



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**

**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**

**ESCUELA DE INGENIERÍA EN SISTEMAS**

**“DESARROLLO DE UN SISTEMA ACADÉMICO ORIENTADO A  
LA WEB PARA LA UNIDAD EDUCATIVA EXPERIMENTAL  
INTERANDINA UTILIZANDO EL FRAMEWORK LARAVEL Y  
MYSQL”**

**TRABAJO DE TITULACIÓN**

**TIPO: PROYECTO TÉCNICO**

Presentado para optar el grado académico

**INGENIERO EN SISTEMAS**

**AUTOR: BYRON FERNANDO HUEBLA SOCAG**

**TUTOR: ING. PATRICIO ADOLFO ROMERO**

Riobamba – Ecuador

2018

© 2018, Byron Fernando Huebla Socag

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

---

Byron Fernando Huebla Socag

**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y CHIBORAZO**  
**ESCUELA DE INGENIERÍA EN SISTEMAS**

El Tribunal del trabajo de titulación certifica que: El trabajo de investigación: Tipo Técnico “**DESARROLLO DE UN SISTEMA ACADÉMICO ORIENTADO A LA WEB PARA LA UNIDAD EDUCATIVA EXPERIMENTAL INTERANDINA UTILIZANDO EL FRAMEWORK LARAVEL Y MYSQL**”, de responsabilidad del señor BYRON FERNANDO HUEBLA SOCAG, ha sido minuciosamente revisado por los miembros del trabajo de titulación, quedando autorizada su presentación.

FIRMA

FECHA

Dr. Julio Santillán \_\_\_\_\_

**VICEDENANO DE LA FACULTAD  
DE INFORMÁTICA Y ELECTRÓNICA**

Ing. Patricio Moreno \_\_\_\_\_

**DIRECTOR DE LA ESCUELA  
DE INGENIERÍA EN SISTEMAS**

Ing. Patricio Adolfo Romero \_\_\_\_\_

**DIRECTOR DEL TRABAJO  
DE TITULACIÓN**

Dr. Mario Paguay \_\_\_\_\_

**MIEMBRO DE TRIBUNAL**

Yo, BYRON FERNANDO HUEBLA SOCAG, soy responsable de las ideas, doctrinas y resultados expuestos en esta Tesis y el patrimonio intelectual de la Tesis de Grado pertenece a la Escuela Superior Politécnica de Chimborazo

BYRON FERNANDO HUEBLA SOCAG

## **DEDICATORIA**

En primer lugar, no podría dejar de nombrar a mi Dios por darme salud y el valor necesario para llegar a cumplir mis objetivos y, sobre todo protegerme y cuidarme en el transcurso de mi vida cotidiana.

A mis padres Andrés Huebla y Rosa Socag a mis hermanos Hugo y Wendy y en especial a mi hijo político Alan quienes fueron mis pilares fundamentales a lo largo de toda mi vida estudiantil, dando todo el apoyo posible en todas sus magnitudes y siendo el hijo y hermano que no les supo fallar.

Byron

## **AGRADECIMIENTO**

El más sincero agradecimiento a Dios y a la Escuela Superior Politécnica por brindarme una noble profesión con el cual pienso ser un profesional que pueda compensar los conocimientos adquiridos en las aulas en el bienestar de la sociedad.

Además, no podría dejar en alto el agradecimiento a mis tíos, primos amigos y demás familiares por darme ese apoyo moral con el cual he podido seguir adelante siendo esa persona humilde y agradecida con todos los que estuvieron a mi lado a lo largo de mi etapa de estudiante.

Byron

## TABLA DE CONTENIDOS

	Páginas
<b>DEDICATORIA</b> .....	v
<b>AGRADECIMIENTO</b> .....	vi
<b>TABLA DE CONTENIDOS</b> .....	vii
<b>ÍNDICE DE TABLAS</b> .....	xi
<b>ÍNDICE DE FIGURAS</b> .....	xiii
<b>ÍNDICE DE ANEXOS</b> .....	xiv
<b>ÍNDICE DE ABRIVIATURAS</b> .....	xv
<b>RESUMEN</b> .....	xvi
<b>SUMMARY</b> .....	xvii
<b>INTRODUCCIÓN</b> .....	1
<b>CAPÍTULO I</b>	
<b>1. INTRODUCCIÓN</b> .....	2
<b>1.1. Antecedentes</b> .....	2
<b>1.2. Formulación del problema</b> .....	3
<b>1.3. Sistematización del problema</b> .....	3
<b>1.4. Justificación de la investigación</b> .....	4
<i>1.4.1. Justificación teórica</i> .....	4
<i>1.4.2. Justificación aplicada</i> .....	5
<b>1.5. Objetivos</b> .....	5
<i>1.5.1. Objetivo general</i> .....	5
<i>1.5.2. Objetivos específicos</i> .....	6
<b>1.6. Marco teórico</b> .....	6
<i>1.6.1. Sistema académico orientado a la web</i> .....	6

1.6.2.	<i>MySQL</i> .....	11
1.6.3.	<i>Análisis: ventajas y desventajas de laravel</i> .....	13
1.6.4.	<i>PHP</i> .....	19
1.6.5.	<i>Metodología de desarrollo SCRUM</i> .....	23
1.6.6.	<i>ISO/IEC 9126-3 calidad del software</i> .....	26

## CAPÍTULO II

2.	<b>MARCO METODOLÓGICO</b> .....	28
2.1.	<b>Introducción</b> .....	28
2.2.	<b>Aplicación de la metodología de desarrollo scrum</b> .....	28
2.3.	<b>Fase de planificación</b> .....	28
2.3.1.	<i>Identificación de personas y roles involucrados en el proyecto</i> .....	28
2.3.2.	<i>Tipos de usuarios y roles de usuario en el sistema</i> .....	29
2.3.3.	<i>Actividades del proyecto</i> .....	29
2.3.4.	<i>Product backlog</i> .....	30
2.3.5.	<i>Definición de Módulos del Sistema</i> .....	33
2.3.6.	<i>Arquitectura del Sistema</i> .....	34
2.3.7.	<i>Diagrama de caso de uso</i> .....	35
2.3.8.	<i>Diagrama de Clases</i> .....	38
2.3.9.	<i>Sprint backlog (planificación)</i> .....	38
2.3.10.	<i>Cronograma de Actividades</i> .....	40
2.3.11.	<i>Reuniones scrum</i> .....	41
2.4.	<b>Fase de desarrollo</b> .....	41
2.4.1.	<i>Instalación de Laravel</i> .....	41
2.4.2.	<i>Descripción de las capas de la aplicación</i> .....	44
2.4.3.	<i>Estándar de codificación</i> .....	44
2.4.4.	<i>Diseño de la base de datos</i> .....	45



2.4.5.	<i>Diccionario de datos</i> .....	47
2.4.6.	<i>Estándar de interfaz del sistema</i> .....	47
2.4.7.	<i>Desarrollo de las Historias de Usuario</i> .....	49
2.5.	<b>Fase de Finalización</b> .....	52
2.5.1.	<i>Gestión del Proyecto</i> .....	53

### CAPÍTULO III

3.	<b>MARCO DE EVALUACIÓN, DISCUSIÓN Y ANÁLISIS DE RESULTADOS</b> ...	55
3.1.	<b>Cálculo de la muestra poblacional para la evaluación del sistema académico</b> .....	55
3.2.	<b>Requerimientos de calidad</b> .....	56
3.3.	<b>Análisis de la funcionalidad del sistema</b> .....	57
3.3.1.	<i>Métricas de evaluación: Adecuidad</i> .....	57
3.3.2.	<i>Métricas de evaluación: Exactitud</i> .....	58
3.3.3.	<i>Métricas de evaluación: Interoperabilidad</i> .....	59
3.3.4.	<i>Métricas de evaluación: Seguridad</i> .....	59
3.3.5.	<i>Métricas de evaluación: Conformidad de la funcionalidad</i> .....	60
3.4.	<b>Análisis de la usabilidad del sistema</b> .....	61
3.4.1.	<i>Métricas de evaluación: Entendibilidad</i> .....	61
3.4.2.	<i>Métricas de evaluación: Aprendibilidad</i> .....	62
3.4.3.	<i>Métricas de evaluación: Operabilidad</i> .....	63
3.4.4.	<i>Métricas de evaluación: Atractivo</i> .....	64
3.4.5.	<i>Métricas de evaluación: Conformidad de la usabilidad</i> .....	65
3.5.	<b>Análisis de la eficacia del sistema</b> .....	66
3.5.1.	<i>Métricas de evaluación: Tiempo de respuesta</i> .....	66
3.5.2.	<i>Métricas de evaluación: Utilización de recursos</i> .....	67
3.5.3.	<i>Métricas de evaluación: Conformidad de la eficiencia</i> .....	68
3.6.	<b>Evaluación de la calidad del sistema</b> .....	69

<b>CONCLUSIONES</b> .....	73
<b>RECOMENDACIONES</b> .....	74
<b>BIBLIOGRAFÍA</b> .....	1

## ÍNDICE DE TABLAS

	Páginas
<b>Tabla 1-1:</b> Comparación entre base de datos .....	12
<b>Tabla 2-1:</b> Comparativa de frameworks.....	14
<b>Tabla 3-1:</b> Características de calidad del software.....	26
<b>Tabla 1-2:</b> Roles del equipo .....	29
<b>Tabla 2-2:</b> Tipos de usuarios en el sistema .....	29
<b>Tabla 3-2:</b> Actividades desarrolladas en el proyecto .....	30
<b>Tabla 4-2:</b> Product backlog.....	30
<b>Tabla 5-2:</b> Módulos del Sistema .....	34
<b>Tabla 6-2:</b> Sprint backlog.....	39
<b>Tabla 7-2:</b> Planificación de actividades .....	40
<b>Tabla 8-2:</b> Estándar de codificación.....	45
<b>Tabla 9-2:</b> Diccionario de datos .....	47
<b>Tabla 10-2:</b> Estándar de interfaces del sistema.....	48
<b>Tabla 11-2:</b> Historia de usuario 01 / Registro estudiantes .....	49
<b>Tabla 12-2:</b> Tareas de ingeniería - Historia usuario 01.....	50
<b>Tabla 13-2:</b> Tarea de ingeniería 01 - Historia de usuario 01.....	51
<b>Tabla 14-2:</b> Prueba de aceptación 01 - Historia de usuario 01 .....	51
<b>Tabla 15-2:</b> Actividades de finalización de proyecto.....	52
<b>Tabla 1-3:</b> Parámetros de medición .....	56
<b>Tabla 2-3:</b> Sub características de calidad.....	56
<b>Tabla 3-3:</b> Métrica de adecuación.....	57
<b>Tabla 4-3:</b> Métrica de exactitud .....	58
<b>Tabla 5-3:</b> Métrica de interoperabilidad.....	59
<b>Tabla 6-3:</b> Métrica de seguridad .....	60
<b>Tabla 7-3:</b> Métrica de conformidad de la funcionalidad.....	60
<b>Tabla 8-3:</b> Métrica de entendibilidad.....	62
<b>Tabla 9-3:</b> Métrica de aprendibilidad.....	63
<b>Tabla 10-3:</b> Métrica de operabilidad.....	64
<b>Tabla 11-3:</b> Métrica de atracción .....	64
<b>Tabla 12-3:</b> Métrica de conformidad de la usabilidad .....	65
<b>Tabla 13-3:</b> Métrica de tiempo de respuesta .....	66

<b>Tabla 14-3:</b> Utilización de recursos .....	67
<b>Tabla 15-3:</b> Métrica de conformidad de la eficiencia .....	68
<b>Tabla 16-3:</b> Especificación de la evaluación de funcionalidad del sistema .....	69
<b>Tabla 17-3:</b> Especificación de la evaluación de usabilidad del sistema.....	70
<b>Tabla 18-3:</b> Especificación de la evaluación de eficiencia del sistema.....	72

## ÍNDICE DE FIGURAS

	Páginas
<b>Figura 1-1:</b> Proceso de Scrum.....	24
<b>Figura 1-2:</b> Arquitectura del Sistema.....	35
<b>Figura 2-2:</b> Diagrama de caso del Administrador.....	36
<b>Figura 3-2:</b> Diagrama de caso del Docente.....	37
<b>Figura 4-2:</b> Diagrama de caso del estudiante.....	37
<b>Figura 5-2:</b> Diagrama de clases .....	38
<b>Figura 6-2:</b> Instalación de Composer en Windows.....	42
<b>Figura 7-2:</b> Prueba instalación Composer.....	42
<b>Figura 8-2:</b> Instalación de Laravel .....	43
<b>Figura 9-2:</b> Diagrama entidad relación .....	46
<b>Figura 10-2:</b> Interfaz del sistema - Pantalla login.....	48
<b>Figura 11-2:</b> Interfaz gráfica del sistema “Diller” .....	49
<b>Figura 12-2:</b> Estructura de archivos sistema Diller.....	53
<b>Figura 13-2:</b> BurnDown Chart.....	54
<b>Figura 1-3:</b> Muestreo Aleatorio Simple .....	55
<b>Figura 2-3:</b> Evaluación de funcionalidad del sistema.....	70
<b>Figura 3-3:</b> Evaluación de usabilidad del sistema.....	71
<b>Figura 4-3:</b> Evaluación de eficiencia del sistema .....	72

## **ÍNDICE DE ANEXOS**

**Anexo A:** Modelo físico de la base de datos

**Anexo B.** Manual de usuario

**Anexo C.** Encuesta Aplicada a los Usuarios

## ÍNDICE DE ABRIVIATURAS

<b>UEI</b>	Unidad Educativa Interandina
<b>ESPOCH</b>	Escuela Superior Politécnica de Chimborazo
<b>HT</b>	Historias Técnicas
<b>HU</b>	Historias de Usuarios
<b>HTML</b>	Hypertext markup lenguaje (Lenguaje de marcas de hipertexto)
<b>HW</b>	Hardware
<b>SW</b>	Software
<b>BD</b>	Base de Datos
<b>JS</b>	JavaScript
<b>SO</b>	Sistemas Operativos
<b>MYSQL</b>	My Structured Query Language (Lenguaje de Consulta Estructurado)
<b>PHP</b>	Hypertext Preprocessor
<b>SGBD</b>	Sistema de Gestor de Base de Datos
<b>TICs</b>	Tecnologías de la Información y Comunicación
<b>MVC</b>	Modelo Vista Controlador
<b>IDE</b>	Integrated development enviroment (Entorno de desarrollo integrado)
<b>DBMS</b>	Database Management System (Sistema de administración de Base de Datos)
<b>API</b>	Application Program Interface (Interfaz de programación de aplicaciones)
<b>ISO</b>	Organización Internacional de Normalización

## RESUMEN

El presente trabajo de titulación tiene como objetivo desarrollar un sistema académico orientado a la web para la Unidad Educativa Experimental “Interandina” utilizando el framework Laravel y MySQL en la comunidad Santa Teresita de “Guabug”, de la parroquia San Juan perteneciente al cantón Riobamba con la finalidad de disminuir los tiempos de acceso a la información académica por parte de docentes, estudiantes y padres de familia; Las herramientas tecnológicas utilizadas para el desarrollo del sistema son: Framework Laravel, MySQL, PHP, HTML5, APACHE, Javascript, SublimeText, CSS. La metodología de desarrollo seleccionada para el sistema es SCRUM con la finalidad de crear un entorno de trabajo óptimo e interactivo entre el Product Owner y el equipo de desarrollo, está compuesto de tres fases de desarrollo las cuales son: Fase de Planificación en esta fase mediante una entrevista se recopilaron los requerimientos denominadas Historias de Usuario, en la Fase de Desarrollo se efectuaron las historias de Usuario agrupados en Sprints, y en la Fase de Cierre se realizaron las pruebas del Sistema. Una vez finalizado el desarrollo del Sistema este se sometió a pruebas de funcionalidad, usabilidad y eficiencia, mediante las encuestas dirigidas a los profesores, estudiantes, administrativos y padres de familia de la Institución y se llegó a determinar que el sistema académico orientado a la web posee un 94 % en lo funcional, 99% en usabilidad y 93% en eficiencia del Sistema Académico obteniendo como promedio global 95 %, tras los resultados obtenidos se concluye que el sistema académico “DILLER” servirá de ayuda a docentes, alumnos, padres de familia y administradores a disminuir tiempos de respuesta y tiempos de atención a requerimientos por lo que se recomienda el uso inmediato del Sistema Académico “DILLER” a todos quienes forman parte de la Unidad Educativa Experimental “Interandina.”

**Palabras clave:** <TECNOLOGÍAS Y CIENCIAS DE LA INGENIERÍA>, <INGENIERÍA DE SOFTWARE>, <SISTEMA ACADÉMICO>, <EFICIENCIA DEL SISTEMA>, <METODOLOGÍA DE DESARROLLO>, <FRAMEWORK LARAVEL>, <ACCESO A LA INFORMACIÓN>, <TIEMPOS DE RESPUESTA>, <TIEMPOS DE ATENCIÓN>, , <SAN JUAN (PARROQUIA)>..



## **SUMMARY**

The present paper aims to develop an academic system by web for “Interandina” Experimental Educational Institution using framework Laravel and MYSQL (My Structured Query Language) in Santa Teresita de “Guabug” Community, San Juan Parish, Riobamba Canton. The professors, students and parents will be able get academic information reducing access times. Framework Laravel, MySQL, PHP, HTML5, APACHE, Javascript, SublimeText, and CSS were used as technological tools to develop the present system. SCRUM was also used so that the work can be good and interactive between the Product Owner and the equipment of development. It is compound of three phases of development as follows: Planning Phase, Development Phase and Close Phase. In the Planning Phase, we gathered the requirements called Users History by interviews; in the Development Phase, Users Histories divided in Sprints were carried out and in the Close Phase system, tests were carried out. When we finished the system development, it was tested about functionality, usability and efficiency by surveys filled out by professors, students, administrative staff and parents of the institution. It was determined that this system has 94 % of functionality, 99% of usability and 93% of efficiency. That means that 95% of quality was gotten I this academic system. From the result, it is concluded that the academic system “DILLER” will be useful and reduce response times and attendance time when teachers, students, administrative staff and parents need to get information. It is recommended that the staff of this institution use the DILLER Academic System immediately.

**KEY WORDS:** <ENGINEERING SCIENCE AND TECHNOLOGY>, <SOFTWARE ENGINEERING>, <ACEDEMIC SYSTEM>, <SYSTEM EFFICIENCY>, <DEVELOPMENT METHODOLOGY>, <FRAMEWORK LARAVEL>, <INFORMATION ACCESS>, <RESPONSE TIMES>, <ATTENDANCE TIMES>, <SAN JUAN (PARISH)>.

## INTRODUCCIÓN

El presente trabajo de titulación tiene como objetivo principal describir el análisis, desarrollo, implementación y evaluación del Sistema Académico orientado a la Web para la Unidad Educativa Experimental “Interandina” utilizando el framework Laravel y el la Base de Datos MySQL, situado en la comunidad Santa Teresita de Guabug, perteneciente a la parroquia San Juan del cantón Riobamba y de esta manera demostrar la eficiencia de una aplicación web frente a una aplicación de escritorio.

Además, se presenta una propuesta de software que ayuden a disminuir los tiempos de respuesta y tiempos de atención a requerimientos y de esta manera poder optimizar los procesos internos y externos.

Mediante el uso de la metodología de desarrollo ágil SCRUM, se obtuvo una comunicación continua con el Product Owner el cual nos sirvió de mucha ayuda al momento del desarrollo y por ende el cumplimiento de cada tarea de una manera eficaz y de mutuo acuerdo entre las dos partes.

Este apartado está compuesto de tres capítulos en el que el primer capítulo hace referencia a la descripción teórica de las herramientas tecnológicas utilizadas, en el segundo capítulo se enfatizó en el desarrollo del Sistema Académico para la Unidad Educativa Experimental “Interandina” utilizando el framework Laravel y el motor de Base de Datos MySql, Finalmente en el capítulo tres se muestran los resultados obtenidos luego de realizar el análisis y evaluación del sistema.

# CAPÍTULO I

## 1. INTRODUCCIÓN

### 1.1. Antecedentes

Hoy en día el uso de las nuevas tecnologías y la automatización en los diferentes procesos han sido enmarcadas por la evolución de las nuevas tecnológicas mediante el desarrollo e implementación de aplicaciones en la Web tanto así que el Software ha llegado a convertirse en una necesidad prioritaria en las diferentes áreas de negocios, comercios, empresas instituciones educativas, etc.

La Unidad Educativa Fiscal “Interandina” fue creada el 28 de marzo de 1995 con el Acuerdo ministerial 039 con la especialidad de Mecánica Industrial como una respuesta a la necesidad de entregar una educación de calidad a los estudiantes de la comunidad Santa Teresita de Guabug, parroquia San Juan y sus alrededores, siendo una institución donde se inculca una educación investigativa, científica, y práctica orientada hacia las nuevas necesidades que la sociedad los exige.

Mediante una entrevista hacia el Msg. Humberto Pagalo en calidad de rector de la Unidad Educativa Experimental “Interandina” supo manifestar que la institución mantiene un pequeño sistema de escritorio que permite el ingreso de notas de cada profesor, control de asistencia diaria a estudiantes y docentes además supo manifestar que el sistema permite realizar el ingreso modificación y eliminación de los registros únicamente por el secretario de la institución.

El proceso para el registro de notas inicia al finalizar cada Quimestre en las fechas indicadas por el señor rector donde cada docente envía el registro de notas mediante un correo electrónico al señor secretario, teniendo un tiempo aproximado de un día de duración para realizar todo el proceso. El Sistema en mención está desarrollado en un Libro de Excel mediante la utilización de macros y conectada a una base de datos en Access.

El señor rector además supo estar consciente de que el Sistema que poseen es muy limitado en todas las dimensiones manifestando que es difícil agregar nuevos cursos, asignar maestros encargados a cada curso o a su vez de que cada maestro pueda subir sus propias notas y que esta sea visible para

los estudiantes y padres de familia ya que se ha visto que muchas de las veces los padres de los estudiantes viven en otra ciudad y sólo se entera de las notas al final de cada año lectivo.

Teniendo un vasto conocimiento en los diferentes lenguajes de programación adquiridos en la Politécnica y una profunda investigación realizada en los repositorios de la COBUEC (Consortio de Bibliotecas Universitarias del Ecuador) y en el DISPACE (Solución de Repositorio Institucional de la ESPOCH) pude encontrar estudios comparativos entre frameworks para PHP en los que aparece Laravel como uno de los mejores frameworks al momento de elaborar Sistemas Académicos.

En el mercado actual tranquilamente se puede encontrar Sistemas Académicos, claro dependiendo de la funcionalidad de estos puede llegar a costar desde los 2000.00 dólares en adelante valor que hoy en día esta institución pública puede costearlo. Con la finalidad de dar solución a las limitantes del sistema actual se ha considerado que Laravel es uno de los mejores frameworks de PHP para el desarrollo de aplicaciones web.

Con este nuevo sistema web se obtendría independencia del SO, distribución de la información y sobre todas las cosas el acceso a la información académica sería de inmediato ya que esta debe ser pública para quienes sean parte de la Institución misma que está conformada por 14 aulas, una biblioteca un centro de cómputo, oficina del rectorado, oficina de secretaría general dos aulas de talleres de corte y confección y Mecánica Industrial, así como una cancha de fútbol de uso estudiantil.

## **1.2. Formulación del problema**

¿Cómo es el proceso de registro de información académica en la actualidad?

## **1.3. Sistematización del problema**

¿De qué manera se da el registro de la información hoy en día?

¿Cuál es el tiempo que se tarda en acceder hoy en día a la información académica?

¿Con una aplicación web se mejorarían los tiempos de acceso a la información académica?

## **1.4. Justificación de la investigación**

### ***1.4.1. Justificación teórica***

En la actualidad la implementación de las nuevas tecnologías de la información en los ambientes de aprendizaje ha alcanzado tal jerarquía, que cada día es mayor el número de instituciones educativas que disponen un sistema web en el cual tiene implementado diferentes tipos de aplicaciones para el aprendizaje, enseñanza, control y difusión de todo tipo de información relevante para quienes forman parte del establecimiento.

Las aplicaciones desarrolladas en la web ofrecen un sin número de ventajas en todos los campos porque usan pocos recursos a diferencia de los programas o aplicaciones instaladas en las PC's, estas aplicaciones resuelven problemas como de concurrencia permitiendo un acceso de forma inmediata desde cualquier lugar de la red disponible y así poder tener acceso a la información de interés tanto como para estudiantes, maestros, autoridades del establecimiento y padres de familia.

Las aplicaciones orientadas a la Web no necesitan ser actualizadas a cada instante del lado del cliente, las aplicaciones de tipo Web no demandan que el usuario se preocupe por obtener la última versión de la Aplicación debido a que esto lo realiza del lado del servidor de una manera rápida e inmediata que tan solo con dar mantenimiento al servidor de aplicaciones los usuarios que hagan uso del sistema ya tendrán acceso al sistema con la última versión del sistema.

En el ambiente académico además de ser un gran recurso pedagógico, sirve de plataforma para la divulgación de información institucional, la publicación del material por parte de los docentes, la conformación y distribución de estudiantes a cada sección, la guía de actividades dentro del año lectivo, servicio de biblioteca y otro tipo de servicios al que tendrá acceso el administrador del Sistema.

Con la finalidad de solventar los problemas antes mencionados y distinguiendo las grandes ventajas que ofrecen las aplicaciones orientadas a la Web, se ha visto conveniente la creación de una aplicación Web que proporcione acceso a la información académica confiable y verídica de estudiantes, docentes, padres de familia y administrativos.

### ***1.4.2. Justificación aplicativa***

La Unidad Educativa Fiscal “Interandina” en este momento cuenta con un Sistema Académico de tipo escritorio, el cual permite realizar en registro de estudiantes en cada nivel, registro de estudiantes y docentes al que solo tiene acceso el secretario de la Institución.

El Sistema académico es de tipo orientado a la Web, ocupando el paradigma orientado a objetos, para el cual se utilizó el lenguaje de programación PHP 5.6 (5.6.30) con soporte de un servidor Apache en su versión 2.4.25 en un ambiente de desarrollo con Sublime Text, utilizando el framework Laravel 5.1 y la base de datos MySQL.

Considerando que hoy en día está en cúspide y con la finalidad de llevar a cabo una solución ágil, el sistema fue desarrollado utilizando la metodología SCRUM con las siguientes etapas:

- **Planificación:** Requerimientos, riesgos, factibilidad, Arquitectura del Sistema, Diseño de la interfaz y Diseño de Base de datos.
- **Desarrollo:** Sprints
- **Cierre:** Evaluación y pruebas de calidad al Sistema.

El desarrollo modular del sistema se dividirá en cinco módulos que son: Configuración, Académico, Matriculación, Calificaciones y Reportes.

- Configuración
- Académico
- Matriculación
- Calificaciones
- Reportes

## **1.5. Objetivos**

### ***1.5.1. Objetivo general***

Desarrollar un Sistema Académico Orientado a la Web para la Unidad Educativa Experimental “Interandina” utilizando el Framework LARAVEL y MYSQL.

### ***1.5.2. Objetivos específicos***

- Estudiar el desarrollo Web utilizando el Framework LARAVEL y MYSQL.
- Realizar un levantamiento de información académica a través de un proceso organizado para el análisis del sistema web.
- Implementar un Sistema Académico orientado a la web para la Unidad Educativa Experimental “Interandina”.
- Evaluar el Sistema Académico para determinar la efectividad de la información académica

## **1.6. Marco teórico**

### ***1.6.1. Sistema académico orientado a la web***

- **Introducción a las aplicaciones web**

Hoy en día las tecnologías de la información forman parte de la vida cotidiana y de la nueva sociedad del conocimiento. La era tecnológica avanza a pasos acelerados donde el internet constituye una herramienta de suma importancia en el campo formativo, el uso de las tecnologías de la información y comunicación (TICs) son un recurso pedagógico de suma importancia para la formación de los estudiantes.

Para la implementación de los TICs se debe considerar el tema del software, el mismo que corresponde a los sistemas y programas informáticos instalados en un computador, dispositivo móvil, tabletas, Televisor inteligente, Ipad u otro dispositivo que tenga acceso a la web.

Las aplicaciones de escritorio tienen sus ventajas frente a las aplicaciones web entre una de las principales que se puede mencionar, que son robusta y estables, envió rápido de respuestas a las peticiones.

En la actualidad las aplicaciones web han evolucionado cuatro generaciones y brindan muchas más ventajas frente las aplicaciones de escritorio, permitiendo el acceso remoto a las aplicaciones de servidor, evitando y personalmente a dar mantenimiento a cada terminal, y su ejecución no depende del sistema operativo desde el cual se acceda, este tipo de aplicaciones además permite el acceso a información de una manera simultánea, rápida y lo mejor de una manera fácil.

Teniendo en cuenta que las aplicaciones implementadas en la web son más vulnerables a los ataques de los hackers y éstos a su vez puedan realizar fraudes, acceso indebido, robo información, manipulación de la información existente, interrupción de la red etc. Por esto es importante para el desarrollo firme, y extensible buscar las tecnologías más apropiadas para aplicaciones empresariales de desarrollo ágil, siendo php el mejor que cumple el requerimiento implementado.

- **Fundamentos de la web**

El éxito de la web se basa en dos factores fundamentales: el protocolo http y lenguaje HTML. El primero permite una implementación sencilla de un sistema de comunicaciones que permite enviar cualquier fichero de forma fácil, simplificando funcionamiento del servidor y posibilitando que servidores poco potentes atiendan cientos o miles de peticiones en simultáneo y reduzcan de este modo los costes de despliegue. El segundo, el lenguaje HTML proporciona un mecanismo sencillo y muy eficiente a la hora de crear páginas dinámicas.

- **Aplicaciones de escritorio frente a aplicaciones web**

### **Aplicación de escritorio**

Una aplicación de escritorio es aquella que se encuentra instalada en el ordenador o sistema de almacenamiento (USB) y podemos ejecutar sin disponer de una conexión a internet en nuestro sistema operativo, al contrario que las aplicaciones en la nube que se encuentran alojados en la nube (servidor) al que accedemos a través de la red o internet.

Su rendimiento depende de las diversas configuraciones de hardware como memoria RAM, disco duro, memoria de video, etc.

A continuación, se plantea las ventajas y desventajas de las aplicaciones de escritorio.



### **Ventajas:**

- ✓ Aprovecha mayormente los recursos de la máquina en el cual se ejecuta.
- ✓ Permite hacer uso de un mayor número de recursos, CPU y RAM del equipo
- ✓ Son más robustas y estables que las aplicaciones Web.
- ✓ Ofrece más estabilidad y posibilidad de recuperación en fallo.
- ✓ Pueden ser más seguras (Dependiendo de quién lo desarrolle).

### **Desventajas:**

- ✓ Ocupa espacio en disco y en ocasiones y según que software se han, mucho espacio.
- ✓ Sólo podemos trabajar con ellas en nuestro equipo, por lo que, si no tenemos acceso a él, dejamos de tener acceso a la aplicación.
- ✓ El software debe instalarse siempre en el pc del usuario.
- ✓ La mayoría del código se ejecuta en el pc del cliente y en muchas ocasiones ralentizando el rendimiento de este.
- ✓ La instalación y actualización es personalizada

### **Aplicación web**

En la ingeniería de software se denomina aplicación Web a aquellas aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de internet o de una intranet mediante un navegador. En otras palabras, es una aplicación (Software) que se codifica en un lenguaje soportado por los navegadores Web.

Las aplicaciones web son populares debido a lo práctico del navegador Web como cliente ligero, a la independencia del sistema operativo, así como a la facilidad para actualizar y mantener aplicaciones Web sin distribuir e instalar Software a miles de usuarios potenciales

A continuación, se plantea las ventajas y desventajas de las aplicaciones web.

### **Ventajas:**

- ✓ Ahora tiempo: se pueden realizar tareas sencillas sin necesidad de descargar ni instalar ningún programa.
- ✓ No hay problema de compatibilidad: basta tener un navegador actualizado para poder utilizarlas.
- ✓ No ocupan espacio en nuestro disco duro.
- ✓ Consumo de recursos bajo: dado que toda (o gran parte) de la aplicación no se encuentra en nuestro ordenador, muchas de las tareas que realiza el Software no consumen recursos nuestros porque se realizan desde otro ordenador.
- ✓ Multiplataforma: se pueden usar desde cualquier sistema operativo porque sólo es necesario tener un navegador

### **Desventajas:**

- ✓ Necesitan sí o sí, conexión a internet, por lo que si no tenemos o la que tenemos es de mala calidad, tendremos un problema de conexión.
- ✓ El aprovechamiento de los recursos de la máquina cliente se ve limitado debido a que se ejecuta en el navegador.
- ✓ El servidor debe tener las prestaciones necesarias para ejecutar la aplicación de manera fluida, no sólo para un usuario sino para todos los que la utilizan de forma concurrente.
- ✓ El tiempo de respuesta puede llegar a ser lento dependiendo de las características del ordenador y de la conexión a internet que se disponga.

### • **Servidor web**

Un servidor web es un programa que utiliza el protocolo de transferencia de Hiper Texto, HTTP (Hypertext transfer protocol), para servir los archivos que forman página web a los usuarios, en respuesta a sus solicitudes, que son reenviados por los clientes http de sus computadoras. Las computadoras y los dispositivos dedicados también pueden denominarse servidores web.

Los servidores web disponen de los siguientes tipos que a continuación se plantean:

**Servidor dedicado:** Un servidor dedicado es una computadora en una red que es reservada para servir determinadas necesidades de una red. Por ejemplo, algunas redes requieren que en una computadora sea la encargada de administrar las comunicaciones entre todas las otras computadoras.

Un servidor dedicado podría también ser una computadora que se encargue de administrar los recursos de impresión.

**Servidor Compartido:** El Servidor Compartido está destinado a alojar varios sitios web en el mismo equipo. En un servidor compartido los recursos de hardware son compartidos por todos los proyectos alojados en el de manera que, aunque su administración sea individual, el rendimiento depende del consumo que hagan el resto de webs alojadas en el servidor.

### **Servidor apache 2.4.25**

Apache es el Servidor Web más utilizado, líder con el mayor número de instalaciones a nivel mundial muy delante de otras soluciones como IIS (Internet Information Server) de Microsoft. Apache es un proyecto de código abierto y uso gratuito, multiplataforma (hay versiones para todos los sistemas operativos más importantes), muy robusto y que destaca por sus seguridad y rendimiento.

### **Historia de apache server**

Apache es el Servidor Web más utilizado, líder con el mayor número de instalaciones a nivel mundial muy por delante de otras soluciones como el IIS (Internet Information Server) de Microsoft. Apache es un proyecto de código abierto y uso gratuito, multiplataforma muy robusto y sobre todo destaca por su seguridad y rendimiento. (Andino Guevara, 2016)

Fueron Brian Behlendorf y Cliff Skolnick quienes a través de una lista de correo coordinaron el trabajo y lograron establecer un espacio compartido de libre acceso para los desarrolladores. En 1999 se formó la Fundación de Software Apache para obtener apoyo financiero, organizativo y legal para el servidor.

## **Ventajas del servidor apache**

- ✓ Es confiable, seguro y administrable en internet.
- ✓ Es altamente configurable.
- ✓ Tiene una amplia aceptación en la red.
- ✓ Posee código abierto y es fácil de conseguir ayuda y soporte.
- ✓ Es multi-plataforma ya que se puede instalar en los diferentes sistemas operativos como Windows, Linux o MacOS

## **Desventajas del servidor apache**

- ✓ Falta de integración.
- ✓ Los formatos de configuración no se encuentran estandarizados.
- ✓ No cuenta con una buena administración.
- ✓ Posee limitaciones en las versiones que no son de la familia Server.

### **1.6.2. MySQL**

Es un sistema de administración de bases de datos (Database Management System, DBMS) para bases de datos relacionales. Existen muchos tipos de bases de datos desde un simple archivo hasta sistemas relacionales orientados a objetos. MySQL como base de datos relacional utiliza múltiples tablas para almacenar y organizar la información. MySQL inicialmente fue escrito en C y C++ y destaca por su gran adaptación a diferentes entornos de desarrollo, permitiendo su interacción con los lenguajes de programación más utilizados como php, perl y java y su integración en distintos sistemas operativos. (Malán Guaranga, 2016)

Además, es compatible con múltiples plataformas informáticas y ofrece una infinidad de aplicaciones, por lo que MySQL es fácil de usar tanto para volúmenes de datos grandes como pequeños, aunque es rapidez es a costa de no implementar ciertos aspectos del SQL.

### **Principales características:**

- ✓ Aprovechan la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- ✓ Gestión de usuarios y passwords, manteniendo un muy buen nivel de seguridad en los datos.

- ✓ Se puede descargar su código fuente, esto ha favorecido muy positivamente en su desarrollo y continuas actualizaciones.
- ✓ Es una de las herramientas más utilizadas por los programadores orientados a internet.
- ✓ Infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación.
- ✓ MySQL, es el manejador de bases de datos considerado como el más rápido de internet.
- ✓ Gran rapidez y facilidad de uso
- ✓ Fácil instalación y configuración.
- ✓ Completo soporte para operadores y funciones en cláusulas select y where.
- ✓ Soporta gran cantidad de datos, MySQL server tienen bases de datos de hasta 50 millones de registros.

El manejo y administración de la base de datos es posible a través de PgAdmin, con el del Wokbench y MySQL Front entre otros.

A pesar de lo descrito, existen otros motores de bases de datos, que permiten el almacenamiento de datos, a continuación, se plantea una tabla comparativa que ayudará a describir de mejor manera porque ha sido seleccionado MySQL como motor de base de datos para la implementación del sistema. Ver (**Tabla 1:1**)

**Tabla 1-1:** Comparación entre base de datos

	DESCRIPCIÓN	EXTENSION	VENTAJAS	DESVENTAJAS
MYSQL	Es un motor de base de datos relacional, multihilo y multiusuario, además dispone de Licencia GNU GPL	Multiusuario funciona sobre múltiples plataformas como AOX, BSD, GNU/Linux, MacOS	<ul style="list-style-type: none"> <li>• Soporte multiplataforma.</li> <li>• Vistas actualizadas</li> <li>• Procedimientos almacenados.</li> <li>• Múltiples motores de almacenamiento.</li> </ul>	<ul style="list-style-type: none"> <li>• Soporte de disparadores básicos.</li> <li>• Los privilegios de una tabla no se borran cuando esta se borra.</li> </ul>

*Continuará*

*Continuación*

<p style="writing-mode: vertical-rl; transform: rotate(180deg);">POSTGRESQL</p>	<p>Es un motor de base de datos relacional, orientado a objetos y de uso libre</p>	<p>Alta concurrencia.</p>	<ul style="list-style-type: none"> <li>• Ideal para el desarrollo y aplicación de tecnologías web.</li> <li>• Amplia variedad de tipos de datos.</li> <li>• Integridad transaccional.</li> </ul>	<ul style="list-style-type: none"> <li>• El consumo de recursos es superior a MySQL, por ende, necesita de un ordenador con amplios recursos.</li> <li>• En cifras es dos veces más lento que MySQL</li> <li>• Sus sentencias son poco intuitivas.</li> </ul>
<p style="writing-mode: vertical-rl; transform: rotate(180deg);">MongoDB</p>	<p>Motor de base de datos NoSQL, en lugar de guardar los datos en tablas los almacena en documentos similares a JSON..</p>	<p>Capaz de poner a disposición de muchos usuarios con gran cantidad de datos de manera simultánea.</p>	<ul style="list-style-type: none"> <li>• Están basadas en key-value pairs.</li> <li>• Incluyen diferentes tipos de almacenes como columnas, documentos, gráficos, objetos, XML, etc.</li> </ul>	<ul style="list-style-type: none"> <li>• No admiten funciones de fiabilidad.</li> <li>• Los desarrolladores deben implementar su propio código, lo que agrega más complejidad.</li> <li>• Incluyen la incompatibilidad con consultas SQL</li> </ul>

Realizado por: HUEBLA, Byron, 2018

Luego de observar en resumen el cuadro comparativo el motor de base de dato utilizado ha sido MySQL con su extensión multiusuario, multi-plataforma y sus ventajas frente a los demás motores de base de datos.

**1.6.3. Análisis: ventajas y desventajas de laravel**

• **Framework Laravel**

Laravel es un framework de código abierto para el desarrollo de aplicaciones web en PHP5 que posee una sintaxis simple y elegante. Su creador Taylor Otwell, mostró por primera vez al mundo en el 2011, por lo que se trata de un framework con un enfoque moderno. Laravel propone una 12 forma de desarrollar aplicaciones web de un modo mucho más ágil. Por ejemplo, en Laravel opcionalmente podemos usar el patrón de diseño MVC (Modelo-Vista-Controlador). El objetivo de Laravel es de ser un frameworks que permita el uso de una sintaxis refinada y de expresiva para crear código de forma sencilla, evitando el "código espagueti" y permitiendo multitud de funcionalidades. Aprovecha

todo lo bueno de otros frameworks y utilizan las características de las últimas versiones de PHP. (Anton, 2015)

La mayor parte de su estructura está formada por dependencias, especialmente de Symfony, lo que implica que el desarrollo de Laravel dependa también del desarrollo de sus dependencias.

- **Características de laravel**

- Autenticación mediante login y password que permite a restringir el acceso y el tipo de permisos de los diferentes usuarios.
- Configuración de acceso a los datos mediante el uso de archivos TXT como XML.
- Abstracción de URLs y sesiones, encargándose el framework de su manejo y liberando de esta tarde al programador.
- Internacionalización que facilita la inclusión de varios idiomas en el desarrollo.
- Controladores fácilmente adaptables a las necesidades del proyecto que gestionan las peticiones y eventos.
- Posee un poderoso conjunto de librerías
- Basado en composer
- Usa componentes de Symfony
- Comunidad creciente en el apoyo de este proyecto (Pims, 2018)

No obstante, existen muchos otros frameworks compatibles con PHP que permiten el desarrollo de Aplicaciones Web, a continuación, se muestra una tabla comparativa que ayudará a describir de mejor manera porque Laravel fue elegido para el desarrollo del Sistema Académico. Ver **(Tabla 2:1)**

**Tabla 2-1:** Comparativa de frameworks

CARACTERISTICAS	SYMFONY	LARAVEL	PHALCON
ARQUITECTURA DE APLICACIONES			
Incorporación del patrón modelo vista controlador orientado a objetos	X	X	X
Operaciones CRUD, asociadas al patrón Active Record.	X	X	

*Continuará*

Continuación

Mapeo de objetos a base de datos relacionales.	X	X	X
Independencia del manejo de base de datos.	X	X	X
Estructura por defecto para aplicaciones.	X	X	
<b>ACCESO A LA WEB</b>			
Despachador de peticiones HTTP.	X	X	X
Generación de URLs amigables.	X	X	X
<b>IMPLEMENTACIÓN DE CÓDIGO HTML</b>			
Uso de plantillas en PHP.	X	X	X
Posibilidad de uso de plantillas en Smarty.	X	X	
Implementación de ayudantes de plantillas.	X	X	
<b>SEGURIDAD</b>			
Manejo propio de sesiones por usuarios.	X	X	X
Manejo de privilegios de acceso a secciones de la aplicación.	X	X	X
Verificación de la salida generada en HTML por procesamiento de peticiones.	X	X	
<b>USABILIDAD Y ACCESO RÁPIDO</b>			
Almacenamiento en caché de las vistas.	X	X	
Almacenamiento en caché de configuración de las aplicaciones	X	X	
<b>DOCUMENTACIÓN PARA USO</b>			
Manual de referencia.	X	X	X
Documentación de la interfaz de programación de aplicaciones.	X	X	X
<b>HERRAMIENTAS DE PROGRAMACIÓN</b>			
Generación de código PHP.	X	X	X
Herramientas de prueba y depuración.		X	
Interfaz de línea de comandos para la creación y mantenimiento de aplicaciones.	X	X	
Almacenamiento de Logs de funcionamiento del frameworks.		X	

Continuará



Continuación

EXTENSIBILIDAD Y OPCIONES ADICIONALES			
Integración con otras herramientas a través de plugins.	X	X	
Soporte para Web Services.	X	X	X
Soporte para el envío de correo electrónico.	X	X	
SOPORTE PHP			
Soporte para PHP 4.	X	X	
Soporte para PHP 5.	X	X	
CARACTERÍSTICAS ADICIONALES			
Licencias libres.	X	X	X
Comunidad activa de usuarios.	X	X	X

Realizado por: HUEBLA, Byron, 2018

- **Estructura de directorios y fundamentos en laravel 5.1**

Todos los proyectos nuevos en Laravel 5.1 tienen un total de 22 componentes en su estructura al momento de su creación, a continuación, anoto alguno de ellos:

- ✓ app/
- ✓ bootstrap/
- ✓ config/
- ✓ database/
- ✓ public/
- ✓ resources/
- ✓ storage/
- ✓ tests/
- ✓ artisan
- ✓ readme.md
- ✓ server.php

A continuación, describiré los directores y archivos más importantes para que nos ayude a entender más el funcionamiento del framework.

## **El directorio app**

App es usado para ofrecer un lugar por defecto a todo el código personal del proyecto eso incluye clases que puedan ofrecer funcionalidad a la aplicación, archivos de configuración y más. Es considerado el directorio más importante de nuestro proyecto ya que es en el que más trabajaremos.

El directorio app tiene a su vez otros subdirectores importantes, pero uno de los más utilizados es el directorio Http en el cual publicaremos de nuestros Controllers, Middlewares y Requests en sus carpetas correspondientes, además dentro del subdirectorio encontramos el archivo routes.php donde describiremos las rutas de la aplicación.

A nivel de la raíz del directorio app encontraremos el modelo User.php, los modelos comúnmente se ubicarán a nivel de la raíz de la carpeta app aunque iguales es posible estructurarlos de forma que queramos puede ser una carpeta es llamada Modelos.

## **El directorio config**

La configuración tanto por el Frameworks como para mi aplicación se mantiene en este directorio. La configuración de LARAVEL existe como un conjunto de archivos PHP que contiene matrices. Entre los archivos más usados del directorio Config se encuentra:

- ✓ app.php: en este archivo nos puede interesar configurar el lenguaje de nuestra aplicación, la zona horaria, los providers y las clases más comunes.
- ✓ database.php: en este archivo podemos configurar principalmente el motor de base de datos al cual deseamos conectarnos.

## **El directorio database**

Aquí encontrarán los archivos relacionados con el manejo de la base datos. Dentro de este directorio se encuentran los subdirectorios:

- ✓ factories: aquí escribí más nuestro model factories.
- ✓ Migrations: todas las migraciones que creamos se ubican en este directorio.
- ✓ seeds: contiene todas las clases de tipo seed.

## **El directorio public**

Dentro de este directorio colocaremos todos los recursos estáticos de nuestra aplicación, es decir, archivos css, js, imágenes y fuentes.

Es recomendable crear una carpeta por cada tipo de recurso.

## **El directorio resources**

Dentro de este directorio se encuentran todos los subdirectorios:

- ✓ assets: este tipo de archivos nos permite desarrollar el front-end.
- ✓ lang: este archivo no sirve para cambiar nuestro proyecto de un idioma a otro.
- ✓ views: aquí ubicar hemos nuestras vistas en formato php, ese recomendable crear una carpeta por cada controlador.

## **El directorio storage**

Cuando Laravel necesita escribir algo en el disco, lo hace en el directorio storage. Por este motivo el servidor web debe poder escribir en esta ubicación. Aquí podemos encontrar otros directorios entre los cuales el más relevante es el subdirectorio framework, es ahí donde se almacena el caché y las vistas compiladas.

## **El directorio test**

Aquí describiremos los archivos de pruebas que serán ejecutadas posteriormente por PHPUNIT.

## **El archivo .env y .env.example**

El archivo .env no existe cuando instalamos Laravel, en este archivo se configurará el modo en que se ejecuta nuestra aplicación, por defecto será el modo debug, además podemos configurar la conexión a la base de datos y la conexión con el servidor de correo electrónico. El archivo .env lo creamos copiando el archivo .env.example y renombrando la copia como .env.

#### **1.6.4. PHP**

##### **Historia de PHP**

Como la mayoría del software libre, PHP pertenecer a la comunidad. Una gran cantidad de personas ha ayudado a lo largo de su vida a crear tanto el núcleo del lenguaje como la enorme cantidad de librerías que dispone. Sin embargo, debemos atribuir su creación originalmente a Rasmus Lerdorf, creador del lenguaje en 1994.

PHP fue escrito en lenguaje C que permitía de interpretación de un número limitado de comandos. El sistema fue denominado Personal Home Page Tools y adquirió relativo éxito gracias a que otras personas solicitaron a su propietario que les permitiese utilizar sus programas en sus propias páginas.

Dada la situación del primer PHP y de manera adicional su creador diseñó un sistema para procesar formularios y el conjunto de estas herramientas serie la primera versión del lenguaje PHP.

##### **PHP 3**

La siguiente gran contribución al lenguaje se realizó a mediados del 97 cuando se volvió a programar el analizador sintáctico, se incluyeron nuevas funcionalidades como el soporte a nuevos protocolos de internet y el soporte a la gran mayoría de las bases de datos comerciales. Todas estas mejoras sentaron las bases de PHP versión 3.

A pesar que por el aquel entonces el lenguaje tenía un largo camino por delante para convertirse en una herramienta indispensable, integraban una nutrida cantidad de funcionalidades de modo que su comunidad de programadores fue creciendo, atraída por su utilidad y la facilidad para comenzar a desarrollar webs.

##### **PHP 4**

En su versión 4 incorporó como novedad el motor “Zend”, desarrollado con mayor meditación para cubrir las necesidades de aquel momento y solucionar algunos inconvenientes de la anterior versión. Algunas mejoras de esta nueva versión son su rapidez gracias a que primero se compila y luego se ejecuta, mientras que antes se ejecutaba mientras se interpretaba el código, su mayor independencia

del servidor web creando versiones de PHP nativas para más plataformas y un API más elaborado y con más funciones.

## **PHP 5**

Sin embargo la madurez definitiva de PHP llegó con la versión 5, que permaneció durante más de once años en el mercado y hoy en día todavía se encuentra el mantenimiento. La principal novedad de la versión 5 fue una mejorada integración del paradigma de la programación orientada a objetos.

Durante todos los años de vida de PHP 5 hubo muchos cambios. Multitud de herramientas se agregaron al lenguaje, permitiendo hacer cosas que eran totalmente demandadas por los desarrolladores y que otros lenguajes más nuevos habrían incorporado de salida, uno de los ejemplos más claros fue el AUTOLOAD de clases, lo que permitió la incorporación del gestor de paquetes Composer.

## **PHP 6**

Sin embargo, once años con la misma versión sugería que lenguaje se había estancado y PHP fue perdiendo adeptos, aunque en términos estadísticos, número de desarrolladores y demanda laboral, su superioridad sigue siendo abrumadora. Entre todo ese tiempo varias situaciones hicieron que no se llegara a presentar PHP 6 y finalmente la comunidad decidió saltar ese número de versión y lanzar directamente PHP 7

## **PHP**

Es el acrónimo de Hipertext Procesor, es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido con una gran librería y mucha documentación.

Un lenguaje del lado del servidor es aquel que se ejecuta en el servidor web justo antes de que se envíe la página a través de Internet al cliente. Las páginas que se ejecutan en el servidor pueden realizar accesos a base de datos, conexiones en red, y otras tareas para crear la página final que verá el cliente.

El cliente solamente recibe una gran página con el código HTML resultante de la ejecución de la PHP. Como la página resultante contiene únicamente código HTML, es compatible con todos los navegadores.

## **Características**

Estas son algunas de sus características de su formidable popularidad:

- Velocidad: PHP no solo es rápido al ser ejecutado, sino que no genera retrasos en la máquina, por esto no requiere grandes recursos del sistema.
- PHP se integra muy bien junto a otras aplicaciones, especialmente bajo ambientes Unix.
- PHP utiliza su propio sistema de recursos y posee de un sofisticado método de manejo de variables, conformando un sistema robusto y estable.
- Seguridad: PHP maneja distintos niveles de seguridad, estos pueden ser configurados desde el archivo .ini.
- Variedad de herramientas para aprender.
- Permite programación orientada a objetos.
- Destaca su conectividad con MySQL y PostgreSQL.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.

## **Ventajas de PHP**

- ✓ Es un lenguaje multiplataforma.
- ✓ Orientada al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.
- ✓ No requiere ningún tipo de licencias.
- ✓ Puede funcionar en un servidor Windows o Linux.
- ✓ El código fuente escrito en PHP es invisible al navegador y al cliente.
- ✓ Ofrece la integración con varias bibliotecas externas, que permiten que el desarrollador haga lo que sea, así como generar documentos en pdf hasta analizar código XML. (Vargas, 2001)

## **Desventajas de PHP**

- ✓ Como es un lenguaje que se interpreta en ejecución para ciertos usos puede resultar un inconveniente que el código fuente no pueda ser ocultado.
- ✓ Es muy difícil de optimizar
- ✓ Promueve creación de código desordenado y con un mantenimiento complejo.

- **MVC**

La arquitectura en la cual se encuentra construida el sistema es la arquitectura Cliente Servidor, haciendo uso de los patrones Modelo Vista Controlador (MVC), el mismo que ha sido seleccionado y diseñado con base en objetivos y restricciones que se han presentad en el desarrollo del sistema.

Este modelo se basa principalmente en la separación del código en tres capas diferentes, limitadas por sus responsabilidades, en lo que se llaman Modelos, Vistas y Controladores. A continuación, describiré cada una de sus capas:

- ✓ **Modelo**

El modelo es un conjunto de clases que representa la información del mundo real que el sistema debe reflejar. Es la parte encargada de representar la lógica de negocio de una aplicación.

El modelo, a nivel teórico no debe tener conocimiento acerca de la existencia de las vistas y del controlador, es decir que es la capa donde se traba con los datos, contiene todas las funciones que accederán a las entidades y harán las correspondientes inserciones, actualizaciones, eliminaciones, cargas, etc.

- ✓ **Vista**

Son la representación visual de los datos, todo lo que tenga que ver con la interfaz gráfica va aquí. Ni el modelo ni el controlador se preocupan de cómo se verán los datos, esa responsabilidad es únicamente de la vista., la cual utiliza un dialecto de acceso a datos basado en clases y objetos del lenguaje de programación, base de datos y frameworks utilizados.

## ✓ **Controlador**

Es una capa que sirve de enlace entre las vistas y los modelos, respondiendo a los mecanismos que puedan requerirse para implementar las necesidades de nuestra aplicación. Sin embargo, su responsabilidad no es manipular directamente datos, ni mostrar ningún tipo de salida, sino servir de enlace entre los modelos y las vistas para implementar las diversas necesidades del desarrollo. Contiene el código necesario para responder a las peticiones que se envían en la aplicación, como editar un documento o un registro en la base de datos. Controla las peticiones del usuario, y controla el flujo de ejecución del sistema. (Malán Guaranga, 2016)

La colaboración entre los elementos que forman una aplicación MVC, los clientes o usuarios mediante la red y por medio de autenticación a nivel de aplicación tendrán acceso al servidor de aplicación y al servidor de base de datos haciendo uso de la arquitectura cliente – servidor.

### ***1.6.5. Metodología de desarrollo SCRUM***

La metodología ágil SCRUM está ligada a la utilización de mejores técnicas de trabajo en equipo donde el entendimiento entre el equipo de desarrollo y el usuario es prioritario para obtener resultados positivos del proyecto.

Las técnicas propuestas por Scrum están relacionadas unas con otras con el fin de dar resultados altamente beneficiosos para el desarrollo del sistema. Permite realizar entregas periódicamente de los avances del proyecto, manteniendo informado al usuario durante el desarrollo del proyecto.

Además, Scrum está diseñado para proyectos grandes medianos y pequeños, en los cuales la obtención de resultados es prioritaria y los requerimientos por parte del usuario pueden ser cambiantes.

- **Características de scrum**

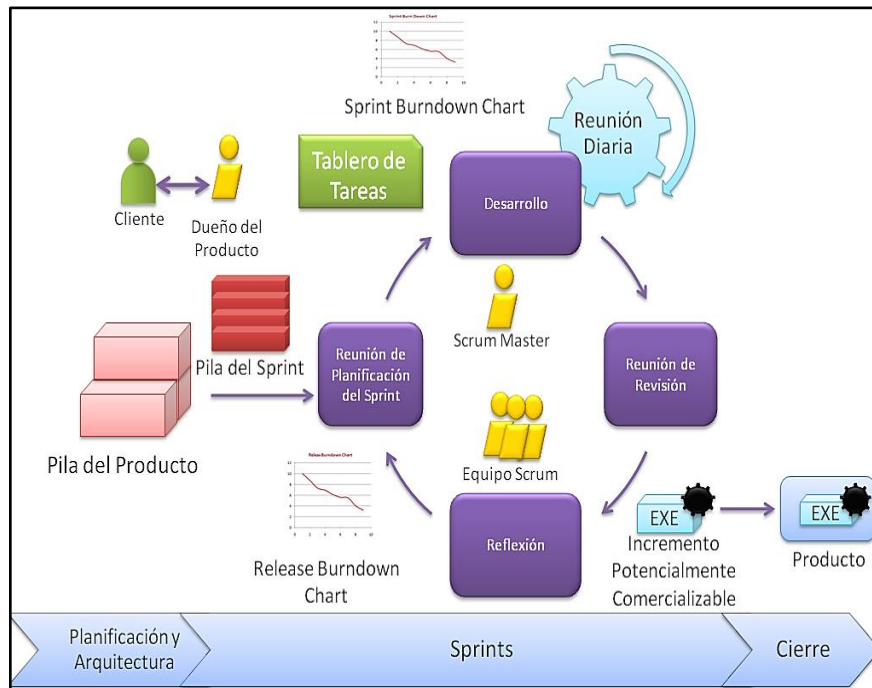
Scrum establece un modelo a seguir durante el desarrollo de todo el proyecto, es el punto de partir el cual registrará las actividades que se realizarán a lo largo del proyecto. Se deben establecer roles dentro de los denominados “teams” a los equipos de trabajo.



Lo primero que se debe definir antes de empezar el trabajo es el Product-Backlog que es una representación de los requisitos que definieron conjuntamente con el cliente al inicio del proyecto, estas actividades a realizar deben estar priorizadas, a todo este conjunto de actividades se las suele dividir en “sprints”, es decir a lo largo del proyecto dependiendo del tamaño del mismo se va a tener algunos sprints, cada sprint debe arrojar como producto un entregable.

Dentro de la metodología se contempla también lo que es un Sprit Planning, esta es una reunión en la que se define qué elementos del Product-Backlog forman cada sprint, además en esta reunión de dueño del producto determina que elementos desea ver terminados.

El proceso de desarrollo mediante la metodología Scrum, contiene las siguientes actividades:



**Figura 1-1:** Proceso de Scrum

Fuente: <http://etherpad.proyectolatin.org/up/aaea7098b048d17f4bbe2a2bb599abeb.png>

- **Roles scrum**

Como se mencionó anteriormente Scrum organiza un equipo de trabajo, que se conoce como team y dentro del cual lógicamente se definen roles. Dentro de los principales tenemos:

✓ **Scrum master**

Más que un jefe es un líder, es el encargado de guiar al equipo y eliminar las barreras que impidan que el equipo complete satisfactoriamente cada sprint.

✓ **Product owner**

Es el encargado de velar por los intereses del cliente, este es el encargado de crear las historias de usuario, priorizarlas y colocarlas dentro del Product Backlog.

✓ **Team**

Es el equipo de trabajo encargado de ejecutar las historias de usuario, normalmente establecido por grupos de 3 – 8 personas, son encargados de entregar los productos.

✓ **Stakeholders**

Normalmente es el cliente propiamente dicho, es para quien se está desarrollando el proyecto, únicamente participan en lo que son las revisiones de cada sprint.

• **Ciclo de vida de scrum**

Este ciclo se encuentra conformado por 4 etapas o fases, las cuales se describen a continuación:

✓ **Planificación**

En esta fase se define todo el trabajo que se va a realizar, además de los equipos de trabajo y los roles de cada miembro dentro de cada equipo.

✓ **Diseño**

En esta fase se definen normalmente los detalles técnicos como: diseño de base de datos, Frameworks, versionadores de código y demás herramientas necesarias para el proyecto.

✓ **Codificación**

Se desarrollan los sprint de cada iteración, es aquí donde realmente empieza a convivir el entorno de trabajo del proyecto, cada entregable al final del proyecto se convierte en el producto final que se entrega a los stakeholders.

✓ **Pruebas**

Normalmente se suele realizar antes de la entrega de cada sprint, aunque al final del proyecto se puede hacer una revisión general del producto a entregar.

**1.6.6. ISO/IEC 9126-3 calidad del software**

Es un estándar internacional para la evaluación de la calidad del software, enfatiza los siguientes puntos importantes:

- Los requisitos del software constituyen el fundamento para medir la calidad.
- Los estándares utilizados definen un conjunto de criterios que guían la manera en que el software se somete a evaluación.

El estándar ISO/IEC 9126-3 define las características de calidad y el modelo del proceso de evaluación del software. A continuación, se analizará las características de calidad que evalúan el estándar. Ver (**Tabla 3-1**)

**Tabla 3-1:** Características de calidad del software

CARACTERÍSTICAS	DESCRIPCIÓN
Funcionalidad	Capacidad del software para proporcionar funciones que satisfagan las necesidades especificadas e implícitas cuando el software se utiliza en las condiciones especificadas.
Fiabilidad	Capacidad del software a mantener un nivel especificado de rendimiento cuando se utiliza en las condiciones especificadas.
Usabilidad	Capacidad del software de facilitar al usuario la interacción y el aprendizaje de una manera atractiva para el usuario.

*Continuará*

*Continuación*

Eficiencia	Capacidad del software para proporcionar el rendimiento apropiado, relativo a la cantidad de recursos utilizados, bajo las especificaciones determinadas.
Mantenibilidad	Capacidad del software para ser modificado, ya que éstas deben estar sujetas a correcciones , mejoras o adaptación del software a cambios es el entorno.
Portabilidad	Capacidad del software de ser transferido de un entorno a otro.

**Realizado por:** HUEBLA, Byron, 2018

Luego de realizar el análisis anteriormente en la tabla 3:1 se ha seleccionado tres características a las cuales se aplicará el modelo de proceso para la evaluación de calidad en el sistema Diller, las cuales están descritas a continuación:

1. **Funcionalidad:** es un conjunto de atributos que se relaciona con una existencia de un conjunto de funciones y sus propiedades específicas, las funciones son aquellos que satisfacción las necesidades implícitas. casco explícitas
2. **Usabilidad:** es un conjunto de tributos relacionados con el esfuerzo necesario para su uso, y en la valoración individual de tal uso, por un establecido o implicado conjunto de usuarios.
3. **Eficiencia:** es un conjunto de atributos relacionados con la relación entre el nivel de desempeño del software y la cantidad de recursos necesarios bajo condiciones establecidas.

## CAPÍTULO II

### 2. MARCO METODOLÓGICO

#### 2.1. Introducción

El presente trabajo, describe el desarrollo de un Sistema Académico orientado a la web para la Unidad Educativa Fiscal “Interandina” utilizando el framework Laravel y MySQL en la comunidad Santa Teresita de Guabug de la parroquia San Juan del cantón Riobamba denominado “Diller” para lo cual se aplicó la metodología de desarrollo ágil SCRUM, es así como la descripción de cada una de las actividades relacionadas en el proceso de construcción de un sistema acorde a las exigencias del Product Owner, mediante el cumplimiento de cada uno de los requerimientos representando en un Sprint del proyecto.

#### 2.2. Aplicación de la metodología de desarrollo scrum

La metodología ágil SCRUM realiza entregas parciales y regulares del producto final, priorizando el beneficio que aportan al usuario del sistema, se utiliza además para la resolución de situaciones en que no se está entregando al cliente lo que se acordó en entregarle. Cuando las entregas se ven muy alargadas, los costes se disparan o la calidad no es aceptable, cuando se necesita capacidad de reacción ante la competencia o cuando la autoestima de los integrantes está por debajo de lo normal y la rotación es elevada, cuando es necesario identificar y solucionar incoherencias o cuando es necesario trabajar utilizando un proceso especializado en el desarrollo del producto, SCRUM se basa en tres fases descritas anteriormente.

#### 2.3. Fase de planificación

##### 2.3.1. *Identificación de personas y roles involucrados en el proyecto*

El equipo involucrado en la gestión y desarrollo del proyecto está conformado por el dueño del producto (Product Owner), el equipo de desarrollo y el director del proyecto (Scrum Master), el detalle de los participantes se encuentra detallada a continuación en la **Tabla 1-2**.

**Tabla 1-2:** Roles del equipo

INTEGRANTES	ROL	CONTACTO
Unidad Educativa Fiscal “Interandina” representado por Magister. Humberto Pagalo	Product Owner	0991257133
Ing. Patricio Adolfo Moreno	Scrum Master	0998700872
Byron Huebla Socag	Team	0998480154

Realizado por: HUEBLA, Byron, 2018

### 2.3.2. Tipos de usuarios y roles de usuario en el sistema

Los usuarios son todas aquellas personas involucrados directamente con el uso del sistema, por lo que a continuación se definirá un tipo de usuario y su rolen el sistema a continuación **Tabla 2-2**.

**Tabla 2-2:** Tipos de usuarios en el sistema

TIPO DE USUARIO	DESCRIPCIÓN	RESPONSABILIDAD	RESPONSABLE
Usuario Administrador del Sistema	Persona con acceso total hacia el Sistema Diller.	Administrar la funcionalidad del sistema(gestionar usuarios, permisos, mantenimiento de acorde a los requerimientos establecidos)	Desarrollador, Administrador Educativo
Usuario Profesor	Encargado de administrar las notas de los estudiantes	Ingreso modificación de notas, promedios, asistencia, material, conducta	Profesor
Usuario Estudiante	Visualización	Visualización de notas, horario de clases, calendario académico, asistencia así como la edición de sus datos correspondientes	Estudiante
Usuario Padre de familia	Visualización	Visualización de notas de sus representados, horario de clases, calendario académico, asistencia así como la edición de sus datos correspondientes	Padre de familia
Usuario Bibliotecario	Encargado de la administración de Libros	Visualización de Libros prestados, total de libros disponibles y prestamistas	Bibliotecario

Realizado por: HUEBLA, Byron, 2018

### 2.3.3. Actividades del proyecto

Las actividades desarrolladas antes, durante y después del desarrollo del presente trabajo de titulación, se encuentran detalladas en la siguiente **Tabla 3-2** a continuación:

**Tabla 3-2:** Actividades desarrolladas en el proyecto

INSTITUCIÓN	ACTIVIDAD	RESPONSABLE
ESPOCH	Recopilación de la información para establecer los requerimientos del sistema	Desarrollador
	Planificación de la solución	Desarrollador
	Diseño de la arquitectura	Desarrollador
	Diseño de la Base de Datos	Desarrollador
	Diseño de los módulos	Desarrollador
ESPOCH	Diseño de Interfaces	Desarrollador
	Pruebas de Calidad y rendimiento	Desarrollador
	Documentación del Sistema	Desarrollador
	Representación de Resultados	Desarrollador
	Capacitación a los usuarios finales	Desarrollador

Realizado por: HUEBLA, Byron, 2018

### 2.3.4. Product backlog

Las historias de usuario (Product backlog), son los requerimientos generados por el equipo de desarrollo SCRUM para definir las actividades realizadas y determinar el número de Sprint (iteraciones) del proyecto a continuación, se detalla cada uno en la **Tabla 4-2**.

**Tabla 4-2:** Product backlog

Historias Técnicas			
Número	Descripción	Horas	Prioridad
HT-01	Como desarrollador necesito realizar el análisis, recolección de información para poder definir el planteamiento de la solución.	25	Alta
HT-02	Como desarrollador necesito realizar el diseño de la Arquitectura del Sistema.	24	Alta
HT-03	Como desarrollador necesito realizar el diseño de la Base de Datos	26	Alta
HT-04	Como desarrollador necesito realizar el diseño de los módulos que tendrá el Sistema.	21	Alta
HT-05	Como desarrollador necesito realizar el diseño de la Interfaz.	22	Alta
HT-06	Como desarrollador necesito instalar y configurar las herramientas necesarias para el desarrollo del sistema.	21	Alta
HT-07	Como desarrollador necesito capacitar a los usuarios que harán uso del Sistema.	21	Alta

*Continuará*

*Continuación*

HT-08	Como desarrollador necesito realizar la documentación del Sistema.	36	Alta
Historias de Usuario			
Número	Descripción	Horas	Prioridad
HU-01	Como desarrollador necesito crear un proceso que permita el registro de estudiantes.	31	Alta
HU-02	Como desarrollador necesito crear un proceso que permita el registro múltiple de estudiantes mediante la carga de un archivo csv.	32	Alta
HU-03	Como desarrollador necesito crear un proceso que permita listar los estudiantes de acuerdo a la clase en que se encuentre matriculado.	34	Alta
HU-04	Como desarrollador necesito crear un proceso que permita modificar los datos de estudiantes.	19	Media
HU-05	Como desarrollador necesito crear un proceso que permita eliminar el registro de los estudiantes.	10	Baja
HU-06	Como desarrollador necesito crear un proceso que permita realizar la matriculación de un estudiante.	34	Alta
HU-07	Como desarrollador necesito crear un proceso que permita a los estudiantes visualizar los horario de clases.	35	Alta
HU-08	Como desarrollador necesito crear un proceso que permita al estudiante visualizar o descargar el material de estudio para retroalimentar su conocimiento	19	Media
HU-09	Como desarrollador necesito crear un proceso que permita al estudiante descargar el Sílabo o Plan Académico de todas las materias que se encuentra cursando.	35	Alta
HU-10	Como desarrollador necesito crear un proceso que permita al estudiante visualizar los puntos que obtuvo en sus exámenes y ver sus promedios finales.	18	Media
HU-11	Como desarrollador necesito crear un proceso que permita al bibliotecario visualizar cuantos libros tengo registrados en la Biblioteca.	18	Media
HU-12	Como desarrollador necesito crear un proceso que permita al bibliotecario ver el número de usuarios que solicitaron un Libro, pueden ser estudiantes o maestros.	35	Alta
HU-13	Como desarrollador necesito crear un proceso que permita registrar Libros en la Biblioteca	37	Alta
HU-14	Como desarrollador necesito crear un proceso que permita modificar los datos personales del Bibliotecario	19	Media
HU-15	Como desarrollador necesito crear un proceso que permita el registro de docentes.	35	Alta
HU-16	Como desarrollador necesito crear un proceso que permita listar a los docentes.	36	Alta

*Continuará*



Continuación

HU-17	Como desarrollador necesito crear un proceso que permita modificar los datos de los docentes	19	Media
HU-18	Como desarrollador necesito crear un proceso que permita eliminar el registros de los docentes	14	Baja
HU-19	Como desarrollador necesito crear un proceso que permita visualizar el perfil completo de todos los estudiantes.	32	Alta
HU-20	Como desarrollador necesito crear un proceso que permita visualizar los promedios académicos de los estudiantes.	17	Media
HU-21	Como desarrollador necesito crear un proceso que permita visualizar todos los profesores que estén registrados en el Sistema Académico	19	Media
HU-22	Como desarrollador necesito crear un proceso que permita al docente listar las materias que tenga asignado en cada curso.	32	Alta
HU-23	Como desarrollador necesito crear un proceso que permita al docente subir materiales de apoyo (Material de Estudios) para que el estudiante pueda descargar y retroalimentar sus conocimientos.	30	Alta
HU-24	Como desarrollador necesito crear un proceso que permita al docente subir el Sílabo o Plan Académico de todas las materias que tenga asignado a su cargo.	30	Alta
HU-25	Como desarrollador necesito crear un proceso que permita al docente listar los horarios académicos.	32	Alta
HU-26	Como desarrollador necesito crear un proceso que permita al docente registrar la asistencia diaria de los estudiantes	32	Alta
HU-27	Como desarrollador necesito crear un proceso que permita al docente asignar notas para su promedios finales.	32	Alta
HU-28	Como desarrollador necesito crear un proceso que permita el registro un representante del estudiante.	32	Alta
HU-29	Como desarrollador necesito crear un proceso que permita modificar los datos del representante del estudiante.	19	Media
HU-30	Como desarrollador necesito crear un proceso que permita eliminar el registros del representante del estudiante.	13	Baja
HU-31	Como desarrollador necesito crear un proceso que permita al representante, visualizar las asistencias de sus representados.	18	Media
HU-32	Como desarrollador necesito crear un proceso que permita al representante, visualizar el horario de clases de sus representados.	18	Media
HU-33	Como desarrollador necesito crear un proceso que permita al representante, visualizar las notas de los exámenes de sus representados.	19	Media
HU-34	Como desarrollador necesito crear un proceso que permita acceder al sistema a todos los usuarios registrados mediante sus credenciales asignadas.	35	Alta

Continuará

*Continuación*

HU-35	Como desarrollador necesito crear un proceso que me permita restablecer la contraseña de acceso al sistema.	35	Alta
HU-36	Como desarrollador necesito crear un proceso que permita visualizar en tiempo real el número de estudiantes, padres y profesores registrados en el sistema	35	Alta
HU-37	Como desarrollador necesito crear un proceso que permita el administrador podrá promover al estudiante al siguiente año lectivo	28	Alta
HU-38	Como desarrollador necesito crear un proceso que permita crear los exámenes al Administrador	24	Alta
HU-39	Como desarrollador necesito crear un proceso para que el administrador pueda crear las reglas de calificaciones.	24	Alta
HU-40	Como desarrollador necesito crear un proceso que permitan a todos los usuarios interactuar internamente a través de un chat académico.	24	Alta
HU-41	Como desarrollador necesito crear un proceso que permita al administrador editar la página Web de la Institución.	24	Alta

**Realizado por:** HUEBLA, Byron, 2018

Luego de haber obtenido los requerimientos del software mediante la priorización de requerimientos que sugiere la metodología ágil SCRUM se deduce que todos estos requerimientos son factibles de poder desarrollar.

### **2.3.5. Definición de Módulos del Sistema**

Previo a la Construcción de un sistema informático, este debe ser analizado y subdividido en componentes de manera separada denominados módulos que abarquen todos los requerimientos a fin de llevar un desarrollo organizado. Frente a un Problema es más fácil si se los maneja en pequeñas piezas.

Para realizar esta subdivisión se ha realizado un análisis desde el punto de vista de programación, los cuales han sido comunicados al producto owner del Sistema. Este análisis se lo ha hecho a partir de las historias de usuario e historias técnicas definidas previamente en base a criterios de funcionalidad de los principales usuarios del sistema (Administrador, Docente, Estudiante y Padres de familia) de los cuales han surgido los siguientes módulos.

**Tabla 5-2: Módulos del Sistema**

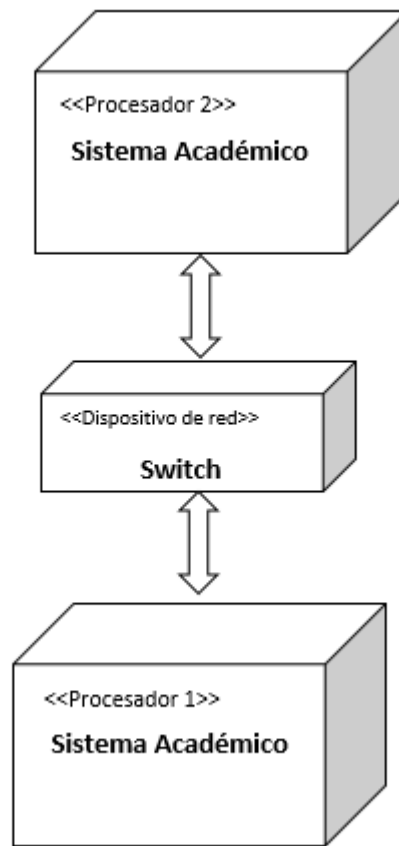
CONFIGURACIÓN	ACADÉMICO	MATRICULACIÓN	CALIFICACIONES	REPORTES
Gestión del Panel Principal	Gestión docentes	Ingreso de datos estudiante	Ingreso de calificaciones	Listado de estudiantes
Normativa de evaluación	Creación cursos	Ingreso de datos del representante	Generación de exámenes en Línea	Listado de docentes
Interfaz Web	Creación de secciones	Asignación de curso	Registro de Asistencia	Listado de asistencia
	Creación de materias	Asignación de docente		Listado de asistencia
	Promoción del estudiante	Asignación de materias		Listado de comportamiento
	Gestión Bibliotecario	Asignación de horario de clases		
	Plan académico			
	Gestión Horario			
	Gestión Exámenes			

Realizado por: HUEBLA, Byron, 2018

### 2.3.6. *Arquitectura del Sistema*

La arquitectura de un sistema de software es el conjunto de disposiciones de diseño tomadas para el sistema. Haciendo una analogía, en la construcción de edificios, se puede decir que la arquitectura representa los planos (de diseño, de organización y de construcción) del software. La arquitectura de software ha sido definida como un medio para documentar las decisiones de diseño de alto nivel para un software.

A continuación, exhibimos el diseño de la arquitectura a implementar para el sistema académico de la Unidad Educativa Experimental “Interandina”.

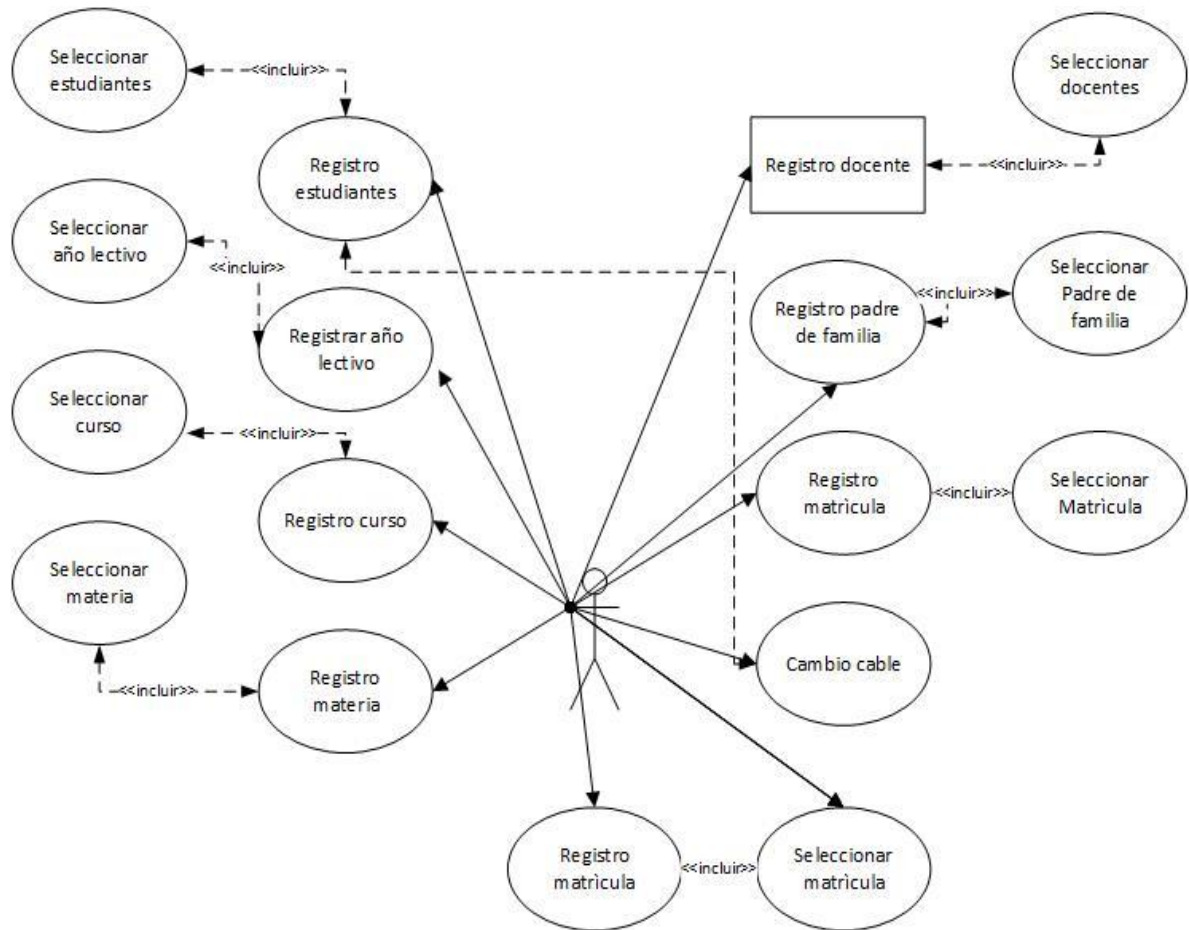


**Figura 1-2:** Arquitectura del Sistema

Realizado por: Byron Huebla

### 2.3.7. Diagrama de caso de uso

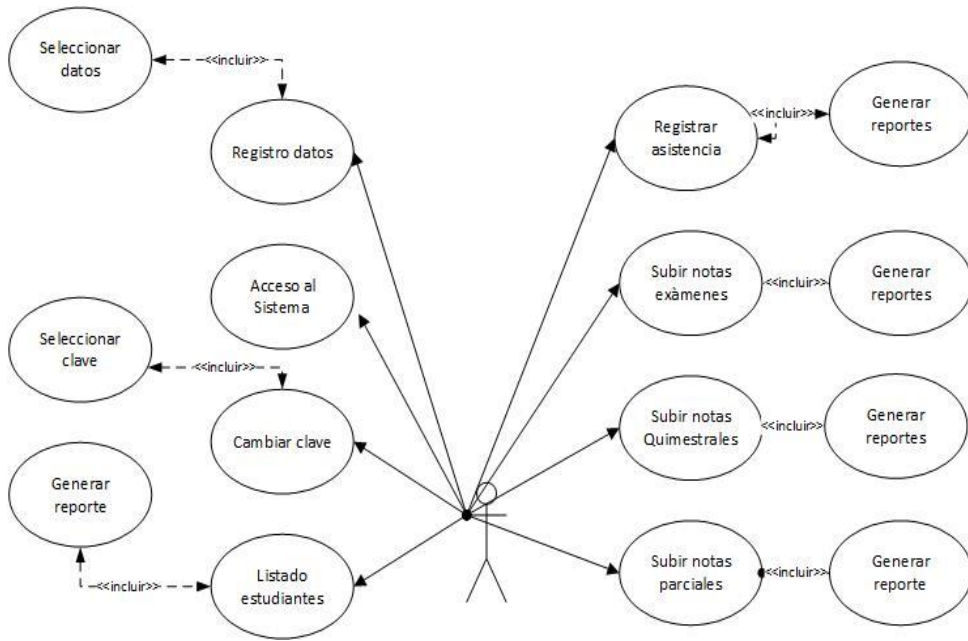
El diagrama de casos de uso del usuario administrador muestra las acciones que efectúa el administrador como son el registro de estudiantes, docente, materia, curso, año lectivo, matrícula, cambio de clave de los usuarios.



**Figura 2-2:** Diagrama de caso del Administrador

**Realizado por:** Byron Huebla

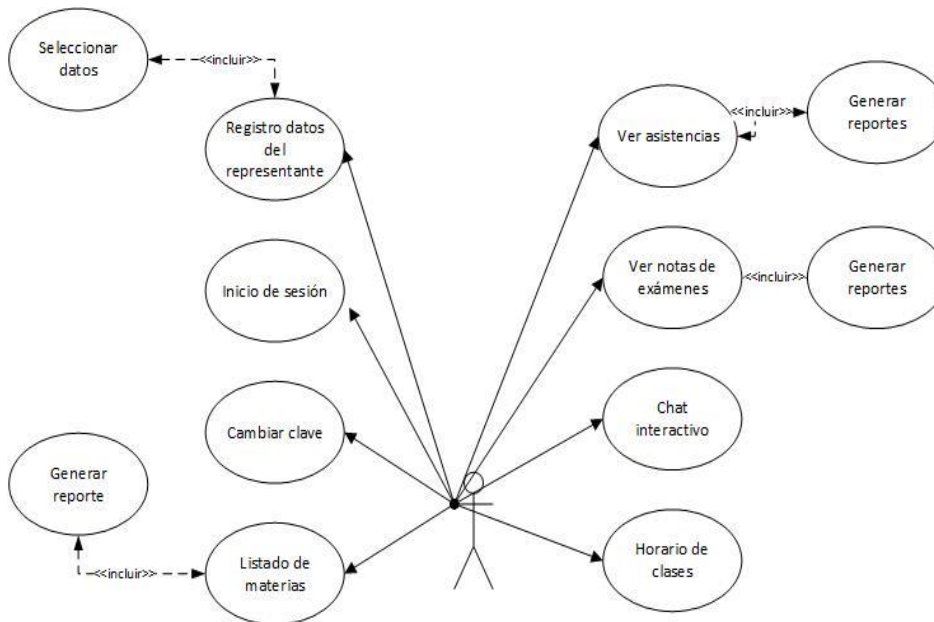
El diagrama de caso de uso del usuario docente muestra las actividades que realiza el docente, como el registro de datos personales, asistencias, notas quimestrales, notas de exámenes, y generar los reportes.



**Figura 3-2:** Diagrama de caso del Docente

Realizado por: Byron Huebla

El diagrama de caso de uso del usuario estudiante muestra las actividades que realiza cada estudiante, como el registro de datos personales, datos del representante, y generar algunos reportes.



**Figura 4-2:** Diagrama de caso del estudiante

Realizado por: Byron Huebla

### 2.3.8. Diagrama de Clases

Muestra el sistema en sus diferentes partes lógicas (Clases) y las relaciones de las mismas. Esto describe a través de una serie de diagramas el funcionamiento y el centro de estructura del Sistema.

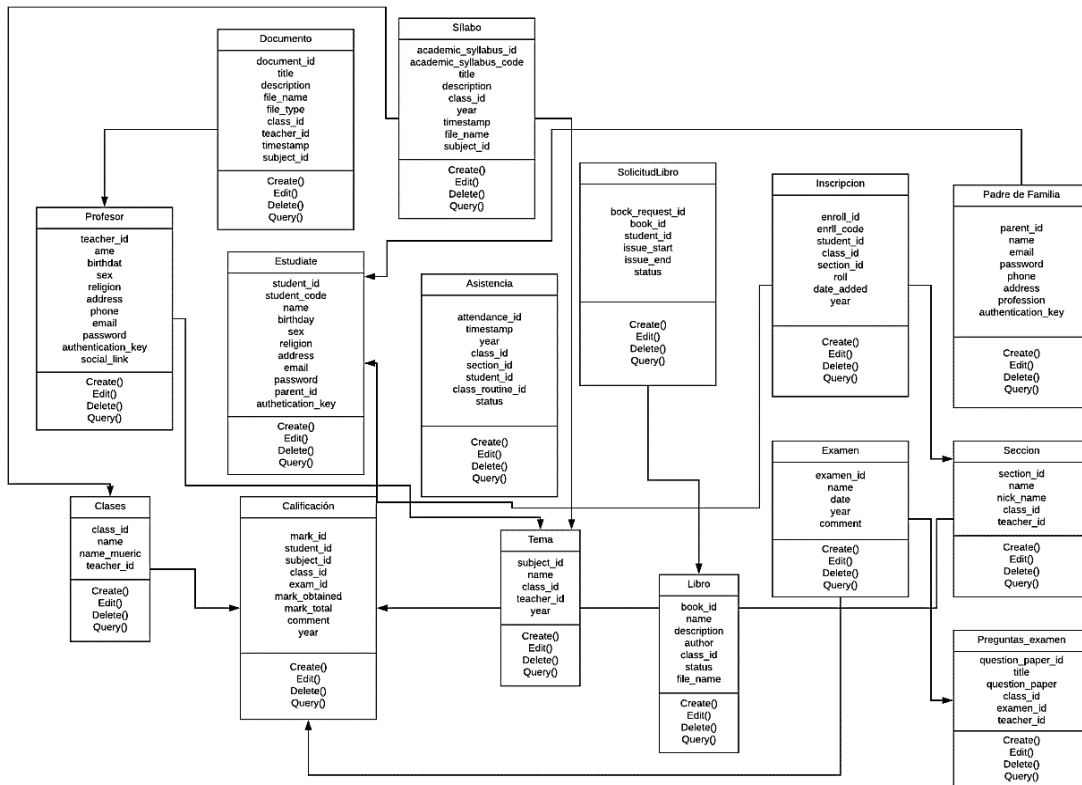


Figura 5-2: Diagrama de clases

Realizado por: Byron Huebla

### 2.3.9. Sprint backlog (planificación)

Los Sprint del proyecto se encuentran distribuidos en la **Tabla 5-2**, en donde cada Sprint cuenta con un cierto número de historias de usuario y técnicas, agrupadas en un rango de tiempo que van de veinte días para el desarrollo del Sistema en mención.

**Tabla 6-2: Sprint backlog**

<b>Sprint</b>	<b>Historia</b>	<b>Responsable</b>	<b>Inicio</b>	<b>Fin</b>	<b>Horas</b>
0	HT-01	Desarrollador	02/01/2018	06/01/2018	25
	HT-02	Desarrollador	06/01/2018	10/01/2018	24
	HT-03	Desarrollador	10/01/2018	14/01/2018	26
	HT-04	Desarrollador	14/01/2018	18/01/2018	21
	HT-05	Desarrollador	18/01/2018	22/01/2018	22
	HT-06	Desarrollador	22/01/2018	26/01/2018	21
	HT-07	Desarrollador	26/01/2018	30/01/2018	21
	HU-01	Desarrollador	30/01/2018	03/02/2018	31
	HU-02	Desarrollador	03/02/2018	07/02/2018	32
	HU-03	Desarrollador	07/02/2018	11/02/2018	34
1	HU-04	Desarrollador	11/02/2018	14/02/2018	19
	HU-05	Desarrollador	14/02/2018	16/02/2018	10
	HU-06	Desarrollador	16/02/2018	20/02/2018	34
2	HU-07	Desarrollador	20/02/2018	24/02/2018	35
	HU-08	Desarrollador	24/02/2018	27/02/2018	19
	HU-09	Desarrollador	27/02/2018	03/03/2018	35
	HU-10	Desarrollador	03/03/2018	06/03/2018	18
	HU-11	Desarrollador	06/03/2018	09/03/2018	18
	HU-12	Desarrollador	09/03/2018	13/03/2018	35
3	HU-13	Desarrollador	13/03/2018	17/03/2018	37
	HU-14	Desarrollador	17/03/2018	20/03/2018	19
	HU-15	Desarrollador	20/03/2018	24/03/2018	35
	HU-16	Desarrollador	24/03/2018	28/03/2018	36
	HU-17	Desarrollador	28/03/2018	31/03/2018	19
	HU-18	Desarrollador	31/03/2018	02/04/2018	14
4	HU-19	Desarrollador	02/04/2018	06/04/2018	32
	HU-20	Desarrollador	06/04/2018	09/04/2018	17
	HU-21	Desarrollador	09/04/2018	12/04/2018	19
	HU-22	Desarrollador	12/04/2018	16/04/2018	32
	HU-23	Desarrollador	16/04/2018	20/04/2018	30
	HU-24	Desarrollador	20/04/2018	24/04/2018	30
5	HU-25	Desarrollador	24/04/2018	28/04/2018	32
	HU-26	Desarrollador	28/04/2018	02/05/2018	32
	HU-27	Desarrollador	02/05/2018	06/05/2018	32
	HU-28	Desarrollador	06/05/2018	10/05/2018	32
	HU-29	Desarrollador	10/05/2018	13/05/2018	19
	HU-30	Desarrollador	13/05/2018	15/05/2018	13

*Continuará*



Continuación

6	HU-31	Desarrollador	15/05/2018	18/05/2018	18
	HU-32	Desarrollador	18/05/2018	21/05/2018	18
	HU-33	Desarrollador	21/05/2018	24/05/2018	19
	HU-34	Desarrollador	24/05/2018	28/05/2018	35
	HU-35	Desarrollador	28/05/2018	01/06/2018	35
	HU-36	Desarrollador	01/06/2018	05/06/2018	35
7	HU-37	Desarrollador	05/06/2018	09/06/2018	28
	HU-38	Desarrollador	09/06/2018	13/06/2018	24
	HU-39	Desarrollador	13/06/2018	17/06/2018	24
	HU-40	Desarrollador	17/06/2018	21/06/2018	24
	HU-41	Desarrollador	21/06/2018	25/06/2018	24
	HT-08	Desarrollador	25/06/2018	30/06/2018	36

Realizado por: HUEBLA, Byron, 2018

### 2.3.10. Cronograma de Actividades

Para dar cumplimiento a los Sprints que contienen el desarrollo del Sistema se ha establecido un cronograma de actividades estimado con una duración de 24 semanas, teniendo como fecha de inicio el 02/01/2018 y fecha fin estimada 15/06/2018

A continuación, en la siguiente **Tabla 6-2** se detalla el cronograma de actividades.

**Tabla 7-2:** Planificación de actividades

ORDEN	ACTIVIDAD	FECHA INICIO	FECHA FIN	RESPONSABLE
1	Entrevistas y determinación del alcance del sistema.	02/01/2018	05/01/2018	Desarrollador
2	Planificación de la solución.	06/01/2018	16/01/2018	Desarrollador
3	Diseño de la arquitectura.	17/01/2018	31/01/2018	Desarrollador
4	Diseño de la Base de Datos.	01/02/2018	10/02/2018	Desarrollador
5	Diseño de los módulos.	11/02/2018	21/02/2018	Desarrollador
6	Diseño de Interfaces.	22/02/2018	04/03/2018	Desarrollador
7	Seleccionar y configurar las herramientas a utilizar.	05/03/2018	12/03/2018	Desarrollador
8	Desarrollo de los requerimientos.	13/03/2018	13/05/2018	Desarrollador
9	Implantación del Sistema.	14/05/2018	27/05/2018	Desarrollador
10	Pruebas de Calidad y rendimiento.	28/05/2018	29/05/2018	Desarrollador
11	Capacitación a los usuarios finales.	30/05/2018	31/05/2018	Desarrollador
12	Documentación del Sistema.	01/06/2018	15/06/2018	Desarrollador

Realizado por: HUEBLA, Byron, 2018

### **2.3.11. Reuniones scrum**

La metodología elegida nos indica que el equipo SCRUM debe mantener reuniones frecuentes para definir las actividades, inquietudes y los cambios de último momento que pueda sugerir el Product Owner.

La primera reunión realizada fue previo a la inicialización del desarrollo en donde se pudo determinar el alcance del proyecto, roles, actividades, tareas y requerimientos de usuario.

Con la finalización de cada Sprint se realizaron reuniones de cierre de Sprint entre todos los participantes con la finalidad de realizar una verificación minuciosa de los productos o historias de usuario que deben ser presentadas y entregadas en las fechas establecidas.

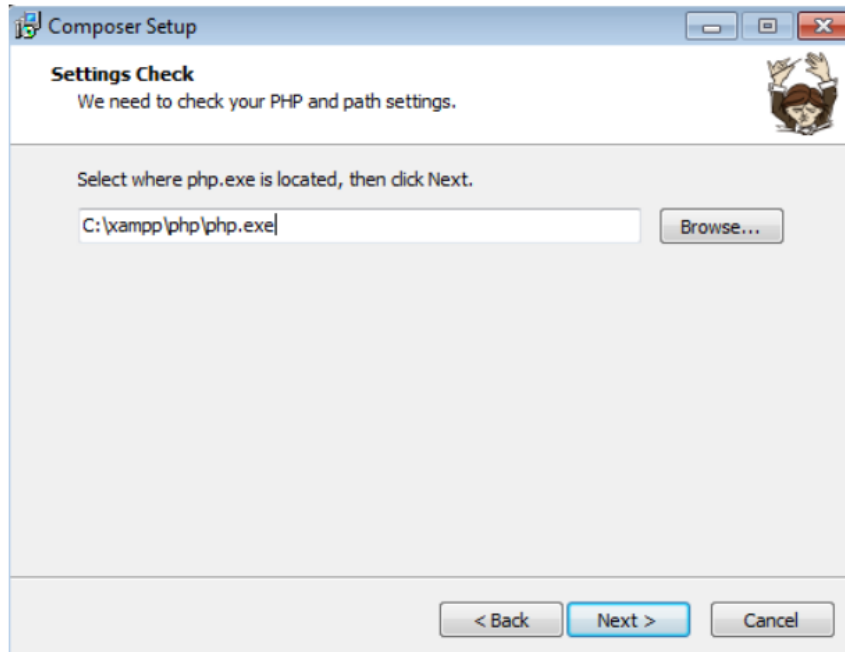
## **2.4. Fase de desarrollo**

En esta sección del documento se detalla cada una de las actividades realizadas en el **“DESARROLLO DE UN SISTEMA ACADÉMICO ORIENTADO A LA WEB PARA LA UNIDAD EDUCATIVA EXPERIMENTAL INTERANDINA UTILIZANDO EL FRAMEWORK LARAVEL Y MYSQL”**

### **2.4.1. Instalación de Laravel**

#### **Instalación de Composer en windows**

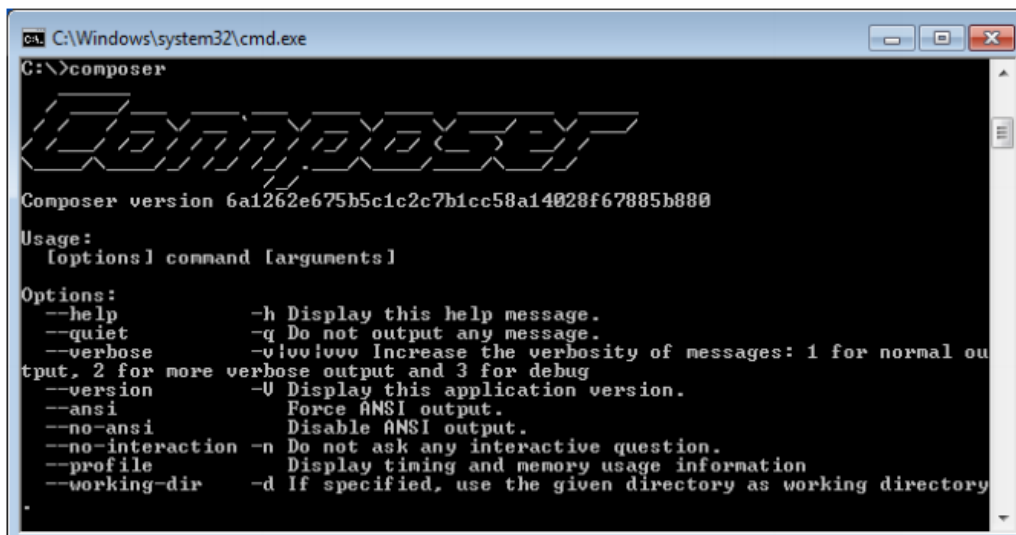
Como primer paso instalar composer en windows la instalación se puede hacer mediante un instalador ejecutable que se descarga en la página de composer. El instalador nos solicitará la ubicación de nuestro php.exe, la cual dependerá de donde hemos instalado XAMPP, como se indica en la Figura 1-1



**Figura 6-2:** Instalación de Composer en Windows

Realizado por: Byron Huebla

El instalador se encargará de modificar la variable PATH para que podamos hacer uso de Composer desde cualquier lugar en la consola. Para probar que la instalación correctamente se ejecuta el siguiente comando(C:\>composer) y nos debería aparecer una lista con todos los posibles comandos que acepta Composer, como indica en la Figura 2-2.



**Figura 7-2:** Prueba instalación Composer

Realizado por: Byron Huebla

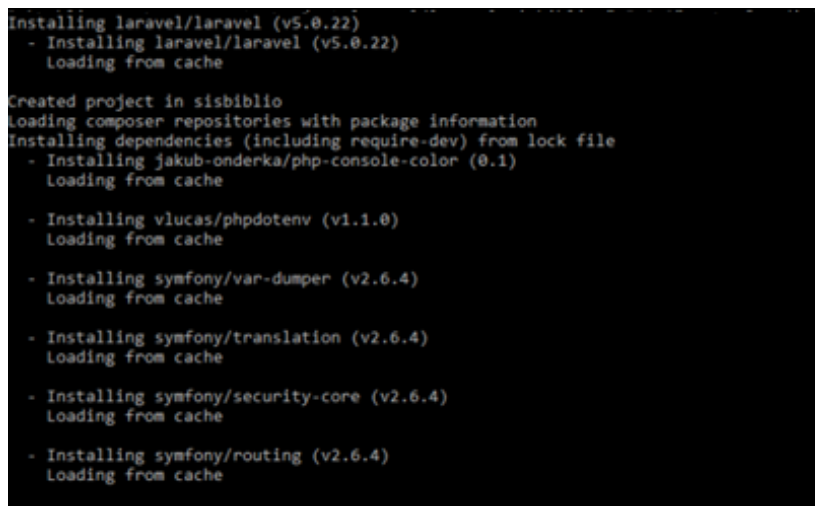
Una vez que ya haya instalado correctamente el composer, se procede a instalar el laravel en el lenguaje php. Para ejecutar aplicaciones con laravel es necesario tener instalado en nuestra computadora un servidor local. Personalmente se recomienda xampp, desde la versión 4 de laravel la creación de un proyecto nuevo se maneja solo con composer.

### **Instalación de laravel.**

Una vez que tengamos composer funcionando correctamente en nuestra computadora podemos descargar una copia de laravel para crear nuestro proyecto. Para esto simplemente abrimos nuestro terminal, vamos a la carpeta que utilice nuestro servidor (htdocs o www) y ejecutamos el siguiente comando (Aplica para cualquier sistema operativo).

```
$ composer create-project laravel/laravel {directory} "~5.0.0" --prefer-dist
```

Este comando descarga una copia completa de la versión más reciente de Laravel desde los repositorios de composer con todas las dependencias y librerías que el framework necesita. El proyecto se creará con el nombre que se le indique en el parámetro, por ejemplo, llamamos escuela, ver la Figura 3-1.



```
Installing laravel/laravel (v5.0.22)
- Installing laravel/laravel (v5.0.22)
  Loading from cache

Created project in sisbiblio
Loading composer repositories with package information
Installing dependencies (including require-dev) from lock file
- Installing jakub-ondrejka/php-console-color (0.1)
  Loading from cache

- Installing vlucas/phpdotenv (v1.1.0)
  Loading from cache

- Installing symfony/var-dumper (v2.6.4)
  Loading from cache

- Installing symfony/translation (v2.6.4)
  Loading from cache

- Installing symfony/security-core (v2.6.4)
  Loading from cache

- Installing symfony/routing (v2.6.4)
  Loading from cache
```

**Figura 8-2:** Instalación de Laravel

**Realizado por:** Byron Huebla

Para probar el proyecto de laravel en la versión 5.1 creado, vamos correr nuestro servidor apache y entrar desde el navegador a localhost/escuela/public y se nos muestra el logo de laravel. Si en caso el

navegador muestra un error de `ErrorException file_put_contents( )`, significa que debemos darle permisos de escritura a la carpeta `/app/storage/`. Esto lo podemos hacer ejecutando el comando `sudo chmod -R 777 storag`.

#### **2.4.2. Descripción de las capas de la aplicación**

El Sistema Académico contará con una estructura basada en tres capas (MVC), cada una detalladas a continuación:

1. **Modelo:** Es todo lo referente a la gestión de la información y la interacción con los datos de nuestra aplicación (Base de Datos), este modelo podrá realizar actualizaciones y depuraciones de los datos gestionados.
2. **Controlador:** Este es el puente entre la vista y el modelo ya que el usuario solicitará información mediante la vista y este pasará por el controlador para posteriormente realizar la petición al modelo. Esta capa también es conocido como la capa de lógica del negocio.
3. **Vista:** Esta capa nos mostrará la información formateada y ordenada, es el resultado de todo lo que el modelo interaccione con los datos, esto lo muestra mediante la interfaz de usuario. Esta capa habitualmente también es conocido como capa de presentación.

#### **2.4.3. Estándar de codificación**

El estándar de codificación nos permite facilitar una correcta utilización de los términos semánticos y procesos de escritura de código, la codificación se realizará mediante el lenguaje de programación PHP, utilizando el framework Laravel, Html5, JQuery, Ajax, Css y Bootstrap. A continuación, detallaré en la **Tabla 7-2** como se nombraron a los diferentes objetos que forman parte de la aplicación.

**Tabla 8-2:** Estándar de codificación

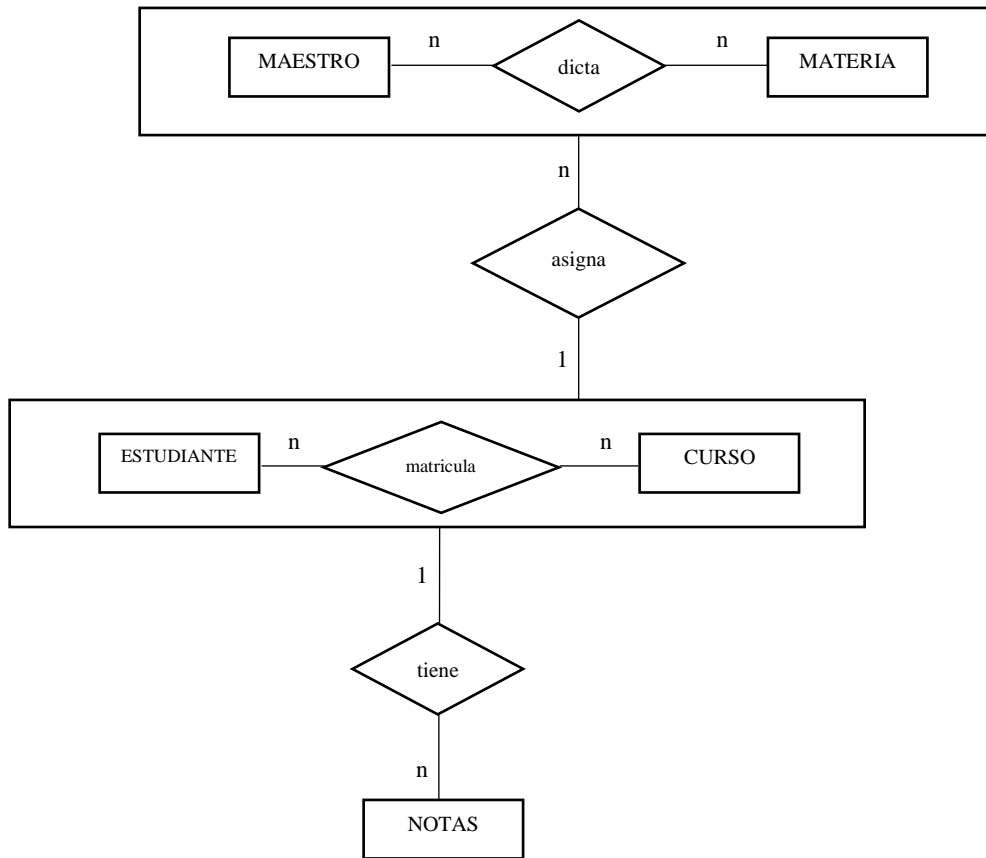
COMPONENTE	NOMBRE	DESCRIPCIÓN
Clases	<Nombreclase> <NombreclaseUnoDos>	La primera letra con mayúscula seguida de letras minúsculas para el nombre de la clase, en caso de ser varias palabras las primeras letras de cada palabra con mayúscula.
VARIABLES	<cont>	Las variables a usar serán declaradas en minúsculas
Objetos declarados	obj_<Nombreobjeto>	Los objetos declarados se los escribirá con el prefijo obj
Paquetes	Acceso_Datos	Los nombres de los paquetes tendrán la primera letra con mayúscula y las demás seguidas de minúsculas separadas por un guion bajo
Botones	btn<Nombre button>	Empieza con el prefijo btn
Label	label<Numero>	Empieza con el prefijo label
TextBox	txt<Nombre TextBox>	Empieza con el prefijo txt
ComboBox	combo<Nombre combo box>	Inicia con el prefijo combo
Panel	panel<Numero>	Prefijo panel y numero

Realizado por: HUEBLA, Byron, 2018

#### 2.4.4. *Diseño de la base de datos*

Para el diseño de la Base de Datos del Sistema Académico se realizó un análisis previo de un enfoque total de funcionalidades del Sistema y de acuerdo a esas consideraciones y funcionalidades se realizó un diseño óptimo de la Base el cual nos garantice que los datos serán manejados correctamente evitando la redundancia de los mismos.

Se empleó el modelo relacional (**Figura 1-2**) mostrando las principales entidades que tiene la Base de Datos para el sistema “Diller”.



**Figura 9-2:** Diagrama entidad relación

Realizado por: Byron Huebla

En la figura anterior podemos visualizar el entorno principal de la funcionalidad del sistema, que indica que un profesor dicta muchas materias, y a su vez una materia puede ser dictada por diversos profesores donde cada una de estas entidades son asignados a un curso donde se encuentra matriculados varios estudiantes y por cada una de las matrículas de los estudiantes tienen varias notas. De tal manera que para la implementación en el motor de base de datos MySQL se procedió a realizar la normalización de la base de datos obteniendo así el diagrama lógico de la base de datos en la segunda forma normal dando como resultado un diagrama mucho más extenso debido a las relaciones entre entidades y al incremento de tipos y otras características propias de cada entidad.

Entonces se obtuvo un total de 38 tablas normalizadas en su tercera forma normal, las demás tablas que comprenden el Sistema se pueden visualizar en el Anexo A. (Modelo Físico)

#### 2.4.5. Diccionario de datos

El diccionario de datos es una representación de las características lógicas que poseen los tipos de datos, nombres de las entidades (tablas) que se van a utilizar en la implementación del sistema a nivel de Base de Datos, estos incluyen nombre de la tabla, campos, tipos de datos, llaves primarias, llaves foráneas. A continuación, se muestra en la **Tabla 8-2** el modelo de diccionario de datos de la entidad Estudiante.

**Tabla 9-2:** Diccionario de datos

CAMPO	TIPO	NULO	LLAVE PRINCIPAL	LLAVE FORANEA	DESCRIPCIÓN
id_estudiante	int(11)	No	Sí	No	auto_increment
codigo_estudiante	longtext	Sí	No	No	N/A
nombre	longtext	Sí	No	No	N/A
cumpleaños	longtext	Sí	No	No	N/A
sexo	longtext	Sí	No	No	N/A
religión	longtext	Sí	No	No	N/A
dirección	longtext	Sí	No	No	N/A
teléfono	longtext	Sí	No	No	N/A
email	longtext	Sí	No	No	N/A
password	longtext	Sí	No	No	N/A
id_representante	int(11)	Sí	No	Sí	N/A
id_autenticacion	longtext	Sí	No	No	N/A

Realizado por: HUEBLA, Byron, 2018

Luego de haber realizado el despliegue de todas las tablas del diccionario de datos de las entidades del sistema se deduce que los principales tipos de datos utilizados en el desarrollo del sistema DILLER, son el tipo de datos int, longtext, text, y varchar.

#### 2.4.6. Estándar de interfaz del sistema

Para la definición del estándar de Interfaces se mantuvo una reunión con el Product Owner, en el que supo manifestar que los mismos deben estar regidos a los lineamientos de aceptación, garantizando con ello la usabilidad, navegación, contenidos, así como la elección del color de la pantalla principal y el tipo de letra. A continuación, en la siguiente **Tabla 9-2** se detalla los componentes y criterios con los que se ha diseñado las interfaces.



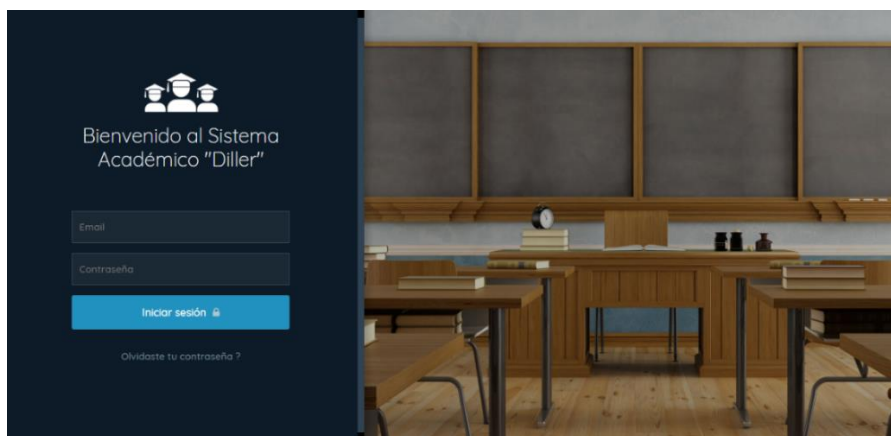
**Tabla 10-2:** Estándar de interfaces del sistema

COMPONENTE	UBICACIÓN	COLOR FONDO/FUENTE
Menú	Body, izquierdo	Gris / Blanco
Calendario	Body, centrado	Blanco / Negro
Búsqueda Rápida	Body, superior izquierdo	Blanco / Negro
Logo	Body, superior izquierdo	Blanco / Negro
Ajustes	Body, inferior izquierdo	Blanco / Negro
Estudiantes registrados	Body, superior derecho	Blanco / Rojo
Profesores registrados	Body, superior derecho	Blanco / Verde
Padres de familia registrados	Body, superior derecho	Blanco / Celeste
Asistencias	Body, superior derecho	Blanco / Azul

**Realizado por:** HUEBLA, Byron, 2018

Una vez implementado se realizó pruebas de aceptación con el usuario tomando en cuenta el aspecto de la visibilidad y el resultado fue que no existen inconvenientes con el tono de colores, además se tomó en cuenta que este color no provoca cansancio al momento de utilizarlas por tiempos prolongados.

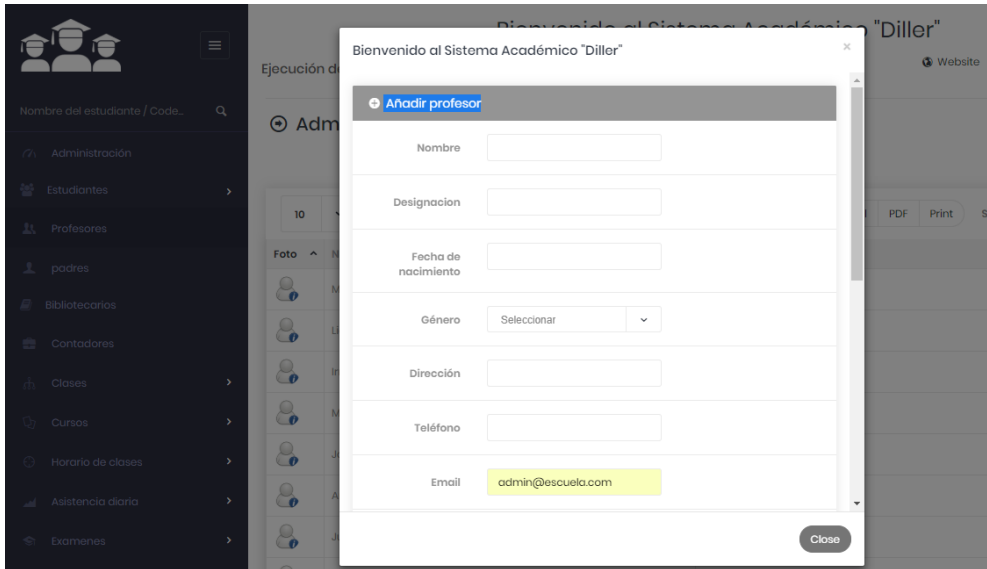
En la **Figura 2-2** podemos observar la interfaz de usuario con las especificaciones antes mencionadas.



**Figura 10-2:** Interfaz del sistema - Pantalla login

**Realizado por:** Byron Huebla

En la creación de los diseños de los formularios resaltan los colores blanco y negro, siendo el blanco quien destaca en casi todos los fondos del Sistema, a continuación, en la **Figura 3-2** mostraré la página de registro de un profesor en el Sistema por parte del Administrador, en el cual observaremos la combinación de colores antes mencionada.



**Figura 11-2:** Interfaz gráfica del sistema “Diller”

**Realizado por:** Byron Huebla

#### 2.4.7. Desarrollo de las Historias de Usuario

Las historias de usuario representan los requisitos provenientes del Sprint Backlog, estas historias cuentan con un identificador, nombre, descripción y responsable; así como sus tareas de ingeniería y pruebas de aceptación, las mismas que verifican el correcto cumplimiento de las historias de usuarios. A continuación, en la **Tabla 10-2** se describe el modelo de historias de Usuario, la misma que servirá de modelo para entender cómo fueron desarrolladas las demás Historias de Usuarios.

Historia de Usuario 01 – Registro de estudiantes

**Tabla 11-2:** Historia de usuario 01 / Registro estudiantes

HISTORIA DE USUARIO	
<b>ID:</b> HU-01	<b>Nombre de la Historia:</b> Registro de Estudiantes
<b>Usuario:</b> Desarrollador	<b>Sprint:</b> 02
<b>Fecha Inicio:</b> 30/01/2018	<b>Fecha Fin:</b> 03/02/2018
<b>Descripción:</b> Como desarrollador necesito crear un proceso que permita el registro de estudiantes	
<b>Pruebas de Aceptación:</b>	
<ul style="list-style-type: none"> <li>Validación del registro del usuario</li> </ul>	

**Realizado por:** HUEBLA, Byron, 2018

Luego de que se detalle todas las historias de usuarios, se concluye que se realizaron un total de 41 historias de Usuarios y 8 historias Técnicas las mismas que se dividen en tareas de ingeniería, a continuación, en la **Tabla 11-2** se detalla un modelo de las tareas de ingeniería realizadas en la historia de usuario detallada anteriormente, donde se describe la fecha de la tarea y el tiempo asignado a la tarea correspondiente.

**Tabla 12-2:** Tareas de ingeniería - Historia usuario 01

FECHA	TAREAS DE INGENIERIA	TIEMPO
30/01/2018	<b>Tarea 1: Definir procedimientos de base de datos para el registro de estudiantes.</b>	<b>1 h</b>
	Creación de conexión a la base de datos.	1 h
30/01/2018	<b>Tarea 2: Creación de Clases y Paquetes</b>	2 h
	Creación de paquetes organizativos	1 h
	Creación de atributos y métodos para la clase estudiante	1 h
30/01/2018	<b>Tarea 3: Creación de métodos del controlador para poder realizar el registro de estudiantes</b>	<b>4 h</b>
	Creación de controlador PHP	4 h
30/01/2018	<b>Tarea 4: Crear la Vista de la Interfaz de usuario para finalizar el registro de estudiantes.</b>	<b>8 h</b>
	Adición de métodos de acceso al controlador para el registro de los mismos	2 h
	Preparación de método controlador para recepción y carga preliminar de datos	2 h
30/01/2018	Procesar envío de datos para el registro al servidor web, aplicación de validaciones de registro.	2 h
	Creación de funciones javascript	2 h

**Realizado por:** HUEBLA, Byron, 2018

Luego de haber listado todas las tareas de ingeniería se procede a realizar una ficha por cada tarea de ingeniería para dar cumplimiento con la historia de Usuario en ejecución a continuación, en la **Tabla 12-2**, se detalla el modelo de ficha de una tarea de ingeniería en el cual se detalla el número de sprint al cual pertenece, número y nombre de la tarea, fecha de inicio y fecha fin, nombre de la historia de usuario y el listado de pruebas de aceptación para dar por terminada la tarea en mención.

**Tabla 13-2:** Tarea de ingeniería 01 - Historia de usuario 01

<b>TAREA DE INGENIERÍA</b>	
<b>Sprint:</b> 02	<b>Número de Tarea:</b> 01
<b>Nombre de la Historia:</b> Registro de Estudiantes.	
<b>Nombre de la Tarea:</b> Definir procedimientos de base de datos para el registro de estudiantes	
<b>Programador Responsable:</b> Desarrollador	<b>Tipo de Tarea:</b> Desarrollo
<b>Fecha Inicio:</b> 30/01/2018	<b>Fecha Fin:</b> 30/01/2018
<b>Descripción:</b> Investigación y ejecución para realizar la conexión entre la aplicación y la Base de Datos.	
<b>Pruebas de Aceptación:</b>	
<ul style="list-style-type: none"> <li>• Test de Conexión</li> </ul>	

Realizado por: HUEBLA, Byron, 2018

Luego de realizar las descripciones de todas las tareas de ingeniería de acuerdo a las historias de usuario a las que pertenece se concluye que se han realizado un total de 70 tareas de ingeniería las mismas que para su finalización se debe desarrollarse una ficha de pruebas de aceptación, a continuación en la **Tabla 13-2** se muestra un modelo de pruebas de aceptación el mismo que tiene entre sus campos: número y nombre de prueba, nombre de Historia de Usuario, condición de ejecución, pasos, resultados y evaluación de la prueba.

**Tabla 14-2:** Prueba de aceptación 01 - Historia de usuario 01

<b>PRUEBA DE ACEPTACIÓN</b>	
<b>Prueba N°:</b> 2.1	<b>Nombre de la Historia:</b> Registro de Estudiantes.
<b>Nombre de la Prueba:</b> Verificación de carga de datos mediante la interfaz.	
<b>Responsable:</b> Desarrollador	<b>Fecha:</b> 20/01/2018
<b>Descripción:</b> Crear la vista de la interfaz de usuario para poder validar el registro de estudiantes a la BD.	
<b>Condiciones de Ejecución:</b>	
<ul style="list-style-type: none"> <li>• Revisar la clase de la carga y la búsqueda de datos del paquete en el modelo de estudiantes.</li> <li>• Instanciación correcta del objeto para la carga de datos.</li> <li>• Especificación correcta de las columnas de la tabla en el componente en la interfaz de Estudiante.</li> </ul>	
<b>Pasos de Ejecución:</b>	
<ul style="list-style-type: none"> <li>• Instanciar objeto de carga de datos desde el controlador.</li> <li>• Llamar al método de carga de datos.</li> <li>• Envío de datos a la interfaz de estudiante.</li> </ul>	
<b>Resultado:</b>	
<ul style="list-style-type: none"> <li>• Carga correcta de datos en la tala del Estudiante.</li> </ul>	
<b>Evaluación de la Prueba:</b>	
<ul style="list-style-type: none"> <li>• Ok</li> </ul>	

Realizado por: HUEBLA, Byron, 2018

Luego de culminar con todas las pruebas de aceptación se llegó a la concluir que todas obtuvieron una evaluación de satisfacción exitoso, esto nos lleva a concluir que todos los requerimientos del software fueron realizado y probados correctamente.

## 2.5. Fase de Finalización

En esta fase de la metodología SCRUM, se anotaron todas las actividades realizadas previo a la finalización del desarrollo del Sistema “DILLER”, para el cual se estableció el Sprint llamado BurnDown Chart, que consiste en un gráfico de trabajo pendiente a lo largo del tiempo donde se muestra la velocidad a la que está completando los objetivos o requisitos planificados al inicio del desarrollo del Sistema.

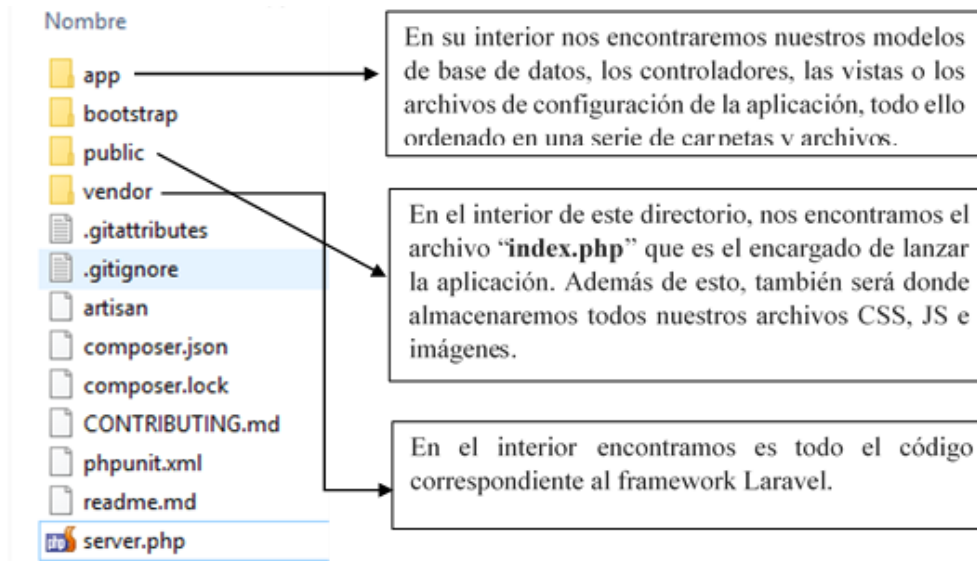
Además, este gráfico nos permitirá conocer si el desarrollador completó el trabajo en el tiempo estimado. A continuación, se describe las actividades realizadas en la fase de Finalización del Proyecto. (Tabla 14-2).

**Tabla 15-2:** Actividades de finalización de proyecto

ACTIVIDAD	DESCRIPCIÓN	REPONSABLE
Evaluación del Sistema	<ul style="list-style-type: none"> <li>• Pruebas de Calidad al Sistema</li> <li>• Encuesta de Satisfacción</li> </ul>	Desarrollador
Documentación del Sistema	<ul style="list-style-type: none"> <li>• Generación del Manual de Usuario</li> </ul>	Desarrollador
Capacitación	<ul style="list-style-type: none"> <li>• Entrega del Manual de Usuario al Product Owner.</li> <li>• Capacitación a usuarios acorde al Manual de Usuario</li> </ul>	Desarrollador

Realizado por: HUEBLA, Byron, 2018

Una vez finalizado el desarrollo del Sistema DILLER se obtuvo la siguiente estructura (Figura 4-2), de todos los archivos y carpetas que podemos apreciar los más importantes son las que se encuentran descritas.



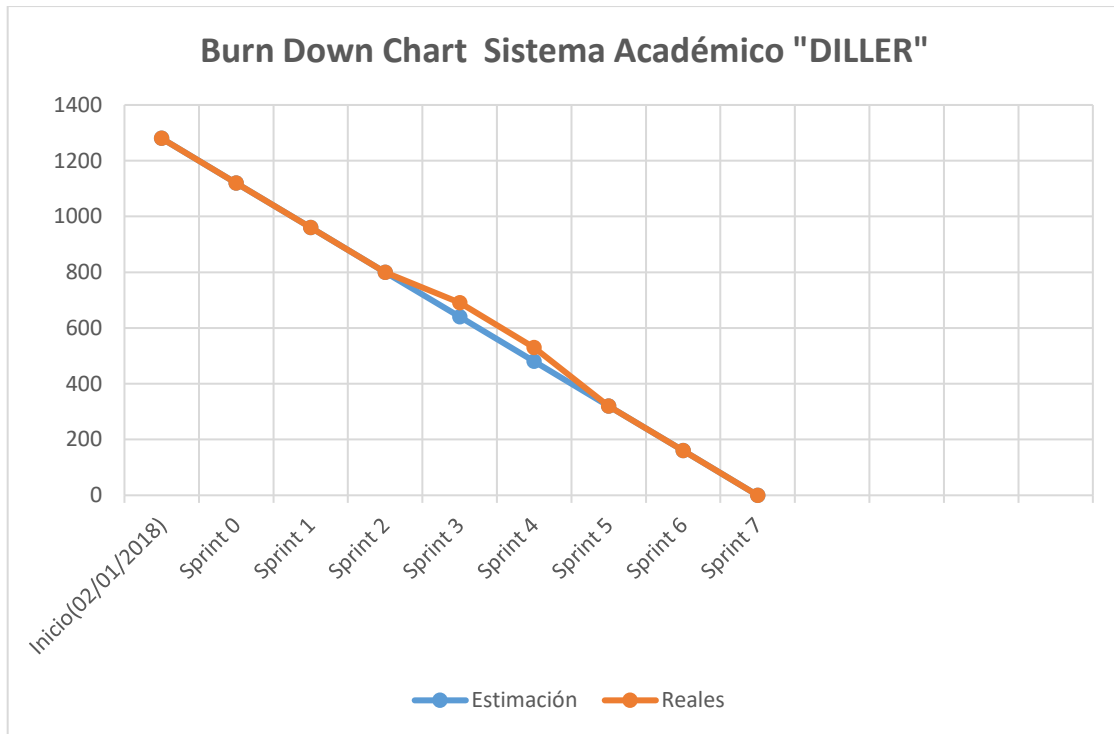
**Figura 12-2:** Estructura de archivos sistema Diller

Realizado por: HUEBLA, Byron, 2018

### 2.5.1. Gestión del Proyecto

Es una parte esencial en el Desarrollo del Sistema cuando se aplica SCRUM, además de ser una manera de mostrar al equipo que pasó durante el desarrollo de cada Sprint. La gestión del proyecto se lo realizó a través de BurnDown Chart el cual es parte de la metodología que se eligió para el desarrollo, el mismo que hace referencia al seguimiento durante el desarrollo del Sistema, a continuación, en la **Figura 5-2** se podemos apreciar la ilustración en donde las fechas del sprint se representan en el eje X en el que se aprecia la fecha de inicio y fecha fin que corresponden a los días comprendidos entre el 02/01/2018 y 30/06/2018.

El esfuerzo está representado en días en el eje Y con un total de 1280 puntos. El gráfico está representado por dos líneas, el de color naranja muestra el desarrollo real del proyecto mientras que la línea de color azul representa el desarrollo ideal.



**Figura 13-2:** BurnDown Chart

**Realizado por:** HUEBLA, Byron, 2018

En el BurnDown Chart del Sistema se puede apreciar cómo va evolucionando el cumplimiento los sprints en base a la planificación que se realizó al inicio del desarrollo del Sistema. El desarrollo de la aplicación se avanzó de una manera correcta justificando las tareas de cada sprint. De todos los sprints (0, 1, 2, 3, 4, 5, 6, 7) se puede visualizar que tan sólo los sprints 3 y 4 no cumplieron con la velocidad de implementación estimada dando como consecuencia no poder entregar versiones completamente funcionales, mientras tanto los sprints 0, 1, 2, 5, 6 y 7 cumplieron con el tiempo estimado lo que permitió entregar versiones completamente funcionales.

## CAPÍTULO III

### 3. MARCO DE EVALUACIÓN, DISCUSIÓN Y ANÁLISIS DE RESULTADOS

En este capítulo se procederá a realizar el análisis de resultados obtenidos luego de culminar con el desarrollo de la aplicación en su evaluación se tomó énfasis en lo referente a tecnologías, herramientas y metodología que ayudaron en la elaboración del “**DESARROLLO DE UN SISTEMA ACADÉMICO ORIENTADO A LA WEB PARA LA UNIDAD EDUCATIVA EXPERIMENTAL INTERANDINA UTILIZANDO EL FRAMEWORK LARAVEL Y MYSQL**”.

De acuerdo a lo señalado se realizó la evaluación de la funcionalidad, usabilidad y eficiencia haciendo uso del estándar ISO/IEC 9126/3 previamente explicado en el primer capítulo.

#### 3.1. Cálculo de la muestra poblacional para la evaluación del sistema académico

Para la evaluación de calidad del sistema académico se utilizó la técnica de encuesta (**Ver Anexo C**) con preguntas cerradas para la recopilación de información acerca de la usabilidad funcionalidad y eficiencia del software. además, para el tamaño de la muestra poblacional se hizo uso del muestreo aleatorio simple puesto que el sistema académico está dirigido a tres tipos de usuarios: administrador o en mi caso es el rector, profesores, estudiantes del décimo año de educación básica, siendo este el curso proporcionado por el rector para llevar a cabo la encuesta. En la **Figura 1-3** se ejemplifica la selección la muestra poblacional dando como resulta un total de 19 usuarios encuestados.



**Figura 1-3:** Muestreo Aleatorio Simple

Realizado por: HUEBLA, Byron, 2018



### 3.2. Requerimientos de calidad

Es un conjunto de sub características propias de la funcionalidad, usabilidad y eficiencia para la evaluación de la calidad del Sistema Académico “DILLER”, según el estándar ISO/IEC 9126-3, se debe establecer métricas y parámetros los cuales permita identificar el cumplimiento de la calidad en el Sistema.

En la siguiente **Tabla 1-3** procederé a describir los parámetros bajo los cuales de evaluó la funcionalidad, usabilidad y eficiencia del Sistema Académico “DILLER”

**Tabla 1-3:** Parámetros de medición

NIVEL DE ACEPTACIÓN	RANGO DE ACEPTACIÓN
Alta	0.71 – 1.00
Media	0.36 – 0.70
Baja	0.00 – 0.35

Realizado por: HUEBLA, Byron, 2018

A continuación, en la **Tabla 2-3**, describiré los niveles bajo los cuales fue evaluado la funcionalidad, usabilidad y eficiencia del Sistema Académico Diller.

**Tabla 2-3:** Sub características de calidad

CARACTERÍSTICAS	SUB CARACTERÍSTICAS	ACEPTACIÓN REQUERIDA
Funcionalidad	Adecuidad	Alta
	Exactitud	Alta
	Interoperabilidad	Media
	Seguridad	Alta
	Conformidad de la Funcionalidad	Media
Usabilidad	Entendibilidad	Alta
	Aprendibilidad	Alta
	Operabilidad	Media
	Atractivo	Baja
	Conformidad de la usabilidad	Media
Eficiencia	Tiempo de respuesta	Alta
	Utilización de recursos	Media
	Conformidad de eficiencia	Media

Realizado por: HUEBLA, Byron, 2018

### 3.3. Análisis de la funcionalidad del sistema

El análisis de la funcionalidad del sistema académico, se ha establecido mediante las métricas de sus características de calidad (adecuada, exactitud, interoperabilidad, seguridad, conformidad de la funcionalidad), establecidas por la norma ISO 9126-3 aplicada a una encuesta (**Anexo C**) realizada a usuarios del sistema, y desarrollador del sistema, se detalla a continuación:

#### 3.3.1. Métricas de evaluación: Adecuidad

A continuación, se detalla las notaciones de variables utilizadas para el análisis de la métrica adecuación **Tabla 3-3**.

**A:** Número de requerimientos faltantes.

**B:** Número de requerimientos planteados. (Total de Historias de Usuarios e Historias Técnicas)

**X:** Resultante de la fórmula de medición.

**Tabla 3-3:** Métrica de adecuación

MÉTRICA DE ADECUIDAD	
Nombre:	Finalización e implementación funcional del Sistema
Propósito:	¿Está completa la implementación funcional del sistema académico DILLER?
Requerimientos Faltantes	No existen requerimientos faltantes.
Medición:	A = 0 B = 41 $X = 1 - (A/B)$ $X = 1 - (0/41)$ X = 1
Interpretación:	$0 \leq X \leq 1$ $0 \leq 1 \leq 1$ Entre más cercano a 1, más completa
Tipo de escala:	Absoluta
Fuente de medición:	Especificación de requisitos, diseño y código fuente

Realizado por: HUEBLA, Byron, 2018

Inmediatamente realizado la métrica de adecuación, se logró un valor numérico de medición “1” con una jerarquía de aceptación “alta”, dentro de la categoría establecida en la norma ISO 9126- 3 que rige para la funcionalidad del sistema.

### 3.3.2. Métricas de evaluación: Exactitud

En este punto de la evaluación, se indica la notación de las variables que han participado en el análisis de la exactitud del producto software, el detalle se describe en la **Tabla 4-3**.

**A:** Números de requerimientos no exactos

**B:** Número de requerimientos planteados. (Total de Historias de Usuarios e Historias Técnicas)

**X:** Resultante de la fórmula de medición

**Tabla 4-3:** Métrica de exactitud

<b>MÉTRICA DE EXACTITUD</b>	
Nombre:	Exactitud del Sistema
Propósito:	¿El sistema académico, realiza los procesos esperados de manera correcta?
Requerimientos no exactos:	Tiempo que tarda la impresión en pantalla de reportes.
Medición:	A = 1 B = 41 $X = 1 - (A/B)$ $X = 1 - (1/41)$ X = 0,97
Interpretación:	$0 \leq X \leq 1$ $0 \leq 97 \leq 1$ Entre más cercano a 1, más completa
Tipo de escala:	Informe de revisión
Fuente de medición:	Informe de revisión

Realizado por: HUEBLA, Byron, 2018

Inmediatamente realizado la métrica de exactitud, se logró un valor numérico de medición “0,97” con una jerarquía de aceptación “alta”, dentro de la categoría establecida en la norma ISO 9126-3 que rige para la funcionalidad del sistema.

### 3.3.3. Métricas de evaluación: Interoperabilidad

En este punto de la evaluación, se indica la notación de las variables que han participado en el análisis de la interoperabilidad del producto software, el detalle se describe en la **Tabla 5-3**.

**A:** Número de sistemas de interacción

**X:** Resultante de la fórmula de medición

**Tabla 5-3:** Métrica de interoperabilidad

MÉTRICA DE INTEROPERABILIDAD	
Nombre:	Interoperabilidad entre Sistemas
Propósito:	¿Interactúa el Sistema Académico DILLER con otros tipos de sistemas?
Sistemas de Interacción:	No existen Sistemas de Interacción
Medición:	A = 0 X = 1 / (A+1) X = 1 / (0+1) X = 1
Interpretación:	0 <= X <= 1 0 <= 1 <= 1
Tipo de escala:	Absoluta
Fuente de medición:	Informe de revisión

Realizado por: HUEBLA, Byron, 2018

Inmediatamente realizado la métrica de interoperabilidad, se logró un valor numérico de medición “1” una jerarquía de aceptación “alta”, dentro de la categoría establecida en la norma ISO 9126-3 que rige para la funcionalidad del sistema.

### 3.3.4. Métricas de evaluación: Seguridad

En este punto de la evaluación, se indica la notación de las variables que han participado en el análisis de la seguridad del producto software, el detalle se describe en la **Tabla 6-3**.

**A:** Número de accesos no controlados

**X:** Resultante de la fórmula de medición

**Tabla 6-3:** Métrica de seguridad

<b>MÉTRICA DE SEGURIDAD</b>	
Nombre:	Seguridad del Sistema Académico
Propósito:	¿El sistema académico restringe el acceso a usuarios no autorizados?
Accesos no Controlados:	No existen usuarios que no se haya registrado que tengan acceso al sistema.
Medición:	A = 0 $X = 1 / (A+1)$ $X = 1 / (0+1)$ X = 1 Entre más cercano a 1, más completa
Interpretación:	$0 \leq X \leq 1$ $0 \leq 1 \leq 1$
Tipo de escala:	Absoluta
Fuente de medición:	Informe de revisión

**Realizado por:** HUEBLA, Byron, 2018

Inmediatamente realizado la métrica de seguridad, se logró un valor numérico de medición “1” con una jerarquía de aceptación “alta”, dentro de la categoría establecida en la norma ISO 9126- 3 que rige para el funcionamiento del sistema.

### 3.3.5. Métricas de evaluación: Conformidad de la funcionalidad

En este punto de la evaluación, se indica la notación de las variables que han participado en el análisis de la conformidad de la funcionalidad del producto software, el detalle se describe en la **Tabla 7-3**.

**A:** Número de normas o estándares de creación

**X:** Resultante de la fórmula de medición

**Tabla 7-3:** Métrica de conformidad de la funcionalidad

<b>MÉTRICA DE CONFORMIDAD DE LA FUNCIONALIDAD</b>	
Nombre:	Conformidad del Sistema
Propósito:	¿Considera usted que el Sistema Académico DILLER, fue desarrollado en base a normas y/o estándares?
Normas o Estándares utilizados en el desarrollo:	Patrón Modelo Vista Controlador Estándar de Codificación Estándar de normalización para Base de Datos relacionales.

Medición:	A = 3 X = (1/A) - 1 X = (1 /3) - 1 X = 0,66
Interpretación:	0<= X <=1 0<=0,66 <= 1 Entre más cercano a 1, más completo
Tipo de escala:	Absoluta
Fuente de medición:	Especificaciones de requerimientos Informe de revisión Código fuente

Realizado por: HUEBLA, Byron, 2018

Inmediatamente realizado la métrica de conformidad de la funcionalidad, se logró un valor numérico de medición “0,66” con una jerarquía de aceptación “media”, dentro de la categoría establecida en la norma ISO 9126-3 que rige para la funcionalidad del sistema.

### 3.4. Análisis de la usabilidad del sistema

El análisis de la usabilidad del sistema académico, se ha establecido mediante las métricas de sus características de calidad (entendimiento, aprendizaje, operabilidad, atracción, conformidad de la usabilidad), establecidas por la norma ISO 9126-3 aplicada a una encuesta realizada a usuarios del sistema. **Anexo C**

#### 3.4.1. Métricas de evaluación: Entendibilidad

En este punto de la evaluación, se indica la notación de las variables que han participado en el análisis del entendimiento del producto software, el detalle se describe en la **Tabla 8-3**.

**A:** Número de funciones (o tipo de funciones) evidentes al usuario.

**B:** Es el total de funciones (o tipo de funciones).

**X:** Resultante de la fórmula de medición

**Tabla 8-3: Métrica de entendibilidad**

<b>MÉTRICA DE ENTENDIBILIDAD</b>	
Nombre:	Herramientas evidentes del sistema
Propósito:	¿Qué herramientas o funciones del Sistema Académico DILLER, son evidentes para el usuario?
Funciones o herramientas evidentes:	Función de acceso Herramienta de interacción para la impresión de reportes. Herramienta de subida de archivos para el registro masivo de estudiantes Funciones del CRUD para estudiantes, profesores, padres de familia, bibliotecario y administrador. Funciones del CRUD para las notas. Funciones del CRUD para las matrículas. Funciones del CRUD para el calendario académico. Funciones del CRUD para la retroalimentación de conocimientos. Funciones del CRUD para la creación de exámenes.
Medición:	A = 15 B = 15 A = B $X = (A/B)$ $X = (9 /9)$ X = 1
Interpretación:	$0 \leq X \leq 1$ $0 \leq 1 \leq 1$ Entre más cercano a 1, más completa
Tipo de escala:	Absoluta
Fuente de medición:	Diseño

Realizado por: HUEBLA, Byron, 2018

Inmediatamente realizado la métrica de entendimiento, se logró un valor numérico de medición “1” con una jerarquía de aceptación “alta”, dentro de la categoría establecida en la norma ISO 9126-3 que rige para la usabilidad del sistema.

### 3.4.2. Métricas de evaluación: Aprendibilidad

En este punto de la evaluación, se indica la notación de las variables que han participado en el análisis del aprendizaje del producto software, el detalle se describe en l **Tabla 9-3**.

**A:** Número de usuarios que no aprendieron a utilizar el sistema

**B:** Número de usuarios encuestados

**X:** Resultante de la fórmula de medición

**Tabla 9-3:** Métrica de aprendibilidad

MÉTRICA DE APRENDIBILIDAD	
Nombre:	Exactitud del Sistema Académico
Propósito:	¿Considera usted que el Sistema Académico DILLER es fácil de aprender?
Medición:	A = 13 B = 1 $X = 1 - (B/A)$ $X = 1 - (1/13)$ X = 0,92
Interpretación:	$0 \leq X \leq 1$ $0 \leq 0,92 \leq 1$ Entre más cercano a 1, es más exacto la interpretación.
Tipo de escala:	Absoluta
Fuente de medición:	Informe de revisión

**Realizado por:** HUEBLA, Byron, 2018

Inmediatamente realizado la métrica de aprendizaje, se logró un valor numérico de medición “0,92” con una jerarquía de aceptación “alta”, dentro de la categoría establecida en la norma ISO 9126-3 que rige para la usabilidad del sistema.

### 3.4.3. Métricas de evaluación: Operabilidad

En este punto de la evaluación, se indica la notación de las variables que han participado en el análisis de la operabilidad del producto software, el detalle se describe en la **Tabla 10-3**.

**A:** Número de usuarios que no pudieron operar el sistema

**B:** Número de usuarios encuestados

**X:** Resultante de la fórmula de medición



**Tabla 10-3:** Métrica de operabilidad

MÉTRICA DE OPERABILIDAD	
Nombre:	Inspección del Sistema Académico
Propósito:	¿Considera usted que el Sistema Académico DILLER es fácil de operar?
Medición:	A = 1 B = 13 $X = 1 - (1/A)$ $X = 1 - (1/13)$ X = 0,92
Interpretación:	$0 \leq X \leq 1$ $0 \leq 0,92 \leq 1$ Entre más cercano a 1, más completa
Tipo de escala:	Absoluta
Fuente de medición:	Verificación Informe de revisión

Realizado por: HUEBLA, Byron, 2018

Inmediatamente realizado la métrica de operabilidad, se logró un valor numérico de medición “0,92” con una jerarquía de aceptación “media”, dentro de la categoría establecida en la norma ISO 9126-3 que rige para la usabilidad del sistema.

#### 3.4.4. Métricas de evaluación: Atractivo

En este punto de la evaluación, se indica la notación de las variables que han participado en el análisis de la atracción del producto software, el detalle se describe en la **Tabla 11-3**.

**A:** Número de usuarios que no le agradaron el diseño del Sistema

**B:** Número de usuarios encuestados

**X:** Resultante de la fórmula de medición

**Tabla 11-3:** Métrica de atracción

MÉTRICA DE ATRACTIVO	
Nombre:	Diseño de la Interfaz del Sistema
Propósito:	¿El sistema académico tiene una agradable apariencia visual en su diseño?
Medición:	A = 2 B = 13 $X = 1 - (A/B)$ $X = 1 - (2/12)$ X = 0,76
Interpretación:	$0 \leq X \leq 1$ $0 \leq 0,83 \leq 1$ Entre más cercano a 1, más completo.

Continúa

*Continuación*

Tipo de escala:	Absoluta
Fuente de medición:	Informe de revisión

Realizado por: HUEBLA, Byron, 2018

Inmediatamente realizado la métrica de atracción, se logró un valor numérico de medición “0,76” con una jerarquía de aceptación “media”, dentro de la categoría establecida en la norma ISO 9126-3 que rige para la usabilidad del sistema.

**3.4.5. Métricas de evaluación: Conformidad de la usabilidad**

En este punto de la evaluación, se indica la notación de las variables que han participado en el análisis de la conformidad de la usabilidad del producto software, el detalle se describe en la **Tabla 12-3**.

**A:** Número de normas o estándares de diseño

**X:** Resultante de la fórmula de medición

**Tabla 12-3:** Métrica de conformidad de la usabilidad

MÉTRICA DE CONFORMIDAD DE USABILIDAD	
Nombre:	Aprobación de la Usabilidad del Sistema
Propósito:	¿Considera usted que el Sistema Académico DILLER fue diseñado visualmente en base a normas y/o estándares?
Normas o estándares utilizados:	Diseño Moderno Responsivo.
Medición:	$X = (1/A) - 1$ $X = (1/1) - 1$ $X = 1$
Interpretación:	$0 \leq X \leq 1$ $0 < 1 \leq 1$ Entre más cercano a 1, más completo
Tipo de escala:	Absoluta
Fuente de medición:	Informe de revisión

Realizado por: HUEBLA, Byron, 2018

Inmediatamente realizado la métrica de conformidad de la usabilidad, se logró un valor numérico de medición “1” con una jerarquía de aceptación “alta”, dentro de la categoría establecida en la norma ISO 9126-3 que rige para la usabilidad del sistema

### 3.5. Análisis de la eficacia del sistema

El análisis de la eficiencia del sistema académico, se ha establecido mediante el uso de métricas de sus características de calidad (comportamiento de respuesta, utilización de recursos, conformidad de la eficiencia), establecidas por la norma ISO 9126-3 aplicada a los tiempos tomados mediante un cronómetro y con la técnica de observación directa

#### 3.5.1. Métricas de evaluación: Tiempo de respuesta

A continuación, se detalla las notaciones de variables utilizadas para el análisis de la métrica tiempo de respuesta **Tabla 13-3**.

**A:** Tiempo (calculado o simulado) que se demora con un Sistema de escritorio

**B:** Tiempo (calculado o simulado) que se demora con un Sistema Web

**X:** Resultante de la fórmula de medición

**Tabla 13-3:** Métrica de tiempo de respuesta

MÉTRICA DE TIEMPO DE RESPUESTA	
Nombre:	Comportamiento o tiempo de respuesta del sistema académico
Propósito:	¿Cuál es el tiempo estimado que consume una acción?
Método de Aplicación:	Evaluar la eficiencia de las llamadas al SO y a la aplicación. Estimar el tiempo de respuesta basado en ello. Puede medirse: todo o partes de las especificaciones de diseño. Probar la ruta completa de una transacción. Probar módulos o partes completas del producto. Producto completo durante la fase de pruebas
Medición:	A = 1 hora B = 0, 10 minutos $X = 1 - (B/A)$ $X = 1 - (0,10/1)$ X = 0,90
Interpretación:	$0 \leq X \leq 1$ $0 \leq 0,90 \leq 1$ Entre más cercano a 1, es más exacto la interpretación.
Tipo de escala:	Proporción
Tipo de medida:	X= tiempo
Fuente de medición:	Sistema Operativo

Realizado por: HUEBLA, Byron, 2018

Inmediatamente realizado la métrica de respuesta, se logró un valor numérico de medición “0.90” con una jerarquía de aceptación “alta”, dentro de la categoría establecida en la norma ISO 9126-3 que rige para la eficiencia del sistema.

### 3.5.2. Métricas de evaluación: Utilización de recursos

En este punto de la evaluación, se indica la notación de las variables que han participado en el análisis de la utilización de recursos del producto software, el detalle se describe en la **Tabla 14-3**.

**A:** Número de hardware o software adicional utilizado con un sistema web.

**B:** Número de hardware o software adicional utilizado con un sistema de escritorio.

**X:** Resultante de la fórmula de medición

**Tabla 14-3:** Utilización de recursos

MÉTRICA DE UTILIZACIÓN DE RECURSOS	
Nombre:	Utilización de Recursos
Propósito:	Los recursos que incluyen en el desarrollo de software, la configuración de hardware y software del sistema, y materiales
Método de Aplicación:	Capacidad del sistema académico para usar recursos (cantidades y tipos) cuando éste funciona desde las condiciones establecidas. Pen drive CDs Hojas de Cálculo Excel
Medición:	A = 2 B = 3 $X = 1 - (A/B)$ $X = 1 - (2/3)$ X = 0,66
Interpretación:	$0 \leq X \leq 1$ $0 \leq 0,66 \leq 1$ Entre más cercano a 1, es más exacto la interpretación.
Tipo de escala:	Proporción

Realizado por: HUEBLA, Byron, 2018

Inmediatamente realizado la métrica de utilización de recursos, se logró un valor numérico de medición “0,66” con una jerarquía de aceptación “media”, dentro de la categoría establecida en la norma ISO 9126-3 que rige para la eficiencia del sistema.

### 3.5.3. Métricas de evaluación: Conformidad de la eficiencia

En este punto de la evaluación, se indica la notación de las variables que han participado en el análisis de la conformidad de la eficiencia del producto software, el detalle se describe en la **Tabla 15-3**.

**A:** Número de normas o estándares de diseño

**X:** Resultante de la fórmula de medición

**Tabla 15-3:** Métrica de conformidad de la eficiencia

MÉTRICA DE CONFORMIDAD DE LA EFICIENCIA	
Nombre:	Conformidad de la eficiencia del sistema académico
Propósito:	¿El sistema académico cumple los requisitos solicitados?
Método de Aplicación:	IEEE ISO/IEC
Medición:	$X = 1 - (1/A)$ $X = 1 - (1/2)$ $X = 0,5$
Interpretación:	$0 \leq X \leq 1$ $0 \leq 0,5 \leq 1$ Entre más cercano a 1, más completa
Tipo de escala:	Absoluta
Fuente de medición:	Informe de revisión.

**Realizado por:** HUEBLA, Byron, 2018

Inmediatamente realizado la métrica de conformidad de la usabilidad, se logró un valor numérico de medición “0,50” con una jerarquía de aceptación “media”, dentro de la categoría establecida en la norma ISO 9126-3 que rige para la eficiencia del sistema.

### 3.6. Evaluación de la calidad del sistema

En la siguiente **Tabla 16-3** se detalla todos los valores entre los niveles requeridos y obtenidos con sus respectivas métricas para realizar la evaluación de funcionalidad del Sistema Académico DILLER.

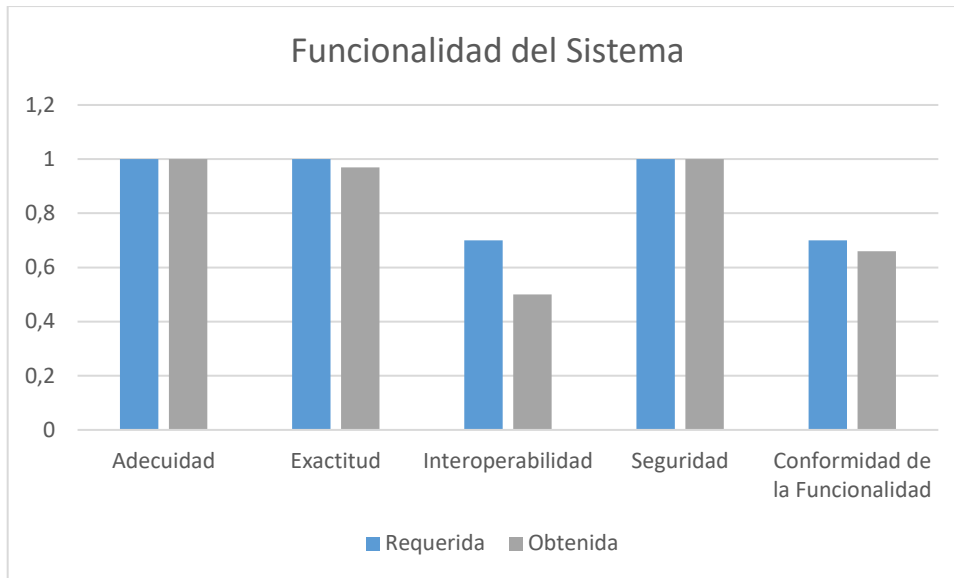
**Tabla 16-3:** Especificación de la evaluación de funcionalidad del sistema

SUB CARACTERÍSTICAS	REQUERIDO		OBTENIDO	
	MÉTRICA DE EVALUACIÓN	NIVEL ACEPTACIÓN	MÉTRICA DE EVALUACIÓN	NIVEL ACEPTACIÓN
Adecuidad	1,00	Alta	1,00	Alta
Exactitud	1,00	Alta	0,97	Alta
Interoperabilidad	0,70	Media	0,50	Media
Seguridad	1,00	Alta	1,00	Alta
Conformidad de la Funcionalidad	0,70	Media	0,66	Media
<b>PORCENTAJE</b>	100,00%	<b>PORCENTAJE</b>	93,86%	

**Realizado por:** HUEBLA, Byron, 2018

Mediante el análisis de la tabla anterior se puede decir que el Sistema Académico Diller es 93,86 % funcional, cálculo realizado de mediante la suma de los valores numéricos de sus métricas, aplicada a una regla de tres simple.

A continuación, se detalla en la **Figura 1-3**, un cuadro estadístico de barras con los valores requeridos en comparación con los valores obtenidos con la finalidad de tener un bosquejo más amplio de la funcionalidad del Sistema Académico Diller.



**Figura 2-3:** Evaluación de funcionalidad del sistema

Realizado por: HUEBLA, Byron, 2018

En la figura anterior muestra la evaluación de la eficiencia del sistema Académico el mismo que tiene dos columnas donde la barra de color azul es el resultado esperado y la barra de color gris es el resultado obtenido

En la siguiente **Tabla 17-3** se detalla todos los valores entre los niveles requeridos y obtenidos con sus respectivas métricas para realizar la evaluación de usabilidad del Sistema Académico Diller.

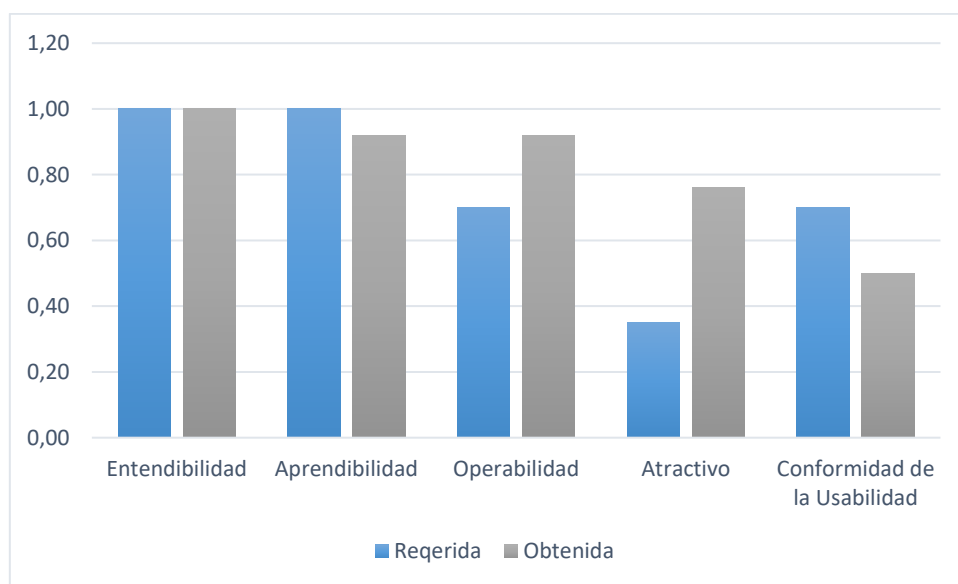
**Tabla 17-3:** Especificación de la evaluación de usabilidad del sistema

SUB CARACTERÍSTICAS	REQUERIDO		OBTENIDO	
	MÉTRICA DE EVALUACIÓN	NIVEL ACEPTACIÓN	MÉTRICA DE EVALUACIÓN	NIVEL ACEPTACIÓN
Entendibilidad	1,00	Alta	1,00	Alta
Aprendibilidad	1,00	Alta	0,92	Alta
Operabilidad	0,70	Media	0,92	Alta
Atractivo	0,35	Baja	0,76	Alta
Conformidad de la Usabilidad	0,70	Media	1	Alta
<b>PORCENTAJE</b>	100,00%	<b>PORCENTAJE</b>	98,93%	

Realizado por: HUEBLA, Byron, 2018

Mediante el análisis de la tabla anterior se puede decir que el Sistema Académico Diller es 98,93 % usable, cálculo realizado de mediante la suma de los valores numéricos de sus métricas, aplicada a una regla de tres simple.

A continuación, se detalla en la **Figura 2-3**, un cuadro estadístico de barras con los valores requeridos en comparación con los valores obtenidos con la finalidad de tener un bosquejo más amplio de la usabilidad del Sistema Académico Diller.



**Figura 3-3:** Evaluación de usabilidad del sistema

**Realizado por:** HUEBLA, Byron, 2018

En la figura anterior muestra la evaluación de la eficiencia del sistema Académico el mismo que tiene dos columnas donde la barra de color azul es el resultado esperado y la barra de color gris es el resultado obtenido

En la siguiente **Tabla 18-3** se detalla todos los valores entre los niveles requeridos y obtenidos con sus respectivas métricas para realizar la evaluación de eficiencia del Sistema Académico Diller.



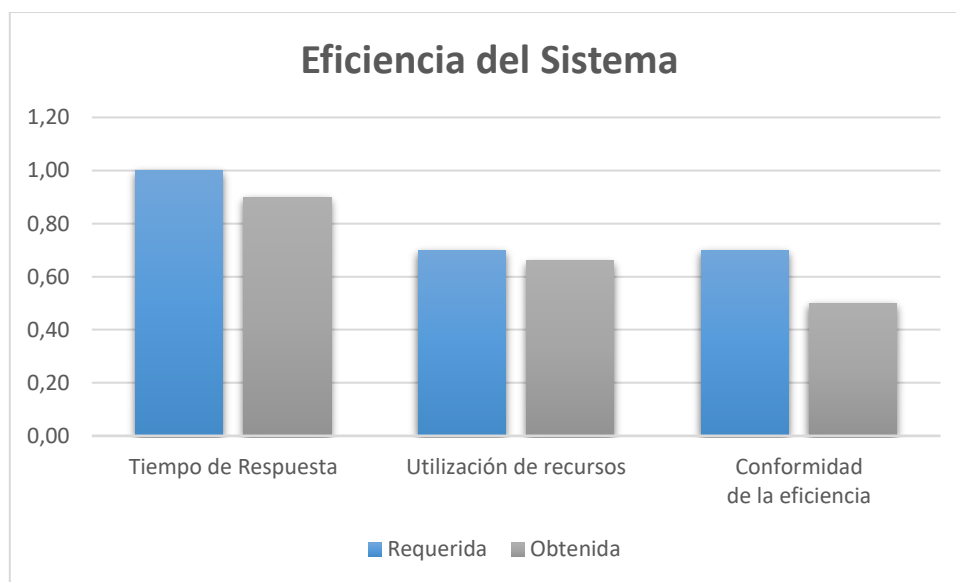
**Tabla 18-3:** Especificación de la evaluación de eficiencia del sistema

SUB CATEGORIAS	REQUERIDO		OBTENIDO	
	MÉTRICA DE EVALUACIÓN	NIVEL ACEPTACIÓN	MÉTRICA DE EVALUACIÓN	NIVEL ACEPTACIÓN
Tiempo de Respuesta	1	Alta	0,90	Alta
Utilización de recursos	0,7	Media	0,66	Media
Conformidad de la eficiencia	0,7	Media	0,50	Media
PORCENTAJE	100,00%	PORCENTAJE	93,33%	

Realizado por: HUEBLA, Byron, 2018

Mediante el análisis de la tabla anterior se puede decir que el Sistema Académico Diller es 93,33 % eficiente, cálculo realizado de mediante la suma de los valores numéricos de sus métricas, aplicada a una regla de tres simple.

A continuación, se detalla en la **Figura 3-3**, un cuadro estadístico de barras con los valores requeridos en comparación con los valores obtenidos con la finalidad de tener un bosquejo más amplio de la eficiencia del Sistema Académico Diller.



**Figura 4-3:** Evaluación de eficiencia del sistema

Realizado por: HUEBLA, Byron, 2018

En la figura anterior muestra la evaluación de la eficiencia del sistema Académico el mismo que tiene dos columnas donde la barra de color azul es el resultado esperado y la barra de color gris es el resultado obtenido

## CONCLUSIONES

- Se realizó el Análisis de la situación actual del Sistema académico de la Unidad Educativa Experimental “Interandina”, donde se determinó la existencia de 5 módulos que fueron implementados en el Sistema Académico denominado “DILLER”.
- La metodología SCRUM utilizada en el desarrollo del sistema académico facilitó la implementación y adaptación de las modificaciones, permitiendo realizar sus tareas en tiempos diferentes y de manera independiente.
- Después de haber analizado la herramienta tecnológica, para el proceso de diseño se utilizó el estándar adoptado a nivel internacional, Lenguaje Unificado de Modelado UML, gracias a que es un lenguaje gráfico se logró identificar actores, entidades, estructuras y comportamiento de los procesos del sistema académico.

## RECOMENDACIONES

- Dar el uso correcto y aprovechar al máximo todas las facilidades y ventajas que el Sistema Académico brinda a todos quienes forman parte de la Unidad Educativa Experimental “Interandina”
- Se recomienda para el desarrollo de aplicaciones web el uso de del framework Laravel por su robustez y simplicidad de usarlo como herramienta tecnológica, facilitando el proceso de desarrollo con una visión considerable, además, se adapta a un considerable número entorno de construcción de software.
- En mayor parte se ha aplicado los conocimientos académicos recibidos en la EIS, pero dichos conocimientos no fueron suficientes, por lo tanto, se recomienda acudir a la auto formación mediante la investigación ante eventos o tareas aún no conocidas.

## BIBLIOGRAFÍA

**ANDINO GUEVARA, C. J.** *Desarrollo de un sistema académico orientado a la Web para la Unidad Educativa Juan de Velasco utilizando Symfony MySQL*. Escuela Superior Politécnica de Chimborazo, Facultad de informática y electrónica, Escuela de Ingeniería en Sistemas Riobamba-Ecuador. 2016.

**Anton, C.** *Laravel, el mejor Framework en PHP*. [En línea], 2015. [Consulta: 20 de Abril 2018]. Disponible en: <https://platsi.com/bloh/laravel-framework-php/>

**Castillo, E.** *Laravel un framework PHP excelente (Rompe las expectativas)*. [En línea], 2018. [Consulta: 4 de junio 2018]. Disponible en: <http://www.cav.jovenclub.cu/comunidad/node/562>.

**De la Torre Muñoz, A.** “*Introduccion a Node.js V8.0*”, [En línea], 2013. [Consulta: 05 de marzo 2018]. Disponible en: [https://dl.dropboxusercontent.com/u/2990198/nodejskoans/Introduccion\\_a\\_Nodejs\\_a\\_traves\\_de\\_Koans\\_ebook.pdf](https://dl.dropboxusercontent.com/u/2990198/nodejskoans/Introduccion_a_Nodejs_a_traves_de_Koans_ebook.pdf)

**De la Torre Muñoz, A.** “*Introduccion a Socket.io*”, [En línea], Copyright (C) Arturo Muñoz de la Torre, pp 157-188. 2013. [Consulta: 09 de mayo del 2018] Disponible en: [https://dl.dropboxusercontent.com/u/2990198/nodejskoans/Introduccion\\_a\\_Nodejs\\_a\\_traves\\_de\\_Koans\\_ebook.pdf](https://dl.dropboxusercontent.com/u/2990198/nodejskoans/Introduccion_a_Nodejs_a_traves_de_Koans_ebook.pdf).

**Dointech, C.** “*Sistemas de Control de Accesos*”. [En línea], 2013. [Consulta: 09 de Mayo del 2018] Disponible en: <http://www.dointech.com.co/control-de-acceso.html>

**Gilfillan, I.** *La biblia de MySQL*. [En línea], Caracas-Venezuela, 2016. [Consulta: 25 de marzo de 2018] Disponible en: [http://es.slideshare.net/rex\\_raptor/labibliademysqldodosobreelgestordebasededatosmysql](http://es.slideshare.net/rex_raptor/labibliademysqldodosobreelgestordebasededatosmysql).

**Hernández Sampieri, R.** “*Metodología de la investigación*.”. [En línea]. 2015, [Consulta: 21 de mayo del 2018] Disponible en: <http://www.dgsc.go.cr/dgsc/documentos/cecadec/metodologia-de-la-investigacion.pdf>

**Largo C.** *Guía técnica para evaluación de software*. [En línea]. Bogotá - Colombia, 2009. [Consulta: 04 de marzo de 2018] Disponible en: [https://jrvargas.files.wordpress.com/2009/03/guia\\_tecnica\\_para\\_evaluacion\\_de\\_software.pdf](https://jrvargas.files.wordpress.com/2009/03/guia_tecnica_para_evaluacion_de_software.pdf)

**MALÁN GUARANGA, F. G. (2016).** *Desarrollo de un sistema de Gestión para la Biblioteca de la unidad educativa Hualcopo Duchicela empleando el Framework Laravel*. [Trabajo de Titulación],

Escuela Superior Politécnica de Chimborazo. Escuela de Ingeniería en Sistemas. Riobamba-Ecuador.2016.

**NOGALES, S.** *Desarrollo de sistema web basado en tecnologías gis para el mapeo de campos electromagnéticos en la ciudad de Riobamba.* [En línea]. [Trabajo de Titulación], Escuela Superior Politécnica de Chimborazo. Facultad de Informática y Electrónica. Escuela de Ingeniería en sistemas informáticos. Riobamba - Ecuador. [Consulta: 04 de julio de 2016] Disponible en:<http://dspace.esPOCH.edu.ec/handle/123456789/475#sthash.rmSsX9MG.dpuf>

**Núñez, J.** *Laravel, Ventajas del framework PHP de moda.* [En línea] 2014, [Consulta: 3 de junio de 2018] Disponible en: <http://www.nubelo.com/blog/ventajas-del-framework-modalaravel/>.

**Ornbo, G.** “¿Programación Node.js”. [En línea] Madrid - España, : Grupo Anaya, S.A., 2014, [Consulta: 27 mayo del 2018] Disponible en: <https://www.nextu.com/web/>

**Ortiz, L.** *Sistema informático de seguimiento de la información docente ocasional y titular de la facultad de informática y electrónica* [En línea]. [Trabajo de Titulación], Escuela Superior Politécnica de Chimborazo. Facultad de Informática y Electrónica. Escuela de Ingeniería en sistemas informáticos. Riobamba-Ecuador. [Consulta: 04 de junio de 2018] Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/4745#sthash.hPTTrNrx.dpuf>

**Otwell, & Taylor.** *Desarrollando Webs Dinámicas.* [En línea] 2013. [Consulta: 04 de julio de 2018] Disponible en: <http://desarrollandowebdinamicas.blogspot.com/2013/03/que-es-laravel.htm>

**Pims, T.** *Laravel un gran framework en PHP.* [En línea] 2018. [Consulta: 05 de junio de 2018] Disponible en: <https://www.arpentetechnologies.com/es/blog/deseno-web/laravel-un-gran-framework-en-php/>

**Roger, P.** “Software Engineering.” . [En línea] Boston: USA, McGraw-Hill., 2001. [Consulta: 05 de Junio del 2018] Disponible en: [http://dinus.ac.id/repository/docs/ajar/RPL-7th\\_ed\\_software\\_engineering\\_a\\_practitioners\\_approach\\_by\\_roger\\_s.\\_pressman\\_.pdf](http://dinus.ac.id/repository/docs/ajar/RPL-7th_ed_software_engineering_a_practitioners_approach_by_roger_s._pressman_.pdf)

**Rosell Tejada, J. & Villalón Huerta, A.** “Shutdown.” . [En línea] , 2015. [Consulta: 23 de julio del 2017] Disponible en: [www.shutdown.es/une195.pdf](http://www.shutdown.es/une195.pdf)

**THE GROUP PHP ESPAÑA,** *PHP: ¿Qué puede hacer PHP?* - [En línea], 2018. [Consulta: 04 de junio de 2018]. Disponible en: <http://php.net/manual/es/intro-whatcando.php>

**Vargas, M. Á.** *Desarrolloweb.com.* [En línea], 2018. [Consulta: 07 de junio de 2018]. Disponible en: <http://www.desarrolloweb.com/articulos/392.php>



## **Anexo B. Manual de usuario**

El documento Manual de Usuario se encuentra adjunto en el CD adjunto al presente trabajo de titulación.

## **Anexo C. Encuesta Aplicada a los Usuarios**

Encuesta aplicada a los usuarios para medir la funcionalidad, eficacia, usabilidad, y fiabilidad.

### **¿Tiene el conjunto de funciones apropiadas para las tareas específicas?**

- 1 Totalmente desacuerdo
- 2 En desacuerdo
- 3 Ni de acuerdo ni en desacuerdo
- 4 De acuerdo
- 5 Totalmente de acuerdo

### **¿El Sistema Realiza lo que fue acordado en forma esperada y correcta?**

- 1 Totalmente desacuerdo
- 2 En desacuerdo
- 3 Ni de acuerdo ni en desacuerdo
- 4 De acuerdo
- 5 Totalmente de acuerdo

### **¿Presenta fallas por defectos o errores?**

- 1 Totalmente desacuerdo
- 2 En desacuerdo
- 3 Ni de acuerdo ni en desacuerdo
- 4 De acuerdo

5 Totalmente de acuerdo

**¿Es fácil de entender y reconocer la estructura y la lógica en su funcionamiento?**

1 Totalmente desacuerdo

2 En desacuerdo

3 Ni de acuerdo ni en desacuerdo

4 De acuerdo

5 Totalmente de acuerdo

**¿Es fácil de aprender a usar?**

1 Totalmente desacuerdo

2 En desacuerdo

3 Ni de acuerdo ni en desacuerdo

4 De acuerdo

5 Totalmente de acuerdo

**¿Es fácil de operar y controlar?**

1 Totalmente desacuerdo

2 En desacuerdo

3 Ni de acuerdo ni en desacuerdo

4 De acuerdo

5 Totalmente de acuerdo

**¿Es atractivo el diseño del software?**

1 Totalmente desacuerdo

2 En desacuerdo

3 Ni de acuerdo ni en desacuerdo



4 De acuerdo

5 Totalmente de acuerdo

**¿El tiempo de respuesta en la ejecución de una función es rápida?**

1 Totalmente desacuerdo

2 En desacuerdo

3 Ni de acuerdo ni en desacuerdo

4 De acuerdo

5 Totalmente de acuerdo