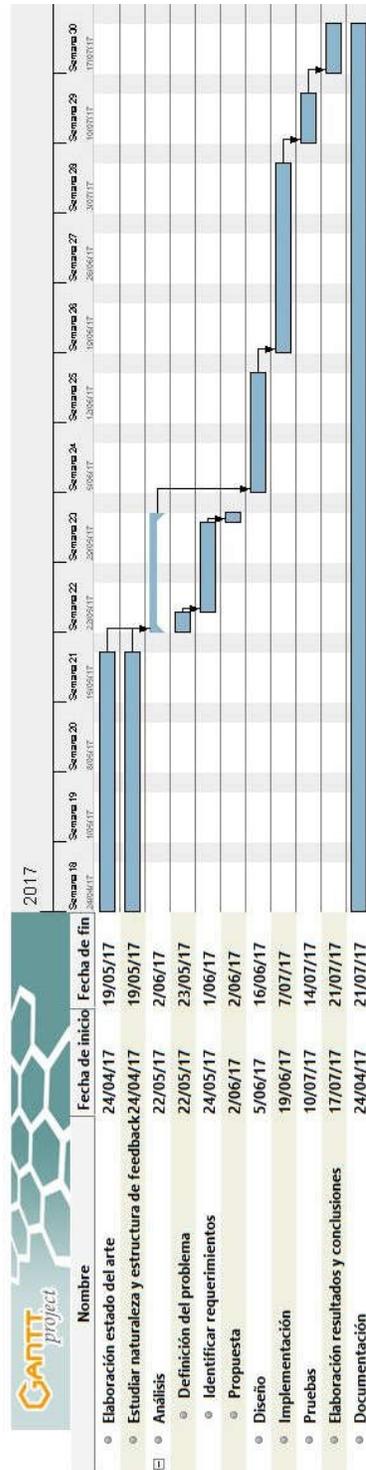
Gkatzia, D., Hastie, H., Janarthnam, S., & Lemon, O. 2013. Generating student feedback from time-series data using reinforcement learning. *Proceedings of the 14th European Workshop on Natural Language Generation*. 115-124p., 14th European Workshop On Natural Language Generation, Sofia, Bulgaria, 8-9 August.

Ramos-Soto, A., Lama, M., Vazquez-Barreiros, B., Bugarín, A., Mucientes, M., & Barro, S. 2015. Towards Textual Reporting in Learning Analytics Dashboards. *IEEE Computer Society Conference Publications*. 260-264p., 15th IEEE International Conference on Advanced Learning Technologies, Hualien, Taiwan, 6-9 July.

Kraft J., Wall A., & Kienle H. 2010. Trace Recording for Embedded Systems: Lessons Learned from Five Industrial Projects. *Runtime Verification* 315-329p., *Runtime Verification: First International Conference, RV 2010, St. Julians, Malta, 1-4 November*.

Rubio-Manzano, C. y Trivino, G. 2016. Improving player experience in Computer Games by using players' behavior analysis and linguistic descriptions. *International Journal of Human-Computer Studies* Vol. 95, Issue C, 27–38p.

17 ANEXO: PLANIFICACIÓN INICIAL DEL PROYECTO



(Fig. 13. Carta Gantt planificación inicial de proyecto.)

17.1.1 Estimación inicial de tamaño

Estimación puntos casos de uso:

FACTOR DE PESO DE LOS ACTORES SIN AJUSTAR				
Tipo de Actor	Descripción	Factor de Peso	Nro Actores	Subtotal (UAW)
Simple	Otro Sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API, Application Programming Interface)	1	0	0
Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto	2	0	0
Complejo	Una persona que interactúa con el sistema mediante una interfaz	3	1	3
Total Actores (UAW)			1	3
FACTOR DE PESOS DE CASO DE USO SIN AJUSTAR				
Tipo de Caso de Uso	Descripción	Factor de Peso	Nro Casos de Uso	Subtotal (UUCW)
Simple	El caso de uso contiene de 1 a 3 transacciones	5	3	15
Medio	El caso de uso contiene de 4 a 7 transacciones	10	0	0
Complejo	El caso de uso contiene más 8 transacciones	15	0	0
Total Use Cases (UUCW)			3	15
Puntos de Caso de Uso Sin Ajustar (UUCP)				18
FACTORES DE COMPLEJIDAD TÉCNICA (TCF)				
Factor	Descripción	Factor de Peso	Valor Asignado (0-5)	Subtotal
T1	Sistema distribuido	2	0	0
T2	Objetivos de performance o tiempo de respuesta	2	5	10
T3	Eficiencia del usuario	1	3	3
T4	Procesamiento interno complejo	1	1	1
T5	El código debe ser reutilizable	1	3	3
T6	Facilidad de instalación	0.5	2	1
T7	Facilidad de uso	0.5	5	2.5
T8	Portabilidad	2	3	6
T9	Facilidad de cambio	1	3	3
T10	Concurrencia	1	0	0
T11	Incluye objetivos especiales de seguridad	1	4	4
T12	Provee accesos directos a terceras partes	1	4	4
T13	Se requieren facilidades especiales de entrenamiento a usuarios	1	1	1
Total Factor				38.5

FACTORES DE AMBIENTE (EF)				
Factor	Descripción	Factor de Peso	Valor Asignado	Subtotal (EF)
E1	Familiaridad con el modelo del proyecto Utilizado (RUP, MSF, Métrica3)	1.5	3	4.5
E2	Experiencia en la aplicación (web, standalone, EAI, SOA, BPM, SOA-BPM)	0.5	3	1.5
E3	Experiencia en orientación a objetos (Proc, OO, Aspect)	1	3	3
E4	Lead analyst capability	0.5	3	1.5
E5	Motivación	1	5	5
E6	Estabilidad de los requerimientos	2	4	8
E7	Personal Part-time	-1	0	0
E8	Dificultad del lenguaje de programación (Java, C#, VB, SQL) <w -1,2>	-1	3	-3
Total Factor Ambiente (EF)				20.5
ESTIMANDO LOS UCPs				
Param	Description	Calculated Value		
UUCP	Unadjusted UCPs	18		
TCF	Technical Complexity Factor	0.985		
ECF o EF	Environmental Complexity Factor	0.785		
UCP	Total Adjusted UCPs	13.92		
Capacity Factor (min 10, avg 20, max 40) hours/UCP		10		

(Tabla. 17. Estimación inicial de puntos casos de uso.)

Cálculo Horas Hombre:

Esfuerzo Horas Hombre = Puntos de Casos de Uso ajustados * Horas-Persona

$$278,4 = 13,92 * 20$$

Cálculo costo total:

Costo total = Horas Hombre * Valor Hora Hombre

$$83,52 \text{ UF} = 278,4 \text{ HH} * 0,3 \text{ UF}$$

18 ANEXO: DICCIONARIO DE DATOS DEL MODELO DE DATOS

Tablas usadas por el sistema en el entorno de producción:

user			
Atributo	Tipo de dato	Tamaño	Descripción
id	integer	11	Corresponde al identificador único del usuario.
username	varchar	255	Corresponde al nombre de usuario único del usuario para iniciar sesión.
email	varchar	255	Corresponde al correo electrónico único del usuario.
password_hash	varchar	60	Corresponde al hash de la contraseña del usuario.
auth_key	varchar	32	Corresponde a la clave de autenticación de la cookie usuario.
confirmed_at	integer	11	Corresponde a la fecha de confirmación del correo electrónico del usuario.
unconfirmed_email	varchar	255	Corresponde a la dirección de correo electrónico no confirmada del usuario.
blocked_at	integer	11	Corresponde a la fecha en la que se bloqueo al usuario.
registration_ip	varchar	45	Corresponde a la IP con la cual se registro el usuario.
created_at	integer	11	Corresponde a la fecha en la que se creó al usuario.
updated_at	integer	11	Corresponde a la última fecha en la que se actualizo el usuario.
flags	integer	11	Corresponde a la fecha de validación de token de confirmación de correo electrónico.
last_login_at	integer	11	Corresponde a la última fecha de de inicio de sesión de usuario.
nombre	varchar	255	Corresponde al nombre completo del usuario.
rut	varchar	12	Corresponde al único RUT del usuario.

token			
Atributo	Tipo de dato	Tamaño	Descripción
user_id	integer	11	Corresponde al identificador único del usuario asociado al token.
code	varchar	32	Corresponde al código de autorización del token.
created_at	integer	11	Corresponde a la fecha de creación del token.
type	smallint	6	Corresponde al tipo de token según su uso.

Tablas usadas por el sistema en el entorno de desarrollo:

migration			
Atributo	Tipo de dato	Tamaño	Descripción
version	varchar	180	Corresponde al descriptor del cambio de versión ocurrido con una migración.
apply_time	integer	11	Corresponde a la fecha en la que se realizó la migración.

Tablas no usadas por el sistema en el entorno de producción, pero existentes para el correcto funcionamiento del módulo Yii2-User del framework Yii2:

profile			
Atributo	Tipo de dato	Tamaño	Descripción
user_id	integer	11	Corresponde al identificador único del usuario asociado al perfil.
name	varchar	255	Corresponde al nombre público del usuario asociado al perfil.
public_email	varchar	255	Corresponde al correo electrónico público del usuario asociado al perfil.
gravatar_email	varchar	255	Corresponde al correo electrónico de gravatar del usuario asociado al perfil.
gravatar_id	varchar	32	Corresponde al identificador de gravatar del usuario asociado al perfil.
location	varchar	255	Corresponde a la localización pública del usuario asociado al perfil.
website	varchar	255	Corresponde a la página web pública del usuario asociado al perfil.
bio	text		Corresponde a la biografía pública del usuario asociado al perfil.
timezone	varchar	255	Corresponde a la zona horaria pública del usuario asociado al perfil.

social_account			
Atributo	Tipo de dato	Tamaño	Descripción
id	integer	11	Corresponde al identificador único de la cuenta de red social.
user_id	integer	11	Corresponde al identificador único del usuario asociado a la cuenta de red social.
provider	varchar	255	Corresponde al proveedor asociado a la cuenta de red social.
client_id	varchar	255	Corresponde al identificador único del cliente con el que se conecta a la cuenta de red social.
data	text		Corresponde a los datos descriptivos de la cuenta de red social.
code	varchar	32	Corresponde al código de autorización de la cuenta de red social.
created_at	integer	11	Corresponde a la fecha de creación de la cuenta de red social.
email	varchar	255	Corresponde al correo electrónico asociado a la cuenta de red social.
username	varchar	255	Corresponde al nombre de usuario de la cuenta de red social.

(Tabla. 18. Tablas diccionario de datos.)

19 ANEXO: MUESTRA DE RESÚMENES

Fernando Béjar:

RESUMEN:

La inteligencia artificial ha demostrado una actitud principalmente valiente durante buena parte de la partida. Las situaciones durante la partida fueron mayoritariamente peligrosas. La inteligencia artificial demostró muchos movimientos kamikaze. El nivel de habilidad exhibido durante la mayor parte de la partida fue tajantemente principiante. La inteligencia artificial demostró ser claramente mejorable. El uso de recursos medido demuestra un funcionamiento que algunas veces es muy eficiente.

Javiera Rain:

RESUMEN:

La inteligencia artificial ha demostrado una actitud principalmente sabia durante buena parte de la partida. Las situaciones durante la partida fueron arriesgadas en gran medida, aunque algunas veces fueron peligrosas. La inteligencia artificial demostró muchos movimientos kamikaze. El nivel de habilidad exhibido durante la mayor parte de la partida fue tajantemente principiante. La inteligencia artificial demostró ser poco habilidosa. El uso de recursos medido demuestra un funcionamiento que muchas veces es muy eficiente.

Hector Riquelme:

RESUMEN:

La inteligencia artificial ha demostrado una actitud principalmente valiente durante buena parte de la partida. Las situaciones durante la partida fueron mayoritariamente peligrosas. La inteligencia artificial demostró muchos movimientos kamikaze. El nivel de habilidad exhibido durante la mayor parte de la partida fue tajantemente principiante. La inteligencia artificial demostró ser mejorable. El uso de recursos medido demuestra un funcionamiento que algunas veces es muy ineficiente.

Jordy Romero:

RESUMEN:

La inteligencia artificial ha demostrado una actitud principalmente sabia durante buena parte de la partida. Las situaciones experimentadas fueron arriesgadas en gran medida. La inteligencia artificial demostró ser capaz de realizar algunos movimientos buenos, pero también movimientos kamikaze. El nivel de habilidad exhibido durante buena parte de la partida fue experto, sin embargo, también fue principiante. La inteligencia artificial demostró ser habilidosa. El uso de recursos medido demuestra un funcionamiento que algunas veces es muy eficiente.

Jose Navarro:

RESUMEN:

La inteligencia artificial ha demostrado una actitud principalmente valiente durante buena parte de la partida. Las situaciones durante la partida fueron mayoritariamente peligrosas. La inteligencia artificial demostró muchos movimientos kamikaze. El nivel de habilidad exhibido durante la mayor parte de la partida fue tajantemente principiante. La inteligencia artificial demostró ser mejorable. El uso de recursos medido demuestra un funcionamiento que algunas veces es muy eficiente y otras veces solo eficiente.

Manuel Arias:

RESUMEN:

La inteligencia artificial ha demostrado una actitud principalmente valiente durante buena parte de la partida. Las situaciones durante la partida fueron mayoritariamente peligrosas. La inteligencia artificial demostró ser capaz de realizar algunos movimientos kamikaze. El nivel de habilidad exhibido durante la partida fue principiante. La inteligencia artificial demostró ser claramente mejorable. El uso de recursos medido demuestra un funcionamiento que muchas veces es eficiente.

Omar Vega:

RESUMEN:

La inteligencia artificial ha demostrado una actitud principalmente valiente durante buena parte de la partida. Las situaciones experimentadas fueron peligrosas en gran medida. La inteligencia artificial demostró ser capaz de realizar algunos movimientos kamikaze. El nivel de habilidad exhibido durante buena parte de la partida fue experto, sin embargo, también fue principiante. La inteligencia artificial demostró ser mejorable. El uso de recursos medido demuestra un funcionamiento que muchas veces es muy ineficiente.

Pablo Soto:

RESUMEN:

La inteligencia artificial ha demostrado una actitud principalmente valiente durante buena parte de la partida. Las situaciones experimentadas fueron peligrosas en gran medida. La inteligencia artificial demostró ser capaz de realizar algunos movimientos kamikaze. El nivel de habilidad exhibido durante la partida fue principiante. La inteligencia artificial demostró ser poco habilidosa. El uso de recursos medido demuestra un funcionamiento que algunas veces es muy ineficiente.

Rodrigo Salas:

RESUMEN:

La inteligencia artificial ha demostrado una actitud principalmente valiente durante buena parte de la partida. Las situaciones experimentadas fueron peligrosas en gran medida. La inteligencia artificial demostró muchos movimientos kamikaze. El nivel de habilidad exhibido durante la mayor parte de la partida fue tajantemente principiante. La inteligencia artificial demostró ser poco habilidosa, no obstante, también demostró ser mejorable. El uso de recursos medido demuestra un funcionamiento que muchas veces es muy eficiente.

Víctor Cáceres:

RESUMEN:

La inteligencia artificial ha demostrado una actitud principalmente sabia durante buena parte de la partida, aunque algunas veces fue valiente. Las situaciones experimentadas fueron arriesgadas en gran medida. La inteligencia artificial demostró ser capaz de realizar algunos movimientos kamikaze. El nivel de habilidad exhibido durante la partida fue principiante. La inteligencia artificial demostró ser poco habilidosa. El uso de recursos medido demuestra un funcionamiento que muchas veces es muy eficiente.

(Fig. 14. Muestra de resúmenes.)

20 ANEXO: TRAZAS DE EJECUCIÓN

Para obtener la información requerida para evaluar el desempeño de un proyecto de inteligencia artificial se utilizó un archivo de traza de ejecución, que se obtiene a partir del registro de trazas. Se entiende por registro de trazas, lo afirmado en Kraft, Wall, & Kienle (2010):

“Trace recording, or tracing, is a commonly used technique useful in debugging and performance analysis. Concretely, trace recording implies detection and storage of relevant events during run-time, for later off-line analysis”. [El registro de trazas, o trazar, es una técnica de uso común útil en la depuración y análisis de rendimiento. Concretamente, el registro de trazas implica la detección y almacenamiento de eventos relevantes durante el tiempo de ejecución, para un análisis posterior fuera de línea.]

Aplicado al videojuego, es un archivo txt con datos formateados en un determinado orden, estos datos nos permiten trazar el comportamiento de los actores del videojuego desde que comienza su ejecución hasta que termina. Los datos de la traza de ejecución se obtienen a partir de cada movimiento que realiza un actor, en otras palabras, los datos asociados al conjunto de movimientos realizados por los actores de principio a fin de una ejecución son los datos de una traza de ejecución.

La organización de los datos dentro archivo de traza de ejecución se definió como las variables capturadas separadas por una coma y un espacio.

Las variables de la traza de ejecución son las siguientes:

Variable	Descripción
jugadorx	Posición x del jugador, en un plano formado por las celdas del escenario.
jugadory	Posición y del jugador, en un plano formado por las celdas del escenario.
adversario1x	Posición x del adversario1, en un plano formado por las celdas del escenario.
adversario1y	Posición y del adversario1, en un plano formado por las celdas del escenario.
adversario2x	Posición x del adversario2, en un plano formado por las celdas del escenario.

adversario2y	Posición y del adversario2, en un plano formado por las celdas del escenario.
adversario3x	Posición x del adversario3, en un plano formado por las celdas del escenario.
adversario3y	Posición y del adversario3, en un plano formado por las celdas del escenario.
energíaj	Cantidad de unidades de energía restantes del jugador
proteccionA1	Cantidad de obstáculos entre el jugador y el adversario, considera un área rectangular formada a partir de las ubicaciones del jugador y el adversario1.
proteccionA2	Cantidad de obstáculos entre el jugador y el adversario, considera un área rectangular formada a partir de las ubicaciones del jugador y el adversario2.
proteccionA3	Cantidad de obstáculos entre el jugador y el adversario, considera un área rectangular formada a partir de las ubicaciones del jugador y el adversario3.
numMovj	Distancia entre el jugador y la recompensa más cercana utilizando: $d = \sqrt{(x - x')^2 + (y - y')^2}$
numMova1	Distancia entre el adversario1 y la recompensa más cercana al jugador utilizando: $d = \sqrt{(x - x')^2 + (y - y')^2}$
numMova2	Distancia entre el adversario2 y la recompensa más cercana al jugador utilizando: $d = \sqrt{(x - x')^2 + (y - y')^2}$
numMova3	Distancia entre el adversario3 y la recompensa más cercana al jugador utilizando: $d = \sqrt{(x - x')^2 + (y - y')^2}$
tiempo	Tiempo transcurrido desde que los actores comienzan a moverse hasta el movimiento actual de un actor.
recompensa	Verdadero o falso dependiendo de si el jugador captura una recompensa en el movimiento actual.
iteraciones	Cantidad de iteraciones realizadas por la aplicación durante la búsqueda que dio lugar al movimiento actual.
memoriae	Cantidad de memoria ocupada por los los estados de la búsqueda que dio lugar al movimiento actual.
agente	J o A dependiendo de si el movimiento actual lo realiza un jugador o un adversario.

(Tabla. 9. Variables de la traza de ejecución.)

Con motivo de generar la traza de ejecución, se programaron dos clases en Java que serían adaptables a los proyectos de los estudiantes.

Estas clases tienen dos funciones, llevar los datos capturados a una tabla, y exportar los datos capturados a un archivo txt.

Captura de datos:

The image shows a spreadsheet with a grid of data. The columns are labeled with letters from A to Z, and the rows are numbered from 1 to 26. The data appears to be organized into several columns, with some cells containing numerical values and others containing text or symbols. The spreadsheet is displayed in a window titled 'Data'.

(Fig. 15. Imagen referencial de una tabla añadida a un de proyecto (Videojuego de Javierra Rain).)

Traza de ejecución:

The image shows a text-based execution trace. It consists of a long list of lines, each containing a series of numbers and text. The text is formatted as follows: `9, 18, 8, 7, 9, 8, 10, 10, 8, 0, 10, 0, 9, 21954457292887, 12.041594578792296, 12.041594578792296, 13.45362404707371, 2, false, 185, 7030, A`. The trace is displayed in a window titled 'traza_Fernando_Seguel - Notepad'.

(Fig. 16. Imagen referencial de una traza de ejecución de un proyecto de Inteligencia artificial.)

21 ANEXO: APLICACIÓN WEB

La aplicación web se desarrolló en PHP, funciona usando datos obtenidos a partir de trazas de ejecución (véase sección 20) subidas por el usuario, estos datos son almacenados en una sesión, descrita en el Manual de PHP (PHP Documentation Group, 2017):

“Las sesiones son una forma sencilla de almacenar datos para usuarios de manera individual usando un ID de sesión único. Esto se puede usar para hacer persistente la información de estado entre peticiones de páginas. Los ID de sesiones normalmente son enviados al navegador mediante cookies de sesión, y el ID se usa para recuperar los datos de sesión existente. La ausencia de un ID o una cookie de sesión permite saber a PHP para crear una nueva sesión y generar un nuevo ID de sesión.”

La obeitivo principal de la aplicación web, subir y evaluar trazas de ejecución, logra por medio de tres vistas (véase sección 8.2) que comparten comparten información a través de una sesión. La primera vista (upload) usa un formulario para subir un archivo, la segunda vista (assess) lee el archivo, lo elimina y entonces realiza el proceso de evaluación de los datos leídos (véase sección 9.4) y la tercera vista (pdf) exporta los datos de la evaluación a un archivo pdf utilizando métodos de la clase mpdf (integrada junto con Yii2 para este propósito). Gracias al uso de estas vistas y a la información persistente proveída por la sesión, una vez subido un archivo, los datos leídos se conservan en la sesión, se puede navegar entre estas vistas correctamente adelante y hacia atrás sin tener que volver a subir el mismo archivo.

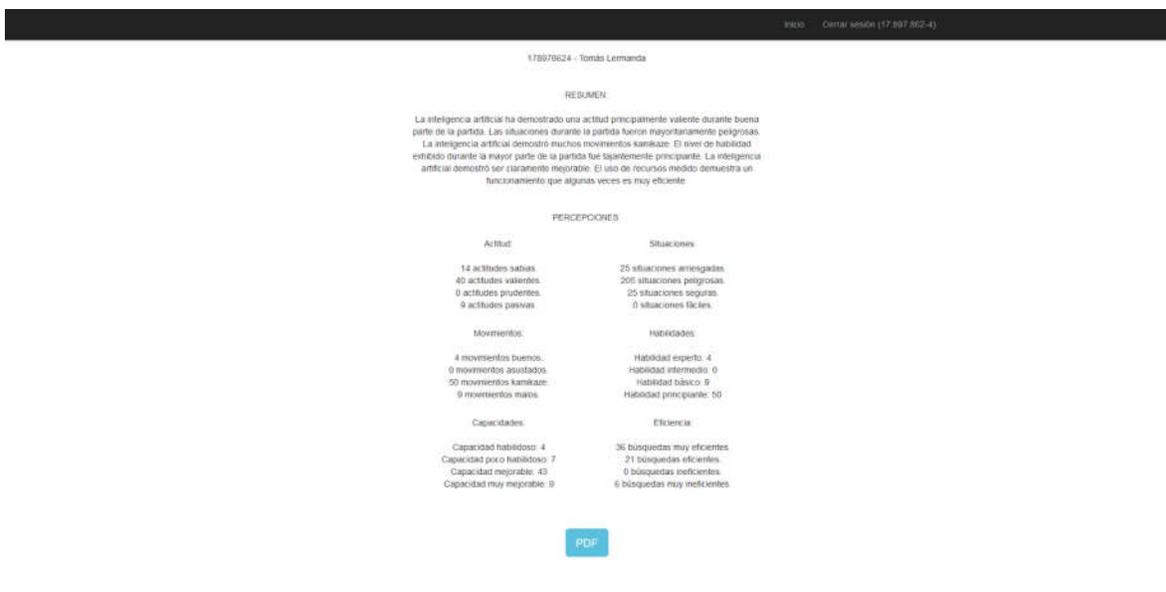
A continuación se muestran imágenes referenciales de las vistas antes mencionadas.

Upload:



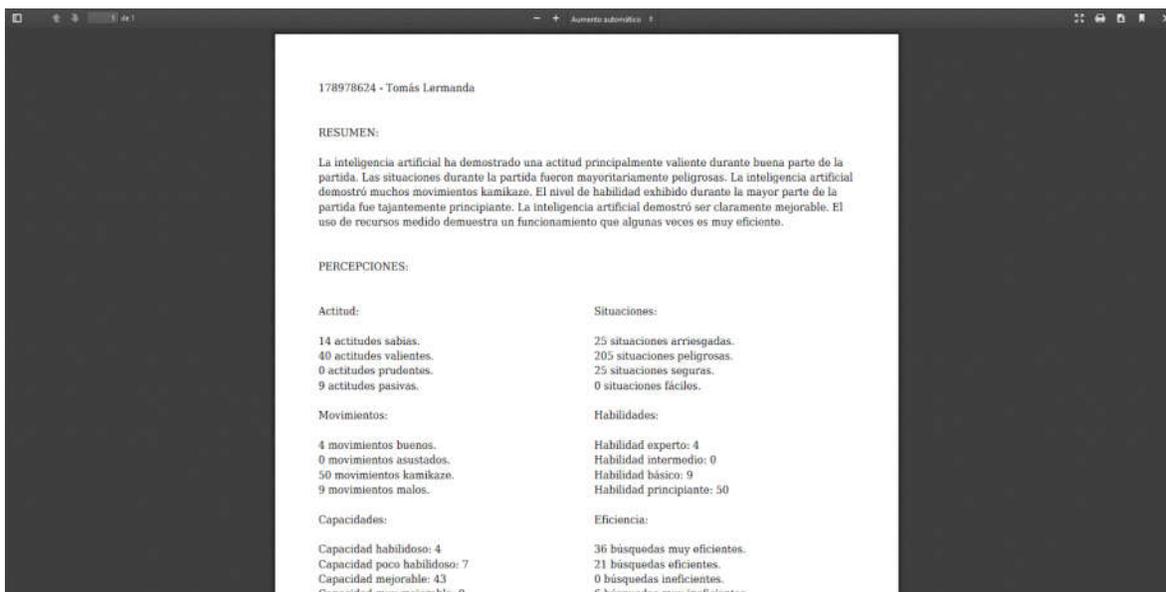
(Fig. 17. Imágen referencial de la vista upload.)

Assess:



(Fig. 18. Imágen referencial de la vista assess.)

Pdf:



(Fig. 19. Imágen referencial de la vista upload.)

22 ANEXO: MANUAL DE USUARIO

Objetivo del documento

El presente documento pretende mostrar al usuario el funcionamiento de la aplicación web “Generador automático de informes de evaluación en lenguaje natural de Proyectos de la asignatura Inteligencia Artificial”.

Objetivos:

Se pretende mostrar de una manera clara y concisa el funcionamiento de la aplicación web “Generador automático de informes de evaluación en lenguaje natural de Proyectos de la asignatura Inteligencia Artificial”.

Limitaciones:

Se asume que el usuario posee una traza de ejecución válida para interactuar con el sistema.

Manual de usuario:

Para acceder al sistema, el usuario debe usar un navegador web de escritorio y escribir la siguiente dirección web en la barra de direcciones:

<http://www.youractionsdefineyou.com/assess>

Una vez se ha accedido a la dirección, el usuario se encontrará en la pantalla inicial del sistema, correspondiente a la página principal de la aplicación web.

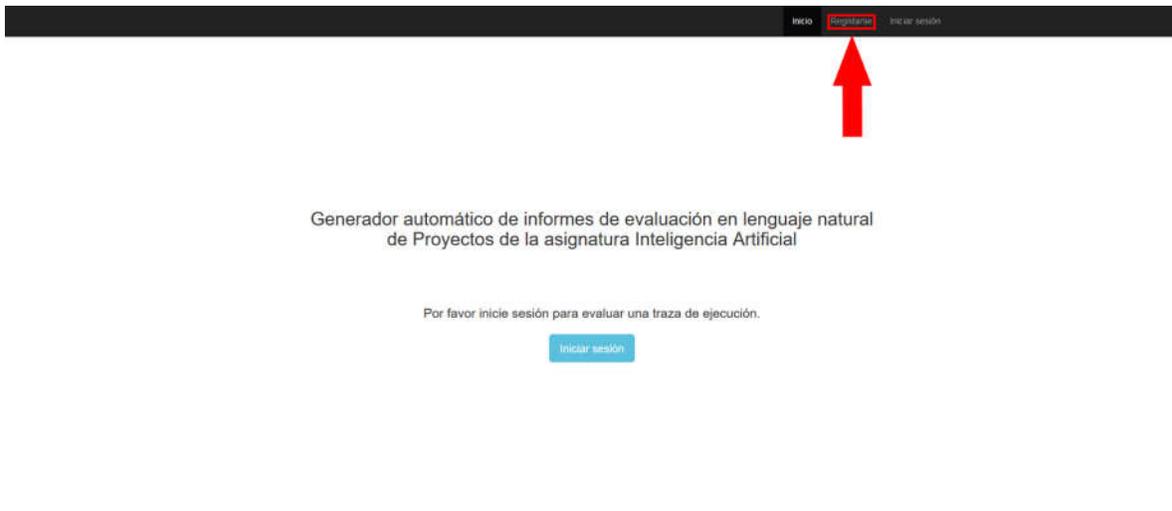
Pantalla inicial:

La pantalla de inicio de la aplicación web da la bienvenida al usuario al sistema, permite al usuario iniciar sesión o registrarse para hacer uso del sistema.

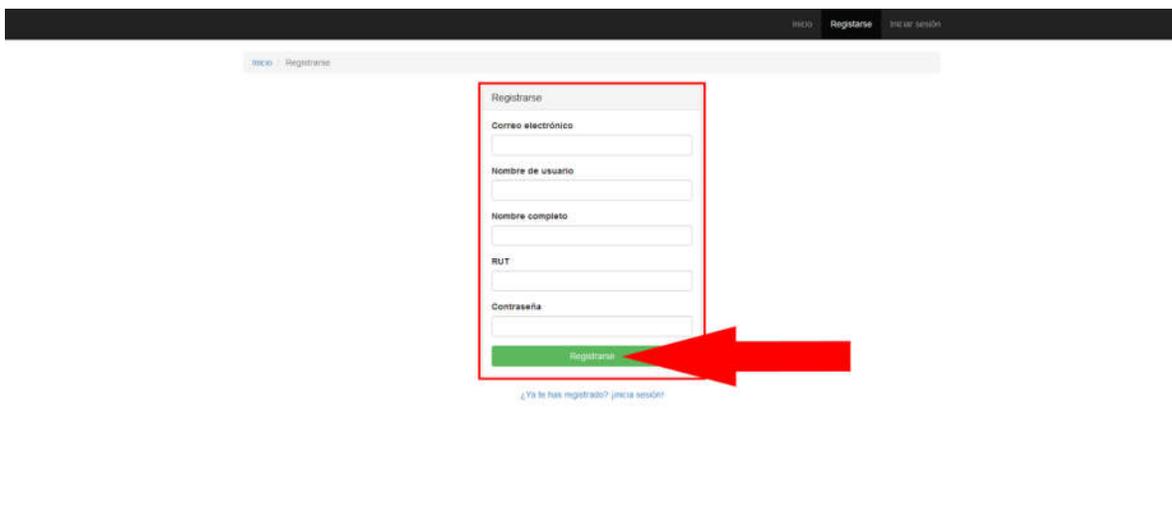


Registro:

1. Si el usuario no tiene registrada una cuenta para iniciar sesión entonces debe registrarse, para hacerlo, desde la pantalla inicial debe hacer clic en Registrarse, ubicado en la barra de navegación.

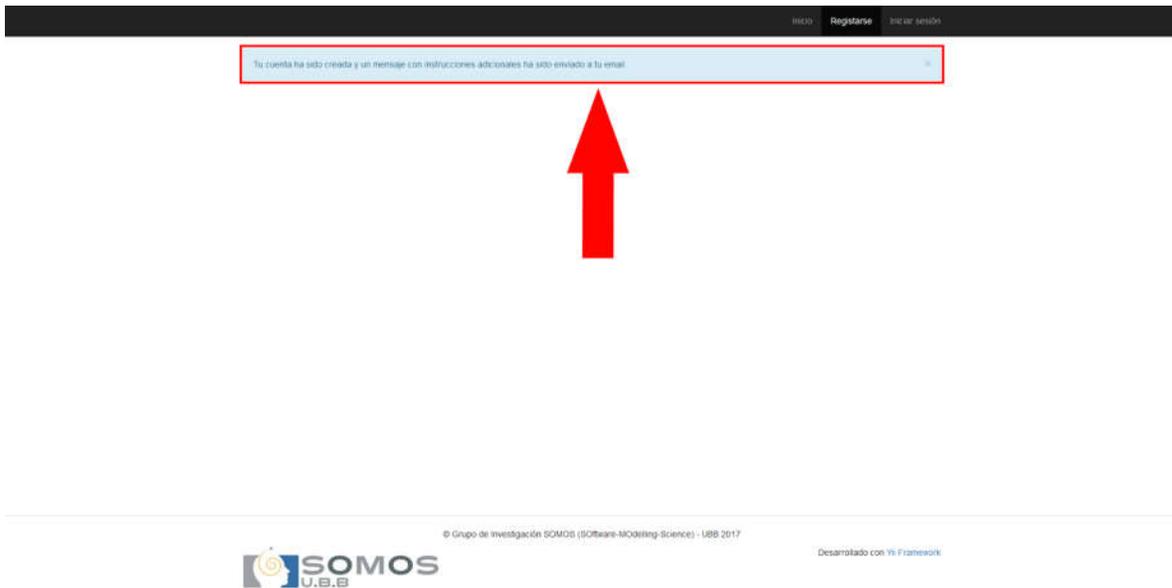


2. Una vez en la página de registro el usuario debe completar los campos y hacer clic en el botón registrarse.



Si el registro fue exitoso será redirigido a otra página donde podrá ver un mensaje de éxito, de lo contrario, permanecerá en la misma página y un mensaje error aparecerá debajo de alguno de los campos que completo.

Pantalla de registro exitoso:



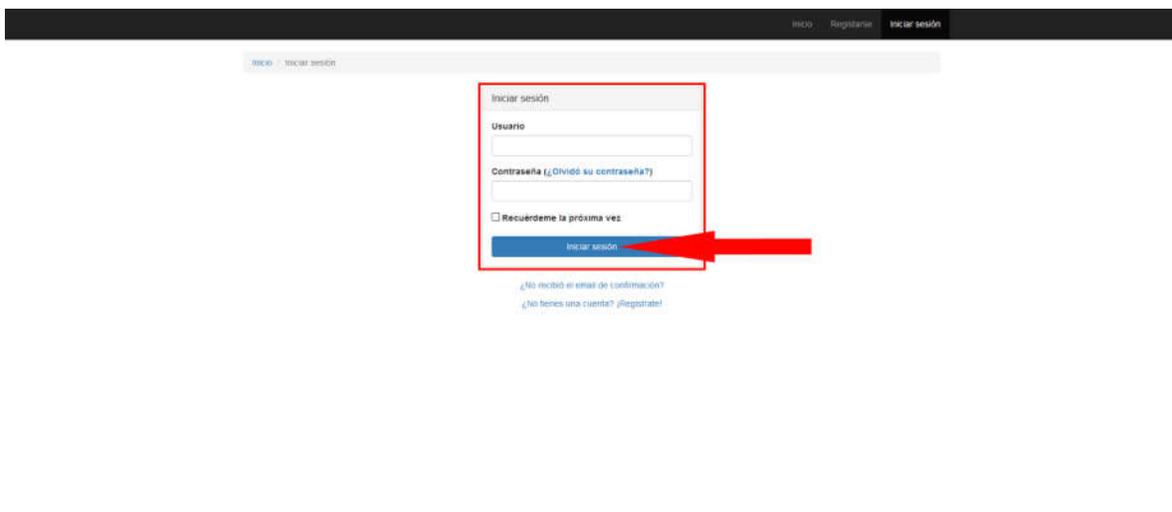
El mensaje de registro exitoso advertirá el envío de un correo electrónico de confirmación a la dirección de correo electrónico del usuario, pero la confirmación no es obligatoria para el uso del sistema y puede ser ignorada por completo.

Iniciar sesión:

1. Si el usuario ya está registrado, entonces, lo siguiente a hacer es iniciar sesión, para hacerlo, desde la pantalla inicial debe hacer clic en iniciar sesión, ubicado en el área central de la pantalla y en la barra de navegación.



2. Una vez en la página de iniciar sesión el usuario debe completar los campos y hacer clic en el botón iniciar sesión.



Si el inicio de sesión fue exitoso será redirigido a la página principal donde que ahora se verá diferente, de lo contrario, permanecerá en la misma página y un mensaje error aparecerá debajo de alguno de los campos que completo.

Pantalla inicial:

Con su inicio de sesión exitoso el usuario está listo para hacer uso del sistema y evaluar una traza de ejecución. La página principal de un usuario que ha iniciado sesión es la página desde donde se interactúa con el sistema, desde esta se subirán archivos para evaluar automáticamente.

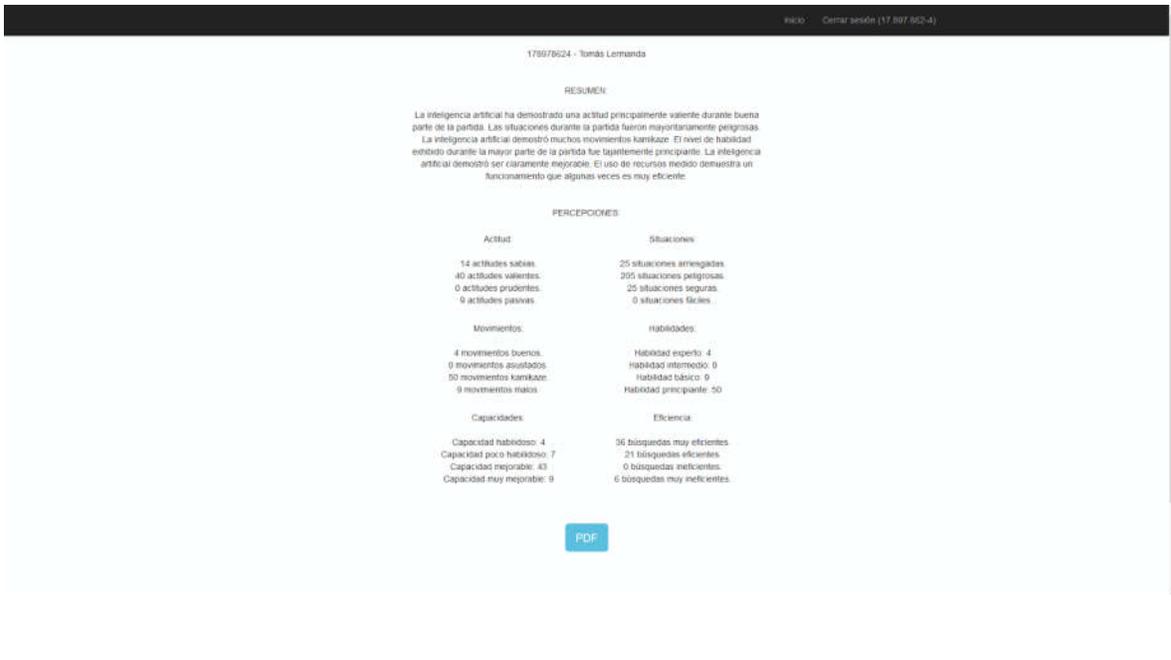


Evaluar traza de ejecución:

1. Para evaluar una traza de ejecución primero es necesario subir un archivo de traza de ejecución, para hacerlo, desde la página principal debe hacer clic en el botón examinar... y seleccionar el archivo de traza de ejecución desde los archivos de su equipo. Posteriormente debe hacer clic en el botón subir.



Si la subida es exitosa, el archivo es evaluado y entonces el usuario será redirigido a una página con los resultados, de lo contrario, permanecerá en la misma página y un mensaje error aparecerá junto al botón examinar o la página se recargará y aparecerá un mensaje de error en el área central sobre el título.



Exportar a un archivo pdf:

1. Si el usuario desea exportar los resultados de la evaluación a un archivo pdf, entonces, desde la página con los resultados de la evaluación debe hacer clic en el botón PDF, ubicado al pie de los resultados.



Esto resulta en la apertura de un archivo pdf en el mismo navegador de ser posible.

