



**UNIVERSIDAD DE TALCA
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA CIVIL EN COMPUTACIÓN**

**Libro de clases virtual y detección de alumnos para
observación**

NICOLÁS EDUARDO PRADENAS VARGAS

Profesor Guía: RODRIGO PAVEZ MADARIAGA

Memoria para optar al título de
Ingeniero Civil en Computación

Curicó – Chile
Octubre, 2020

CONSTANCIA

La Dirección del Sistema de Bibliotecas a través de su encargado Biblioteca Campus Curicó certifica que el autor del siguiente trabajo de titulación ha firmado su autorización para la reproducción en forma total o parcial e ilimitada del mismo.



Two circular official stamps and handwritten signatures in blue ink. The left stamp is from the 'DIRECCIÓN SISTEMA DE BIBLIOTECAS UNIVERSIDAD DE TALCA' and the right stamp is from the 'SISTEMA DE BIBLIOTECAS CAMPUS CURICO'.

Curicó, 2022

Dedicado a Cecilia Vargas y Ricardo Pradenas, mis padres

AGRADECIMIENTOS

A mi madre Cecilia Vargas Vergara, por ser un pilar fundamental durante todo mi desarrollo personal y académico. Su fuerza y carácter han sido una fuente de inspiración durante toda mi vida, y estoy totalmente agradecido de ella por ayudar a convertirme en el hombre que soy hoy en día.

A mi padre Ricardo Pradenas Villalobos, que desde el momento que lo conocí, supe que me ayudaría en cada etapa de mi vida. Me ha acompañado en cada momento importante y espero que siga así durante mucho tiempo más. Con cada palabra o consejo que me ha entregado, se que me ha influenciado para seguir el camino correcto.

A gran parte de mi familia, a mis abuelos Liliam Vergara y José Vargas, quienes siempre estuvieron ahí para mí en todo momento. A mi tío Fernando Vargas, por las salidas de fin de semana, las conversaciones honestas, los consejos y el tiempo que pasamos juntos. A mis hermanos Héctor y Alonso, por todos esos momentos felices y alocados que hemos pasado todos estos años. A mis primas Carolina Luna y Celeste Coutiño, siento que he ganado dos hermanas teniéndolas siempre a mi lado.

A mis amigos, Juan, Álvaro y Cristóbal, los compañeros de casa conocidos en Curicó como los Manolis. A mi gran amigo Jaime, un pilar importante en mi estancia en Curicó. A Christian, Gabriel, Juan Pablo y Raimundo, el mejor grupo de amigos que pude tener en San Javier. Al mejor equipo que pudo ir a la ICPC, Fabián, Diego y al coach (couch) Erik. A Catalina, quien siempre creyó en mí.

A mi profesor guía Rodrigo Pavez, por todo su apoyo, comprensión, confianza y dedicación. De él aprendí que, a pesar de tener problemas, siempre se puede seguir adelante para conseguir las metas propuestas.

A los profesores Rodrigo Paredes, Federico Meza y Luis Silvestre, por ser profesionales y profesores de calidad, que siempre han demostrado la pasión que tienen por lo que hacen.

TABLA DE CONTENIDOS

| | página |
|--|-------------|
| Dedicatoria | I |
| Agradecimientos | II |
| Tabla de Contenidos | III |
| Índice de Figuras | VI |
| Índice de Tablas | VIII |
| Resumen | IX |
| 1. Introducción | 10 |
| 1.1. Descripción del Contexto | 10 |
| 1.2. Definición del problema | 12 |
| 1.3. Objetivos | 13 |
| 1.3.1. Objetivos generales | 13 |
| 1.3.2. Objetivos específicos | 13 |
| 1.4. Propuesta de solución | 14 |
| 1.5. Alcances | 15 |
| 2. Marco teórico | 16 |
| 2.1. Conceptos | 16 |
| 2.2. Trabajos relacionados | 18 |
| 2.3. Tecnologías utilizadas | 20 |
| 2.3.1. Contexto | 20 |
| 2.3.2. Herramientas | 21 |
| 3. Metodología | 24 |
| 3.1. Metodología de desarrollo de Software | 24 |
| 3.1.1. FDD | 24 |
| 3.1.2. Roles de FDD | 26 |
| 3.1.3. Lista de Características | 27 |

| | | |
|-----------|---|-----------|
| 3.1.4. | Ruta crítica | 28 |
| 3.1.5. | Planificación de una característica | 29 |
| 3.2. | Metodología de pruebas | 31 |
| 3.2.1. | Pruebas de usabilidad | 31 |
| 3.2.2. | Pruebas de caja negra | 31 |
| 4. | Desarrollo | 33 |
| 4.1. | Diseño | 33 |
| 4.1.1. | Arquitectura física | 33 |
| 4.1.2. | Arquitectura lógica | 34 |
| 4.2. | Modelo de datos | 36 |
| 4.3. | Iteraciones | 36 |
| 4.3.1. | Iteración 1: 13 al 26 de mayo del 2019 | 37 |
| 4.3.2. | Iteración 2: 20 al 30 de abril del 2020 | 37 |
| 4.3.3. | Iteración 3: 1 de 13 de mayo del 2020 | 39 |
| 4.3.4. | Iteración 4: 14 al 27 de mayo del 2020 | 40 |
| 4.3.5. | Iteración 5: 28 de mayo al 10 de junio del 2020 | 42 |
| 4.3.6. | Iteración 6: 11 al 23 de junio del 2020 | 43 |
| 4.3.7. | Iteración 7: 24 de junio al 7 de julio del 2020 | 44 |
| 5. | Pruebas y resultados | 46 |
| 5.1. | Pruebas de caja negra | 46 |
| 5.2. | Pruebas de usabilidad | 48 |
| 5.3. | Resultados pruebas de caja negra | 48 |
| 5.3.1. | Prueba T1 | 49 |
| 5.4. | Resultados prueba de usabilidad SUS | 50 |
| 6. | Conclusiones y trabajo futuro | 52 |
| | Bibliografía | 55 |
| | Anexos | |
| A: | Pruebas de caja negra | 59 |

| | |
|--|-----------|
| B: Resultados pruebas de caja negra | 66 |
| B.1. Prueba T1 | 66 |
| B.2. Prueba T2 | 66 |
| B.3. Prueba T3 | 67 |
| B.4. Prueba T4 | 68 |
| B.5. Prueba T5 | 69 |
| B.6. Prueba T6 | 70 |
| B.7. Prueba T7 | 70 |
| B.8. Prueba T8 | 71 |
| B.9. Prueba T9 | 72 |
| B.10. Prueba T10 | 73 |
| B.11. Prueba T11 | 74 |
| B.12. Prueba T12 | 75 |
| B.13. Prueba T13 | 76 |
| B.14. Prueba T14 | 77 |
| C: Resultados prueba SUS | 79 |

ÍNDICE DE FIGURAS

| | página |
|--|--------|
| 2.1. Representación del patrón MVC | 21 |
| 3.1. Representación visual de la ruta crítica del proyecto | 29 |
| 3.2. Flujo de trabajo de una iteración | 30 |
| 3.3. Diagrama que representa el flujo de una prueba de caja negra | 32 |
| 4.1. Diagrama que representa la arquitectura física del sistema | 33 |
| 4.2. Diagrama simplificado que representa la arquitectura lógica del sistema | 34 |
| 4.3. Diagrama Entidad Relación que representa el modelo de datos del proyecto | 36 |
| 4.4. Diálogo para crear cursos | 38 |
| 4.5. Diálogo para crear asignaturas | 38 |
| 4.6. Diálogo para crear estudiantes | 39 |
| 4.7. Diálogo para crear apoderados | 39 |
| 4.8. Vista del detalle de una asignatura | 40 |
| 4.9. Vista para calificar una evaluación | 41 |
| 4.10. Vista de promedios de una asignatura | 42 |
| 4.11. Vista de estudiantes que se encuentran en observación por comportamiento | 43 |
| 4.12. Diálogo para agregar una cuenta de profesor | 44 |
| 4.13. Diálogo para cargar una asignatura a un profesor | 44 |
| 4.14. Vista de la página de inicio desde el servidor web | 45 |
| 5.1. Formato prueba de usabilidad SUS modificada | 48 |
| 5.2. T1 - Datos de entrada | 49 |
| 5.3. T1 - Salida obtenida | 49 |
| 5.4. Prueba de usabilidad - Hoja 2 firmada por Rodrigo Cancino | 50 |
| B.1. T2 - Datos de entrada | 66 |
| B.2. T2 - Salida obtenida | 67 |
| B.3. T3 - Datos de entrada | 67 |
| B.4. T3 - Salida obtenida | 68 |
| B.5. T4 - Datos de entrada | 68 |
| B.6. T4 - Salida obtenida | 69 |

| | |
|--|----|
| B.7. T5 - Datos de entrada | 69 |
| B.8. T5 - Salida obtenida | 69 |
| B.9. T6 - Datos de entrada | 70 |
| B.10. T6 - Salida obtenida | 70 |
| B.11. T7 - Datos de entrada | 71 |
| B.12. T7 - Salida obtenida | 71 |
| B.13. T8 - Datos de entrada | 72 |
| B.14. T8 - Salida obtenida | 72 |
| B.15. T9 - Datos de entrada | 73 |
| B.16. T9 - Salida obtenida | 73 |
| B.17. T10 - Datos de entrada | 74 |
| B.18. T10 - Salida obtenida | 74 |
| B.19. T11 - Datos de entrada | 75 |
| B.20. T11 - Salida obtenida | 75 |
| B.21. T12 - Datos de entrada | 76 |
| B.22. T12 - Salida obtenida | 76 |
| B.23. T13 - Datos de entrada | 77 |
| B.24. T13 - Salida obtenida | 77 |
| B.25. T14 - Datos de entrada | 78 |
| B.26. T14 - Salida obtenida | 78 |
| | |
| C.1. Prueba de usabilidad - Hoja 1 | 79 |
| C.2. Prueba de usabilidad - Hoja 2 Cecilia Vargas | 80 |
| C.3. Prueba de usabilidad - Hoja 2 Karla Riquelme | 80 |
| C.4. Prueba de usabilidad - Hoja 2 Margarita Córdova | 81 |
| C.5. Prueba de usabilidad - Hoja 2 Maria Soto | 81 |
| C.6. Prueba de usabilidad - Hoja 2 Valeska Fuentes | 82 |

ÍNDICE DE TABLAS

| | página |
|--|--------|
| 2.1. Tabla comparativa de los sistemas existentes | 20 |
| 5.1. Prueba #1 caja negra - Registro de curso | 47 |
| 5.2. Promedio y desviación estándar en base a los resultados de las encuestas . | 51 |
| A.1. Prueba #2 caja negra - Registro de asignatura | 59 |
| A.2. Prueba #3 caja negra - Registro de estudiante | 60 |
| A.3. Prueba #4 caja negra - Registro de apoderado | 60 |
| A.4. Prueba #5 caja negra - Registro de evaluación | 61 |
| A.5. Prueba #6 caja negra - Registro de asistencia | 61 |
| A.6. Prueba #7 caja negra - Registro de observación | 62 |
| A.7. Prueba #8 caja negra - Registro de actividad | 62 |
| A.8. Prueba #9 caja negra - Cálculo y actualización de promedios | 63 |
| A.9. Prueba #10 caja negra - Ingreso de estudiante a observación | 63 |
| A.10. Prueba #11 caja negra - Edición de estudiante en observación | 64 |
| A.11. Prueba #12 caja negra - Registro de cuenta de usuario profesor | 64 |
| A.12. Prueba #13 caja negra - Designación de asignaturas a un profesor | 65 |
| A.13. Prueba #14 caja negra - Usuario ingresa al sistema | 65 |

RESUMEN

El manejo de los registros escolares, es un problema que afecta a una gran cantidad de establecimientos educacionales que poseen libros de clases en formato manuscrito, esto porque todos los procesos se realizan de forma manual, sin contar con las restricciones que están impuestas por el mismo sistema educativo. En este contexto, los profesores deben invertir tiempo para planificar sus clases, calificar evaluaciones, ingresar la asistencia diaria, registrar observaciones por comportamiento y verificar la conducta de los estudiantes.

Actualmente, la gestión de estos registros se realiza mediante el uso de aplicaciones procesadoras de texto y planillas de cálculo, sin embargo, podrían aparecer problemas por la utilización de estas herramientas, esto porque se requiere costear las licencias para cada uno de los profesionales del establecimiento u obtener licencias piratas para su utilización.

Ante esto, se detecta la necesidad de un nuevo sistema o plataforma, que ayude a los profesores a centralizar toda la información de los distintos cursos en una sola aplicación, con el fin de agilizar el proceso de manipulación de los registros académicos.

En base a lo anterior, se propone un sistema web capaz de manejar la información de los cursos, las asignaturas, estudiantes, apoderados, evaluaciones, asistencia, observaciones y actividades. Además, permite calcular promedios y obtener reportes de la asistencia diaria de los cursos. También, se incluye funcionalidad que permite detectar estudiantes que tengan mal comportamiento, sin considerar lo que posean un bajo rendimiento académico, debido a que este proceso se realiza en base a las evaluaciones y cada establecimiento utiliza distintos criterios.

La plataforma se construye utilizando *Feature Driven Development*, como metodología de desarrollo de software. Se detalla el avance logrado en cada iteración, indicando que característica y módulos se completan, con el fin de que el lector pueda percibir de forma clara el avance incremental al aplicar la metodología.

Dentro de este documento, se especifican detalles claves para el desarrollo del sistema, tales como arquitectura física, arquitectura lógica y el modelo de datos que lo soportan. Para finalizar, la validación de la plataforma se realiza utilizando pruebas de caja negra para corroborar la completitud de las características, además de las pruebas de usabilidad SUS, para identificar las apreciaciones que tienen los usuarios respecto al sistema.

1. Introducción

El presente capítulo está orientado a describir las bases que llevan a la definición del trabajo realizado para esta memoria. En primer lugar, se describe el contexto en el cual se han detectado los problemas abordados, luego se definen los objetivos a cubrir, la propuesta de proyecto y finalmente los alcances del desarrollo.

1.1. Descripción del Contexto

Cuando se habla del trabajo de un profesor de enseñanza prebásica, básica o media, no se tiene una noción de la cantidad de tiempo que debe invertir tanto dentro como fuera de las aulas de clases. Para tener una idea, según un informe presentado en septiembre del 2016 por la Organización para la Cooperación y el Desarrollo Económico (OCDE), los docentes chilenos trabajan 2006 horas al año, de las cuales 1146 corresponden a horas dentro de una sala de clases [1].

Además de tener su horario dentro de las aulas, deben cumplir con horas destinadas a la planificación de clases, corregir evaluaciones, asistir a reuniones de carácter académico, atender apoderados, registrar las calificaciones en el libro de clases u otras actividades similares [2]. Las actividades que no logran ser completadas durante la jornada laboral son llevadas a los hogares para finalizarlas, en caso de tener un libro de clases físico, los docentes no tienen permitido sacar el libro del establecimiento educacional, esto porque es un elemento requerido ante una posible fiscalización por parte de la Superintendencia de Educación [3].

Según la Circular número 1 para establecimientos subvencionados, municipales y privados, Versión 04 del año 2014, existen dos opciones para el formato del libro de clases,

electrónico o manuscrito, debiendo cumplir con la estructura y contenidos mínimos entregados en dicha circular, además de lo señalado por el Ministerio de Educación. Cabe destacar, para los libros de clases en formato electrónico, los establecimientos educacionales deben entregar los antecedentes requeridos a la Superintendencia de Educación para que se lleven a cabo las fiscalizaciones, asimismo, deben generar reportes, los cuales tienen que contener la misma información que se encuentran en los libros de clases en formato manuscrito [3].

El caso que aborda este documento, hace referencia a los establecimientos educacionales que cuentan con el formato manuscrito del libro de clases, con la finalidad de que utilicen una herramienta que les permita continuar con sus actividades, tales como, registrar calificaciones, obtener información tanto de las notas y observaciones sin la necesidad de sacar el libro de clases de los recintos. Asimismo, se enfoca en un contexto bastante específico: el uso de la aplicación en el Colegio Particular Piaget de la comuna de San Javier.

El proceso de llenado del libro de clases, inicia al momento de realizar la etapa de matrícula de los estudiantes para el año académico, estos quedan registrados en una planilla de cálculo, hasta que el profesor los ingresa al libro físico.

Las funciones que debe cumplir un docente se separan en dos categorías: las tareas que desarrolla dentro del aula, tales como, registrar la asistencia, ingresar las actividades realizadas durante la clase o dejar una observación del comportamiento de un estudiante. Por otra parte, se encuentran las actividades que se llevan a cabo fuera de una sala de clases, por ejemplo, la revisión de evaluaciones, registrar calificaciones, calcular promedios, realizar resúmenes de los periodos académicos, etc.

Si se considera un contexto global, en lo que se refiere al completar las actividades que no están relacionadas con la realización de las clases, algunos profesores utilizan distintas herramientas de software para facilitar el trabajo, como es el caso del colegio Piaget. Por ejemplo, software de planillas de cálculo, el cual es empleado para registrar las calificaciones de los distintos cursos y asignaturas, mientras que otra aplicación utilizada consiste en procesadores de texto, para guardar las actividades y las planificaciones de las clases. Sin embargo, el uso de estas herramientas requiere el pago de las licencias respectivas para todos los funcionarios del establecimiento, o en su defecto, se emplean licencias piratas

del software, esto podría causar infección del equipo por algún *malware*, falta de soporte en caso de fallas, generar un *backdoor* para ataques de ciberdelincuentes, etc. Por otra parte, al utilizar este tipo de software, los docentes deben tener en consideración la forma de gestionar los archivos generados, esto porque se tienen que separar por las asignaturas que dictan en cada uno de los cursos.

Un problema subyacente, para los establecimientos que no poseen herramientas de gestión académica, es el mantener los libros de clases en formato manuscrito de años anteriores, en caso de que se presente la Superintendencia de Educación a realizar una fiscalización. Con la nueva normativa, en lugar de guardar los libros físicos, se permite mantener los reportes generados de años anteriores como respaldo, lo que representaría un ahorro de espacio en bodega, además de apoyar a la organización de los documentos del establecimiento.

1.2. Definición del problema

En esta sección se describen las necesidades identificadas que ayudan a definir la solución a entregar.

1. La falta de una herramienta que cubra la necesidad de gestionar de los recursos académicos de cada asignatura, tales como, calificar evaluaciones, marcar asistencia, registrar observaciones del comportamiento de estudiantes y agregar las actividades realizadas en clases.
2. La carencia de un sistema que permita automatizar el proceso de cálculo de promedios, además de detectar a los estudiantes que requieran ser enviados a observación por su comportamiento. Actualmente estos procesos se deben realizar de forma manual, a menos de que se utilice alguna herramienta de cálculo para verificar los promedios y revisar las observaciones negativas de un estudiante. No se considera al estudiantado que tenga bajo rendimiento académico, esto porque no existe una estandarización del proceso de envío de estudiantes a reforzamiento en las distintas asignaturas.
3. La ausencia de una aplicación para generar y mantener reportes finales de cada periodo y año académico, esto produce la acumulación de material de años anteriores,

además de dificultar el proceso de fiscalización por parte de la Superintendencia de Educación. Hoy en día, estos documentos se encuentran guardados en la dirección del establecimiento, por lo que se debe ir a buscar el material para ser entregado al fiscalizador.

4. La ausencia de una tecnología que posea todos los módulos necesarios para su utilización en un establecimiento educacional, o que requiera de una gran cantidad de recursos para ser implementado. Esto porque las tecnologías existentes, no poseen todas las funcionalidades requeridas por los establecimientos educacionales chilenos, por lo tanto, se deben implementar nuevos complementos para su utilización. Por otra parte, dichas tecnologías contienen módulos extras que complican la utilización de la aplicación.
5. La falta de una herramienta que sea desarrollada en conjunto con el establecimiento educacional, esto quiere decir, a pesar de existir ofertas en el mercado que puedan satisfacer parte de las carencias de una institución educativa, no hay una aplicación que esté desarrollada en base a las necesidades de dicha institución.

1.3. Objetivos

En esta sección se definen tanto el objetivo general como los objetivos específicos del proyecto.

1.3.1. Objetivos generales

- Reducir el tiempo que los profesores deben dedicar para gestionar los recursos del libro de clases, en base a una herramienta web que permitan administrar dichos recursos.

1.3.2. Objetivos específicos

- Crear un ambiente seguro para cada asignatura dictada, en el cual se van a registrar evaluaciones, marcar asistencia, ingresar observaciones del comportamiento de los estudiantes y agregar actividades.
- Diseñar e implementar un entorno seguro donde se almacenan las calificaciones de los estudiantes.

- Establecer un entorno de estudiantes que son enviados a observación por comportamiento.
- Construir una herramienta que permita la generación de reportes requeridos para la finalización de cada periodo y año académico.

1.4. Propuesta de solución

Se considera el desarrollo de una plataforma web, que permita a los profesores manejar los recursos del libro de clases.

En primer lugar, se le va a permitir al administrador de la aplicación, que es designado por el colegio, registrar los cursos, cargar las asignaturas por curso y crear las cuentas de acceso a los profesores para vincularlos a las asignaturas correspondientes. Por otra parte, los profesores asignados para cada curso serán los encargados de registrar a los estudiantes y sus apoderados.

El software permitirá a los profesores iniciar sesión con su cuenta personal, visualizar los cursos en los que hace clases y sólo las asignaturas que dicta en cada curso. En cada asignatura, el docente tiene a disposición los siguientes módulos:

- Registro de actividades.
- Marcar asistencia.
- Gestión de evaluaciones.
- Cálculo de promedios, que depende del tipo de periodo académico (bimestre, trimestre o semestre).
- Registro de observaciones.
- Generación de reportes.

Además de poder utilizar las mismas funcionalidades que un profesor, el administrador es capaz de utilizar los siguientes módulos:

- Estudiantes en observación.
- Gestión de periodos académicos.

- Revisión de datos históricos.

Cabe destacar que la solución una vez proporcionada para su utilización, será entregada como software libre, esto para que la plataforma pueda competir con las soluciones de pago ofrecidas en el mercado.

1.5. Alcances

Dentro del desarrollo de este proyecto se han definido los siguientes alcances:

- Este trabajo se limita a ser desarrollado en base al contexto aplicado dentro del Colegio Particular Piaget de la comuna de San Javier. A pesar de ello, se quiere que sea escalable, para algún futuro ser utilizada en cualquier otro establecimiento educacional.
- Se va a generar un producto mínimo viable para que el establecimiento optimice los procesos que se desarrollen de forma manual.

2. Marco teórico

El siguiente capítulo está enfocado en proveer al lector los conocimientos necesarios para comprender cada uno de los conceptos referentes al desarrollo de la plataforma, así como el trabajo relacionado e información sobre las tecnologías utilizadas.

2.1. Conceptos

Esta sección define conceptos básicos necesarios para comprender el contexto del desarrollo del proyecto.

- **Aplicación web:** Herramientas que los usuarios pueden usar accediendo a un servidor web a través de internet o de una intranet. Para acceder a dichas aplicaciones sólo se requiere un navegador [4].
- **Framework:** Es una plataforma para desarrollar aplicaciones de software, proporciona una base sobre la cual los desarrolladores pueden crear una aplicación para una plataforma específica [5].
- **Gestor de Bases de Datos:** Conjunto de programas que permiten el almacenamiento, modificación y extracción de la información en una base de datos [6].
- **Sistema educativo:** En Chile se divide en cuatro niveles:
 - **Parvulario:** Que comprenden desde la sala cuna hasta kínder.
 - **Educación básica:** Compuesta por los cursos que se encuentran desde primero hasta octavo año básico.

- **Educación media:** Se encuentran los cursos desde primero hasta el cuarto año medio.
- **Educación superior:** Se componen de cuatro tipos de establecimientos, los Centros de Formación Técnica (CFT), Institutos Profesionales (IP), Universidades e Instituciones de educación superior de las Fuerzas Armadas [7].
- **Colegio Particular:** Establecimiento educacional perteneciente a una persona o grupo de ellas, no obtienen financiamiento por parte del estado por lo que la matrícula y mensualidad deben ser cubiertas por parte del apoderado.
- **Colegio Particular Subvencionado:** Establecimiento educacional privado, que, a diferencia del anterior recibe financiamiento por parte del estado en donde un segmento de la población escolar se puede ver beneficiado con becas para no pago de matrícula, mensualidad o ambas.
- **Colegio Público:** Establecimiento educacional perteneciente al estado, por lo tanto, el ministerio es el dueño y sostenedor de dicha institución.
- **Profesor jefe:** Profesor a cargo de un curso, es el encargado de coordinar todas las actividades extraacadémicas que realizan en el curso.
- **Libro de clases:** Corresponde al registro que debe poseer todo establecimiento educacional por cada curso que se imparta, esto a contar desde el primer día de clases del presente año escolar.
- **Actividades:** Son el conjunto de tareas que son realizadas en una asignatura en específico, su formato consiste en la fecha en las que se realiza las actividades y los contenidos abordados.
- **Hoja de Vida:** Sección donde al alumno se le dejan observaciones en base a su comportamiento.
- **Observaciones:** Registro del comportamiento de un estudiante o curso completo, dependiendo de las acciones, ésta puede ser positiva o negativa.
- **UTP:** Unidad Técnica Pedagógica. En esta unidad es donde se validan los contenidos que serán abordados en cada curso y cada asignatura, además, de revisar los temas de todas las evaluaciones antes de ser realizadas.

- **Jefe de UTP:** Encargado de la UTP.
- **Alumno en observación:** Estudiante que debe tener un seguimiento por parte del Profesor encargado del curso y Jefe de UTP, con el propósito de poder tomar acciones correctivas o verificar los avances que tiene en su conducta.

2.2. Trabajos relacionados

Con respecto al software encontrado, se encuentran las siguientes propuestas:

- **Colegium** [8]. Ofrece servicios para las instituciones de carácter público y subvencionados, en esa plataforma online se administra las calificaciones, asistencia, horarios, actividades extracurriculares, toda la información académica y administrativa de los alumnos. Asimismo, posee un libro de clases digital para llevar el registro académico y la generación de certificados. En el caso de las instituciones privadas, entrega un servicio de gestión escolar que permite la administración de información relevante del colegio de manera simple, con la posibilidad de generar informes. Mientras tanto, el resto de las aplicaciones disponibles se encuentran por separado, incluyendo el sistema de detección de alumnos para ser enviados a observación.
- **Papinotas** [9]. Se caracterizan por entregar soluciones tecnológicas con foco pedagógico, agrupando en una sola plataforma todas las necesidades del establecimiento para una óptima gestión escolar. Los servicios que poseen son los siguientes: comunicación escuela-familia, campaña de convivencia escolar, asistencia, gestión escolar, plataforma de asistencia digital, libro de clases digital, planificaciones, biblioteca y aula virtual.
- **Syscol** [10]. Es un sistema de gestión escolar de pago, posee un libro de clases virtual el cual permite registrar calificaciones, asistencia, atrasos, salidas y contenidos de la clase. Además, ofrece reportes y análisis por alumno, asignatura y curso. Maneja la matrícula de los alumnos nuevos y antiguos, posee un sistema de biblioteca escolar para gestionar los libros, materiales, préstamos y lectores.
- **TECED** [11]. Es un sistema de gestión capaz de administrar una institución educativa en los ámbitos administrativos, escolares y financieros. Permite administrar los planes de estudio, matrículas, fichas con estudiantes, promociones, cambios de

curso, etc. Permite registrar asistencia, justificativos y reportes inmediatos. Además, se pueden ingresar notas de forma asistida, permite subir planificaciones que son verificadas desde UTP.

- **Legios** [12]. Es un sistema de gestión escolar web, ofrece una completa configuración para cada periodo académico (fechas, días festivos, configuración horaria, jornadas, etc.). Cuenta con un libro de clases virtual en donde los profesores pueden ver la ficha general del alumno, ingresar observaciones, asistencia, evaluaciones y comunicarse con alumnos y/o apoderados.
- **Google Classroom** [13]. Esta herramienta considera los módulos para la asignación de actividades para los alumnos, además de tener la capacidad de poder calificar las tareas y descargar los archivos. Se encuentra enlazado con las otras herramientas como Google Docs y Drive. Además, ofrece al docente la posibilidad de fijar y realizar reuniones con los alumnos.
- **Moodle** [14]. Esta herramienta considera los siguientes módulos: Gestión de los usuarios, accesos y roles, gestionar recursos didácticos y actividades de formación, controlar y hacer seguimiento del proceso de aprendizaje de los alumnos, evaluar a los alumnos y generar reportes, establecer vías de comunicación entre el profesor y los alumnos, etc.
- **Microsoft Excel y Microsoft Word**: Son las herramientas utilizadas por varios docentes. Excel se basa en una plantilla que es rellena con la nómina de alumnos y sus calificaciones, mientras que Word es un editor de texto el cual da formato a la información, y donde además pueden agregar tablas que mantienen su orden. En el contexto en el que nos encontramos, Excel y Word se ajustan a las necesidades básicas del colegio. Sus interfaces basadas en texto y celdas las hace herramientas que satisfacen las necesidades de los usuarios.

| | Colegium | Papinotas | Syscol | TECED | Legios | Classroom | Moodle | Libro Virtual | |
|---------------------------|----------|-----------|---------|---------|---------|-----------|----------|---------------|----------|
| Gestión de Evaluaciones | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Marcar Asistencia | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Registro de Observaciones | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ | ✓ | |
| Registro de Actividades | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Comunicación con Alumnos | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ | |
| Gestión Horaria | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ | |
| Tipo de Licencia | De pago | | De pago | De pago | De pago | De pago | Gratuita | Gratuita | Gratuita |

Cuadro 2.1: Tabla comparativa de los sistemas existentes

Teniendo en cuenta la actualidad de algunos establecimientos educacionales, sobre todo en comunas con una menor densidad poblacional, se hace difícil la obtención de un sistema de gestión escolar, esto es debido a que no se posee una solvencia financiera para realizar una inversión de este tamaño.

Por otro lado, la cantidad de elementos que se consideran extras, sumando los módulos que deben ser implementados y el desconocimiento del funcionamiento de las nuevas tecnologías, no se hace viable el desarrollo basado en software existente.

2.3. Tecnologías utilizadas

En la presente sección se describen las tecnologías utilizadas durante el desarrollo del proyecto y una breve justificación del porqué de la elección.

2.3.1. Contexto

Como se ha mencionado previamente, el sistema consiste en una plataforma web. En donde la mayor interacción se va a encontrar del lado del usuario, por lo que se recomienda utilizar una arquitectura de tres capas basada en MVC (modelo-vista-controlador) [15].

- Modelo:** Las clases que se utilizan en el modelo, son usadas para implementar la lógica del dominio de datos. Estas son las que se utilizan para recuperar, insertar,

actualizar o eliminar la información desde la base de datos asociada a nuestra aplicación [15].

- **Vistas:** Las vistas son las clases que consisten en la interfaz de la aplicación. Los usuarios utilizan la interfaz para que puedan interactuar con la aplicación [15].
- **Controlador:** Las “clases controladores”, son las encargadas de realizar las acciones solicitadas por los usuarios. Estas clases funcionan con la clase modelo y seleccionan la vista adecuada en la que debe mostrar al usuario lo que ha sido solicitado [15].

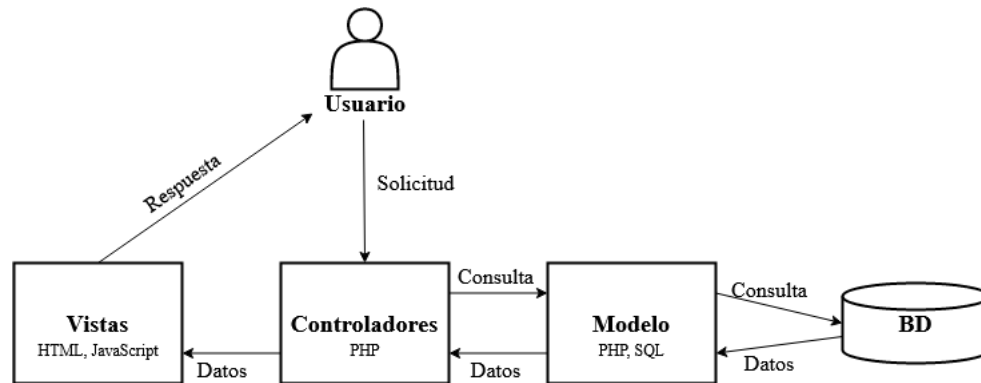


Figura 2.1: Representación del patrón MVC

2.3.2. Herramientas

Durante el desarrollo del proyecto se ha utilizado un *framework* para la aplicación web. Esto se conversó con los clientes al momento de presentar la idea, por lo que la elección de las tecnologías queda a libertad del desarrollador del proyecto.

A continuación, se presentan las herramientas seleccionadas para la construcción del proyecto:

Framework: CodeIgniter

CodeIgniter es un poderoso *framework* basado en PHP con una huella muy pequeña, creado para programadores de PHP que requieren un conjunto de herramientas simples y

elegantes para el desarrollo de aplicaciones web con todas las funciones [16].

Dentro de las ventajas que presenta CodeIgniter, se pueden detallar las siguientes [16]:

- Liviano, rápido, simple, fácil de aprender.
- La migración de un servidor a otro es fácil, sólo se requiere cambiar la URL.
- Su instalación es muy sencilla, hay que descargar la carpeta CodeIgniter, extraer el archivo comprimido y pegarlo en el servidor.
- Una de las características más importantes es la completa documentación que posee.
- Permite la utilización de Ajax y jQuery.

Motor de Bases de Datos: MySQL

Es un gestor de bases de datos relacionales más usados y reconocidos del mercado, se caracteriza porque es fácil de utilizar. Al ser una herramienta de código abierto, permite a los desarrolladores tener una solución fiable y estandarizada de las aplicaciones [17].

Dentro de las ventajas que presenta MySQL se encuentran las siguientes:

- Es gratis, sólo se requiere instalar y está listo para su utilización.
- Es rápida y poderosa para bases de datos, tanto pequeñas como medianas.
- Se va actualizando periódicamente: los desarrolladores lanzan nuevas características de forma constante, incluyendo soporte para transacciones y procedimientos almacenados [18].

Alternativas

- **ASP.NET Core:** Framework de desarrollo web para crear aplicaciones en la plataforma ASP.NET. Es una versión de código abierto que se ejecuta en macOS, Linux y Windows. Pueden existir varias versiones en un mismo servidor, esto quiere decir, la aplicación puede adoptar la última versión, mientras que las otras aplicaciones siguen ejecutándose en la versión que se probaron [19].
- **Microsoft SQL Server:** Gestor de bases de datos relacionales que presenta escalabilidad, estabilidad y seguridad. Gracias a sus funciones de memoria integrada

directamente en los motores SQL Server, mejora en gran medida la flexibilidad y facilita su uso [20].

3. Metodología

Este capítulo detalla la metodología utilizada para construir la solución al problema en cada etapa del proyecto, además de la definición de la lista de características a desarrollar y la ruta crítica a seguir para su implementación. Por último, se identifica la metodología de pruebas que se le realizan al sistema. Estas pruebas se enfocan en demostrar el cumplimiento tanto de los objetivos generales como los específicos.

3.1. Metodología de desarrollo de Software

Como no se requiere hacer énfasis en los requisitos del proyecto y se necesita flexibilidad, pues en caso de que se precise realizar cambios al sistema, la metodología seleccionada será una adaptación a FDD (*Feature Driven Development*). La decisión se basa en la experiencia del autor con metodologías ágiles y porque no es necesario un gran adiestramiento para utilizar dicho procedimiento [21].

3.1.1. FDD

FDD consiste en un enfoque ágil para el desarrollo de software. En este planteamiento no se hace énfasis en la obtención de requisitos, sino que en las fases de diseño y construcción. Además, se hace hincapié en aspectos de calidad durante todo el desarrollo del proyecto e incluye un seguimiento permanente del avance del sistema [21].

Lo normal en los proyectos con iteraciones cortas, es que se consideran procesos iterativos orientados a características, en donde se comienzan con los requerimientos y el modelado, el diseño y la construcción basada en incrementos. La diferencia radica en que existen fases que se realizan solamente una vez y al llegar a la etapa de construcción, se

vuelve un proceso iterativo generando un producto funcional con nuevo valor agregado con respecto a la iteración anterior [21].

A continuación, se presentan los elementos más relevantes de la metodología y como se adapta a las condiciones del presente proyecto.

Develop an Overall Model

Se presenta un paso a paso inicial en lenguaje de alto nivel sobre el alcance del sistema y su contexto. A continuación, se construye el esqueleto de un primer modelo del sistema a cargo del Arquitecto en Jefe.

El dominio se separa en distintas áreas que se deben desarrollar y generar un modelo más detallado de dicha área. Una vez finalizada, se juntan todas las áreas y se genera un modelo global [21].

Build a Feature List

Se identifican las características, se agrupan, se priorizan y se ponderan. En las iteraciones siguientes del proceso, el equipo se divide en subgrupos que se especializan en áreas relacionadas a las características [21].

Plan by Feature

En base a la lista de características de la etapa anterior, se establecen hitos y se diseña un cronograma de diseño y construcción [21].

Design by Feature

Ya en la fase iterativa de la metodología, se toma la siguiente característica a ser diseñada. Se trabaja en el diagrama de secuencia correspondiente y a cada clase involucrada se le realiza una descripción, además de incluir los métodos que debe poseer [21].

Build by Feature

Se agregan los métodos a cada clase de la característica correspondiente, luego se realiza el testing unitario para cada una de dichas clases y se hace una inspección del código.

Después, se elabora un main build en el cual se hace la integración con la funcionalidad antes realizada, finalizando con el testing de integración correspondiente [21].

Adaptación de la metodología

En el ambiente que se encuentra el proyecto, se requiere realizar cambios menores a la metodología presentada. En cuanto a los procesos, se invierte el orden de las fases de diseño y construcción de lista de características, esto porque se necesita obtener una lista informal con las funcionalidades antes de avanzar al diseño de la aplicación, por lo tanto, es conveniente para el proyecto que se cambie el orden de dichas etapas.

3.1.2. Roles de FDD

Un equipo de FDD está compuesto por un *Project Manager*, *Chief Architect*, *Development Manager*, *Chief Programmer* y *Class Owner*. Estos roles son detallados a continuación:

- **Project Manager:** Es el líder administrativo y financiero del proyecto. Una de sus tareas principales consiste en evitar distracciones externas y que el equipo pueda trabajar en óptimas condiciones. En la metodología, es la persona que tiene la última palabra sobre temas referidos al alcance, tiempo y personal [21].
- **Chief Architect:** Es el encargado del diseño global del sistema y de la ejecución de todas las etapas del diseño. Tiene la última palabra de todas las decisiones que estén relacionadas [21].
- **Development Manager:** Lleva diariamente las actividades del desarrollo y resuelve los posibles conflictos que se puedan dar en el equipo. Asimismo, este rol tiene la responsabilidad de resolver problemas con respecto a los recursos [21].
- **Chief Programmer:** Es un desarrollador con experiencia, el cual participa en el análisis de requerimientos y diseño del proyecto. Es el encargado de guiar a los pequeños equipos en el análisis, diseño y construcción de nuevas funcionalidades. Selecciona las características que serán realizadas en la siguiente iteración. Resuelve problemas técnicos, junto con reportar el progreso del equipo semanalmente [21].

- **Class Owner:** Es el encargado del desarrollo de las clases que se le asignaron como propias. Para cada iteración, participa en la decisión de cuáles serán incluidas en la lista de funcionalidades del próximo ciclo [21].

3.1.3. Lista de Características

En esta sección se presenta una adaptación a la lista de características formal, estas características están agrupadas por nombre, las cuales presentan un conjunto de módulos a implementar.

Característica Curso

- Crear, ver, editar, eliminar Cursos.
- Crear, ver, editar, eliminar Asignaturas de Cursos.
- Crear, ver, editar, eliminar Estudiantes de Cursos.
- Crear, ver, editar, eliminar Apoderado de un Estudiante.

Característica Asignatura

- Crear, ver, editar, eliminar Evaluaciones.
- Crear, ver, editar, eliminar Observaciones de un Estudiante.
- Marcar la asistencia de la asignatura.
- Crear, ver, editar, eliminar Actividad de la asignatura.

Característica Evaluación

- Calificar evaluaciones por Asignatura.
- Calcular promedios de las asignaturas por periodo académico.

Característica Alumnos en Observación

- Agregar de manera automática estudiantes a lista de observación (tres observaciones de tipo negativo envían al alumno a observación).
- Editar mensaje del alumno en observación.
- Eliminar alumno en observación.

Característica Cuentas de Usuario

- Crear cuenta de modo administrador.
- Crear, ver, editar o eliminar cuentas de profesores.
- Inicio de sesión con contraseñas cifradas.
- Agregar asignaturas a un profesor.

Característica Archivos

- Cargar archivo Excel con la nómina de los estudiantes.
- Exportar archivos con promedios de asignaturas.
- Exportar archivos con promedios de cursos.

Para definir el orden en el que se desarrolla el proyecto, se utilizó el método de la ruta crítica.

3.1.4. Ruta crítica

El método de la ruta crítica es utilizado para el cálculo de tiempos y plazos en la planificación de proyectos. Consiste en una secuencia de los elementos terminales en la realización de un proyecto, se determina el tiempo más corto en el que es posible completar todas las actividades. La duración de la ruta crítica determina el tiempo real que se requiere para finalizar el proyecto [22].

Lo que se necesita para utilizar el método de la ruta crítica es construir un modelo que incluya lo siguiente [22]:

- Una lista con todas las actividades requeridas para finalizar la aplicación (generalmente se incluye un desglose del trabajo a realizar).
- La duración que lleva completar cada actividad.
- Las dependencias entre actividades.
- Puntos finales lógicos, tales como hitos o elementos entregables.

Las herramientas otorgadas por el método de la ruta crítica, permiten al usuario seleccionar un punto final lógico para el proyecto, además de identificar la serie más larga de actividades dependientes [22]. En el contexto del presente proyecto, las actividades se desarrollan de forma lineal, por lo tanto, la generación de la ruta crítica está basada en las dependencias entre las distintas características.

Una vez que se define la lista de características, presentadas en el presente capítulo, el siguiente paso es definir la duración que se requiere para la implementación. En este caso, se necesita una iteración de dos semanas para terminar una característica.

En segundo lugar, debemos identificar las características que tengan dependencias, en este caso en particular, las dependencias son las siguientes:

- La característica Asignatura depende de Curso.
- La característica Cuentas de Usuario necesita a Curso.
- Las características Evaluación y Alumnos en Observación dependen de Asignatura.
- La característica Archivos necesita que esté lista Evaluación.

Para finalizar, los hitos están establecidos al término de cada iteración, esto quiere decir, cada vez que se termine una característica se obtiene un nuevo incremento en el proyecto.

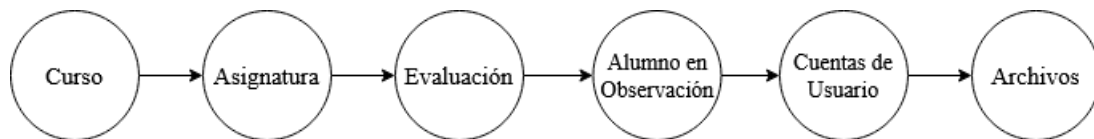


Figura 3.1: Representación visual de la ruta crítica del proyecto

3.1.5. Planificación de una característica

En esta sección, se presentan los procesos necesarios para la planificación de una iteración.

- **Diseño de vistas:** Se define la forma que van a tener las vistas de la característica a desarrollar. En esta etapa se seleccionan los componentes a utilizar.

- **Diseño de clases:** Se definen los métodos que van a utilizar los controladores y el modelo.
- **Implementación de clases y métodos:** Se implementan las clases y los métodos que vienen desde la etapa del diseño. Además, se crean las vistas y se unen a sus respectivos controladores.
- **Pruebas unitarias:** Se realizan pruebas de cada una de las funciones implementadas para su característica, en esta etapa se revisa el comportamiento de la aplicación y en caso de tener fallas, retorna a la etapa de implementación.
- **Integración:** Una vez que la característica pasa las pruebas unitarias, se ejecuta el proceso de integración con el resto de la aplicación.

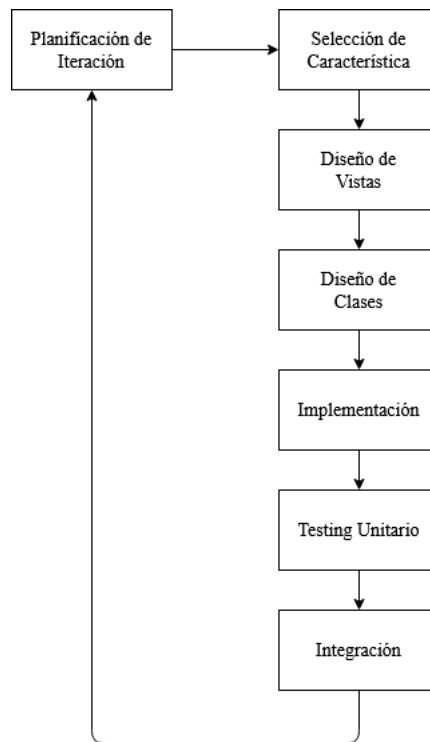


Figura 3.2: Flujo de trabajo de una iteración

3.2. Metodología de pruebas

En esta sección, se presentan las metodologías de pruebas utilizadas para la validación del proyecto, estas se encuentran basadas en el cumplimiento tanto de los objetivos generales como de los objetivos específicos.

3.2.1. Pruebas de usabilidad

El Sistema de escala de usabilidad SUS (*System Usability Scale*), es una escala Likert [23]. Se destaca por ser rápida de administrar y económica de utilizar, corresponde a una de las formas más eficientes de recopilar datos estadísticamente válidos, con el fin de darle al sitio web una puntuación clara y razonablemente precisa [24].

En comparación con otras pruebas, el sistema de escala de usabilidad tiene las siguientes características:

- Bajo costo de implementación: no es un cuestionario extenso, por lo que no requiere una asignación importante de recursos.
- Rapidez para implementar: existen plantillas definidas, por lo que parte del trabajo ya viene implementado. En lugar de investigar y diseñar un nuevo estudio, solo se debe copiar, pegar y ajustar a lo que se necesite [24].

El detalle en profundidad, con respecto al cuestionario y cómo se aplica en el proyecto, se encuentra en el capítulo 5 del presente documento.

3.2.2. Pruebas de caja negra

Las pruebas de caja negra, o también conocidas como pruebas de comportamiento, se utilizan para ver los aspectos funcionales y no funcionales del software, sin tener en cuenta la estructura interna de la aplicación [25].

Con las técnicas de caja negra, se intenta identificar errores del tipo [25]:

- Funciones incorrectas o faltantes.
- Errores de iniciación y terminación.

- Errores de interfaz.
- Errores de estructura.

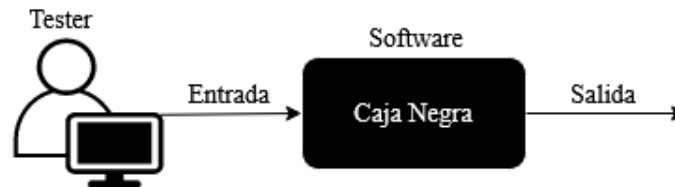


Figura 3.3: Diagrama que representa el flujo de una prueba de caja negra

El detalle en profundidad de la definición y aplicación de las pruebas es abordado en el capítulo 5 del documento.

4. Desarrollo

El presente capítulo describe los elementos y/o etapas claves en la fase de desarrollo de software llevada a cabo en la ejecución de este proyecto.

4.1. Diseño

Esta sección describe una de las etapas más importantes en el desarrollo de software; el diseño del software. En esta fase se definen la arquitectura física y lógica del sistema, pero sin considerar aspectos visuales, debido a que estos se revisan en cada iteración.

La importancia de la fase del diseño consiste en la necesidad de contener la propagación de errores que se podrían introducir en el proyecto.

4.1.1. Arquitectura física

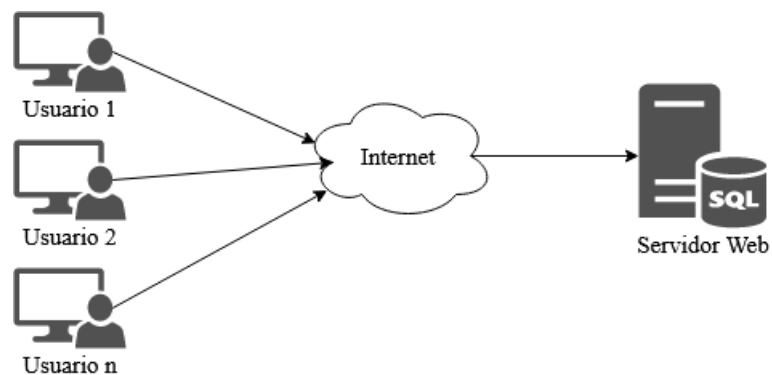


Figura 4.1: Diagrama que representa la arquitectura física del sistema

Se trata de una arquitectura cliente-servidor de n-capas, en su variante de 2 capas, donde la primera son los usuarios y la segunda consiste en el servidor web (donde está incluido el servidor de bases de datos).

En la arquitectura cliente-servidor de dos niveles, el cliente y el servidor interactúan directamente entre sí. En el contexto de la aplicación, el servidor contiene la lógica de la aplicación [26].

Para mejorar la seguridad de la arquitectura que se está utilizando, se ha implementado un certificado SSL (*Secure Socket Layer*), para mantener seguras las conexiones a internet, así como para proteger la información confidencial que se envía entre el usuario y la aplicación, para impedir que posibles delincuentes lean y modifiquen los datos que se transfieren entre el software y los distintos usuarios.

Al momento de escribir este documento, la aplicación se encuentra en proceso de pruebas usando la arquitectura mencionada en esta sección, se encuentra hospedada en Hostinger y la base de datos está siendo ejecutada localmente por el servidor. El servidor se encuentra ubicado en Estados Unidos.

4.1.2. Arquitectura lógica

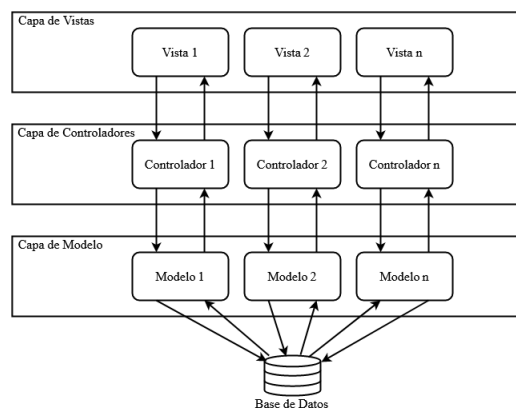


Figura 4.2: Diagrama simplificado que representa la arquitectura lógica del sistema

Como se menciona en el **Apartado 2.3.1**, la arquitectura lógica del proyecto está basada en el patrón de Modelo-Vista-Controlador (MVC). En el contexto en el que se encuentra la aplicación, los componentes del patrón MVC interactúan de la siguiente manera:

- **Modelo – Vista:** Los elementos de las vistas pueden consultar al modelo para obtener información que se le muestra al usuario, esta interacción no es directa debido a que las consultas son recibidas por el controlador para ser enviadas al modelo.
- **Controlador – Modelo:** Las acciones que vienen hacia el controlador desde las vistas, invocan a los métodos del modelo para que realicen las transacciones solicitadas. En las funciones del modelo, se ejecutan las acciones de actualización en la base de datos y esta puede o no retornar una acción [27].
- **Controlador – Vista:** Los controladores responden a las solicitudes originadas desde las vistas, además de ser el encargado de determinar la información que debe ir a las mismas. Por último, desde los controladores se determina que elementos de las vistas son presentados al usuario [27].

Lo que respecta a la capa de las vistas, se realiza una categorización con respecto a los privilegios que tienen los distintos usuarios, es decir, se mantienen separadas en carpetas en base al tipo. En lo que respecta a las vistas compartidas, se mantienen en la carpeta principal de las mismas, con el fin de evitar mantener archivos duplicados.

Los controladores y modelo, implementados en PHP, se encuentran relacionados 1 es a 1. Por cada controlador existe una clase modelo que realiza consultas a la base de datos, dichas consultas están ligadas a un módulo en específico o relaciones en las que se encuentra dicho módulo.

4.2. Modelo de datos

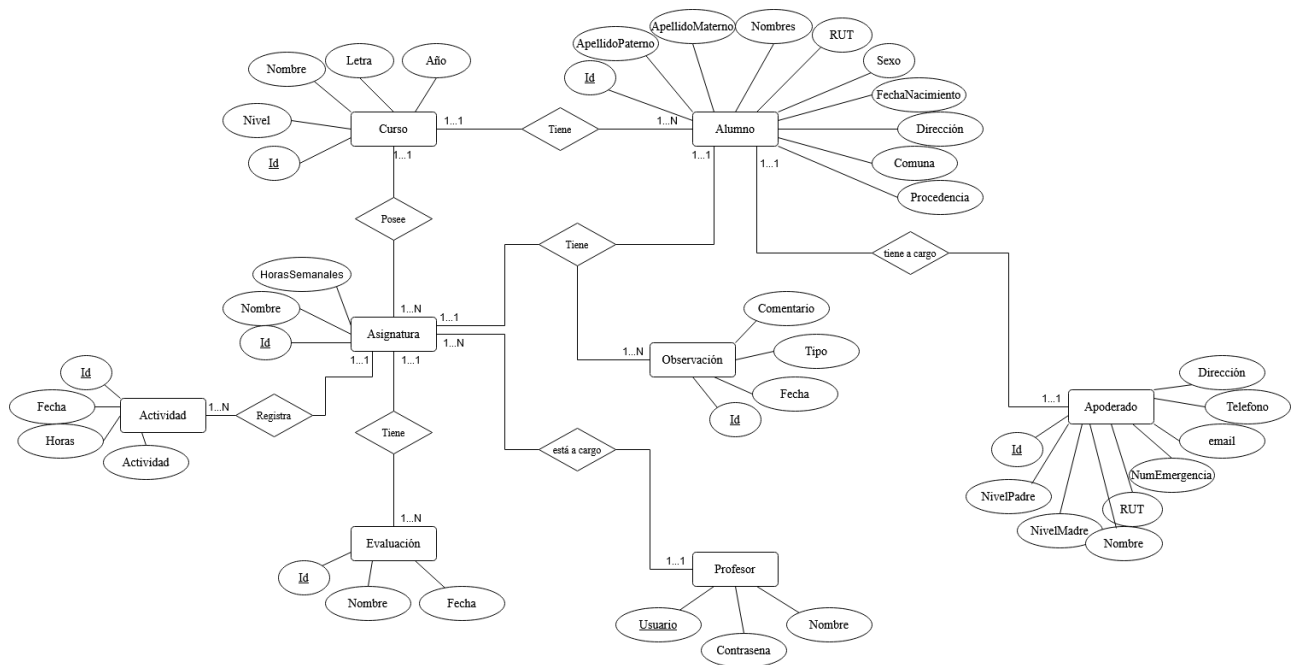


Figura 4.3: Diagrama Entidad Relación que representa el modelo de datos del proyecto

CodeIgniter tiene un enfoque bastante flexible para la utilización del patrón MVC, ya que no requiere la implementación de los modelos. En la situación del presente proyecto, las clases modelos son las encargadas de realizar la conexión y hacer las consultas requeridas a la base de datos.

Existe la necesidad de diseñar y codificar solamente el *script* de la base de datos, las consultas son realizadas con las funciones que vienen implementadas en el *framework*, pero en casos específicos es necesario utilizar el lenguaje de consultas SQL más las funciones.

4.3. Iteraciones

Como se menciona en la **Sección 3.1**, se ha utilizado la metodología *Feature Driven Development* (FDD) y la duración de los *sprints* es de dos semanas. Para completar con el proyecto, se han planificado ocho iteraciones, sin embargo, en base a los cambios en la

planificación propias del desarrollo ágil, se ha ejecutado en siete iteraciones.

Los siguientes apartados enumeran y describen las iteraciones realizadas.

4.3.1. Iteración 1: 13 al 26 de mayo del 2019

Al ser la primera iteración, es usada para la configuración inicial de elementos como:

- Configuraciones iniciales, esto consiste en la creación del repositorio en *GitHub*.
- Levantamiento de características.
- Diseño inicial del modelo de datos.

Durante esta iteración no se presentan avances en el prototipo.

4.3.2. Iteración 2: 20 al 30 de abril del 2020

En esta iteración, se comienza a trabajar en las primeras etapas del prototipo, considerando la característica **Curso**.

Entre otras cosas, durante este *sprint* se realizan las siguientes tareas:

- Finalización del modelo de datos implementado en MySQL.
- Diseño de las vistas Curso, Asignatura, Alumno y Apoderado.
- Diseñar métodos para cada uno de los controladores y modelos (Curso, Asignatura, Alumno, Apoderado).
- Implementación de los métodos anteriormente mencionados. Dichos métodos, están relacionados con el ingreso de información, mostrar los registros de los distintos módulos, además de editar o eliminar algún registro en específico.
- Pruebas unitarias de los distintos módulos.

Para el prototipo se presentan los avances entregados en las Figuras 4.4, 4.5, 4.6 y 4.7.

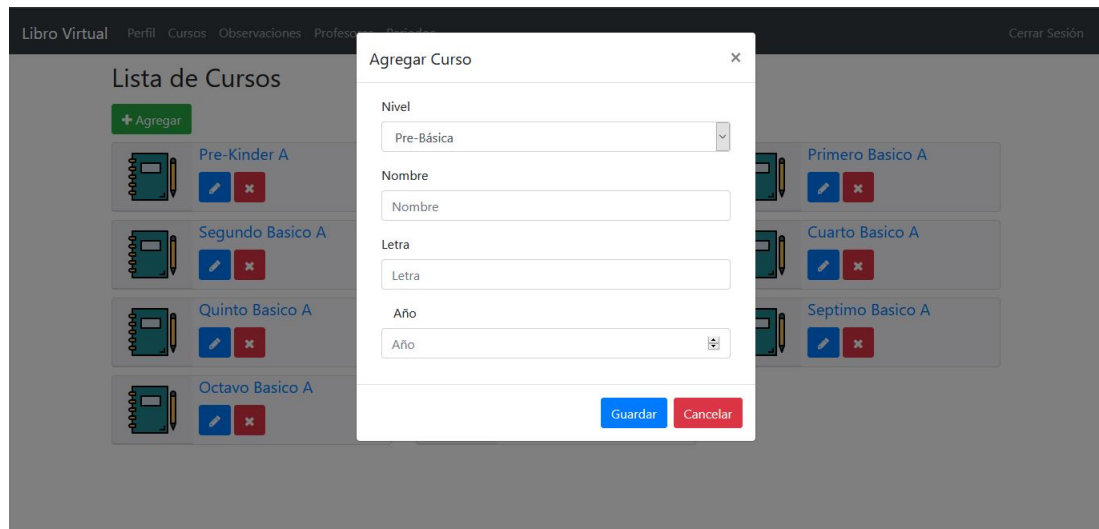


Figura 4.4: Diálogo para crear cursos

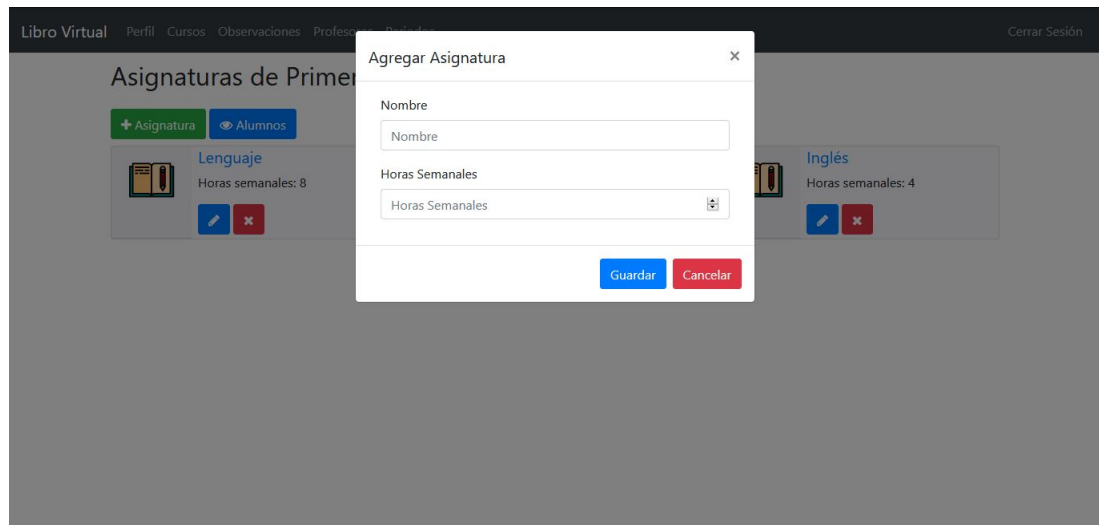


Figura 4.5: Diálogo para crear asignaturas

Libro Virtual Perfil Cursos Observaciones Profesores Períodos

Lista de Alumnos Prin

Apellido Paterno

Apellido Materno

Nombres

RUT

Sexo

Fecha de Nacimiento

Dirección

Comuna

Procedencia

Guardar Cancelar

Figura 4.6: Diálogo para crear estudiantes

Libro Virtual Perfil Cursos Observaciones Profesores Períodos

Apoderado de Catalin

Nivel Educativo Padre

Nivel Educativo Madre

Nombre

RUT

Dirección

Teléfono

Correo

N° de Emergencia

Guardar Cancelar

Figura 4.7: Diálogo para crear apoderados

4.3.3. Iteración 3: 1 de 13 de mayo del 2020

En esta iteración, se continúa con el desarrollo de la aplicación a nivel local, siguiendo con la planificación, la característica implementada es **Asignatura**.

Además, durante este *sprint* se considera un nuevo método para la implementación de la base de datos, se toma en cuenta el diseño del modelo de datos global y a esto se le

modifica lo necesario que se requiera después de diseñar los métodos del modelo.

Durante el desarrollo de este *sprint*, se realizaron las siguientes tareas:

- Diseño de vistas Detalle de Asignaturas, Evaluación, Asistencia, Actividad y Observación.
- Diseño de modelos y controladores Evaluación, Asistencia, Actividad y Observación.
- Implementación de modelos y controladores anteriormente mencionados. Se desarrollan las funciones para agregar, editar y eliminar las evaluaciones, observaciones y actividades, por otra parte, se desarrolla el formulario para marcar la asistencia de los estudiantes en una asignatura en específico.
- Pruebas unitarias de cada módulo.

En las Figura 4.8, se muestran avances de la interfaz del Detalle de una Asignatura.

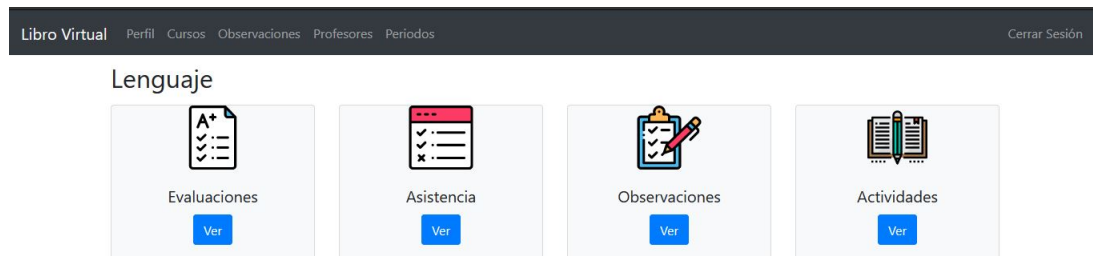


Figura 4.8: Vista del detalle de una asignatura

4.3.4. Iteración 4: 14 al 27 de mayo del 2020

En este *sprint*, se aborda la característica **Evaluación**, lo que implica que en esta iteración ya se incluyen las funcionalidades de calificar las evaluaciones por estudiante, agregar

los periodos académicos requeridos y calcular los promedios por asignatura y periodo.

Las actividades desarrolladas durante este *sprint* son las siguientes:

- Diseño de vistas para Calificar Evaluación, Periodos y Promedios Calculados.
- Diseñar el modelo y controlador de Periodos.
- Agregar al diseño del modelo y controlador de Evaluación los métodos relacionados con la calificación de los estudiantes y los promedios.
- Se implementa el modelo y controlador de Periodos, dicho módulo, es el encargado de determinar las evaluaciones que són utilizadas para el cálculo de promedios. Por otra parte, se agregan la funcionalidades para calificar evaluaciones y obtener los promedios al módulo Evaluación.
- Pruebas unitarias para el módulo.

En las Figuras 4.9 y 4.10, se muestran los avances respectivos a la iteración.



| Apellido Paterno | Apellido Materno | Nombres | Nota |
|------------------|------------------|----------------------|----------------------------------|
| Aguilar | Arriagada | Catalina | <input type="text" value="6.5"/> |
| Alegría | Alvarez | Cristopher Alejandro | <input type="text" value="4.3"/> |
| Arriagada | Castillo | Mia Antonella | <input type="text" value="1.2"/> |
| Ávila | Chamorro | Camila Constanza | <input type="text" value="6.5"/> |

Figura 4.9: Vista para calificar una evaluación



| N° | Apellido Paterno | Apellido Materno | Nombres | Promedio |
|----|------------------|------------------|----------------------|----------|
| 1 | Aguilar | Arriagada | Catalina | 6.4 |
| 2 | Alegría | Alvarez | Cristopher Alejandro | 5.7 |
| 3 | Arriagada | Castillo | Mia Antonella | 6.2 |
| 4 | Ávila | Chamorro | Camila Constanza | 6.1 |

Figura 4.10: Vista de promedios de una asignatura

4.3.5. Iteración 5: 28 de mayo al 10 de junio del 2020

Para esta iteración, se cumple con la característica **Alumno en Observación**, las funcionalidades que se integran se encuentran descritas en la lista de características del **Apartado 3.1.3**.

Las tareas realizadas durante este *sprint* son las siguientes:

- Diseño de vista de Alumnos en observación.
- Agregar al diseño del modelo y controlador Observación los métodos necesarios para las funcionalidades.
- Implementar los métodos en el modelo y controlador mencionado. Al momento de registrar la tercera observación del tipo negativa, el estudiante es enviado de forma automática a la lista de alumnos en observación. El administrador tiene permitido editar el mensaje entregado. Por último, si el administrador lo considera conveniente, se elimina al estudiante de la lista.
- Pruebas unitarias.

Los avances del prototipo se presentan en la Figura 4.11.

| Curso | Alumno | Comentario |
|----------------|----------------------------|--|
| Primer Basic A | Catalina Aguilar Arriagada | Se recomienda que apoderado venga a hablar con jefe de UTP |

Figura 4.11: Vista de estudiantes que se encuentran en observación por comportamiento

4.3.6. Iteración 6: 11 al 23 de junio del 2020

En este *sprint*, la característica implementada corresponde a la integración de **Cuentas de Usuario**, acá se tiene que hacer la diferencia entre el usuario administrador y profesor. Hasta este punto, todo lo desarrollado es utilizado por los dos tipos de usuario, salvo las características relacionadas con la mantención del sistema, se verifican los perfiles de usuario y separan las vistas que corresponden a cada tipo.

Las tareas realizadas durante esta iteración son las siguientes:

- Se diseñan las vistas tanto para el inicio de sesión y gestionar cuentas de profesores.
- Se diseñan los métodos tanto para controladores y modelo.
- Implementación del inicio de sesión, para esto se crea un sistema para la detección de los roles y validación de las credenciales de usuario. El administrador crea las cuentas del tipo profesor, además, vincula las asignaturas con el docente. Por último, se crea un perfil para los usuarios que permite la edición de la contraseña.
- Pruebas unitarias de los módulos.

Los avances presentados en el prototipo se muestran en las Figuras 4.12 y 4.13.

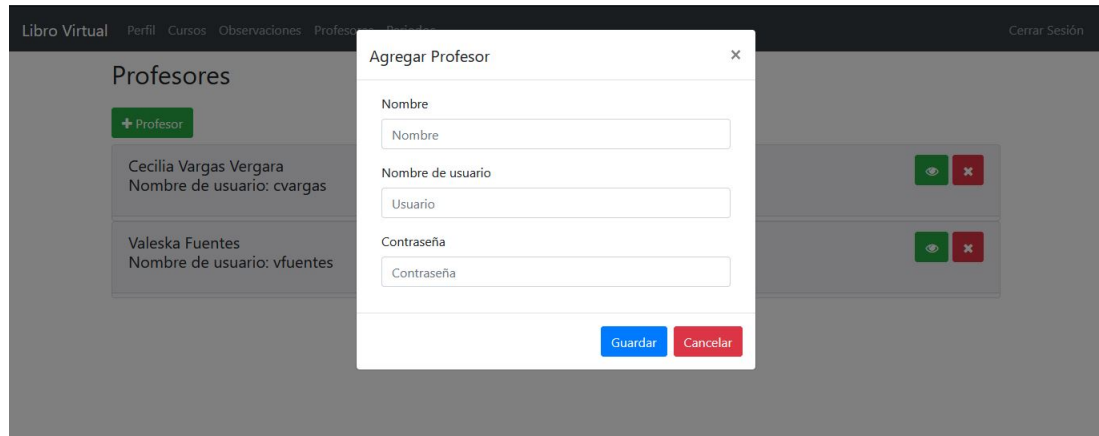


Figura 4.12: Diálogo para agregar una cuenta de profesor

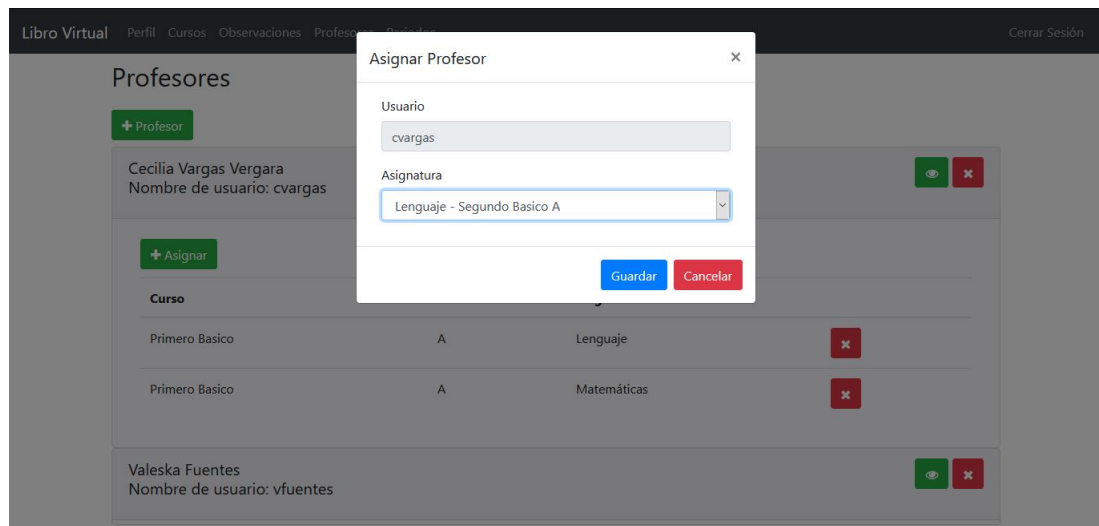


Figura 4.13: Diálogo para cargar una asignatura a un profesor

4.3.7. Iteración 7: 24 de junio al 7 de julio del 2020

En esta iteración, corresponde la subida de la aplicación al servidor web y la compra del dominio para la realización de pruebas de caja negra. Una vez finalizado este *sprint*, la aplicación es entregada para hacer las pruebas de usabilidad.

Las actividades desarrolladas durante este *sprint* son las siguientes:

- Contratación del servicio de hosting *Hostinger*. Con dicho procedimiento se incluye un certificado SSL, dominio y dos meses del alojamiento de la página.

- Montar la aplicación.
- Configurar la base de datos en el servidor.
- Realización de pruebas de caja negra.

Como se muestra en la Figura 4.14, la aplicación ya se encuentra hospedada en el servidor.

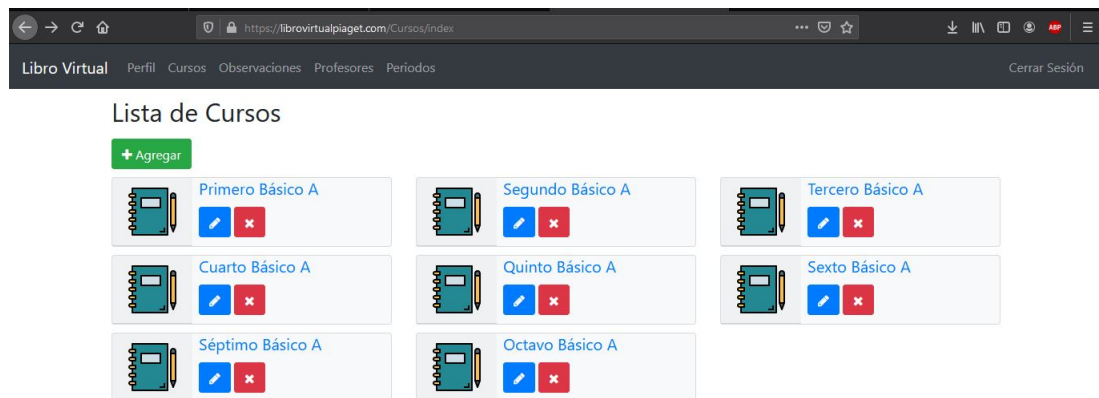


Figura 4.14: Vista de la página de inicio desde el servidor web

5. Pruebas y resultados

Como se menciona en la **Sección 3.2**, se utilizan dos tipos de validaciones: pruebas de caja negra y pruebas de usabilidad (SUS). En este capítulo, se aborda la definición, aplicación y análisis de las mismas.

5.1. Pruebas de caja negra

En esta sección se describen formalmente las pruebas de caja negra a ejecutar. Sin embargo, los antecedentes de las pruebas realizadas están en el apartado de los anexos del documento.

Cabe destacar, las pruebas de caja negra están orientadas a demostrar el cumplimiento de las características. Dichas pruebas fueron realizadas una vez que la aplicación fue subida al servidor web.

Por temas de practicidad, en esta sección se entrega la definición formal sólo de una de las pruebas de caja negra. Para el resto de las pruebas sólo se entrega la descripción en este capítulo, la definición completa aparece en el Apéndice A.

| | |
|-------------------------|---|
| Código: | T1 |
| Característica: | Curso |
| Descripción: | Usuario administrador puede agregar un nuevo curso al sistema. |
| Entrada: | Nivel, Nombre, Letra, Año. |
| Salida Esperada: | <ul style="list-style-type: none"> ■ Curso ingresado en la base de datos. ■ Actualización de lista de cursos. |
| Salida Obtenida: | <ul style="list-style-type: none"> ■ Curso guardado exitosamente. ■ Lista de cursos actualizada. |

Cuadro 5.1: Prueba #1 caja negra - Registro de curso

La descripción del resto de las pruebas de caja negra es la siguiente:

- **T2:** Usuario administrador puede agregar una nueva asignatura al sistema.
- **T3:** Usuario administrador puede agregar un nuevo estudiante al sistema.
- **T4:** Usuario administrador puede agregar un nuevo apoderado al sistema.
- **T5:** Usuario administrador o profesor puede agregar una nueva evaluación al sistema.
- **T6:** Usuario administrador o profesor puede marcar asistencia de una asignatura.
- **T7:** Usuario administrador o profesor puede agregar una nueva observación al sistema.
- **T8:** Usuario administrador o profesor puede agregar una nueva actividad al sistema
- **T9:** Usuario administrador o profesor puede calcular el promedio de los alumnos de una asignatura
- **T10:** El sistema envía estudiante a observación, después de ingresada una tercera observación negativa.
- **T11:** Usuario administrador puede editar el mensaje de un estudiante en observación
- **T12:** Usuario administrador puede agregar una nueva cuenta de profesor.

- **T13:** Usuario administrador puede designar una asignatura a un profesor.
- **T14:** Usuario administrador o profesor puede ingresar al sistema.

5.2. Pruebas de usabilidad

La prueba de usabilidad aplicada en el proyecto está basada en el *framework* SUS (*System Usability Scale*). SUS consiste en una escala Likert, la cual consiste en 10 preguntas realizadas a los usuarios de la aplicación web. Los participantes evalúan cada pregunta en una escala de uno (“Totalmente en desacuerdo”) a cinco (“Totalmente de acuerdo”) [24]. En esta ocasión, el conjunto de preguntas a evaluar se ha modificado levemente, sin cambiar el sentido de ellas. De esta forma la encuesta está más orientada al contexto específico del proyecto.

| Preguntas | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1. Creo que me gustaría usar este sistema con frecuencia | | | | | |
| 2. Creo que el sistema es innecesariamente complejo | | | | | |
| 3. Pensé que el sistema sería más fácil de usar | | | | | |
| 4. Creo que necesito ayuda técnica (de un experto) para usar el sistema | | | | | |
| 5. Las distintas funciones del sistema se complementan entre ellas | | | | | |
| 6. Todas las cosas parecidas funcionan de la misma manera (no hay inconsistencias) | | | | | |
| 7. Creo que el sistema es lo suficientemente simple como para que cualquiera lo use rápidamente | | | | | |
| 8. El sistema me pareció muy engorroso de usar | | | | | |
| 9. Puedo usar el sistema sin temor a equivocarme | | | | | |
| 10. Necesitaba aprender muchas cosas antes de comenzar a usar el sistema | | | | | |

Figura 5.1: Formato prueba de usabilidad SUS modificada

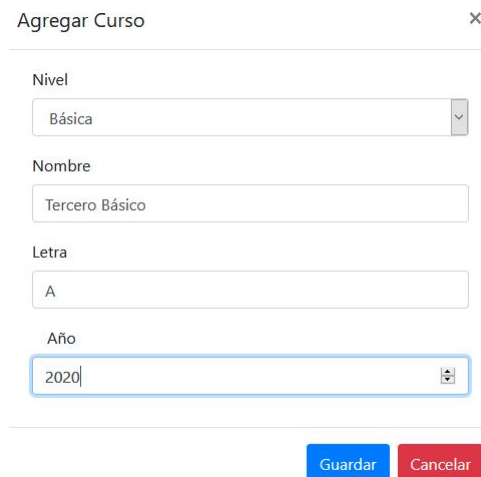
5.3. Resultados pruebas de caja negra

Esta sección demuestra la ejecución de las pruebas de caja negra y la completitud de las mismas. Para minimizar la extensión del documento en este apartado, sólo se incluye

la primera de ellas. Por otra parte, en el Anexo B se incluyen las pruebas restantes que demuestran la aprobación exitosa de la totalidad de pruebas de caja negra descritas en la **Sección 5.1**.

5.3.1. Prueba T1

La Figura 5.2, muestra los datos de entrada para la prueba descrita en el Cuadro 5.1 (nivel: Básica, nombre: Tercero Básico, Letra: A y el año: 2020). Por su parte, la Figura 5.3 muestra la salida obtenida, el registro correcto del curso en la tabla curso.



Formulario "Agregar Curso" con los siguientes campos:

- Nivel: Básica
- Nombre: Tercero Básico
- Letra: A
- Año: 2020

Botones: Guardar (azul), Cancelar (rojo).

Figura 5.2: T1 - Datos de entrada

Lista de Cursos



Lista de Cursos:

- + Agregar
- Primero Básico A
- Segundo Básico A
- Tercero Básico A

Cada curso tiene un ícono de libro y botones de editar (pencil) y eliminar (x).

Figura 5.3: T1 - Salida obtenida

En la especificación de cada prueba de caja negra, se indica que característica corresponde. Esto se diseñó de tal forma que, para cada prueba aprobada, implica la completitud de las funcionalidades de un módulo que pertenece a dicha característica.

5.4. Resultados prueba de usabilidad SUS

Se decidió aplicar la encuesta de usabilidad a cinco docentes del Colegio Piaget de manera aleatoria, además de incluir como administrador al jefe de UTP. La planta total corresponde a 12 profesores, por lo tanto, se considera al cuarenta por ciento del universo total de docentes. Los resultados de las encuestas fueron promediados para obtener una medida de tendencia central representativa, asimismo, la medida de dispersión considerada es la desviación estándar.

La Figura 5.4 corresponde a la segunda parte de la encuesta aplicada a Rodrigo Cancino, jefe de UTP del Colegio Piaget, acá se muestran las respuestas y los comentarios obtenidos. En el Anexo C se incluye la primera parte de la encuesta, además de las respuestas entregadas por cada profesor.

| Preguntas | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1. Creo que me gustaría usar este sistema con frecuencia | | | | | X |
| 2. Creo que el sistema es innecesariamente complejo | X | | | | |
| 3. Pensé que el sistema sería más fácil de usar | X | | | | |
| 4. Creo que necesito ayuda técnica (de un experto) para usar el sistema | X | | | | |
| 5. Las distintas funciones del sistema se complementan entre ellas | | | | | X |
| 6. Todas las cosas parecidas funcionan de la misma manera (no hay inconsistencias) | | | | | X |
| 7. Creo que el sistema es lo suficientemente simple como para que cualquiera lo use rápidamente | | | | | X |
| 8. El sistema me pareció muy engorroso de usar | X | | | | |
| 9. Puedo usar el sistema sin temor a equivocarme | | | | | X |
| 10. Necesitaba aprender muchas cosas antes de comenzar a usar el sistema | X | | | | |

Comentarios: Es un sistema intuitivo, de fácil uso sin complejidades, es susceptible de ser optimizado con más herramientas, sin embargo, cumple con el objetivo como libro virtual de clases.



Firma

Figura 5.4: Prueba de usabilidad - Hoja 2 firmada por Rodrigo Cancino

El resultado obtenido, de un total de 100 puntos, es el siguiente:

$$Puntaje = 32 * 2,5 = 80 \quad (5.1)$$

Con la referencia de la primera encuesta, se computan los puntajes de las otras cinco (67.5, 80, 80, 80, 85). Se calcula el promedio y la desviación estándar de las seis mediciones, y los resultados son los siguientes:

| Dato Estadístico | Valor |
|------------------|-------|
| \bar{x} | 78.75 |
| s | 5.86 |

Cuadro 5.2: Promedio y desviación estándar en base a los resultados de las encuestas

Los resultados de la prueba indican que la usabilidad del sistema “*se encuentra bien, pero falta para ser excelente*”. Los encuestados, en la sección de comentarios, hacen referencia a las funcionalidades y se destacan los siguientes aspectos:

- Facilidad de usar.
- Está todo relacionado, por lo que se puede mantener de forma ordenada los quehaceres del docente.
- Simplifica las labores administrativas en el establecimiento.
- Se requieren integrar más herramientas, pero no se hace una mención de cuáles.
- Para personas con dificultades con la tecnología, con una instrucción básica se podría trabajar sin dificultades.

Posteriormente, se van a realizar nuevas entrevistas para agregar más características al sistema, además se deben acordar reuniones para analizar la aplicación y medir el impacto que podrían tener las nuevas funcionalidades una vez integradas a la plataforma.

6. Conclusiones y trabajo futuro

La presente memoria detalla el diseño, metodología y desarrollo de una plataforma web que gestiona los recursos de un libro de clases, además de detectar estudiantes que son enviados a observación por motivos conductuales. Esta plataforma, solventa las carencias del procedimiento actual de trabajo, asimismo, fomenta el uso de nuevas tecnologías en los colegios, con el fin de ayudar a la eficiencia y facilitación de los procesos.

En aspectos generales, el sistema permite la administración de cursos y las asignaturas pertenecientes a cada uno. También, gestiona la asistencia, evaluaciones, actividades y observaciones de una asignatura. Además, la aplicación detecta las observaciones negativas que tiene un estudiante, cuando tiene 3 o más, es enviado a una lista de observación por comportamiento.

La aplicación en cuestión se ha validado utilizando pruebas de caja negra, las cuales han sido definidas de forma tal, que cada una de ellas está directamente relacionada con al menos un módulo de una característica. Considerando la ruta crítica del proyecto, además de la aprobación de todas las pruebas de caja negra, indica que la plataforma cumple con las funcionalidades para ser un producto mínimo viable. Específicamente, el sistema aprobó el 100% de las pruebas de caja negra, esto quiere decir, que el sistema cumple con todas las características implementadas.

Igualmente, se ha aplicado una encuesta de usabilidad SUS a cinco profesores más el jefe de UTP del colegio, con un promedio de 78.75pts. y una desviación estándar de 5.86pts. El máximo que se puede obtener es 100pts., implicando que *“el sistema es bueno, pero puede ser aún mejor”*. Los resultados obtenidos en esta encuesta permiten la detec-

ción de errores de consistencia, retroalimentación de vital importancia para mejorar la plataforma. Asimismo, entrega una base para añadir más funcionalidades como trabajo futuro.

A modo de desglose del cumplimiento de los objetivos específicos propuestos, se tiene lo siguiente:

- Objetivo “*Crear un ambiente seguro para cada asignatura dictada. . .*” se cumple en su totalidad, esto porque se entrega un sistema que gestiona los recursos de cada asignatura solamente al personal autorizado, además de cifrar la comunicación entre la aplicación y el usuario para evitar la manipulación de la información por parte de terceras personas.
- Objetivo “*Diseñar e implementar un entorno seguro donde se almacenan las calificaciones de los estudiantes*” se cumple en su totalidad, debido a que el acceso a las calificaciones está restringido sólo para los usuarios autorizados, además de cifrar la comunicación entre los usuarios y la aplicación.
- Objetivo “*Establecer un entorno de estudiantes que son enviados a observación. . .*” es cumplido en su totalidad, esto porque los estudiantes que obtienen tres anotaciones negativas en su hoja de vida, son enviados de forma automática a una lista de estudiantes en observación. En este punto, el administrador puede tomar las medidas correspondientes en conjunto con los profesores jefes de cada curso.
- Objetivo “*Construir una herramienta que permita la generación de reportes. . .*” se cumple de manera parcial, esto porque los reportes se visualizan desde la aplicación, pero la herramienta no genera dichos reportes para ser descargados al dispositivo.

Lo anterior está respaldado por los comentarios realizados por los usuarios del sistema, una vez leídos dichos comentarios en la sección de la encuesta de usabilidad SUS.

Por lo tanto, basados en la planificación, falta implementar una característica para obtener el sistema completo, pero la aplicación alcanza a ser un producto mínimo viable. Además, considerando los resultados de las pruebas, se puede afirmar que el sistema cumple con las funcionalidades implementadas y los objetivos específicos, por consiguiente,

también se cumple el objetivo general casi en su totalidad.

Finalmente, a modo de trabajo futuro, se propone una nueva sesión de entrevistas con usuarios, para añadir nuevas características a la aplicación. Además, se pretende implementar un módulo para la detección de estudiantes con bajo rendimiento académico, con la finalidad de estandarizar y automatizar un método para aplicar en los distintos establecimientos. Adicionalmente, se propone entregar el código de la aplicación como software libre, para que pueda ser utilizada de forma gratuita por distintas organizaciones educativas a nivel país.

Bibliografía

- [1] G. Moreno. “*Estudio OCDE: profesores chilenos son quienes más horas pasan en las salas de clases*”. URL: <https://www.latercera.com/noticia/estudio-ocde-profesores-chilenos-son-quienes-mas-horas-pasan-en-las-salas-de-clases/>. (accedido 15-05-2020).
- [2] F. Rivas. “*¿Cuántas horas no remuneradas trabajan los profesores chilenos cada semana?*”. URL: <https://www.biobiochile.cl/noticias/nacional/educacion/2016/06/20/cuantas-horas-no-remuneradas-trabajan-los-profesores-chilenos-cada-semana.shtml>. (accedido 15-05-2020).
- [3] Superintendencia de Educación Escolar. *Circular Número 1 Establecimientos Educativos Subvencionados Municipales y Particulares*. (2014-02-21). [En línea]. URL: http://gobta.supereduc.cl/docs/1_amarillo.pdf.
- [4] Google Sites. “*Aplicación Web*”. URL: <https://sites.google.com/site/smr2teresa/definicion>. (accedido 08-07-2020).
- [5] P. Christensson. “*Framework Definition*”. URL: <https://techterms.com/definition/framework>. (accedido 08-07-2020).
- [6] Google Sites. “*Sistema de gestión de bases de datos*”. URL: <https://sites.google.com/site/basdededatosrelacionales/home/contenido/subtema-1/sistema-de-gestion-de-base-de-datos> (accedido 08-07-2020).
- [7] Ministerio de Educación. “*Niveles y Modalidades de Enseñanza Subvencionada por el Estado*”. URL: <https://www.ayudamineduc.cl/ficha/niveles-y-modalidades-de-ensenanza-subvencionada-por-el-estado-4>. (accedido 08-07-2020).

- [8] Colegium. “*Libro de Clases Virtual*”. URL: <https://www.colegium.com/libro-de-clases-digital/>. (accedido 15-05-2020).
- [9] Papi Notas. “*Servicios*”. URL: <https://papinotas.cl/servicios/>. (accedido 15-05-2020).
- [10] Syscol. “*Syscol.net Sistema Gestión Escolar*”. URL: <https://www.syscol.com/>. (accedido 08-07-2020).
- [11] TECED. “*Gestión Escolar*”. URL: <https://www.teced.cl/gestionescolar/>. (accedido 08-07-2020).
- [12] Legios. “*Sistema Legios*”. URL: <http://www.legios.cl/sistema/>. (accedido 08-07-2020).
- [13] F. Boillos. “*10 herramientas digitales imprescindibles para los profes Flipped Classroom*”. URL: <https://www.magisnet.com/2020/03/10-herramientas-digitales-imprescindibles-para-los-profes-flipped-classroom/>. (accedido 05-11-2020).
- [14] Moodle. “*Educación Primaria*”. URL: <https://moodle.com/es/soluciones/escuelas/>. (accedido 05-11-2020).
- [15] A. Majeed y I. Rauf. “*MVC Architecture: A Detailed Insight to the Modern Web Applications Development*”. Peer Review Journal of Solar & Photoenergy System. Septiembre. 2018. [En línea]. URL: <https://crimsonpublishers.com/prsp/>.
- [16] L. Yicheng. “*Development of a blog system using CodeIgniter framework*”. Trabajo fin de grado. Oamk. Oulu. Finlandia. 2011.
- [17] A. Robledano. “*Qué es MySQL: Características y ventajas*”. URL: <https://openwebinars.net/blog/que-es-mysql/>. (accedido 15-05-2020).
- [18] J. Greenspan y B. Bulger. “*MySQL/PHP Database Applications*”. Foster City. M & T Books. 2001.
- [19] Microsoft. “*What is ASP.NET Core*”. URL: <https://dotnet.microsoft.com/learn/aspnet/what-is-aspnet-core>. (accedido 15-08-2020).
- [20] M. Parada. “*Qué es SQL Server*”. URL: <https://openwebinars.net/blog/que-es-sql-server/>. (accedido 15-05-2020).
- [21] L. Calabria. “*Metodología FDD*”. Cátedra Ing. Software. Dept Ing. ORT. Montevideo. 2003.

- [22] L. Espinal. “*Método de Ruta Crítica - CPM (Critical path method)*”. URL: http://www.cca.org.mx/ps/profesores/cursos/dahdeca/html/m4/acts_eva/ruta.pdf (accedido 15-07-2020).
- [23] J. Brooke. “*SUS – A quick and dirty usability scale*”. Redhatch Consulting Ltd. Reading. Inglaterra.
- [24] N. Thomas. “*How to use the System Usability Scale (SUS) to evaluate the usability of your website*”. URL: <https://usabilitygeek.com/how-to-use-the-system-usability-scale-sus-to-evaluate-the-usability-of-your-website/>.(accedido 13-07-2020).
- [25] J.M. Sánchez. “*Pruebas de Software. Fundamentos y Técnicas*”. Dept. Teoría de la Señal y Comunicaciones. UPM. Madrid. 2015.
- [26] S. Tayib. “*Client-Server Model*”. School of Computing. UUM. Sintok. Malasia.
- [27] R. Grove, E. Ozkan. “*The MVC-Web Design Pattern*”. Dept. Comp. Sc. JMU. Harrisonburg. VA. Estados Unidos.

ANEXOS

A. Pruebas de caja negra

| | |
|-------------------------|---|
| Código: | T2 |
| Característica: | Curso |
| Descripción: | Usuario administrador puede agregar una nueva asignatura al sistema. |
| Entrada: | Nombre, Horas semanales. |
| Salida Esperada: | <ul style="list-style-type: none">■ Asignatura ingresada en la base de datos.■ Actualización de lista de asignaturas en curso. |
| Salida Obtenida: | <ul style="list-style-type: none">■ Asignatura guardada exitosamente.■ Lista de asignaturas de un curso actualizada. |

Cuadro A.1: Prueba #2 caja negra - Registro de asignatura

| | |
|-------------------------|--|
| Código: | T3 |
| Característica: | Curso |
| Descripción: | Usuario administrador puede agregar un nuevo estudiante al sistema |
| Entrada: | Apellido Paterno, Apellido Materno, Nombres, RUT, Sexo, Fecha de Nacimiento, Dirección, Comuna, Procedencia. |
| Salida Esperada: | <ul style="list-style-type: none"> ■ Alumno ingresado a la base de datos. ■ Actualización de lista de alumnos del curso. |
| Salida Obtenida: | <ul style="list-style-type: none"> ■ Alumno guardado exitosamente. ■ Lista de alumnos del curso actualizada. |

Cuadro A.2: Prueba #3 caja negra - Registro de estudiante

| | |
|-------------------------|--|
| Código: | T4 |
| Característica: | Curso |
| Descripción: | Usuario administrador puede agregar un nuevo apoderado al sistema |
| Entrada: | Nivel Educativo Padre, Nivel Educativo Madre, Nombre, RUT, Dirección, Teléfono, Correo, Número de Emergencia. |
| Salida Esperada: | <ul style="list-style-type: none"> ■ Apoderado ingresado a la base de datos. ■ Actualización de lista de apoderado del alumno. |
| Salida Obtenida: | <ul style="list-style-type: none"> ■ Apoderado guardado exitosamente. ■ Lista de apoderados de un alumno actualizada. |

Cuadro A.3: Prueba #4 caja negra - Registro de apoderado

| | |
|-------------------------|---|
| Código: | T5 |
| Característica: | Asignatura |
| Descripción: | Usuario administrador o profesor puede agregar una nueva evaluación al sistema. |
| Entrada: | Nombre, Fecha. |
| Salida Esperada: | <ul style="list-style-type: none"> ■ Evaluación ingresada a la base de datos. ■ Actualización de lista de evaluaciones de una asignatura. |
| Salida Obtenida: | <ul style="list-style-type: none"> ■ Evaluación guardada exitosamente. ■ Lista de evaluaciones de una asignatura actualizada. |

Cuadro A.4: Prueba #5 caja negra - Registro de evaluación

| | |
|-------------------------|---|
| Código: | T6 |
| Característica: | Asignatura |
| Descripción: | Usuario administrador o profesor puede marcar asistencia de una asignatura. |
| Entrada: | Fecha, Asistencia por alumno. |
| Salida Esperada: | <ul style="list-style-type: none"> ■ Asistencia ingresada en la base de datos. ■ Actualización del resumen de asistencia del curso. |
| Salida Obtenida: | <ul style="list-style-type: none"> ■ Asistencia guardada exitosamente. ■ Lista de asistencia de asignaturas es actualizada. |

Cuadro A.5: Prueba #6 caja negra - Registro de asistencia

| | |
|-------------------------|--|
| Código: | T7 |
| Característica: | Asignatura |
| Descripción: | Usuario administrador o profesor puede agregar una nueva observación al sistema. |
| Entrada: | Fecha, Tipo, Comentario. |
| Salida Esperada: | <ul style="list-style-type: none"> ■ Observación ingresada en la base de datos. ■ Actualización de la lista de observaciones de un alumno. |
| Salida Obtenida: | <ul style="list-style-type: none"> ■ Observación ingresada exitosamente. ■ Lista de observaciones de un alumno es actualizada. |

Cuadro A.6: Prueba #7 caja negra - Registro de observación

| | |
|-------------------------|--|
| Código: | T8 |
| Característica: | Asignatura |
| Descripción: | Usuario administrador o profesor puede agregar una nueva actividad al sistema. |
| Entrada: | Fecha, Número de Horas, Actividades. |
| Salida Esperada: | <ul style="list-style-type: none"> ■ Actividad ingresada en la base de datos. ■ Actualización de lista de actividades de una asignatura. |
| Salida Obtenida: | <ul style="list-style-type: none"> ■ Actividad guardada exitosamente. ■ Lista de actividades de una asignatura actualizada. |

Cuadro A.7: Prueba #8 caja negra - Registro de actividad

| | |
|-------------------------|--|
| Código: | T9 |
| Característica: | Evaluación |
| Descripción: | Usuario administrador o profesor puede calcular el promedio de los alumnos de una asignatura. |
| Entrada: | Periodo académico. |
| Salida Esperada: | <ul style="list-style-type: none"> ■ Promedios ingresados a la base de datos. ■ Actualización de promedios de los estudiantes. |
| Salida Obtenida: | <ul style="list-style-type: none"> ■ Promedios guardados exitosamente. ■ Lista de promedios de una asignatura actualizada. |

Cuadro A.8: Prueba #9 caja negra - Cálculo y actualización de promedios

| | |
|-------------------------|--|
| Código: | T10 |
| Característica: | Alumno en Observación |
| Descripción: | El sistema envía estudiante a observación, después de ingresada una tercera observación negativa. |
| Entrada: | Tercera observación negativa a un estudiante. |
| Salida Esperada: | <ul style="list-style-type: none"> ■ Alumno en observación ingresado a la base de datos. ■ Actualización de lista de alumnos en observación. |
| Salida Obtenida: | <ul style="list-style-type: none"> ■ Alumno en observación ingresado exitosamente. ■ Lista de alumnos en observación actualizada. |

Cuadro A.9: Prueba #10 caja negra - Ingreso de estudiante a observación

| | |
|-------------------------|--|
| Código: | T11 |
| Característica: | Alumno en Observación |
| Descripción: | Usuario administrador puede editar el mensaje de un estudiante en observación. |
| Entrada: | Comentario. |
| Salida Esperada: | <ul style="list-style-type: none"> ■ Alumno en observación actualizado en la base de datos. |
| Salida Obtenida: | <ul style="list-style-type: none"> ■ Alumno en observación editado exitosamente. |

Cuadro A.10: Prueba #11 caja negra - Edición de estudiante en observación

| | |
|-------------------------|--|
| Código: | T12 |
| Característica: | Cuentas de Usuario |
| Descripción: | Usuario administrador puede agregar una nueva cuenta de profesor. |
| Entrada: | Nombre, Nombre de Usuario, Contraseña. |
| Salida Esperada: | <ul style="list-style-type: none"> ■ Cuenta de profesor ingresado en la base de datos. ■ Actualización de lista de profesores. |
| Salida Obtenida: | <ul style="list-style-type: none"> ■ Cuenta de profesor guardada exitosamente. ■ Lista de profesores actualizada. |

Cuadro A.11: Prueba #12 caja negra - Registro de cuenta de usuario profesor

| | |
|-------------------------|---|
| Código: | T13 |
| Característica: | Cuentas de Usuario |
| Descripción: | Usuario administrador puede designar una asignatura a un profesor. |
| Entrada: | Asignatura de un curso. |
| Salida Esperada: | <ul style="list-style-type: none"> ■ Asignatura designada al profesor ingresada a la base de datos. ■ Actualización de las asignaturas que dicta un profesor. ■ Asignatura cargada no queda disponible para ser dictada por otro profesor. |
| Salida Obtenida: | <ul style="list-style-type: none"> ■ Designación de asignatura guardada exitosamente. ■ Lista de asignaturas que dicta un profesor actualizada. ■ Asignatura cargada no aparece disponible para ser designada a otro profesor. |

Cuadro A.12: Prueba #13 caja negra - Designación de asignaturas a un profesor

| | |
|-------------------------|---|
| Código: | T14 |
| Característica: | Cuentas de Usuario |
| Descripción: | Usuario administrador o profesor puede ingresar al sistema. |
| Entrada: | Nombre de Usuario, Contraseña. |
| Salida Esperada: | <ul style="list-style-type: none"> ■ Usuario ingresa a la página al sistema.. |
| Salida Obtenida: | <ul style="list-style-type: none"> ■ Usuario acredita sus credenciales de acceso e ingresa exitosamente. |

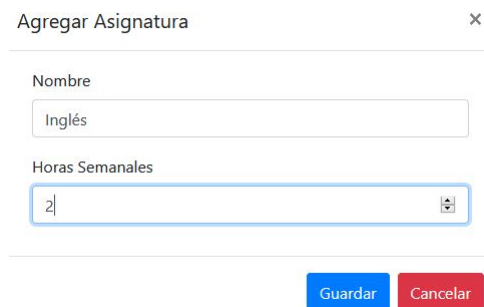
Cuadro A.13: Prueba #14 caja negra - Usuario ingresa al sistema

B. Resultados pruebas de caja negra

B.1. Prueba T1

B.2. Prueba T2

En las Figuras B.1 y B.2 muestran los datos de entrada y salida obtenida para la prueba de caja negra T2.



A screenshot of a web form titled "Agregar Asignatura" with a close button (x) in the top right corner. The form contains two input fields: "Nombre" with the text "Inglés" and "Horas Semanales" with the number "2". Below the fields are two buttons: "Guardar" (blue) and "Cancelar" (red).

Figura B.1: T2 - Datos de entrada

Asignaturas de Primero Básico A

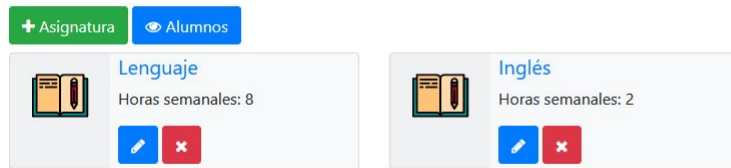


Figura B.2: T2 - Salida obtenida

B.3. Prueba T3

En las Figuras B.3 y B.4 muestran los datos de entrada y salida obtenida para la prueba de caja negra T3. Consiste en ingresar un alumno en un curso.

The image shows a form titled 'Agregar Alumno' with a close button (X) in the top right corner. The form contains the following fields and values:

- Apellido Paterno: Aguilar
- Apellido Materno: Arriagada
- Nombres: Catalina
- RUT: 22.739.346-7
- Sexo: Femenino (dropdown menu)
- Fecha de Nacimiento: 06-06-2008 (calendar icon)
- Dirección: Los Conquistadores 1346, Población Bicentenario
- Comuna: San Javier
- Procedencia: Piaget

At the bottom of the form are two buttons: 'Guardar' (blue) and 'Cancelar' (red).

Figura B.3: T3 - Datos de entrada

Lista de Alumnos Primero Básico A




| Apellido Paterno | Apellido Materno | Nombres | + Alumno |
|------------------|------------------|----------|---|
| Aguilar | Arriagada | Catalina |    |

Figura B.4: T3 - Salida obtenida

B.4. Prueba T4

En las Figuras B.5 y B.6 muestran los datos de entrada y salida obtenida. Agregar un apoderado a un alumno.

Agregar Apoderado ✕

Nivel Educativo Padre

Nivel Educativo Madre

Nombre

RUT

Dirección

Teléfono

Correo

N° de Emergencia

Figura B.5: T4 - Datos de entrada

Apoderado de Catalina Aguilar Arriagada

+ Apoderado



| Nombre | Dirección | Teléfono | Correo | N° de Emergencia | |
|--------------------------|---|----------|--------|------------------|---|
| Paola Arriagada Gonzalez | Los Conquistadores 1346, Población Bicentenario | 61552022 | - | 61552022 |   |

Figura B.6: T4 - Salida obtenida

B.5. Prueba T5

La imagen B.7 muestra los datos de entrada para agregar una evaluación, la salida obtenida en la figura B.8 corresponde a la evaluación creada.

Agregar Evaluación ×

Nombre

Fecha

Figura B.7: T5 - Datos de entrada

Evaluaciones de Lenguaje

+ Evaluación

👁 Promedios




| | |
|---|---|
| Prueba del Libro 1 Fecha: 03-04-2020 |    |
|---|---|

Figura B.8: T5 - Salida obtenida

B.6. Prueba T6

En la imagen B.9 se muestran los datos de entrada para marcar asistencia de los estudiantes en una asignatura. La figura B.10 muestra la salida obtenida, correspondiente a un resumen de las asistencias por día.

Lista de Alumnos

Fecha

| N° | Apellido Paterno | Apellido Materno | Nombres | Asistencia |
|----|------------------|------------------|----------|---|
| 1 | Aguilar | Arriagada | Catalina | <input checked="" type="radio"/> Presente <input type="radio"/> Ausente |

Figura B.9: T6 - Datos de entrada

Resumen de Asistencias Primero Básico A

| Fecha | Presentes | Ausentes |
|------------|-----------|----------|
| 03-04-2020 | 1 | 0 |

Figura B.10: T6 - Salida obtenida

B.7. Prueba T7

La figura B.11 muestra la entrada para agregar una observación a un estudiante. Mientras que la imagen B.12, muestra la salida obtenida luego de registrar la observación.

Agregar Anotación

Asignatura
Lenguaje

Fecha
03 - 04 - 2020

Tipo
Positiva

Comentario
Alumna coopera con las actividades de la clase

Guardar Cerrar

Figura B.11: T7 - Datos de entrada

Observaciones

Aguilar Arriagada Catalina

+ Agregar

| Asignatura | Fecha | Tipo | Comentario |
|------------|------------|----------|--|
| Lenguaje | 2020-04-03 | Positiva | Alumna coopera con las actividades de la clase |

Figura B.12: T7 - Salida obtenida

B.8. Prueba T8

La figura B.13 presenta los datos de entrada para crear una actividad en una asignatura. Por otra parte, la imagen B.14 muestra la salida obtenida.

Agregar Actividad ×

Fecha
04 - 03 - 2020

N° de Horas
2

Objetivo, Contenido y Actividades
Presentación formal de la clase, se le da la bienvenida a los estudiantes y se realiza una lectura

Guardar Cancelar

Figura B.13: T8 - Datos de entrada

Actividades de Lenguaje

+ Actividad

Fecha: 04-03-2020 ✎ ✖

Cantidad de Horas: 2
Presentación formal de la clase, se le da la bienvenida a los estudiantes y se realiza una lectura

Figura B.14: T8 - Salida obtenida

B.9. Prueba T9

La imagen B.15 muestra la selección de un periodo académico para calcular el promedio de la asignatura. La figura B.16, presenta la salida obtenida para la prueba.

Seleccione el periodo para calcular promedios ×

Periodo

1

Aceptar Cancelar

Figura B.15: T9 - Datos de entrada

Promedios de Lenguaje

| N° | Apellido Paterno | Apellido Materno | Nombres | Promedio |
|----|------------------|------------------|----------|----------|
| 1 | Aguilar | Arriagada | Catalina | 6.0 |

Figura B.16: T9 - Salida obtenida

B.10. Prueba T10

La figura B.17 presenta como entrada la tercera observación negativa de un alumno. Mientras que la imagen B.18, muestra la salida (actualización de la lista de alumnos en observación).

Editar Anotación ×

Asignatura

Fecha

Tipo

Comentario

Figura B.17: T10 - Datos de entrada

Alumnos en Observación

| Curso | Alumno | Comentario | |
|------------------|----------------------------|---|---|
| Primero Básico A | Catalina Aguilar Arriagada | Se recomienda al profesor llamar al apoderado | <input type="button" value="✎"/> <input type="button" value="✖"/> |

Figura B.18: T10 - Salida obtenida

B.11. Prueba T11

La imagen B.19 muestra de entrada el cambio del mensaje de un alumno que está en observación. Por otra parte, la figura B.20 presenta la actualización del mensaje.

Editar Comentario ×

Nombre
Catalina

Apellido Paterno
Aguilar

Apellido Materno
Arriagada

Comentario
Apoderado asiste a reunión con profesor jefe del estudiante. Fecha 15-07-2020

Guardar Cerrar

Figura B.19: T11 - Datos de entrada

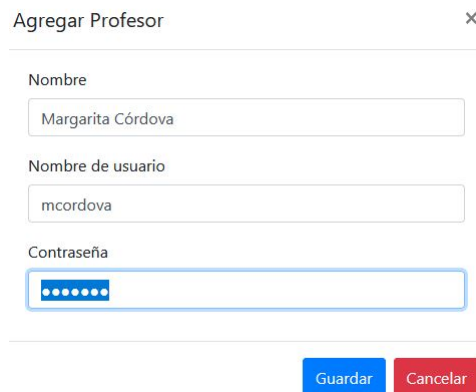
Alumnos en Observación

| Curso | Alumno | Comentario | |
|------------------|----------------------------|---|-------------------------------|
| Primero Básico A | Catalina Aguilar Arriagada | Apoderado asiste a reunión con profesor jefe del estudiante. Fecha 15-07-2020 | ✎ ✖ |

Figura B.20: T11 - Salida obtenida

B.12. Prueba T12

En la figura B.21 se muestra de entrada los datos para crear una cuenta de profesor. Mientras que la imagen B.22, presenta la cuenta ya creada.



Agregar Profesor ×

Nombre
Margarita Córdova

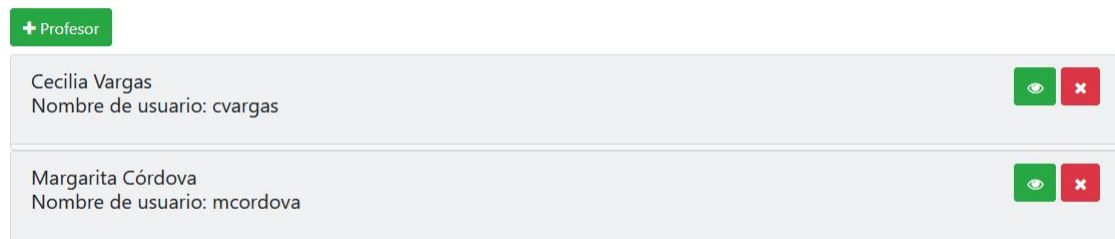
Nombre de usuario
mcordova

Contraseña
•••••

Guardar Cancelar

Figura B.21: T12 - Datos de entrada

Profesores







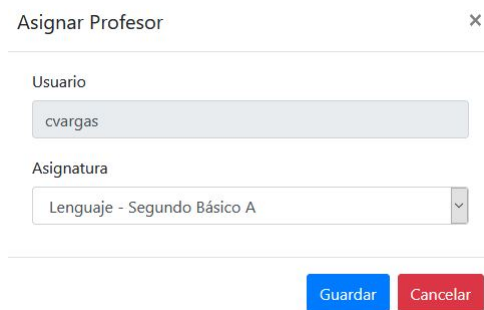
| + Profesor | | |
|--|---|---|
| Cecilia Vargas Nombre de usuario: cvargas |  |  |
| Margarita Córdova Nombre de usuario: mcordova |  |  |

Figura B.22: T12 - Salida obtenida

B.13. Prueba T13

En la imagen B.23 se presenta los datos de entrada para asignar a un profesor una asignatura. Mientras que la figura B.24, se muestra la lista actualizada de asignaturas que dicta el profesor.



Asignar Profesor

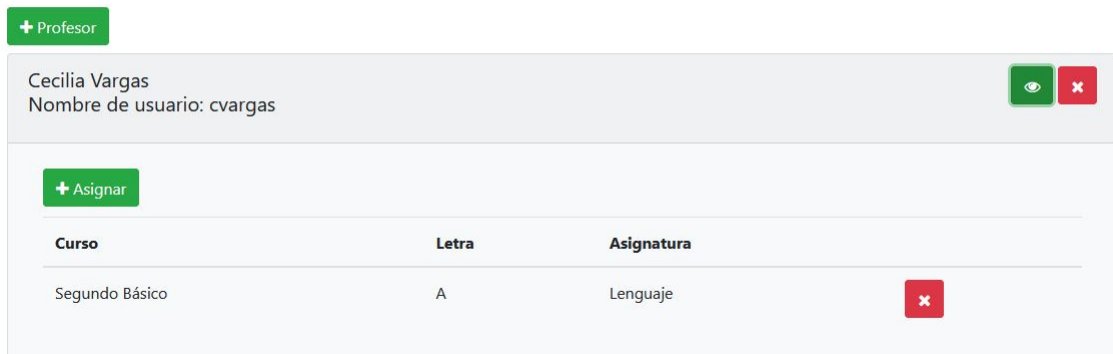
Usuario
cvargas

Asignatura
Lenguaje - Segundo Básico A

Guardar Cancelar

Figura B.23: T13 - Datos de entrada

Profesores



+ Profesor

Cecilia Vargas
Nombre de usuario: cvargas

+ Asignar

| Curso | Letra | Asignatura |
|----------------|-------|------------|
| Segundo Básico | A | Lenguaje |

Figura B.24: T13 - Salida obtenida

B.14. Prueba T14

En la figura B.25 se muestra que los datos de entrada son las credenciales de acceso para entrar a la plataforma. Mientras que en la imagen B.26, se presenta que el usuario acreditó sus datos e ingresó al sistema.



Figura B.25: T14 - Datos de entrada

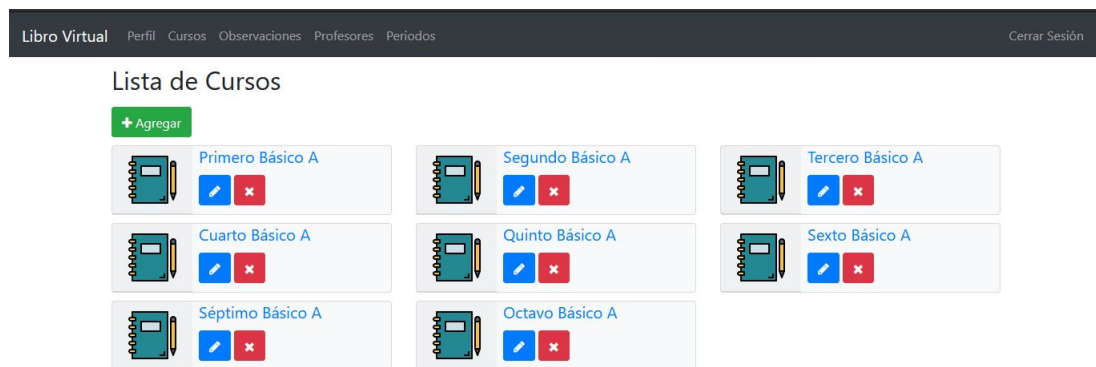


Figura B.26: T14 - Salida obtenida

C. Resultados prueba SUS

La figura 5.4 muestra los resultados de la encuesta de usabilidad aplicada a uno de los usuarios. A continuación, en la imagen C.1, se muestra la primera hoja aplicada de la encuesta.



Libro de Clases Virtual y envío de alumnos a observación.

Pruebas de Usabilidad (SUS)

Nicolás Eduardo Pradenas Vargas

17 de julio de 2020

Datos Personales

- Nombre: Rodrigo Cancino Brito
- Funciones que desempeña: Jefe Unidad Técnico Pedagógica
- Fecha: 18 de julio 2020

Instrucciones

- La encuesta consta de 10 preguntas, en las cuales no existe un límite de tiempo para responder.
- Comenzará haciendo uso del sistema propuesto mediante algún equipo que tenga disponible para su utilización (se recomienda un notebook o computador de escritorio).
- A medida que usa el sistema, procederá a responder las preguntas de la siguiente sección bajo el siguiente criterio.
 - 1 = Totalmente desacuerdo.
 - 2 = En desacuerdo.
 - 3 = Indeciso/a. Ni de acuerdo ni en desacuerdo.
 - 4 = De acuerdo.
 - 5 = Totalmente de acuerdo.
- Marque con una (X) la opción que desea expresar para cada pregunta.

Figura C.1: Prueba de usabilidad - Hoja 1

| Preguntas | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1. Creo que me gustaría usar este sistema con frecuencia | | | | | X |
| 2. Creo que el sistema es innecesariamente complejo | X | | | | |
| 3. Pensé que el sistema sería más fácil de usar | X | | | | |
| 4. Creo que necesito ayuda técnica (de un experto) para usar el sistema | X | | | | |
| 5. Las distintas funciones del sistema se complementan entre ellas | | | | | X |
| 6. Todas las cosas parecidas funcionan de la misma manera (no hay inconsistencias) | | | | | X |
| 7. Creo que el sistema es lo suficientemente simple como para que cualquiera lo use rápidamente | | | | | X |
| 8. El sistema me pareció muy engorroso de usar | X | | | | |
| 9. Puedo usar el sistema sin temor a equivocarme | | | | | X |
| 10. Necesitaba aprender muchas cosas antes de comenzar a usar el sistema | X | | | | |

Comentarios: Fue muy fácil utilizar el libro de clases. Pienso que es un gran herramienta tecnológica que simplifica y aporte al trabajo docente, pues permite ahorrar tiempo, ya que con un clic voy a la sección que necesito.

Cecilia Marozzi Del Carmen Vargas Vergara
 Profesora Colegio Particular Piaget
 San Javier
 Firma

Figura C.2: Prueba de usabilidad - Hoja 2 Cecilia Vargas

| Preguntas | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1. Creo que me gustaría usar este sistema con frecuencia | | | | | X |
| 2. Creo que el sistema es innecesariamente complejo | X | | | | |
| 3. Pensé que el sistema sería más fácil de usar | X | | | | |
| 4. Creo que necesito ayuda técnica (de un experto) para usar el sistema | X | | | | |
| 5. Las distintas funciones del sistema se complementan entre ellas | | | | | X |
| 6. Todas las cosas parecidas funcionan de la misma manera (no hay inconsistencias) | | | | | X |
| 7. Creo que el sistema es lo suficientemente simple como para que cualquiera lo use rápidamente | | | | | X |
| 8. El sistema me pareció muy engorroso de usar | X | | | | |
| 9. Puedo usar el sistema sin temor a equivocarme | | | | | X |
| 10. Necesitaba aprender muchas cosas antes de comenzar a usar el sistema | X | | | | |

Comentarios: Es un excelente sistema, fácil de usar que ayuda a simplificar mis labores administrativas en el establecimiento, ordenando en forma clara los datos necesarios de las asignaturas y cursos en los cuales me desempeño. **FELICITACIONES EXCELENTE TRABAJO.**


 Karla Riquelme Villalón
 Firma

Figura C.3: Prueba de usabilidad - Hoja 2 Karla Riquelme

| Preguntas | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1. Creo que me gustaría usar este sistema con frecuencia | | | | | X |
| 2. Creo que el sistema es innecesariamente complejo | X | | | | |
| 3. Pensé que el sistema sería más fácil de usar | X | | | | |
| 4. Creo que necesito ayuda técnica (de un experto) para usar el sistema | X | | | | |
| 5. Las distintas funciones del sistema se complementan entre ellas | | | | | X |
| 6. Todas las cosas parecidas funcionan de la misma manera (no hay inconsistencias) | | | | | X |
| 7. Creo que el sistema es lo suficientemente simple como para que cualquiera lo use rápidamente | | | | | X |
| 8. El sistema me pareció muy engorroso de usar | X | | | | |
| 9. Puedo usar el sistema sin temor a equivocarme | | | | | X |
| 10. Necesitaba aprender muchas cosas antes de comenzar a usar el sistema | X | | | | |

Comentarios:

El sistema de libro virtual es un sistema muy necesario en educación porque al tenerlo digitalizado permitiría como institución entregar un mensaje directo a la comunidad educativa sobre el cuidado de nuestro medio ambiente y la tarea de abordaje sobre la tecnología y sus usos aplicados a toda esfera; nos permitiría trabajar también desde casa; permitiría la corrección de errores sin tener el problema de los borrones no permitidos en el actual libro de clases. Además permitiría a la institución tener fácil acceso a datos de años anteriores, sin necesidad de desempolvar libros que están ocupando espacio.

Para las personas que todavía tienen dificultades con la tecnología, con una instrucción básica podrían trabajar el el sin dificultades.

Margarita Iris Córdova Carreño

Figura C.4: Prueba de usabilidad - Hoja 2 Margarita Córdova

| Preguntas | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1. Creo que me gustaría usar este sistema con frecuencia | | | | | x |
| 2. Creo que el sistema es innecesariamente complejo | x | | | | |
| 3. Pensé que el sistema sería más fácil de usar | | | | x | |
| 4. Creo que necesito ayuda técnica (de un experto) para usar el sistema | x | | | | |
| 5. Las distintas funciones del sistema se complementan entre ellas | | | | | x |
| 6. Todas las cosas parecidas funcionan de la misma manera (no hay inconsistencias) | | | | | x |
| 7. Creo que el sistema es lo suficientemente simple como para que cualquiera lo use rápidamente | | | | | x |
| 8. El sistema me pareció muy engorroso de usar | x | | | | |
| 9. Puedo usar el sistema sin temor a equivocarme | | | | | x |
| 10. Necesitaba aprender muchas cosas antes de comenzar a usar el sistema | | x | | | |

Comentarios: Este libro virtual me pareció muy útil, sencillo y fácil de usar. Todo está relacionado y me parece que es una forma de mantener todo el quehacer docente en forma ordenada. Felicidades al profesional que lo creó y espero poder usarlo en un futuro próximo.

Firma: María Felicinda Soto Chamorro

Figura C.5: Prueba de usabilidad - Hoja 2 Maria Soto

| Preguntas | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1. Creo que me gustaría usar este sistema con frecuencia | | | | | X |
| 2. Creo que el sistema es innecesariamente complejo | | | | | X |
| 3. Pensé que el sistema sería más fácil de usar | X | | | | |
| 4. Creo que necesito ayuda técnica (de un experto) para usar el sistema | X | | | | |
| 5. Las distintas funciones del sistema se complementan entre ellas | | | | X | |
| 6. Todas las cosas parecidas funcionan de la misma manera (no hay inconsistencias) | | | | | X |
| 7. Creo que el sistema es lo suficientemente simple como para que cualquiera lo use rápidamente | | | | | X |
| 8. El sistema me pareció muy engorroso de usar | X | | | | |
| 9. Puedo usar el sistema sin temor a equivocarme | | | | | X |
| 10. Necesitaba aprender muchas cosas antes de comenzar a usar el sistema | X | | | | |

Comentarios:

Valeska Fuentes Baltierra

Firma

Figura C.6: Prueba de usabilidad - Hoja 2 Valeska Fuentes