Universitat Oberta de Catalunya

uoc.edu

Diseño de una aplicación móvil para la gestión del campeonato de fútbol de los colegios de educación media de la ciudad de Esmeraldas.

Nombre Estudiante: Kléber Posligua Flores
Máster universitario de Desarrollo de Aplicaciones para Dispositivos Móviles
Trabajo final de máster DADM

Nombre Consultor/a: Pau Dominkovics Coll
Profesor/a responsable de la asignatura: Carles Garigues Olivella

Fecha Entrega: 30-12-2020

Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada 3.0 España de Creative Commons.

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Diseño de una aplicación móvil para la gestión del campeonato de fútbol de los colegios de educación media de la ciudad de Esmeraldas.*
Nombre del autor:	*Kléber Posligua Flores*
Nombre del consultor/a:	*Pau Dominkovics Coll*
Nombre del PRA:	*Carles Garigues Olivella*
Fecha de entrega (mm/aaaa):	12/2010
Titulación::	*Máster universitario de Desarrollo de Aplicaciones para Dispositivos Móviles*
Área del Trabajo Final:	*Trabajo final de máster DADM*
Idioma del trabajo:	*Español*
Palabras clave	*Gestión, Fútbol, Android*

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo.*

Anualmente los colegios de educación media de la ciudad de Esmeraldas (Ecuador), celebran un campeonato con motivo de confraternizar entre colegas docentes; el presente proyecto de investigación se ha desarrollado con el objetivo principal de ayudar a organizar este campeonato.

Primeramente, se ha considerado el estudio de las aplicaciones existentes, analizando sus pros y contras, luego basado en el diseño centrado en el usuario se han determinado los escenarios de uso y el diseño de las interfaces de usuario. A continuación, se ha desarrollado un prototipo de la aplicación con sus funcionalidades más importantes, la misma que sirvió como base para la aplicación final.

Como resultado, se tiene una aplicación móvil, que permite registrar las instituciones educativas y la categoría deportiva en la que participa, el ingreso de jugadores con sus respectivas fotos, el registro de fechas de juegos y resultados. La aplicación automáticamente genera la tabla de posiciones por etapas en las distintas categorías deportivas. Además, permite a los usuarios consultar las fechas de juegos y resultados de los encuentros deportivos.

Se concluye que, la elaboración de este proyecto de software brinda un enfoque global sobre la gestión de un proyecto de software, desde la fase de planificación, hasta la puesta en marcha de la aplicación.

La elaboración de un prototipo proporciona una idea general de las funcionalidades y la interacción que el usuario realizará sobre la aplicación,

pero es necesario considerar las limitaciones de los dispositivos en los que va a funcionar, para que la implementación sea viable.

Abstract (in English, 250 words or less):

Every year the high schools of the city of Esmeraldas (Ecuador), celebrate a championship with the purpose of fraternizing among teaching colleagues; the present research project has been developed with the main objective of helping to organize this championship.

First, it has been considered the study of existing applications, analyzing their pros and cons, then based on the user-centered design have been determined use scenarios and design of user interfaces. Then, a prototype of the application with its most important functionalities has been developed, which served as the basis for the final application.

As a result, we have a mobile application, which allows the registration of educational institutions and the sports category in which they participate, the entry of players with their respective photos, the registration of game dates and results. The application automatically generates the table of positions by stages in the different sports categories. In addition, it allows users to consult the dates of games and results of sports meetings.

It is concluded that, the development of this software project provides a global approach to the management of a software project, from the planning phase, to the implementation of the application.

The elaboration of a prototype provides a general idea of the functionalities and the interaction that the user will perform on the application, but it is necessary to consider the limitations of the devices on which it will work, so that the implementation is viable.

Dedicatoria

A mis padres Pedro y Máxima, por su esfuerzo incansable y fructífero. A mis hijos Kémilly, Klarisse, Kléber y Kévin, por formar parte de mi vida. A mi esposa Kelly, por su apoyo incondicional.

Índice

1. Introducción ... 1
1.1 Contexto y justificación del Trabajo ... 1
1.2 Objetivos del Trabajo ... 4
1.3 Enfoque y método seguido ... 5
1.4 Planificación del Trabajo ... 6
1.5 Breve sumario de productos obtenidos ... 8
1.6 Breve descripción de los otros capítulos de la memoria ... 8
2. Diseño ... 10
2.1 Usuarios y contexto de uso ... 10
2.2 Diseño conceptual ... 11
2.3 Árbol de navegación ... 12
2.4 Prototipado ... 13
2.5 Evaluación ... 21
2.6 Definición de casos de uso ... 22
2.7 Diseño de la arquitectura ... 27
2.8 Diseño gráfico ... 32
2.9 Nombre de la APP ... 32
2.10 Lenguaje de la aplicación ... 33
3. Implementación ... 34
3.1 API RESTful ... 34
3.2 Backend ... 35
3.3 Desarrollo de la app ... 36
3.4 Pruebas ... 36
4. Conclusiones ... 39
4.1 Conclusiones ... 39
4.2 Consecución de los objetivos planteados ... 39
4.3 Análisis de planificación ... 39
4.4 Líneas de trabajos futuros ... 39
5. Glosario ... 41
6. Bibliografía ... 42
7. Anexos ... 44

Índice de figuras

Figura 1. Diagrama de Gantt 7
Figura 2. Árbol de navegación 12
Figura 3. Botón de interacción con el prototipo 13
Figura 4. Prototipo - Screen splash y Login 14
Figura 5. Prototipo - Nuevo usuario 14
Figura 6. Prototipo - Menú principal 15
Figura 7. Prototipo - Registro de equipos 15
Figura 8. Prototipo - Registro de jugadores 16
Figura 9. Prototipo - Fecha de juegos y Resultados 16
Figura 10. Prototipo – Registro de amonestaciones y menú de barra lateral (registro de información) 17
Figura 11. Prototipo - Nuevo usuario administrador y notificaciones 18
Figura 12. Prototipo - formato de campeonato y barra lateral menú configuración 18
Figura 13. Prototipo - Reportes resultados y tablas de posiciones 19
Figura 14. Prototipo - Reportes fechas de juegos y amonestados 20
Figura 15. Prototipo - Barra lateral (menú reportes) 20
Figura 16. Formato para cargas masivas de las fechas a jugarse 22
Figura 17. Casos de Uso 26
Figura 18. Diagrama correspondiente al diseño de la base de datos 27
Figura 19. Diagrama UML diseño de clases 28
Figura 20. Autenticación basada en cookies 29
Figura 21. Diagrama de secuencias - Guardar Equipo 30
Figura 22. Diagrama de secuencias - tabla de posiciones 31
Figura 23. Diagrama de secuencias - configura notificaciones 31
Figura 24. Logotipo de la aplicación 32
Figura 25. Backend - registro de fechas de juegos y resultados 36

Índice de tablas

Tabla 1. Aplicación Winner - Pros y Contras 2
Tabla 2- Aplicación Mis tornes - Pros y Contras 3
Tabla 3. Aplicación Copa fácil - Pros y Contras 3
Tabla 4. Plan de trabajo 6
Tabla 5. Presupuesto - Talento humano 8
Tabla 6. Presupuesto - Recursos 8
Tabla 7. Caso de Uso N° 1 22
Tabla 8. Caso de Uso N° 2 23
Tabla 9. Caso de Uso N° 3 24
Tabla 10. Caso de Uso N° 4 24
Tabla 11. Caso de Uso N° 5 25
Tabla 12. Caso de Uso N° 6 25
Tabla 13. Caso de Uso N° 7 26
Tabla 14. Modelo rutas API RESTful 30
Tabla 15. Resumen de peticiones HTTP 35
Tabla 16. Prueba - Registro de instituciones 37

Tabla 17. Prueba - Registro de jugadores 37
Tabla 18. Prueba - Consulta de resultados 38
Tabla 19. Prueba - Consulta de tabla de posiciones 38

1. Introducción

1.1 Contexto y justificación del Trabajo

Las Instituciones educativas de nivel medio de la ciudad de Esmeraldas (Ecuador), anualmente organizan actividades deportivas, con el propósito de confraternizar entre colegas docentes. Estas actividades se organizan en un campeonato, que inicia en el mes de agosto y se extiende hasta mediados del mes de diciembre. El año pasado se disputó la edición número 39 de este campeonato, por tanto, se ha convertido en una tradición esperada, que genera espacios de camaradería, recreación y salud.

Las actividades deportivas están agrupan en varias disciplinas, como: fútbol para varones mayores de 40 años, indor fútbol para damas y varones, y voleibol para damas. Actualmente, la organización y control de los eventos deportivos se realiza de manera manual con la ayuda de comisiones deportivas -integradas por docentes de las instituciones participantes-, es decir que, las incidencias de los juegos, novedades, próximas fechas, resultados, sanciones, informes, etcétera, se registran en papeles, para luego guardarlos en carpetas; lo que implica para los organizadores, aportar con un tiempo considerable mantener al día la gestión del torneo; otro problema no menos importante es no contar con respaldos digitalizados de la información generada.

De acuerdo con este contexto, se necesita gestionar de manera automatizada las actividades que la organización del campeonato conlleva. La aplicación móvil que se propone diseñar, pretende facilitar las tareas de las distintas comisiones de la siguiente manera:

- En la comisión de disciplina, llevar el control de jugadores sancionados: tarjetas rojas, amarillas, tiempo de suspensión y notificación automática de los jugadores sancionados a los dirigentes de cada equipo.
- En la comisión técnica y de fichaje: formulario de inscripción y habilitación de nuevos jugadores, registro de resultados de cada fecha, generación automática de tabla de posiciones, tabla de goleadores, notificar a cada equipo sobre los resultados y próximas fechas, entre otros.
- En la comisión de prensa y propaganda: enviar los resultados de forma automática a los medios de prensa, y las opciones necesarias para publicar en las redes sociales como Facebook y Twitter.
- Y por supuesto, acceso a toda la información generada por la aplicación, para las autoridades principales del campeonato, como presidente, vicepresidente, secretario/a, tesorera/a etc.

La aplicación móvil será desarrollada para dispositivos Android y para la persistencia de datos, se utilizará el servidor de base de datos MySql. Esta base de datos será publicada utilizando los servicios de Digital Ocean. Además, en este servidor, también se implementarán las API RESTful necesarias para la manipulación de los datos, que serán consumidos por la app móvil.

Por otra parte, luego de revisar en la tienda de Google Play, se pudo comprobar que existen varias aplicaciones móviles en el mercado (Winner, Mis torneos, Copa Fácil, Tournament Manager, Virtual Competition, entre otras), unas dedicadas exclusivamente a pronósticos deportivos, otras a informar de resultados, otras que gestionan torneos; pero solo consideran la creación del formato de juegos (enfrentamiento entre equipos). La app propuesta en este trabajo, considera -aparte de la difusión de resultados- ayudar en la gestión y organización del torneo, brindando a cada una de las comisiones organizadoras, las herramientas adecuadas para automatizar las tareas que tienen asignadas. Otro dato importante a mencionar, es que, permitirá la personalización de los jugadores con su propia fotografía; y será gratuita y sin anuncios comerciales.

Las tablas siguientes muestran el análisis realizado:

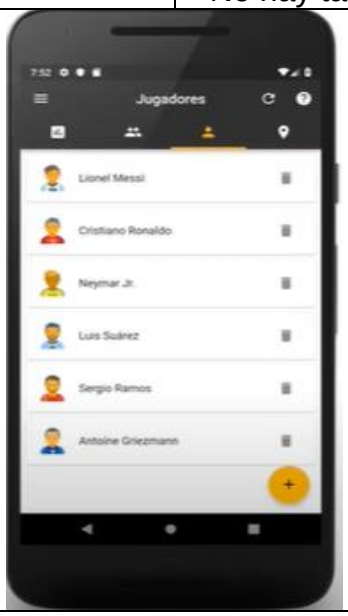
a) Nombre: Winner	
Pros	Contras
- Permite crear equipos y jugadores - Permite configurar el calendario de juegos - Permite compartir con amigos	- No personaliza las fotos de los jugadores. - No permite establecer responsabilidades por grupos de personas (comisiones). - No hay tabla de posiciones.
	

Tabla 1. Aplicación Winner - Pros y Contras

b) Nombre: Mis torneos	
Pros	Contras
- Permite compartir con amigos. - Permite controlar estadísticas como puntajes, goleadores. - Permite configurar el formato del campeonato.	- Contiene anuncios publicitarios. - En ciertas ocasiones muestra mensajes de error de conexión. - Las opciones de menú, son botones que ocupan toda la pantalla. - No permite agregar la foto del jugador.

Tabla 2- Aplicación Mis tornes - Pros y Contras

c) Nombre: Copa fácil	
Pros	Contras
- Interfaz intuitiva y fácil de usar. - Registro de jugadores y equipos. - Estadísticas variadas del tornero.	- Luego de la versión de prueba, solicita tus datos financieros. - Permite pocos jugadores en la versión de prueba. - No permite personalizar la fotografía del jugador.

Tabla 3. Aplicación Copa fácil - Pros y Contras

Del análisis realizado a las apps existentes en el mercado, se pueden extraer las siguientes conclusiones:

a) Existen varias aplicaciones que permiten organizar torneos, informar de resultados y otras para pronósticos deportivos; sin embargo, no se ajustan a las funcionalidades exigidas para la app del presente proyecto.
b) En las apps analizadas, los mensajes comerciales, invitan indirectamente al usuario a desinstalar la aplicación. La app de este proyecto será diseñada sin anuncios comerciales, ya que no tiene fines de lucro y, por tanto, será ofrecida de forma gratuita.
c) La posibilidad de personalizar la fotografía del jugador en la app de este proyecto, será una ventaja para destacar sobre las apps analizadas.

1.2 Objetivos del Trabajo

A continuación, se enumeran los objetivos propuestos en este trabajo de fin de master:

Objetivos de la aplicación:

- Diseñar una aplicación móvil basada en la plataforma Android para apoyar en la gestión del campeonato deportivo.
- Implementar todas las funcionalidades propuestas para lograr un óptimo rendimiento de la aplicación.

Objetivos del trabajo:

- Generar los prototipos de bajo y alto nivel para verificar que la aplicación cuenta con las características específicas planteadas.
- Diseñar una interfaz móvil de fácil e intuitivo manejo para todos los usuarios de la aplicación.
- Implementar las notificaciones automáticas para que los usuarios estén al tanto de resultados y detalles del campeonato.
- Enviar notificaciones automáticas en las redes sociales como Facebook y Twitter para promocionar el campeonato.

Objetivos personales:

- Aplicar todos los conocimientos adquiridos en este master mediante el desarrollo de una aplicación móvil.
- Desarrollar nuevas habilidades en el campo de la gestión de proyectos de aplicaciones móviles.

Requisitos funcionales:

- Incluir procedimientos de autorización de usuarios mediante credenciales tipo usuario/contraseña.
- Permitir a la comisión de disciplina el registro y control de jugadores sancionados.
- Permitir a la comisión de prensa y propaganda, el envío de novedades del campeonato a los medios de prensa y a las redes sociales.
- Permitir a la comisión técnica y de fichaje, el registro de equipos, jugadores (fotografía incluida), resultados y próximas fechas.

- Informar a los usuarios sobre resultados, sanciones, próximas fechas, tabla de goleadores, tablas de posiciones.
- Notificar automáticamente a los miembros de cada equipo sobre las novedades del torneo.
- Guardar la información histórica generada por la aplicación.
- Generar la API RESTful para compartir información con otros aplicativos.

Requisitos no funcionales:

- La aplicación de asegurar que los datos estén protegidos de accesos no autorizados.
- La navegación y el tiempo de respuestas deben ser rápidos para evitar tiempos de espera prolongados al usuario.
- La interfaz de la aplicación debe ser intuitiva y amigable para facilitar la navegación.

1.3 Enfoque y método seguido

Para garantizar el avance de la aplicación, se propone la utilización de código existente en internet y que ya ha sido probado su funcionalidad, es decir, se plantea aprovechar las comunidades de desarrollo de aplicaciones móviles para dispositivos Android. De las aplicaciones similares existentes en el mercado se pretende utilizar como guía, la forma de organizar el modelo de campeonato.

Para gestionar proyectos de software existen muchos métodos de desarrollo, cada uno con sus ventajas y desventajas. De acuerdo con las características del presente trabajo, el modelo waterfall (cascada) es el que mejor se adapta a las necesidades de la aplicación propuesta, debido a la secuencialidad de las fases que se deben cumplir como la planificación, los tiempos, fechas límites y presupuesto. Sin embargo, se pretende que el método elegido no sea una camisa de fuerza; más bien utilizar los puntos fuertes de otras metodologías de acuerdo a la fase en que vaya el proyecto, por ejemplo, utilizar el método “desarrollo rápido de aplicaciones” para entregas que requieren tiempos cortos; o la metodología ágil, para priorizar más el software que la documentación exhaustiva.

Las fases que se utilizarán serán las siguientes:

- Planificación, que incluyen los hitos del proyecto que ayudarán a medir su avance.
- Análisis y diseño, para obtener y clasificar los requisitos del sistema; estructurar el software, asignar recursos.
- Desarrollo, para realizar la codificación, pruebas unitarias, documentar el código.
- Pruebas de funcionamiento, pruebas en emuladores y dispositivos reales.
- Entrega, elaboración de manuales y distribución de la aplicación.

1.4 Planificación del Trabajo

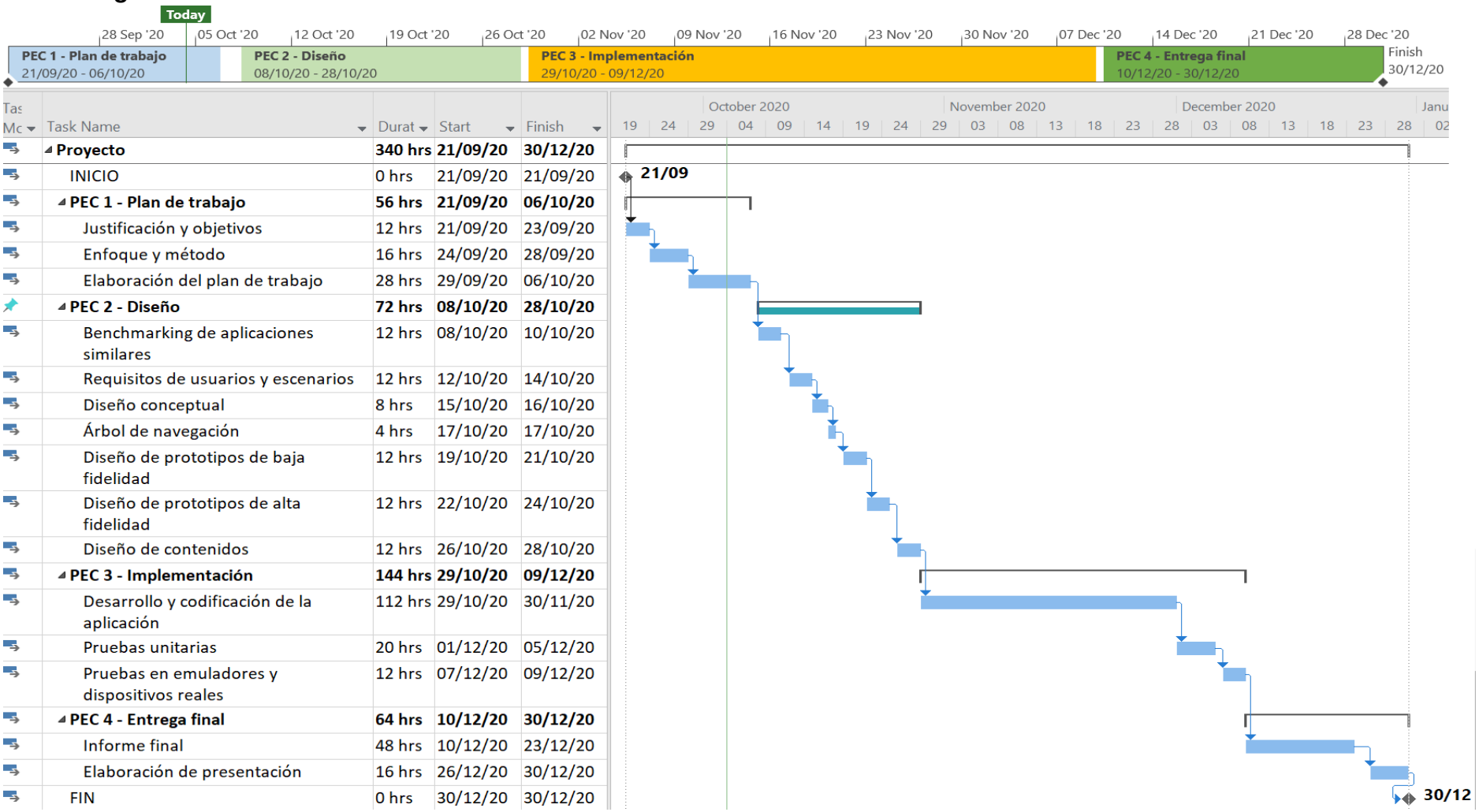
El proyecto inicia el 16 de septiembre de 2020 y se extiende hasta el 30 de diciembre de 2020, coincidiendo estas fechas con el principio y fin de la programación semestral de la asignatura 'Trabajo de fin de master'. Se dedicarán 4 horas diarias en días laborables y 2 horas diarias en días festivos. En el calendario se ha considerado trabajar los días sábados un mínimo de 4 horas; y el domingo solo en el caso que haya retraso en la consecución de los hitos.

A continuación, se muestra en horas, la planificación detallada:

Hito	Duración	Fecha inicial	Fecha final
PEC1 - Plan de trabajo	**56 horas**	**21/09/2020**	**07/10/2020**
Justificación y objetivos	12 horas	21/09/2020	23/09/2020
Enfoque y método	16 horas	24/09/2020	28/09/2020
Elaboración del plan de trabajo	28 horas	29/09/2020	06/10/2020
PEC2 - Diseño	**72 horas**	**08/10/2020**	**28/10/2020**
Benchmarking de aplicaciones similares	12 horas	08/10/2020	10/10/2020
Requisitos de usuarios y escenarios	12 horas	12/10/2020	14/10/2020
Diseño conceptual	8 horas	15/10/2020	16/10/2020
Árbol de navegación	4 horas	17/10/2020	17/10/2020
Diseño de prototipos de baja fidelidad	12 horas	19/10/2020	21/10/2020
Diseño de prototipos del alta fidelidad	12 horas	22/10/2020	24/10/2020
Diseño de contenidos	12 horas	26/10/2020	28/10/2020
PEC3 - Implementación	**140 horas**	**29/10/2020**	**09/12/2020**
Desarrollo y codificación de la aplicación	112 horas	29/10/2020	30/11/2020
Pruebas unitarias	20 horas	01/12/2020	05/12/2020
Pruebas en emuladores y dispositivos reales	8 horas	07/12/2020	08/12/2020
PEC4 - Entrega final	**64horas**	**10/12/2020**	**30/12/2020**
Informe final	48 horas	10/12/2020	23/12/2020
Elaboración de presentación	16 horas	26/12/2020	30/12/2020

Tabla 4. Plan de trabajo

1.4.1. Diagrama de Gantt



Task Name	Durat	Start	Finish
Proyecto	**340 hrs**	**21/09/20**	**30/12/20**
INICIO	0 hrs	21/09/20	21/09/20
PEC 1 - Plan de trabajo	**56 hrs**	**21/09/20**	**06/10/20**
Justificación y objetivos	12 hrs	21/09/20	23/09/20
Enfoque y método	16 hrs	24/09/20	28/09/20
Elaboración del plan de trabajo	28 hrs	29/09/20	06/10/20
PEC 2 - Diseño	**72 hrs**	**08/10/20**	**28/10/20**
Benchmarking de aplicaciones similares	12 hrs	08/10/20	10/10/20
Requisitos de usuarios y escenarios	12 hrs	12/10/20	14/10/20
Diseño conceptual	8 hrs	15/10/20	16/10/20
Árbol de navegación	4 hrs	17/10/20	17/10/20
Diseño de prototipos de baja fidelidad	12 hrs	19/10/20	21/10/20
Diseño de prototipos de alta fidelidad	12 hrs	22/10/20	24/10/20
Diseño de contenidos	12 hrs	26/10/20	28/10/20
PEC 3 - Implementación	**144 hrs**	**29/10/20**	**09/12/20**
Desarrollo y codificación de la aplicación	112 hrs	29/10/20	30/11/20
Pruebas unitarias	20 hrs	01/12/20	05/12/20
Pruebas en emuladores y dispositivos reales	12 hrs	07/12/20	09/12/20
PEC 4 - Entrega final	**64 hrs**	**10/12/20**	**30/12/20**
Informe final	48 hrs	10/12/20	23/12/20
Elaboración de presentación	16 hrs	26/12/20	30/12/20
FIN	0 hrs	30/12/20	30/12/20

Figura 1. Diagrama de Gantt

1.4.2 Presupuesto

Talento humano:

Actividad	Dedicación	Valor en dólares
Gestor de proyectos	3 meses	$ 350.00
Diseñador gráfico	1 mes	$ 200.00
Desarrollador de software	1 mes	$ 350.00
TOTAL		**$ 900.00**

Tabla 5. Presupuesto - Talento humano

Recursos:

Item	Valor en dólares
Computador	$ 800.00
Software Android Studio	$ 0.00
Herramientas wireframes	$ 0.00
Emuladores	$ 0.00
Hosting para la parte del servidor	$ 100.00
TOTAL	**$ 900.00**

Tabla 6. Presupuesto - Recursos

Presupuesto total:
Mil ochocientos dólares americanos ($ 1800.00)

1.5 Breve sumario de productos obtenidos

El presente trabajo de fin de master, generará los siguientes productos:

- Una aplicación Android completamente operativa y funcional.
- Servicio API RESTful que pueda ser consumido por la app desarrollada, y por cualquier software que desee consultar datos.
- La memoria del trabajo de fin de máster.
- Diapositivas y videos para la defensa del TFM

1.6 Breve descripción de los otros capítulos de la memoria

En el capítulo 2, se realizará el análisis y diseño de la aplicación móvil mediante la extracción de los requisitos, con la finalidad de realizar un diseño centrado en el usuario. Se trabajará en el flujo de la información que servirá como base para construir el árbol de navegación. Se diseñarán los wireframes para estructurar las pantallas de la aplicación móvil.

En el capítulo 3, se realizará la implementación de la aplicación, empezando con la codificación, continuando con las pruebas para verificar que la aplicación cumple con todas las funcionalidades solicitadas.

En el capítulo 4, se elaborarán las conclusiones del trabajo. También se valorará si se ha podido cumplir con los objetivos del proyecto, y con la

planificación propuesta. Así mismo, se propondrán mejoras que se pueden realizar a la aplicación móvil; estas servirán de base para trabajos futuros.

2. Diseño

2.1 Usuarios y contexto de uso

Características de los usuarios de la App

Los potenciales usuarios para la aplicación que será desarrollada en este proyecto, son los miembros de las instituciones educativas de nivel medio de la ciudad de Esmeraldas. Este sería el grupo de usuarios principales, conformados por docentes de planta y contratados, empleados administrativos y el personal de servicio; que utilizarían la aplicación para gestionar el campeonato (miembros de las distintas comisiones), y otros, para mantenerse informado de los resultados e incidencias de los juegos realizados semana a semana. Adicionalmente, la app sería utilizada por un grupo de usuarios secundarios, entre los que destacamos: familiares de los jugadores, estudiantes, prensa local y ciudadanos en general, que quieran mantenerse informado de las novedades del campeonato.

Necesidades de los usuarios de la App

Los usuarios de la aplicación, dependiendo del rol que desempeñan en el campeonato, deben realizar actividades variadas en la app. Los usuarios pertenecientes a la comisión de disciplina, necesitan las pantallas adecuadas para registrar a jugadores sancionados; los usuarios de la comisión técnica y de fichaje necesitan ingresar, por cada categoría deportiva, los nombres de los equipos, jugadores y dirigentes, además: registrar resultados y fechas a jugarse semanalmente; los miembros de la comisión de prensa y propaganda, necesitan, dentro de la app, el espacio adecuado para informar de los resultados a los medios de difusión como la prensa local y redes sociales. Y para los usuarios en general que desean mantenerse informados, la app, debe brindar, la interfaz adecuada que les permita consultar todas las novedades del campeonato.

También es importante, que la aplicación notifique de manera automática a todos los usuarios sobre resultados, fechas a jugarse y detalles importantes del evento deportivo.

Objetivos de los usuarios de la App

El objetivo fundamental que cada usuario, está ligado directamente a la necesidad que tiene en la aplicación y al rol que desempeña en el evento deportivo. Para los usuarios principales que hacen la función directiva, su objetivo principal será: contar con una herramienta de apoyo la gestión de las tareas de organización y difusión del campeonato. Para los usuarios secundarios, el objetivo será: mantenerse informado de los pormenores del campeonato.

2.2 Diseño conceptual

Escenario de uso N° 1: El usuario desea registrar una institución participante.
El usuario abre la app e ingresa con sus credenciales y se validan sus permisos. A continuación, pulsa sobre la opción 'Registro de Instituciones', y la app mostrará la pantalla solicitando: Nombre del equipo, Nombre de la institución educativa, nombre de los delegados. Al hacer clic en 'Guardar', se valida que el registro no haya sido ingresado previamente y se procede a guardarlos en base de datos.

Escenario de uso N° 2: El usuario desea registrar los jugadores de un equipo.
El usuario abre la app e ingresa con sus credenciales y se validan sus permisos; luego pulsa sobre la opción 'Registro de jugadores', y la app mostrará la pantalla solicitando la siguiente información: Institución Educativa, disciplina deportiva en la que desea participar, apellidos, nombres, fotografía, fecha nacimiento, cédula de identidad, sexo de cada jugar participante; además nombre del director técnico. La app valida los datos y procede a guardarlos en base de datos.

Escenario de uso N° 3: El usuario desea registrar los resultados de la fecha jugada. El usuario abre la app e ingresa con sus credenciales y se validan sus permisos; luego pulsa la opción 'Registro de resultados'. La app responde con una ventana, para que seleccione la categoría deportiva; y dependiendo de la categoría seleccionada, se muestran los encuentros deportivos de la fecha en curso, listos para registrar los resultados.

Escenario de uso N° 4: El Usuario desea registrar los jugadores amonestados. El usuario abre la app e ingresa con sus credenciales y se validan sus permisos; luego selecciona la opción 'Registro de amonestados'. La app muestra la pantalla que solicita los datos siguientes: Equipo, jugador, tipo de tarjeta (amarilla o roja), y la posibilidad de notificar inmediatamente al jugador y al dirigente.

Escenario de uso N° 5: El usuario desea recibir notificaciones.
El usuario abre la app (no es necesario logearse). Luego selecciona la opción 'Configuración de notificaciones'. En esta pantalla puede configurar para recibir: notificaciones de resultados, notificación de amonestados, notificación de próximos partidos.

Escenario de uso N° 6: El usuario desea consultar los resultados.
El usuario abre la app, para esta opción, no es necesario hacer login. Luego en el menú reportes selecciona la opción 'Consulta de resultados'. Enseguida se muestran los resultados de la fecha actual, con la posibilidad de consultar fechas anteriores. La categoría fútbol se muestra por defecto, pero se puede seleccionar otra categoría deportiva.

Escenario de uso N° 7: EL usuario desea consultar la tabla de posiciones.
El usuario abre la app, no necesita autenticarse. Luego en el menú reportes selecciona la opción 'Tabla de posiciones'. Inmediatamente se muestra la tabla de posiciones con la categoría Fútbol por defecto, pero está presente la opción para seleccionar otra categoría deportiva.

2.3 Árbol de navegación

Con la finalidad de comprender los flujos de la información y la interacción del usuario con la aplicación, se expone el árbol de navegación. El árbol ha sido diseñado a partir de los apartados precedentes.

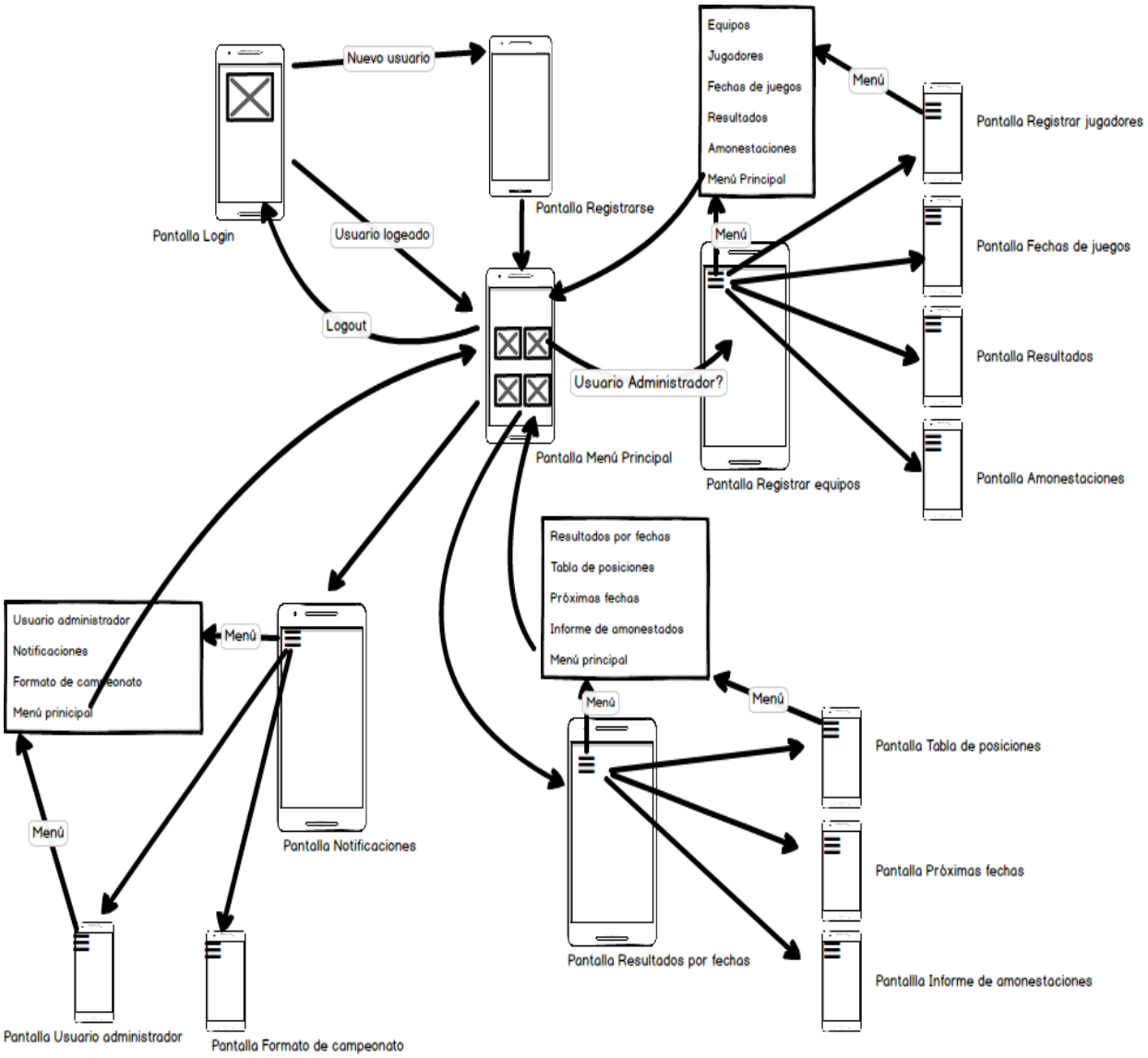


Figura 2. Árbol de navegación

Entre los patrones de diseño que utilizará la app tenemos: Una estructura estilo hub para el menú de navegación (pantalla principal) para permitir aprovechar

todo el espacio del dispositivo, presentando con imágenes las 4 opciones en una cuadrícula de forma clara y fácil de interpretar. Para acceder a los submenús disponibles en cada pantalla se utilizará un menú de barra lateral (botón hamburguesa) ya que éste solo se muestra al tocarlo y una vez seleccionada la opción, se oculta automáticamente; de esta manera no reduce/ocupa el espacio de la pantalla actual.

Opciones del menú principal:

Registros: Está opción permitirá acceder a todas las opciones de registro de datos como: ingreso de instituciones, ingreso de equipos, ingreso de resultados, ingreso de amonestados, etc.

Reportes: Permitirá consultar: resultados de juegos, próximas fechas, tablas de posiciones, amonestados.

Configuración: Permitirá al usuario decidir si quiere recibir: notificaciones de resultados, notificaciones de próximas fechas, notificaciones de amonestados. La opción 'Usuario Administrador' permitirá crear un usuario con acceso a todas las funcionalidades de la aplicación.
Aquí también se encuentra la opción 'formato de campeonato'.

Logout: Cierra la sesión del usuario.

2.4 Prototipado

Con el objetivo de agregar interactividad al prototipo, se ha utilizado la herramienta Axure RP. Se trata de herramienta muy completa, orientada a diseñar wireframes y prototipos básicos o avanzados de manera fácil, para aplicaciones móviles, web y de escritorio. A continuación, se muestra el prototipo generado:

Nota importante: En el siguiente enlace, se puede acceder al prototipo interactivo: https://1p1q6p.axshare.com . Para la interacción con los menús, utilice el botón situado en la parte superior izquierda la aplicación.

Figura 3. Botón de interacción con el prototipo

Pantallas de la aplicación:

Screen splash **Login**

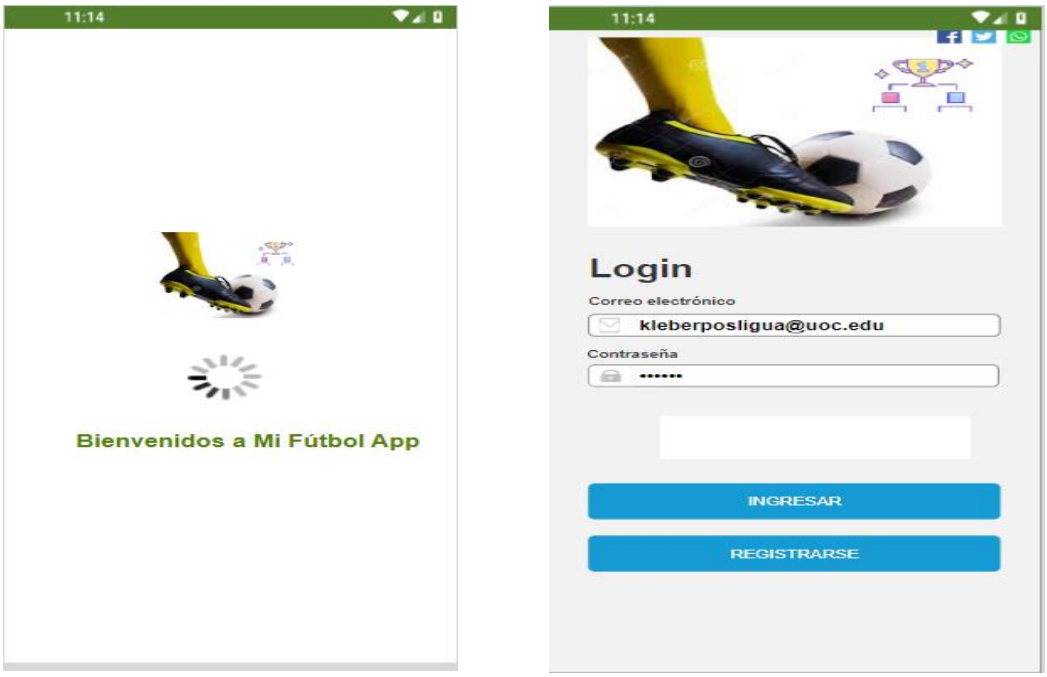


Figura 4. Prototipo - Screen splash y Login

Al momento de abrir la aplicación, se mostrará la 'screen splash' con la finalidad de dar a conocer el logotipo. Luego, se pedirán las credenciales del usuario. Se ha decido que por defecto el primero usuario administrador, sea el autor de este proyecto.

Nuevo usuario

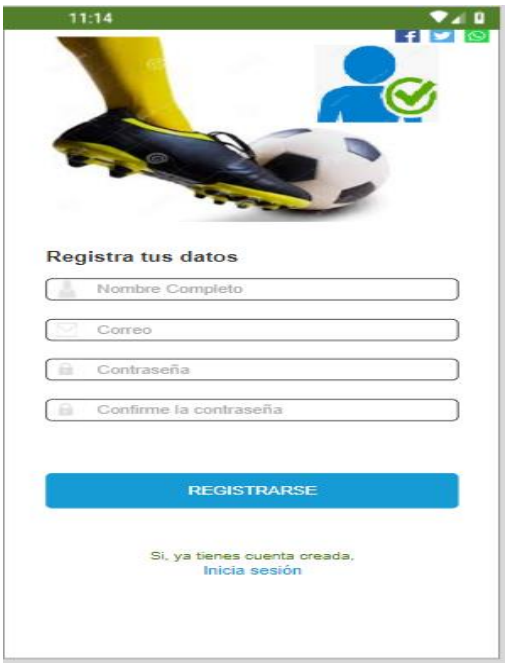


Figura 5. Prototipo - Nuevo usuario

Con el objetivo de capturar los datos del usuario, la primera vez que éste abra la aplicación, se le pedirá que se registre.

Menú principal

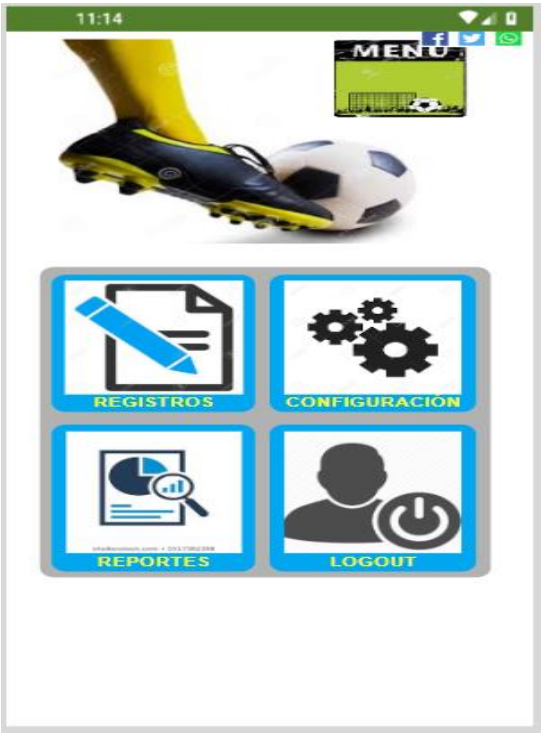


Figura 6. Prototipo - Menú principal

El menú principal tiene cuatro opciones, organizadas en forma de hub, para ocupar todo el espacio de la pantalla. Las imágenes utilizadas aparecerán como imagen del menú de barra lateral, para que el usuario sepa en todo momento en que lugar de la aplicación se encuentra.

Registro de equipos

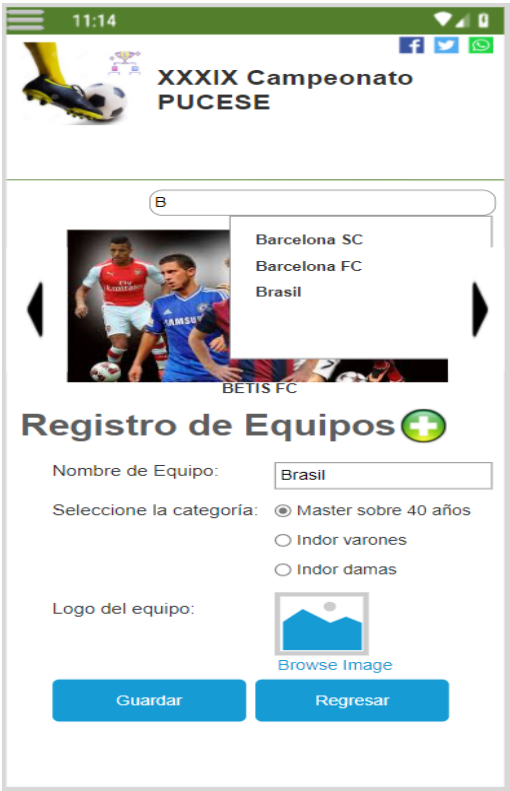


Figura 7. Prototipo - Registro de equipos

La idea del buscador, es ayudar al usuario para saber que equipos ya están registrados. Pulsando las primeras letras, se muestras los equipos registrados.
De manera gráfica, también se pueden buscar rápidamente a los equipos ingresados, con la ayuda del slider de imágenes.
Se utiliza el flat buttón de signo '+' para agregar un registro vacío, listo para recibir los datos. Los botones radio ayudan a seleccionar la categoría deportiva de manera rápida.
También se puede agregar la foto de un equipo pulsando en link buttón 'Browse Image'.
Para acceder a las opciones adicionales del menú 'Registros', se debe hacer clic en el botón ubicado en la parte superior izquierda (menú de barra lateral). Se escogió este diseño de menú, porque una vez seleccionada la opción, inmediatamente se esconde.

Registro de jugadores

Figura 8. Prototipo - Registro de jugadores

Esta pantalla, también tiene la posibilidad de buscar jugadores previamente registrados.
Adicionalmente, se puede buscar el jugador por su fotografía, con la ayuda del slider de imágenes; y al hacer clic sobre la imagen deseada, se cargan su datos.
Para agregar un registro vacio se pulsa sobre el flat button de signo '+'.
Se utiliza el mismo menú de barra lateral que la pantalla anterior.

Fecha de juegos **Resultados**

Figura 9. Prototipo - Fecha de juegos y Resultados

Con la finalidad de ingresar la información de manera cronológica, para las pantallas ‘fechas de juegos’ y ‘resultados’, se ha optado por la barra con fechas en ambos lados.

Registro de amonestaciones **Barra lateral (Registro de información)**

Figura 10. Prototipo – Registro de amonestaciones y menú de barra lateral (registro de información)

Para el registro de jugadores amonestados, se usan cuadros combinados, cargados previamente con los datos del equipo. Si se selecciona otro equipo, la lista de jugadores se actualiza. Se ha diseñado de esta manera, para evitar cometer errores al aplicar una sanción; el usuario no debe escribir la información, solo debe seleccionarla. Además, es importante mostrar un histórico de sanciones del equipo en pantalla, esa es la utilidad de la tabla, que sigue al cuadro combinado ‘tarjeta’.
Siguiendo el mismo diseño que las pantallas anteriores, se conserva el menú de barra lateral, utilizado por las razones antes indicadas.

Nuevo usuario administrador **Notificaciones**

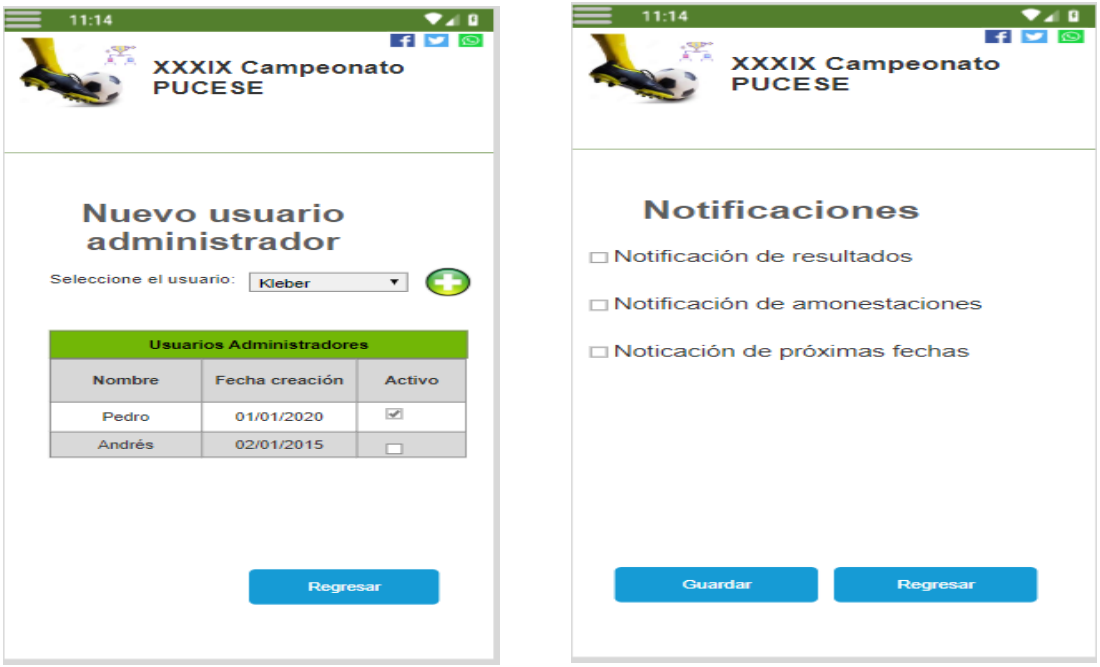


Figura 11. Prototipo - Nuevo usuario administrador y notificaciones

Dentro de la opción de configuraciones, se encuentra la opción para crear un nuevo usuario administrador. Para que el usuario realice una selección segura, se ha cargado la lista de usuarios con los nombres de los jugadores de los equipos. Luego aparece una tabla que muestra todos los usuarios administradores creados.

Las notificaciones, utilizan check box por la facilidad de interpretación de acción que debe realizar el usuario en esta pantalla: una marca para aceptar, sin marca para cancelar la notificación.

Formato de campeonato **Barra lateral (configuraciones)**

Figura 12. Prototipo - formato de campeonato y barra lateral menú configuración

En primera instancia se había pensado dejar esta pantalla para generar el formato de campeonato, pero de acuerdo al análisis realizado en el aparatado '2.5 evaluación', será reemplazado por una opción de carga masiva de datos. También se muestra el menú de barra lateral de la opción de configuraciones.

Reportes (resultados) **Reportes (tabla de posiciones)**

Figura 13. Prototipo - Reportes resultados y tablas de posiciones

Estas son opciones del menú reportes. Este diseño permite al usuario común, revisar rápida y cronológicamente los resultados y tablas de posiciones. Los cuadros combinados, listan las categorías deportivas. Si el usuario selecciona una categoría diferente, la tabla que muestra los datos, se refresca para mostrar la información relativa al elemento seleccionado.

Reportes (fechas de juegos) **Reportes (Amonestados)**

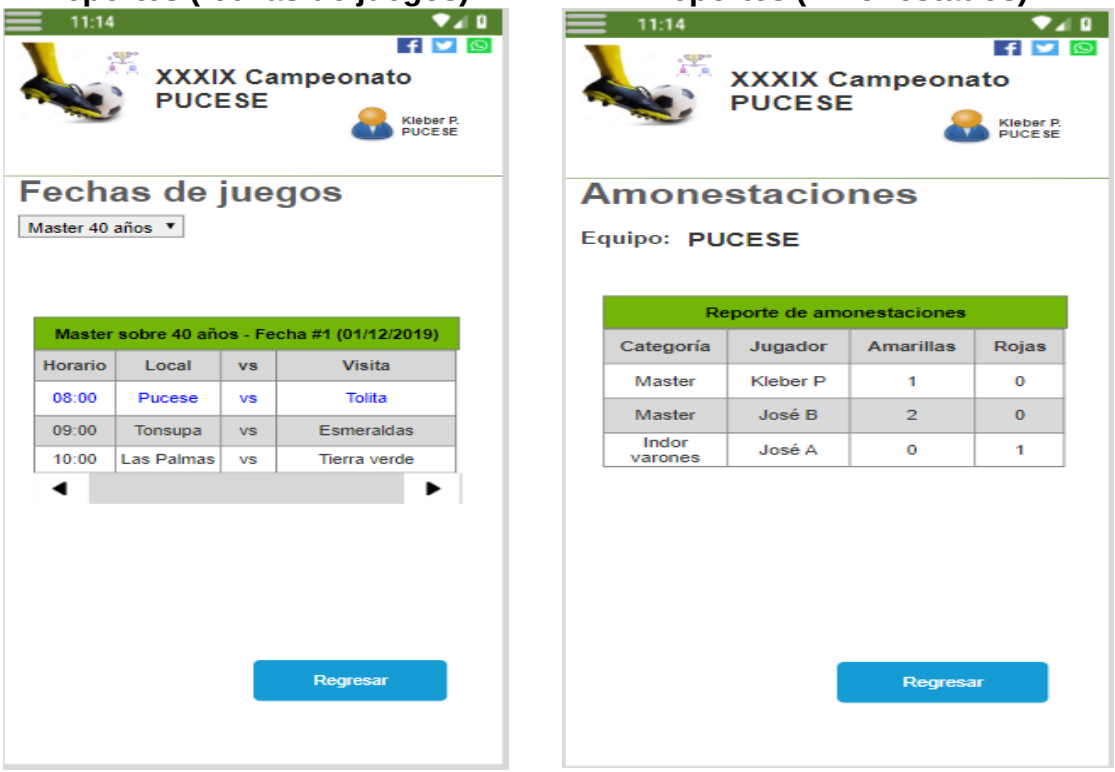
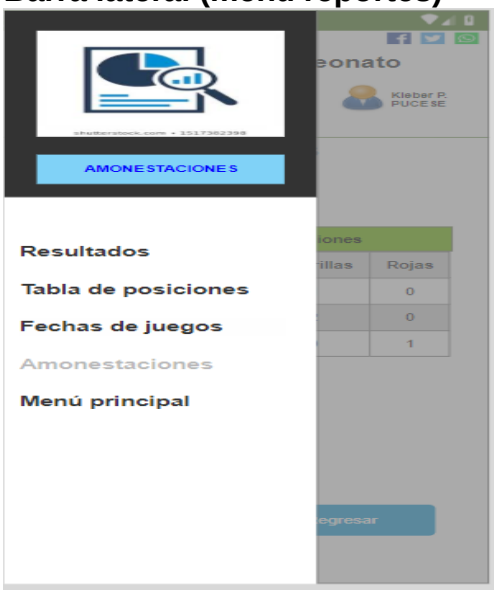


Figura 14. Prototipo - Reportes fechas de juegos y amonestados

Estas también son opciones de reportes. Por defecto, al usuario, puede ver la última fecha de juegos, pero está la posibilidad de revisar fechas anteriores y los horarios en que fueron programadas.
El reporte de amonestados, muestra la información en una tabla de manera organizada y fácil de comprender. Solo se muestran los amonestados del equipo al que pertenece el usuario actual de la aplicación.

Barra lateral (menú reportes)



Menú de barra lateral para las opciones de reportes. Hay que destacar, que la opción de este menú que actualmente está seleccionada, aparece difuminada, en relación al resto de opciones del menú.

Figura 15. Prototipo - Barra lateral (menú reportes)

2.5 Evaluación

Una vez realizado el diseño y la evaluación del prototipo, se concluye que la aplicación será intuitiva y fácil de utilizar, porque ordena correctamente los elementos en la pantalla, organiza el espacio entre ellos, y transmite correctamente la idea de la app. Los colores y fuentes utilizados, transmiten un diseño limpio y que atrae la atención del usuario. El prototipo, utiliza distintos patrones de diseño existentes, para presentar la información adecuada y eficientemente: imágenes, datos, elementos multimedia, entre otros; aprovechando las características del dispositivo seleccionado.

Para mostrar varias imágenes relativas a un tema específico se utiliza un slider de imágenes; este diseño se puede apreciar en varias pantallas de la app, porque permiten agrupar en un único sitio las imágenes que guardan relación entre sí.

Para mostrar el texto que indica una orden o mensaje para el usuario, se ha utilizado etiquetas, textos de encabezados h1, h2, h3 con los colores y tamaños apropiados para no cansar la vista del usuario. En los reportes, se usan tablas para organizar la información de manera estructurada facilitando la comprensión e interpretación del usuario.

Para las entradas de datos, se utilizan las típicas cajas de texto, algunas de éstas con una imagen en miniatura y un texto explicativo (placeholder) para que el usuario interprete la orden/mensaje rápidamente. También, cuadros combinados, botones de radio y check box, con la finalidad de ahorrar tiempo al usuario al momento de seleccionar un elemento de la pantalla.

Para confirmar las acciones, una vez registrados los datos por parte del usuario, se utilizan los típicos botones: button command, float button y link button. Para mostrar los mensajes de confirmación y/o error al momento de interactuar con la aplicación, se muestran cuadros de dialogo y mensajes que son visibles por unos instantes.

Hay un aspecto importante a destacar una vez evaluado el prototipo, se trata de la pantalla ‘formato de campeonato’, la misma que fue diseñada, considerando la modalidad todos contra todos. Pero en la práctica, considerando que no se conoce el número de equipos participantes por categoría, e incluso que, se aceptan inscripciones de instituciones fuera de plazo establecido, se opta por registrar estos datos en un formato prestablecido, que puede ser un archivo excel o csv, para posteriormente, con el script adecuado, cargarlo en forma masiva al servidor. Otra opción sería registrarlos en una pantalla de la aplicación uno por uno, que resultaría muy tedioso. A continuación, se muestra el formato del archivo para la carga masiva:

A	B	C	D	E
Categoría	**Fecha**	**Local**	**Visitante**	**Lugar**
Fulbito	14/12/2019	PUCESE	Eloy Alfaro	Colegio Luis Tello
Fulbito	14/12/2019	5 de Agosto	Atacames	Colegio Luis Tello
Fulbito	14/12/2019	Vargas Torres	Sagrado Corazón	Colegio Luis Tello
Fulbito	14/12/2019	Prado Viteri	Daniel Comboni	Colegio Luis Tello
Indor damas	14/12/2019	Margarita Cort	PUCESE	Colegio Luis Tello
Indor damas	14/12/2019	Luis Tello	Inmaculada	Colegio Luis Tello
Indor damas	14/12/2019	Vargas Torres	Prado Viteri	Colegio Luis Tello
Indor varones	14/12/2019	Sagrado Coraz	Atacames	Colegio Luis Tello
Indor varones	14/12/2019	Luis Tello	PUCESE	Colegio Luis Tello
Indor varones	14/12/2019	5 de Agosto	Vargas Torres	Colegio Luis Tello
Fulbito	21/12/2019	Vargas Torres	PUCESE	Colegio Luis Tello
Fulbito	21/12/2019	Atacames	Eloy Alfaro	Colegio Luis Tello

Figura 16. Formato para cargas masivas de las fechas a jugarse

2.6 Definición de casos de uso

A continuación, se muestran los principales casos de uso, con las secuencias de acción que lleva a cabo el usuario cuando interactúa con la aplicación.

Caso de uso	CU. 1 Registrar instituciones
Actor	Comisión técnica y de fichaje
Descripción	El usuario quiere registrar una institución participante.
Pre-condiciones	- El dispositivo móvil debe tener conexión a Internet. - Debe validarse su rol como miembro de la comisión técnica y de fichaje.
Flujo básico	1. Abrir la app: El usuario hace clic en el menú ‘registros’, luego clic en registro de instituciones. 2. Registrar la información solicitada: El usuario ingresa: nombre de equipo, nombre de la institución educativa, nombre de los delegados. 3. Guardar
Flujo alternativo	Excepción 1: No se puede conectar con el servidor No se dispone de conexión a internet o por otros motivos, no se tienen acceso al servidor. Excepción 2: La institución ya fue registrada anteriormente. La institución ya existe en la base de datos. Se muestra en pantalla el mensaje adecuado, indicando el problema ocurrido.
Post-condiciones	Se muestra el mensaje confirmando que los datos de la institución fueron registrados con éxito.
Notas	Esta es la opción por defecto del menú ‘registros’

Tabla 7. Caso de Uso N° 1

Caso de uso	CU. 2 Registrar jugadores
Actor	Comisión técnica y de fichaje
Descripción	El usuario quiere registrar los jugadores de un equipo.
Pre-condiciones	- El dispositivo móvil debe tener conexión a Internet. - Debe validarse su rol como miembro de la comisión técnica y de fichaje.
Flujo básico	1. Abrir la app: El usuario hace clic en el menú 'registros', luego clic en registro de jugadores. 2. Registrar la información solicitada: El usuario ingresa para cada jugador: Institución educativa (desde un cuadro combinado), disciplina deportiva (desde un cuadro combinado), apellidos, nombres, fotografía, fecha de nacimiento, cédula de identidad, sexo, nombre del director técnico. 3. Guardar
Flujo alternativo	Excepción 1: No se puede conectar con el servidor No se dispone de conexión a internet o por otros motivos, no se tienen acceso al servidor. Excepción 2: Jugador registrado previamente. La cédula de identidad ya fue registrada con anterioridad. Se muestra en pantalla el mensaje adecuado, indicando el problema ocurrido.
Post-condiciones	Se muestra el mensaje confirmando que los datos del jugador fueron registrados con éxito.
Notas	Esta opción está disponible desde el menú de barra lateral.

Tabla 8. Caso de Uso N° 2

Caso de uso	CU. 3 Registrar resultados
Actor	Comisión técnica y de fichaje
Descripción	El usuario quiere registrar resultados de la fecha jugada.
Pre-condiciones	- El dispositivo móvil debe tener conexión a Internet. - Debe validarse su rol como miembro de la comisión técnica y de fichaje.
Flujo básico	1. Abrir la app: El usuario hace clic en el menú 'registros', luego clic en 'resultados'. 2. Registrar la información solicitada: El usuario selecciona la categoría deportiva, y dependiendo de esta selección, debe registrar los goles del equipo local y del equipo visitante. 3. Guardar
Flujo alternativo	Excepción 1: No se puede conectar con el servidor No se dispone de conexión a internet o por otros motivos, no se tienen acceso al servidor.

	Se muestra en pantalla el mensaje adecuado, indicando el problema ocurrido.
Post-condiciones	Se muestra el mensaje confirmando que los resultados de la fecha se han guardado correctamente.
Notas	Esta opción está disponible desde el menú de barra lateral.

Tabla 9. Caso de Uso N° 3

Caso de uso	CU. 4 Registrar amonestados
Actor	Comisión de disciplina
Descripción	El usuario quiere registrar jugadores amonestados
Pre-condiciones	- El dispositivo móvil debe tener conexión a Internet. - Debe validarse su rol como miembro de la comisión de disciplina.
Flujo básico	1. Abrir la app: El usuario hace clic en el menú 'registros', luego clic en 'Amonestaciones'. 2. Registrar la información solicitada: El usuario selecciona el equipo. Dependiendo de esta selección, se muestra un cuadro combinado con la lista de jugadores. Selecciona el jugador y el tipo de tarjeta amarilla o roja. En esta pantalla también puede notificar de forma inmediata, al dirigente y al jugador sobre la sanción. 3. Guardar
Flujo alternativo	Excepción 1: No se puede conectar con el servidor No se dispone de conexión a internet o por otros motivos, no se tienen acceso al servidor. Excepción 2: Jugador registrado previamente. El jugador amonestado ya estaba registrado previamente. Se muestra en pantalla el mensaje adecuado, indicando el problema ocurrido.
Post-condiciones	Se muestra el mensaje confirmando que la amonestación ha sido guardada. Si se pulso sobre 'notificar al jugador', se envía la notificación inmediatamente.
Notas	Esta opción está disponible desde el menú de barra lateral.

Tabla 10. Caso de Uso N° 4

Caso de uso	CU. 5 Configurar notificaciones
Actor	Usuario normal
Descripción	El usuario quiere configurar las notificaciones.
Pre-condiciones	- El dispositivo móvil debe tener conexión a Internet.
Flujo básico	1. Abrir la app: El usuario hace clic en el menú 'Configuración 2. Marca la configuración deseada: El usuario puede marcar una o varias de las opciones: - Notificación de resultados - Notificación de amonestaciones - Notificación de próximas fechas

	3. Guardar
Flujo alternativo	Excepción 1: No se puede conectar con el servidor No se dispone de conexión a internet o por otros motivos, no se tienen acceso al servidor. Se muestra en pantalla el mensaje adecuado, indicando el problema ocurrido.
Post-condiciones	Se muestra el mensaje confirmando que se han guardado las notificaciones.
Notas	

Tabla 11. Caso de Uso N° 5

Caso de uso	CU. 6 Consultar Resultados
Actor	Usuario normal
Descripción	El usuario quiere consultar los resultados de los juegos.
Pre-condiciones	- El dispositivo móvil debe tener conexión a Internet.
Flujo básico	1. Abrir la app: El usuario hace clic en el menú 'Reportes' 2. Hace clic en 'Consulta de resultados': - Por defecto se muestran los resultados de la categoría Fulbito. 3. El usuario puede seleccionar otra categoría. 4. El usuario puede consultar resultados de fechas pasadas. 5. Regresar
Flujo alternativo	Excepción 1: No se puede conectar con el servidor No se dispone de conexión a internet o por otros motivos, no se tienen acceso al servidor.
Post-condiciones	
Notas	

Tabla 12. Caso de Uso N° 6

Caso de uso	CU. 7 Consultar tabla de posiciones
Actor	Usuario normal
Descripción	El usuario quiere consultar la tabla de posiciones.
Pre-condiciones	- El dispositivo móvil debe tener conexión a Internet.
Flujo básico	1. Abrir la app: El usuario hace clic en el menú 'Reportes' 2. Hace clic en 'Tabla de posiciones': - Por defecto se muestra la tabla de posiciones de la categoría Fulbito. 3. El usuario puede seleccionar otra categoría. 4. Regresar.
Flujo alternativo	Excepción 1: No se puede conectar con el servidor No se dispone de conexión a internet o por otros motivos, no se tienen acceso al servidor.

Post-condiciones	
Notas	

Tabla 13. Caso de Uso N° 7

En la gráfica siguiente, se muestran los usuarios de la aplicación y su interacción con los casos de uso:

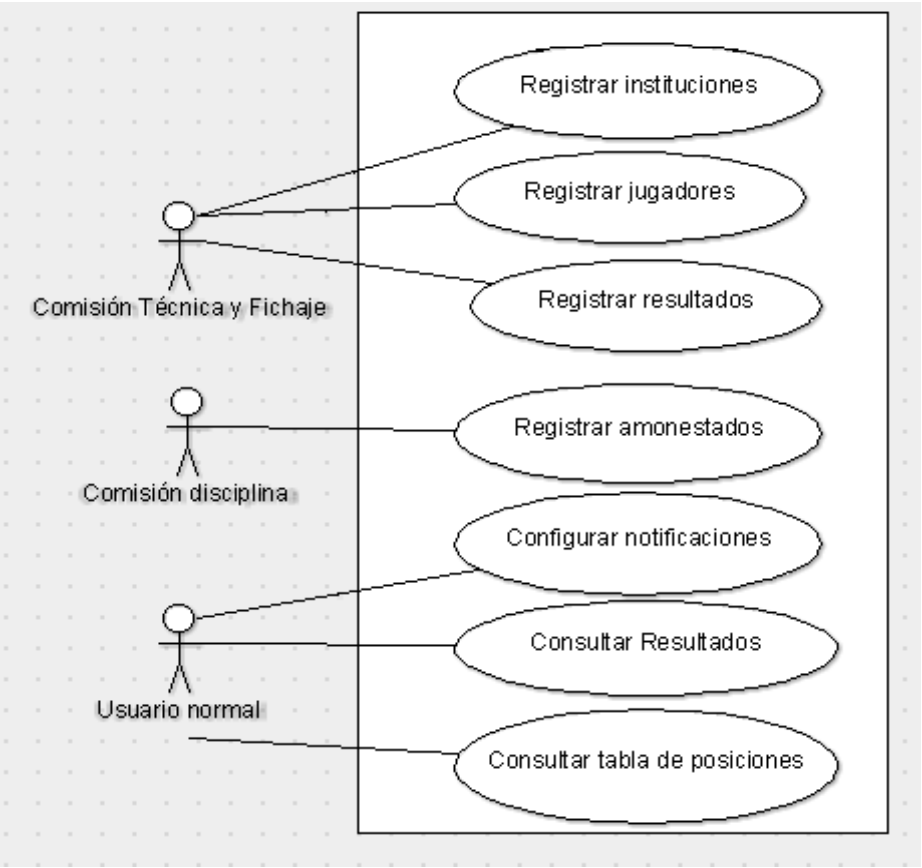


Figura 17. Casos de Uso

2.7 Diseño de la arquitectura

Herramienta de desarrollo.

La aplicación será desarrolla utilizando Flutter, por la facilidad que presenta este framework para crear aplicaciones nativas de forma fácil y rápida. A pesar que, la presente app fue pensada para trabajar en un entorno Android, con el uso de Flutter, queda abierta la posibilidad para que se ejecute sobre iOS.

Persistencia de datos y backend.

La app, guardará de forma permanente, la información procesada, en un gestor de base de datos relacional como MySql Server. Este motor es seguro, fuerte y robusto, y permite organizar la información de forma sistemática y estructura en tablas. La base de datos estará alojada en un servidor en la plataforma Digital Ocean. Para manipular los datos, se desarrollará una API RESTful, en Node Js. Este servicio también estará alojado en este servidor.

Diagrama correspondiente al diseño de la base de datos:

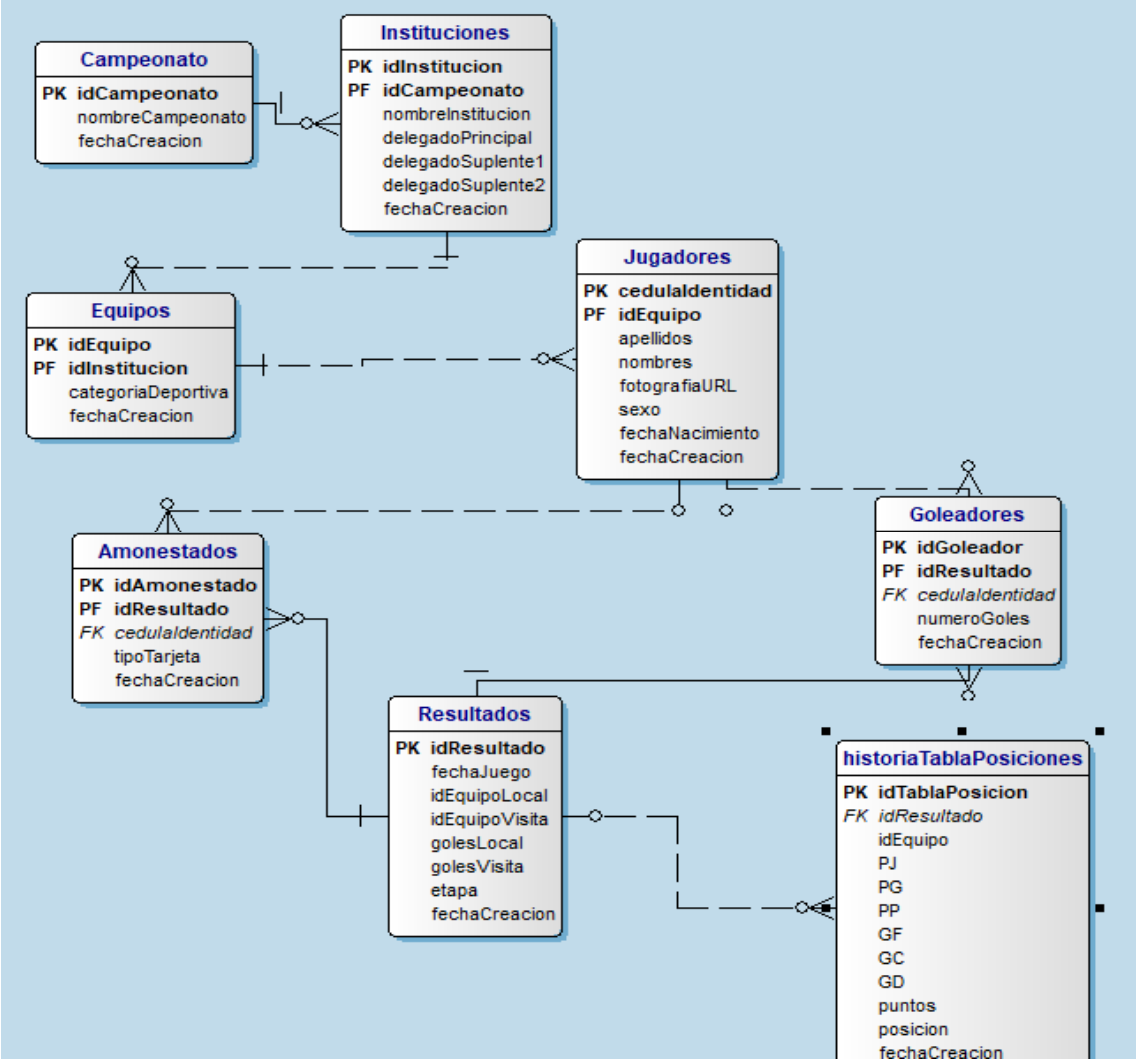


Figura 18. Diagrama correspondiente al diseño de la base de datos

Diagrama UML correspondiente al diseño de clases:

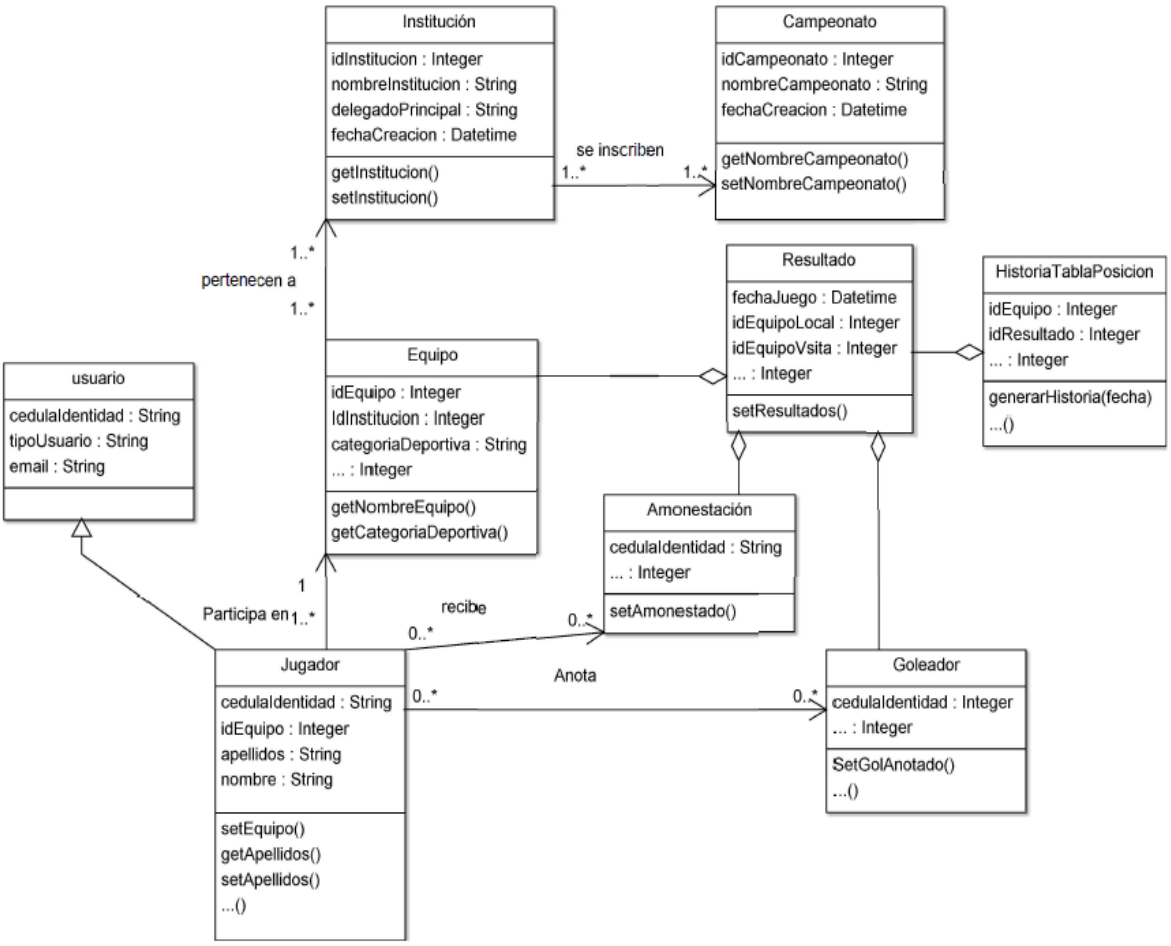


Figura 19. Diagrama UML diseño de clases

API Restful

Para la autenticación con el servidor, se utilizará la tradicional basada en cookies, en donde el servidor guarda la cookie para autenticar en cada petición; y por tanto, es necesario un almacén de sesiones, por ejemplo, en base de datos. Aunque no se descarta, usar la autenticación basada en tokens.

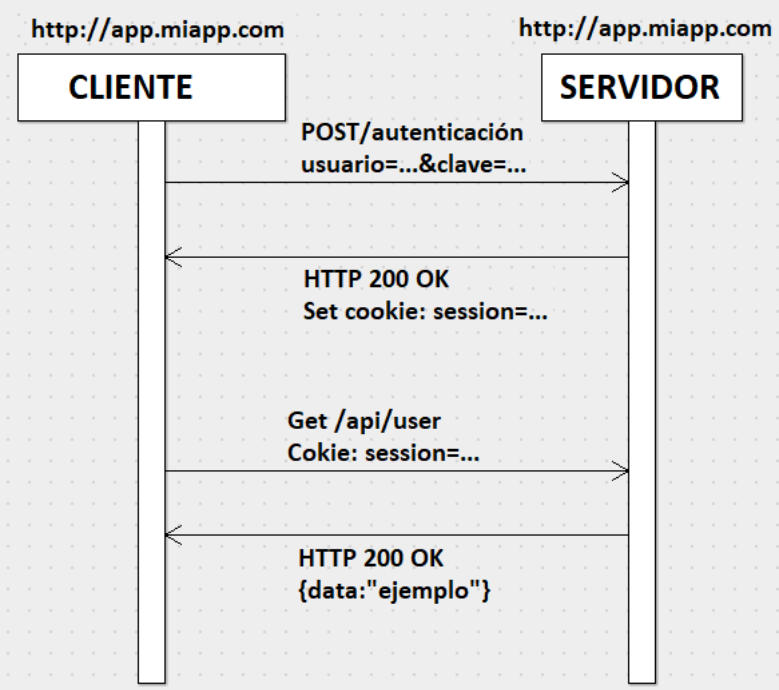


Figura 20. Autenticación basada en cookies

Rutas de la API RESTful que utilizará la aplicación.

A continuación, se propone el siguiente modelo de rutas, para consumir la API:

Ruta	Verbo http	Descripción
/futbol/jugador	GET	Listado de todos los jugadores.
/futbol/tablaposiciones?etapa=1&categoria=indor damas	GET	Tabla de posiciones por etapa y categoría deportiva
/futbol/equipo/:idEquipo	GET	Datos de un equipo específico.
/futbol/equipo	POST	Alta de un equipo.
/futbol/equipo/:idEquipo	PUT	Modifica los datos de un equipo.

/futbol/equipo/:idEquipo	DELETE	Borra la información de un equipo.

Tabla 14. Modelo rutas API RESTful

Para la creación de la API RESTful que será consumida por la app, se utilizará node.js y se escribirá en JSON, por tratarse de un formato de texto sencillo para el intercambio de datos.

Arquitectura del sistema

Para la arquitectura del sistema, se utilizará el paradigma Modelo-Vista-Controlador(MVC), por cuanto es un enfoque clásico y que añade poca complejidad para proyectos de este tipo. Este modelo se adapta sin problemas para aplicaciones Android y también para aplicaciones Flutter (en este caso será necesario usar Provider o BLoC para la gestión del estado). A continuación, se grafican varios diagramas de secuencias en base a MVC.

Diagrama de secuencias para guardar un equipo participante:

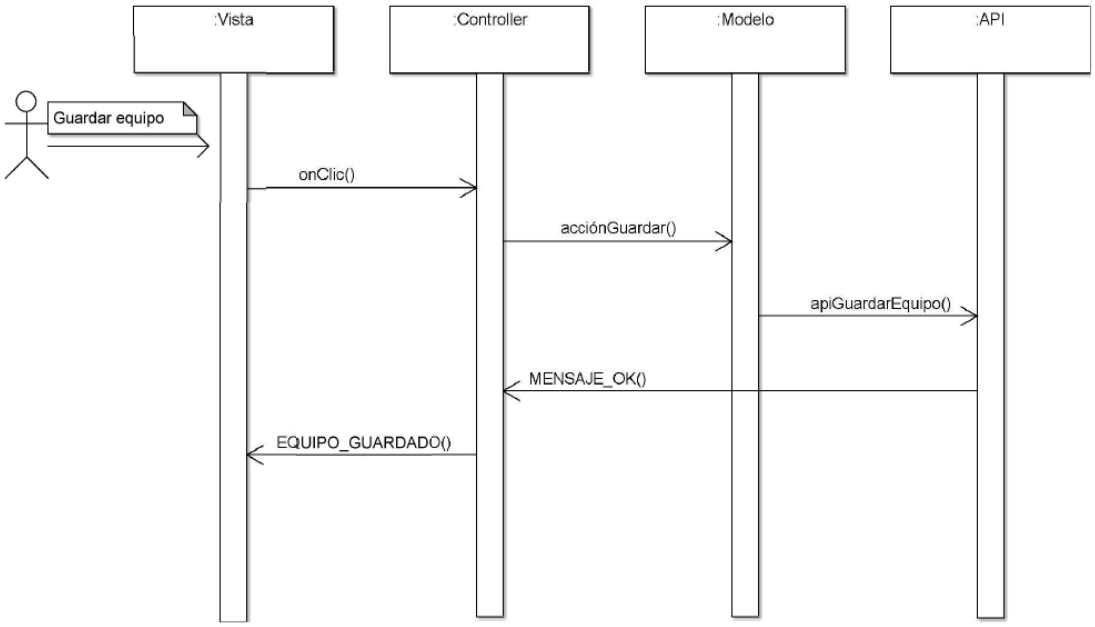


Figura 21. Diagrama de secuencias - Guardar Equipo

Diagrama de secuencias para consultar la tabla de posiciones:

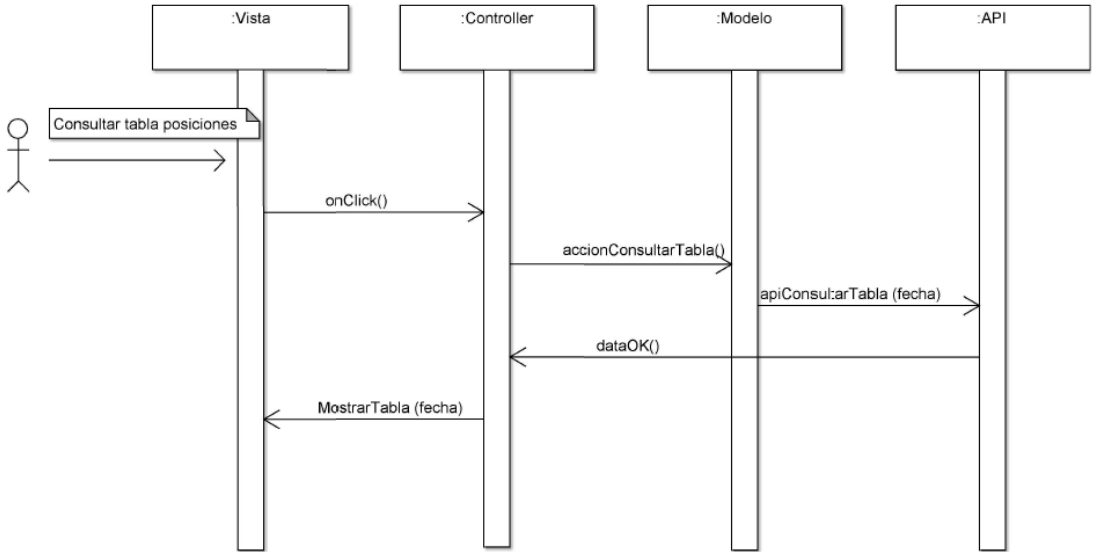


Figura 22. Diagrama de secuencias - tabla de posiciones

Diagrama de secuencias para configurar notificaciones:

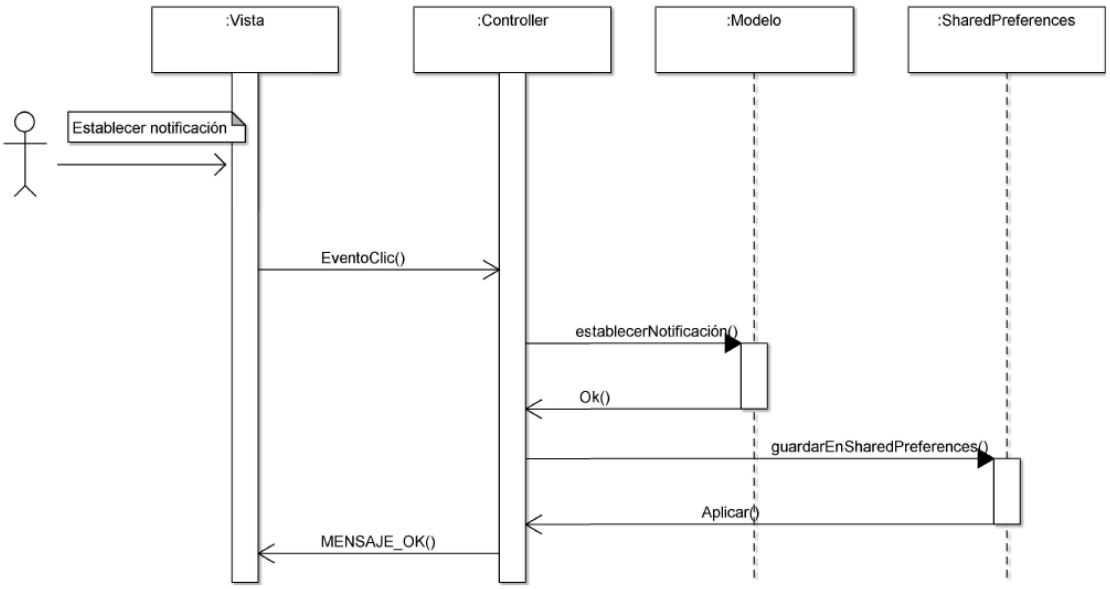


Figura 23. Diagrama de secuencias - configura notificaciones

2.8 Diseño gráfico

Para el diseño gráfico de la aplicación se utilizará el mismo del prototipado, es decir, para los colores se utilizará mayormente el blanco, también verde claro, azul y negro que, combinándolos de la manera correcta, dan claridad y frescura a la app, conservando la estética y simplicidad muy agradable a la vista del usuario.

	Verde claro	#8BC34A
	Azul	#3F51B5
	Negro	#000000

2.8.1 Logotipo

A continuación, se muestra el logotipo de la aplicación:

Figura 24. Logotipo de la aplicación

2.9 Nombre de la APP

De acuerdo con las funcionalidades de la aplicación orientadas a gestionar y apoyar la organización del campeonato, se ha elegido el siguiente nombre para la aplicación: “Gestion@ Fútbol”.

2.10 Lenguaje de la aplicación

Debido a que la aplicación está pensada para resolver un problema puntual, como es la gestión del campeonato de fútbol de los colegios de la ciudad de Esmeraldas, y que está enmarcada para un grupo de usuarios específicos; por tanto, solo utilizará el idioma español.

3. Implementación

3.1 API RESTful

En el desarrollo de la API Restful, se han utilizado las siguientes herramientas de software:
- Node js versión 12.18.4 (para procesar las peticiones http del cliente)
- MariaDB versión 10.4.11 (para la persistencia de datos)

Para poder consumir este servicio, primero es necesario autenticarse. A continuación, se muestra como ejemplo, los datos devueltos luego de la petición del cliente:

a) Url de petición
http://gestionafutbol.com:3000/futbol/autenticarcorreo

b) Credenciales pasadas en el Body en formato JSON:

```
{
    "correo": "kleberposligua@gmail.com",
    "clave": "isabel00"
}
```

c) Metodo: Post

d) Resultado:


```
{
    "idUsuario": 10,
    "usuario": "kleberposligua",
    "clave": "isabel00",
    "correo": "kleberposligua@gmail.com",
    "tipoUsuario": "invitado",
    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkVXN1YXJpbyI6MTAsIm
NvcnJlbyI6ImtsZWJlcnBvc2xpZ3VhQGdtYWlsLmNvbSJ9LCJpYXQiOjE2MDkxNzgzNjYsImV4cCI6MTY0MDc
xNDM2Nn0.aoaP_zhUu4uGVryPrVimszDnFTsBlABn0FIA0ifT2MI"
}
```


Una vez obtenido el token, se lo pasa en la cabecera de todas las peticiones utilizando el formato “bearer token”:
bearer

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkVXN1YXJpbyI6MTA
sImNvcnJlbyI6ImtsZWJlcnBvc2xpZ3VhQGdtYWlsLmNvbSJ9LCJpYXQiOjE2M
DkxNzgzNjYsImV4cCI6MTY0MDcxNDM2Nn0.aoaP_zhUu4uGVryPrVimszDnF
TsBlABn0FIA0ifT2MI:

Resumen de peticiones:

Ruta	Verbo http	Descripción
/futbol/autenticarcorreo	POST	Autentica a un usuario.
/futbol/institucion	POST	Agrega una institución
/futbol/instiitucion	GET	Listado de todas las instituciones participantes.
/futbol/institucion/:idInstitucion	GET	Obtiene una institución por su Id.
/futbol/institución/:idInstiitucion	PUT	Actualiza una institución.
/futbol/institución/:idInstiitucion	DELETE	Elimina una institución.
/futbol/equipo	GET	Listado de todos los equipos.
/futbol/equipo	POST	Agrega un equipo.
/futbol/equipo/:idEquipo	PUT	Actualiza un equipo.
/futbol/equipo/:idEquipo	DELETE	Elimina un equipo.
/futbol/equipo/:idEquipo	GET	Obtiene un equipo por su Id.
/futbol/equipocat	GET	Listado de equipos por categoría deportiva.
/futbol/jugador	GET	Listado de todos los jugadores.
/futbol/jugador	POST	Agrega un jugador.
/futbol/jugador/:idJugador	PUT	Actualiza un jugador.
/futbol/jugador/:idJugador	DELETE	Elimina un jugador.
/futbol/jugador/:idJugador	GET	Obtiene un jugador por su Id.
/futbol/jugadorfiltro/:nombre	GET	Listado de jugadores que coincide con el parámetro 'nombre'.
/futbol/resultado /futbol/resultado?q=indor damas&f=2020-08-01	GET	Calendario y resultados por categoría deportiva. Si no se pasan parámetros, se muestran todos los resultados. Parámetro q: categoría deportiva; parámetro f=fecha de juego.
/futbol/resultfechasjuegos?q=indor damas	GET	Listado de fechas jugadas por categoría deportiva.
/futbol/tablaposiciones?etapa=1&categoria=indor damas	GET	Tabla de posiciones por etapas y categorías deportivas.
/futbol/etapas	GET	Listado de etapas registradas en el torneo.
/futbol/usuario	POST	Agrega un nuevo usuario de la app.
/futbol/usuario	GET	Listado de todos los usuarios de la app.
/futbol/usuario/:idUsuario	GET	Obtiene un usuario por su Id.
/futbol/usuario/:idUsuario	PUT	Actualiza un usuario
/futbol/usuario/:idUsuario	DELETE	Elimina un usuario.
/futbol/upload	PATCH	Sube un archivo al servidor
/futbol/files	GET	Listado de archivos subidos al servidor.

Tabla 15. Resumen de peticiones HTTP

3.2 Backend

Para el backend se utilizó el lenguaje de programación php con el framework yii2. Su finalidad es permitir registrar rápidamente las fechas de juegos; y una vez que los partidos se han realizado, modificar los resultados.

Ruta de acceso:
http://gestionafutbol.com/uocbackend/web/index.php?r=resultados

Resultados

Agregar Resultado

Showing **1-6** of **6** items.

#	Num Fecha	Fecha Juego	Equipo Local	Equipo Visita	
1	1	2020-08-01	PUCESE-Indor damas	MARGARITA CORTES-Indor damas	
2	1	2020-08-01	ANDRADE MARIN-Indor varones	SAGRADO CORAZON-Indor varones	
3	1	2020-08-01	SAGRADO CORAZON-Indor damas	ANDRADE MARIN-Indor damas	
4	2	2020-08-08	ANDRADE MARIN-Indor damas	PUCESE-Indor damas	
5	2	2020-08-08	PUCESE-Master sobre 40 años	SAGRADO CORAZON-Master sobre 40 años	
6	2	2020-08-08	COLEGIO ANGEL BARBISOTTI-Indor varones	COLEGIO 5 DE AGOSTO-Indor varones	

Figura 25. Backend - registro de fechas de juegos y resultados

3.3 Desarrollo de la app

Varios problemas y contratiempos surgieron al momento de desarrollar la aplicación móvil. Al inicio de la fase de codificación, se utilizó el IDE de Android Studio y el lenguaje de programación Java, pero se presentaron dificultades al momento de configurar el menú de barra lateral con los fragments, esto es debido al poco dominio de las herramientas utilizadas. Luego se optó por utilizar el framework Flutter con el lenguaje de programación Dart, por las facilidades al construir las interfaces de usuario, y por su curva de aprendizaje relativamente rápida.

Otra dificultad surgió al momento de codificar el servidor de peticiones API RESTful, debido que a que tocaba construir desde cero el servicio, y probar desde el emulador y/o celular, hasta verificar que se tenía la funcionalidad deseada. Esto tomaba una cantidad considerable de tiempo.

A pesar que se ha utilizado un entorno de desarrollo multiplataforma en este proyecto, se deben considerar las siguientes características al momento de su instalación:

- Sistema operativo: Android
- Api mínima: Android 4.4 Api level 19
- Hardware: Dispositivos móviles con pantallas igual o superior a 4.5 pulgadas.

La API 19, corresponde con la mayoría de los dispositivos en el contexto donde se utilizará la app.

3.4 Pruebas

3.4.1 Plan de pruebas de calidad

Se estableció una batería de pruebas para garantizar que la aplicación desarrollada, cumple con las funcionalidades esperadas. Para estas pruebas se utilizó el emulador de Android y el dispositivo móvil real Samsung Galaxy A30s.

A continuación, se detallan las pruebas realizadas, correspondientes con los casos de uso más representativos:

Caso de uso asociado	**CU1**
Descripción de la prueba	Registro de instituciones
Entradas	**Salidas**
Iniciar app	Se muestra el menú principal
Menú registros, luego seleccionar del menú lateral, la opción 'Instituciones'	Listado de todas las instituciones registradas
Seleccionar una institución	Pantalla para editar los datos de la institución
Clic floating actión button '+'	Pantalla para agregar nueva institución
Clic botón 'Guardar'	Mensaje de confirmación y regreso a la lista de instituciones creadas.

Tabla 16. Prueba - Registro de instituciones

Caso de uso asociado	**CU2**
Descripción de la prueba	Registro de jugadores
Entradas	**Salidas**
Iniciar app	Se muestra el menú principal
Menú registros, luego seleccionar del menú lateral, la opción 'Jugadores'	Listado de todas los jugadores inscritos
Seleccionar un jugador	Pantalla para editar los datos del jugador
Clic en ícono 'cambiar foto'	Galería de imágenes para seleccionar la foto del jugador
Clic botón 'Guardar'	Mensaje de confirmación y regreso a la lista de jugadores registrados.
Clic floating actión button '+'	Pantalla para agregar un nuevo jugador
Clic botón 'Guardar'	Mensaje de confirmación y regreso a la lista de jugadores registrados.

Tabla 17. Prueba - Registro de jugadores

Caso de uso asociado	**CU6**
Descripción de la prueba	Consultar resultados
Entradas	**Salidas**
Iniciar app	Se muestra el menú principal
Menú reportes, luego seleccionar del menú lateral, la opción 'Resultados'	Se muestran los resultados de todas las categorías deportivas
Seleccionar una categoría, por ejemplo 'Indor Damas'	Se listan los resultados de todas las fechas jugadas para la categoría

	seleccionada.
Seleccionar una fecha de juego	Para la categoría seleccionada, se listan los resultados correspondientes a la fecha consultada.

Tabla 18. Prueba - Consulta de resultados

Caso de uso asociado	**CU7**
Descripción de la prueba	Consultar tabla de posiciones
Entradas	**Salidas**
Iniciar app	Se muestra el menú principal
Menú reportes, luego seleccionar del menú lateral, la opción 'Tabla de posiciones'	Por defecto se muestra la tabla de posiciones de la categoría 'Indor varones' para la etapa 1
Seleccionar una categoría, por ejemplo 'Indor Damas'	Se lista la tabla de posiciones para la categoría seleccionada en la etapa 1
Seleccionar una etapa de juego (primera etapa, todos contra todos,1,2, etc)	Para la categoría seleccionada, se lista la tabla de posiciones para la etapa consultada

Tabla 19. Prueba - Consulta de tabla de posiciones

3.4.2 Resultados de las pruebas

El plan de pruebas de calidad se completó con éxito, validando adecuadamente las funcionalidades establecidas en los casos de uso. De esta manera se garantiza que la aplicación cumple correctamente los requerimientos del sistema.

4. Conclusiones

4.1 Conclusiones

La elaboración de este proyecto de software, ayuda a integrar todos los conocimientos adquiridos durante el master; brinda un enfoque global sobre la gestión de un proyecto de software, desde la fase de planificación, hasta la puesta en marcha de la aplicación.

La elaboración de un prototipo de la aplicación proporciona una idea general de las funcionalidades y la interacción que el usuario realizará sobre la aplicación, pero es necesario considerar las limitaciones de los dispositivos en los que va a funcionar, para que la implementación sea viable.

Es importante mencionar que, tomando el contexto global de desarrollo de un proyecto de software, se incurriría en atrasos si la planificación no es la adecuada, si los requisitos del usuario no han sido recopilados correctamente y si no se consiguen con disciplina los hitos establecidos.

4.2 Consecución de los objetivos planteados

Debido a las limitaciones de tiempo, no se han podido cumplir con todos los objetivos inicialmente planteados. En particular, no se pudo implementar las notificaciones automáticas para que los usuarios estén al tanto de los resultados, claro está que una vez que abran la aplicación, los tendrán a su disposición. Otro objetivo no cumplido, se refiere a la funcionalidad de notificaciones en las redes sociales como Facebook y Twitter, para promocionar el campeonato.

4.3 Análisis de planificación

De acuerdo con la planificación de trabajo establecida para este proyecto, y siguiendo las fechas establecidas en el calendario para la asignatura TFM, se ha podido cumplir sin contratiempos las primeras entregas correspondientes a los dos primeros hitos (PEC1 – PEC2); sin embargo, en la fase de implementación correspondiente al tercer hito (PEC3), ha habido un retraso por lo corto del tiempo y el poco dominio de las herramientas de desarrollo seleccionadas en primera instancia. Este retraso fue corregido en la entrega del último hito (PEC4).

4.4 Líneas de trabajos futuros

- Se puede desarrollar una versión de esta aplicación que incluya la funcionalidad de generar automáticamente el formato de campeonato, en base a plantillas o modelos preestablecidos.

- Agregar la funcionalidad de notificaciones automáticas, tanto para usuarios como en redes sociales.
- Cuando se consulta la tabla posiciones, sería de mucha utilidad que, al seleccionar un equipo, se muestre el historial de todos sus encuentros con sus respectivos resultados.

5. Glosario

Android: Sistema operativo que se emplea en dispositivos móviles, basado en el núcleo de Linux.

API: Del inglés Application Programming Interface (Interfaz de programación aplicada). Es un programa que permite que otros programas se comuniquen con un programa en específico.

Dart: Es un lenguaje de código abierto, estructurado y flexible, orientado a objetos, basado en clases, con herencia simple y soporte de interfaces, clases abstractas y tipado opcional de datos.

Flutter: Es un framework de código abierto para el desarrollo de aplicaciones móviles, creado por Google.

Fulbito: Modalidad del fútbol en el que juegan 9 jugadores por cada equipo.

JSON: (JavaScript Object Notation – Notación de Objetos de JavaScript) es un formato de texto sencillo para el intercambio de datos.

Modelo Waterfall: Es conocido en Ingeniería de software como modelo en cascada. Es un enfoque metodológico utilizado para guiar de manera secuencial, el ciclo de vida de un proyecto de software.

REST: Es una arquitectura para aplicaciones basadas en redes (como internet). Sus siglas REpresentational State Transfer.

RESTful: Es un servicio web que está basado en la arquitectura REST.

Slider de imágenes: Es un elemento de una página web que muestra múltiples imágenes y texto que se alternan entre ellas.

Wireframes: Es un esquema visual que transmite un nivel básico de comunicación, estructura y comportamiento durante el diseño de un sistema.

PHP: Hypertext Preprocessor, es un lenguaje de programación de código abierto, muy popular especialmente para el desarrollo de aplicaciones web.

Yii: Es un framework de PHP de alto rendimiento, basado en componentes para desarrollar aplicaciones web modernas en poco tiempo.

6. Bibliografía

6.1 Referencias externas

Abdul Aziz Ahwan. (2020). 13 Show Categories Item in DropdownButtonFormField—Flutter Sqflite TodoList App Tutorial. https://www.youtube.com/watch?v=PO24ydfdBv0

Arquitectura de una API REST · Desarrollo de aplicaciones web. (s. f.). Recuperado 26 de octubre de 2020, de https://juanda.gitbooks.io/webapps/content/api/arquitectura-api-rest.html

Chamorro, A. (2020, junio 14). Arquitecturas y patrones de diseño en Flutter—Capítulo 1—MVVM. Medium. https://medium.com/flutter-madrid/arquitecturas-y-patrones-de-dise%C3%B1o-en-flutter-cap%C3%ADtulo-1-mvvm-591ad3d63bb

Construyendo un formulario con validaciones. (s. f.). Recuperado 28 de diciembre de 2020, de https://flutter-es.io/docs/cookbook/forms/validation

Creación de una API con node.js · Desarrollo de aplicaciones web. (s. f.). Recuperado 26 de octubre de 2020, de https://juanda.gitbooks.io/webapps/content/api/creacion_de_una_api_con_nodejs.html

Dev Clips. (2020). DataTable Widget, JSON Parsing and Provider API in Flutter. https://www.youtube.com/watch?v=g6EU6J3RmSE

Dev Stack. (2020, julio 22). NodeJs: Rest-API End-Point creation for Upload an Image with help of Multer || 24. https://www.youtube.com/watch?v=HCD5JYTdxvY&list=PLtIU0BH0pkKqypuOtDhcXZ4oATJfji49r&index=17

Diagrama de clases. Teoría y ejemplos. (s. f.). Recuperado 26 de octubre de 2020, de https://diagramasuml.com/diagrama-de-clases/

Fazt Code. (2019, noviembre 25). Nodejs & DigitalOcean | Subida de Aplicacion con SSH, PM2 y nginx. https://www.youtube.com/watch?v=gmfUNDmJDuk

Instalar LAMP (Linux, Apache, MySQL y PHP) en GNU/Linux (Ubuntu y derivadas). (s. f.). Recuperado 28 de diciembre de 2020, de https://www.gestionatuweb.net/instalar-lamp-linux-apache-mysql-y-php-en-gnu-linux-ubuntu-y-derivadas/

JSON. (s. f.). Recuperado 26 de octubre de 2020, de https://www.json.org/json-es.html

Juanda. (s. f.). Introducción a Node.js · Desarrollo de aplicaciones web. Recuperado 26 de octubre de 2020, de https://juanda.gitbooks.io/webapps/content/javascript/node.html

Khan, M. A. (2020, enero 11). How To Fetch Data using API Calls in Flutter. Medium. https://medium.com/@maffan/how-to-fetch-data-using-api-calls-in-flutter-99668a60170c

Node.js—RESTful API - Tutorialspoint. (s. f.). Recuperado 26 de octubre de 2020, de https://www.tutorialspoint.com/nodejs/nodejs_restful_api.htm

Primeros pasos: Instalar Yii. (s. f.). Yii Framework. Recuperado 28 de diciembre de 2020, de /doc/guide/2.0/es/start-installation

Que es REST y que Es RESTful. (s. f.). Recuperado 26 de octubre de 2020, de https://www.codigonaranja.com/restful-web-service

6.2 Libros

Amatller, A. M. (2020). Exposición de contenidos en vídeo. Barcelona: UOC.

Guía Desarrollo Servidor. (2020). Barcelona: UOC.

José, R. (2020). El trabajo final como proyecto. Barcelona: UOC.

José, R. (2020). La gestión del proyecto a lo largo del trabajo final. Barcelona: UOC.

Montagut, R. B. (2020). Presentación de documentos y eleaboración de presentaciones. Barcelona: UOC.

Nita, S. H., & Rut, V. (2020). Redacción de textos científico-técnicos. Barcelona: UOC.

Robert, C. (2020). Introducción al trabajo final. Barcelona: UOC.

7. Anexos

- Memoria final del proyecto
  Archivo: TFM_Plantilla_Memoria_es_v6_KleberPosligua_pec4.pdf

- Código fuente de la aplicación desarrollada
  Archivo: tfmuoc.zip

  Github: https://github.com/kleberposligua/tfmuoc.git

- Código fuente servidor API RESTful
  Archivo: serverapi.zip

  Dirección web del servidor: http://gestionafutbol.com:3000/futbol/

  Github: https://github.com/kleberposligua/tfmuoc_serverapi.git

- Código fuente backend creado con el framework yii
  Archivo: Uocbackend.zip

  Página web de acceso:
  http://gestionafutbol.com/uocbackend/web/index.php?r=resultados

  Github: https://github.com/kleberposligua/uocbackend.git

- Video presentación del proyecto final en formato mp4
  Subido en Present@

- Archivo instalable (apk) de la aplicación
  Archivo: app-release.apk

- Manual de usuario
  Archivo: Manual de usuario.pdf

- Script de creación de base de datos
  Archivo: gestionafutbol_bdd.zip

- Presentación en power point
  Archivo: Presentación_defensa_tfm_uoc.pptx

- Archivo de indicaciones de instalación
  Archivo: Indicaciones.pdf