



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

**DESARROLLO DE UNA APLICACIÓN MÓVIL ANDROID PARA
EL CROSSFIT COYOTE FITNESS**

Autor

Luis Enrique Serrano Costales

Año

2020



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

DESARROLLO DE UNA APLICACIÓN MÓVIL ANDROID PARA
EL CROSSFIT COYOTE FITNESS

Trabajo de Titulación presentado en conformidad con los requisitos establecidos
para optar por el título de Ingeniero en Sistemas de Computación e Informática

Profesor guía

Msc. Nelly Patricia Acosta Vargas

Autor

Luis Enrique Serrano Costales

Año

2020

DECLARACIÓN PROFESOR GUÍA

Declaro haber dirigido el trabajo, Desarrollo de una aplicación móvil Android para el Crossfit Coyote Fitness, a través de reuniones periódicas con el estudiante Luis Enrique Serrano Costales, en el semestre 202010, orientado a sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los trabajos de Titulación.

A handwritten signature in blue ink, appearing to read 'Patricia Acosta Vargas', with a long horizontal flourish extending to the right.

Nelly Patricia Acosta Vargas

Magister En Docencia En Instituciones De Educación Superior

CI. 1802452944

DECLARACIÓN PROFESOR CORRECTOR

Declaro haber revisado este trabajo, Desarrollo de una aplicación móvil Android para el Crossfit Coyote Fitness, de Luis Enrique Serrano Costales, en el semestre 202010, dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación.



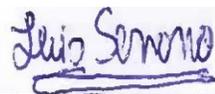
Paulo Roberto Guerra

Magister en Software y Sistemas

CI: 1002856050

DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes.



Luis Enrique Serrano Costales

CI. 1716985740

AGRADECIMIENTOS

Agradezco a Dios por las bendiciones que me ha otorgado. A mi madre Mónica por su apoyo incondicional y esfuerzo para poder cumplir esta meta; a mis amigos que han estado conmigo a lo largo de este camino. A mi tutora Patricia Acosta por compartir sus conocimientos en este proyecto de titulación.

DEDICATORIA

Dedico este trabajo de titulación a mi madre Mónica por su ejemplo de esfuerzo y dedicación que me motivaron a alcanzar mi meta. Mami te quiero muchísimo.

RESUMEN

Actualmente, existen varios centros deportivos, pero no todos cuentan con aplicaciones móviles que ayuden a los clientes a optimizar las actividades y el tiempo. Para dar solución a este problema se propone el desarrollo de una aplicación móvil en el sistema Android que ayude a los clientes y al centro deportivo a gestionar las actividades, el tiempo y los recursos, de tal forma que la aplicación cuente con parámetros de accesibilidad para los usuarios con baja visión, específicamente para solucionar los problemas de contraste. El centro deportivo Crossfit Coyote Fitness, imparte varios cursos para que el cliente pueda alternar las opciones de actividades y horarios, por lo cual la aplicación cuenta con un parámetro atractivo que permita personalizar las actividades, la asignación de instructores y los horarios. Adicionalmente, la aplicación ofrece un acceso fácil a una lista de cursos disponibles, con las actividades y los horarios, resultando interactivo para el usuario final. El proyecto se realizó en el centro deportivo Crossfit Coyote Fitness, gracias a la información otorgada por el administrador del lugar, y que será el encargado de administrar el sistema desde la parte web. Asimismo, se podrá realizar el registro del pago de los estudiantes de forma fácil desde el acceso a la aplicación por el celular. Durante todo el ciclo de desarrollo de este proyecto se aplicó el marco de trabajo Scrum que permite gestionar con flexibilidad, productividad y calidad los requerimientos del cliente.

ABSTRACT

Currently, there are several sport centers, but not all of them have mobile applications that help customers to optimize their activities and time. To solve this problem, we propose the development of a mobile application in Android system to help customers and sport center to manage activities, time and resources, at the same time, the application has accessibility parameters to low vision users, specially to solve contrast problems. Crossfit Coyote Fitness sport center, has several courses and the client can alternate the activities options and schedules, so the application has an attractive parameter that allows you to customize the activities and the instructors and schedule assignment. Additionally, the application offers easy access to a list of available courses, with activities and schedules, to make it interactive for the final user. The project was done at Crossfit Coyote Fitness sports center, thanks for the information provided by the administrator, who will use the system manage from the website. Likewise, the student's payment record can be made easily from the access to the application by the cell phone. Throughout the development cycle of this project, the Scrum framework was applied, allowing the client's requirements to be managed with flexibility, productivity and quality.

ÍNDICE

1. Capítulo I. Introducción	1
1.1. Antecedentes	1
1.2. Objetivos	2
1.2.1. Objetivo general	2
1.2.2. Objetivos específicos.....	3
1.3. Justificación.....	3
1.4. Alcance	3
1.5. Metodologías por utilizar	4
2. Capítulo II. Fundamentación teórica y metodológica	5
2.1. Marco teórico	5
2.1.1. Antecedentes investigativos	5
2.2. Marco conceptual	5
2.2.1. Metodología de desarrollo	5
2.2.2. Marco de trabajo SCRUM.....	7
2.2.3. Lenguajes de programación	9
2.2.4. FrontEnd Web	10
2.2.4.1. Framework Angular 6.....	10
2.2.4.2. Lenguaje Typescript y javascript	10
2.2.4.3. IDE Visual Studio Code.....	11
2.2.5. FrontEnd Móvil	11
2.2.5.1. Framework Flutter	11
2.2.5.2. Lenguaje Dart	12
2.2.5.3. IDE Android Studio.....	12
2.2.6. BackEnd	12
2.2.6.1. Framework Spring 5.....	13
2.2.6.1.1. SpringBoot.....	14

2.2.6.1.2. SpringData	14
2.2.6.1.3. SpringSecurity	14
2.2.6.2. JPA	15
2.2.6.3. Framework Hibernate 5.....	15
2.2.6.4. IDE IntelliJ IDEA.....	16
2.2.6.5. Lenguaje SQL	17
2.2.6.6. Gestor de base de datos MySQL	18
2.2.7. Diseño de arquitectura MVC.....	19
2.2.7.1. Spring MVC.....	20
2.2.7.2. MVC de Angular.....	21
2.2.8. Control de versiones GIT	22
2.2.9. Arquitectura Cliente – Servidor	23
2.2.10. Apache Tomcat embebido en SpringBoot.....	23
2.2.11. Accesibilidad.....	24
2.2.12. Test de accesibilidad App.....	25
3. Capítulo III. Implementación de la solución.....	25
3.1. Planificación de la iteración.....	25
3.1.1. Product Backlog	25
3.1.2. Diagrama de casos de uso	27
3.1.3. Diagrama de flujo de datos	30
3.1.4. Selección de requisitos.....	31
3.1.5. Planificación de iteración	48
3.1.6. Incremento	52
3.2. Inspección y adaptación	53
3.2.1. Sprint 0.....	53
3.2.1.1. Ejecución de la iteración	53
3.2.1.2. Revisión	54
3.2.1.2.1. Diagrama de entidad - relación.....	54
3.2.1.2.2. Diagrama de clases.....	55

3.2.1.3.	Retrospectiva	56
3.2.2.	Sprint 1	56
3.2.2.1.	Ejecución de la iteración	56
3.2.2.2.	Revisión	57
3.2.2.2.1.	Presentación – Pantalla de login	57
3.2.2.3.	Retrospectiva	58
3.2.3.	Sprint 2	58
3.2.3.1.	Ejecución de la iteración	58
3.2.3.2.	Revisión	59
3.2.3.2.1.	Presentación – Pantalla principal.....	59
3.2.3.2.2.	Presentación – Administración de usuarios	60
3.2.3.2.3.	Presentación – Agregar usuario	61
3.2.3.2.4.	Presentación – Editar usuario.....	61
3.2.3.2.5.	Presentación – Eliminar usuario	62
3.2.3.2.6.	Presentación – Validar el pago de los usuarios	63
3.2.3.2.7.	Presentación – Búsqueda de usuarios	63
3.2.3.3.	Retrospectiva	64
3.2.4.	Sprint 3	65
3.2.4.1.	Ejecución de la iteración	65
3.2.4.2.	Revisión	65
3.2.4.2.1.	Presentación – Administración de actividades	65
3.2.4.2.2.	Presentación – Agregar actividad	66
3.2.4.2.3.	Presentación – Editar actividad	66
3.2.4.2.4.	Presentación – Eliminar actividad.....	67
3.2.4.3.	Retrospectiva	67
3.2.5.	Sprint 4	68
3.2.5.1.	Ejecución de la iteración	68
3.2.5.2.	Revisión	69
3.2.5.2.1.	Presentación – Administración de curso	69
3.2.5.2.2.	Presentación – Agregar curso	70

3.2.5.2.3. Presentación – Editar curso.....	71
3.2.5.2.4. Presentación – Eliminar curso	73
3.2.5.3. Retrospectiva	73
3.2.6. Sprint 5.....	74
3.2.6.1. Ejecución de la iteración	74
3.2.6.2. Revisión.	75
3.2.6.2.1. Presentación – Login móvil.....	75
3.2.6.2.2. Presentación – Menú que visualiza el profesor.....	76
3.2.6.2.3. Presentación – Menú que visualiza el estudiante	76
3.2.6.2.4. Presentación – Pago de estudiantes	77
3.2.6.2.5. Presentación – Visualizar horario del profesor.....	78
3.2.6.3. Retrospectiva	80
3.2.7. Sprint 6.....	80
3.2.7.1. Ejecución de la iteración	80
3.2.7.2. Revisión	81
3.2.7.2.1. Presentación – Visualizar estado del pago	81
3.2.7.2.2. Presentación – Visualizar cursos disponibles para reserva	82
3.2.7.3. Retrospectiva	84
3.2.7.4. Retrospectiva – Prueba de accesibilidad.....	85
3.2.7.4.1. Etiqueta del elemento	85
3.2.7.4.2. Contraste del texto	85
3.2.7.4.3. Contraste del botón	86
3.2.7.4.4. Contraste del botón	87
3.3.2.7.5. Contraste del texto	87
4. Conclusiones y recomendaciones.....	89
4.1. Conclusiones	89
4.2. Recomendaciones	90
REFERENCIAS	91
ANEXOS	96

ÍNDICE DE TABLAS

Tabla 1: Características de una metodología de desarrollo de software	6
Tabla 2: Beneficios del marco de trabajo SCRUM	7
Tabla 3: Ciclo de desarrollo con el marco de trabajo Scrum.....	8
Tabla 4: Componentes de Scrum.....	9
Tabla 5: Diferencia entre typescript y javascript	10
Tabla 6: Ventajas y desventajas de Flutter.....	11
Tabla 7: Características de Android Studio	12
Tabla 8: Características de Spring	13
Tabla 9: Características de SpringBoot.....	14
Tabla 10: Ventajas de JPA.....	15
Tabla 11: Características de Hibernate	16
Tabla 12: Ventajas y desventajas de IntelliJ IDEA	17
Tabla 13: Características de SQL	17
Tabla 14: Características de MySQL.....	18
Tabla 15: Ventajas y desventajas de MySQL.....	18
Tabla 16: Ventajas de la arquitectura MVC	19
Tabla 17: Características de GIT.....	22
Tabla 18: Características de Apache Tomcat.....	24

ÍNDICE DE FIGURAS

Figura 1: Proceso de desarrollo de software	6
Figura 2: Ciclo de desarrollo Scrum	8
Figura 3: Modelo MVC	20
Figura 4: MVC de Spring.....	21
Figura 5: MVC de Angular.....	22
Figura 6: Modelo cliente-servidor del proyecto	23
Figura 7: Product Backlog de las historias de usuario	26
Figura 8: Diagrama de caso de uso - Administrador.....	27
Figura 9: Diagrama de caso de uso - Profesor	28
Figura 10: Diagrama de caso de uso – Estudiante.....	29
Figura 11: Diagrama de flujo de datos	30
Figura 12: Historia de usuario 1	32
Figura 13: Prototipo de pantalla web – Login	33
Figura 14: Prototipo de pantalla web–Pantalla principal	33
Figura 15: Historia de usuario 2	34
Figura 16: Prototipo de pantalla web–Administración de usuarios.....	35
Figura 17: Prototipo de pantalla web–Modal nuevo usuario	35
Figura 18: Prototipo de pantalla web–Modal editar usuario	36
Figura 19: Historia de usuario 3	36
Figura 20: Prototipo de pantalla web – Administración de actividades	37
Figura 21: Prototipo de pantalla web – Modal nueva actividad.....	37
Figura 22: Prototipo de pantalla web – Modal editar actividad.....	38
Figura 23: Historia de usuario 4	39
Figura 24: Prototipo de pantalla web–Administración de cursos	40
Figura 25: Prototipo de pantalla web – Modal nuevo curso	40
Figura 26: Prototipo de pantalla web – Modal editar curso.....	41
Figura 27: Prototipo de pantalla web – Modal visualizar horarios del curso.....	41
Figura 28: Prototipo de pantalla web – Modal visualizar actividades del curso.....	42

Figura 29: Prototipo de pantalla web – Modal visualizar asistentes del curso	42
Figura 30: Historia de usuario 5	43
Figura 31: Prototipo de pantalla móvil – Login	44
Figura 32: Prototipo de pantalla móvil – Menú principal – profesor	44
Figura 33: Prototipo de pantalla móvil – Pantalla pago mensualidad – profesor...	45
Figura 34: Prototipo de pantalla móvil – Visualizar cursos – profesor.....	45
Figura 35: Historia de usuario 6	46
Figura 36: Prototipo de pantalla móvil – Menú principal – estudiante	47
Figura 37: Prototipo de pantalla móvil – Visualizar mensualidad – estudiante.....	47
Figura 38: Prototipo de pantalla móvil – Reservar cursos – estudiante	48
Figura 39: Tareas para el Sprint 0.....	49
Figura 40: Tareas para el sprint 1	49
Figura 41: Tareas para el sprint 2	50
Figura 42: Tareas para el sprint 3	50
Figura 43: Tareas para el sprint 4	51
Figura 44: Tareas para el sprint 5	51
Figura 45: Tareas para el sprint 6	52
Figura 46: Lista de los sprints del proyecto	52
Figura 47: Tablero de control 9/08/2019.....	53
Figura 48: Tablero de control 8/08/2019.....	53
Figura 49: Diagrama de entidad - relación	54
Figura 50: Diagrama de clases	55
Figura 51: Tablero de control 15/08/2019.....	56
Figura 52: Tablero de control 22/08/2019.....	57
Figura 53: Presentación – Pantalla de login	57
Figura 54: Tablero de control 29/08/2019.....	59
Figura 55: Tablero de control 5/09/2019.....	59
Figura 56: Presentación – Pantalla principal – Administración de usuario.....	60
Figura 57: Presentación – Ingreso de usuario	61
Figura 58: Presentación – Edición de usuario	62

Figura 59: Presentación – Eliminar de usuario	62
Figura 60: Presentación – Validar el pago de los usuarios	63
Figura 61: Presentación – Búsqueda de usuarios	64
Figura 62: Tablero de control 12/09/2019.....	65
Figura 63: Presentación – Administración de actividades	66
Figura 64: Presentación – Ingreso de una actividad.....	66
Figura 65: Presentación – Edición de una actividad.....	67
Figura 66: Presentación – Eliminación de una actividad	67
Figura 67: Tablero de control 19/09/2019.....	68
Figura 68: Tablero de control 26/09/2019.....	69
Figura 69: Presentación – Administración de cursos.....	70
Figura 70: Presentación – Agregar curso	71
Figura 71: Presentación – Editar las actividades.....	72
Figura 72: Presentación – Editar horarios	72
Figura 73: Presentación – Visualizar los asistentes al curso	72
Figura 74: Presentación – Eliminación de un curso.....	73
Figura 75: Tablero de control 3/10/2019.....	74
Figura 76: Tablero de control 10/10/2019.....	74
Figura 77: Presentación – Login móvil	75
Figura 78: Presentación – Menú que el profesor puede visualizar	76
Figura 79: Presentación – Menú que el estudiante puede visualizar	76
Figura 80: Presentación – Pago de estudiantes	77
Figura 81: Presentación – Estudiante sin deuda	78
Figura 82: Presentación – Estudiante con deuda	78
Figura 83: Presentación – Visualización de cursos del profesor.....	79
Figura 84: Presentación – Descripción de los cursos	79
Figura 85: Presentación – Asistentes del curso.....	79
Figura 86: Tablero de control 17/10/2019.....	81
Figura 87: Tablero de control 24/10/2019.....	81
Figura 88: Presentación – Visualización del estado del pago del estudiante	82

Figura 89: Presentación – Visualización de los cursos para reservar	83
Figura 90: Presentación – Reserva de un curso.....	83
Figura 91: Presentación – Disponibilidad de un curso.....	84

1. Capítulo I. Introducción

1.1. Antecedentes

Crossfit es considerado en la actualidad como un método de entrenamiento intenso compuesto por una gran variedad de ejercicios físicos, cuyo objetivo principal es desarrollar las capacidades físicas de una persona (CrossFit SingularBox, 2019). *Coyote Fitness Crossfit* es una empresa que lleva tres años en el mercado, brindando un sistema de preparación y entrenamiento físico, hasta el momento cuenta con más de tres instructores y más de treinta clientes.

La actividad principal de la empresa es formar distintas clases con una serie de actividades, y ofrecer a los clientes en distintos horarios para facilitar la asistencia a las mismas. Además, las clases tienen un costo que el cliente debe cancelar mensualmente para poder asistir. Por el momento, toda esta información descrita, se registra de forma manual y la respectiva información se brinda verbalmente a los clientes, siendo esta actividad cada vez más complicada debido al incremento de clientes.

Por los argumentos anteriores es necesario implementar un sistema que permita realizar esta actividad, permitiéndoles trabajar de forma fácil y rápida, y lo más importante, mantener informado al cliente sobre los datos relevantes del curso.

Los parámetros que debe incluir el sistema dentro de este centro deportivo son los siguientes:

El administrador del centro deportivo requiere tener la capacidad de registrar tanto los clientes e instructores con los que cuenta, para poder brindar a los mismos la información adecuada.

El administrador del centro debe tener acceso para crear los cursos que se impartirá en el centro deportivo y que pueda definir tanto las diferentes actividades de los que estará compuesto cada curso, como los horarios en los que se impartirán los mismos.

El instructor del centro deportivo debe registrar los pagos que realicen los clientes y también pueda observar los asistentes que estarán en las clases impartidas por el mismo.

El cliente pueda visualizar los diferentes cursos con los horarios y actividades de tal forma que pueda reservar la clase que más le convenga, además es necesario que pueda visualizar la información de la fecha de pago y el estado en el que se encuentra.

Se requiere que las actividades realizadas por el administrador, se las realice vía web, mientras que los instructores y clientes, tendrán acceso a una aplicación móvil disponible en *Android*.

Con el propósito de administrar los requisitos planteados, se requiere contar con los perfiles y los roles de usuario que garantice la adecuada manipulación de módulos de la aplicación.

1.2. Objetivos

1.2.1. Objetivo general

Desarrollar un sistema web que permita registrar las actividades que se realizan en el centro deportivo *Coyote Fitness Crossfit*, y una aplicación móvil para la visualización, reserva y pago de los cursos.

1.2.2. Objetivos específicos

- Identificar las necesidades del centro deportivo *Coyote Fitness Crossfit* para establecer de una manera adecuada la funcionalidad del sistema.
- Diseñar el modelo de base de datos para el registro adecuado del sistema web y la aplicación móvil.
- Desarrollar el sistema web que automatizará la administración que se lleva a cabo en el centro deportivo.
- Desarrollar la aplicación móvil que facilitará la visualización, reserva y pago de los cursos.
- Probar la funcionalidad del sistema en el centro deportivo *Coyote Fitness Crossfit* para recibir su aprobación.
- Realizar un análisis de accesibilidad para generar recomendaciones para que la aplicación móvil sea más accesible e inclusiva para personas con discapacidad visual leve.

1.3. Justificación

La automatización de las actividades que se realizan en el centro deportivo *Coyote Fitness Crossfit*, mediante un sistema web, registra los clientes e instructores, además, administra los cursos con el detalle de las actividades y el horario en los que se impartirán las clases. Mediante la aplicación móvil se visualizarán el detalle de los cursos, las reservas y el pago realizado por el cliente.

1.4. Alcance

El sistema web permitirá al administrador el centro deportivo *Coyote Fitness Crossfit* registrar los cursos que se van a impartir con el detalle de sus actividades y el

horario en los que se impartirán, además podrá registrar los usuarios que tendrán acceso a la aplicación móvil, que básicamente serán los instructores y los clientes. Un instructor podrá visualizar los cursos que impartirá, con el horario y los asistentes a cada curso. Además, podrá registrar el pago que los clientes realizan mensualmente.

El cliente podrá visualizar los cursos creados por el administrador y el detalle de las actividades, además, del horario, de tal forma que pueda reservar el curso que más le convenga. Finalmente, podrá visualizar la fecha en que debe realizar los pagos, y el estado en el que se encuentra.

Se realizará un análisis de accesibilidad de la aplicación móvil, para personas con discapacidad visual leve. Para este análisis se tomará en cuenta factores como el tamaño de la tipografía, tamaño de los elementos y el contraste de la aplicación tomando en cuenta los estándares recomendados para estos factores. Una vez construida la aplicación se realizará este análisis se utilizará herramientas para dar recomendaciones para que la aplicación sea accesible e inclusiva.

1.5. Metodologías por utilizar

Para el desarrollo de un proyecto de software, es necesario estructurar, planificar y controlar el proceso de desarrollo, para lo cual se utiliza las metodologías de desarrollo (Molina Montero, Vite Cevallos, & Dávila Cuesta, 2018).

La metodología *SCRUM* permite trabajar en equipo, centrándose en proyectos en los que los requerimientos son cambiantes. Lo que le caracteriza es que se deben ejecutar iteraciones de corta duración, ya que las entregas deben ser el requisito con mayor prioridad para el cliente (Sifuentes & Macario, 2019).

Es por esta interacción de las personas durante toda la ejecución del proyecto, se ha seleccionado esta metodología de desarrollo, ya que toda la información se la realiza dentro del centro deportivo *Coyote Fitness Crossfit*.

2. Capítulo II. Fundamentación teórica y metodológica

2.1. Marco teórico

2.1.1. Antecedentes investigativos

Después de validar los requerimientos del cliente, y valorar su existencia dentro del centro deportivo *Coyote Fitness Crossfit*, se llega a la conclusión que no existe ningún aplicativo o sistema que cumpla con dichas peticiones por lo que se ve la necesidad de crearlo.

2.2. Marco conceptual

2.2.1. Metodología de desarrollo

Se puede describir a la metodología de desarrollo de software como una secuencia de pasos que se deben seguir rigurosamente para que los resultados de lo solicitado sean exactamente lo que se solicitó. En la Imagen 1 se puede visualizar la representación del proceso de desarrollo de software, en el cual se receipta requisitos nuevos o modificados y retorna un sistema nuevo o modificado. Para realizar este proceso se cuenta con personas que tienen distintos roles que cumplen ciertas actividades para que todo tenga un orden y una estructura (Universitat de Girona, 2013).



Figura 1: Proceso de desarrollo de software

Adaptado de (Universitat de Girona, 2013).

Tabla 1

Características de una metodología de desarrollo de software

Característica	Descripción
Reglas predefinidas	Que defina fases, técnicas, documentación estándar, tareas, herramientas, productos, ayuda al desarrollo.
Cobertura del ciclo de desarrollo	Pasos a seguir desde la planeación hasta el paso a mantenimiento de un sistema.
Verificación intermedia	Generar correcciones sobre el producto generado en cada fase.
Planificación	Para controlar los costos de desarrollo.
Comunicación	Facilita el trabajo de equipo entre los desarrolladores y el usuario final.
Flexible	Utiliza la misma metodología para todos los proyectos.
Capacitación	Los desarrollares deben conocer procesos y técnicas.
Herramientas CASE	Herramientas automatizadas para una mejor productividad y resultados.
Actividades que mejora procesos	Debe contar con herramientas que mejore la efectividad, calidad y costos de un proceso.
Soporte	Debe ser tomado en cuenta la facilidad para las modificaciones sobre los sistemas existentes.
Reutilización de software	Incluir procedimientos para crear, mantener y recuperar componentes que se puedan reutilizar.

Tomado de (Guevara, 2019).

2.2.2. Marco de trabajo SCRUM

SCRUM es un marco de trabajo de desarrollo de software ágil, cuyo objetivo es maximizar el retorno a la inversión de una empresa, basándose en construir principalmente la funcionalidad que el cliente considere de mayor valor. Por lo cual es importante el compromiso del cliente ya que será integrado en cada iteración de los procesos (Softeng, 2019).

Tabla 2

Beneficios del marco de trabajo SCRUM

Beneficio	Descripción
Expectativas cumplidas	Indica la prioridad de cada historia del proyecto.
Cambios flexibles	Adaptación a cambios requeridos por el cliente.
Reducción de tiempo	El funcional puede utilizar ciertas funcionalidades antes de que esté finalizado por completo.
Software de calidad	La calidad aumenta al obtener una versión después de cada iteración.
Productividad	Eliminación de burocracia.
Mayor retorno de inversión	Producción de un producto que priorice el retorno de la inversión (ROI)
Estimación de tiempo	Ayuda a determinar el tiempo de un sprint o historia de proyecto, por lo que es posible estimar cuándo se dispondrá una funcionalidad.
Disminución de riesgos	Los procesos de esta metodología ayudan a disminuir riesgos eficaz y anticipadamente.

Tomado de (Softeng, 2019).

Tabla 3

Ciclo de desarrollo con el marco de trabajo Scrum

Fase	Actividades
Concepto	<ul style="list-style-type: none"> • Definir características del producto • Asignar equipo de desarrollo
Especulación	<ul style="list-style-type: none"> • Revisar y desarrollar requisitos generales • Establecer límites y funcionalidades del desarrollo • Establecer costos • Definir el plan de entrega, midiendo el esfuerzo de cada proyecto
Exploración	<ul style="list-style-type: none"> • Ampliar el producto con las funcionalidades definidas
Revisión	<ul style="list-style-type: none"> • Revisar construcción del proyecto y comparar con el objetivo del mismo
Cierre	<ul style="list-style-type: none"> • Entregar una versión, en el que puede haber cambios “mantenimientos”

Tomado de (Trigas, 2019).

Las fases explicadas en la tabla 3 con sus actividades, se puede visualizar de forma gráfica en el Imagen 2.



Figura 2: Ciclo de desarrollo Scrum

Adaptado de (Trigas, 2019).

Tabla 4

Componentes de Scrum

Componente	Elementos	Descripción
Reuniones	1. Planificación <i>Backlog</i>	Define un documento donde se describe los requisitos dependiendo la prioridad del usuario
	2. Seguimiento	Se revisa el avance diario de las tareas de sprint
	3. Revisión del sprint	Presentación de los resultados finales y una versión
Roles	1. <i>ProductOwner</i>	Toma las decisiones ya que conoce el negocio del cliente. Escribe historias y sus prioridades.
	2. <i>ScrumMaster</i>	Comprueba que la metodología funcione, y ayuda a que el proceso fluya.
	3. Equipo de desarrollo	Organizan y toman decisiones para conseguir su objetivo. Estima tiempos.
	4. Usuarios	Funcional del producto
	5. <i>Stakeholder</i>	Participan durante las revisiones del sprint, ya que el proyecto les produce un beneficio
	6. Manager	Toma la decisión final involucrándose al momento de seleccionar objetivos y requisitos

Tomado de (Trigas, 2019).

2.2.3. Lenguajes de programación

Es un programa con un lenguaje que organiza algoritmos y procesos lógicos, que controla el comportamiento físico, lógico y la comunicación del usuario con el

computador. Se caracteriza porque contiene símbolos y reglas semánticas, que son instrucciones que construyen el código fuente de la aplicación (Raffino, 2018).

2.2.4. FrontEnd Web

2.2.4.1. Framework Angular 6

Angular sirve para escribir aplicaciones web en HTML y *Typescript*, y tiene diferentes librerías utilizadas para el correcto funcionamiento de la aplicación. Con Angular se puede generar *templates* controlados por la lógica del componente, y se puede iniciar la aplicación gracias al *bootstrapping* del *root* module (Molina, 2018).

2.2.4.2. Lenguaje Typescript y javascript

JavaScript es un lenguaje que permite crear páginas interactivas, del lado del cliente, por lo que se ejecuta desde un navegador sin un servidor, sin embargo, no está diseñado para aplicaciones complejas. *Typescript* por otra parte está desarrollado con *Javascript*, y se lo puede ejecutar con *Node.js* o navegadores que soporten *ECMAScript 3*. *Typescript* brinda un software más robusto y fácil de implementar, comparando con *Javascript* (GURU99, 2019).

Tabla 5

Diferencia entre typescript y javascript

Typescript	Javascript
Lenguaje orientado a objetos.	Lenguaje de secuencia de comandos.
Tiene una función llamada escritura estática.	No tiene esta función.
Tiene soporte para módulos.	No admite módulos.
Tiene interfaz.	No tiene interfaz.

Admite función de parámetro opcional. No admite función de parámetro opcional.

Tomado de (GEEKS FOR GEEKS, 2019).

2.2.4.3. IDE Visual Studio Code

Es un editor de programación multiplataforma, basada en el *framework* para aplicaciones gráficas de escritorio *Electron*, bajo licencia MIT, pero los ejecutables son de licencia gratuita (SINTES, 2019).

2.2.5. FrontEnd Móvil

2.2.5.1. Framework Flutter

Google presentó un *framework* diseñado para desarrollar aplicaciones con lenguaje *Dart*, cuyo objetivo principal es tener un código único para *Android* e *IOS*. Cuenta con todo lo necesario para usar animaciones, gestos de pantalla (acceso a servicios del teléfono como Bluetooth, GPS, etc.), y tiene una lista de *widgets* (POZAS, 2017).

Tabla 6

Ventajas y desventajas de Flutter

Ventajas	Desventajas
Los cambios se reflejan inmediatamente.	Requiere el lenguaje <i>Dart</i> necesariamente.
Rápida y constante renderización de vistas.	Es un <i>framework</i> con una comunidad pequeña.
Multiplataforma para <i>Android</i> y <i>IOS</i> .	Solo se enfoca a móvil.
Algunos gestos de pantalla requieren acceso a funciones nativas.	Tiene librerías limitadas.

Tomado de (OPENINNOVA, 2019).

2.2.5.2. Lenguaje Dart

Google creó *Dart* con la idea de ser soportado por navegadores, y aunque al inicio, ningún navegador lo soportaba, fueron mejorando hasta convertirse un mejor compilador fuertemente tipeado, se basa en clases y trata las funciones como objetos (LÓPEZ, 2018).

2.2.5.3. IDE Android Studio

Nombrado por Google como el IDE oficial para desarrollar aplicaciones para *Android* (DEVELOPERS ANDROID STUDIO, 2019).

Tabla 7

Características de Android Studio

Característica	Descripción
Flexible	Compilación flexible basada en <i>Gradle</i> .
Emulador rápido	Rápido y con gran cantidad de funciones.
Cambios en caliente	Inserta cambios sin reiniciar la aplicación.
Integración con <i>GitHub</i>	Ayuda a compilar funciones comunes.
Marcos de trabajo	Variedad con herramientas de prueba.
Herramientas de <i>Lint</i>	Identifica problemas de rendimiento, usabilidad y compatibilidad.
Lenguajes compatibles	C++ y NDK
Integración con Google <i>CloudPlatform</i>	Integra Google <i>Cloud Messaging</i> y <i>App engine</i> .

Tomado de (DEVELOPERS ANDROID STUDIO, 2019).

2.2.6. BackEnd

2.2.6.1. Framework Spring 5

Spring surgió con la idea de inyección de dependencias o el uso de objetos convencionales (*POJOs*), cuando los servidores de aplicaciones consumían muchos recursos y los EJB eran pesados, inflexibles y complejos. Esto lograría desarrollar rápida y sencillamente aplicaciones ligeras (UNIVERSIDAD DE ALICANTE, 2014).

Tabla 8

Características de Spring

Característica	Descripción
Tecnologías	Soporta inyección de dependencias, recursos, enlace de datos, eventos, validación, i18n, conversión de tipo.
Acceso a datos	Soporte ORM, DAO, JDBC.
Gestión de transacciones	
Integración	JMS, correo electrónico, Comunicación remota, JCA, tareas, JMX, programación, caché.
Pruebas	<i>Spring MVC</i> prueba, simulacro de objetos, <i>WebTestClient</i> , <i>TestContext</i> .
Programación orientada a aspectos (AOP)	Implementa rutinas transversales.
MVC e Inyección de datos	Facilita su programación.
Seguridad	Tiene mucho enfoque en la seguridad.
<i>Frameworks web</i>	Compatible con <i>Spring MVC</i> y <i>SpringWebFlux</i> .
Procesamiento de datos por lotes	
Administración remota	Configuración de la visibilidad y gestión de objetos Java vía JMX.
Framework liviano	Por su implementación POJO.

Tomado de (EQUIPO GEEK, 2018).

2.2.6.1.1. SpringBoot

Para simplificar el concepto de *SpringBoot* se obtiene las siguientes características:

Tabla 9

Características de SpringBoot

Característica	Descripción
Incorporación de aplicación de servidores	Se elimina la necesidad de incluir un archivo <i>war</i> en el servidor para probar la aplicación.
Configuración <i>Maven</i>	Simplifica mediante los POM que significa Project <i>ObjectModels</i> .
Configuración automática	Spring lo configura automáticamente en la medida posible.
Métricas	Cuenta con servicios que controla el estado de la aplicación, de la memoria, de los <i>beans</i> , etc.

Tomado de (IONOS, 2019).

2.2.6.1.2. SpringData

El objetivo principal es facilitar el acceso a las tecnologías de persistencia, ofreciendo las funcionalidades de CRUD (*create – read – update – delete*) en los DAOs (*Data Access Objects*) con interfaces genéricas. Hasta el momento *Spring Data* ofrece soporte para JPA y JDBC, *Apache Hadoop*, *GemFire*, *Redis*, *MongoDB*, *Neo4j*, *HBase* (SÁNCHEZ, 2012).

2.2.6.1.3. SpringSecurity

Este *framework* permite gestionar todo lo referente a la seguridad de la aplicación, toman en cuenta los protocolos de seguridad hasta los roles que cada usuario

tendrá para el acceso a la información de la aplicación. Lo sencillo de Spring Security es que todo se configura mediante un fichero (CLEVENTY, s.f.).

2.2.6.2. JPA

Permite interactuar con la base de datos mediante objetos, al implementar un Framework *Object Relational Mapping* (ORM). JPA es más un documento con reglas que se deben cumplir para desarrollar una implementación de JPA. Algunas de las implementaciones de JPA que más se utilizan son *Hibernate*, *ScilpseLink* y *TopLink* (BLANCARTE, <https://www.oscarblancarteblog.com>, 2019).

Tabla 10

Ventajas de JPA

Ventajas

Uso de ORM para simplificar la lógica de la persistencia

El rendimiento de la aplicación depende del conocimiento de la tecnología

Cuenta con parámetros *tunning* que mejora el rendimiento

JPA se puede utilizar fuera del contenedor EJB

Tomado de (JUNTA DE ANDALUCÍA, s.f.).

2.2.6.3. Framework Hibernate 5

Es una herramienta para Java que facilita el mapeo de atributos mediante archivos XML o anotaciones en los *beans* de las entidades (GUERRERO, 2015).

Tabla 11

Características de Hibernate

Característica	Descripción
Programación natural	Diseña objetos persistentes (incluye polimorfismo, relaciones, colecciones y tipos de datos)
Escalabilidad	Tiene caché de doble capa y puede ser utilizado en <i>cluster</i>
Contextos de persistencia de larga vida	Tiene bloqueos automáticos
Gratuito	
EJB 3.0	Gestiona API de persistencia en Java y mapeado objeto-relación
Persistencia transparente	Soporte en colecciones de Java, estilo de persistencia JavaBeans que abstraen al usuario
Mapeo flexible	Asociaciones bidireccionales, persistencia transitiva, colecciones de tipos básicos
Facilidad en consultas	Orientado a objetos
Facilidad en metadatos	Soporta mapeado de XML y basado en anotaciones

Tomado de (JUNTA DE ANDALUCÍA, s.f.).

2.2.6.4. IDE IntelliJ IDEA

Gracias a la nueva programación con micro-servicios, se puede trabajar con IntelliJ IDEA ya que no requiere levantar un servidor de aplicación porque se puede usar servidores embebidos. A pesar de que IntelliJ IDEA no es *open source* se puede trabajar con la *community* versión (HIDALGO, s.f.).

Tabla 12

Ventajas y desventajas de IntelliJ IDEA

Ventajas	Desventajas
Autocompletado de código	Es de pago
Integración con GIT	
Tiene los <i>pluggins</i> necesarios	
Fiabilidad y robustez	
Refactorización inteligente	

Tomado de (RODRIGUEZ, 2012).

2.2.6.5. Lenguaje SQL

Es un lenguaje de acceso a base de datos utilizado en la mayoría de los gestores de base de datos, que permite hacer operaciones como recuperar información y hacer cambios en una base de datos (GARCÍA M. , 2012).

Tabla 13

Características de SQL

Característica	Opciones
DDL (Lenguaje de definición de datos)	<i>Create, Alter, Drop</i>
DML (Lenguaje de manipulación de datos)	<i>Insert, Update, Delete</i>
DCL (Lenguaje de control de datos)	<i>Grant, Revoke</i>
Control de transacción	<i>Commit, Rollback, Savepoint</i>
Restricción de integridad	Referencial, datos

Tomado de (EXES, s.f.).

2.2.6.6. Gestor de base de datos MySQL

Es un sistema de código abierto, sirve para gestionar bases de datos, y se basa en el lenguaje estructurado SQL. Se asocia con aplicaciones basadas en la web y publicaciones en línea, y se ejecuta en todas las plataformas (ROUSE, 2015).

Tabla 14

Características de MySQL

Característica	Descripción
Tareas multiprocesador	Ya que tiene la opción de trabajo multihilo
Cantidad de datos	Puede ingresar una gran cantidad por columna de trabajo
Dispone de APIs	Para los principales lenguajes de programación
Portabilidad	
Capacidad de índices	Soporta hasta 32 índices de tablas diferentes
Seguridad	Soporta usuarios con claves individuales

Tomado de (CULTURACION, s.f.).

Tabla 15

Ventajas y desventajas de MySQL

Ventajas	Desventajas
Gratuito	Por ser gratuito, no tiene soluciones oficiales.
Tiene licencia DPL	La mayoría de la utilización de MySQL no tiene documentación.
No requiere hardware o software de alto rendimiento	Requiere monitorear las aplicaciones en busca de falencias.
Rapidez y buen rendimiento	No es muy intuitivo.

Instalación y configuración sencilla	No es muy eficaz cuando se necesita una constante modificación de escritura.
Soporte casi al 100%	
Baja probabilidad de daño en datos	
Seguridad y encriptación	

Tomado de (HOSTINGPEDIA, 2019).

2.2.7. Diseño de arquitectura MVC

Es un patrón arquitectónico que intenta dar una solución a la arquitectura de software, y consta de 3 partes: Modelo (funcionalidad y datos básicos) – Vista (información que ve el usuario) – Controlador (maneja la entrada del usuario). El objetivo es la creación y reutilización de componentes (HUAMAN, 2018).

Tabla 16

Ventajas de la arquitectura MVC

Ventaja	Descripción
Implementación de componentes	Existe separación entre los componentes por lo que se los puede implementar por separado
API bien definido	El Modelo, Vista y Controlador pueden ser reemplazados fácilmente
Conexión dinámica entre Modelo y Vista	Se produce cuando la aplicación se está ejecutando, no cuando se la compila.

Tomado de (CATALANI, 2007).



Figura 3: Modelo MVC

Adaptado de (UNIVERSIDAD DE ALICANTE, s.f.).

En la Imagen 3 se demuestra de forma gráfica como se trabaja con el modelo MVC, como se puede observar, desde el navegador se envía una solicitud http al controlador, el mismo envía los parámetros de ejecución al modelo donde se los procesa y retorna la matriz de datos resultantes al controlador, el mismo envía a la vista para procesarlo la parte visual y presentar en el navegador.

2.2.7.1. Spring MVC

Es un *framework* que colabora con la elaboración de páginas web sencillas, se apoya en el concepto de que un único *servlet* recibe las peticiones (GET, POST, etc.), la ejecuta y devuelve un resultado. Por lo tanto, sus 3 partes son: Modelo (encapsula datos de la aplicación) – Vista (representa los datos del modelo) – Controlador (procesa las solicitudes de usuario y crea un modelo para pasarlo a la vista) (CAULES, 2013).

En la Imagen 4 se visualiza las solicitudes y las respuestas de cada parte descrita en el modelo.

Proporciona componentes que se los pueden usar en un desarrollo de aplicaciones web flexibles. Este patrón separa los diferentes aspectos de la aplicación, que son la lógica de entrada, la lógica de negocios y la lógica de UI, y los acopla (TUTORIALSPPOINT, 2019).

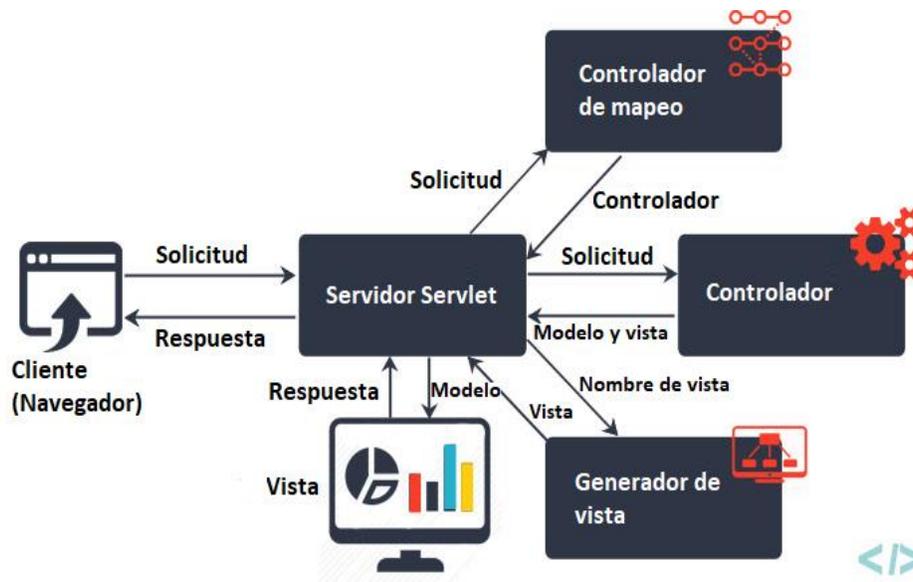


Figura 4: MVC de Spring

Adaptado de (CODENUCLEAR, 2017).

2.2.7.2. MVC de Angular

Es un patrón utilizado por Angular con la finalidad de añadir o modificar fácilmente algún aspecto de la aplicación. Sus 3 partes son: Modelo (estructura de datos) – Vista (interfaz del usuario) – Controlador (intermediario entre el modelo y la vista) (DAVID, 2017). En la Imagen 5 se visualiza como se representa cada una de las partes descritas anteriormente.

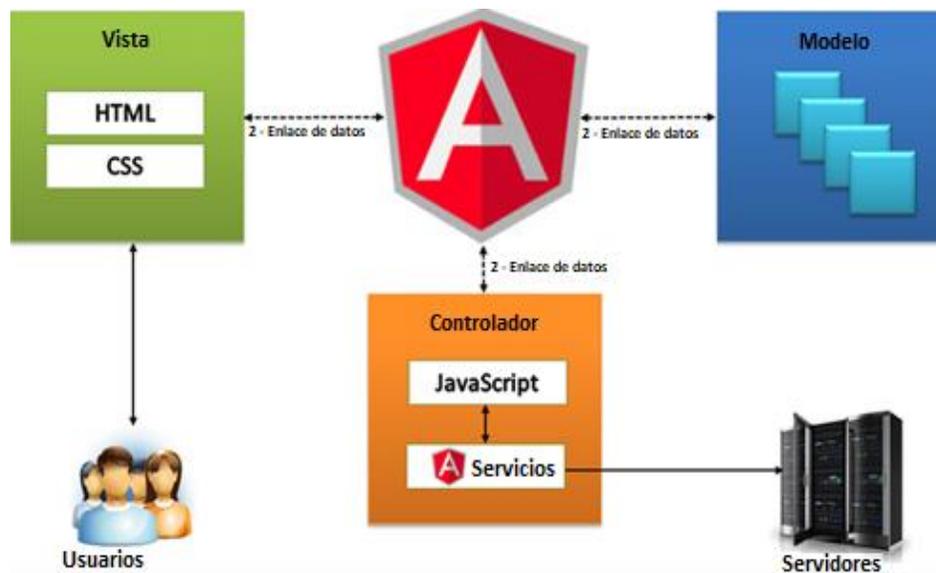


Figura 5: MVC de Angular
Adaptado de (FATIMA, 2018).

2.2.8. Control de versiones GIT

GIT es una herramienta que facilita el control del registro de uno o más archivos de un proyecto, tiene la facilidad de visualizar las versiones y puede verificar el usuario que ha ido realizando los cambios que alimentan el proyecto, esta tarea suele dificultarse si se lo realiza de forma manual, y por lo general cuando el proyecto tiene un gran número de archivos (FDLWIKI ELP, 2017).

Tabla 17

Características de GIT

Característica	Descripción
No lineal	Gestiona ramas rápidamente
Gestión distribuida	Se puede fusionar las ramas local y remotamente
Gestión eficiente	Cuando los proyectos son grandes
Re almacenamiento	Se almacena periódicamente en paquetes

Tomado de (ISMA, 2015).

2.2.9. Arquitectura Cliente – Servidor

El objetivo principal del modelo cliente – servidor es distribuir la información entre los proveedores de recursos o servicios (servidores) y los demandantes (clientes), por lo que los usuarios finales pueden estar dispersos y acceder a recursos comunes (RUIZ, 2013). A continuación, en la Imagen 6 se describe lo descrito para un mejor entendimiento.

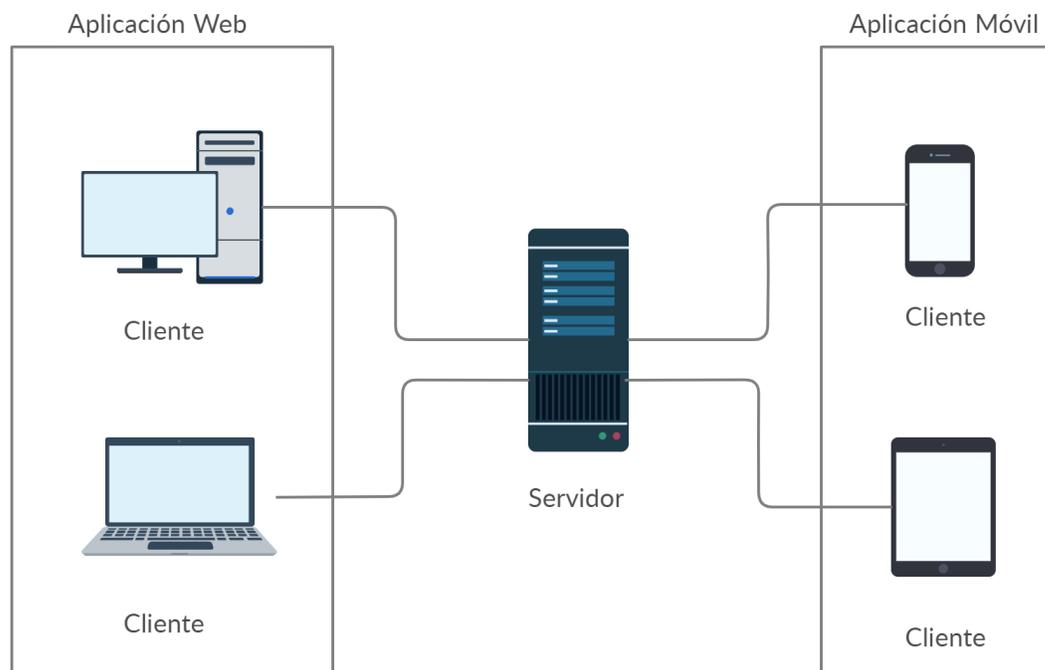


Figura 6: Modelo cliente-servidor del proyecto

2.2.10. Apache Tomcat embebido en SpringBoot

El Apache Tomcat es un servidor web para *servlets* y *JSPs* (se convierte en *servlets* mediante el compilador Jasper que ya viene incluido en Tomcat), desarrollado en Java, por lo que funciona en cualquier sistema operativo (DÍAZ, s.f.).

Tabla 18

Características de Apache Tomcat

Característica	Descripción
Asíncrono	Tiene soporte asíncrono.
Configuración dinámica	Librerías pueden embeber parte de un web.xml
Anotaciones	Soporta anotaciones sin necesidad de crear un web.xml.
API <i>Servlet</i> extendida	Se puede añadir <i>servlets</i> y <i>filters</i> cuando la aplicación está corriendo.
Embebido simplificado	Para esto incluye un API.
Mejoras en <i>Loggin</i>	Mediante un formateador que permite escribir logs en una línea.
Alias	Se puede incluir ficheros externos para poder compartirlos en todas las aplicaciones.
<i>MemoryLeaks</i>	Cuenta con detección y prevención de <i>MemoryLeaks</i> .
Seguridad	Tienen mejoras en la seguridad.

Tomado de (GARCÍA L. , 2010).

Al ver la necesidad de aprovechar el potencial de las anotaciones en *Spring*, y evitar configuraciones tediosas, se levantó un contexto nuevo de *SpringBoot*, en el que se aprovecha para tener al *Tomcat* embebido (ROMERO, 2017).

2.2.11. Accesibilidad

Se refiere a la forma en que las personas puedan interactuar, navegar, percibir y entender la aplicación móvil. Una condición de accesibilidad, es el cumplimiento de no discriminar a las personas, maximizar los usuarios potenciales, garantizar la equivalencia de los contenidos entre los diferentes dispositivos y mejorar la indexación en los motores de búsqueda (VARELA, y otros, s.f.).

2.2.12. Test de accesibilidad App

Es una herramienta que escanea aplicaciones para Android en tiempo real y permite generar recomendaciones para que las aplicaciones sean más accesibles. Permite realizar pruebas de corrección de color, ampliación, puntero de mouse, inversión de color, texto de alto contraste y efectos visuales avanzados generando recomendaciones en cada una de las pruebas para que las aplicaciones sean más accesibles (Accessibility Scanner - Android Accessibility Help, 2019).

3. Capítulo III. Implementación de la solución

3.1. Planificación de la iteración

3.1.1. Product Backlog

No	Historia	Tarea	Estimación	Horas
1	Login del sistema web	Servicios web para login	2	8
		Pantalla del login	3	24
		Pruebas y correcciones	3	24
2	Administración de usuarios	Servicios web para la administración de usuarios	2	16
		Pantalla principal	3	24
		Búsqueda de usuarios	2	16
		Agregar usuario nuevo	2	16
		Editar y eliminar usuario	1	8
		Pruebas y correcciones	3	24
3	Administración de actividades	Servicios web para la administración de actividades	2	16

		Búsqueda de actividades	1	8
		Agregar actividad nueva	1	8
		Editar y eliminar actividad	1	8
		Pruebas y correcciones	3	24
4	Administración de cursos	Servicios web para la administración de cursos	5	40
		Búsqueda de cursos	1	8
		Agregar curso nuevo	3	24
		Editar y eliminar curso	2	16
		Visualizar horarios	1	8
		Visualizar actividades	1	8
		Visualizar asistentes del curso	2	16
		Pruebas y correcciones	3	24
5	Función del rol de Profesor	Servicios web para realizar las funciones del profesor desde el móvil	3	24
		Login del móvil	2	16
		Pantalla de menú dependiendo del rol	2	16
		Pago de estudiantes	2	16
		Visualizar horarios por curso del profesor	2	16
		Pruebas y correcciones	3	24
6	Función del rol Estudiante	Servicios web para las funciones del estudiante desde el móvil	2	16
		Visualización del pago	2	16
		Visualización y reserva de cursos	5	40
		Pruebas y correcciones	3	24

Figura 7: Product Backlog de las historias de usuario

Cada historia de usuario tiene una lista de tareas que se van a realizar para cumplir con los requerimientos del usuario, la estimación se ha establecido de acuerdo a las horas que se necesitan para realizar la tarea.

3.1.2. Diagrama de casos de uso

Para el sistema encontramos 3 actores o usuarios principales que serán los usuarios directos del sistema.

Administrador

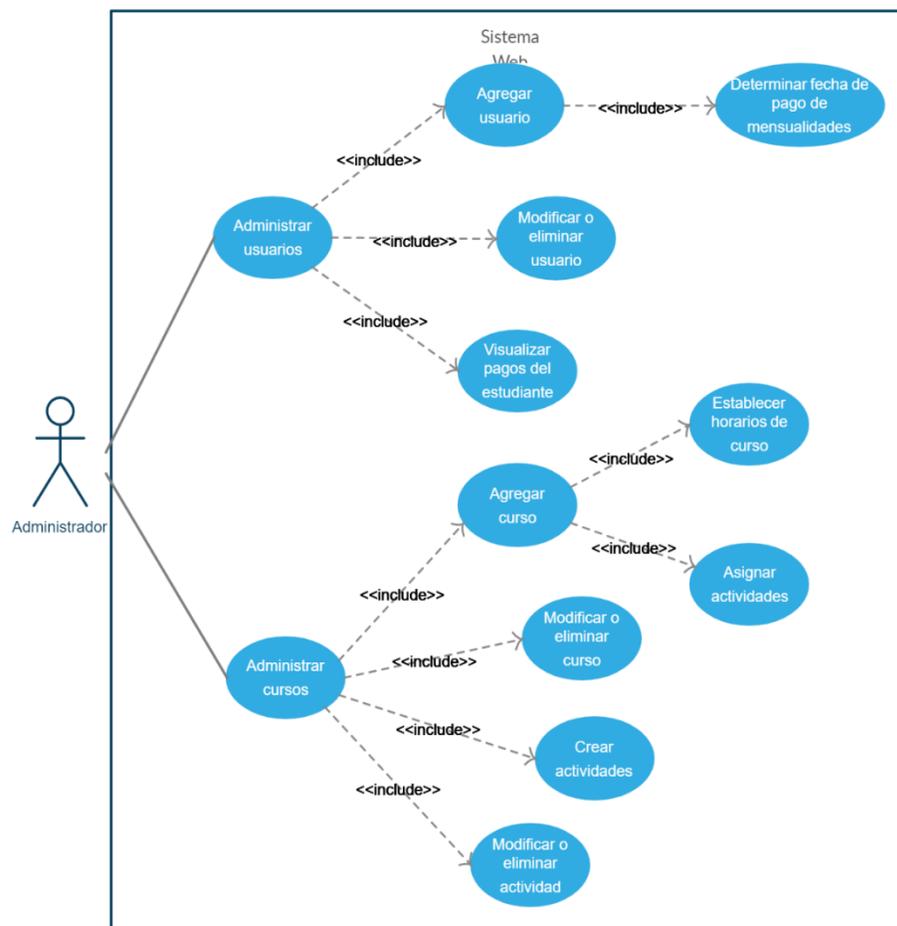


Figura 8: Diagrama de caso de uso - Administrador

Profesor

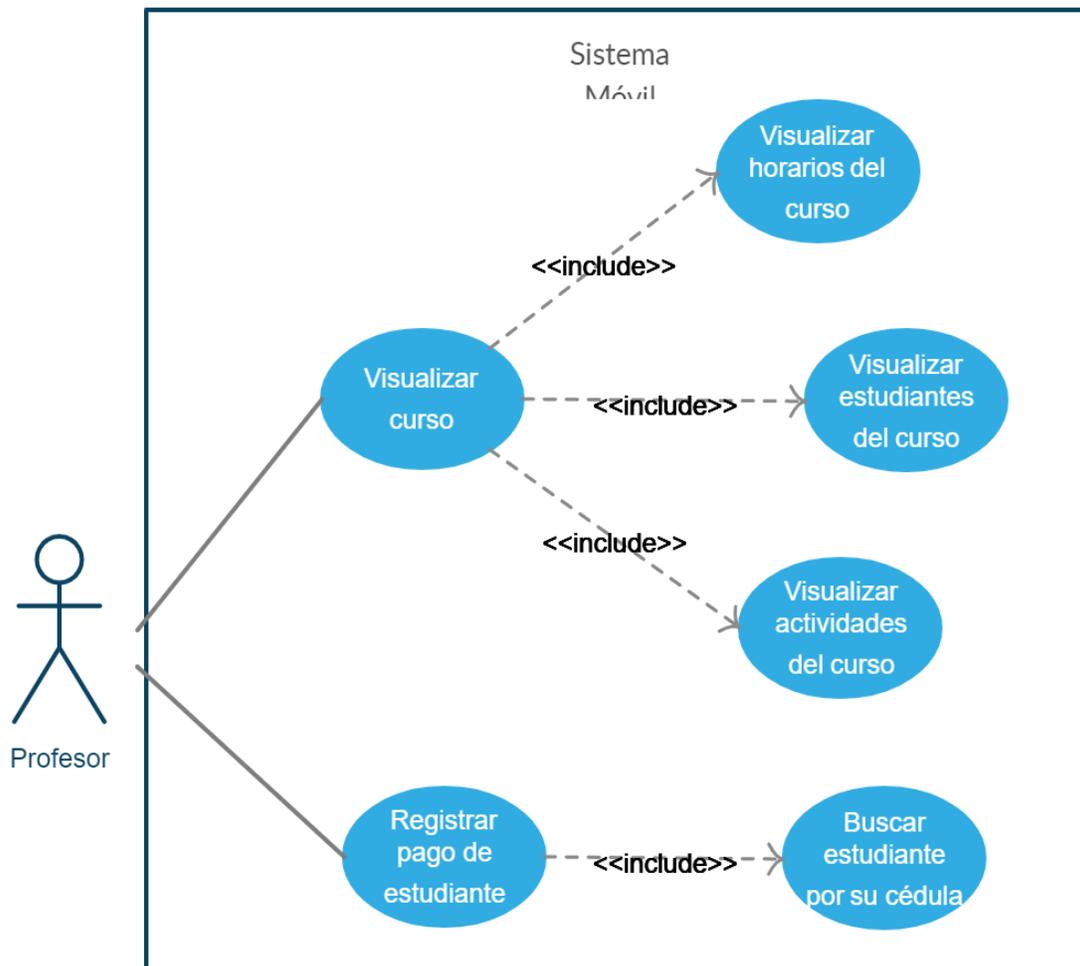


Figura 9: Diagrama de caso de uso - Profesor

Estudiante

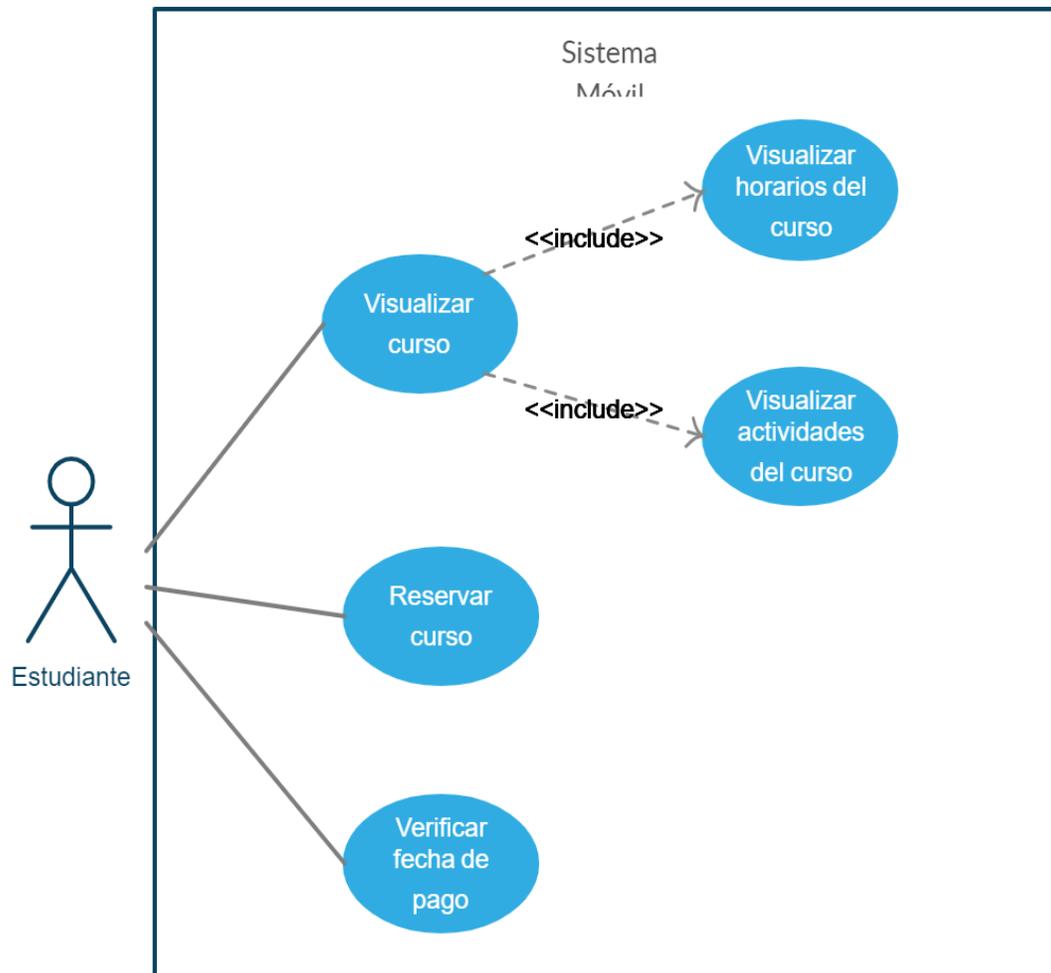


Figura 10: Diagrama de caso de uso – Estudiante

3.1.3. Diagrama de flujo de datos

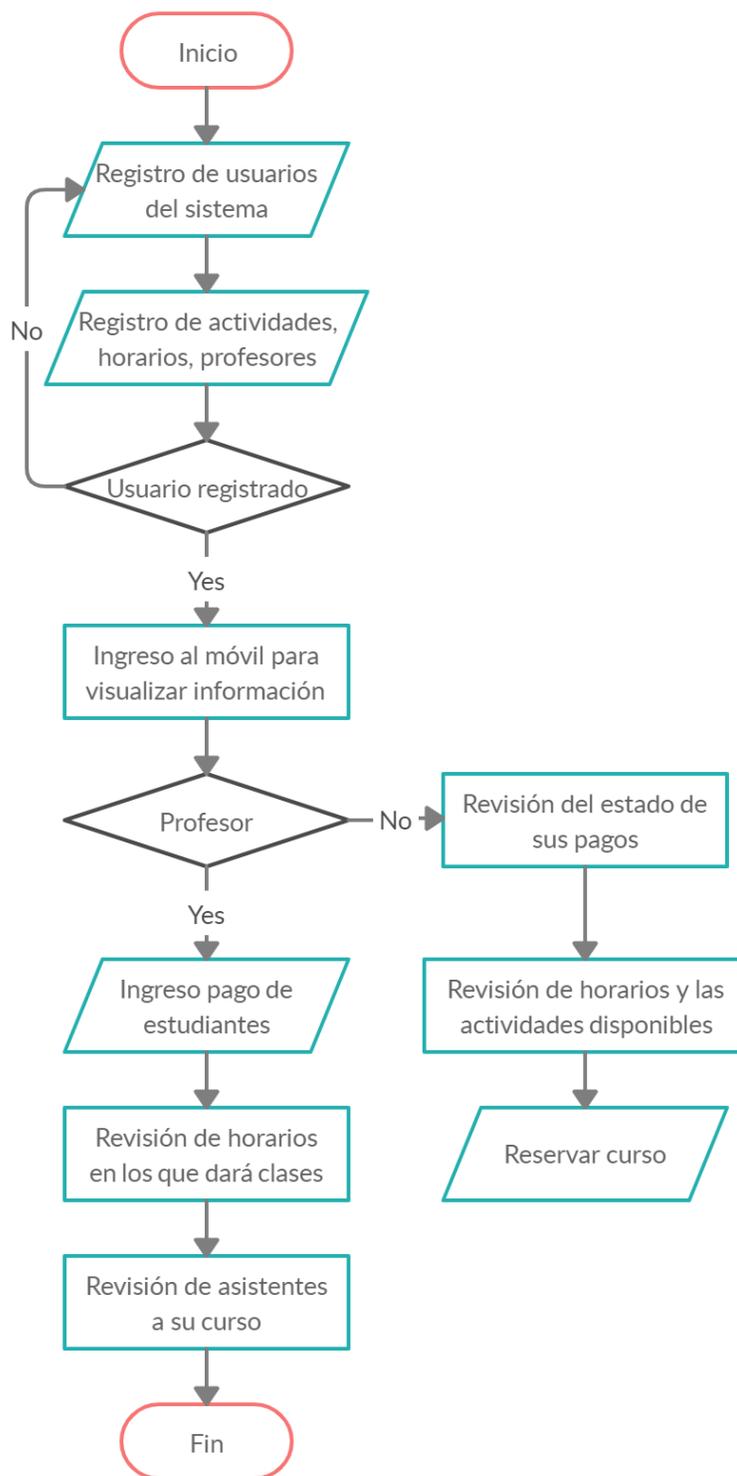


Figura 11: Diagrama de flujo de datos

3.1.4. Selección de requisitos

La recepción de requerimientos del usuario se realizó mediante reuniones con el usuario funcional. El acta de la reunión se encuentra en el Anexo 1.

Cada historia de usuario está conformada con la siguiente información:

1. *Número*: número de la historia
2. *Usuario*: persona usuaria del requerimiento descrito en la historia de usuario
3. *Nombre de historia*: nombre como se le va a identificar a la historia
4. *Prioridad en negocio*: prioridad del requerimiento dentro del negocio.
 - Alta: es un requerimiento muy importante dentro del negocio.
 - Media: es un requerimiento con importancia media dentro del negocio.
 - Baja: es un requerimiento con poca importancia dentro del negocio.
5. *Riesgo de desarrollo*: posible riesgo en el proyecto dentro del desarrollo.
 - Alta: riesgo alto de desarrollo dentro del proyecto.
 - Media: riesgo medio de desarrollo dentro del proyecto.
 - Baja: bajo riesgo de desarrollo dentro del proyecto.
6. *Puntos estimados*: puntos que indican la importancia de la historia de usuario, de acuerdo con las necesidades del cliente, donde 5 es muy importante y 1 menos importante. La estimación se la realiza con la técnica de planning poker, técnica usada para que todos los integrantes del equipo puedan dar su estimación sin dejarse influencias por las otras estimaciones (Kniberg, 2007).
7. *Iteración asignada*: el número de sprint o iteración en la que se va a presentar.
8. *Programador responsable*: persona responsable de desarrollar la historia de usuario.
9. *Descripción*: descripción detallada de lo que se trata la historia de usuario.
10. *Criterios de aceptación*: criterios que se van a tomar en cuenta para validar el correcto funcionamiento de la historia de usuario.

Historia de usuario			
Número:	1	Usuario:	Usuario del sistema
Nombre de historia:	Login del sistema web		
Prioridad en negocio:	Media	Riesgo en desarrollo:	Alta
Puntos estimados:	20	Iteración asignada:	1
Programador responsable:	Luis Serrano		
Descripción: El usuario del sistema, requiere tener acceso a cierta información del sistema, por lo que es necesario que el usuario tenga un perfil determinado.			
Criterios de aceptación:			
<ol style="list-style-type: none"> 1. El nombre del usuario debe ser el correo electrónico del usuario 2. La contraseña es la definida por el administrador desde la pantalla de usuarios 3. El usuario Administrador, debe tener acceso solo a la parte web 4. El usuario Profesor o Estudiante, tendrán acceso solo a la parte móvil 5. El usuario Administrador tiene acceso a los módulos de Usuarios, de Cursos y de Actividades 6. El usuario Profesor tiene acceso a pago de mensualidad y descripción de los cursos que imparte el usuario 7. El usuario Estudiante, tiene acceso al visualizar el estado de su mensualidad y los cursos a los que puede acceder para reservar 			

Figura 12: Historia de usuario 1

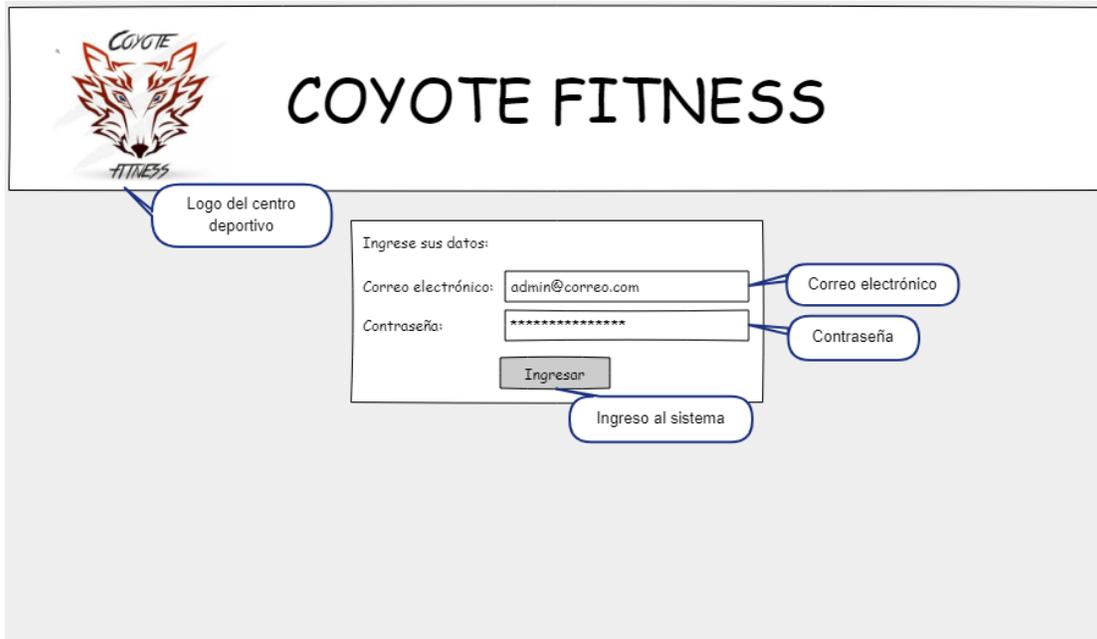


Figura 13: Prototipo de pantalla web – Login

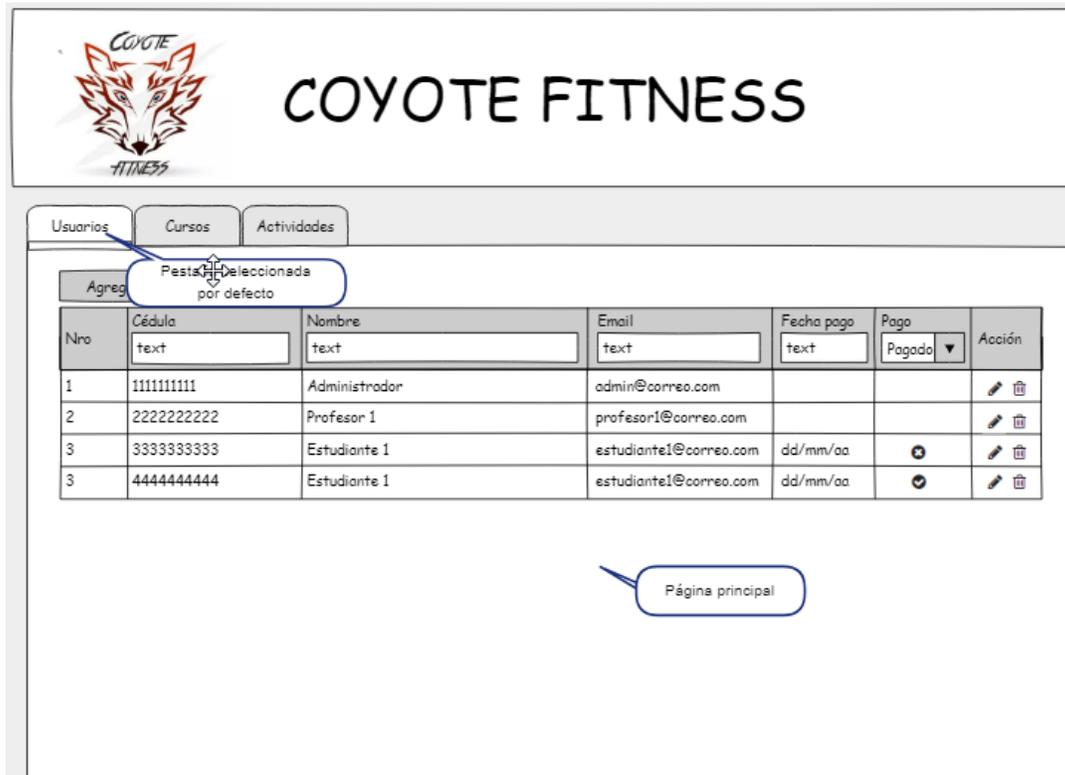


Figura 14: Prototipo de pantalla web – Pantalla principal

Historia de usuario			
Número:	2	Usuario:	Administrador
Nombre de historia:	Administración de usuarios		
Prioridad en negocio:	Media	Riesgo en desarrollo:	Alta
Puntos estimados:	12	Iteración asignada:	2
Programador responsable:	Luis Serrano		
Descripción: El Administrador, requiere poder administrar los usuarios que podrán acceder al sistema, para validar su información y restringir el acceso a ciertos módulos.			
Criterios de aceptación:			
<ol style="list-style-type: none"> 1. Ingresar al módulo de Usuarios 2. Visualizar los usuarios existentes 3. Tener filtros de búsqueda para visualizar ciertos usuarios 4. Agregar usuarios nuevos 5. Permitir seleccionar un rol al usuario 6. El correo electrónico y el número de cédula son datos requeridos. 7. La cantidad a pagar (monto) es requerida. 8. Se puede modificar la información de los usuarios 9. Se puede eliminar el usuario que seleccione 			

Figura 15: Historia de usuario 2

COYOTE FITNESS

Usuarios | **Agregar nuevo usuario** | Roles

Agregar

Filtros de búsqueda

Editar usuario

Eliminar usuario

Nro	Cédula	Nombre	Email	Fecha pago	Pago	Acción
1	1111111111	Administrador	admin@correo.com		Pagado	
2	2222222222	Profesor 1	profesor1@correo.com			
3	3333333333	Estudiante 1	estudiante1@correo.com	dd/mm/aa		
3	4444444444	Estudiante 1	estudiante1@correo.com	dd/mm/aa	3 meses	

Fecha de pago, solo usuario con rol Estudiante

Estado de su mensualidad, si tiene deudas, presenta el tiempo

Figura 16: Prototipo de pantalla web—Administración de usuarios

Usuario [X]

Datos del usuario

Cédula *:

Nombre *:

Email *:

Fecha nacimiento:

Fecha pago:

Rol *:

Guardar un usuario

Fecha pago	Pago
ext	Pagado
mm/aa	
mm/aa	

Rol que se asigna al usuario

Figura 17: Prototipo de pantalla web—Modal nuevo usuario

Modal de edición

Usuario

Cédula *: 1111111111

Nombre *: Administrador

Email *: admin@correo.com

Fecha nacimiento: dd/mm/aa

Fecha pago:

Rol *: Administrador

Guardar

Datos del registro seleccionado para editar

Pagado

Acc

dd/mm/aa

dd/mm/aa

Figura 18: Prototipo de pantalla web–Modal editar usuario

Historia de usuario			
Número:	3	Usuario:	Administrador
Nombre de historia:	Administración de actividades		
Prioridad en negocio:	Media	Riesgo en desarrollo:	Baja
Puntos estimados:	13	Iteración asignada:	3
Programador responsable:	Luis Serrano		
Descripción: El Administrador, requiere administrar todas las actividades que se realizan en el centro deportivo para posteriormente poder asignarlas a distintos cursos.			
Criterios de aceptación:			
1. Ingresar al módulo de Actividades			
2. Visualizar todas las actividades existentes			
3. Tener filtros de búsqueda para visualizar ciertas actividades			
4. Agregar actividades nuevas			
5. El nombre de la actividad y la descripción es importante			
6. Se puede modificar la información de cierta actividad			

Figura 19: Historia de usuario 3

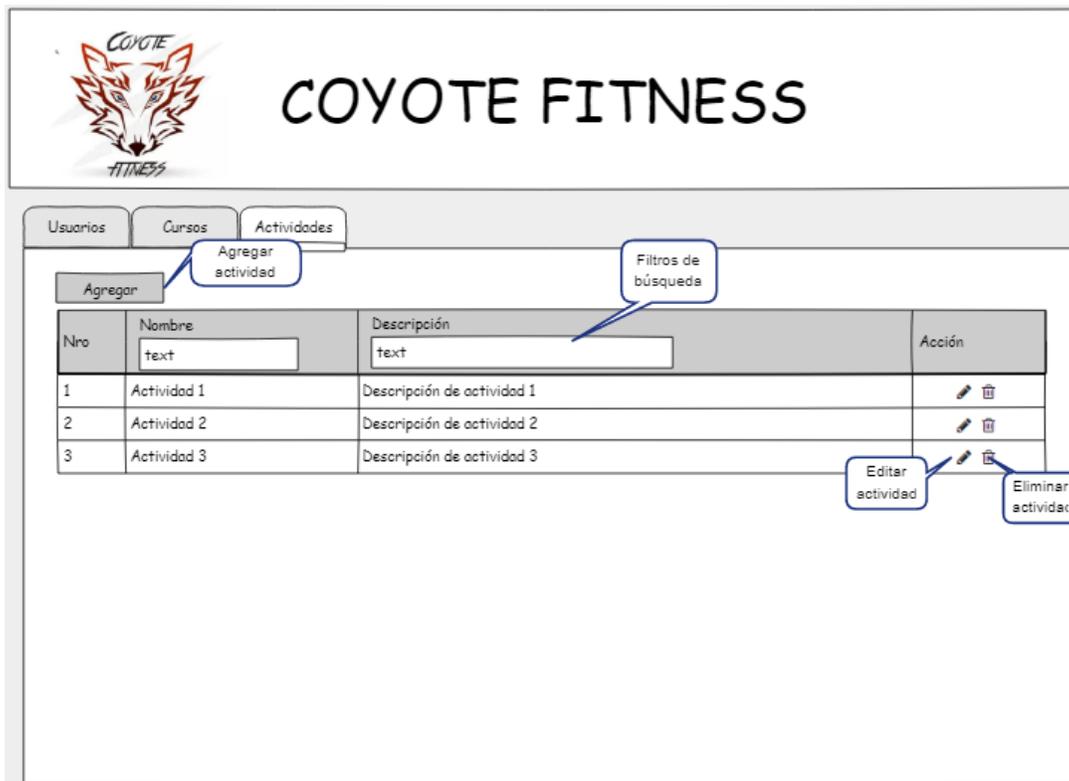


Figura 20: Prototipo de pantalla web – Administración de actividades

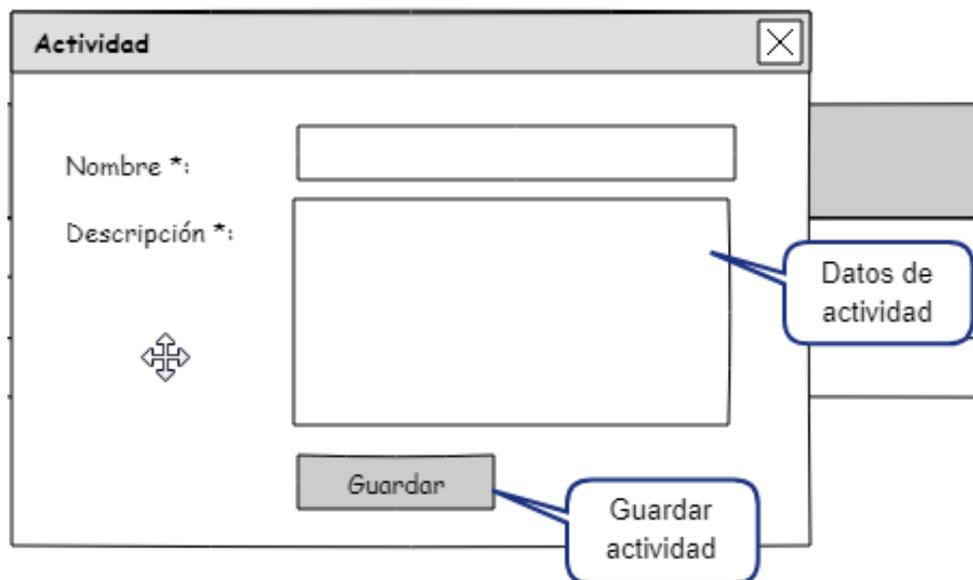


Figura 21: Prototipo de pantalla web – Modal nueva actividad

Actividad

Nombre *: Actividad 1

Descripción *: Descripción de actividad 1

Guardar

Guardar actividad

Datos de la actividad

Figura 22: Prototipo de pantalla web – Modal editar actividad

Historia de usuario			
Número:	4	Usuario:	Administrador
Nombre de historia:	Administración de cursos		
Prioridad en negocio:	Media	Riesgo en desarrollo:	Baja
Puntos estimados:	18	Iteración asignada:	4
Programador responsable:	Luis Serrano		
Descripción: El Administrador, requiere administrar los cursos que se impartirán en el centro deportivo, para que puedan visualizar tanto los profesores como los alumnos.			
Criterios de aceptación:			
<ol style="list-style-type: none"> 1. Ingresar al módulo de Cursos 2. Visualizar todos los cursos existentes 3. Tener filtros de búsqueda para visualizar ciertos cursos 4. Agregar un curso Nuevo 5. Es importante asignar fechas y horarios en los que se impartirán los cursos 6. Se debe asignar un profesor a cada horario de cada curso 7. Establecer el máximo de personas que puede asistir a cada horario del curso 8. Se pueden editar las actividades de cada curso 9. Se puede eliminar los horarios de clases de cada curso y agregar nuevos 10. Se puede visualizar los horarios del curso seleccionado 11. Se puede visualizar las actividades que se van a impartir en cada curso con el tiempo de duración del curso seleccionado 12. Los horarios del curso no se puede cruzar entre si con el mismo profesor, pero si con profesores distintos 			

Figura 23: Historia de usuario 4

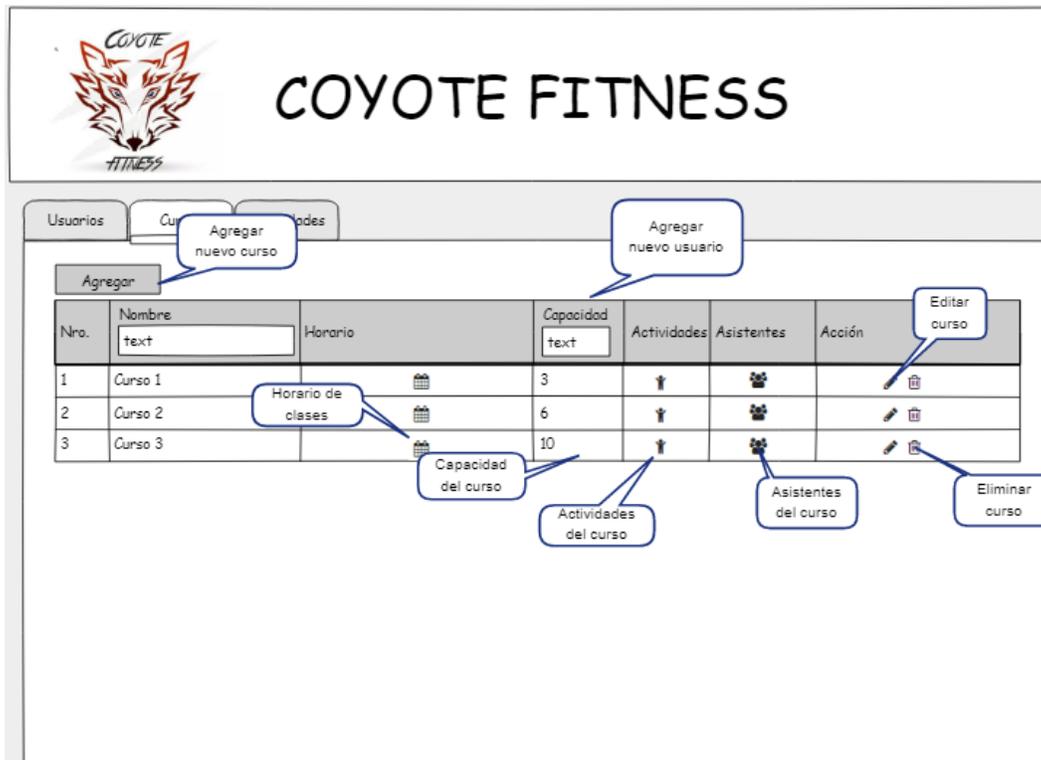


Figura 24: Prototipo de pantalla web–Administración de cursos

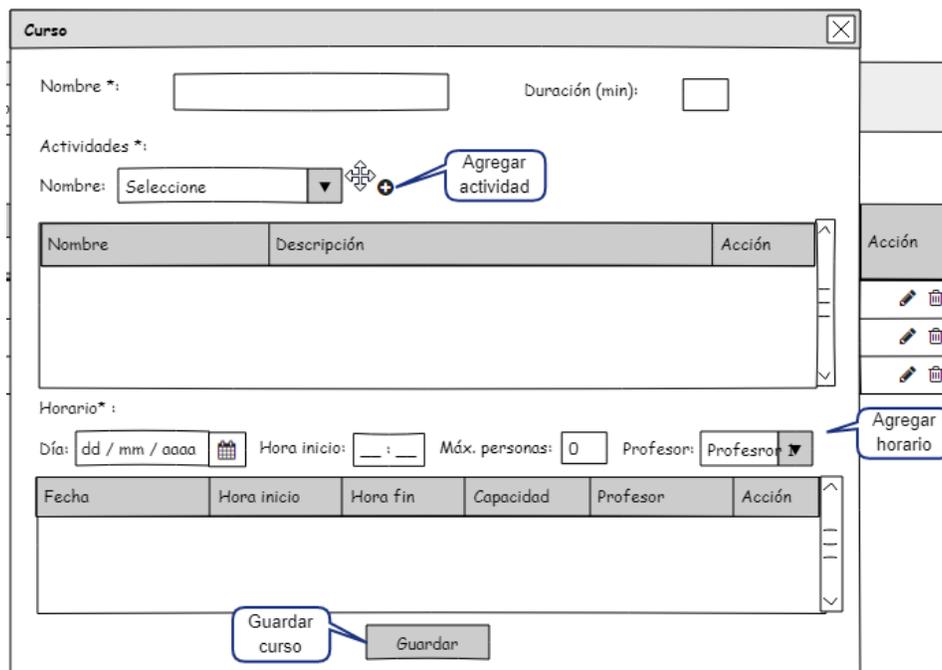


Figura 25: Prototipo de pantalla web – Modal nuevo curso

Curso

Nombre *: Duración (min):

Actividades*:

Nombre:

Nombre	Descripción	Acción
Actividad 1	Descripción 1	<input type="button" value="Eliminar actividad"/>
Actividad 2	Descripción 2	<input type="button" value="Eliminar actividad"/>
Actividad 3	Descripción 3	<input type="button" value="Eliminar actividad"/>

Datos de la actividad

Horario*:

Día: Hora inicio: Máx. personas: Profesor:

Fecha	Hora inicio	Hora fin	Capacidad	Profesor	Acción
dd / mm / aaaa	07:00	08:00	6	Profesor 1	<input type="button" value="Eliminar horario"/>
dd / mm / aaaa	08:00	08:30	10	Profesor 1	<input type="button" value="Eliminar horario"/>

Datos del horario del curso

Figura 26: Prototipo de pantalla web – Modal editar curso

Horario

Nombre *: Duración (min):

Horario*:

Fecha	Hora inicio	Hora fin	Capacidad	Profesor
dd / mm / aaaa	07:00	08:00	6	Profesor 1
dd / mm / aaaa	08:00	08:30	10	Profesor 1

Información de los horarios del curso

Figura 27: Prototipo de pantalla web – Modal visualizar horarios del curso

Curso

Nombre *: Duración (min):

Actividades *:

Nombre	Duración (min)
Actividad 1	10
Actividad 2	30
Actividad 3	20

Información de las actividades del curso

Figura 28: Prototipo de pantalla web – Modal visualizar actividades del curso

Curso

Nombre *: Duración (min):

Horario*:

	Fecha	Hora inicio	Hora fin	Capacidad	Profesor
>	dd / mm / aaaa	07:00	08:00	6	Profesor 1
<	dd / mm / aaaa	08:00	08:30	10	Profesor 1

Nro	Nombre
1	Alumno 1
2	Alumno 2
3	Alumno 3
4	Alumno 4
5	Alumno 5

Información de los asistentes del curso

Figura 29: Prototipo de pantalla web – Modal visualizar asistentes del curso

Historia de usuario			
Número:	5	Usuario:	Profesor
Nombre de historia:	Función del rol de Profesor		
Prioridad en negocio:	Media	Riesgo en desarrollo:	Media
Puntos estimados:	14	Iteración asignada:	5
Programador responsable:	Luis Serrano		
Descripción: El Profesor, requiere tener una aplicación móvil para poder registrar los pagos de los Estudiantes y poder visualizar los horarios y descripción de los cursos asignados a mi persona.			
Criterios de aceptación:			
<ol style="list-style-type: none"> 1. Ingresar como usuario cuyo rol sea de Profesor 2. Visualizar únicamente la opción de Pagos y Cursos 3. Pagos: el profesor debe registrar el pago del estudiante que no ha pagado hasta hace un mes. Debe poder filtrar por número de documento 3. Pagos: se puede seleccionar el estudiante (de la lista de estudiantes cargada) que va a cancelar su mensualidad registrada por el administrador. Aparecerá su nombre, su número de documento y correo electrónico 4. Pagos: Una vez seleccionado puede proceder a pagar su mensualidad si aún no ha pagado el mes, si ya pagó se muestra la fecha en la que pagó y no le deja registrar el pago hasta un mes después del último pago 5. Cursos: Visualizar los cursos que estén asignados al profesor logueado 6. Cursos: Visualizar los horarios de los cursos 7. Cursos: Visualizar la lista de los asistentes registrados en el curso 			

Figura 30: Historia de usuario 5

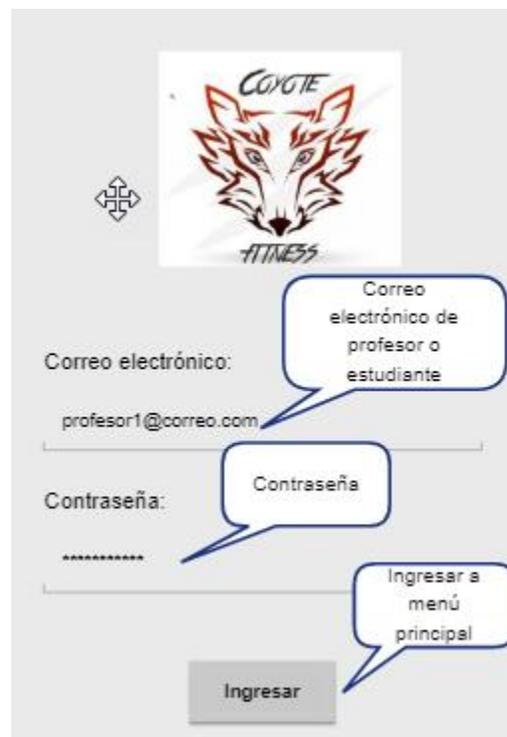


Figura 31: Prototipo de pantalla móvil – Login



Figura 32: Prototipo de pantalla móvil – Menú principal – profesor

Pagos

Cédula:
3333333333

Estudiante 1

Pago (\$):

Registrar

Figura 33: Prototipo de pantalla móvil – Pantalla pago mensualidad – profesor

Cursos

Curso 1

dd / mm / aaaa 07:00 - 08:00

dd / mm / aaaa 08:00 - 08:30

1 - Alumno 1

2 - Alumno 2

3 - Alumno 3

Figura 34: Prototipo de pantalla móvil – Visualizar cursos – profesor

Historia de usuario			
Número:	6	Usuario:	Administrador
Nombre de historia:	Función del rol Estudiante		
Prioridad en negocio:	Media	Riesgo en desarrollo:	Media
Puntos estimados:	12	Iteración asignada:	6
Programador responsable:	Luis Serrano		
Descripción: El Estudiante, requiero tener una aplicación móvil para poder visualizar la fecha máxima de pago y si ya se ha realizado el pago, además requiero visualizar los cursos que se van a impartir con sus horarios y actividades para poder hacer la reserva del curso que más me convenga.			
Criterios de aceptación:			
<ol style="list-style-type: none"> 1. Ingresar como usuario cuyo rol sea de Estudiante 2. Visualizar únicamente la opción de Mis pagos y Mis Cursos 3. En la opción de Mis pagos, debe poder visualizar la fecha de pago de mensualidad y el monto pagado registrado 4. En la opción Mis Cursos debe poder visualizar todos los cursos disponibles 5. Debe poder seleccionar el curso y visualizar la lista de actividades con su descripción 6. Visualizar la lista de los profesores y horarios, la capacidad del horario (y su disponibilidad) 7. Seleccionar el horario que más le convenga o desmarcar si ya no quiere asistir o reservar el horario 8. Debe permitir guardar su selección 			

Figura 35: Historia de usuario 6



Figura 36: Prototipo de pantalla móvil – Menú principal – estudiante

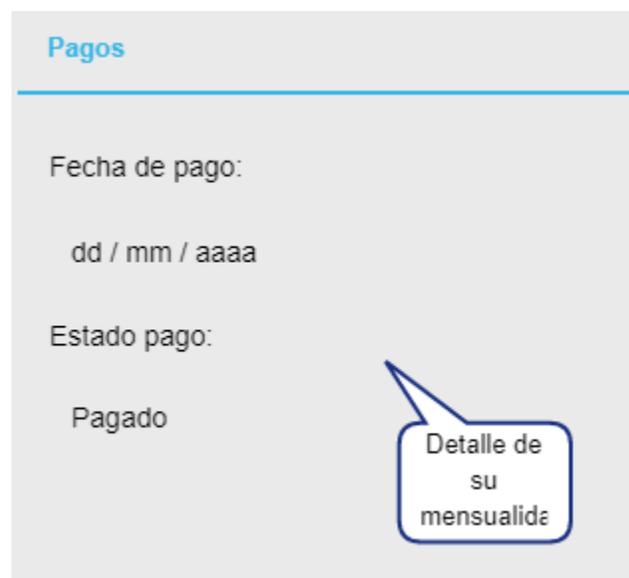


Figura 37: Prototipo de pantalla móvil – Visualizar mensualidad – estudiante

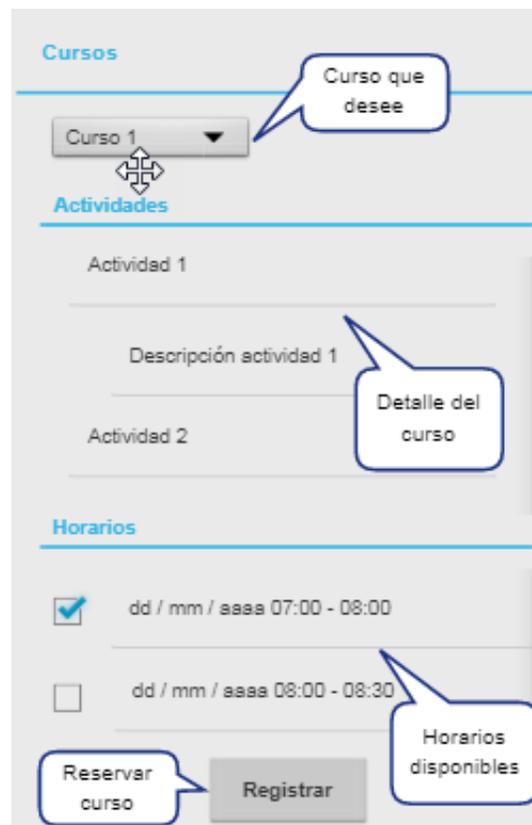


Figura 38: Prototipo de pantalla móvil – Reservar cursos – estudiante

3.1.5. Planificación de iteración

Por cada historia de usuario se describe las tareas que se van a realizar en cada Sprint. Se ha determinado el Sprint 0 y el Sprint 1, se coloca las configuraciones para empezar con la programación, y prepara el ambiente para trabajar con las diferentes plataformas a utilizar.

A continuación, se detalla los Sprints que se van a desarrollar durante el proyecto, con la lista de tareas, para posteriormente empezar con el desarrollo de cada Sprint. En la columna Tarea se detalla el nombre de la tarea que se va a realizar, en Horas está el tiempo establecido para realizar la tarea, en Estimación es determinada de acuerdo a la importancia del desarrollo (es decir que se debe empezar desarrollando

algo para continuar con el resto), la fecha de inicio y fin de la tarea y por último el responsable de realizarla.

Sprint 0						
No	Tarea	Horas	Estimación	F. inicio	F. fin	Responsable
1	Establecer plataformas tecnológicas	16	2	9/8/2019	11/8/2019	Luis Serrano
2	Diseño de la base de datos	24	3	11/8/2019	14/8/2019	Luis Serrano
3	Establecer la arquitectura de software	16	2	14/8/2019	16/8/2019	Luis Serrano
4	Configurar la arquitectura seleccionada	40	5	16/8/2019	21/8/2019	Luis Serrano
		96	12	9/8/2019	21/8/2019	

Figura 39: Tareas para el Sprint 0

Sprint 1						
No	Tarea	Horas	Estimación	F. inicio	F. fin	Responsable
1	Servicios web para login	8	2	21/8/2019	22/8/2019	Luis Serrano
2	Pantalla del login	24	3	22/8/2019	25/8/2019	Luis Serrano
3	Pruebas y correcciones	24	3	25/8/2019	28/8/2019	Luis Serrano
		56	8	21/8/2019	28/8/2019	

Figura 40: Tareas para el sprint 1

Sprint 2						
No	Tarea	Horas	Estimación	F. inicio	F. fin	Responsable
1	Servicios web para la administración de usuarios	16	2	28/8/2019	30/8/2019	Luis Serrano
2	Pantalla principal	24	3	30/8/2019	2/9/2019	Luis Serrano
3	Búsqueda de usuarios	16	2	2/9/2019	4/9/2019	Luis Serrano
4	Agregar usuario nuevo	16	2	4/9/2019	6/9/2019	Luis Serrano
5	Editar y eliminar usuario	8	1	6/9/2019	7/9/2019	Luis Serrano
6	Pruebas y correcciones	24	3	7/9/2019	10/9/2019	Luis Serrano
		104	13	28/8/2019	10/9/2019	

Figura 41: Tareas para el sprint 2

Sprint 3						
No	Tarea	Horas	Estimación	F. inicio	F. fin	Responsable
1	Servicios web para la administración de actividades	16	2	10/9/2019	12/9/2019	Luis Serrano
2	Búsqueda de actividades	8	1	12/9/2019	13/9/2019	Luis Serrano
3	Agregar actividad nueva	8	1	13/9/2019	14/9/2019	Luis Serrano
4	Editar y eliminar actividad	8	1	14/9/2019	15/9/2019	Luis Serrano
5	Pruebas y correcciones	24	3	15/9/2019	18/9/2019	Luis Serrano
		64	8	10/9/2019	18/9/2019	

Figura 42: Tareas para el sprint 3

Sprint 4						
No	Tarea	Horas	Estimación	F. inicio	F. fin	Responsable
1	Servicios web para la administración de cursos	40	5	18/9/2019	23/9/2019	Luis Serrano
2	Búsqueda de cursos	8	1	23/9/2019	24/9/2019	Luis Serrano
3	Agregar curso nuevo	24	3	24/9/2019	27/9/2019	Luis Serrano
4	Editar y eliminar curso	16	2	27/9/2019	29/9/2019	Luis Serrano
5	Visualizar horarios	8	1	29/9/2019	30/9/2019	Luis Serrano
6	Visualizar actividades	8	1	30/9/2019	1/10/2019	Luis Serrano
7	Visualizar asistentes del curso	16	2	1/10/2019	3/10/2019	Luis Serrano
8	Pruebas y correcciones	24	3	3/10/2019	6/10/2019	Luis Serrano
		144	18	18/9/2019	6/10/2019	

Figura 43: Tareas para el sprint 4

Sprint 5						
No	Tarea	Horas	Estimación	F. inicio	F. fin	Responsable
1	Servicios web para realizar las funciones del profesor desde el móvil	24	3	6/10/2019	9/10/2019	Luis Serrano
2	Login del móvil	16	2	9/10/2019	11/10/2019	Luis Serrano
3	Pantalla de menú dependiendo del rol	16	2	11/10/2019	13/10/2019	Luis Serrano
4	Pago de estudiantes	16	2	13/10/2019	15/10/2019	Luis Serrano
5	Visualizar horarios por curso del profesor	16	2	15/10/2019	17/10/2019	Luis Serrano
6	Pruebas y correcciones	24	3	17/10/2019	20/10/2019	Luis Serrano
		112	14	6/10/2019	20/10/2019	

Figura 44: Tareas para el sprint 5

Sprint 6						
No	Tarea	Horas	Estimación	F. inicio	F. fin	Responsable
1	Servicios web para las funciones del estudiante desde el móvil	16	2	20/10/2019	22/10/2019	Luis Serrano
2	Visualización del pago	16	2	22/10/2019	24/10/2019	Luis Serrano
3	Visualización y reserva de cursos	40	5	24/10/2019	29/10/2019	Luis Serrano
4	Pruebas y correcciones	24	3	29/10/2019	1/11/2019	Luis Serrano
		96	12	20/10/2019	1/11/2019	

Figura 45: Tareas para el sprint 6

3.1.6. Incremento

Sprint	Historia	F. inicio	F. fin	Responsable
Sprint 0	Base de proyecto	9/8/2019	21/8/2019	Luis Serrano
Sprint 1	Historia 1	21/8/2019	28/8/2019	Luis Serrano
Sprint 2	Historia 2	28/8/2019	10/9/2019	Luis Serrano
Sprint 3	Historia 3	10/9/2019	18/9/2019	Luis Serrano
Sprint 4	Historia 4	18/9/2019	6/10/2019	Luis Serrano
Sprint 5	Historia 5	6/10/2019	20/10/2019	Luis Serrano
Sprint 6	Historia 6	20/10/2019	1/11/2019	Luis Serrano

Figura 46: Lista de los sprints del proyecto

3.2. Inspección y adaptación

3.2.1. Sprint 0

3.2.1.1. Ejecución de la iteración

Para llevar un mejor control del proyecto, se realiza retroalimentaciones semanales en las que se colocan en un tablero lo que se tiene hecho, lo que se está haciendo y lo que falta por hacer en cada sprint. Los tableros se los presenta por fecha de retroalimentación, los cuales se realizan todos los jueves a las 9 am.

Fecha: 9 de agosto de 2019

Por hacer	En proceso	Finalizado
<ul style="list-style-type: none"> • Establecer plataformas tecnológicas • Diseño de procesos y arquitectura 		

Figura 47: Tablero de control 9/08/2019

Fecha: 8 de agosto de 2019

Por hacer	En proceso	Finalizado
<ul style="list-style-type: none"> • Diseñar la interfaz de usuario • Diseño de la base de datos • Establecer la arquitectura de software 		<ul style="list-style-type: none"> • Establecer plataformas tecnológicas • Diseño de procesos y arquitectura

Figura 48: Tablero de control 8/08/2019

3.2.1.2. Revisión

Las plataformas seleccionadas para la realización del proyecto son vía web y Android. En la web se encuentra la administración de la aplicación, y en el móvil el acceso que tendrá el profesor y el estudiante.

3.2.1.2.1. Diagrama de entidad - relación

Los objetos y relaciones entre objetos utilizados en el desarrollo del sistema se presentan en el siguiente diagrama.

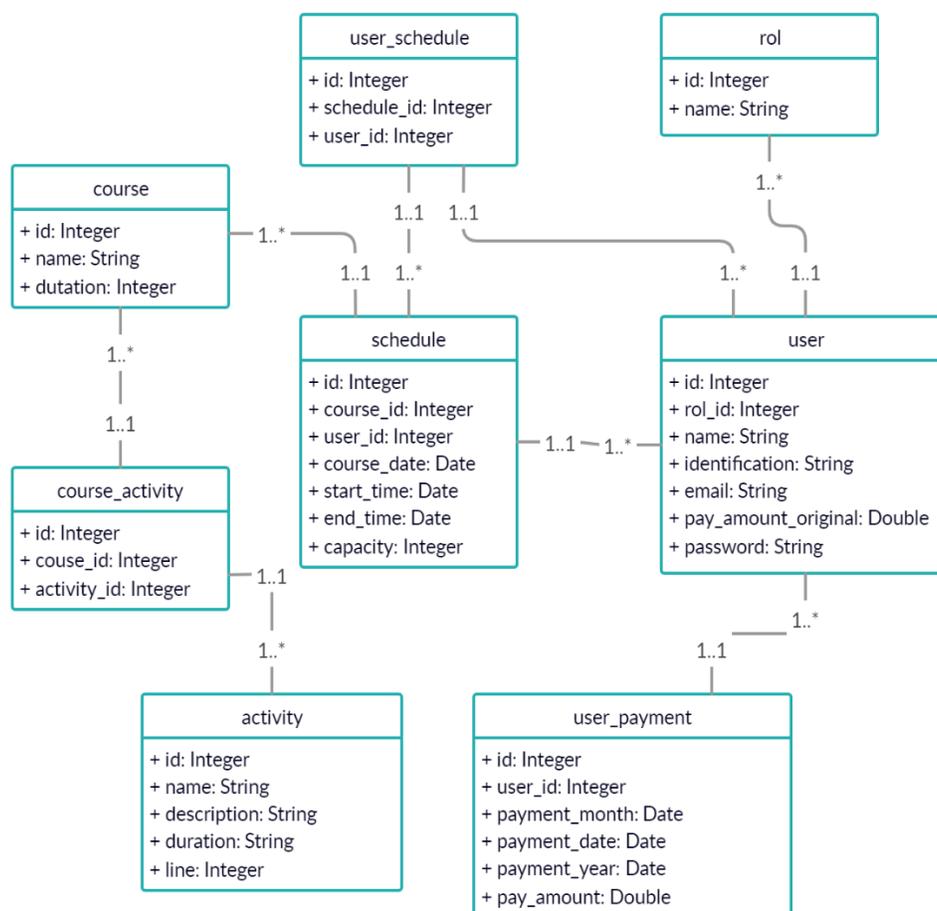


Figura 49: Diagrama de entidad - relación

3.2.1.2.2. Diagrama de clases

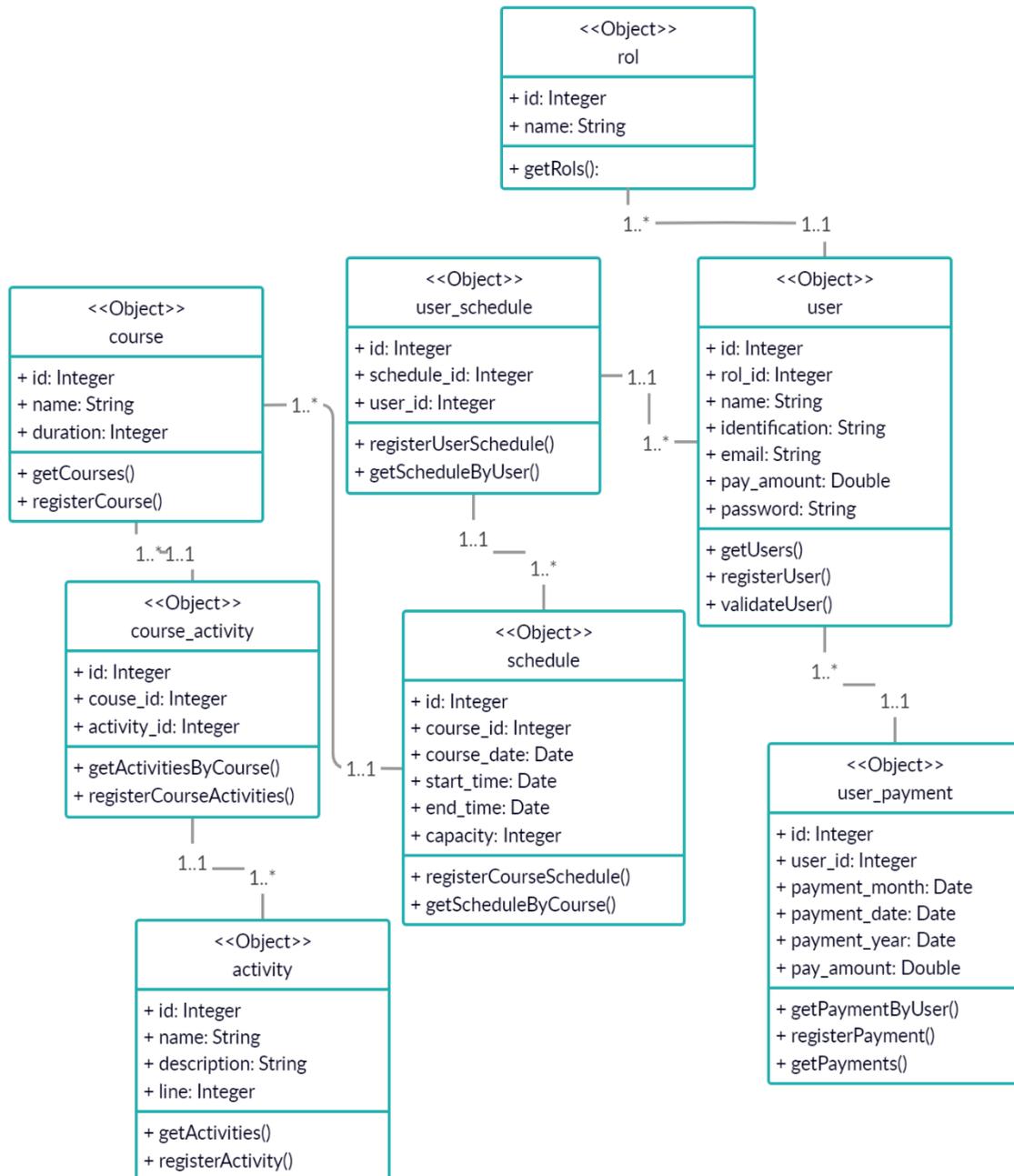


Figura 50: Diagrama de clases

3.2.1.3. Retrospectiva

Las reuniones realizadas para la presentación de los sprints se realizan los días en que finaliza cada sprint en la noche, por facilidad del usuario ya que en las mañanas tiene muchas actividades por realizar.

El primer sprint “Sprint 0” es el más largo, ya que se realiza la configuración de todas las tecnologías seleccionadas para el sistema, además del prototipo de las pantallas a gusto del usuario final.

En el Anexo 2 se encuentra el acta de la reunión en la que se presentó el primer sprint al usuario final, dando como resultado un éxito, ya que aprobó las tecnologías escogidas para el desarrollo de su sistema y las pantallas le pareció de fácil manejo.

3.2.2. Sprint 1

3.2.2.1. Ejecución de la iteración

Fecha: 15 de agosto de 2019

Por hacer	En proceso	Finalizado
<ul style="list-style-type: none"> • Configurar la arquitectura seleccionada • Servicios web para login 	<ul style="list-style-type: none"> • Establecer la arquitectura de software 	<ul style="list-style-type: none"> • Diseñar la interfaz de usuario • Diseño de la base de datos

Figura 51: Tablero de control 15/08/2019

Fecha: 22 de agosto de 2019

Por hacer	En proceso	Finalizado
<ul style="list-style-type: none"> • Pantalla del login • Pruebas y correcciones • Servicios web para la administración de usuarios 	<ul style="list-style-type: none"> • Servicios web para login 	<ul style="list-style-type: none"> • Establecer la arquitectura de software • Configurar la arquitectura seleccionada

Figura 52: Tablero de control 22/08/2019

3.2.2.2. Revisión

3.2.2.2.1. Presentación – Pantalla de login

La pantalla inicial es la del login de la parte web, es importante ya que en esta pantalla se limita el ingreso inadecuado de personas no autorizadas.

Los campos correo y contraseña son requeridos por lo que se pintan de rojo hasta que ingrese la información.



INGRESO DE USUARIO

Correo

Contraseña

[➔ INGRESAR](#)

Figura 53: Presentación – Pantalla de login

3.2.2.3. Retrospectiva

En el Anexo 2 se encuentra el acta de la reunión en la que se presentó el primer sprint “Sprint 0” al usuario final, en la cual, adicionalmente se realizó la presentación de la pantalla del *login*, que se presentó sin problemas y cumplió con las expectativas del usuario final.

¿Qué se hizo bien?

- ✓ Las tecnologías propuestas fueron aceptadas.
- ✓ La arquitectura propuesta cumple con lo que el usuario requiere, por lo que fue aprobada.
- ✓ El diseño de las pantallas cumple con las funcionalidades que el usuario quiere cubrir, por lo que son aprobadas.
- ✓ La presentación del login funciona perfectamente, por lo que la pantalla inicial ha sido aprobada.

¿Qué se puede mejorar?

Al ser la pantalla del login, el usuario no encontró fallas por lo que por el momento no se ha propuesto ninguna mejora en lo presentado.

3.2.3. Sprint 2.

3.2.3.1. Ejecución de la iteración

Fecha: 29 de agosto de 2019

Por hacer	En proceso	Finalizado
<ul style="list-style-type: none"> • Pantalla principal • Búsqueda de usuarios • Agregar usuario nuevo 	<ul style="list-style-type: none"> • Servicios web para la administración de usuarios 	<ul style="list-style-type: none"> • Servicios web para login • Pantalla del login • Pruebas y correcciones

Figura 54: Tablero de control 29/08/2019

Fecha: 5 de septiembre de 2019

Por hacer	En proceso	Finalizado
<ul style="list-style-type: none"> • Editar y eliminar usuario • Pruebas y correcciones • Servicios web para la administración de actividades 	<ul style="list-style-type: none"> • Agregar usuario nuevo 	<ul style="list-style-type: none"> • Servicios web para la administración de usuarios • Pantalla principal • Búsqueda de usuarios

Figura 55: Tablero de control 5/09/2019

3.2.3.2. Revisión

En el Sprint 2 se presenta al usuario el módulo de la administración de usuarios, y todas las funcionalidades con las que cuenta.

3.2.3.2.1. Presentación – Pantalla principal

En la pantalla principal se encuentra la administración de los usuarios como página por defecto.

Se encuentra las 3 opciones de los módulos a los que tiene acceso el usuario, que son Usuarios, Cursos y Actividades.

Nombre completo	Correo	Documento	Rol	Fecha de pago	Estado del pago	Cantidad a pagar
admin	admin@crossfit.com	0123456782	ADMINISTRADOR			
student # 5	student5@crossfit.com	1231231231	ESTUDIANTE	2019-11-24	✓	\$90.00
student # 2	student2@crossfit.com	9090909090	ESTUDIANTE	2019-11-13	✓	\$58.00
student # 3	student3@crossfit.com	0123456782	ESTUDIANTE	2019-11-10	✓	\$55.00
student # 4	student4@crossfit.com	0123456781	ESTUDIANTE	2019-11-13	✓	\$70.00
student # 6	student6@crossfit.com	0123456789	ESTUDIANTE	2019-11-13	✓	\$88.00
student	students@crossfit.com	0123456780	ESTUDIANTE	2019-10-27	✗	\$68.00
test	test@crossfit.com	124323464	ESTUDIANTE	2019-11-13	✓	\$41.00
student # 7	student7@crossfit.com	7766776677	ESTUDIANTE		✗	\$77.00
student # 8	student8@crossfit.com	8787878787	ESTUDIANTE	2019-11-17	✓	\$80.00
student # 9	student9@crossfit.com	9090909091	ESTUDIANTE	2019-11-13	✓	\$90.00
student # 10	student10@crossfit.com	1010101111	ESTUDIANTE	2019-11-17	✓	\$100.00
student # 11	student11@crossfit.com	1122334499	ESTUDIANTE	2019-11-17	✓	\$92.00
student # 12	student12@crossfit.com	1219191213	ESTUDIANTE		✗	\$55.00
student1	student1@crossfit.com	1111222233	ESTUDIANTE	2019-11-30	✓	\$60.00

21 Usuarios

Figura 56: Presentación – Pantalla principal – Administración de usuario

3.2.3.2.2. Presentación – Administración de usuarios

En la pantalla de la administración de usuarios, se puede observar la lista de los usuarios que tienen acceso a los sistemas. Las acciones que se pueden realizar en esta administración son:

- Edición de un usuario.
- Registro de un nuevo usuario.
- Eliminación de un usuario.
- Búsqueda de un usuario por nombre, correo, documento o el rol que tiene.
- Visualiza el usuario que tiene alguna deuda por pagar.

3.2.3.2.3. Presentación – Agregar usuario

Para ingresar un usuario nuevo se debe ingresar todos los datos solicitados, ya que todos son importantes para el registro de este. Por este motivo los campos se encuentran en rojo hasta que el usuario ingrese los datos solicitados.

En la sección rol se selecciona el tipo administrador, profesor y estudiante, para las distintas funcionalidades que tendrá el sistema.



The screenshot shows a web form titled 'USUARIO' with a close button (X) in the top right corner. The form contains several input fields, all of which are highlighted with a red border, indicating they are required. The fields are: 'CORREO', 'CONTRASEÑA', 'CONFIRME LA CONTRASEÑA', 'NOMBRE COMPLETO', 'NO. DOCUMENTO', and 'ROL'. The 'ROL' field is a dropdown menu with the text 'Seleccione' and a downward arrow. At the bottom right of the form, there are two buttons: a green button with a checkmark and the text 'Guardar', and a red button with an 'X' and the text 'Cancelar'.

Figura 57: Presentación – Ingreso de usuario

3.2.3.2.4. Presentación – Editar usuario

La edición de un usuario se puede modificar toda la información del usuario registrado con anterioridad. Si no existe información en el campo, se pinta de color rojo hasta ingresarlo.

Si selecciona Actualizar Contraseña, se puede cambiar la contraseña, por lo que la administración de usuarios es únicamente para un usuario con rol administrador.

USUARIO

CORREO
admin@crossfit.com

ACTUALIZAR CONTRASEÑA

CONTRASEÑA

CONFIRME LA CONTRASEÑA

NOMBRE COMPLETO
admin

NO. DOCUMENTO
0123456782

ROL
ADMINISTRADOR ▼

✓ Guardar ✗ Cancelar

Figura 58: Presentación – Edición de usuario

3.2.3.2.5. Presentación – Eliminar usuario

En la pantalla principal de la administración de usuarios, en cada registro se presenta el ícono de eliminar, para borrar el registro del usuario. Si selecciona esta opción se presenta una ventana emergente de la confirmación para realizar el registro.

Confirmación

⚠ Confirme que desea eliminar el usuario

✓ Yes ✗ No

Figura 59: Presentación – Eliminar de usuario

3.2.3.2.6. Presentación – Validar el pago de los usuarios

El usuario administrador puede revisar si el usuario ha cancelado su pago o no. El ícono de visto verde significa que, si tiene cancelado su mensualidad, y la X roja significa que falta el pago. En la columna de fecha de pago se puede validar cuando ha cancelado la mensualidad.

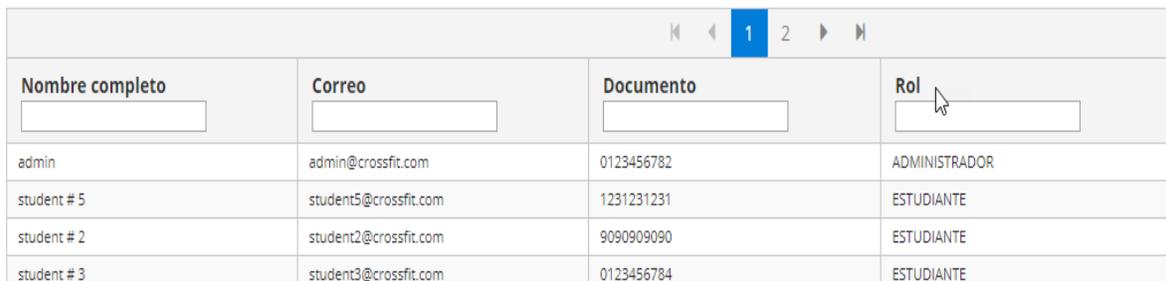
Fecha de pago	Estado del pago
2019-11-24	✓
2019-11-13	✓
2019-11-10	✗
2019-11-13	✓
2019-11-13	✓
2019-10-27	✗

Figura 60: Presentación – Validar el pago de los usuarios

3.2.3.2.7. Presentación – Búsqueda de usuarios

Los filtros de búsqueda de los usuarios se encuentran en la cabecera de la lista de usuarios existentes, por lo que, si desea obtener usuarios dependiendo de determinado criterio de búsqueda, se ingresa en los filtros descritos.

Los criterios de búsqueda disponibles son: el nombre del usuario, correo electrónico, número de documento y rol del usuario.



Nombre completo	Correo	Documento	Rol
admin	admin@crossfit.com	0123456782	ADMINISTRADOR
student # 5	student5@crossfit.com	1231231231	ESTUDIANTE
student # 2	student2@crossfit.com	9090909090	ESTUDIANTE
student # 3	student3@crossfit.com	0123456784	ESTUDIANTE

Figura 61: Presentación – Búsqueda de usuarios

3.2.3.3. Retrospectiva

En el Anexo 3 está el acta de la reunión en la que se presenta al usuario final la administración de usuarios. El resultado de la presentación fue exitoso ya que el flujo fue el correcto y la funcionalidad de la pantalla cumplió con las expectativas del usuario final.

¿Qué se hizo bien?

- ✓ Las pruebas del ingreso de nuevos usuarios fueron correctas y aprobadas.
- ✓ Las pruebas de modificación de usuarios fueron correctas y aprobadas.
- ✓ Las pruebas de eliminación de usuarios fueron correctas y aprobadas.
- ✓ La asignación de rol es correcta y la visualización de la información dependiendo del rol es correcta.

¿Qué se puede mejorar?

Al ser la primera pantalla de administración, el usuario visualizó que la facilidad en la interacción con el sistema y la facilidad para administrar usuarios, por lo que por el momento no se ha propuesto ninguna mejora en lo presentado.

3.2.4. Sprint 3

3.2.4.1. Ejecución de la iteración

Fecha: 12 de septiembre de 2019

Por hacer	En proceso	Finalizado
<ul style="list-style-type: none"> • Búsqueda de actividades • Agregar actividad nueva • Editar y eliminar actividad • Pruebas y correcciones • Servicios web para la administración de cursos 		<ul style="list-style-type: none"> • Agregar usuario nuevo • Editar y eliminar usuario • Pruebas y correcciones • Servicios web para la administración de actividades

Figura 62: Tablero de control 12/09/2019

3.2.4.2. Revisión

En el sprint 3 se presenta al usuario la administración de las actividades, y las funcionalidades que se pueden encontrar en ella. Las actividades son con las que cada curso contará.

3.2.4.2.1. Presentación – Administración de actividades

En la pantalla de administración de actividades, está la lista de actividades y las funcionalidades que se puede hacer que son:

- Registro de una nueva actividad
- Edición de una actividad existente
- Eliminar una actividad
- Buscar las actividades por nombre o descripción



COYOTE FITNESS

USUARIOS CURSOS ACTIVIDADES

ACTIVIDADES

NUEVA

Nombre	Descripción	
actividad # 1	actividad # 1, actividad # 1, actividad # 1, activ	 
actividad # 2	actividad # 2	 
actividad # 3	actividad # 3	 
actividad # 4	actividad # 4	 
actividad # 5	actividad # 5	 

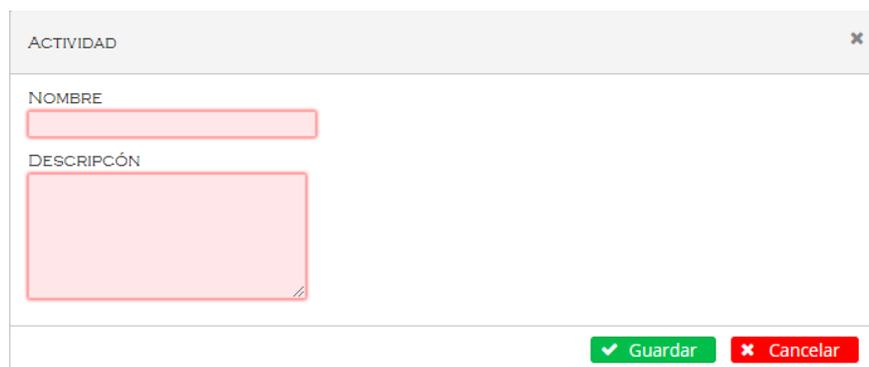
5 Actividades

Figura 63: Presentación – Administración de actividades

3.2.4.2.2. Presentación – Agregar actividad

Para ingresar una nueva actividad, se debe ingresar los 2 campos importantes de la actividad que son su nombre y la descripción de lo que hace la actividad.

Los campos están pintados de rojo hasta que se ingrese información en ellos.



ACTIVIDAD

NOMBRE

DESCRIPCIÓN

Guardar Cancelar

Figura 64: Presentación – Ingreso de una actividad

3.2.4.2.3. Presentación – Editar actividad

Cuando ya se ha agregado una actividad y se desea modificar ya sea su nombre o la descripción de la actividad, aparece una ventana emergente con la información

para modificarla. Cuando el campo no tiene información se pintan de color rojo para identificar los campos obligatorios.

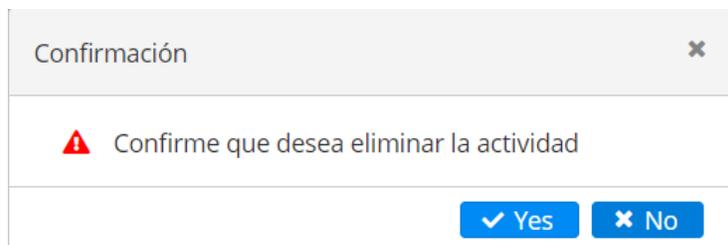


The screenshot shows a window titled 'ACTIVIDAD' with a close button (X) in the top right corner. Below the title bar, there are two input fields. The first is labeled 'NOMBRE' and contains the text 'actividad # 1'. The second is labeled 'DESCRIPCIÓN' and contains the text 'actividad # 1, actividad # 1, actividad # 1, activ'. At the bottom right of the window, there are two buttons: a green button with a checkmark icon labeled 'Guardar' and a red button with an 'X' icon labeled 'Cancelar'.

Figura 65: Presentación – Edición de una actividad

3.2.4.2.4. Presentación – Eliminar actividad

Cuando se desea eliminar una actividad, previamente ingresada, aparece una ventana emergente para confirmar la eliminación de la actividad.



The screenshot shows a dialog box titled 'Confirmación' with a close button (X) in the top right corner. Below the title bar, there is a red warning triangle icon followed by the text 'Confirme que desea eliminar la actividad'. At the bottom right of the dialog, there are two buttons: a blue button with a checkmark icon labeled 'Yes' and a blue button with an 'X' icon labeled 'No'.

Figura 66: Presentación – Eliminación de una actividad

3.2.4.3. Retrospectiva

En el Anexo 4 está el acta de la reunión en la que se presenta la administración de las actividades. El flujo fue correcto sin la presencia de errores y cumpliendo con las expectativas del usuario.

¿Qué se hizo bien?

- ✓ Las pruebas de ingreso de una actividad fueron correctas y aprobadas.
- ✓ Las pruebas de modificación de una actividad fueron correctas y aprobadas.
- ✓ Las pruebas de eliminación de una actividad fueron correctas y aprobadas.
- ✓ Estas actividades serán una lista que se presentarán cuando se vaya a registrar un curso.

¿Qué se puede mejorar?

La administración de actividades es un módulo que beneficia el ingreso de cursos, y la interacción es de forma fácil y sencilla, por lo que no se ha propuesto ninguna mejora en lo presentado.

3.2.5. Sprint 4

3.2.5.1. Ejecución de la iteración

Fecha: 19 de septiembre de 2019

Por hacer	En proceso	Finalizado
<ul style="list-style-type: none"> • Búsqueda de cursos • Agregar curso nuevo 	<ul style="list-style-type: none"> • Servicios web para la administración de cursos 	<ul style="list-style-type: none"> • Búsqueda de actividades • Agregar actividad nueva • Editar y eliminar actividad • Pruebas y correcciones

Figura 67: Tablero de control 19/09/2019

Fecha: 26 de septiembre de 2019

Por hacer	En proceso	Finalizado
<ul style="list-style-type: none"> • Editar y eliminar curso • Visualizar horarios • Visualizar actividades • Visualizar asistentes del curso 	<ul style="list-style-type: none"> • Agregar curso nuevo 	<ul style="list-style-type: none"> • Servicios web para la administración de cursos • Búsqueda de cursos

Figura 68: Tablero de control 26/09/2019

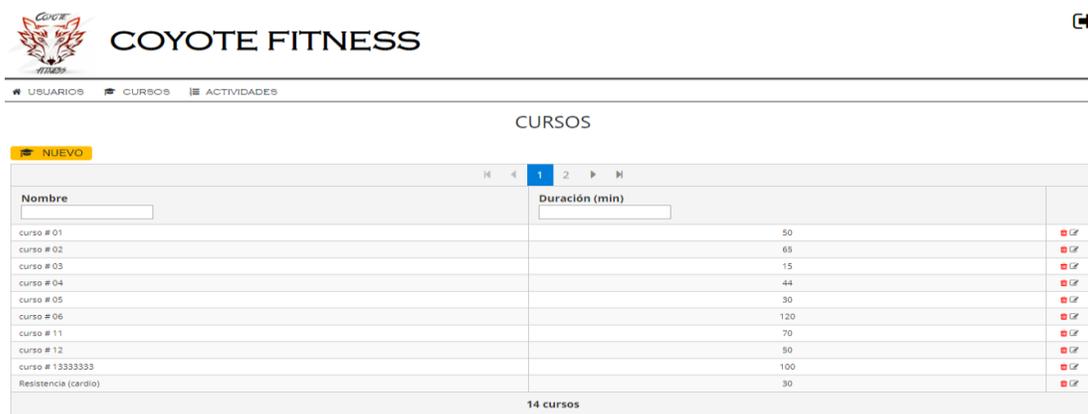
3.2.5.2. Revisión

En el sprint 4 se presenta la administración de los cursos y las funcionalidades, los cursos que se registran son los que se presentará a los usuarios tipo profesor para que tengan conocimiento de sus horarios y actividades que debe realizar, y también los usuarios tipo estudiante podrán observar para hacer la reserva en el horario que les convenga.

3.2.5.2.1. Presentación – Administración de curso

La pantalla de la administración de cursos está la lista de los cursos agregados por el administrador, y las funcionalidades que esta administración tiene, que son las siguientes:

- Búsqueda de los cursos por nombre y duración
- Visualización de las actividades de cada curso
- Visualización del horario de cada curso
- Visualización de los asistentes a cada curso
- Ingreso de cursos
- Edición de cursos
- Eliminación de cursos



COYOTE FITNESS

USUARIOS CURSOS ACTIVIDADES

CURSOS

NUEVO

Nombre	Duración (min)	
curso # 01	50	 
curso # 02	65	 
curso # 03	15	 
curso # 04	44	 
curso # 05	30	 
curso # 06	120	 
curso # 11	70	 
curso # 12	50	 
curso # 13333333	100	 
Resistencia (cardio)	30	 

14 cursos

Figura 69: Presentación – Administración de cursos

3.2.5.2.2. Presentación – Agregar curso

Para agregar un curso es necesario ingresar el nombre del curso, las actividades que se harán en el curso, los horarios en los que estará disponible el curso con el profesor que impartirá y la capacidad de estudiantes que podrá asistir al mismo.

Las actividades que aparecerán para poder ser seleccionadas son las que han sido previamente ingresadas en el módulo de administración de actividades, y tiene la opción de ubicar el orden de las actividades como convenga al usuario.

Los horarios ingresados para cada curso deben tener especificado a parte de la fecha y hora, el profesor encargado de impartirla y el máximo de estudiantes que alcanzan en el curso.

CURSO

NOMBRE

DURACIÓN (MIN)

Actividades *:

SELECCIONE LA ACTIVIDAD NO. REPETICIONES

Horarios *:

Día * Hora inicio * Hora fin *

Capacidad de estudiante * Profesor *

Fecha	Hora Inicio	Hora Fin	Capacidad	Profesor
		1		

Guardar Cancelar

Figura 70: Presentación – Agregar curso

3.2.5.2.3. Presentación – Editar curso

Para la edición del curso, se puede modificar toda la información agregada previamente al agregar un nuevo curso.

Se pueden eliminar tanto las actividades ingresadas, como los horarios establecidos. Adicionalmente se puede visualizar la lista de los asistentes a los diferentes horarios establecidos, esto es en caso de que los estudiantes hayan realizado la reserva del curso desde sus móviles.

Actividades *:

SELECCIONE LA ACTIVIDAD

ACTIVIDAD # 3 - REPETICIONES: 1

ACTIVIDAD # 3 - REPETICIONES: 1

ACTIVIDAD # 2 - REPETICIONES: 20

^

^^

v

vv

Figura 71: Presentación – Editar las actividades

Horarios *:

Día * Hora inicio * Hora fin *

Capacidad de estudiante * Profesor *

SELECCIONE EL PROFESOR

1

Fecha	Hora Inicio	Hora Fin	Capacidad	Profesor	
2019-10-01	01:00:00	07:00:00	12	teacher # 3	<input type="button" value="🗑️"/> <input type="button" value="👤"/>
2019-10-03	06:00:00	12:00:00	10	teacher # 13	<input type="button" value="🗑️"/> <input type="button" value="👤"/>

Figura 72: Presentación – Editar horarios

ESTUDIANTES DEL CURSO # 02

1

Nombre
student # 3
student # 2
student # 5

Figura 73: Presentación – Visualizar los asistentes al curso

3.2.5.2.4. Presentación – Eliminar curso

Al eliminar un curso aparece una ventana emergente de confirmación para asegurar que va a eliminar el curso.

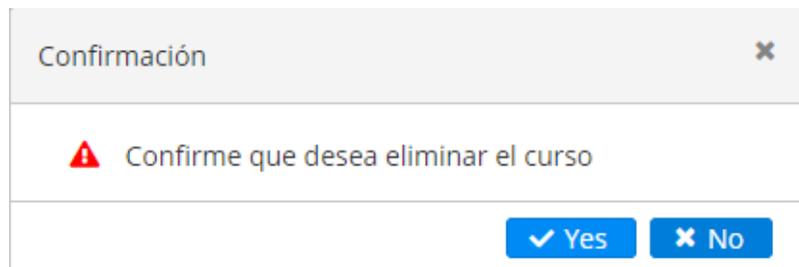


Figura 74: Presentación – Eliminación de un curso

3.2.5.3. Retrospectiva

En el Anexo 5 se presenta la reunión con el usuario en la que se le presenta la administración de los cursos. Fue una presentación exitosa, ya que cumple con las expectativas del usuario y no se presentó ningún error.

¿Qué se hizo bien?

- ✓ Las pruebas de ingreso de un curso nuevo fueron correctos y aprobados.
- ✓ Las pruebas de modificación de un curso fueron correctos y aprobados.
- ✓ Las pruebas de eliminación de un curso fueron correctos y aprobados.
- ✓ La visualización de los horarios del curso es correcta.
- ✓ La visualización de los asistentes al curso es correcta, se visualizan asistentes ingresados por script ya que esta información se llenará cuando el estudiante reserve el curso.

¿Qué se puede mejorar?

La administración de cursos es un módulo que básicamente va a interactuar con todos los requerimientos del usuario, y la interacción es de forma fácil y sencilla, por lo que no se ha propuesto ninguna mejora en lo presentado.

3.2.6. Sprint 5

3.2.6.1. Ejecución de la iteración

Fecha: 3 de octubre de 2019

Por hacer	En proceso	Finalizado
<ul style="list-style-type: none"> • Pruebas y correcciones • Servicios web para realizar las funciones del profesor desde el móvil • Login del móvil 		<ul style="list-style-type: none"> • Agregar curso nuevo • Editar y eliminar curso • Visualizar horarios • Visualizar actividades • Visualizar asistentes del curso

Figura 75: Tablero de control 3/10/2019

Fecha: 10 de octubre de 2019

Por hacer	En proceso	Finalizado
<ul style="list-style-type: none"> • Pantalla de menú dependiendo del rol • Pago de estudiantes • Visualizar horarios por curso del profesor 	<ul style="list-style-type: none"> • Login del móvil 	<ul style="list-style-type: none"> • Pruebas y correcciones • Servicios web para realizar las funciones del profesor desde el móvil

Figura 76: Tablero de control 10/10/2019

3.2.6.2. Revisión.

Una vez finalizada la parte de administración en la web, se desarrolla la parte móvil, por lo que en el sprint 5 se presenta la aplicación móvil.

Primero se presenta el login, por ende, las distintas funcionalidades a la que tiene acceso los diferentes roles que son profesor y estudiante, y todas las acciones que puede realizar el usuario tipo profesor.

3.2.6.2.1. Presentación – Login móvil

El login en la parte móvil es similar al de la parte web, en cuanto a que se debe ingresar el correo electrónico y la contraseña enviada por el usuario administrador.

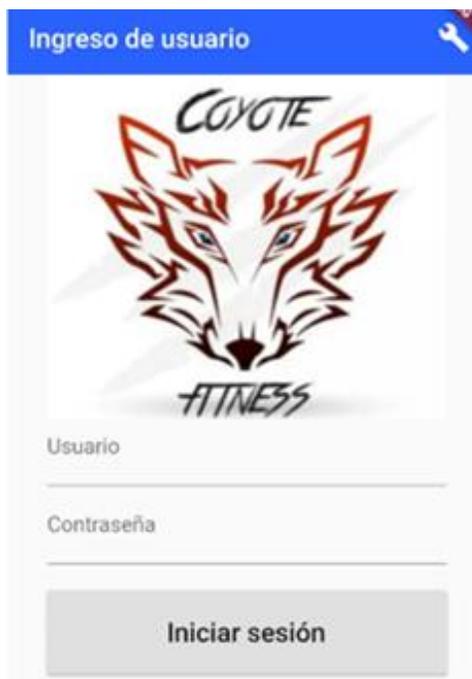


Figura 77: Presentación – Login móvil

3.2.6.2.2. Presentación – Menú que visualiza el profesor

Las opciones que tiene el usuario tipo profesor, son la recepción de pago del estudiante, y la visualización de los cursos a los que está asignado el profesor logueado.

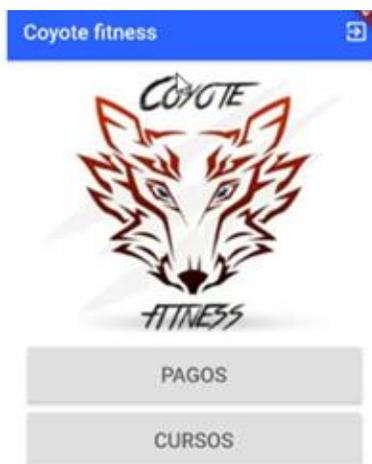


Figura 78: Presentación – Menú que el profesor puede visualizar

3.2.6.2.3. Presentación – Menú que visualiza el estudiante

Las opciones que tiene el usuario tipo estudiante son las opciones de visualizar el estado del pago de sus mensualidades y la reserva de cursos.

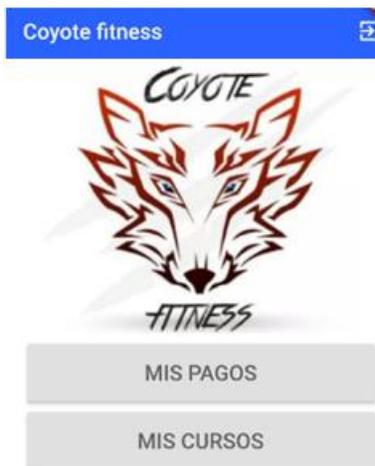


Figura 79: Presentación – Menú que el estudiante puede visualizar

3.2.6.2.4. Presentación – Pago de estudiantes

En la pantalla del pago de estudiante, el usuario profesor puede registrar el pago del estudiante que seleccione, también puede realizar la búsqueda de un estudiante ingresando el número de cédula.

Si el estudiante seleccionado no tiene deudas, se presenta un mensaje informativo de color verde, con la fecha del último pago. Pero si el estudiante no ha cancelado todavía, se presenta un mensaje en rojo con la fecha del último pago y el valor que debe cancelar, así el usuario profesor tendrá el conocimiento del valor que adeuda el estudiante.

Después de seleccionar el estudiante, escoge el botón guardar y automáticamente se cambia el estado del estudiante.

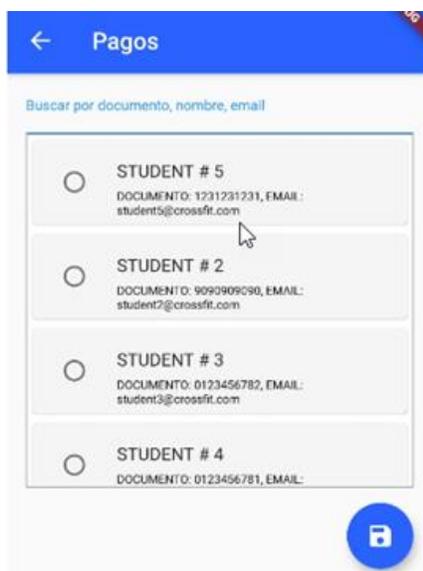


Figura 80: Presentación – Pago de estudiantes

Buscar por documento, nombre, email

STUDENT # 5
DOCUMENTO: 1731231231, EMAIL: student5@crossfit.com

STUDENT # 2
DOCUMENTO: 9090909090, EMAIL: student2@crossfit.com

STUDENT # 3
DOCUMENTO: 0123456782, EMAIL: student3@crossfit.com

STUDENT # 4
DOCUMENTO: 0123456781, EMAIL:

Fecha del ultimo pago: 11/24/2019

Figura 81: Presentación – Estudiante sin deuda

Buscar por documento, nombre, email

0123456780

STUDENT
DOCUMENTO: 0123456780, EMAIL: student@crossfit.com

Fecha del ultimo pago: 10/27/2019

El estudiante no ha pagado

Registrar el pago de:

\$65.0



Figura 82: Presentación – Estudiante con deuda

3.2.6.2.5. Presentación – Visualizar horario del profesor

La pantalla para visualizar el horario del profesor presenta los cursos registrados bajo la responsabilidad del profesor, y al ingresar a cada uno de esos cursos se puede visualizar la información detallada de cada curso, como la lista de actividades que el profesor deberá realizar en el curso, y los horarios que tiene cada profesor asignado a ese curso.

Además, si selecciona uno de los horarios de un profesor específico, se presenta la lista de asistentes que han reservado su clase, para que pueda tener en cuenta quien puede ingresar y quién no.



Figura 83: Presentación – Visualización de cursos del profesor

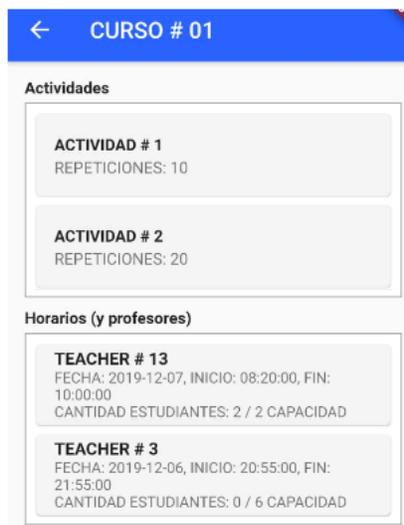


Figura 84: Presentación – Descripción de los cursos

Estudiantes de la clase.
Cantidad: 2 / 2



CERRAR

Figura 85: Presentación – Asistentes del curso

3.2.6.3. Retrospectiva

En el Anexo 6, se realiza la primera presentación con el dispositivo móvil, con la cual simula el ingreso del usuario que tendrá el rol de profesor. En conclusión, la aplicación móvil cumplió con las expectativas del usuario a pesar que se realizó unos ligeros cambios en la presentación a lo que se les había presentado en los prototipos de pantalla.

¿Qué se hizo bien?

- ✓ Las pruebas de login fueron correctos y aprobados.
- ✓ Las pruebas de la visualización de cursos fueron correctos y aprobados.
- ✓ Las pruebas del registro del pago de las mensualidades del estudiante fueron correctas y aprobadas.

¿Qué se puede mejorar?

La presentación del dispositivo móvil fue correcta y al usuario le gustó la parte visual y la forma de interactuar con el sistema, por lo que no se ha propuesto ninguna mejora en lo presentado.

3.2.7. Sprint 6

3.2.7.1. Ejecución de la iteración

Fecha: 17 de octubre de 2019

Por hacer	En proceso	Finalizado
<ul style="list-style-type: none"> • Pruebas y correcciones • Servicios web para realizar las funciones del estudiante desde el móvil • Visualización del pago 		<ul style="list-style-type: none"> • Login del móvil • Pantalla de menú dependiendo del rol • Pago de estudiantes • Visualizar horarios por curso del profesor

Figura 86: Tablero de control 17/10/2019

Fecha: 24 de octubre de 2019

Por hacer	En proceso	Finalizado
<ul style="list-style-type: none"> • Visualización y reserva de cursos • Pruebas y correcciones 		<ul style="list-style-type: none"> • Pruebas y correcciones • Servicios web para realizar las funciones del estudiante desde el móvil • Visualización del pago

Figura 87: Tablero de control 24/10/2019

3.2.7.2. Revisión

En el sprint 6 se presenta las funcionalidades a lo que tiene acceso el usuario tipo estudiante desde la parte móvil.

3.2.7.2.1. Presentación – Visualizar estado del pago

En esta pantalla el estudiante puede visualizar el estado de sus mensualidades, es decir el detalle de sus pagos, en donde encontrará la fecha y hora en la que fueron registrados sus pagos.



Figura 88: Presentación – Visualización del estado del pago del estudiante

3.2.7.2.2. Presentación – Visualizar cursos disponibles para reserva

En esta pantalla el estudiante puede visualizar todos los cursos disponibles, adicionalmente puede visualizar la lista de actividades que conforman este curso, y los horarios en los que se imparte clases, de esta forma el estudiante puede seleccionar el que mejor le convenga y hacer la reserva de este.

Si el curso seleccionado no tiene disponibilidad de curso, presenta al estudiante un mensaje informándole el inconveniente para que pueda seleccionar otro curso. La capacidad de cada curso se especifica al momento de registrar los cursos en al parte web por un usuario administrador.

Para reservar un curso lo único que el estudiante debe hacer es seleccionar el curso deseado y guardar la información, automáticamente el curso queda reservado para ese estudiante.

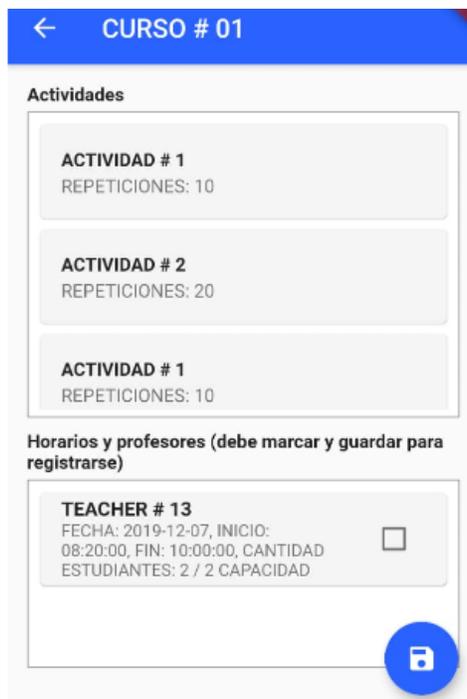


Figura 89: Presentación – Visualización de los cursos para reservar

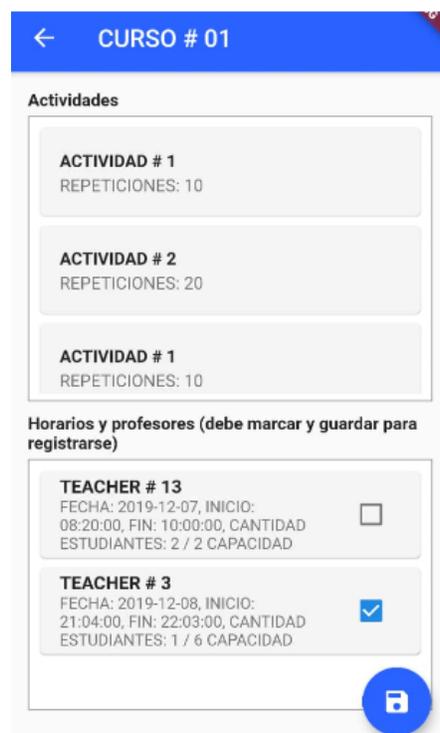


Figura 90: Presentación – Reserva de un curso

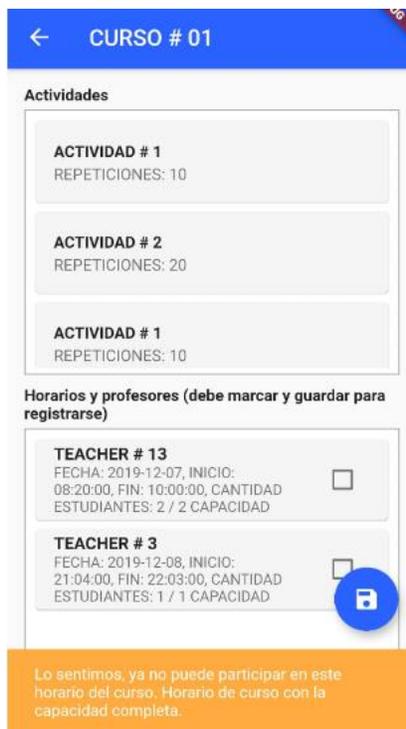


Figura 91: Presentación – Disponibilidad de un curso

3.2.7.3. Retrospectiva

En el Anexo 7, se presenta al usuario la aplicación que en futuro hará uso el usuario tipo estudiante, el flujo fue el esperado por usuario y recorrió sin problema alguno.

¿Qué se hizo bien?

- ✓ Se prueba la visualización de los datos del pago del estudiante correctamente y aprobado.
- ✓ Se prueba la visualización de la información de los cursos disponibles correctamente y aprobado.
- ✓ Se prueba el registro de los cursos correctamente y sin errores.

¿Qué se puede mejorar?

La parte en que el usuario interactúa como estudiante desde la parte móvil, es de forma fácil y sencilla, por lo que no se ha propuesto ninguna mejora en lo presentado.

3.2.7.4. Retrospectiva – Prueba de accesibilidad

Luego de haber realizado la prueba de accesibilidad en la aplicación móvil, se obtienen las siguientes recomendaciones:

3.2.7.4.1. Etiqueta del elemento

Es posible que este elemento no tenga una etiqueta que pueda leer un lector de pantalla.

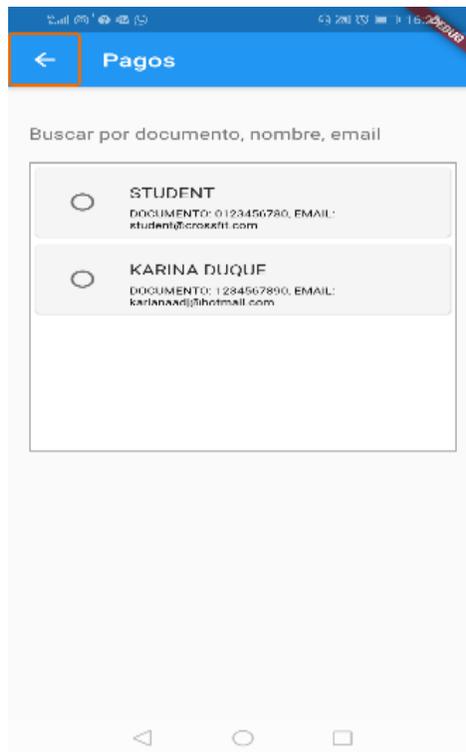
3.2.7.4.2. Contraste del texto

La relación de contraste del texto del elemento, se basa en un color de primer plano estimado de #FFFFFF y un color de fondo estimado de #2196F3. ¿Por qué no utilizas una relación de contraste superior a 4,50 para textos de pequeño tamaño o a 3,00 para textos de gran tamaño?



3.2.7.4.3. Contraste del botón

La relación de contraste del color del botón de la pantalla. Esta relación se basa en un color de primer plano estimado de #FFFFFF y un color de fondo estimado de #2196F3. ¿Por qué no utilizas una relación de contraste superior a 4,50 para textos de pequeño tamaño o a 3,00 para textos de gran tamaño?



3.2.7.4.4. *Contraste del botón*

La relación de contraste del botón de la pantalla. Esta relación se basa en un color de primer plano estimado de #FFFFFFF y un color de fondo estimado de #2196F3. ¿Por qué no utilizas una relación de contraste superior a 4,50 para textos de pequeño tamaño o a 3,00 para textos de gran tamaño?

3.3.2.7.5. *Contraste del texto*

La relación de contraste del elemento. Esta relación se basa en un color de primer plano estimado de #FFFFFFF y un color de fondo estimado de #2196F3. ¿Por qué no utilizas una relación de contraste superior a 4,50 para textos de pequeño tamaño o a 3,00 para textos de gran tamaño?



Luego de haber tomado en cuenta las recomendaciones en todas las pantallas para etiquetas en las imágenes y contraste de colores las mismas se obtiene el siguiente resultado:



4. Conclusiones y recomendaciones

4.1. Conclusiones

La información proporcionada por el centro deportivo Crossfit Coyote Fitness, detalla el tiempo que los estudiantes pierden al esperar la disponibilidad del curso que desean, y la dificultad de saber que estudiante está adeudando el curso, por el cual se proponer una mejora para la realización de su trabajo automatizando los procesos.

Ya que el proyecto requiere una parte web y una parte móvil, el diseño y el desarrollo del proyecto debe enfocarse a la integración de las dos partes, por el cual se crea un modelo que cumple con las necesidades de desarrollo del sistema.

La información que se visualiza en la parte móvil tiene total administración desde la parte web, por lo que la persona con acceso a la parte web, puede ingresar la información que desee.

El sistema móvil es totalmente amigable y brinda al usuario la facilidad de visualizar y reservar los cursos de su preferencia, registrar los pagos del estudiante y visualizar la asistencia a los diferentes cursos.

Gracias al marco de trabajo utilizado, SCRUM, se tiene un constante contacto con el usuario para desarrollar el sistema de acuerdo con las necesidades principales del mismo, presentando resultados y recibiendo retroalimentación.

Mediante el análisis de accesibilidad, el proyecto presenta los estándares de accesibilidad necesarios para personas con discapacidad visual leve después de haber tomado en cuenta factores como el tamaño del texto, contraste de colores, inversión de colores y efectos visuales.

4.2. Recomendaciones

La información de los pagos de los estudiantes es un elemento importante para el manejo del centro deportivo, por lo que se recomienda que se implemente políticas de respaldo de información en caso de fallo de base de datos.

El centro deportivo cuenta con una pensión fija de pago de los servicios que brinda, para futuras implementaciones se recomienda que tengan en cuenta valores variables para que ofrezca al estudiante combos o promociones.

El conocimiento de los movimientos del centro deportivo es importante para conocer el historial de la empresa, se recomienda considerar el desarrollo de datos históricos para conocer el avance de la empresa.

Para que la aplicación móvil sea más accesible se debe realizar pruebas manuales siguiendo los estándares de accesibilidad W3C mismos que son los más aceptados a nivel mundial para lograr un mayor grado de accesibilidad.

Para atraer más estudiante se recomienda tener una parte en la aplicación móvil que sea informativa y no requiera tener un ingreso de usuario para poder visualizarlo.

Para que los estudiantes tengan presente sus clases se recomienda que pongan notificaciones o recordatorios sobre su horario de clases.

REFERENCIAS

- Accessibility Scanner - Android Accessibility Help. (2019). Recuperado el 8 de diciembre de 2019 de <https://support.google.com/accessibility/android/faq/6376582?hl=en>
- Blancarte, O. (2019). *Java Persistence Api*. Recuperado el 1 de septiembre de 2019 de <https://www.oscarblancarteblog.com/tutoriales/java-persistence-api-jpa/>
- Blancarte, O. (s.f.). *Java Persistence Api*. Recuperado el 4 de septiembre 2019 de <https://www.oscarblancarteblog.com/tutoriales/java-persistence-api-jpa/>
- Catalani, E. (2007). Arquitectura Modelo Vista Controlador. Recuperado el 9 de septiembre de 2019 de <https://exequielc.wordpress.com/2007/08/20/arquitectura-modelovistacontrolador/>
- Caules, C. Á. (2013). *Spring MVC*. Recuperado el 11 de septiembre 2019 de <https://www.arquitecturajava.com/spring-mvc-configuracion/>
- Cleventy. (s.f.). *Spring Security Configuration*. Recuperado el 16 de septiembre de 2019 de <https://cleventy.com/configurando-spring-security/>
- Codenuclear. (2017). *Spring MVC Flow*. Recuperado el 2019 de <https://www.codenuclear.com/spring-mvc-flow-with-example/>
- CrossFit SingularBox. (2019). *CrossFit SingularBox*. Recuperado el 17 de septiembre de 2019 de <http://crossfitsingularbox.com/about-us/que-es-crossfit/>
- Culturacion. (s.f.). Que es MySQL. Recuperado el 20 de septiembre de 2019 de <http://culturacion.com>: <http://culturacion.com/que-es-mysql/>
- David. (2017). Mvc En AngularJS. Recuperado el 23 de septiembre de 2019 de <https://www.guidacode.com/2017/angularjs/explicacion-del-patron-mvc-en-angularjs/>

- Developers Android Studio. (2019). *Developers Android Studio*. Recuperado el 25 de septiembre de 2019 de <https://developer.android.com/studio/intro/index.html?hl=es-419>
- Díaz, A. J. (s.f.). *Tomcat , Apache Tomcat*. Recuperado el 30 de septiembre de 2019 de <http://ajpdsoft.com/modules.php?name=Encyclopedia&op=content&tid=769>
- Equipo geek. (2018). *Spring Framework* . Recuperado el 3 de octubre de 2019 de <https://ifgeekthen.everis.com/es/spring-framework>
- Exes. (s.f.). El Lenguaje SQL. Recuperado el 8 de octubre de 2019 de http://mundoracle.com/el-lenguaje-sql.html?Pg=sql_plsql_2.htm
- Fatima, H. (2018). *Understanding The Differences Between Angular 1.0 And Angular 4.0*. Recuperado el 11 de octubre de 2019 de <https://blog.resellerclub.com/understanding-the-differences-between-angular-1-0-and-angular-4-0/>
- Fdlwiki Elp. (2017). Control De Versiones Con Git Y GitHub. Recuperado el 15 de octubre de 2019 de http://wikis.fdi.ucm.es/ELP/Control_de_versiones_con_Git_y_GitHub
- García, L. (2010). *Tomcat 7: Nuevas Características* . Recuperado el 19 de octubre 2019 de <https://unpocodejava.com/2010/08/09/tomcat-7-nuevas-caracteristicas/>
- García, M. (2012). SQL Conceptos. Recuperado el 21 de octubre de 2019 de https://geotalleres.readthedocs.io/es/latest/conceptossql/conceptos_sql.html
- Geeks For Geeks. (2019). *Geeks for geeks. A computer science portal for geeks*. Recuperado el 24 de octubre de 2019 de <https://www.geeksforgeeks.org/difference-between-typescript-and-javascript/>
- Guerrero. (2015). *Java Hibernate*. Recuperado el 27 de octubre de 2019 de <http://programaenlinea.net/que-es-java-hibernate/>

- Guevara, I. J. (2019). Análisis y diseño detallado de aplicaciones informáticas de gestión. Recuperado el 31 de octubre de 2019 de <https://sites.google.com/site/adai6jfm/home/metodologas-de-desarrollo>
- Guru99. (2019). *Typescript vs Javascript: What's The Difference*. Recuperado el 4 de noviembre de 8 de noviembre de 2019 de <https://www.guru99.com/typescript-vs-javascript.html>
- Hidalgo, H. (s.f.). *Intellij*. Recuperado el 2019 de <https://medium.com/@hhugohm/https-medium-com-hhugohm-intellij-ide-43e68801bc5a>
- Hostingpedia. (2019). *MySQL*. Recuperado el 10 de noviembre de 2019 de <https://hostingpedia.net>: <https://hostingpedia.net/mysql.html>
- Huaman, W. C. (2018). Los 10 Patrones Comunes De Arquitectura De Software. Recuperado el 13 de noviembre 2019 de <https://medium.com/@maniakhitoccori/los-10-patrones-comunes-de-arquitectura-de-software-d8b9047edf0b>
- Ionos. (2019). Spring Boot Tutorial Para Crear Aplicaciones En Java. Recuperado el 15 de noviembre 2019 de <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/spring-boot-tutorial/>
- Isma, E. (2015). Que Es Git. Recuperado el 18 de noviembre 2019 de <https://codigofacilito.com>: <https://codigofacilito.com/articulos/que-es-git>
- Junta De Andalucía. (s.f.). *JPA*. Recuperado el 21 de noviembre de 2019 de <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/96>
- Junta De Andalucía. (s.f.). *Hiberbanate*. Recuperado el 2019 de <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/97>
- López, M. M. (2018). Dart 2.1 Es Ahora Mas Pequeño Y Rapido. Recuperado el 22 de noviembre de 2019 de <https://www.unocero.com/software/apps/dart-2-1-es-ahora-mas-pequeno-y-rapido/>
- Molina Montero, B., Vite Cevallos, H., & Dávila Cuesta, J. (2018). Metodologías ágiles frente a las tradicionales en el proceso de desarrollo de software. *EspiraesRevistaMultidisciplinaria de investigación*, 17. Recuperado el 24 de

- noviembre de 2019 de <https://okhosting.com/blog/metodologias-del-desarrollo-de-software/>
- Molina, T. (2018). Angular: Arquitectura del framework. Recuperado el 26 de noviembre de 2019 de <https://medium.com/angular-chile/angular-arquitectura-del-framework-e46204f38fef>
- Openinnova. (2019). *Google Flutter*. Recuperado el 28 de noviembre de 2019 de <https://www.openinnova.es/google-flutter-espanol-desarrollo-apps-hibridas/>
- Pozas, O. (2017). Flutter I: Introduccion Al Desarrollo. Recuperado el 29 de noviembre de 2019 de <https://medium.com/maybeoscar/flutter-i-introducci%C3%B3n-al-desarrollo-8c2250d60186>
- Raffino, M. (2018). Lengaje de programación. Recuperado el 30 de noviembre de 2019 de <https://concepto.de/lenguaje-de-programacion/>
- Rodriguez, M. (2012). Trabajando Con IntelliJ IDEA 11. Recuperado el 1 de diciembre de 2019 <https://www.adictosaltrabajo.com/2012/03/26/trabajando-intellijidea/>
- Romero, H. C. (2017). Como Configurar el Tomcat embebido de Spring Boot. Recuperado el 2 de diciembre de 2019 de <https://www.adictosaltrabajo.com/2017/08/01/configurar-el-tomcat-embebido-de-spring-boot/>
- Rouse, M. (2015). MySQL. Recuperado el 3 de diciembre de 2019 de <https://searchdatacenter.techtarget.com/es/definicion/MySQL>
- Ruiz, P. (2013). Arquitectura Cliente Servidor. Recuperado el 6 de diciembre de 2019 de <http://somebooks.es/arquitectura-clienteservidor/>
- Sánchez, J. M. (2012). *Spring Data Jpa*. Recuperado el 8 de diciembre de 2019 de <https://www.adictosaltrabajo.com/2012/10/08/spring-data-jpa/>
- Sifuentes, L., & Macario, A. (2019). Evaluación del uso de metodología scrum en procesos de desarrollo de software area de sistemas GMD 2017. Recuperado el 10 de diciembre de 2019 de <http://repositorio.cientifica.edu.pe:8080/xmlui/handle/UCS/700>

- Sintes, B. (2019). *Visual Studio Code*. Recuperado el 11 de diciembre de 2019 de <http://www.mclibre.org/consultar/informatica/lecciones/vsc.html>
- Softeng. (2019). Metodología Scrum para desarrollo de software. Recuperado el 13 de diciembre de 2019 de <https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum.html>
- Trigas, M. (2019). Gestion de proyectos informáticos. Recuperado el 15 de diciembre 2019 de http://openaccess.uoc.edu/webapps/o2/bitstream/10609/17885/1/mtrigasTF_C0612memoria.pdf
- Tutorialspoint. (2019). *Spring MVC - Framework*. Recuperado el 17 de diciembre de 2019 de https://www.tutorialspoint.com/spring/spring_web_mvc_framework.htm
- Universidad De Alicante. (2014). Introduccion a Spring. Recuperado el 19 de diciembre de 2019 de <http://www.jtech.ua.es/j2ee/publico/spring-2012-13/sesion01-apuntes.html>
- Universidad De Alicante. (s.f.). *Modelo Vista Controlador*. Recuperado el 21 de diciembre de 2019 de <https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>
- Universitat de Girona. (2013). Introduccion a las Metodologias de Desarrollo de Software. Recuperado el 23 de diciembre de 2019 de <http://ima.udg.edu/~sellares/EINF-ES2/Present1011/MetodoPesadesDocumentacio.pdf>
- Varela, C., Miñán, A., Hilera, J. R., Restrepo, F. A., Amado, H., Córdova, M. A., & villaverde, A. (s.f.). Estándares y legislación sobre accesibilidad web. Recuperado el 26 de diciembre de 2019 de www.aenor.org

ANEXOS

Anexo 1: Acta de reunión de recepción de requerimientos

Fecha: 29 de Julio de 2019

Objetivos

- Obtener la información global de lo que requiere el usuario.

Participantes

- Mauricio Mera
- Luis Serrano

Temas tratados

1. Requerimientos globales del usuario
 - El usuario requiere un sistema con el cual pueda administrar su centro deportivo por internet.
 - Requiere administrar los usuarios que van a tener acceso a su sistema.
 - Requiere administrar las actividades que imparte en cada curso.
 - Requiere administrar los horarios de los cursos que imparte por semana.
 - Requiere controlar la capacidad de los estudiantes a su curso.
 - Requiere conocer las mensualidades de los estudiantes.
 - Requiere que el profesor tenga la capacidad de registrar los pagos de los estudiantes.
 - Requiere que el profesor pueda ver quiénes son los asistentes a sus cursos.
 - Requiere que el profesor conozca sus horarios de clases y los cursos que va a impartir.
 - Requiere que el estudiante pueda visualizar el estado de su pago.
 - Requiere que el estudiante pueda visualizar los cursos disponibles en la semana.
 - Requiere que el estudiante pueda hacer las reservas del curso que le convenga.

Acciones y compromisos

- ✓ Propuesta al usuario las mejores plataformas para el uso del sistema que requiere.
- ✓ Propuesta al usuario de las mejores tecnologías para el desarrollo de su sistema.
- ✓ Propuesta al usuario de los prototipos de pantallas para cumplir con los estándares solicitados.

Anexo 2: Acta de reunión Sprint 1

Fecha: 28 de agosto de 2019

Objetivos

- Presentación del primer Sprint al usuario.
- Presentar las plataformas con las que se va a trabajar.
- Presentar la arquitectura seleccionada para el desarrollo del proyecto.
- Presentar los prototipos de las pantallas del sistema.
- Presentar la pantalla del login del usuario en la parte web.

Participantes

- Mauricio Mera
- Luis Serrano

Temas tratados

1. Se presenta las plataformas con las que se va a trabajar y son:
 - Para el administrador va a existir una pantalla en la web.
 - Para el perfil profesor y estudiante van a tener una aplicación móvil Android.
2. Se presenta la arquitectura con la que se va a trabajar:
 - Se presenta el modelo de datos.
 - Los lenguajes con los que se va a trabajar.
 - Los flujos que el sistema va a seguir.
3. Se presenta los prototipos de pantallas del sistema.
 - Prototipos de la parte web.
 - Prototipos de la parte móvil.
4. Se presenta la pantalla inicial que es el login de la parte web.
 - Se presenta y se hace la prueba del funcionamiento de la pantalla.

Resultados de la presentación

- ✓ Las tecnologías propuestas fueron aceptadas.
- ✓ La arquitectura propuesta cumple con lo que el usuario requiere, por lo que fue aprobada.
- ✓ El diseño de las pantallas cumple con las funcionalidades que el usuario quiere cubrir, por lo que son aprobadas.
- ✓ La presentación del login funciona perfectamente, por lo que la pantalla inicial ha sido aprobada.

Anexo 3: Acta de reunión Sprint 2

Fecha: 10 de septiembre de 2019

Objetivos

- Presentación del segundo Sprint.
- Presentar la pantalla de la administración de usuarios.

Participantes

- Mauricio Mera
- Luis Serrano

Temas tratados

1. Presentación de la página de la administración de usuarios.
 - Se puede agregar un nuevo usuario.
 - Se puede editar un usuario.
 - Se puede eliminar un usuario.
 - A los usuarios se le puede asignar un rol, tipo Administrador, Profesor o Estudiante.
 - Los campos requeridos tienen color rojo hasta que se ingrese la información.

Resultados de la presentación

- ✓ Las pruebas del ingreso de nuevos usuarios fueron correctas y aprobadas.
- ✓ Las pruebas de modificación de usuarios fueron correctas y aprobadas.
- ✓ Las pruebas de eliminación de usuarios fueron correctas y aprobadas.
- ✓ La asignación de rol es correcta y la visualización de la información dependiendo del rol es correcta.

Anexo 4: Acta de reunión Sprint 3

Fecha: 18 de septiembre de 2019

Objetivos

- Presentación del Sprint 3.
- Presentar la administración de las actividades que van a tener los cursos.

Participantes

- Mauricio Mera
- Luis Serrano

Temas tratados

1. Presentación de la página de administración de actividades.
 - Se puede agregar una actividad.
 - Se puede modificar una actividad.
 - Se puede eliminar una actividad.
 - La lista de actividades se visualizará cuando se vaya a agregar un curso.

Resultados de la presentación

- ✓ Las pruebas de ingreso de una actividad fueron correctas y aprobadas.
- ✓ Las pruebas de modificación de una actividad fueron correctas y aprobadas.
- ✓ Las pruebas de eliminación de una actividad fueron correctas y aprobadas.
- ✓ Estas actividades serán una lista que se presentarán cuando se vaya a registrar un curso.

Anexo 5 Acta de reunión Sprint 4

Fecha: 6 de octubre de 2019

Objetivos

- Presentación del cuarto Sprint.
- Presentar la administración de cursos.

Participantes

- Mauricio Mera
- Luis Serrano

Temas tratados

1. Presentación de la página de administración de cursos.
 - Se puede agregar un curso.
 - Se puede editar un curso.
 - Se puede eliminar un curso.
 - En un curso se debe seleccionar un horario específico para que posteriormente se pueda presentar el horario a los profesores y estudiantes.
 - Cada curso tiene un profesor asignado para que el profesor seleccionado pueda ver sus cursos asignados.
 - En cada curso se puede visualizar la lista de asistentes.

Resultados de la presentación

- ✓ Las pruebas de ingreso de un curso nuevo fueron correctos y aprobados.
- ✓ Las pruebas de modificación de un curso fueron correctos y aprobados.
- ✓ Las pruebas de eliminación de un curso fueron correctos y aprobados.
- ✓ La visualización de los horarios del curso es correcta.

- ✓ La visualización de los asistentes al curso es correcta, se visualizan asistentes ingresados por script ya que esta información se llenará cuando el estudiante reserve el curso.

Anexo 6: Acta de reunión Sprint 5

Fecha: 20 de octubre de 2019

Objetivos

- Presentación del Sprint 5.
- Presentar el login del móvil.
- Presentar las opciones con las que va a trabajar el perfil profesor.

Participantes

- Mauricio Mera
- Luis Serrano

Temas tratados

1. Presentar la pantalla del login de la aplicación móvil Android.
 - Se utiliza los mismos servicios que el login de la parte web por lo que su funcionamiento es similar.
2. Presentación de las pantallas que puede administrar un profesor.
 - Se puede registrar los pagos que realicen los estudiantes.
 - Puede visualizar los cursos en los que impartirá clases.

Resultados de la presentación

- ✓ Las pruebas de login fueron correctos y aprobados.
- ✓ Las pruebas de la visualización de cursos fueron correctos y aprobados.
- ✓ Las pruebas del registro del pago de las mensualidades del estudiante fueron correctas y aprobadas.

Anexo 7: Acta de reunión Sprint 6

Fecha: 1 de noviembre de 2019

Objetivos

- Presentación del Sprint 6.
- Presenta las funciones que debe realizar el perfil estudiante.

Participantes

- Mauricio Mera
- Luis Serrano

Temas tratados

1. Presentación de la pantalla de visualización de mensualidades.
 - Es estudiante puede visualizar el estado del pago de sus mensualidades.
2. Presentación de la pantalla de reserva de cursos.
 - El estudiante puede visualizar los cursos disponibles de la semana.
 - Se puede visualizar las actividades de cada curso.
 - Se puede visualizar los horarios de cada curso para validar la disponibilidad.
 - Se puede reservar el curso deseado.

Resultados de la presentación

- ✓ Se prueba la visualización de los datos del pago del estudiante correctamente y aprobado.
- ✓ Se prueba la visualización de la información de los cursos disponibles correctamente y aprobado.
- ✓ Se prueba el registro de los cursos correctamente y sin errores.

