



UNIVERSIDAD DEL BÍO-BÍO
FACULTAD DE CIENCIAS EMPRESARIALES
ESCUELA INGENIERÍA CIVIL INFORMÁTICA

Sistema de monitoreo y alertas tempranas de errores para
la plataforma educativa Adecca UBB.

15 de enero de 2021
Chillán - Chile

Cristóbal Ignacio Bravo Chávez

Profesor Guía
Carlos Eduardo San Juan Contreras

Profesor Co-guía
Marlene Elena Muñoz Sepúlveda

Memoria para optar al título de Ingeniero Civil en Informática.

Resumen

Este proyecto se presenta para dar conformidad a los requisitos exigidos por la Universidad del Bío-Bío para el proceso de titulación de la carrera Ingeniería Civil en Informática. El proyecto lleva por título “Sistema de monitoreo y alertas tempranas de errores para la plataforma educativa Adecca UBB” y tiene como finalidad implementar un sistema de monitorización a tiempo real de los eventos que ocurren al momento de utilizar la plataforma Adecca de la Universidad del Bío-Bío.

Dentro de las principales características del sistema se encuentra el poder capturar los eventos ocurridos en la plataforma Adecca, el cual consiste en almacenar todas las acciones y errores realizados por los usuarios de la plataforma Adecca, para generar estadísticas, sugerencias de optimización y alertas tempranas que ayudarán a la toma de decisiones del constante desarrollo de la plataforma Adecca.

La aplicación se realiza dentro del marco de la metodología iterativa incremental, usando una variante de la arquitectura modelo-vista-controlador, nos referimos al modelo-vista vista-modelo (MVVM), debido al uso del framework Angular para el desarrollo del front-end. Además, se utiliza el lenguaje de programación PHP junto al framework Laravel, necesario para la API Rest HTTP, y el lenguaje de modelado UML, con el objetivo de desarrollar un proyecto de calidad, pensando en satisfacer las necesidades del cliente.

Abstract

This Project is presented to solve the requirements demanded by the Bío Bío University to the Engineering in Computer Science degree process. This project has by title “Monitoring and Early Error Alert System for the Educational Platform Adecca UBB”, and has a purpose, implement a real time monitoring system of the events that take place at the time of use the Adecca platform of the Bío Bío University.

Within the main system characteristics is the capacity of capture the events occurred in the Adecca platform that consists on register and store all the actions and errors made by the users of Adecca, to generate statistics, give optimization suggestions and early alerts that will help on the decision making about the constant development of the Adecca platform.

The application was developed inside a iterative incremental methodology frame, due to use of the Angular framework to the Front-End development, the application was developed using a Model-View-Controller variant called Model-View Model-View (MVVM). Besides, the programming language used was PHP beside Laravel framework, which is necessary to develop the HTTP API Rest. And with purpose of develop a quality project and satisfy the client requirements it was decided to use the Unified Modeling Language (UML).

Índice General

1	<u>INTRODUCCIÓN</u>	10
2	<u>DEFINICIÓN DE LA INSTITUCIÓN</u>	12
2.1	Descripción de la Institución	12
2.2	Descripción del área de estudio	14
2.2.1	Departamento del Servicio Tecnológico.....	14
2.3	Descripción de la problemática	15
3	<u>DEFINICIÓN PROYECTO</u>	17
3.1	Objetivos del proyecto	17
3.2	Ambiente de Ingeniería deSoftware.....	17
3.3	Definiciones, Siglas y Abreviaciones.....	21
4	<u>ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE</u>	23
4.1	Alcances	23
4.2	Objetivo del software.....	23
4.3	Descripción Global del Producto.....	23
4.3.1	Interfaz de usuario	23
4.3.2	Interfaz de Hardware	24
4.3.3	Interfaz Software	24
4.3.4	Interfaces de comunicación	24
4.4	Requerimientos Específicos	24
4.4.1	Requerimientos Funcionales del sistema	24
4.4.2	Interfaces externas de entrada	25
4.4.3	Interfaces externas de Salida	26
4.4.4	Atributos del producto	26
4.4.4.1	Usabilidad:	26
4.4.4.2	Seguridad:	27
4.4.4.3	Interfaz Gráfica:	27
4.4.4.4	Disponibilidad.....	27
5	<u>FACTIBILIDAD</u>	28
5.1	Factibilidad técnica.	28
5.2	Factibilidad operativa.....	29
5.3	Factibilidad económica.....	30
5.3.1	Determinación de Costos	30
5.3.2	Estimación de los Beneficios	31
5.3.3	Flujo de Caja Neto.....	32
5.4	Conclusión de la factibilidad.....	33
6	<u>ANÁLISIS</u>	34
6.1	Procesos de Negocios futuros.....	34
6.2	Diagrama de casos de uso	36
6.2.2	Casos de Uso y descripción	37
6.2.3	Especificación de los Casos deUso	39
6.3	Modelamiento de datos.....	59
7	<u>DISEÑO</u>	62

7.1	Diseño Físico de la Base de datos.....	62
7.2	Diseño de arquitectura funcional.....	63
7.3	Diseño interfaz y navegación	66
7.3.1	Interfaz ingreso al sistema	67
7.3.2	Interfaz de configuración del usuario.....	67
7.3.2	Interfaz de configuración de las sugerencias.....	69
7.4	Especificación de módulos	70
8	<u>PRUEBAS</u>	72
8.1	Elementos de prueba	72
8.1.1	Módulo Estadísticas.....	72
8.1.2	Módulo Configuración Sugerencias.....	72
8.1.3	Módulo Configuración Usuarios.....	72
8.1.4	Módulo Ingreso de usuario.....	72
8.1.5	Módulo Sugerencias.....	72
8.1.6	Módulo Alertas.....	72
8.2	Especificación de las pruebas	73
8.3	Responsables de las pruebas	78
8.4	Calendario de pruebas	79
8.5	Detalle de las pruebas.....	79
8.5.1	Verificación de la calidad del producto	80
8.6	Conclusiones de Prueba	81
9	<u>PLAN DE CAPACITACIÓN Y ENTRENAMIENTO</u>	82
9.1	Usuarios a capacitar.....	82
9.2	Tipo de capacitación o entrenamiento	82
9.3	Funcionalidades o aspectos que serán abordados	82
9.4	Responsable de la capacitación	83
9.5	Tiempo estimado.....	83
9.6	Recursos requeridos	83
10	<u>RESUMEN ESFUERZO REQUERIDO</u>.....	84
11	<u>CONCLUSIONES</u>.....	85

Índice Tablas

Tabla 1: Detalle de detección del problema.....	16
Tabla 2: Requerimientos Funcionales.....	25
Tabla 3: Interfaces externas de entrada.....	25
Tabla 4: Interfaces Externas de Salida.....	26
Tabla 5: Especificación Pc-Laboratorio.....	28
Tabla 6: Especificación Servidor.....	29
Tabla 7: Flujo Caja Neto.....	32
Tabla 8: Especificación del Actor Administrador.....	36
Tabla 9: Especificación del Actor Desarrollador.....	36
Tabla 10: Especificación de los casos de uso: Autenticarse.....	39
Tabla 11: Especificación de los casos de uso: Seleccionar Alertas.....	40
Tabla 12: Especificación de los casos de uso: Seleccionar Sugerencias.....	41
Tabla 13: Especificación de los casos de uso: Configurar Sugerencia.....	42
Tabla 14: Especificación de los casos de uso: Ver Estadísticas de Memoria.....	43
Tabla 15: Especificación de los casos de uso: Ver Estadísticas de Navegadores.....	44
Tabla 16: Especificación de los casos de uso: Ver Estadísticas de Tiempo De Ejecución.....	45
Tabla 17: Especificación de los casos de uso: Ver DashBoard.....	46
Tabla 18: Especificación de los casos de uso: Asignar Privilegios.....	47
Tabla 19: Especificación de los casos de uso: Crear Usuario.....	48
Tabla 20: Especificación de los casos de uso: Modificar Usuario.....	49
Tabla 21: Especificación de los casos de uso: Eliminar Usuario.....	50
Tabla 22: Especificación de los casos de uso: Ver Sugerencia Tiempo de Ejecución.....	51
Tabla 23: Especificación de los casos de uso: Ver Sugerencia de Memoria Usada.....	52
Tabla 24: Especificación de los casos de uso: Ver Sugerencias de Memoria Máxima.....	53
Tabla 25: Especificación de los casos de uso: Ver Alerta de Errores PHP.....	54
Tabla 26: Especificación de caso de uso: Ver Alertas de Errores de Base de Datos.....	55
Tabla 27: Buscar alerta por mensaje de error.....	56
Tabla 28: Especificación de los casos de uso: Ver Notificación.....	57
Tabla 29: Especificación de los casos de uso: Ignorar Notificación.....	58
Tabla 30: Especificación módulo ingreso de sistema.....	70
Tabla 31: Especificación módulo Crear Usuario.....	71
Tabla 32: Especificación módulo Editar Usuario.....	71
Tabla 33: Especificación módulo Eliminar Usuario.....	71
Tabla 34: Especificación módulo Editar Configuración.....	71
Tabla 35: Especificación de prueba: Ingreso de Sistema.....	73
Tabla 36: Especificación de prueba: Estadísticas.....	74
Tabla 37: Especificación de prueba: Sugerencias.....	75
Tabla 38: Especificación de prueba: Alertas.....	76
Tabla 39: Especificación de prueba: Configuración de Sugerencias.....	77
Tabla 40: Especificación de prueba: Configuración de Usuario.....	78
Tabla 41: Resumen Pruebas.....	79
Tabla 42: Resumen Encuesta verificación de la calidad del producto.....	80
Tabla 43: Esfuerzo Requerido.....	84
Tabla 44: Funcionalidades Primer Incremento.....	91
Tabla 45: Funcionalidades Segundo Incremento.....	92
Tabla 46: Funcionalidades Tercer Incremento.....	93
Tabla 47: Caso de prueba 1: Crear Usuario.....	94
Tabla 48: Caso de prueba 2: Eliminar Usuario.....	94

<i>Tabla 49: Caso de prueba 3: Editar usuario.....</i>	<i>95</i>
<i>Tabla 50: Caso de prueba 4: Editar Configuración de Sugerencias</i>	<i>96</i>
<i>Tabla 51: Caso de prueba 5: Ingreso de usuario.</i>	<i>97</i>
<i>Tabla 52: Caso de prueba 6: Sugerencias Memoria Máxima.....</i>	<i>99</i>
<i>Tabla 53: Caso de Prueba 7: Sugerencias de Tiempo de Ejecución.....</i>	<i>100</i>
<i>Tabla 54: Caso de prueba 8: Sugerencias de Memoria Usada</i>	<i>101</i>
<i>Tabla 55: Caso de prueba 9: Visualizar Alerta de Errores.....</i>	<i>102</i>
<i>Tabla 56: Caso de prueba 10: Visualizar alerta en base de un mensaje</i>	<i>103</i>
<i>Tabla 57: Caso de prueba 11: Visualizar estadísticas.....</i>	<i>104</i>
<i>Tabla 58: Encuesta 1</i>	<i>117</i>
<i>Tabla 59: Encuesta 2</i>	<i>118</i>
<i>Tabla 60: Encuesta 3</i>	<i>119</i>

Índice Figuras

Figura 1: Organigrama de la Universidad del Bío-Bío.....	13
Figura 2: Organigrama del departamento de servicios tecnológicos.....	14
Figura 3: Esquema del modelo de desarrollo iterativo e incremental.....	18
Figura 4: Jerarquía de los Elementos de un sistema de Monitoreo.....	22
Figura 5: BPMN Sistema de Monitoreo Adecca.....	35
Figura 6: Diagrama de Casos de Uso.....	38
Figura 7: Esquema de documento: Configuración de Sugerencias.....	60
Figura 8: Esquema de documento: Alertas Notificaciones.....	60
Figura 9: Esquema de documento: Users.....	60
Figura 10: Esquema de datos: log_system.....	61
Figura 11: Formato de colección: alerta_notificaciones.....	62
Figura 12: Formato de colección: users.....	62
Figura 13: Formato de colección: configuración_sugerencias.....	63
Figura 14: Formato de colección: log_system.....	63
Figura 15: Diseño de la arquitectura funcional.....	65
Figura 16: Vista de Cabecera.....	66
Figura 17: Vista menú lateral.....	66
Figura 18: Vista Autenticación de Usuario.....	67
Figura 19: Vista listado de usuario.....	67
Figura 20: Vista creación de usuario.....	68
Figura 21: Vista edición de usuario.....	68
Figura 22: Vista mensaje de seguridad de eliminación de usuario.....	69
Figura 23: Vista de configuración de sugerencias.....	69
Figura 24: Vista de edición de configuración de sugerencia.....	70
Figura 25: Vista de venta emergente de confirmación de edición de configuración.....	70
Figura 26: Calendario de pruebas.....	79
Figura 27: Carta Gantt.....	89
Figura 28: Estructura Módulo Dashboard.....	105
Figura 29: Gráfico de cantidades estudiantes conectados en la plataforma.....	106
Figura 30: Usuarios Conectados.....	106
Figura 31: Dispositivos Usados por los Usuarios.....	106
Figura 32: Tops usuarios que más se conectan.....	107
Figura 33: Top Navegadores Usados por los Usuarios.....	107
Figura 34: Top páginas más visitadas.....	108
Figura 35: Sistemas operativos utilizados.....	108
Figura 36: Estructura de Módulo Estadísticas, sección memoria.....	109
Figura 37: URI que utilizan más memoria.....	110
Figura 38: Controladores que utilizan más memoria.....	110
Figura 39: Métodos que utilizan más memoria.....	110
Figura 40: Estadísticas de Navegadores.....	111
Figura 41: Estructura Módulo Sugerencias.....	112
Figura 42: Datos de análisis.....	113
Figura 43: Reporte de sugerencias.....	113
Figura 44: Estructura Módulo alerta.....	114
Figura 45: Errores por plataforma.....	115
Figura 46: Top 5 código de errores.....	115
Figura 47: Reporte de alertas.....	115

Figura 48: Listado de notificación..... 115
Figura 49: Detalle de notificación..... 116

1 INTRODUCCIÓN

La Universidad del Bío-Bío, está comprometida con entregar servicios de calidad a sus usuarios y a las distintas unidades, siempre orientada al mejoramiento continuo de procesos y servicios. Uno de sus grandes objetivos es fortalecer las plataformas informáticas enfocada a la educación y así complementar el aprendizaje presencial y no presencial, logrando una interacción más adecuada entre los estudiantes y profesores.

Siendo la plataforma Adecca una de las principales de la institución, se ha hecho necesario monitorear la actividad de los diferentes usuarios de manera personalizada para mejorar sus servicios, dado que, esta carece de una herramienta que capture y analice los múltiples eventos ocurridos dentro de ella.

En el presente trabajo se documenta la implementación de un sistema de monitoreo y de alerta temprana de errores para la plataforma educativa Adecca, cuya finalidad máxima es generar una mejora continua a través de estadísticas, sugerencias y notificaciones de alertas en tiempo real que apoyan la toma de decisiones. Importante es destacar que este sistema muestra de forma resumida y amigable, el análisis de millones de datos (eventos) que han sido capturado y almacenados en las bases de datos corporativas.

Para la elaboración de este sistema se ha utilizado la metodología de desarrollo de software Iterativo Incremental, la cual cuenta con etapas de análisis de requisitos, diseño, codificación y pruebas, en particular, este proyecto se dividió en 3 incrementos para asegurar la obtención de un software de calidad.

Uno de los propósitos de diferenciación es que este trabajo ha priorizado el uso de tecnologías y herramientas de vanguardia para conseguir un producto actual, robusto y escalable en el tiempo, por ejemplo, se ha utilizado Laravel (PHP), Angular (TypeScript) y MongoDB (NoSql) que actualmente son frameworks muy demandados por la industria del desarrollo.

Por último, este informe tiene el propósito de presentar la documentación del sistema desarrollado con el fin de entender la problemática y solución que se plantea. El contenido del documento está estructurado por capítulos, que son descritos brevemente a continuación.

En el capítulo 2 se describe la información de la empresa como el nombre o su estructura organizativa y también se detalla el problema que se aborda.

En el capítulo 3 se presentan los objetivos del proyecto y el ambiente en el cual será desarrollado.

En el capítulo 4 se presenta la especificación de los requerimientos, los alcances, el objetivo del software, descripción de los atributos del producto y los requerimientos específicos de este.

Luego, el capítulo 5 se detallan los resultados de la factibilidad del proyecto en los ámbitos técnico, operativo y económico.

Además, el capítulo 6 se presenta el análisis del proyecto donde se muestra y explica el proceso de negocios que tendrá la organización con el sistema implementado, los casos de uso y un modelo entidad relación.

En el capítulo 7 se enfoca en el diseño del sistema, donde se muestra el diseño físico de la base de datos, la arquitectura funcional y la interfaz junto a la navegación dentro del sistema. Además, se especifican los módulos de la aplicación.

De forma consecutiva, el capítulo 8 se presentan las pruebas que se realizan, especificando el tipo de pruebas, los responsables, y el respectivo resultado.

En el capítulo 9 se indica el plan de capacitación o entrenamiento que se utiliza para instruir al usuario final con el objetivo de enseñar a usar el sistema de forma correcta.

Inmediatamente en el capítulo 10 se muestra el resumen de esfuerzo requerido para desarrollar el proyecto completo.

En consecuencia, en el capítulo 11 se encuentran las conclusiones obtenidas del proyecto.

Finalmente, en el capítulo 12 se presentan la bibliografía donde se referencia todo el material de apoyo como páginas web, libros, documentos, entre otros.

2 DEFINICIÓN DE LA INSTITUCIÓN

En este capítulo se realizará una descripción de la institución y el área de estudio en la que se enfoca la problemática a solucionar.

2.1 Descripción de la Institución

Antecedentes generales de la Institución

- Nombre: Universidad del Bío-Bío.
- Dirección: Avenida Andrés Bello #720.
- Rubro: Educación.
- Productos y servicios: Principalmente educación superior de pregrado y post grado, formación continua e investigación científica. (Universidad del Bio Bio, s.f.)
- Competencia directa: Universidad de Concepción, Universidad Tecnológica Inacap, Universidad Adventista, entre otras instituciones de educación superior.

Misión

La Universidad del Bío-Bío, a partir de su naturaleza pública, responsable socialmente y estatal, tiene por misión, desde las Regiones del Biobío y Ñuble, aportar a la sociedad con la formación de personas integrales, a través de una Educación Superior de excelencia. Comprometida con los desafíos regionales y nacionales, 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 Bio Bio, s.f.)

Visión

Ser reconocida a nivel nacional e internacional como una Universidad pública, responsable socialmente y regional que, comprometida con su rol estatal, desde las Regiones del Biobío y Ñuble, forma personas integrales de excelencia y aporta a través de su quehacer al desarrollo sustentable de las regiones y el país. (Universidad del Bio Bio, s.f.)

Objetivos de la Institución

El objetivo principal es contribuir, en la educación, la investigación, asistencia técnica y capacitación, para la formación de profesionales competentes y autónomos que conjuguen en su desempeño laboral, las dimensiones de la formación humana y disciplinares, que satisfaga las necesidades de la sociedad y de un mercado profesional cambiante. Además, la universidad busca el desarrollo regional en el territorio donde realiza sus actividades, sin perjuicio de poder extender sus actividades, si las condiciones así lo requieren, al ámbito nacional e internacional. (Universidad del Bío-Bío, 2018)

Estructura Organizativa

En la **Figura 1** se presenta el modelo de organigrama de la Universidad del Bío-Bío. **Figura 1: Organigrama de la Universidad del Bío-Bío**

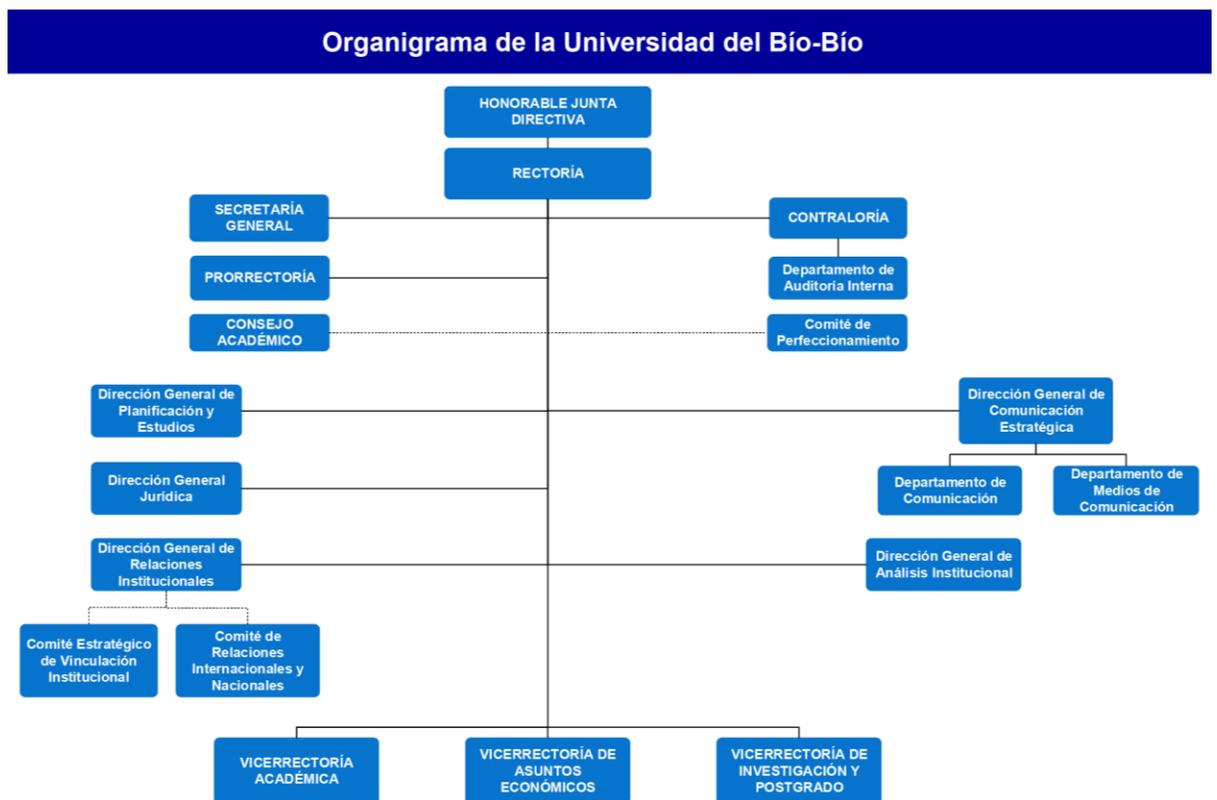


Figura 1: Organigrama de la Universidad del Bío-Bío

2.2 Descripción del área de estudio

2.2.1 Departamento del Servicio Tecnológico

El Departamento de Servicios Tecnológicos, tiene como finalidad administrar los recursos de redes, datos, software, hardware y telefónicos de la Sede Chillán. Para cumplir con su misión debe coordinarse con la Dirección de Informática, dependiente de la Vicerrectoría de Asuntos Económicos para prestar los servicios que demanden los usuarios, velando por el cumplimiento de las políticas informáticas de la Institución y apoyando el accionar de ambas Vicerrectorías y las distintas unidades de nuestra Universidad. (Universidad Del Bio-Bio, 2017)

En la **Figura 2** se muestra el organigrama del Departamento de Servicios Tecnológicos.

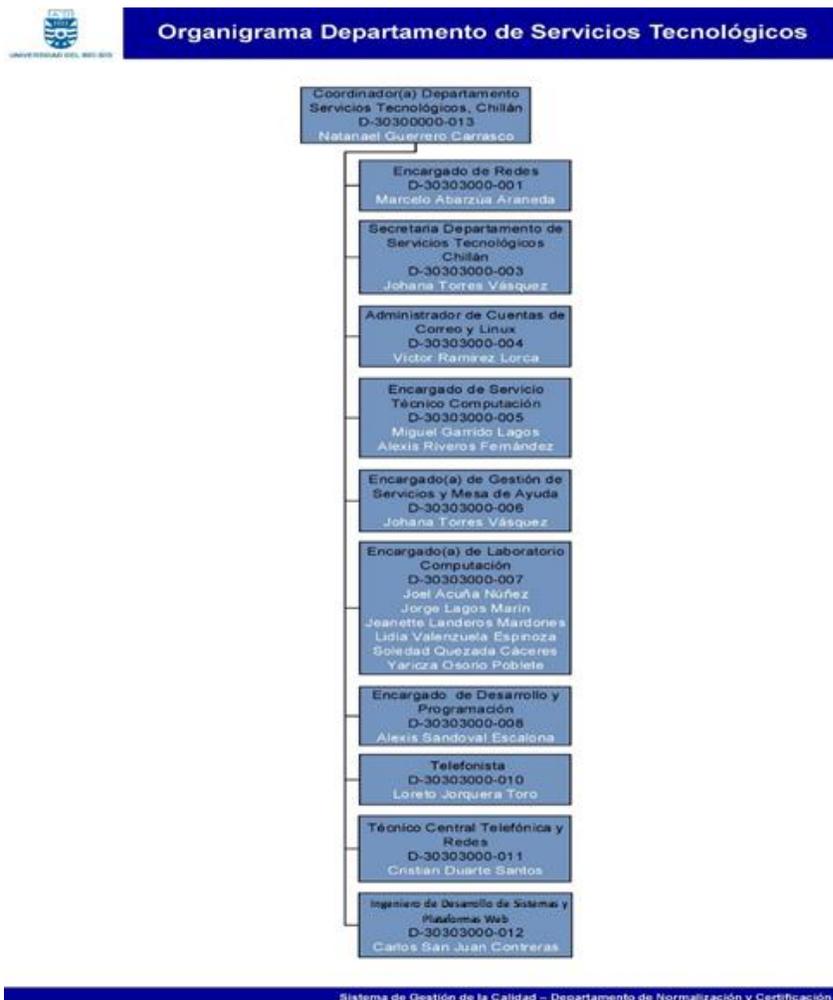


Figura 2: Organigrama del departamento de servicios tecnológicos

2.2.1.1 Área de Desarrollo de Sistemas del Departamento de Servicios Tecnológicos.

El Área de Desarrollo de Sistemas, tiene como objetivo desarrollar y mantener los sistemas de información necesarios para apoyar y optimizar la gestión del Departamento de Servicios Tecnológicos y de la propia Universidad. (Universidad del Bío-Bío, 2017)

2.3 Descripción de la problemática

Adecca es una Plataforma Educativa que tiene por finalidad prestar servicios de apoyo a la docencia de Pregrado y Postgrado de la Universidad del Bío-Bío, favoreciendo el uso de estrategias activas de enseñanza para potenciar ambientes de trabajo colaborativos. Sin embargo, Adecca, actualmente no cuenta con un sistema de monitoreo que capture los eventos ocurridos en ella, generados por los usuarios y por la propia plataforma, siendo la plataforma Adecca una de las más utilizadas en la Universidad del Bío-Bío.

Producto de lo anterior, los eventos, errores y acciones automáticas que ocurren en la plataforma Adecca tales como: la inoperabilidad del servidor, que el estudiante no pueda ingresar a un módulo o no pueda rendir una evaluación y/o no logre guardar o editar un documento en la plataforma, entre otras múltiples funcionalidades que tiene esta plataforma, origina que el tiempo de identificación y la resolución de los problemas mencionados sea mayor. Tales registros son de suma importancia porque permiten generar información relevante, por ejemplo: un proceso de auditoría y/o ayudar a la toma de decisiones, en cuanto se refiere al desarrollo de la plataforma.

En resumen, la plataforma carece de una herramienta que permita, al personal de desarrollo de la plataforma visualizar de una forma organizada y explícita los eventos, errores y acciones, en consecuencia, se hace relevante un sistema que permita capturar y almacenar los eventos ocurridos, entregar información sobre ellos y generar cambios en base a ellos.

En la **Tabla 1: Detalle de detección del problema.**Tabla 1 se presenta un detalle de identificación de problemas detectados.

N °	Problema
1	No almacena eventos, errores y acciones.
2	Imposibilidad de auditoria.
3	Costos por detección.
4	Altos tiempos de búsquedas de error.
5	Escasa visualización de incidencia de los errores.

Tabla 1: Detalle de detección del problema.

3 DEFINICIÓN PROYECTO

A continuación, este capítulo presenta los objetivos del proyecto y el ambiente de ingeniería de software en el cual se desarrollará. Además, se define las siglas y abreviaciones usados en el presente documento, para una mayor comprensión.

3.1 Objetivos del proyecto

General:

Capturar datos de los eventos, procesos automáticos y acciones ejecutadas por los usuarios que utilizan la plataforma educativa Adecca UBB, permitiendo la eficiente toma de decisiones al entregar información estadística, alertas y sugerencias en tiempo real de dichos eventos.

Específicos:

1. Capturar eventos de los usuarios para la realización de auditorías informáticas.
2. Generar información estadística,
3. Producir alertas y sugerencias en tiempo real.
4. Proveer información para la toma de decisiones.

3.2 Ambiente de Ingeniería de Software

Metodología

El proyecto se realizó bajo la metodología de desarrollo de software Iterativo Incremental, ya que esta metodología permite una constante retroalimentación del software por parte del cliente, mediante la división en secciones del proyecto, estos bloques contendrán funcionalidades sobre los requerimientos solicitados generando que por cada sección se ocasione un incremento. Lo anterior permite que, por cada incremento se origine una evaluación, retroalimentación o cambios en la sección, concluyendo en la suma de la totalidad de los incrementos, un software ideal al cliente. (Navarro Cadavid, Fernández Martínez, & Morales Vélez, 2013).

Cada iteración cuenta con análisis de requisitos, diseño, codificación y pruebas:

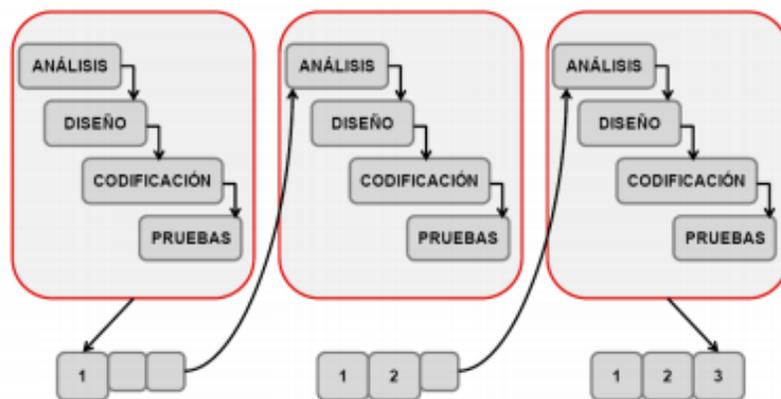
Análisis: El análisis se conforma por el reconocimiento del problema, generación de evaluaciones, obteniendo una síntesis del problema que se presentó, las especificaciones a realizar, el modelo de flujo de datos y la revisión, con el objetivo de desarrollar una validación general de los procesos.

Diseño: Es la designación de la arquitectura del software, la interfaz y utilización de algoritmos.

Codificación: Esta actividad comprende desarrollar el diseño en líneas de código legibles a la máquina, para la generación y configuración de comportamiento de los ambientes virtuales.

Pruebas: Finalizado el proceso de codificación, se inicia la etapa de pruebas al software, el cual se centra en asegurar el correcto funcionamiento del desarrollo de la codificación. (Cadavid, Juan, & Jonathan, 2013)

En la **Figura 3** se presenta un esquema para representar el correspondiente modelo iterativo incremental



Fuente: (Cadavid, Juan, & Jonathan, 2013)

Figura 3: Esquema del modelo de desarrollo iterativo e incremental.

Tecnologías

A continuación, se definen brevemente las tecnologías de desarrollo para dar solución al problema planteado.

- **CSS3:** Es la última evolución del lenguaje de las Hojas de Estilo en Cascada (Cascading Style Sheets), y pretende ampliar la versión CSS2.1. Trae consigo muchas novedades altamente esperadas, como las esquinas redondeadas, sombras, gradientes, transiciones o animaciones, y nuevos layouts como multi-columnas, cajas flexibles o maquetas de diseño en cuadrícula (grid layouts). (Mozilla and individual contributors, 2019).
- **GIT:** Es un sistema de control de versiones distribuido gratuito y de código abierto diseñado para manejar todo, desde proyectos pequeños hasta muy grandes, con velocidad y eficiencia. (GIT SCM, s.f.).
- **HTML5:** Su sigla significa HyperText Markup Language, es la última versión más reciente y que por lo tanto lleva el número cinco por su versión, además contiene un conjunto más amplio de tecnologías que permite a los sitios Web y a las aplicaciones ser más diversas y de gran alcance. (MDN web docs, 2020).
- **JavaScript:** Es un lenguaje de programación basado en prototipos, multi-paradigma, de un solo hilo, dinámico, con soporte para programación orientada a objetos, imperativa y declarativa. (MDN web docs, 2020).
- **TypeScript:** Es un lenguaje de programación libre y de código abierto desarrollado y mantenido por Microsoft. Es un superconjunto de JavaScript, que esencialmente añade tipos estáticos y objetos basados en clases. (TypeScript, 2020)
- **NoSQL:** Las bases de datos NoSQL están diseñadas específicamente para modelos de datos específicos y tienen esquemas flexibles para crear aplicaciones modernas. Las bases de datos NoSQL son ampliamente reconocidas porque son fáciles de desarrollar, por su funcionalidad y el rendimiento a escala (aws, s.f.).
- **PHP:** (Acrónimo recursivo de *PHP: Hypertext Preprocessor*) Es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML. (PHP, s.f.)

- **Websockets:** Es una tecnología avanzada que hace posible abrir una sesión de comunicación interactiva entre el navegador del usuario y un servidor con la finalidad de enviar mensajes a un servidor y recibir respuestas controladas por eventos sin tener que consultar al servidor (MDN Contributor, 2019).
- **Apache:** Es un servidor web HTTP de código abierto, para plataformas Unix, Microsoft Windows, Macintosh entre otros, que implementa protocolo HTTP/1.1 y la noción de sitio virtual según la normativa RFC 2616.
- **Redis:** Es un motor de base de datos en memoria, basado en el almacenamiento en tablas hashes pero que opcionalmente puede ser usada como una base de datos durable o persistente (Redislabs, 2019).

Herramientas

A continuación, se definen brevemente las herramientas a utilizar durante el proceso de desarrollo del proyecto.

- **MongoDB Compass:** Es la Interfaz gráfica de usuario (GUI) para MongoDB. El cual, tiene características como explorar visualmente los datos, ejecutar consultas y optimizar rendimiento de las consultas (mongoDb, s.f.).
- **Visual Studio Code:** Es un editor de código fuente ligero pero potente que se ejecuta en el escritorio. Viene con soporte incorporado para JavaScript, TypeScript y Node.js y tiene un rico ecosistema de extensiones para otros lenguajes (como C ++, C #, Java, Python, PHP, Go) y tiempos de ejecución (como .NET y Unity). (Visual Studio Code, s.f.).
- **Bizagi Modeler:** Es un poderoso modelador de procesos de negocio compatible con el estándar BPMN 2.0, diseñado para mapear, modelar y diagramar.
- **yEd Graph Editor:** Es una aplicación potente para generar diagramas de alta calidad de manera rápida y efectiva. Permite crear diagramas manualmente o importar datos externos para su análisis (yworks, s.f.).
- **Gitlab:** Es una plataforma DevOps completa, permitiendo que su organización se libere de las restricciones de una cadena de herramientas (GitLab, s.f.).
- **Firefox:** Es un navegador web libre y de código abierto desarrollador para Linux, Windows entre otros coordinado por la Corporación Mozilla y Fundación Mozilla.

Frameworks

A continuación, se definen brevemente los Frameworks a utilizar durante el proceso de desarrollo del proyecto.

- **Angular:** Es un marco de diseño de aplicaciones y una plataforma de desarrollo para crear aplicaciones eficientes y sofisticadas de una sola página. (Angular, s.f.).
- **Laravel:** Es un Framework de código abierto que ayuda a desarrollar diversas aplicaciones y servicios web de forma elegante y simple con el lenguaje de programación PHP desde la versión 5(Laravel, s.f.).

Gestor de Base de Datos

A continuación, se definen brevemente el gestor de base de datos a utilizar durante el proceso de desarrollo del proyecto.

- **MongoDB:** Es una base de datos distribuida, basada en documentos y de uso general que ha sido diseñada para desarrolladores de aplicaciones modernas y para la era de la nube (mongoDB, s.f.)

3.3 Definiciones, Siglas y Abreviaciones

En esta sección se incluye la definición de las siglas, abreviaciones, conceptos técnicos o de negocios que son necesarios para la comprensión del documento.

- **Eventos:** Corresponde a los sucesos automáticos producidos por el sistema ante una acción del usuario en la plataforma Adecca.
- **Sugerencias:** Corresponde a la proposición de observaciones obtenidas y analizadas, con la finalidad de mejorar la plataforma Adecca.
- **Errores:** Peticiones incompletas del servidor o del cliente que está acompañado de un código que tiene un significado.
- **Datos:** Corresponde a la información de las acciones realizadas por los usuarios de la plataforma Adecca.
- **Alertas:** Señal o aviso de peligro al repetir o superar eventos que generen caídas o baja de rendimiento de la plataforma Adecca UBB.

Respecto a la jerarquía, esta se basa en una adaptación propia al protocolo de registro de sistemas RFC 5424. Los elementos dentro de un sistema de monitoreo se representan en la **Figura 4**, donde se muestra en el nivel más alto los Datos, que tiene de hijos los Errores, Alertas y Sugerencias. En el siguiente nivel se encuentran los Errores, la cual tiene de hijo a las Alertas, posteriormente están las Alertas, que tiene de hijos las Sugerencias, Finalmente en el cuarto y último nivel, ya que no presenta hijos, se encuentra las Sugerencias

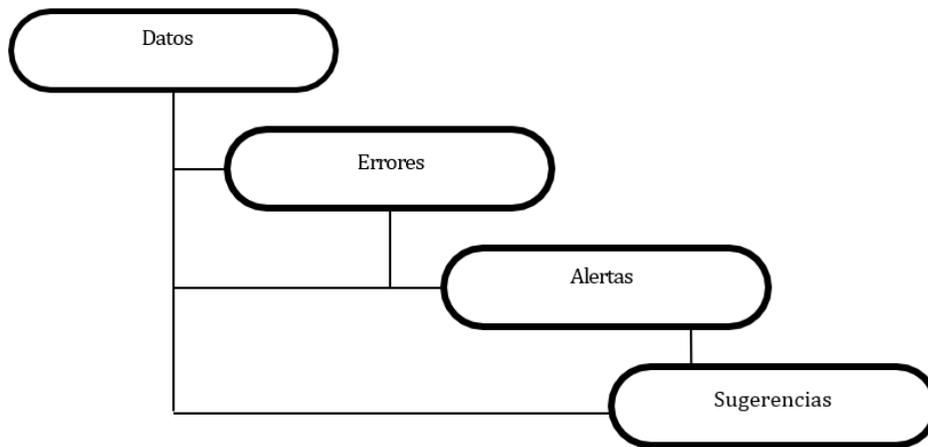


Figura 4: Jerarquía de los Elementos de un sistema de Monitoreo.

Fuente: Elaboración Propia

4 ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE

En este capítulo se definen los alcances y limitaciones del sistema, el principal objetivo del software, la descripción del producto y los requerimientos específicos de este.

4.1 Alcances

- El sistema debe ser desarrollado como aplicación web.
- El sistema debe ser responsive, es decir, que sea adaptable a pantallas de diversos tamaños, ya sea en un ordenador o un dispositivo móvil.
- El usuario debe autenticarse para utilizar el sistema.
- El sistema le debe otorgar acceso con privilegios a los administradores de la plataforma Adecca.
- El sistema debe entregar sugerencias, alertas, reporte y/o estadísticas en tiempo real.

4.2 Objetivo del software

Permitir, a los desarrolladores de la plataforma Adecca identificar de forma explícita y amigable los eventos ocurridos durante la utilización de la plataforma, con el fin de mejorar la toma decisiones e impulsar la mejora continua, gracias a la información entregada por el sistema de monitoreo.

4.3 Descripción Global del Producto

4.3.1 Interfaz de usuario

El sistema tiene una cabecera el que tiene el nombre del sistema y el logo de Adecca, además, una barra lateral (menú) donde se ubican las distintas opciones del sistema, Desde el punto de vista de la usabilidad, mantiene los colores de la plataforma Adecca; los estilos y gráficos que son designados por el desarrollador del sistema, asimismo, debe mantener un vocabulario uniforme para entregar un uso claro al usuario, siendo intuitiva, es decir de fácil uso, adaptación y entendimiento, permitiendo en un tiempo acotado que el usuario pueda identificar y moverse en las distintas opciones que dispone el sistema.

4.3.2 Interfaz de Hardware

La interfaz de hardware a nivel de los dispositivos a utilizar, solo se necesita para generar acciones en el sistema, tales como: ratón, teclado, monitor, CPU o un teléfono (smartphone). No se requiere de ningún hardware externo especial. Por último, destacar que requiere conexión a internet para interactuar con el sistema.

4.3.3 Interfaz Software

El sistema necesita de un navegador web, como: Firefox y conexión a internet.

Para este sistema se utiliza algunos softwares como, Apache Http Server, encargado del servidor web y MongoDB para la conexión y administración de la base de datos.

4.3.4 Interfaces de comunicación

El sistema presenta un servidor central conectado a Internet, con un protocolo de comunicación TCP/IP, por su amplio uso y seguridad que posee, cuando la información llega al receptor.

4.4 Requerimientos Específicos

4.4.1 Requerimientos Funcionales del sistema

Los requerimientos funcionales definen las actividades y servicios que un sistema debe proveer. Normalmente este tipo de requerimientos están vinculados con las entradas, las salidas de los procesos y los datos, estos están basados en las 3 métricas principales, que son: métodos, controladores y las URI debido a la arquitectura Modelo Vista Controlador de Adecca. A continuación, en la **Tabla 2**, se describen los requerimientos funcionales del sistema.

Id	Nombre	Descripción
RF1	Autenticación	El sistema controla el acceso mediante las credenciales del usuario que consta de su RUN y los últimos dígitos del RUN sin contar el guión ni el dígito verificador. Esta es la clave por defecto para el primer acceso. Luego, al usuario se le solicita obligatoriamente un cambio de contraseña para mejorar la seguridad de su cuenta.
RF2	Generar Sugerencias	El sistema proporciona sugerencias sobre los tiempos de ejecución o uso de memoria de la plataforma Adecca en base a los datos capturados y almacenados de la plataforma.

RF3	Generar Estadísticas	El sistema entrega tablas de datos ordenados, gráficos y/o diagramas en base a los datos producidos y almacenados de la plataforma Adecca.
RF4	Generar Alertas	El sistema entrega alertas en base a eventos críticos, por ejemplo, la petición incompleta a un método genera un código y un mensaje de error, dicha información se almacena y notifica al usuario.
RF5	Gestionar Usuario	El sistema permite crear, modificar y eliminar un usuario. Al crearlo, se debe definir el rol que tiene dentro del sistema (administrador o desarrollador).
RF6	Configurar Sugerencias	El sistema permite configurar las sugerencias en base a porcentajes de tolerancias, que son representadas con valores numéricos del 0 al 1, siendo el valor 1 la correspondencia del 100%.
RF7	Asignar rol de usuario	El sistema permite asignar un rol a los usuarios, según la asignación de este rol, los usuarios pueden contar con los privilegios de gestión (rol administrador) y/o visualizar contenidos (rol desarrollador).
RF8	Notificar sugerencias	El sistema permite notificar de forma automatizada ocurrencias de sugerencias producidos por aumento de memoria, tiempo de ejecución o memoria máxima utilizada, información que es generada por los múltiples eventos de los usuarios al visitar diferentes secciones de la plataforma.

Tabla 2: Requerimientos Funcionales.

4.4.2 Interfaces externas de entrada. Tabla 3

Id	Nombre del ítem.	Detalle de Datos contenidos en ítem
IEE01	Datos de creación de un usuario	Nombre del usuario. Apellido del usuario. Correo electrónico. RUN del usuario. Rol del usuario.
IEE02	Datos de configuración de sugerencias.	Porcentaje de tolerancia al método. Porcentaje de tolerancia al controlador. Porcentaje de tolerancia del identificador de recurso uniforme (URI).

Tabla 3: Interfaces externas de entrada

4.4.3 Interfaces externas de Salida. Tabla 4

Id	Nombre del ítem.	Detalle de Datos contenidos en ítem	Medio Salida
IES01	Lista de Usuario	Nombre del usuario. Apellido del usuario. Correo electrónico del usuario. Rol del usuario.	Pantalla
IES02	Lista de sugerencias	Nombre de la sugerencia. Memoria máxima alcanzada. Cantidad sobre el porcentaje tolerado. Tolerancia de la sugerencia. Promedio de memoria máximo tolerado de la sugerencia.	Pantalla
IES03	Lista de alertas	Nombre de la alerta.	Pantalla
IES04	Lista de la estadística	Nombre de la estadística. Cantidad de la estadística. Top de las estadísticas.	Pantalla
IES05	Configuración de las sugerencias	Nombre de la configuración de la sugerencia. Tiempo máximo de ejecución. Memoria máxima. Memoria usada.	Pantalla

Tabla 4: Interfaces Externas de Salida

4.4.4 Atributos del producto

A continuación, se indican los atributos críticos de calidad del producto que se desarrolla.

4.4.4.1 Usabilidad:

El sistema web cumple con un diseño responsivo con la finalidad de que el usuario pueda visualizar el contenido del sistema en distintas pantallas y así evitar limitaciones del dispositivo a utilizar, además, el sistema provee alertas en caso de alguna modificación o eliminación de datos (eventos críticos), y también, el despliegue de mensajes de éxito o error para que el usuario tenga siempre conocimiento de sus acciones en el sistema.

4.4.4.2 Seguridad:

El sistema solo permite el ingreso de los usuarios que tengan una cuenta registrada, cada vez que ingresen al sistema, se les solicitarán sus credenciales, en donde se verifica que los datos ingresados sean correctos.

Este atributo incluye las siguientes características:

- **Integridad:** Para generar modificaciones de los datos solo el sistema permite a aquellos usuarios que cuenten con privilegios que son definidos según su rol, con la finalidad de mantener la información segura en su totalidad.
- **Confidencialidad:** El sistema provee acceso sólo a usuarios que tengan una cuenta generada por el administrador del sistema.

4.4.4.3 Interfaz Gráfica:

El sistema ha sido desarrollado siguiendo el manual de desarrollo y las normas gráficas de la Universidad del Bío-Bío.

4.4.4.4 Disponibilidad

La información que entrega el sistema debe estar siempre disponible para las personas o usuarios que la necesiten, lo anterior es alcanzado gracias a que los servidores en el cual se almacena la aplicación tienen tecnologías, las cuales evitan que el servidor se apague ante un corte de luz, además de backups o respaldo de la información y registros de las bases de datos.

5 FACTIBILIDAD

La factibilidad se refiere a la disponibilidad de los recursos necesarios para llevar a cabo los objetivos del proyecto. La factibilidad presenta tres aspectos:

- **Técnico:** proporciona la información que hace referencia a los recursos de hardware y software necesarios.
- **Operativo:** determina el nivel de impacto del sistema y su funcionamiento.
- **Económico:** determina si la solución propuesta es económicamente rentable.

El grado de factibilidad que presenta cada aspecto mencionado anteriormente determina el éxito del proyecto.

5.1 Factibilidad técnica.

Para realizar este estudio se solicita ayuda al personal del Departamento de Servicios Tecnológicos (DST) con la finalidad de averiguar qué recursos tiene la Universidad, para generar una conclusión que determine si cumple técnicamente con la tecnología solicitada y requerida para el desarrollo del proyecto. Actualmente, la universidad cuenta con computadores que están ubicados en los laboratorios centrales, sus características se ven representadas en la **Tabla 5**, cumpliendo con los requerimientos básicos para el desarrollo y funcionamiento del proyecto.

SOFTWARE	SISTEMA OPERATIVO WINDOWS 10.
	Procesador I5-6500 de 3.2 Ghz.
	Memoria RAM de 8 GB.
HARDWARE	Disco duro de 1 TB.
	Tarjeta de video NVIDIA Gforce GTX1050 de 2GB.
	Monitor LG Full HD de 22”.

Tabla 5: Especificación Pc-Laboratorio.

Además, la universidad dispone de un rack de 5 servidores específicamente destinado para el hospedaje de la plataforma Adecca, su base de datos y los archivos de sus usuarios, las características de cada servidor se visualizan en la **Tabla 6**.

SOFTWARE	SISTEMA OPERATIVO FEDORA 29. Procesador de 8 núcleos Xeon Silver 4110 2.1 GHz.
HARDWARE	Memoria RAM de 16GB. 3 discos duros de 600GB.

Tabla 6: Especificación Servidor.

Requerimientos de Implementación

Las herramientas necesarias para el desarrollo del sistema son:

- Lenguaje de programación: PHP, TypeScript, NoSQL.
- Tecnologías: Laravel, Angular, MongoDB.

Conclusión de la Factibilidad Técnica

Esta solución presenta una implementación viable, ya que la Universidad tiene acceso a la tecnología necesaria, concluyendo que es Técnicamente Factible.

5.2 Factibilidad operativa.

Consiste en evaluar el impacto que tendrá el sistema dentro de la comunidad universitaria, con respecto a la resistencia al uso que puedan presentar los usuarios de la aplicación. (Arias, 2020)

Actualmente, el departamento de Soporte Tecnológico cuenta con el personal capacitado para utilizar el sistema, ya que estos pertenecen al área de la Tecnología de la información. Por lo tanto, no existirá dificultad ni resistencia por parte de los usuarios respecto de la creación de este nuevo sistema para Adecca.

Conclusión a la factibilidad operacional

Ante el análisis expuesto, se concluye que es Operacionalmente Factible dado el aumento progresivo del uso que tiene la plataforma Adecca, por lo que se hace necesario monitorearla para resolver eventos y/o realizar mejoras en un menor tiempo, con el objetivo que se encuentre habilitada y disponible para los requerimientos de los usuarios.

5.3 Factibilidad económica.

Consiste en estimar los costos y beneficios de llevar a cabo el proyecto. Para evaluar este aspecto, se ha decidido utilizar el Valor Actual Neto (VAN), cuyo resultado determina si el proyecto es rentable. (H.S., 2019)

5.3.1 Determinación de Costos

5.3.1.1 Costos de inversión

Para este proyecto se necesita contar con hardware y software que permitan enfrentar el desarrollo, los cuales se detallan en la sección de factibilidad técnica. El servidor para alojar la aplicación ya se encuentra disponible para utilizar en la universidad por lo tanto no tendrá ningún gasto, además, el software utilizado para el desarrollo del sistema es de licencia libre y su costo es de \$0 pesos.

En relación con el costo del personal, específicamente se requiere solo de un Ingeniero Civil en Informática.

- En el mercado actual se puede calcular el costo de un Ingeniero Civil en Informática al precio de \$5.000 por hora*.
- El tiempo de desarrollo estimado es de 6 meses, trabajando 20 horas a la semana, dando un total de 480 horas de producción. Por último, la suma total arroja como resultado \$2.400.000., siendo este mismo valor el total de inversión.

5.3.1.2 Costo de instalación y capacitación

La solución que se propone contempla la instalación y capacitación de usuarios ya que, a pesar de que sea un sistema automatizado se requerirá interacción de algún agente en el sistema, implicando un tiempo de instalación de 2 días, trabajando 8 horas diarias.

El precio por hora del Ingeniero Civil en Informática será el mismo valor presentado en los costos de inversión, es decir \$5.000 por hora, lo que da un total de \$80.000 por el coste de instalación.

5.3.1.3 Costos de operación y mantención

En lo que se refiere a costo de operación, existe el personal con el conocimiento suficiente demostrado en el tiempo que está instalado Adecca como plataforma educativa.

Por otro lado, el costo de mantención del sistema está estimado en un total de \$40.000 anuales, ya que se necesita el servicio del ingeniero civil en informática para realizar mantenciones cada 3 meses con tiempo de duración por mantención de 2 horas.

En conclusión, el total de los costos de la solución es de \$2.480.000. esta cifra corresponde a la suma de costo de inversión e instalación que presenta el proyecto. Además, se debe agregar el costo anual de mantención del sistema, después del año 0, que son \$40.000.

5.3.2 Estimación de los Beneficios

5.3.2.1 Beneficios Tangibles

Economizar Horas/Hombres: La Universidad del Bío-Bío hoy en día cuenta con un desarrollador y un administrador para la administración y programación de la plataforma educativa Adecca. Estos se verán beneficiados directamente al utilizar el Sistema de Monitoreo; sí el problema lo puede solucionar sólo el desarrollador el tiempo que gasta en encontrar y solucionar un evento que ocurre en la plataforma es de 1 hora, la cantidad de eventos mínimos que ocurren durante un mes es 10, además, hay que tener en cuenta que ocurren eventos de mayor complejidad de solución o mejora que necesitaría una intervención del administrador, estos eventos ocurren 4 vez al mes con tiempo de estrategia 3 horas y 6 horas de desarrollo más a asignar al desarrollador. En la Universidad la plataforma Adecca, el desarrollador tiene como promedio un valor por hora de \$5.500, el administrador con un valor por hora de \$9.740*.

En resumen, el desarrollador gasta 16 horas mensuales en solucionar eventos que ocurren en la plataforma, con un costo por hora de \$5.500 dando un total de \$88.000 mensuales. También el administrador genera estrategia de solución con un tiempo de 12 horas por \$ 9.740 por hora dando un total de \$ 116.880. Esto deduce que el ahorro por mes es de \$204.800 mensual y anual de \$2.048.000, ya que, la Universidad trabaja en promedio 10 meses.

5.3.2.2 Beneficios Intangibles

Control de la plataforma: El personal de la Universidad que trabaje en la plataforma Adecca tendrá un mejor control sobre los eventos que ocurran en ella de manera fácil, entendible y rápida.

5.3.3 Flujo de Caja Neto

Para verificar la factibilidad económica del proyecto, se calculará el VAN (Valor Actual Neto) demostrado en la **Tabla 7**. Para este cálculo tendremos los siguientes aspectos:

- El impuesto aplicado es del 10%.
- La estimación de vida útil de 5 años.

Flujo Caja Neto						
	Año 0	Año 1	Año 2	Año 3	Año 4	Año 5
Beneficio	\$ -	\$ 2.048.000	\$ 2.048.000	\$ 2.048.000	\$ 2.048.000	\$ 2.048.000
Costo Fijo	\$ -	\$ -40.000	\$ -40.000	\$ -40.000	\$ -40.000	\$ -40.000
Costo de Inversión	\$-2.400.000					
Costo de Instalación	\$ -90.000					
Total de Costos	\$-2.490.000	\$ -40.000	\$ -40.000	\$ -40.000	\$ -40.000	\$ -40.000
Flujo Caja	\$-2.490.000	\$ 2.008.000	\$ 2.008.000	\$ 2.008.000	\$ 2.008.000	\$ 2.008.000

Tabla 7: Flujo Caja Neto

5.3.4 Análisis del VAN

Para verificar si el proyecto es factible económicamente se utilizará el indicador VAN, que permite saber si los beneficios son mayores que los costos en un periodo de 5 años como horizonte.

$$VAN = \sum_{i=0}^N \frac{B_i - C_i}{(1+r)^i}$$

Fórmula 1: VAN

Donde

B_i: beneficio del periodo i

C_i: costo del periodo i

i: periodo

n: vida útil del proyecto.

r: tasa de descuento

$$VAN(10\%) = \frac{2.048.000}{(1+0.1)^1} + \frac{2.048.000}{(1+0.1)^2} + \frac{2.048.000}{(1+0.1)^3} + \frac{2.048.000}{(1+0.1)^4} + \frac{2.048.000}{(1+0.1)^5} - 2.490.000$$

$$VAN (10\%) = 5.121.900$$

5.4 Conclusión de la factibilidad

A partir del análisis de la factibilidad técnica, operativa y económica del proyecto se afirma que la implementación del sistema es viable, teniendo una proyección exitosa por el indicativo positivo que entrega el VAN, implicando también un ahorro de tiempo tanto para el desarrollador como para el administrador de la plataforma educativa Adecca.

6 ANÁLISIS

En este capítulo se desarrolla un análisis de las características del software por medio de distintos diagramas, en primera instancia se muestra el diagrama de proceso de negocios, luego la representación de los requerimientos funcionales a través de un diagrama de casos de uso y, finalmente, el modelamiento de los datos mediante un diagrama para modelos no relacionales.

6.1 Procesos de Negocios futuro

Debido a que la Universidad del Bío-Bío no cuenta con un sistema implementado de monitoreo para la plataforma Adecca, se realiza el modelo de procesos de negocios futuro, que representa el funcionamiento de forma general del nuevo sistema y la interacción entre un usuario de la plataforma Adecca y el usuario del sistema de monitoreo.

Este modelo del proceso de negocios sobre el nuevo sistema contiene dos “pool”, uno de ellos representa un usuario de la plataforma Adecca y el usuario del sistema de monitoreo representado por el Departamento de Servicio Tecnológico (DST). El modelo inicia cuando el usuario de la plataforma Adecca ingresa y genera una acción en la plataforma, estas acciones pueden ser: ingresar a un módulo, responder cuestionario, subir archivo, entre otros. Una vez que el usuario de la plataforma no requiera realizar más acciones en la plataforma, el usuario de la plataforma cierra su sesión terminada su participación.

Las acciones del usuario serán almacenadas en una base de datos, la que entrega información para generar las sugerencias, alertas y datos al Desarrollador y Administrador dando inicio a su participación en el “pool”, el cual consulta y analiza sugerencias, alertas o estadísticas generadas de la base de datos. Si no requiere atención del Administrador, el desarrollador genera una estrategia de solución o mejora, terminando el proceso. En caso que sí requiera atención, el Desarrollador entrega detalles al Administrador para desplegar una estrategia de solución que implementa el Desarrollador finalizando el proceso de negocio.

En la **Figura 5** se visualizará el diagrama de procesos de negocios para el sistema de monitoreo Adecca.

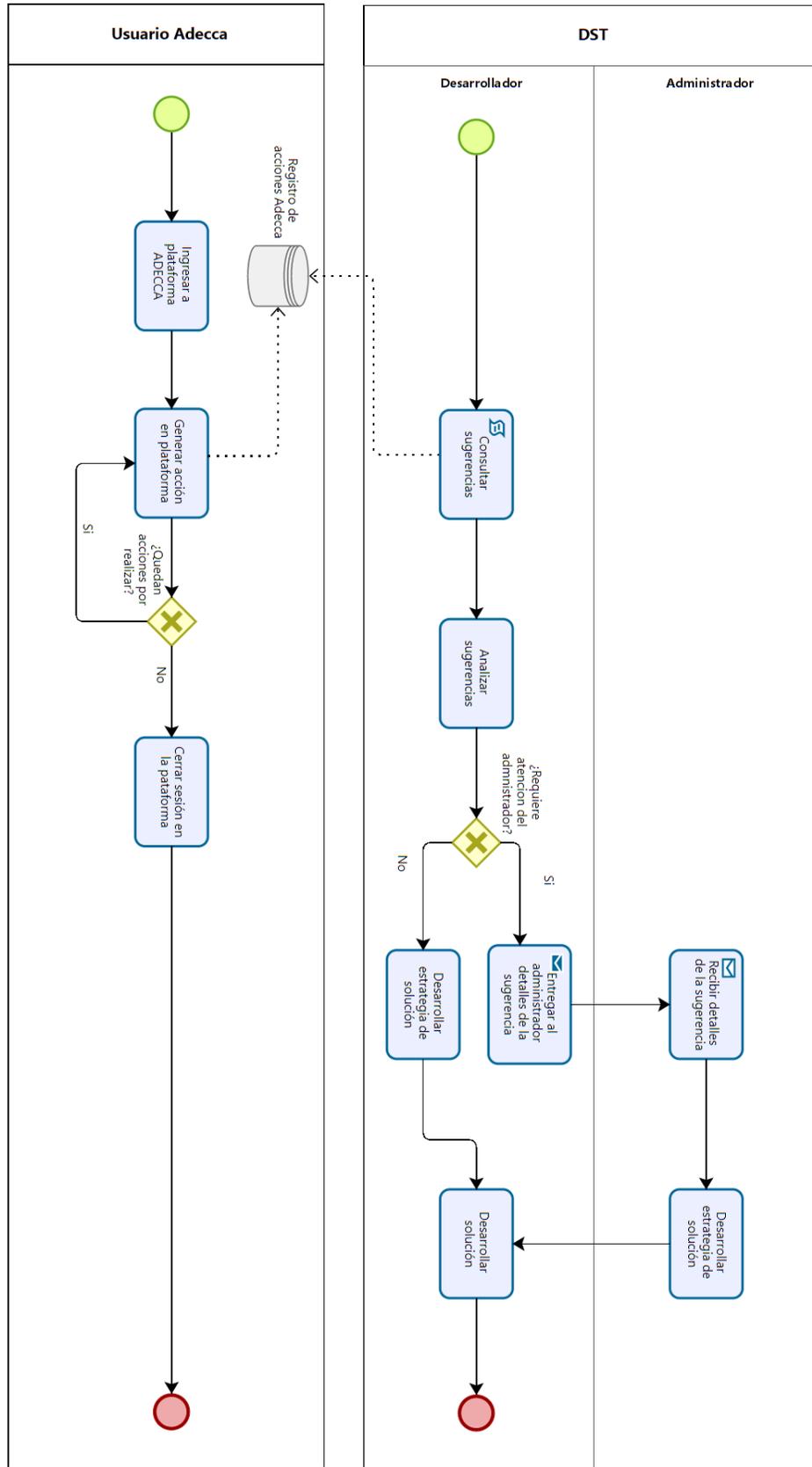


Figura 5: BPMN Sistema de Monitoreo Adecca

6.2 Diagrama de casos de uso

En esta sección se realiza el análisis de los actores y la interacción en la que tiene cada uno de estos con los casos de uso del sistema mostrado en la **Tabla 8** y **Tabla 9**.

6.2.1 Actores

Actor	Administrador	Identificador: Actor01
Descripción	Es un usuario registrado del sistema que administra la plataforma Adecca y el sistema de monitoreo.	
Características	El Administrador tiene acceso a todas las funcionalidades, además, gestiona a los usuarios y crear nuevos usuarios en el sistema.	
Relación	El Administrador se relaciona directamente con el Desarrollador.	
Referencias	Casos de uso en los que interviene: CU01, CU02, CU03, CU04, CU05, CU06, CU07, CU08, CU09, CU10, CU11, CU12, CU13, CU14, CU15, CU16, CU17, CU18, CU19, CU20.	

Tabla 8: Especificación del Actor Administrador.

Actor	Desarrollador	Identificador: Actor02
Descripción	Es un usuario registrado del sistema que corresponde a Desarrollador de la plataforma Adecca.	
Características	Desempeña todas las funcionalidades de la plataforma, pero no administrar ni crea usuarios.	
Relación	El Desarrollador se relaciona directamente con el Administrador.	
Referencias	Casos de uso en los que interviene: CU01, CU02, CU03, CU04, CU05, CU06, CU07, CU08, CU13, CU14, CU15, CU16, CU17, CU18, CU19, CU20.	

Tabla 9: Especificación del Actor Desarrollador.

6.2.2 Casos de Uso y descripción

El la **Figura 6** se muestra los casos de uso que contempla el sistema de monitoreo Adecca. Este contiene 20 casos de uso que hacen referencia a los requisitos funcionales del sistema. Además, existen dos actores que representan al Administrador y Desarrollador del sistema y que también comparte el mismo rol en la plataforma Adecca.

Las funcionalidades del Desarrollador y Administrador son compartidas en muchos de los casos de uso, ya que, todos tendrán accesos a las funcionalidades principales del sistema como lo sería visualizar las estadísticas o autenticarse en el sistema.

Por otra parte, está el actor que cumple el rol de Administrador, es el único actor que tiene la posibilidad de gestionar los usuarios que ocuparán el sistema, además de asignar los privilegios correspondientes al uso del sistema de monitoreo.

Adicionalmente, para especificar, el caso de uso que contiene la palabra “Gestionar”, hacen referencia al grupo de funcionalidades como lo son Crear, Modificar, Eliminar y Listar, o también conocidos como CRUD.



Figura 6: Diagrama de Casos de Uso

6.2.3 Especificación de los Casos de Uso

La especificación de los casos de uso del sistema se mostrará desde la **Tabla 10** a la **Tabla 29**. En cada una de estas se realiza una descripción de un caso de uso que está directamente asociado a un requerimiento funcional, y se indica el flujo principal y los actores que interactúan con el caso de uso.

6.2.3.1 Caso de Uso: <Autenticarse>

Caso de uso: Autenticarse	
ID	CU01.
Descripción	Ingresar al sistema por medio de credenciales de usuario y contraseña.
Actores	Desarrollador, Administrador.
Pre-Condiciones	Estar registrado como usuario.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor debe ingresar sus credenciales (Rut y contraseña). 2. El actor debe definir si el sistema recuerde su sesión. 3. El actor debe presionar ingresar. 4. El sistema redirige a la ventana de Dashboard de la sección de estadísticas.
Flujo de Eventos Alternativos	<ol style="list-style-type: none"> 2.a Si el actor decide recordar sus credenciales, estos datos se guardan en el almacenamiento local del navegador. 3.a Si el actor no ingresa las credenciales correctas, el sistema emite una alerta con una ventana emergente con el mensaje "datos incorrectos".
Post-Condiciones	Se genera un token con la información cifrada del usuario.

Tabla 10: Especificación de los casos de uso: Autenticarse.

6.2.3.2 Caso de Uso: <Seleccionar Alertas >

Caso de uso: Seleccionar Alertas	
ID	CU02.
Descripción	Seleccionar las alertas del Sistema Adecca.
Actores	Desarrollador, Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor ingresa a la sección de “Alertas” desde el menú. 2. En el listado de las alertas el usuario escoge la opción de errores. 3. El sistema despliega una ventana emergente con un mensaje “Estamos cargando los datos, espere por favor.”. 4. El sistema despliega alertas de base de datos por defecto.
Flujo de Eventos Alternativos	No existe Flujo Alternativo.
Post-Condiciones	No existe Post-condiciones.

Tabla 11: Especificación de los casos de uso: Seleccionar Alertas.

6.2.3.3 Caso de Uso: <Seleccionar Sugerencias>

Caso de uso: Seleccionar Sugerencias	
ID	CU03.
Descripción	Seleccionar las sugerencias de la plataforma Adecca.
Actores	Desarrollador, Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor ingresa a la sección de “Sugerencias” desde la opción menú. 2. En el listado de las sugerencias el usuario escoge una sugerencia. 3. El sistema visualiza la información de la sugerencia escogida.
Flujo de Eventos Alternativos	No existe Flujo Alternativo.
Post-Condiciones	No existe Post-condiciones.

Tabla 12: Especificación de los casos de uso: Seleccionar Sugerencias.

6.2.3.4 Caso de Uso: <Configurar Sugerencia>

Caso de uso: Configurar Sugerencia	
ID	CU04.
Descripción	Configuración porcentajes de tolerancia para sugerencias emitidas por el sistema.
Actores	Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor ingresa a la sección de “Configuración” desde el menú. 2. El sistema muestra el listado de configuraciones. 3. El actor presiona el botón “Editar”. 4. El actor ingresa los porcentajes de tolerancia de utilización estándar de memoria y la utilización máxima de memoria y tiempo de ejecución. 5. El actor presiona el botón “guardar”.
Flujo de Eventos Alternativos	5.a. Si el actor no ingresa parámetros en los campos, el sistema muestra una alerta en una ventana emergente para ingresar los parámetros solicitados.
Post-Condiciones	Almacenar variables de tolerancia máxima y estándar en la base de datos.

Tabla 13: Especificación de los casos de uso: Configurar Sugerencia

6.2.3.5 Caso de Uso: <Ver Estadísticas Memoria>

Caso de uso: Ver Estadísticas de Memoria	
ID	CU05.
Descripción	Visualizar las estadísticas de uso de memoria que utiliza la plataforma Adecca.
Actores	Desarrollador, Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor debe ingresar a la sección de estadísticas. 2. En la lista desplegada el actor debe seleccionar memoria. 3. El sistema despliega las estadísticas de la utilización de la memoria de la plataforma Adecca.
Flujo de Eventos Alternativos	No existe flujos alternativos.
Post-Condiciones	No existe Post-Condiciones

Tabla 14: Especificación de los casos de uso: Ver Estadísticas de Memoria.

6.2.3.6 Caso de Uso: <Ver Estadísticas Navegadores>

Caso de uso: Ver Estadísticas de Navegadores	
ID	CU06.
Descripción	Visualizar las estadísticas de uso de navegadores que utiliza la plataforma Adecca.
Actores	Desarrollador, Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor debe ingresar a la sección de estadísticas. 2. En la lista desplegada el actor debe seleccionar Navegadores. 3. El sistema despliega las estadísticas de la utilización de los navegadores de la plataforma Adecca.
Flujo de Eventos Alternativos	No existe flujos alternativos.
Post-Condiciones	No existe Post-Condiciones.

Tabla 15: Especificación de los casos de uso: Ver Estadísticas de Navegadores.

6.2.3.7 Caso de Uso: <Ver Estadísticas Tiempo Ejecución>

Caso de uso: Ver Estadísticas de Tiempo de Ejecución	
ID	CU07.
Descripción	Visualizar las estadísticas de tiempo de ejecución que utiliza la plataforma Adecca.
Actores	Desarrollador, Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor debe ingresar a la sección de “estadísticas”. 2. En la lista desplegada, el actor debe seleccionar “tiempo de ejecución”. 3. El sistema despliega las estadísticas del tiempo de ejecución de la plataforma.
Flujo de Eventos Alternativos	No existe flujos alternativos.
Post-Condiciones	No existe Post-Condiciones.

Tabla 16: Especificación de los casos de uso: Ver Estadísticas de Tiempo De Ejecución.

6.2.3.8 Caso de Uso: <Ver Dashboard>

Caso de uso: Ver Dashboard	
ID	CU08.
Descripción	Visualizar las estadísticas de tiempo de ejecución que utiliza la plataforma Adecca.
Actores	Desarrollador, Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor debe ingresar a la sección de “estadísticas”. 2. En la lista desplegada el actor debe seleccionar “Dashboard”. 3. El sistema despliega las estadísticas relevantes para la plataforma Adecca a tiempo real.
Flujo de Eventos Alternativos	No existe flujos alternativos.
Post-Condiciones	No existe Post-Condiciones.

Tabla 17: Especificación de los casos de uso: Ver DashBoard.

6.2.3.9 Caso de Uso: <Asignar Privilegios>

Caso de uso: Asignar Privilegios	
ID	CU09.
Descripción	Asignar privilegios a un usuario en específico.
Actores	Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor ingresa a la sección de “Configuración”. 2. El sistema muestra el listado de Configuraciones. 3. El actor presiona el botón “Usuario”. 4. El actor escoge el usuario al cual asignar privilegios. 5. El sistema muestra el privilegio actual del usuario escogido. 6. El actor selecciona nuevo privilegio del usuario. 7. El sistema muestra un mensaje de confirmación con los datos del usuario a asignar el privilegio. 8. El sistema muestra un mensaje de asignación exitosa.
Flujo de Eventos Alternativos	7.a. En caso de que el actor cancele la asignación del usuario, la ventana emergente se cierra.
Post-Condiciones	Modificación en la base de datos del privilegio del usuario escogido.

Tabla 18: Especificación de los casos de uso: Asignar Privilegios.

6.2.3.10 Caso de Uso: <Crear Usuario>

Caso de uso: Crear Usuario	
ID	CU10.
Descripción	Crear un usuario.
Actores	Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor ingresa a la sección de “Configuración”. 2. El sistema muestra el listado de Configuraciones. 3. El actor presiona el botón “Usuario”. 4. El actor presiona el botón “Crear usuario”. 5. El sistema despliega los campos a llenar. 6. El actor debe llenar los campos. 7. El actor debe presionar el botón guardar usuario. 8. El sistema despliega un mensaje de usuario creado con éxito. 9. El sistema redirecciona al listado de usuarios.
Flujo de Eventos Alternativos	<ol style="list-style-type: none"> 7.a. Si el actor no ingresa caracteres o solo ingresa espacios en blanco, el sistema debe mostrar un mensaje con el nombre del campo que falta llenar.
Post-Condiciones	Se registra en la base de datos el nuevo usuario.

Tabla 19: Especificación de los casos de uso: Crear Usuario.

6.2.3.11 Caso de Uso: <Modificar Usuario>

Caso de uso: Modificar Usuario	
ID	CU11.
Descripción	Modificar un Usuario existente.
Actores	Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor ingresa a la sección de “Configuración”. 2. El sistema muestra el listado de Configuraciones. 3. El actor presiona el botón “Usuario”. 4. El sistema despliega una tabla con los usuarios creados. 5. El actor debe escoger un usuario y presionar el botón de “editar”. 6. El sistema llena los campos con la información del usuario escogido. 7. El actor debe cambiar los campos que requiera modificar. 8. El actor debe presionar guardar usuario. 9. El sistema despliega un mensaje de usuario editado con éxito. 10. El sistema redirecciona al listado de usuarios.
Flujo de Eventos Alternativos	8.a. Si el actor no ingresa caracteres o solo ingresa espacios en blanco, el sistema debe mostrar un mensaje con el nombre del campo que falta.
Post-Condiciones	Se realiza la actualización del usuario en la base de datos.

Tabla 20: Especificación de los casos de uso: Modificar Usuario.

6.2.3.12 Caso de Uso: <Eliminar Usuario>

Caso de uso: Eliminar Usuario	
ID	CU12.
Descripción	Eliminar un Usuario existente.
Actores	Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor ingresa a la sección de “Configuración”. 2. El sistema muestra el listado de Configuraciones. 3. El actor presiona el botón “Usuario”. 4. El sistema despliega una tabla con los usuarios creados. 5. El actor debe escoger un usuario y presionar el botón de “eliminar”. 6. El sistema despliega una ventana emergente con la información del usuario confirmando si queremos eliminarlo. 7. El actor debe confirmar la eliminación. 8. El sistema despliega un mensaje de usuario eliminado con éxito.
Flujo de Eventos Alternativos	7.a. En caso de que el actor cancele la eliminación del usuario, la ventana emergente se cierra.
Post-Condiciones	Se elimina el registro del usuario en la base de datos.

Tabla 21: Especificación de los casos de uso: Eliminar Usuario.

6.2.3.13 Caso de Uso: <Ver Sugerencia Tiempo Ejecución>

Caso de uso: Ver Sugerencias Tiempo Ejecución	
ID	CU13.
Descripción	Visualizar sugerencias sobre tiempo de ejecución, en base a los porcentajes de tolerancia ya configurado.
Actores	Desarrollador, Administrador.
Pre-Condiciones	<ol style="list-style-type: none"> 1. Estar autenticado en el sistema. 2. Haber configurado los porcentajes de tolerancia de las sugerencias.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor ingresa a la sección de “Sugerencias”. 2. El sistema muestra el listado de Sugerencias. 3. El actor presiona la opción “Tiempo de Ejecución”. 4. El sistema despliega una ventana emergente con un mensaje “Estamos cargando los datos, espere por favor.” 5. El sistema despliega en casillas seleccionables las opciones de sugerencias de “Controller”, “Method” y “URI”. 6. El sistema despliega el listado de sugerencias de “Controller” por defecto.
Flujo de Eventos Alternativos	<ol style="list-style-type: none"> 4.a. En el caso de que los datos no carguen, el sistema desplegará un mensaje “Error, no hemos podido cargar los datos”. 5.a. En el caso de que el actor seleccione la opción Method, el sistema despliega el listado de sugerencias de los métodos de Adecca. 5.b. En el caso de que el actor seleccione la opción URI, el sistema despliega el listado de sugerencias del identificador de recursos uniforme de Adecca.
Post-Condiciones	No existe Post-Condiciones.

Tabla 22: Especificación de los casos de uso: Ver Sugerencia Tiempo de Ejecución.

6.2.3.14 Caso de Uso: <Ver Sugerencia Memoria Usada>

Caso de uso: Ver Sugerencias Memoria Usada	
ID	CU14.
Descripción	Visualizar sugerencias sobre memoria usada, en base a los porcentajes de tolerancia ya configurado.
Actores	Desarrollador, Administrador.
Pre-Condiciones	<ol style="list-style-type: none"> 1. Estar autenticado en el sistema. 2. Haber configurado los porcentajes de tolerancia de las sugerencias.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor ingresa a la sección de “Sugerencias”. 2. El sistema muestra el listado de Sugerencias. 3. El actor presiona la opción “Memoria Usada”. 4. El sistema despliega una ventana emergente con un mensaje “Estamos cargando los datos, espere por favor.” 5. El sistema despliega en casillas seleccionables las opciones de sugerencias de “Controller”, “Method” y “URI”. 6. El sistema despliega el listado de sugerencias de “Controller” por defecto.
Flujo de Eventos Alternativos	<ol style="list-style-type: none"> 4.a. En el caso de que los datos no carguen, el sistema desplegará un mensaje “Error, no hemos podido cargar los datos”. 5.a. En el caso de que el actor seleccione la opción Method, el sistema despliega el listado de sugerencias de los métodos de Adecca. 5.b. En el caso de que el actor seleccione la opción URI, el sistema despliega el listado de sugerencias del identificador de recursos uniforme de Adecca.
Post-Condiciones	No existe Post-Condiciones.

Tabla 23: Especificación de los casos de uso: Ver Sugerencia de Memoria Usada.

6.2.3.15 Caso de Uso: <Ver Sugerencias Memoria Maxima>

Caso de uso: Ver Sugerencias Memoria Máxima	
ID	CU15.
Descripción	Visualizar sugerencias sobre memoria máxima, en base a los porcentajes de tolerancia ya configurado.
Actores	Desarrollador, Administrador.
Pre-Condiciones	<ol style="list-style-type: none"> 1. Estar autenticado en el sistema. 2. Haber configurado los porcentajes de tolerancia de las sugerencias.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor ingresa a la sección de “Sugerencias”. 2. El sistema muestra el listado de Sugerencias. 3. El actor presiona la opción “Memoria Máxima”. 4. El sistema despliega una ventana emergente con un mensaje “Estamos cargando los datos, espere por favor.” 5. El sistema despliega en casillas seleccionables las opciones de sugerencias de “Controller”, “Method” y “URI”. 6. El sistema despliega el listado de sugerencias de “Controller” por defecto.
Flujo de Eventos Alternativos	<ol style="list-style-type: none"> 4.a. En el caso de que los datos no carguen, el sistema desplegará un mensaje “Error, no hemos podido cargar los datos”. 5.a. En el caso de que el actor seleccione la opción Method, el sistema despliega el listado de sugerencias de los métodos de Adecca. 5.b. En el caso de que el actor seleccione la opción URI, el sistema despliega el listado de sugerencias del identificador de recursos uniforme de Adecca.
Post-Condiciones	No existe Post-Condiciones.

Tabla 24: Especificación de los casos de uso: Ver Sugerencias de Memoria Máxima.

6.2.3.16 Caso de Uso: <Ver Alertas Errores PHP>

Caso de uso: Ver Alertas de Errores PHP	
ID	CU16.
Descripción	Visualizar alertas sobre errores de ejecución de líneas de programación.
Actores	Desarrollador, Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor ingresa a la sección de “Alertas”. 2. El actor presiona la opción “Errores”. 3. El sistema despliega una ventana emergente con un mensaje “Estamos cargando los datos, espere por favor.”. 4. El sistema despliega en casillas seleccionables las opciones de sugerencias de “Base de datos” y “PHP”. 5. El sistema despliega los errores de base de datos por defecto. 6. El actor debe seleccionar PHP. 7. El sistema despliega el listado de errores de PHP.
Flujo de Eventos Alternativos	No existe flujos alternativos.
Post-Condiciones	No existe Post-Condiciones.

Tabla 25: Especificación de los casos de uso: Ver Alerta de Errores PHP.

6.2.3.17 Caso de Uso: <Ver Alertas Errores de base de datos>

Caso de uso: Ver Alertas de Errores de Base de Datos	
ID	CU17.
Descripción	Visualizar alertas sobre errores de ejecución de consultas de base de datos.
Actores	Desarrollador, Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor ingresa a la sección de “Alertas”. 2. El actor presiona la opción “Errores”. 3. El sistema despliega una ventana emergente con un mensaje “Estamos cargando los datos, espere por favor.”. 4. El sistema despliega en casillas seleccionables las opciones de sugerencias de “Base de datos” y “PHP”. 5. El actor debe seleccionar “Base de datos”. 6. El sistema despliega el listado de errores de Base de datos.
Flujo de Eventos Alternativos	No existe flujos alternativos.
Post-Condiciones	No existe Post-Condiciones.

Tabla 26: Especificación de caso de uso: Ver Alertas de Errores de Base de Datos

6.2.3.18 Caso de Uso: <Buscar alerta por mensaje de error>

Caso de uso: Buscar alerta por mensaje de error	
ID	CU18.
Descripción	Buscar reporte de error por un mensaje especificado.
Actores	Desarrollador, Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor ingresa a la sección de “Alertas”. 2. El actor presiona la opción “Errores”. 3. El sistema despliega una ventana emergente con un mensaje “Estamos cargando los datos, espere por favor.”. 4. El sistema despliega en casillas seleccionables las opciones de sugerencias de “Base de datos” y “PHP”. 5. El actor debe seleccionar “Base de datos”. 6. El actor debe ingresar el mensaje que desea buscar. 7. El sistema despliega una ventana emergente con un mensaje “Estamos buscando tu alerta”. 8. El sistema despliega el listado de alerta correspondiente al mensaje buscado.
Flujo de Eventos Alternativos	No existe flujos alternativos.
Post-Condiciones	No existe Post-Condiciones.

Tabla 27: Buscar alerta por mensaje de error.

6.2.3.19 Caso de Uso: <Ver Notificación>

Caso de uso: Ver Notificación	
ID	CU19.
Descripción	Visualizar notificaciones sobre las alertas que detecta el sistema.
Actores	Desarrollador, Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor se ubica en la cabecera de la plataforma. 2. El actor presiona el ícono ubicado en la cabecera de la página. 3. El sistema despliega una lista de notificaciones del sistema. 4. El actor selecciona una notificación. 5. El sistema despliega una ventana emergente con el detalle de la notificación.
Flujo de Eventos Alternativos	No existe flujos alternativos.
Post-Condiciones	No existe Post-Condiciones.

Tabla 28: Especificación de los casos de uso: Ver Notificación.

6.2.3.20 Caso de Uso: <Ignorar Notificación>

Caso de uso: Ignorar Notificación	
ID	CU20.
Descripción	Ignorar una notificación descubierta por el sistema.
Actores	Desarrollador, Administrador.
Pre-Condiciones	Estar autenticado en el sistema.
Flujo de Eventos Básicos	<ol style="list-style-type: none"> 1. El actor se ubica en la cabecera de la plataforma. 2. El actor presiona el ícono de campana. 3. El sistema despliega una lista de notificaciones del sistema. 4. El actor selecciona una notificación. 5. El sistema despliega una ventana emergente con el detalle de la notificación. 6. El actor presiona el botón ignorar. 7. El sistema registra la notificación como “ignorada”
Flujo de Eventos Alternativos	No existe flujos alternativos.
Post-Condiciones	Se registra la notificación en la base de datos como ignorada además de la fecha en la cual se ignora.

Tabla 29: Especificación de los casos de uso: Ignorar Notificación.

6.3 Modelamiento de datos

En esta sección se presenta el modelo de datos que ha sido creado, el que está relacionado con el Sistema de monitoreo y alertas tempranas de errores para la plataforma educativa Adecca UBB.

Para el desarrollo del sistema se utiliza un modelo no relacional, por lo cual, este modelo no tiene relaciones ni tampoco entidades presentes, difiriendo del tradicional modelo entidad relación. En consecuencia, existen colecciones de documentos representando las entidades y clave-valor de los documentos para la representación de los atributos. No existe una definición teórica de modelo que ilustra una base de datos no relacional, sin embargo, (Hernández Chillón, 2017), aporta con una propuesta de visualización de esquema para un modelo de datos no relacional utilizando árboles cuyos nodos hoja son valores atómicos (Number, String, Boolean, etc.) y en la raíz se encuentra el nombre de la Entidad o Colección. Por lo tanto, la estructura de un esquema sería la misma que la de los objetos de esa Entidad.

Hemos usado el termino de Entidad para referirnos a todos los objetos almacenados que representan la información de una misma entidad.

Posteriormente, se realiza una breve descripción de las colecciones de documentos, para mayor entendimiento del lector.

A continuación, se describen las colecciones representadas en el Modelo de datos no relacional.

- **log_system:** En esta colección, **Figura 10**, se representa los logs o registros que se almacenarán del sistema Adecca UBB y que corresponden a los múltiples eventos generados.
- **configuracion_sugerencias:** En esta colección, **Figura 7**, se registra la información para la generación de sugerencias.
- **users:** En esta colección, **Figura 12**, se almacena la información del ente o usuario que utiliza el sistema, es decir, su información personal y credenciales.
- **Alertas_notificaciones:** En esta colección, **Figura 8Figura 11**, se registra la información que se visualiza en las notificaciones del sistema.

Posteriormente, se muestra los esquemas de modelo de datos no relacional del sistema.

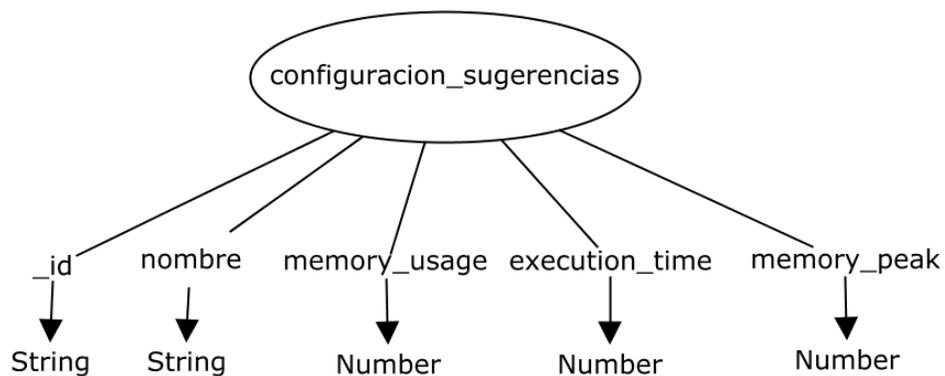


Figura 7: Esquema de documento: Configuración de Sugerencias.

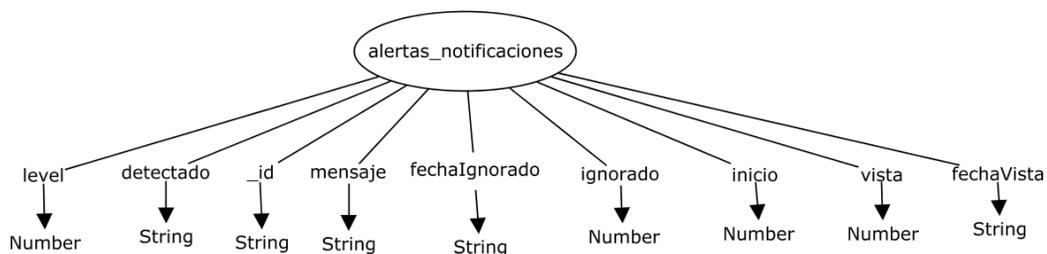


Figura 8: Esquema de documento: Alertas Notificaciones.

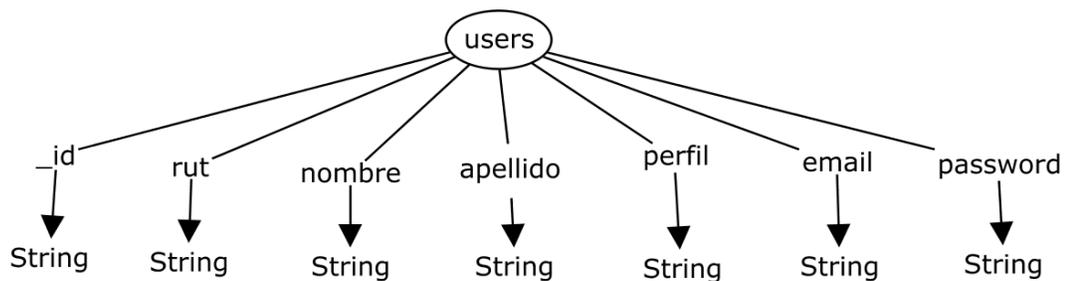


Figura 9: Esquema de documento: Users.

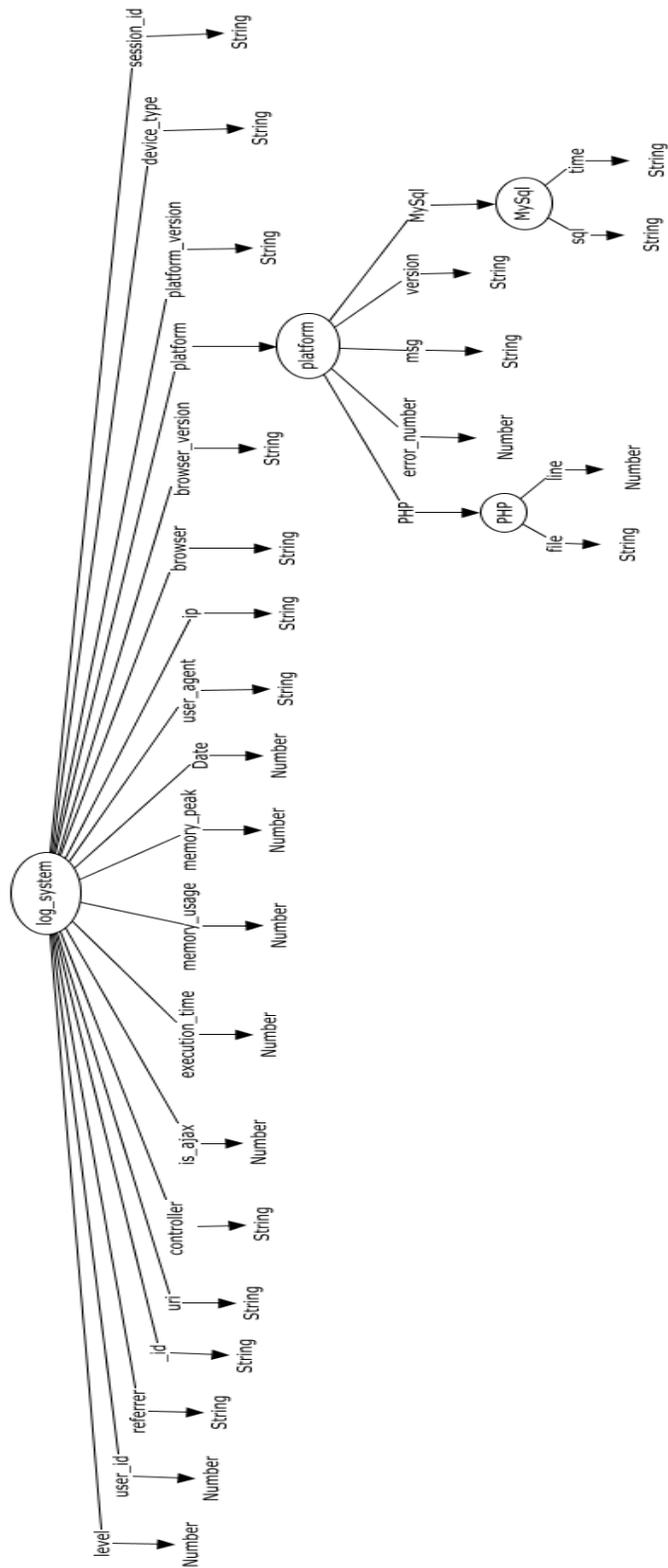


Figura 10: Esquema de datos: `log_system`.

7 DISEÑO

Este capítulo se destaca el Diseño Físico de la Base de datos, Diseño de Arquitectura Funcional, Diseño de Interfaz de Software y Especificación de Módulos.

7.1 Diseño Físico de la Base de datos

En las **Figura 11**, **Figura 12**, **Figura 13** y **Figura 14** se muestran cómo se registran los datos en cada uno de los documentos de las colecciones, cabe recordar que las bases de datos no relacionales para guardar sus documentos ocupan el formato JSON (notación de objeto de JavaScript).

```
{
  "_id": "String",
  "mensaje": "String",
  "level": "Number",
  "detectado": "String",
  "ignorado": "Number",
  "inicio": "Number",
  "vista": "Number",
  "fechaVista": "String",
  "fechaIgnorado": "String"
}
```

Figura 11: Formato de colección: alerta_notificaciones.

```
{
  "_id": "String",
  "nombre": "String",
  "apellido": "String",
  "rut": "String",
  "perfil": "String",
  "email": "String",
  "password": "String"
}
```

Figura 12: Formato de colección: users

```

{
  "_id": "String",
  "nombre": "String",
  "memory_usage": "Number",
  "execution_time": "Number",
  "memory_peak": "Number"
}

```

Figura 13: Formato de colección: configuración_sugerencias

```

{
  "_id": "String",
  "uri": "String",
  "controller": "String",
  "method": "String",
  "is_ajax": "Number",
  "execution_time": "Number",
  "memory_usage": "Number",
  "memory_peak": "Number",
  "user_agent": "String",
  "ip": "String",
  "browser": "String",
  "browser_version": "String",
  "platform": "String",
  "platform_version": "String",
  "device_type": "String",
  "referrer": "String",
  "date": "Number",
  "session_id": "String",
  "user_id": "Number",
  "level": "Number"
}

```

Figura 14: Formato de colección: log_system

7.2 Diseño de arquitectura funcional

Para este sistema se utilizará una arquitectura basada en componentes y se apoya con la tecnología REDIS.

La utilización de Angular en el desarrollo del sistema se selecciona para mitigar la sobrecarga de peticiones del servidor, debido a que su arquitectura de diseño está basada en componentes.

La arquitectura basada en componente, es un acercamiento a la ingeniería de software al diseño y desarrollo de sistemas. Esta arquitectura se enfoca en la descomposición del diseño en componentes funcionales o lógicos que tengan comunicaciones bien definidas. Además, es una abstracción mayor a los principios de orientación a objetos, ya que no se enfoca en los protocolos de comunicación y la forma de cómo comparte el estado. (Pelaez, 2009)

Esta arquitectura tiene las siguientes características:

- Un estilo de diseño para aplicaciones compuestas de componentes individuales.
- Pone énfasis en la descomposición del sistema en componentes funcionales o lógicos que tiene interfases bien definidas.
- Define una aproximación de diseño que usa componentes discretos, los que se comunican a través de interfaces que contienen métodos, eventos y propiedades.

Principios Fundamentales:

Un componente es un objeto de software específicamente diseñado para cumplir cierto propósito, los cuales deben ser:

- **Re-usable:** Los componentes son usualmente utilizados en distintos escenarios para distintas aplicaciones, pero también, pueden ser definidos para tareas específicas.
- **Sin contexto específico:** Tienen que operar para diferentes ambientes o contextos. La información específica como, los estados de los datos deben ser pasadas al componente en vez de incluir o permitir al componente acceder a ellos.
- **Extensibles:** Un componente debe ser extensible desde un componente existente a un nuevo componente.
- **Encapsulado:** Los componentes exponen interfaces que permiten al programar su funcionalidad, sin revelar los detalles internos.
- **Independiente:** Los componentes están diseñados para tener una dependencia mínima de otros componentes, por cual pueden ser instalados en otros ambientes sin afectar a otros componentes. (Pelaez, 2009)

Beneficios:

Los principales beneficios de arquitectura basados en componentes son los siguientes:

- **Facilidad de instalación:** Reemplazo de la versión, por una que esté disponible, sin afectar a otros componentes.
- **Costos reducidos:** El uso de componentes permite distribución de costo de desarrollo y mantenimiento.
- **Facilidad de desarrollo:** Los componentes implementan una interface bien definida, permitiendo el desarrollo sin impactar otras partes del sistema.
- **Re-usable:** Se utiliza para distribuir el desarrollo y el mantenimiento entre múltiples aplicaciones y sistemas.
- **Mitigación de complejidad técnica:** El uso de contenedores de componentes y sus servicios generan una baja complejidad de uso, estos incluyen activación de componentes, gestión de la vida de los componentes y la cola de mensajes del componente y transacciones. (Pelaez, 2009)

Debido a que el sistema en una semana carga 1.7 millones de datos, se ve la necesidad de agregar a la arquitectura funcional REDIS. Esta tecnología tiene una dualidad de función, ya que puede ser configurado como cola para el envío de peticiones asíncronas y síncronas, además de almacenamiento en memoria principal, la cual, para este sistema permite almacenar y recuperar datos en memoria principal evitando la sobrecarga innecesaria de peticiones de datos a la base de datos y generando una respuesta en menor tiempo, logrando obtener un mejoramiento en la optimización de uso de recursos del servidor.

Adicionalmente, se destaca el uso de Laravel, que se utiliza para crear la API HTTP que hace el papel de intermediario entre los servicios, Angular y la base de datos MongoDB.

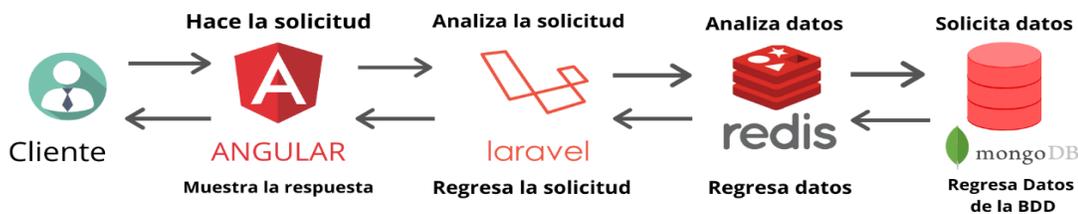


Figura 15: Diseño de la arquitectura funcional.

Fuente: Elaboración propia.

7.3 Diseño interfaz y navegación

El diseño de la interfaz y navegación del sistema, presentado a continuación, se realiza con el fin de ayudar al usuario a navegar de una manera práctica, haciendo énfasis en la visualización de los datos.

Cabe destacar que, para el diseño de la interfaz, tiene como formato una cabecera visualizada en la **Figura 16** y una barra lateral vista en la **Figura 17**, el que contiene el menú de opciones de la aplicación.

La cabecera muestra al inicio el logo de la plataforma y el menú de navegación, además, contiene el botón con la funcionalidad de extender a pantalla completa, el nombre de la aplicación, el ícono de la campana que simboliza las notificaciones de alertas y la sección de perfil de usuario.



Figura 16: Vista de Cabecera.

La barra lateral contiene el menú de navegación del sistema.

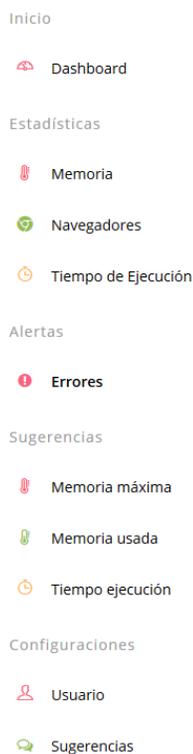


Figura 17: Vista menú lateral.

7.3.1 Interfaz ingreso al sistema.

Para la utilización del sistema se requiere una autenticación de usuario referido a la **Figura 18**; donde, en la parte superior se encuentra el ícono de Adecca y el nombre del sistema, más abajo, para el ingreso al sistema, el usuario tiene que ingresar el RUN y su contraseña correspondiente, posterior a esto existe una casilla seleccionable que permite al usuario, si quiere, que el sistema gestione sus credenciales y el botón ingresar.



Figura 18: Vista Autenticación de Usuario.

7.3.2 Interfaz de configuración del usuario.

Esta interfaz permite administrar los usuarios que pueden utilizar el sistema, en donde se puede observar en la **Figura 19** tiene un título, en el extremo izquierdo con un recuadro con el número 1 un botón para crear usuarios, debajo de este, una tabla con el listado de los usuarios registrado, el cual permite dos acciones que están marcadas por el recuadro 2 y 3, el recuadro 2 permite editar un usuario y el recuadro 3 permite eliminar un usuario.

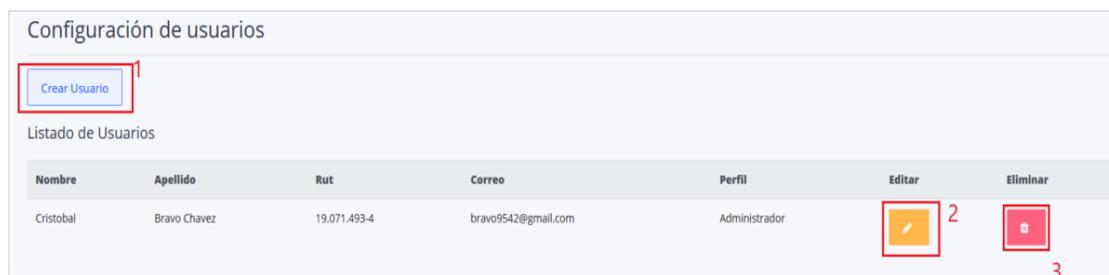


Figura 19: Vista listado de usuario.

Una vez presionado el botón de crear usuario este redirecciona a la pantalla de creación **Figura 20**, donde tiene al inicio, el título de la sección, debajo de él un botón para volver a la vista de la **Figura 19**, luego están las casillas de ingreso de datos vacíos para el ingreso de los datos del usuario a registrar, estos son: nombre, apellido, email y Rut, además, una casilla de tipo selección para escoger que tipo de rol tiene el usuario en la plataforma, finalizando con el botón de guardar, para almacenar el usuario ingresado.

Crear Usuario

Volver

Nombre : Ingrese solo su nombre

Apellido : Ingrese solo su apellido

Email : Email@email.com

Rut : 11111111-1

Perfil : Desarrollador

Guardar

Figura 20: Vista creación de usuario.

Como se mencionaba en la **Figura 19**, existe el botón de editar usuario donde este tiene la función de redireccionar a la interfaz de edición de usuario, esta interfaz, **Figura 21**, contiene el mismo formulario de la **Figura 20**, pero con la diferencia en el título y casilla de ingreso de datos, que están llenos con los datos del usuario escogido.

Editar Usuario

Volver

Nombre : Cristobal

Apellido : Bravo Chavez

Email : bravo9542@gmail.com

Rut : 19.071.493-4

Perfil : Administrador

Guardar

Figura 21: Vista edición de usuario.

En el último botón eliminar de la **Figura 19**, tiene como funcionalidad eliminar el usuario seleccionado, al momento de presionar el botón emerge una ventana emergente, el que tiene un título con la información del usuario, el ícono de peligro y al final dos botones, estos botones confirman la eliminación del usuario o cancela la eliminación, esto esta implementado para evitar las eliminaciones accidentales de los usuarios (**Figura 22**).



Figura 22: Vista mensaje de seguridad de eliminación de usuario

7.3.2 Interfaz de configuración de las sugerencias.

Esta interfaz tiene como finalidad la configuración de sugerencias, **Figura 23**, donde está distribuida con el título de la interfaz, luego con un subtítulo y un listado de 3 opciones a configurar marcado en el recuadro1, este tiene por objetivo cambiar la información contenida en la tabla y permitir editar esa información dentro de las opciones, posterior a este, existe una tabla que contiene los datos de la configuración actual en base a la opción escogida del recuadro1, donde los datos están definidos en porcentajes, finalizando con el botón de editar marcado en el recuadro2.



Figura 23: Vista de configuración de sugerencias.

En el último botón mencionado en la **Figura 23**, es el botón de editar el cual tiene como funcionalidad editar la configuración escogida, al momento de presionar el botón se despliegan 3 campos de ingreso de datos, **Figura 24**, donde estos datos serán los necesarios para gestionar la generación de sugerencias, finalizando con un botón de guardar que almacena la información de la configuración, desplegando una ventana emergente con un mensaje de éxito al guardar la configuración de sugerencia.

Figura 24: Vista de edición de configuración de sugerencia.

Una vez que se presione el botón de guardar se despliega una ventana emergente, **Figura 25**, que contiene, el ícono de éxito, el título y un botón para informar al usuario que su información ha sido guardado con éxito.

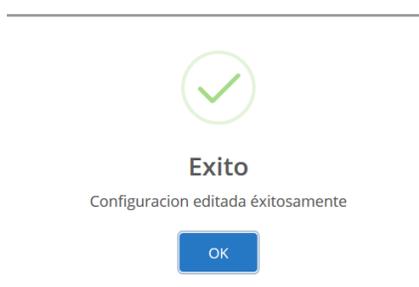


Figura 25: Vista de venta emergente de confirmación de edición de configuración.

7.4 Especificación de módulos

En esta sección se presenta la especificación de los módulos del sistema.

N° Módulo: Mod01		Nombre Módulo: Ingreso de sistema	
Parámetros de entrada		Parámetros de Salida	
Nombre:	Tipo de dato:	Nombre:	Tipo de dato:
Rut	String	Sin parámetro de salida	Ninguno
Contraseña	String		

Tabla 30: Especificación módulo ingreso de sistema

N° Módulo: Mod02		Nombre Módulo: Crear Usuario	
Parámetros de entrada		Parámetros de Salida	
Nombre:	Tipo de dato:	Nombre:	Tipo de dato:
Nombre	String	Mensaje de resultado	String
Apellido	String		
Email	String		
Rut	String		
Perfil	String		

Tabla 31: Especificación módulo Crear Usuario.

N° Módulo: Mod03		Nombre Módulo: Editar Usuario	
Parámetros de entrada		Parámetros de Salida	
Nombre:	Tipo de dato:	Nombre:	Tipo de dato:
Id	int	Mensaje de resultado	String
Nombre	String		
Apellido	String		
Email	String		
Rut	String		
Perfil	String		

Tabla 32: Especificación módulo Editar Usuario

N° Módulo: Mod04		Nombre Módulo: Eliminar Usuario	
Parámetros de entrada		Parámetros de Salida	
Nombre:	Tipo de dato:	Nombre:	Tipo de dato:
Id	int	Mensaje de resultado	String

Tabla 33: Especificación módulo Eliminar Usuario

N° Módulo: Mod05		Nombre Módulo: Editar Configuración	
Parámetros de entrada		Parámetros de Salida	
Nombre:	Tipo de dato:	Nombre:	Tipo de dato:
Id	int	Mensaje de resultado	String
Tiempo máximo de ejecución	int		
Memoria Máxima	int		
Memoria usada	int		

Tabla 34: Especificación módulo Editar Configuración.

8 PRUEBAS

A continuación, se da a conocer en esta sección, el resultado obtenido de las pruebas aplicadas al software con el objetivo de encontrar posibles errores que se han pasado por alto en la codificación y comprobar el cumplimiento de los requisitos. El tipo de pruebas realizadas son de caja negra.

8.1 Elementos de prueba

Los elementos de prueba son todos los módulos del software. Para una mejor organización en la ejecución de las pruebas, los módulos fueron agrupados por similitud de concepto (Estadísticas, Sugerencias, etc.).

8.1.1 Módulo Estadísticas.

El módulo de estadísticas contiene las funcionalidades de entregar informaciones estadísticas en base a los datos capturados de la utilización de la plataforma Adecca.

8.1.2 Módulo Configuración Sugerencias.,

El módulo de configuración de sugerencias es aquel que contiene las funcionalidades del Administrador para configurar y administrar la tolerancia y, generar las sugerencias de mejora a la plataforma Adecca.

8.1.3 Módulo Configuración Usuarios.

El módulo de configuración de usuarios es aquel que permite al Administrador del sistema crear, editar, eliminar un usuario del sistema.

8.1.4 Módulo Ingreso de usuario.

El módulo que permite el uso del sistema según el rol que contenga el usuario.

8.1.5 Módulo Sugerencias.

El módulo de sugerencias es aquel que genera las recomendaciones de mejora a la plataforma Adecca, estas son generadas en base a la configuración de sugerencias y los datos capturados de la utilización de la plataforma.

8.1.6 Módulo Alertas.

Este módulo contiene las funcionalidades para las generaciones de las alertas, estas son generadas por los datos capturados de la utilización de la plataforma Adecca.

8.2 Especificación de las pruebas

Especificación de prueba para Ingreso de Sistema		Actividades de prueba
Características a probar	Funcionalidad y usabilidad.	<ol style="list-style-type: none"> 1. Ubicarse en el login de la plataforma. 2. Ingresar credenciales. 3. Verificar si se muestran mensajes de éxito o error.
Nivel de prueba	Unidad	
Objetivo de la prueba	Que se cumplan los requerimientos y encontrar error de funcionamiento.	
Enfoque para la definición de casos de prueba	Caja Negra	
Técnicas para la definición de casos de prueba	Valores límites y partición equivalente.	
Criterios de cumplimiento	Los valores obtenidos deben cumplir con los requerimientos.	

Tabla 35: Especificación de prueba: Ingreso de Sistema.

Especificación de prueba para Estadísticas		Actividades de prueba
Características a probar	Funcionalidad y usabilidad.	<ol style="list-style-type: none"> 1. Ingresar el sistema como administrador. 2. Ingresar a la sección estadísticas. 3. Verificar si se muestran mensajes de éxito o error.
Nivel de prueba	Unidad	
Objetivo de la prueba	Que se cumplan los requerimientos y encontrar error de funcionamiento.	
Enfoque para la definición de casos de prueba	Caja Negra	
Técnicas para la definición de casos de prueba	Valores límites y partición equivalente.	
Criterios de cumplimiento	Los valores obtenidos deben cumplir con los requerimientos.	

Tabla 36: Especificación de prueba: Estadísticas.

Especificación de prueba para Sugerencias		Actividades de prueba
Características a probar	Funcionalidad y usabilidad.	<ol style="list-style-type: none"> 1. Ingresar el sistema como administrador. 2. Ingresar a la sección sugerencias. 3. Seleccionar la opción de sugerencias. 4. Verificar si se muestran mensajes de éxito o error.
Nivel de prueba	Unidad	
Objetivo de la prueba	Que se cumplan los requerimientos y encontrar error de funcionamiento.	
Enfoque para la definición de casos de prueba	Caja Negra	
Técnicas para la definición de casos de prueba	Valores límites y partición equivalente.	
Criterios de cumplimiento	Los valores obtenidos deben cumplir con los requerimientos.	

Tabla 37: Especificación de prueba: Sugerencias

Especificación de prueba para Alertas		Actividades de prueba
Características a probar	Funcionalidad y usabilidad.	<ol style="list-style-type: none"> 1. Ingresar el sistema como administrador. 2. Ingresar a la sección Alertas. 3. Verificar si se muestran mensajes de éxito o error.
Nivel de prueba	Unidad	
Objetivo de la prueba	Que se cumplan los requerimientos y encontrar error de funcionamiento.	
Enfoque para la definición de casos de prueba	Caja Negra	
Técnicas para la definición de casos de prueba	Valores límites y partición equivalente.	
Criterios de cumplimiento	Los valores obtenidos deben cumplir con los requerimientos.	

Tabla 38: Especificación de prueba: Alertas.

Especificación de prueba para Configuración de Sugerencias		Actividades de prueba
Características a probar	Funcionalidad y usabilidad.	<ol style="list-style-type: none"> 1. Ingresar al sistema como Administrador. 2. Ingresar a la sección de configurar sugerencias. 3. Editar datos de la configuración de sugerencias. 4. Verificar si se muestran mensajes de éxito o error.
Nivel de prueba	Unidad	
Objetivo de la prueba	Que se cumplan los requerimientos y encontrar error de funcionamiento.	
Enfoque para la definición de casos de prueba	Caja Negra.	
Técnicas para la definición de casos de prueba	Valores límites y partición equivalente.	
Criterios de cumplimiento	Los valores obtenidos deben cumplir con los requerimientos.	

Tabla 39: Especificación de prueba: Configuración de Sugerencias.

Especificación de prueba para Configuración de Usuario		Actividades de prueba
Características a probar	Funcionalidad y usabilidad.	<ol style="list-style-type: none"> 1. Ingresar al sistema como Administrador. 2. Ingresar a la sección de configurar usuarios. 3. Crear un usuario. 4. Editar un usuario. 5. Eliminar un usuario. 6. Verificar si se muestran mensajes de éxito o error.
Nivel de prueba	Unidad	
Objetivo de la prueba	Que se cumplan los requerimientos y encontrar error de funcionamiento.	
Enfoque para la definición de casos de prueba	Caja Negra.	
Técnicas para la definición de casos de prueba	Valores límites y partición equivalente.	
Criterios de cumplimiento	Los valores obtenidos deben cumplir con los requerimientos.	

Tabla 40: Especificación de prueba: Configuración de Usuario

8.3 Responsables de las pruebas

Las pruebas de caja negra fueron realizadas por Cristóbal Bravo Chávez, quien es estudiante memorista y desarrollador del sistema.

Por otra parte, las pruebas de usabilidad fueron aplicadas a una persona externa al desarrollo del proyecto, quién es el actual Desarrollador de la plataforma Adecca, el profesional Carlos San Juan.

8.4 Calendario de pruebas

Durante el desarrollo del software se realizaron las pruebas correspondientes a las funcionalidades designadas a la metodología Iterativo Incremental, es decir, en cada incremento se comprueba el funcionamiento, para así asegurar la calidad del software durante el periodo que fue asignado por la metodología. Este calendario difiere de la planificación inicial (**véase en ANEXO: PLANIFICACION INICIAL DEL PROYECTO**), debido al aumento de funcionalidades que se termina desarrollando.

Sistema de monitoreo y alertas temprana de errores Adecca UBB																																				
N°	Nombre	Duración	Inicio	Terminado	12-oct-20					16-nov-20					23-nov-20				28-dic-20																	
					L	M	M	J	V	L	M	M	J	V	L	M	M	J	V	L	M	M	J	V												
1	Modulo Estadísticas	2 días	14-10-20 13:00	15-10-2020 17:00																																
2	Modulo Ingreso de Sistema	2 días	16-11-20 13:00	17-11-2020 17:00																																
3	Modulo Sugerencias	2 días	18-11-20 13:00	19-11-2020 17:00																																
4	Modulo Configuración de usuario	2 días	20-11-20 13:00	23-11-2020 17:00																																
5	Modulo Configuración de sugerencias	2 días	24-11-20 13:00	25-11-2020 17:00																																
6	Modulo Alertas	2 días	28-12-20 13:00	29-12-2020 17:00																																

Figura 26: Calendario de pruebas

8.5 Detalle de las pruebas

El detalle de las pruebas se encuentra en el **ANEXO: ESPECIFICACIÓN DE LAS PRUEBAS**. El resumen de la ejecución de estas pruebas se da a conocer en la **Tabla 41**.

Módulo de Prueba	Número de Pruebas realizadas	Cantidad de Errores encontrados
Módulo Ingreso de Sistema.	4	0
Módulo Estadísticas.	4	0
Módulo Sugerencias.	9	6
Módulo Alertas.	5	1
Módulo Configuración de usuario	5	1
Módulo Configuración de sugerencias	3	1
Total	30	9

Tabla 41: Resumen Pruebas

De acuerdo a los resultados obtenidos en la **Tabla 41**, se hallaron 9 errores del total de 30 pruebas realizadas. Se destaca que los errores encontrados fueron corregidos.

8.5.1 Verificación de la calidad del producto

Los aspectos de calidad a verificar son los especificados en la sección 4.4.4. El propósito de la verificación es el aseguramiento de calidad del software y el cumplimiento de los requisitos no funcionales solicitados por el cliente. El resultado de todas las encuestas se encuentra en el **ANEXO: RESULTADO VERIFICACIÓN ATRIBUTOS DEL SOFTWARE.**

A través de un cuestionario se recogió la evaluación de los usuarios entrevistados:

- Cliente.
- Desarrollador de Adecca.
- Administrador de Adecca.

La moda de las respuestas obtenidas se presenta en la siguiente tabla.

Encuesta			
Característica	Pregunta	Evaluación	Respuesta
Usabilidad	¿El software genera para cada una de las operaciones realizadas un mensaje que indique el éxito o fracaso de éstas, para que el usuario pueda interpretar claramente lo realizado?	Bueno	X
		Regular	
		Malo	
	¿El tamaño de fuente es el ideal para leer la información que entrega sistema?	Bueno	X
		Regular	
		Malo	
¿Se entiende el contenido de los datos solicitados en los formularios?	Bueno	X	
	Regular		
	Malo		
Seguridad	Nivel de protección sobre los datos sensibles del usuario.	Bueno	X
		Regular	
		Malo	
Interfaz Gráfica	¿Encuentra visualmente amigable el sistema?	Bueno	X
		Regular	
		Malo	
	Nivel de representatividad de los íconos e imágenes que aparecen en la interfaz del sistema	Bueno	X
		Regular	
		Malo	
Nivel de representatividad que tienen los nombres de los botones	Bueno	X	
	Regular		
	Malo		

Tabla 42: Resumen Encuesta verificación de la calidad del producto

8.6 Conclusiones de Prueba

Se realizaron pruebas de caja negra para comprobar el correcto funcionamiento de los diferentes módulos del sistema. Además, se logra estimar el nivel de usabilidad del sistema según las pruebas aplicadas a usuarios externos.

9 PLAN DE CAPACITACIÓN Y ENTRENAMIENTO

El plan de capacitación y entrenamiento consta de dos clases y será para el Desarrollador y el Administrador de Adecca, para que conozcan e interactúen con la interfaz y las funcionalidades del sistema.

9.1 Usuarios a capacitar

Los usuarios que reciben capacitación del sistema corresponden a dos de los tipos de usuario del sistema, el Administrador y Desarrollador de la plataforma Adecca.

9.2 Tipo de capacitación o entrenamiento

La capacitación se realiza como un taller con clases presenciales o de forma semipresencial (virtual a través de video conferencias), dependiendo de las condiciones sanitarias existentes, para los dos tipos de usuarios mencionados (Administrador y Desarrollador) y se lleva a cabo, mediante el siguiente procedimiento:

- Primero, se presenta el funcionamiento general del sistema, para mostrar la navegabilidad de opciones y usabilidad general del sistema.
- Después se solicita al Desarrollador que accese al sistema con sus credenciales correspondientes y lo utilice, con el propósito de realizar consultas al encargado de la capacitación. Además, se entrega un manual de usabilidad general del sistema.
- Finalmente, se le solicita que genere las acciones correspondientes en el sistema, pero ahora sin el acompañamiento del guía tutor.

9.3 Funcionalidades o aspectos que serán abordados

Los primeros puntos a abordar son la presentación y navegabilidad del sistema. El siguiente aspecto son las funciones que pueda realizar el Administrador o Desarrollador de la plataforma. Estas funcionalidades son:

- Gestión de un Usuario (Crear, editar y eliminar).
- Gestión de configuración de sugerencias (Editar).
- Visualización, configuración y aplicación de los filtros en sección de las alertas.
- Visualización, configuración y aplicación de los filtros en sección de las sugerencias.
- Ingreso de sistema.
- Visualización de Estadísticas.

9.4 Responsable de la capacitación

El encargado de realizar la capacitación es el desarrollador del sistema Cristóbal Bravo, ya que tiene el conocimiento y manejo sobre el funcionamiento del sistema.

9.5 Tiempo estimado

Como fue antes mencionado en la Factibilidad económica, se realizan dos capacitaciones; una por día. La primera capacitación tiene una duración aproximada de 4 horas junto al Administrador y Desarrollador de la plataforma Adecca, la segunda capacitación tiene una duración estimada a la anterior.

9.6 Recursos requeridos

Los recursos requeridos para la capacitación son un computador o notebook con conexión a internet, un espacio físico que tenga mesas y sillas, además de un proyector para visualizar el uso del software. Un ejemplo de espacio físico son los laboratorios centrales de la Universidad del Bío-Bío.

En el caso de que la capacitación sea de manera virtual, los recursos requeridos son un computador o notebook con conexión a internet para la ejecución de una video conferencia.

10 RESUMEN ESFUERZO REQUERIDO

El esfuerzo del desarrollador para llevar a cabo este proyecto fue de 667 horas en total, esto se divide en 2 periodos. En el primer periodo se trabaja desde el 24 de agosto de 2020 hasta el 08 de octubre del mismo año trabajando 5 días a la semana 8 horas diarias. Luego, un segundo periodo, debido al ingreso de clases, se resuelve trabajar 4 días a la semana 8 horas diarias desde el 12 de octubre hasta el 31 de diciembre de 2020. En la **Tabla 43** se presenta el resumen de la carga horaria del esfuerzo requerido.

Actividades/fases	N° Horas
Requerimientos	45
Análisis y diseño	62
Desarrollo	403
Pruebas	60
Capacitación	8
Documentación	89
Total	667

Tabla 43: Esfuerzo Requerido

11 CONCLUSIONES

Alcanzado este punto y a través de las diferentes páginas de este informe, ha sido posible evidenciar todo el proceso de desarrollo de software desde sus etapas más básicas, como es la captura de requerimientos hasta culminar con la implementación de un producto final de vanguardia, siendo este último, un reflejo del conocimiento adquirido durante todos estos años en la Universidad y por supuesto, de un constante proceso de investigación y aprendizaje autónomo.

Para dar cumplimiento a los objetivos específicos, es posible destacar el desarrollo de una API para capturar y almacenar, en una base de datos no relacional (MongoDB), los diferentes eventos generados por los usuarios de la plataforma educativa Adecca. Dicha información permite ser analizada para desplegar diferentes estadísticas, alertas y sugerencias las cuales fueron realizadas de manera efectiva, con la utilización de Framework Laravel. En este punto se hace hincapié en el uso de la tecnología Redis para lograr una ganancia evidente en los tiempos de respuesta del sistema, mejorando la experiencia de uso del usuario final.

La visualización de las alertas, sugerencias y estadísticas forman parte esencial del software, con una interfaz gráfica comprensible y amigable al usuario que fue implementada con la tecnología del Framework Angular. La información entregada por las diferentes secciones del sistema permite cumplir con el objetivo de mejorar la toma de decisiones en la plataforma, optimizando los recursos.

Concluido el proyecto, es posible destacar la completitud de los objetivos, los cuales eran diseñar un sistema de monitoreo y alertas de errores tempranas para la plataforma educativa Adecca de la Universidad del Bío-Bío con tecnologías renovadas, eficiente y escalable en el tiempo.

Trabajos Futuros

El sistema de monitoreo es una aplicación renovada, eficiente, con múltiples funcionalidades, pero también escalable, por cual, no está exenta de mejoras, ratificando nuevas sugerencias de parte de los clientes luego de las reuniones realizadas. Estas posibles implementaciones son:

- Exportación de datos en Excel: permitirá generar un registro físico e histórico de los datos del sistema.
- Notificaciones personalizadas: generará la posibilidad de establecer filtros o configuraciones de las notificaciones según lo prefiera el usuario de la plataforma.
- Filtrado de fechas dinámicos: mejorará la visualización de datos según el rango de fecha requerido por el usuario
- Inteligencia artificial para sugerencias: Actualmente los rangos de tolerancia son establecidos por el usuario, por lo que, al implementar una inteligencia artificial, permitiría generar sugerencias más específicas en base a los datos estudiados.

Bibliografía

Angular. (s.f.). Obtenido de <https://angular.io/docs>

Arias, E. (19 de Febrero de 2020). *Factibilidad técnica*. Obtenido de Economipedia.

aws. (s.f.). Obtenido de <https://aws.amazon.com/es/nosql/>

Cadavid, N., Juan, F., & Jonathan, M. (2013). Revisión de metodologías ágiles para el desarrollo de software. *PROSPECTIVA*, 32-35.

GIT SCM. (s.f.). Obtenido de <https://git-scm.com/>

H.S., C. (28 de Julio de 2019). *Factibilidad económica: qué es y cómo se hace*. Obtenido de Liferder: <https://www.liferder.com/factibilidad-economica/>

Hernández Chillón, A. (2017). Visualización de Esquemas en Bases de Datos NoSQL basadas en documentos. *jornadas de Ingeniería del Software y Bases de Datos JSBD*, (pág. 4 5 6). La Laguna.

Laravel. (s.f.). Obtenido de <https://laravel.com/>

MDN Contributor. (20 de Mayo de 2019). *WebSockets - Referencia de la API Web | MDN*. Obtenido de Developer.mozilla.org: https://developer.mozilla.org/es/docs/Web/API/WebSockets_API

MDN web docs. (26 de julio de 2020). Obtenido de <https://developer.mozilla.org/es/docs/Web/JavaScript>

mongoDB. (s.f.). Obtenido de <https://www.mongodb.com/es>

Mozilla and individual contributors. (25 de julio de 2019). *MDN web docs*. Obtenido de CSS3: <https://developer.mozilla.org/es/docs/Archive/CSS3#:~:text=CSS3%20es%20la%20%C3%BAltima%20evoluci%C3%B3n,1>.

Pelaez, J. C. (18 de Abril de 2009). Obtenido de Blog de Juan Peláez en Geeks.ms: <https://geeks.ms/jkpelaez/2009/04/18/arquitectura-basada-en-componentes/>

PHP. (s.f.). Obtenido de <https://www.php.net/manual/es/intro-what-is.php>

RabbitMQ. (2 de mayo de 2020). Obtenido de Messaging that just works — RabbitMQ: <https://www.rabbitmq.com/>

Redislabs. (23 de Mayo de 2019). *Redis*. Obtenido de Redis.io: <https://redis.io/>

The Apache Software Foundation. (s.f.). *Apache Jmeter™*. Obtenido de <https://jmeter.apache.org/usermanual/index.html>

TypeScript. (2020). Obtenido de <https://www.typescriptlang.org/>

Universidad del Bio Bio. (s.f.). Obtenido de Universidad del Bío-Bío, única universidad pública de las regiones del Biobío y Ñuble. Acreditada por la CNA.: http://www.ubiobio.cl/w/#Vision_y_Mision

Universidad Del Bio-Bio. (06 de 03 de 2017). Obtenido de Inducción UBB: <https://induccionptubb.wordpress.com/laboratorio-de-computacion-lc/>

Universidad del Bío-Bío. (06 de Marzo de 2017). *Laboratorio de computación*. Obtenido de <https://induccionptubb.wordpress.com/laboratorio-de-computacion-lc/>

Universidad del Bío-Bío. (2018). *ESTUDIO DE VINCULACIÓN ENTRE EL PREGRADO Y POSTGRADO DE LA UNIVERSIDAD DEL BÍO-BÍO*.

Visual Studio Code. (s.f.). Obtenido de <https://code.visualstudio.com/docs>

ANEXO: PLANIFICACION INICIAL DEL PROYECTO

En esta sección se muestra la planificación inicial del proyecto que fue estimada al comenzar el desarrollo del sistema.

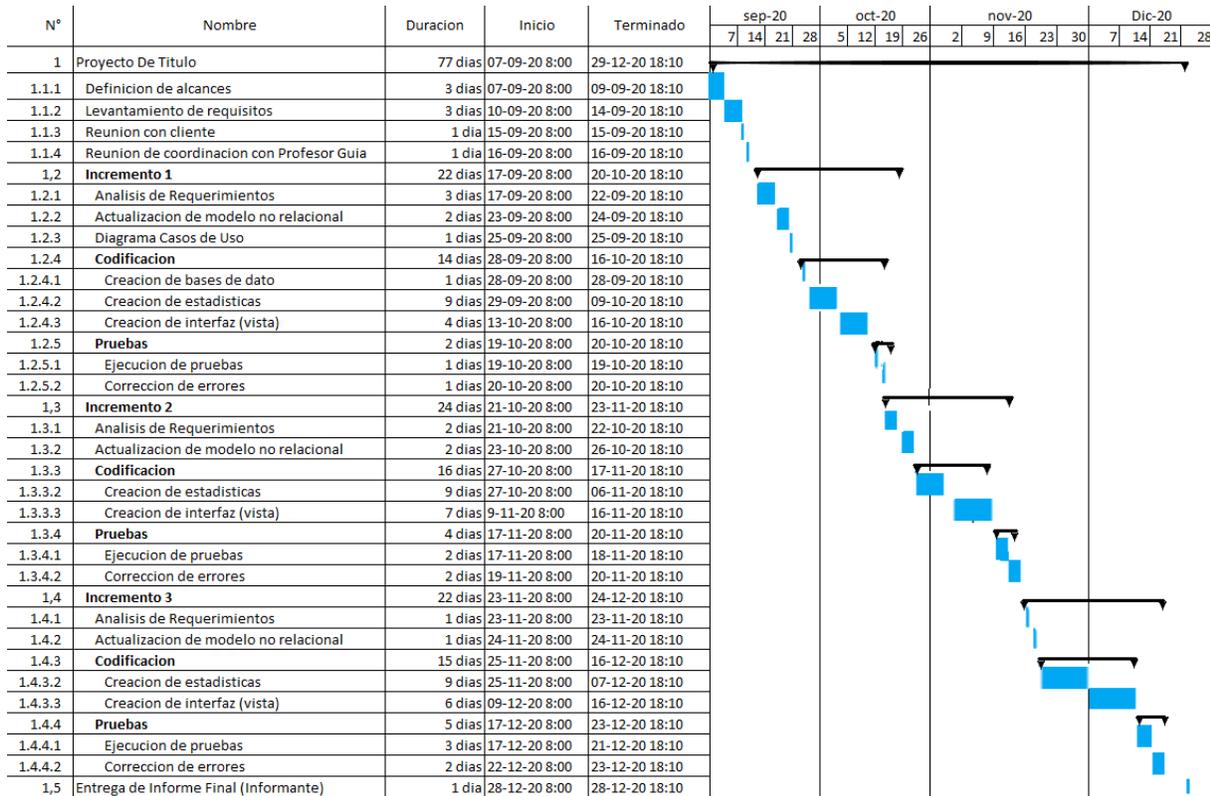


Figura 27: Carta Gantt

ANEXO: RESULTADOS DE ITERACIONES EN EL DESARROLLO

En esta sección se detalla los resultados de las iteraciones definidas para el proyecto, identificando las funcionalidades desarrolladas en cada iteración, este proyecto está constituido en 3 iteraciones, recordar que para este proyecto se utilizó la metodología Iterativo Incremental definido en **3.2 Ambiente de Ingeniería de Software**.

Primer Incremento

Tabla 44, da a conocer el listado de funcionalidades desarrolladas en el primer incremento.

Funcionalidades
Estudiantes conectados en la plataforma.
Usuarios Online.
Dispositivos usados por los usuarios.
Tops usuarios que más se conectan.
Top páginas más visitadas.
Top Navegadores usados por los usuarios.
Versión de Navegadores más utilizado.
Sistema Operativos Utilizados.
URI que utilizan más memoria en promedio.
Controladores que utilizan más memoria en promedio.
Métodos que utilizan más memoria en promedio.

Top versiones más utilizadas ascendentes y descendentes según navegador.
Uri que utilizan más tiempo de ejecución en promedio.
Controladores que utilizan más tiempo de ejecución en promedio.
Métodos que utilizan más tiempo de ejecución en promedio.

Tabla 44: Funcionalidades Primer Incremento

En este incremento se desarrollaron las funcionalidades correspondientes al módulo de Dashboard y el módulo de Estadísticas, además, en este incremento fue en el cual hubo mayor investigación sobre las tecnologías que había que implementar incluyendo el cambio de paradigma de realización de consultas con la base de datos no relacional.

Segundo Incremento

En **Tabla 45** , se visualizarán el listado de funcionalidades desarrolladas en el segundo incremento.

Funcionalidades
Creación de tolerancias para la configuración de las sugerencias.
Edición de tolerancias para configuración de las sugerencias.
Listar usuarios.
Crear usuarios.
Editar usuarios.
Eliminar usuarios.
Autenticación de usuario.

Cambio de contraseña inicial.
Cerrado de sesión.
Obtener sugerencias de memoria máxima según la tolerancia configurada por controlador.
Obtener sugerencias de memoria máxima según la tolerancia configurada por métodos.
Obtener sugerencias de memoria máxima según la tolerancia configurada por URI.
Obtener sugerencias de memoria usada según la tolerancia configurada por controlador
Obtener sugerencias de memoria usada según la tolerancia configurada por métodos.
Obtener sugerencias de memoria usada según la tolerancia configurada por URI.
Obtener sugerencias de tiempo de ejecución según la tolerancia configurada por controlador.
Obtener sugerencias de tiempo de ejecución según la tolerancia configurada por métodos.
Obtener sugerencias de memoria usada según la tolerancia configurada por URI.

Tabla 45: Funcionalidades Segundo Incremento

En este incremento se desarrollaron las funcionalidades correspondientes al módulo de Sugerencias, configuración y autenticación de usuarios para el sistema, se destaca la realización de configuración de las sugerencias ya que, gracias a las tolerancias registradas, permitirán al usuario un mayor uso de recurso de la plataforma Adecca.

Tercer Incremento

En **Tabla 46**, se visualizarán el listado de funcionalidades desarrolladas en el tercer y último incremento.

Funcionalidades
Obtener alertas de base de datos.
Obtener alertas de PHP.
Top errores por plataforma (base de datos y PHP).
Top 5 código de errores.
Obtener el listado de notificaciones.
Marcar como vistas las notificaciones visualizadas por el usuario.
Ver detalle de notificación.
Ignorar notificación.

Tabla 46: Funcionalidades Tercer Incremento

En este incremento es donde se desarrollaron menor cantidad de funcionalidades, ya que la realización de notificación tuvo mayor complejidad para desarrollar ya que estas tienen que ser entregadas en tiempo real, estas funcionalidades son correspondiente al módulo de alertas y notificaciones del sistema.

ANEXO: ESPECIFICACION DE LAS PRUEBAS

En el siguiente anexo se da a conocer los detalles de las pruebas funcionales realizadas en el sistema, estas serán separadas según el módulo que corresponda.

Pruebas de Módulo Configuración de usuario

Caso de Prueba: CP-01 Crear Usuario				
ID Prueba	CP-01			
Propósito	Creación de un nuevo usuario en el sistema			
Enfoque	Caja Negra			
Caso de Uso	Gestionar Usuarios	Actores	Administrador	
Prueba N	Valores de Prueba	Resultado esperado	Resultado obtenido	Evaluación
Caso N°1	- Nombre: Juan - Apellido: Perez - Email: pruebas@gmail.com - Rut: 190714934 - Perfil: Desarrollador	<ul style="list-style-type: none"> Se muestra el siguiente mensaje: "Juan Perez se creó correctamente". El usuario es listado. 	Juan Perez se creó correctamente	Éxito

Tabla 47: Caso de prueba 1: Crear Usuario

Caso de Prueba: CP-02 Eliminar Usuario				
ID Prueba	CP-02			
Propósito	Eliminación de un usuario existente en el sistema			
Enfoque	Caja Negra			
Caso de Uso	Gestionar Usuarios	Actores	Administrador	
Prueba N	Valores de Prueba	Resultado esperado	Resultado obtenido	Evaluación
Caso N°1	- Usuario Seleccionable: Sí.	<ul style="list-style-type: none"> El usuario ha sido eliminado. 	El usuario ha sido eliminado.	Éxito
Caso N°2	- Usuario Seleccionable: No.	<ul style="list-style-type: none"> El usuario no ha sido eliminado. 	El usuario no ha sido eliminado.	Éxito

Tabla 48: Caso de prueba 2: Eliminar Usuario

Caso de Prueba: CP-03 Editar Usuario				
ID Prueba	CP-03			
Propósito	Edición de un usuario existente en el sistema			
Enfoque	Caja Negra			
Caso de Uso	Gestionar Usuarios	Actores	Administrador	
Prueba N	Valores de Prueba	Resultado esperado	Resultado obtenido	Evaluación
Caso N°1	- Nombre: Juan - Apellido: Perez - Email: pruebas1@gmail.com - Rut: 190714934 - Perfil: Administrador	<ul style="list-style-type: none"> • Se muestra el siguiente mensaje: "Juan Perez se actualizó correctamente". • El usuario es listado con sus atributos actualizados. 	No se actualizó el correo electrónico del usuario.	Fracaso
Caso N°2	- Nombre: Juan - Apellido: Perez - Email: pruebas2@gmail.com - Rut: 190714934 - Perfil: Administrador	<ul style="list-style-type: none"> • Se muestra el siguiente mensaje: "Juan Perez se actualizó correctamente". • El usuario es listado con sus atributos actualizados. 	Juan Perez se actualizó correctamente.	Éxito.

Tabla 49: Caso de prueba 3: Editar usuario

Prueba de módulo Configuración de Sugerencias.

Caso de Prueba: CP-04 Editar Configuración de Sugerencias				
ID Prueba	CP-04			
Propósito	Edición de valores de tolerancia para realización de sugerencias.			
Enfoque	Caja Negra			
Caso de Uso	Configurar Sugerencias	Actores	Administrador	
Prueba N	Valores de Prueba	Resultado esperado	Resultado obtenido	Evaluación
Caso N°1	- Opción: Method - Tiempo máximo de ejecución: vacío. - Memoria máxima: vacío. - Memoria Usada: vacío.	<ul style="list-style-type: none"> Se muestra el siguiente mensaje: "Error de Ingreso Los valores de memoria máxima, tiempo ejecución y memoria usada debe estar entre 0 y 1". 	Error de Ingreso Los valores de memoria máxima, tiempo ejecución, memoria usada debe estar entre 0 y 1	Éxito.
Caso N°2	- Opción: Method - Tiempo máximo de ejecución: a. - Memoria máxima: a. - Memoria Usada: a.	<ul style="list-style-type: none"> Se muestra el siguiente mensaje: "Error de Ingreso Los valores deben ser solo número". 	Configuración editada exitosamente	Fracaso.
Caso N°3	- Opción: Method - Tiempo máximo de ejecución: 0.2. - Memoria máxima: 0.3. - Memoria Usada: 0.4.	<ul style="list-style-type: none"> Configuración editada exitosamente. Listado de actualizado de configuración de sugerencias. 	Configuración editada exitosamente	Éxito.

Tabla 50: Caso de prueba 4: Editar Configuración de Sugerencias

Prueba módulo Ingreso de usuario.

Caso de Prueba: CP-05 Ingreso de usuario				
ID Prueba	CP-05			
Propósito	Ingresar al sistema con credenciales correspondiente al usuario.			
Enfoque	Caja Negra			
Caso de Uso	Autenticarse	Actores	Administrador, Desarrollador	
Prueba N	Valores de Prueba	Resultado esperado	Resultado obtenido	Evaluación
Caso N°1	- Rut: vacío. - Contraseña: contraseña.	<ul style="list-style-type: none"> Se muestra el siguiente mensaje: "Oops no ha ingresado el rut". 	Oops no ha ingresado el rut.	Éxito.
Caso N°2	- Rut: 190714934. - Contraseña: vacío.	<ul style="list-style-type: none"> Se muestra el siguiente mensaje: "Oops no ha ingresado la contraseña". 	Oops no ha ingresado la contraseña.	Éxito.
Caso N°3	- Rut: 190714934. - Contraseña: invalida.	<ul style="list-style-type: none"> Se muestra el siguiente mensaje: "Oops credenciales inválidas". 	Oops credenciales inválidas.	Éxito.
Caso N°4	- Rut: 190714934. - Contraseña: contraseña.	<ul style="list-style-type: none"> Se redirige a la vista de Dashboard. 	Redirección a vista Dashboard.	Éxito.

Tabla 51: Caso de prueba 5: Ingreso de usuario.

Prueba de módulo de Sugerencias

Caso de Prueba: CP-06 Sugerencias de Memoria Máxima.				
ID Prueba	CP-06			
Propósito	Verificación de la correcta visualización de datos			
Enfoque	Caja Negra			
Caso de Uso	Ver sugerencias de Memoria Máxima.	Actores	Administrador, Desarrollador	
Prueba N	Valores de Prueba	Resultado esperado	Resultado obtenido	Evaluación
Caso N°1	- Opción: Controller.	<ul style="list-style-type: none"> • Visualización del promedio de memoria máxima tolerado de controller, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada 	<ul style="list-style-type: none"> • Visualización del promedio de memoria máxima tolerado de controller, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada y la opción escogida. 	Éxito.
Caso N°2	- Opción: Method.	<ul style="list-style-type: none"> • Visualización del promedio de memoria máxima tolerado de Method, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada y la opción escogida. 	<ul style="list-style-type: none"> • Visualización del promedio de memoria máxima tolerado de controller, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada y la opción escogida. 	Fracaso.
Caso N°3	- Opción: URI.	<ul style="list-style-type: none"> • Visualización del promedio de memoria máxima tolerado de URI, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a 	<ul style="list-style-type: none"> • Visualización del promedio de memoria máxima tolerado de controller, con el porcentaje de memoria máxima configurado. • Listado de 	Fracaso.

		la tolerancia configurada y la opción escogida.	sugerencias correspondiente a la tolerancia configurada y la opción escogida.	
--	--	---	---	--

Tabla 52: Caso de prueba 6: Sugerencias Memoria Máxima

Caso de Prueba: CP-07 Sugerencias de Tiempo de Ejecución.				
ID Prueba	CP-07			
Propósito	Verificación de la correcta visualización de datos			
Enfoque	Caja Negra			
Caso de Uso	Ver sugerencias de Tiempo de Ejecución.	Actores	Administrador, Desarrollador	
Prueba N	Valores de Prueba	Resultado esperado	Resultado obtenido	Evaluación
Caso N°1	- Opción: Controller.	<ul style="list-style-type: none"> • Visualización del promedio de tiempo de ejecución tolerado de controller, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada 	<ul style="list-style-type: none"> • Visualización del promedio de tiempo de ejecución tolerado de controller, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada y la opción escogida. 	Éxito.
Caso N°2	- Opción: Method.	<ul style="list-style-type: none"> • Visualización del promedio de tiempo de ejecución tolerado de Method, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada y la opción escogida. 	<ul style="list-style-type: none"> • Visualización del promedio de tiempo de ejecución tolerado de controller, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada y la opción escogida. 	Fracaso.

Caso N°3	- Opción: URI.	<ul style="list-style-type: none"> • Visualización del promedio de tiempo de ejecución tolerado de URI, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada y la opción escogida. 	<ul style="list-style-type: none"> • Visualización del promedio de tiempo de ejecución tolerado de controller, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada y la opción escogida. 	Fracaso.
----------	----------------	---	--	----------

Tabla 53: Caso de Prueba 7: Sugerencias de Tiempo de Ejecución.

Caso de Prueba: CP-08 Sugerencias de Memoria Usada.				
ID Prueba	CP-08			
Propósito	Verificación de la correcta visualización de datos			
Enfoque	Caja Negra			
Caso de Uso	Ver sugerencias de Memoria Usada.	Actores	Administrador, Desarrollador	
Prueba N	Valores de Prueba	Resultado esperado	Resultado obtenido	Evaluación
Caso N°1	- Opción: Controller.	<ul style="list-style-type: none"> • Visualización del promedio de memoria Usada tolerado de controller, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada 	<ul style="list-style-type: none"> • Visualización del promedio de memoria Usada tolerado de controller, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada y la opción escogida. 	Éxito.

Caso N°2	- Opción: Method.	<ul style="list-style-type: none"> • Visualización del promedio de memoria Usada tolerado de Method, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada y la opción escogida. 	<ul style="list-style-type: none"> • Visualización del promedio de memoria Usada tolerado de controller, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada y la opción escogida. 	Fracaso.
Caso N°3	- Opción: URI.	<ul style="list-style-type: none"> • Visualización del promedio de memoria Usada tolerado de URI, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada y la opción escogida. 	<ul style="list-style-type: none"> • Visualización del promedio de memoria Usada tolerado de controller, con el porcentaje de memoria máxima configurado. • Listado de sugerencias correspondiente a la tolerancia configurada y la opción escogida. 	Fracaso.

Tabla 54: Caso de prueba 8: Sugerencias de Memoria Usada

Prueba de Modulo Alerta

Caso de Prueba: CP-09 Visualizar Alerta de Errores				
ID Prueba	CP-09			
Propósito	Verificación de la correcta visualización de las alertas.			
Enfoque	Caja Negra			
Caso de Uso	Seleccionar Alertas.	Actores	Administrador, Desarrollador	
Prueba N	Valores de Prueba	Resultado esperado	Resultado obtenido	Evaluación
Caso N°1	- Opción: "PHP".	<ul style="list-style-type: none"> • Visualización de errores de plataforma. • Visualización de grafico de código de error. • Visualización de reporte de alertas según la opción escogida. 	<ul style="list-style-type: none"> • Visualización de errores de plataforma. • Visualización de grafico de código de error. • Visualización de reporte de alertas según la opción escogida. 	Éxito.
Caso N°2	- Opción: "Base de datos".	<ul style="list-style-type: none"> • Visualización de errores de plataforma. • Visualización de grafico de código de error. • Visualización de reporte de alertas según la opción escogida. 	<ul style="list-style-type: none"> • Visualización de errores de plataforma. • Visualización de grafico de código de error. • Visualización de reporte de alertas según la opción escogida. 	Éxito.

Tabla 55: Caso de prueba 9: Visualizar Alerta de Errores.

Caso de Prueba: CP-10 Visualizar alerta en base a un mensaje.				
ID Prueba	CP-10			
Propósito	Verificación de la correcta visualización de las alertas según un mensaje a buscar.			
Enfoque	Caja Negra			
Caso de Uso	Buscar Alertas por mensaje de error.	Actores	Administrador, Desarrollador	
Prueba N	Valores de Prueba	Resultado esperado	Resultado obtenido	Evaluación
Caso N°1	- mensaje: "vacío".	<ul style="list-style-type: none"> • Visualización de mensaje "Error valor a buscar en blanco". 	<ul style="list-style-type: none"> • Visualización de errores de plataforma. 	Fracaso.
Caso N°2	- mensaje: "mensaje invalido".	<ul style="list-style-type: none"> • Visualización de mensaje "Error valor a buscar no existe". 	<ul style="list-style-type: none"> • Visualización de mensaje "Error valor a buscar no existe". 	Éxito.
Caso N°3	- mensaje: "mensaje invalido".	<ul style="list-style-type: none"> • Visualización de mensaje "Error valor a buscar no existe". 	<ul style="list-style-type: none"> • Visualización de mensaje "Error valor a buscar no existe". 	Éxito.

Tabla 56: Caso de prueba 10: Visualizar alerta en base de un mensaje

Prueba de Modulo Estadísticas

Caso de Prueba: CP-11 Visualizar estadística.				
ID Prueba	CP-11			
Propósito	Verificación de la correcta visualización de las estadísticas.			
Enfoque	Caja Negra			
Caso de Uso	Seleccionar alertas	Actores	Administrador, Desarrollador	
Prueba N	Valores de Prueba	Resultado esperado	Resultado obtenido	Evaluación
Caso N°1	- Opción: “Estadísticas de memoria”.	<ul style="list-style-type: none"> • Visualización de mensaje “espere estamos cargando los datos”. • Lista de gráficos según la opción escogida. 	<ul style="list-style-type: none"> • “espere estamos cargando los datos”. • Lista de gráficos según la opción escogida. 	Éxito.
Caso N°2	- Opción: “Estadísticas de navegador”.	<ul style="list-style-type: none"> • Visualización de mensaje “espere estamos cargando los datos”. • Lista de gráficos según la opción escogida. 	<ul style="list-style-type: none"> • “espere estamos cargando los datos”. • Lista de gráficos según la opción escogida. 	Éxito.
Caso N°3	- Opción: “Estadísticas de tiempo de ejecución”.	<ul style="list-style-type: none"> • Visualización de mensaje “espere estamos cargando los datos”. • Lista de gráficos según la opción escogida. 	<ul style="list-style-type: none"> • “espere estamos cargando los datos”. • Lista de gráficos según la opción escogida. 	Éxito.
Caso N°4	- Opción: “Dashboard”.	<ul style="list-style-type: none"> • Lista de gráficos y reportes según la opción escogida. 	<ul style="list-style-type: none"> • Lista de gráficos y reportes según la opción escogida. 	Éxito.

Tabla 57: Caso de prueba 11: Visualizar estadísticas

ANEXO: INTERFAZ DEL SOFTWARE.

En esta sección se presentará capturas de pantalla de las interfaces restantes del software. En la **Figura 28**, se visualiza la estructura del módulo Dashboard, además, desde **Figura 31** hasta **Figura 35** se detallarán las funcionalidades representadas en él.



Figura 28: Estructura Módulo Dashboard.

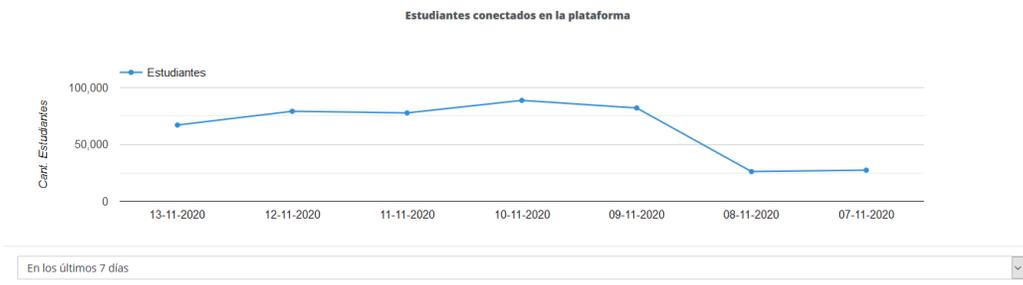


Figura 29: Gráfico de cantidades estudiantes conectados en la plataforma

Usuarios Online

Conectados: 100

Figura 30: Usuarios Conectados.

Dispositivos usados por los usuarios

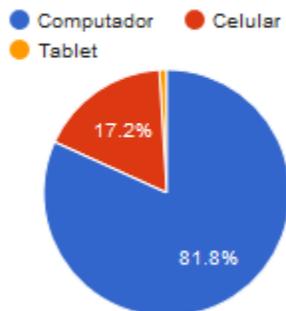


Figura 31: Dispositivos Usados por los Usuarios.

Tops usuarios que más se conectan

Top	Usuario_id	Cantidad
1	29303	579
2	21998	524
3	8610	440
4	28639	417
5	16713	413
6	23225	403
7	35737	381
8	30686	367
9	22643	365
10	26935	362

En los últimos 7 días

Figura 32: Tops usuarios que más se conectan.

Top navegadores usados por los usuarios

Navegador	Usuarios ↓
1. Chrome	9,355 (92.96 %)
2. Firefox	661 (6.57 %)
3. Safari	359 (3.57 %)
4. IE	9 (0.09 %)
5. Edge	7 (0.07 %)
6. Opera	6 (0.06 %)
7. Opera Mini	1 (0.01 %)

% del total: 100 %
10,064

En los últimos 7 días

Figura 33: Top Navegadores Usados por los Usuarios

Top páginas más visitadas

Top	URI	Cantidad
1	cursos	61,424
2	/	49,450
3	curso/21578/modulo/183916	1,859
4	curso/21579/modulo/183917	1,248
5	curso/21402	738
6	curso/22274	714
7	curso/22530/modulo/183050	609
8	curso/22304	597
9	curso/21427	593
10	curso/21943	547

En los últimos 7 días

Figura 34: Top páginas más visitadas.

Sistemas operativos utilizados

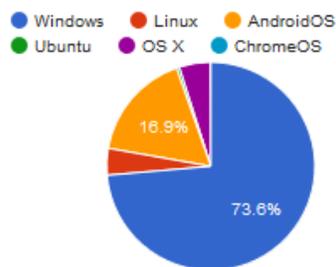


Figura 35: Sistemas operativos utilizados

En la siguiente figura se visualiza la estructura del módulo Estadísticas, sección memoria, además, desde la **Figura 37** hasta la **Figura 39** se detallarán las funcionalidades pertenecientes a la sección.



Figura 36: Estructura de Módulo Estadísticas, sección memoria

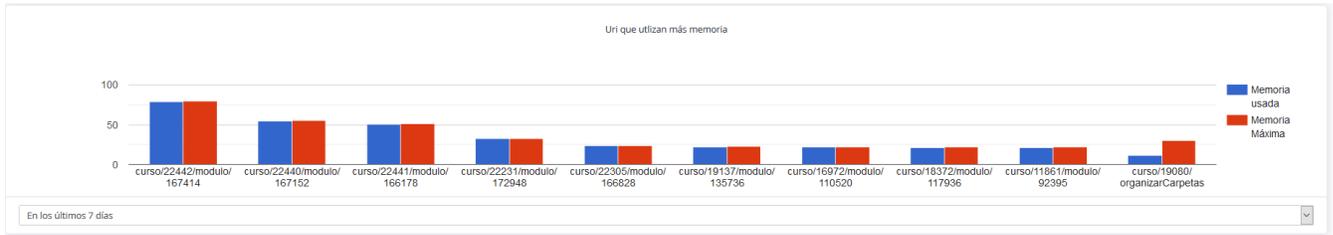


Figura 37: URI que utilizan más memoria



Figura 38: Controladores que utilizan más memoria



Figura 39: Métodos que utilizan más memoria

Posteriormente, en la **Figura 40** se visualiza la estructura de modulo estadísticas, sección navegadores.

Tops versiones según navegador

Firefox		Descendentes		Ascendentes	
Top	Versión	Cantidad	Top	Versión	Cantidad
1	82.0	24.230	1	76.0	1
2	81.0	904	2	73.0	3
3	83.0	314	3	43.0	6
4	80.0	217	4	60.0	6
5	79.0	202	5	75.0	8

En los últimos 7 días

Figura 40: Estadísticas de Navegadores

En la **Figura 41**, se visualiza la estructura del módulo de sugerencias, asimismo, en las **Figura 42** y **Figura 43** se especificarían los detalles de la sección.

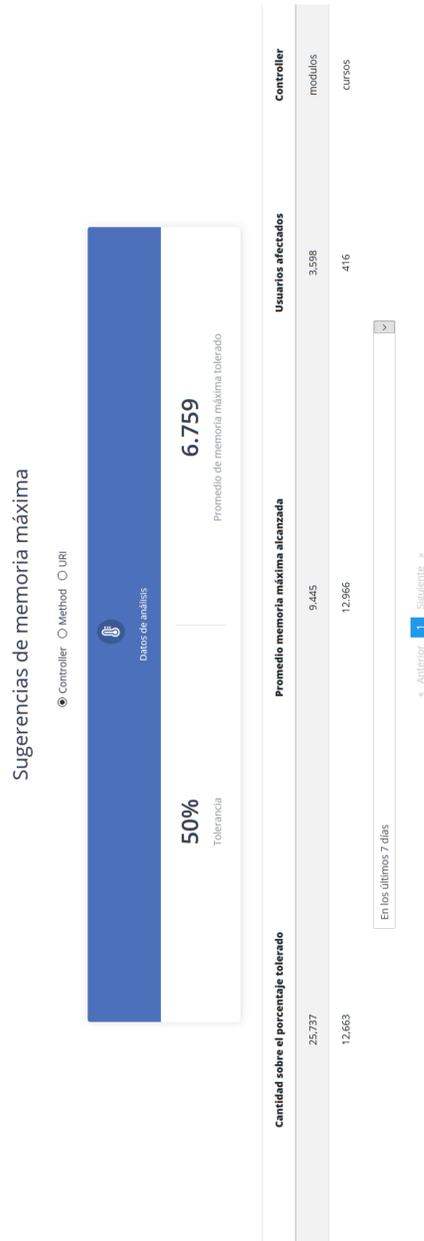


Figura 41: Estructura Módulo Sugerencias



Figura 42: Datos de análisis

Cantidad sobre el porcentaje tolerado	Promedio memoria máxima alcanzada	Usuarios afectados	Controller
25.737	9.445	3.598	modulos
12.663	12.966	416	cursos

En los últimos 7 días

< Anterior 1 Siguiente >

Figura 43: Reporte de sugerencias

En la **Figura 44**, se visualiza la estructura del módulo de alertas, también, en las **Figura 45**, **Figura 46**, **Figura 47** se especificarían los detalles de la sección.

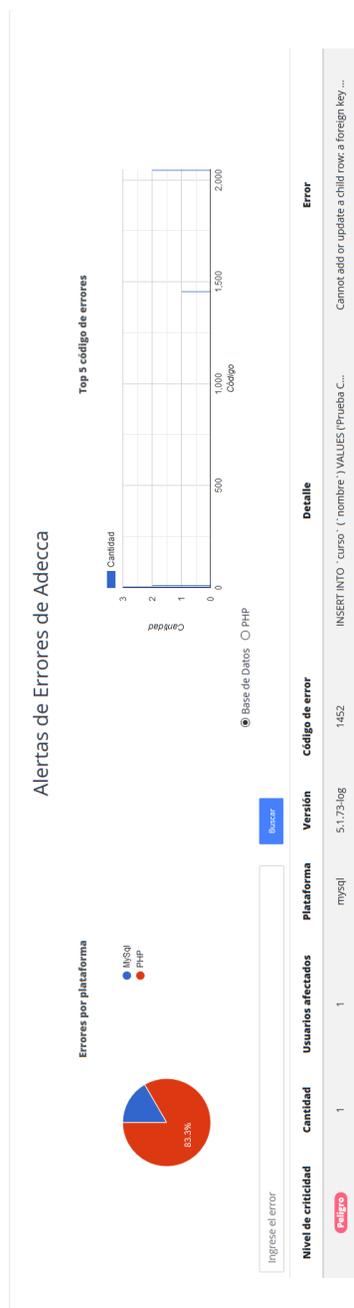


Figura 44: Estructura Módulo alerta

Errores por plataforma

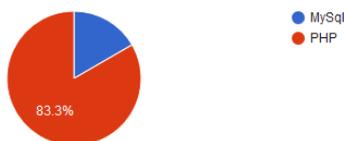


Figura 45: Errores por plataforma

Top 5 código de errores

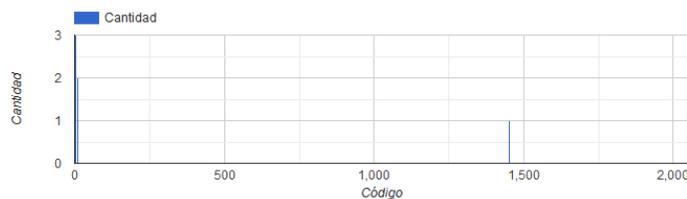


Figura 46: Top 5 código de errores

Ingrese el error

Nivel de criticidad	Cantidad	Usuarios afectados	Plataforma	Versión	Código de error	Detalle	Error
Peligro	1	1	mysql	5.1.73-log	1452	INSERT INTO `curso` (`nombre`) VALUES ('Prueba C...	Cannot add or update a child row: a foreign key ...

< Anterior 1 Siguiente >

Figura 47: Reporte de alertas

Finalmente, en las **Figura 48** y **Figura 49**, se visualizará los detalles de las notificaciones.

Notificaciones Nuevas

18/12/2020 21:06:31 Información
 [Runtime Notice] Accessing static property
 Intranet::\$connection as non static
 visto

18/12/2020 21:06:31 Advertencia
 [Warning] mssql_query() expects parameter 2 to be
 resource, null given
 visto

18/12/2020 21:06:31 Advertencia
 [Warning] mssql_num_rows() expects parameter 1
 to be resource, null given
 visto

18/12/2020 21:06:31 Advertencia
 [Warning] mssql_connect(): Unable to connect to
 server: xsistemas
 visto

18/12/2020 21:06:31 Información
 [Notice] Undefined property: Intranet::\$connection
 visto

Figura 48: Listado de notificación

Error de PHP

Detalle Error

Nivel de criticidad: **información** Número de Error: 2048

Mensaje: [Runtime Notice] Accessing static property
Intranet::\$connection as non static



Archivo: libraries/intranet.php Línea: 76
Versión: 5.6.40-38+ubuntu16.04.1+deb.sury.org+1

Ignorar

Sistema De Monitoreo y Alertas Tempranas Adecca

Figura 49: Detalle de notificación.

ANEXO: RESULTADO VERIFICACIÓN ATRIBUTOS DEL SOFTWARE

Encuesta			
Característica	Pregunta	Evaluación	Respuesta
Usabilidad	¿El software genera para cada una de las operaciones realizadas un mensaje que indique el éxito o fracaso de éstas, para que el usuario pueda interpretar claramente lo realizado?	Bueno	X
		Regular	
		Malo	
	¿El tamaño de fuente es el ideal para leer la información que entrega sistema?	Bueno	X
		Regular	
		Malo	
	¿Se entiende el contenido de los datos solicitados en los formularios?	Bueno	X
		Regular	
		Malo	
Seguridad	Nivel de protección sobre los datos sensibles del usuario.	Bueno	X
		Regular	
		Malo	
Interfaz Gráfica	¿Encuentra visualmente amigable el sistema?	Bueno	X
		Regular	
		Malo	
	Nivel de representatividad de los íconos e imágenes que aparecen en la interfaz del sistema	Bueno	X
		Regular	
		Malo	
	Nivel de representatividad que tienen los nombres de los botones	Bueno	X
		Regular	
		Malo	

Tabla 58: Encuesta 1

Encuesta			
Característica	Pregunta	Evaluación	Respuesta
Usabilidad	¿El software genera para cada una de las operaciones realizadas un mensaje que indique el éxito o fracaso de éstas, para que el usuario pueda interpretar claramente lo realizado?	Bueno	X
		Regular	
		Malo	
	¿El tamaño de fuente es el ideal para leer la información que entrega sistema?	Bueno	X
		Regular	
		Malo	
	¿Se entiende el contenido de los datos solicitados en los formularios?	Bueno	X
		Regular	
		Malo	
Seguridad	Nivel de protección sobre los datos sensibles del usuario.	Bueno	X
		Regular	
		Malo	
Interfaz Gráfica	¿Encuentra visualmente amigable el sistema?	Bueno	X
		Regular	
		Malo	
	Nivel de representatividad de los íconos e imágenes que aparecen en la interfaz del sistema	Bueno	X
		Regular	
		Malo	
	Nivel de representatividad que tienen los nombres de los botones	Bueno	X
		Regular	
		Malo	

Tabla 59: Encuesta 2

Encuesta			
Característica	Pregunta	Evaluación	Respuesta
Usabilidad	¿El software genera para cada una de las operaciones realizadas un mensaje que indique el éxito o fracaso de éstas, para que el usuario pueda interpretar claramente lo realizado?	Bueno	X
		Regular	
		Malo	
	¿El tamaño de fuente es el ideal para leer la información que entrega sistema?	Bueno	X
		Regular	
		Malo	
	¿Se entiende el contenido de los datos solicitados en los formularios?	Bueno	X
		Regular	
		Malo	
Seguridad	Nivel de protección sobre los datos sensibles del usuario.	Bueno	
		Regular	X
		Malo	
Interfaz Gráfica	¿Encuentra visualmente amigable el sistema?	Bueno	X
		Regular	
		Malo	
	Nivel de representatividad de los íconos e imágenes que aparecen en la interfaz del sistema	Bueno	X
		Regular	
		Malo	
	Nivel de representatividad que tienen los nombres de los botones	Bueno	X
		Regular	
		Malo	

Tabla 60: Encuesta 3