



UNIVERSIDAD TECNOLÓGICA DE DURANGO

DESARROLLO Y GESTIÓN DE SOFTWARE

Parcial 2

Evaluación II: Aplicación Móvil

Integrantes:

Diana Itzel Ayala Herrera

Carlos Emmanuel De la Cruz Cenicerros

Jorge Israel González Chávez

Irving Ulises Rodríguez Romero

Docente: M.A Raúl Iván Herrera González

Victoria de Durango, Durango. Noviembre 2021

Tabla de contenido

Tabla de contenido.....	2
Tabla de ilustraciones.....	3
Introducción.....	4
Contenido	5
Metodología ágil: Kanban	5
Repositorio de código fuente: Git y Github	7
Git	7
Github.....	7
Arquitectura de software: SOA	8
Patrón de diseño: MVC.....	9
Modelo:	9
Vista:	9
Controlador:	9
Framework: Flutter.....	10
Flutter:.....	10
Dart:	10
Mecanismos de seguridad: Auth0.....	11
Auth0:.....	11
Api de terceros: The Movie Database (TMDB).....	12
Servicios en la nube: Firebase.....	13
Firebase:	13
Conclusión	14
Referencias	15

Tabla de ilustraciones

Ilustración 1.- Tablero Kanban	6
Ilustración 2.- Trello (Logo)	6
Ilustración 3.- Git	7
Ilustración 4.- Github	7
Ilustración 5.- Arquitectura de software SOA	8
Ilustración 6.- MVC	9
Ilustración 7.- Flutter	10
Ilustración 8.- Dart	10
Ilustración 9.- Auth0	11
Ilustración 10.- Api The Movie DB (TMDB)	12
Ilustración 11.- Firebase	13

Introducción

El desarrollo de aplicaciones móviles es el conjunto de procesos y procedimientos involucrados en la escritura de software para dispositivos informáticos pequeños e inalámbricos, como teléfonos inteligentes y otros dispositivos portátiles. A través de este documento se explicará el procedimiento realizado durante el desarrollo de la aplicación móvil la cual será considerada como la evaluación del segundo parcial.

Al igual que el desarrollo de aplicaciones web, el desarrollo de aplicaciones móviles tiene sus raíces en el desarrollo de software más tradicional. Sin embargo, una diferencia fundamental es que las aplicaciones móviles a menudo se escriben específicamente para aprovechar las características únicas de un dispositivo móvil en particular.

Dentro del documento se hablará acerca de la metodología ágil la cual fue implementada para llevar a cabo el control de los tiempos de entrega de los diversos componentes que conforman la aplicación.

Del mismo modo se explicarán las diversas tecnologías las cuales fueron utilizadas para llevar a cabo el desarrollo de la app las cuales son el framework utilizado, el repositorio que permitirá llevar el control de las versiones, las Apis para terceros, los patrones de diseño aplicados para la creación de la interfaz, los servicios en la nube utilizados y los diversos mecanismos de seguridad utilizados para llevar a cabo la protección de la aplicación y mantener la información almacenada segura.

Contenido

Metodología ágil: Kanban

La metodología ágil que fue implementada para llevar a cabo el control de los tiempos de entrega fue Kanban.

El objetivo fundamental de este método es el de minimizar los desperdicios o residuos en el proceso de fabricación sin que eso afecte a la producción. Así, se pueden controlar mejor los gastos y conseguir una mayor eficiencia. El equivalente en otras áreas que no sean la línea de fabricación es que se minimizan los atascos en la productividad.

Para llevar a cabo la aplicación de la metodología ágil Kanban se utilizó como herramienta la plataforma de internet Trello el cual es un software de administración de proyectos con interfaz web y con cliente para iOS y Android para organizar proyectos en el cual se creó el tablero el cual cuenta con un total de 4 secciones las cuales son:

- Por hacer
- En proceso
- Terminado
- Pruebas

Dentro de la sección “Por hacer” se encuentran todas las tareas las cuales se encuentran libres para que los integrantes del equipo puedan llevar a cabo su realización.

La sección de “En proceso” contiene todas las tareas que están siendo realizadas por los integrantes del equipo y están por terminar

En la sección de “Terminado” se encuentran todas las tareas que ya han sido realizadas por algún integrante del equipo sin embargo estas aún deben pasar por un último proceso el cual es el de pruebas.

Dentro de la sección de “Pruebas” se encuentran las tareas que se encuentran siendo probadas por el tester de equipo con el propósito de garantizar la óptima funcionalidad de las funcionalidades de la aplicación

La siguiente ilustración permitirá apreciar el tablero Kanban creado a través de la plataforma web Trello.

En la ilustración se logra observar las diversas secciones en las que se encuentra dividido el tablero además de las diversas tareas que fueron realizadas para la creación de la aplicación.

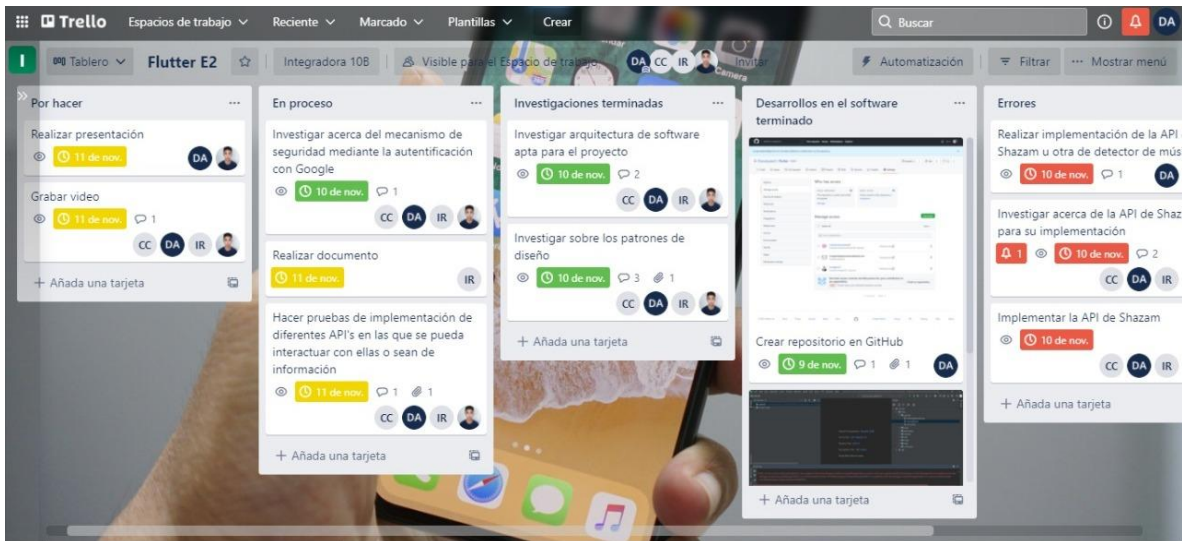


Ilustración 1.- Tablero Kanban



Ilustración 2.- Trello (Logo)

Repositorio de código fuente: Git y Github

Para llevar a cabo el control de las versiones del código fuente de la aplicación se utilizó Git como software de control de versiones y Github fue utilizado como plataforma de alojamiento de repositorio del código.

Git

Es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia, la confiabilidad y compatibilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente. Su propósito es llevar registro de los cambios en archivos de computadora incluyendo coordinar el trabajo que varias personas realizan sobre archivos compartidos en un repositorio de código.



Ilustración 3.- Git

Github

GitHub es una forja para alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza principalmente para la creación de código fuente de programas de ordenador. El software que opera GitHub fue escrito en Ruby on Rails.



Ilustración 4.- Github

Arquitectura de software: SOA

Para llevar a cabo la creación de la aplicación fue necesario la implementación de una arquitectura de software la cual nos permitiera definir la estructura que tendrá la app aplicando un conjunto de patrones y abstracciones coherentes que proporcionan un marco definido y claro para interactuar con el código fuente del software. En este caso nosotros utilizamos como arquitectura una de las más conocidas y utilizadas por el mundo del desarrollo del software la cual es la arquitectura orientada a servicios de cliente también conocida como SOA.

La arquitectura SOA, o arquitectura orientada a servicios, define una manera de hacer que los componentes de software sean reutilizables a través de interfaces de servicio. Estas interfaces utilizan estándares de comunicación comunes entre sí, de tal manera que pueden incorporarse rápidamente a nuevas aplicaciones sin tener que realizar una integración profunda cada vez.

Cada servicio de una SOA incorpora el código y las integraciones de datos necesarias para ejecutar una función de negocios completa y discreta

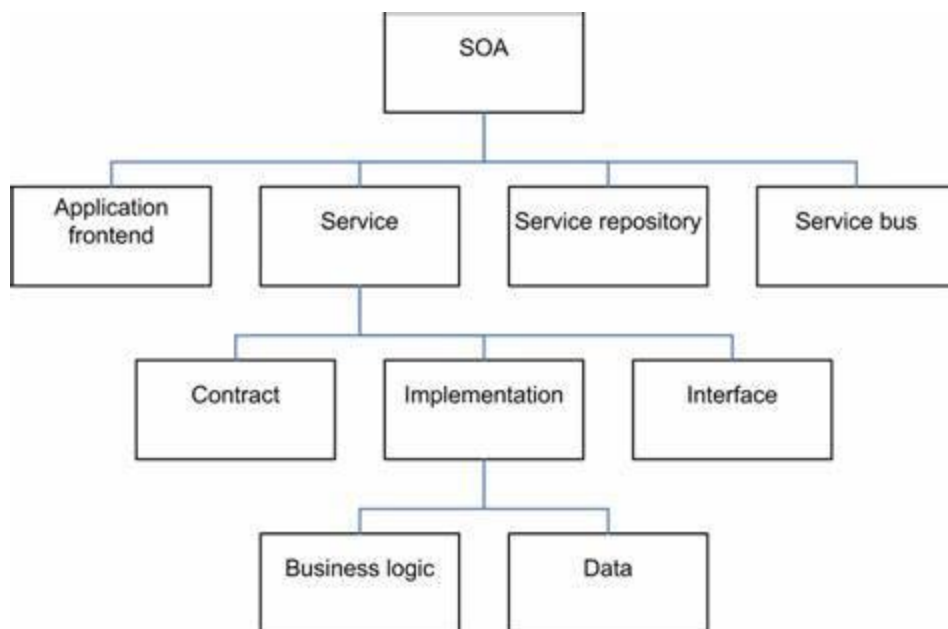


Ilustración 5.- Arquitectura de software SOA

Patrón de diseño: MVC

El patrón de diseño que fue implementado durante el desarrollo de la aplicación fue el modelo vista controlador (MVC) ya que este patrón nos permite dividir la lógica de la aplicación en tres segmentos diferentes los cuales son los modelos, las vistas y los controladores.

Modelo:

El modelo define qué datos debe contener la aplicación. Si el estado de estos datos cambia, el modelo generalmente notificará a la vista (para que la pantalla pueda cambiar según sea necesario) y, a veces, el controlador (si se necesita una lógica diferente para controlar la vista actualizada).

Vista:

Los componentes de la Vista son los responsables de generar la interfaz de nuestra aplicación, es decir, de componer las pantallas, páginas, o cualquier tipo de resultado utilizable por el usuario o cliente del sistema.

Controlador:

El controlador contiene una lógica que actualiza el modelo y / o vista en respuesta a las entradas de los usuarios de la aplicación.

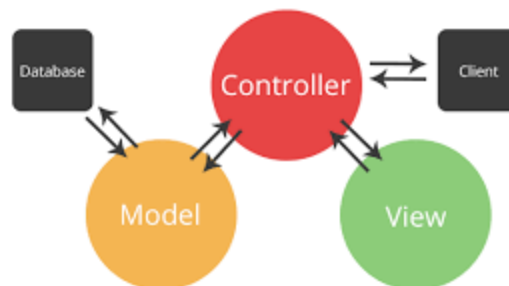


Ilustración 6.- MVC

Framework: Flutter

Para llevar a cabo la creación de la aplicación se utilizó un framework el cual permitiera al equipo optimizar el proceso de desarrollo de la app por lo cual se optó por utilizar Flutter ya que puede ser complementado con Dart.

Flutter:

Flutter es un SDK de código fuente abierto de desarrollo de aplicaciones móviles creado por Google. Suele usarse para desarrollar interfaces de usuario para aplicaciones en Android, iOS y Web, así como método primario para crear aplicaciones para Google Fuchsia.



Ilustración 7.- Flutter

Dart:

Dart es un lenguaje de programación open source, relativamente nuevo, que fue desarrollado por Google y que lanzó su primera versión en 2011. Este lenguaje se creó con el objetivo de permitir a los desarrolladores utilizar un lenguaje orientado a objetos y con análisis estático de tipo.



Ilustración 8.- Dart

Mecanismos de seguridad: Auth0

Uno de los mecanismos de seguridad implementado en la aplicación fue Auth0 ya que este es un mecanismo que es utilizado por muchas empresas grandes para brindar seguridad a sus aplicaciones y plataformas por lo cual brindara gran seguridad a nuestra aplicación

Auth0:

Auth0 es un estándar abierto para la delegación de acceso, comúnmente utilizado como una forma para que los usuarios de Internet otorguen a sitios web o aplicaciones acceso a su información en otros sitios web, pero sin darles las contraseñas. Este mecanismo es utilizado por empresas como Amazon, Google, Facebook, Microsoft y Twitter para permitir a los usuarios compartir información sobre sus cuentas con aplicaciones o sitios web de terceros.



Ilustración 9.- Auth0

Api de terceros: The Movie Database (TMDB)

Para llevar a cabo la funcionalidad de la aplicación la cual es similar a una app de estilo cartelera de películas de cine fue necesario implementar una api de terceros que nos permite obtener como resultado todas las películas existentes hasta el momento incluyendo las que se encuentran presentándose en los cines.

La api utilizada para esto fue “The Movie Database” esta api nos permite obtener a través de promesas un listado con la información de todas las películas que coincidan con la búsqueda, sin embargo, también nos permite obtener todas las películas más populares y los próximos estrenos que están por salir que realizó el usuario sin embargo también nos permite obtener la información de los actores y las empresas a las cuales pertenecen. Esta api se utiliza a través de peticiones HTTP que van dirigidas a un endpoint el cual nos otorga toda la información en diversos tipos de formatos sin embargo nosotros utilizamos el formato más conocido y utilizado el cual es json.



Ilustración 10.- Api The Movie DB (TMDB)

Servicios en la nube: Firebase

Durante el desarrollo de la aplicación se implementaron servicios en la nube como lo es Firebase. Este servicio fue utilizado para llevar a cabo la gestión de la información que almacenará la aplicación la cual será protegida gracias a los mecanismos de seguridad mencionados anteriormente.

Firebase:

Firebase de Google es una plataforma en la nube para el desarrollo de aplicaciones web y móvil. Está disponible para distintas plataformas (iOS, Android y web), con lo que es más rápido trabajar en el desarrollo.

Su función esencial es hacer más sencilla la creación de tanto aplicaciones webs como móviles y su desarrollo, procurando que el trabajo sea más rápido, pero sin renunciar a la calidad requerida.

Firebase dispone de diferentes funcionalidades, que se pueden dividir básicamente en 3 grupos: Desarrollo (Develop), Crecimiento (Grow) y Monetización (Earn), a los que hay que sumar la Analítica (Analytics).



Ilustración 11.- Firebase

Conclusión

La creación de esta aplicación móvil nos permitió poner en práctica diversos conceptos importantes aprendidos durante las clases. Durante el desarrollo de la app se aplicaron los conocimientos referentes a los mecanismos de seguridad los cuales nos permitieron brindar mayor seguridad a nuestra aplicación para mantener segura la información que se está manejara para ello se utilizó un mecanismo externo muy utilizado por grandes empresas el cual fue Auth0 que nos permite autenticar que usuarios pueden tener acceso a la aplicación por medio de una autenticación.

Se aplicaron los conocimientos referentes a los patrones de diseño ya que se utilizó uno de los más utilizados por los desarrolladores el cual es el modelo vista controlador también conocido como MVC. Este patrón de diseño fue utilizado ya que nos permitió dividir la lógica de la aplicación en segmentos que permitieron facilitar más el proceso de desarrollo de la aplicación y tener un mejor orden.

De igual forma se aplicaron los conocimientos acerca de las metodologías ágiles y su aplicación ya que durante la creación de la aplicación se puso en práctica la metodología Kanban con el propósito de controlar los tiempos de entrega de cada uno de los apartados y funcionalidades de la aplicación, así como también las pruebas realizadas para asegurar el funcionamiento óptimo.

Para llevar a cabo el control de las versiones el cual fue otro concepto visto durante las clases se implementó una de las tecnologías más conocidas por los desarrolladores el cual es el Git, este es un software de control de versiones de código fuente sin embargo también se utilizó la plataforma Github con el propósito de alojar nuestro proyecto a través de un repositorio y controlar las versiones de la aplicación por medio de una interfaz gráfica lo cual facilita el proceso.

Referencias

- Cordón, M. J. (10 de 11 de 2021). *Hiberus Blog*. Obtenido de <https://www.hiberus.com/crecemos-contigo/que-es-el-lenguaje-de-programacion-dart/>
- desarrolladorWeb.com. (10 de 11 de 202). *desarrolladorWeb.com*. Obtenido de <https://desarrolloweb.com/articulos/que-es-mvc.html>
- ICATECH. (10 de 11 de 2021). *ICATECH*. Obtenido de <https://www.icatech.edu.mx/que-es-flutter-y-por-que-utilizarlo-en-la-creacion-de-tus-apps/>
- IEBS. (10 de 11 de 2021). *IEBS*. Obtenido de <https://www.iebschool.com/blog/firebase-que-es-para-que-sirve-la-plataforma-desarrolladores-google-seo-sem/>
- Rodas, G. (10 de 11 de 2021). *Platzi*. Obtenido de <https://platzi.com/clases/1439-autenticacion-oauth/15832-que-es-auth0/>