



softserve

KOTLIN CRASH COURSE

FLASHCARDS

Proyecto: QuizKard

Ana Sofía Espinoza Durán

Jorge Sebastián Alejandro Chiu Molina Ramírez

Pablo Vega Cibrián

Gerardo Espino Espindola

Jesús Alejandro Aranda Reyes

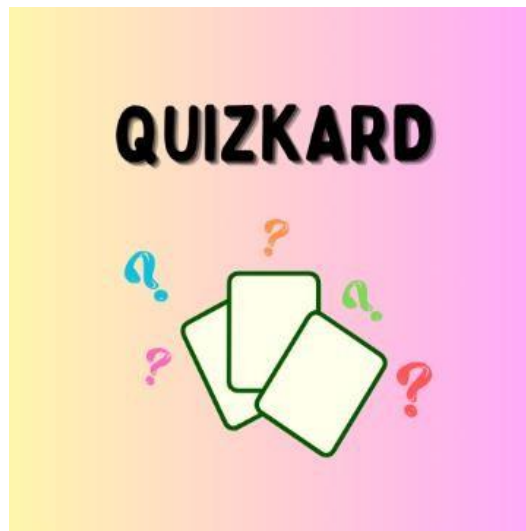
Manuel Guillermo Legorreta Ramírez

Diego Pérez Tabullo

Rivera Vergara Susana

Introducción

Este proyecto se enfoca en el desarrollo de una aplicación de tarjetas de estudio en Android utilizando el lenguaje de programación Kotlin. El proyecto se divide en dos entregables y Sprints. El primer entregable, "Configuración del Proyecto y Estructura de las Tarjetas de Estudio," se centra en establecer la base del proyecto, desarrollar la interfaz de usuario y definir la estructura de datos para las tarjetas de estudio. El segundo entregable, "Modo de Cuestionario e Interfaz de Usuario," se enfoca en crear la interfaz de usuario para el modo de cuestionario, implementar la lógica del cuestionario y habilitar la función de volteo de tarjetas.



Problemática que deseamos cubrir

El proceso de aprendizaje es fundamental en nuestra vida, y cada uno de nosotros encuentra maneras únicas de absorber y retener información. Uno de los métodos más efectivos para estudiar y recordar datos clave es mediante el uso de flashcards. Las tarjetas de estudio, o flashcards, son una herramienta versátil que se ha convertido en una parte esencial de la vida de estudiantes, profesionales y entusiastas del aprendizaje.

En estas tarjetas, se presenta información importante de manera concisa y organizada, lo que facilita su revisión y memorización. Las flashcards pueden abordar una amplia gama de temas, desde vocabulario y conceptos matemáticos hasta hechos históricos y fórmulas científicas. Además, se pueden utilizar de diversas maneras, ya sea de forma tradicional en papel o de manera electrónica a través de aplicaciones y programas específicos.

En esta serie de flashcards, exploraremos diferentes aspectos del estudio, incluyendo técnicas efectivas de aprendizaje, consejos para la gestión del tiempo, estrategias de memorización y mucho más. Al dominar el uso de las flashcards y otras herramientas de estudio, estarás mejor preparado para el éxito en tus desafíos académicos y profesionales. ¡Comencemos a expandir nuestros conocimientos y habilidades de estudio!

Objetivo principal:

Crear una aplicación que se enfoca en el desarrollo de una aplicación de tarjetas de estudio en Android utilizando el lenguaje de programación Kotlin, para jóvenes de secundaria hasta universidad.

Objetivos específicos:

Cubrir la necesidad de estudio con la aplicación flashcard.

Desarrollar una pantalla para ingresar el tema a estudiar junto con la pregunta y la respuesta.

Crear navegación entre pantallas para comenzar el estudio.

Desarrollar la función de barajear.

Deliverable 2: Quiz Mode and User Interface

Sprint 2:

6. Quiz UI design: Create the quiz mode user interface with a start button and flashcard display space.
7. Navigation: Implement basic navigation between the main screen and the quiz screen.
8. Quiz logic: Write code to shuffle and display flashcards in random order during the quiz.
9. Flip feature: Allow users to flip between questions and answers when a flashcard is tapped.

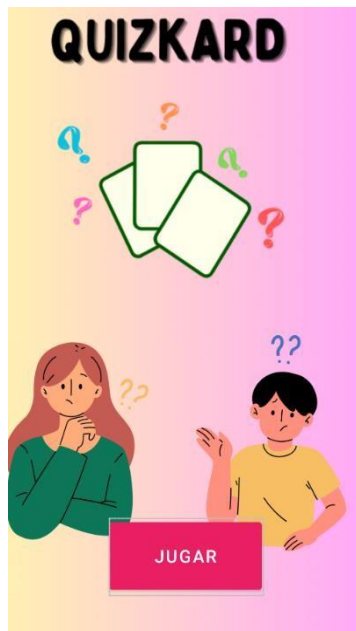
Basic UI: Develop a main screen and a quiz screen.

Para la implementación del proyecto decidimos implementar interfaces intuitivas y amigables para el usuario, al abrir QuizKard, el usuario podrá ver la pantalla principal:

MOCKUP



DIAGRAMA DE FLUJO BAJO LA METODOLOGÍA DESIGN THINKING



Al presionar el botón, el usuario podrá crear una "ronda" de flashcards

The image shows a form titled 'Crear nuevas flashcards' in a teal header bar. Below the header, there is a pink box with the text 'Categoría para tus flashcards'. Underneath this is a text input field labeled 'Categoría'. Below that is another text input field labeled 'Pregunta'. Below that is a third text input field labeled 'Respuesta'. At the bottom of the form, there are two buttons: a purple button labeled 'SIGUIENTE PREGUNTA' and an orange button labeled 'GUARDAR'.

- El usuario ingresará en la categoría de su preferencia para sus flashcards
- El usuario de igual forma ingresará la pregunta con su respectiva respuesta
- Al presionar el botón "siguiente pregunta" la pregunta será almacenada con su respectiva respuesta, y los campos quedarán en blanco para que el usuario pueda ingresar otra pregunta



SIGUIENTE PREGUNTA

El usuario así puede agregar sus preguntas, con un máximo de 10

- Al presionar el botón “guardar”, la ronda de flashcards será almacenada y el usuario podrá ver la ronda que acaba de crear y todas las que han sido creadas



- Podemos visualizar y seleccionar todas las rondas creadas por el usuario



Historia

Matemáticas

Geografía

La categoría ingresada por el usuario es almacenada igual que las preguntas y repuestas



Geografía



Cuál es el lugar más frío de la tierra?

Para mostrar la respuesta, el usuario debe usar el botón “Mostrar Respuesta”



MOSTRAR RESPUESTA



Flashcards

Geografía

La Antártida

OCULTAR RESPUESTA



III ○ <

La respuesta se muestra, de igual manera, al pulsar el botón, la pregunta se ocultará, mostrando la respuesta

Los botones en la parte posterior permiten al usuario desplazarse en las flashcards

Para mayor practicidad, al dirigirse a las rondas guardadas, el usuario podrá eliminar las rondas



Flashcards

Historia

Matemáticas

Geografía

Componentes de la app

Para la implementación del sprint2, hemos implementado una nueva función que logre que las preguntas se muestren en modo aleatorio, esto hará que la interacción entre el usuario y la app se vuelva más dinámica e interesante.

Para lograr esto, usamos:

```
if (preguntas.isNotEmpty()) {  
    val preguntasMezcladas = preguntas.zip(respuestas).shuffled()  
    preguntas = preguntasMezcladas.map { it.first }.toMutableList()  
    respuestas = preguntasMezcladas.map { it.second }.toMutableList()  
  
    preguntaView.text = preguntas[preguntaActual]  
    respuestaButton.text = "Mostrar respuesta"  
}
```

Con esto logramos mezclar nuestra preguntas almacenadas y así lograr que el usuario las visualice de manera aleatoria.



De igual manera con la implementación del botón, se visualizará la respuesta, con esto logrando una flip feature



Metodología

La metodología que se utilizará para desarrollar nuestra aplicación será SCRUM [9], se eligió ya que Scrum proporciona flexibilidad y capacidad de respuesta a los cambios, lo que es esencial en nuestro proyecto de desarrollo de software.

La metodología Scrum fue propuesta por primera vez por Jeff Sutherland y Ken Schwaber a mediados de la década de 1990. Ambos son reconocidos como co-creadores de Scrum y han sido figuras influyentes en la promoción y evolución de esta metodología ágil.

La metodología se caracteriza por los siguientes ciclos importantes:

Formación del Equipo Scrum:

Forma un equipo Scrum que incluya a los miembros del proyecto, como desarrolladores, diseñadores y posiblemente expertos en la industria hotelera. Define los roles de Scrum Master y Product Owner.

Creación del Backlog del Producto:

El Product Owner (dueño del producto) trabaja en colaboración con los stakeholders para crear un Backlog del Producto que contenga todas las funcionalidades y características deseadas para el sistema.

Planificación de Sprint:

El equipo selecciona un conjunto de elementos del Backlog del Producto para trabajar durante un Sprint (un período de tiempo fijo, como 2-4 semanas).

Reuniones Diarias de Scrum:

Durante el Sprint, el equipo se reúne todos los días en una reunión diaria de Scrum para compartir el progreso y discutir los obstáculos.

Desarrollo Incremental:

El equipo trabaja en el desarrollo del sistema en incrementos, priorizando las funcionalidades más importantes y valiosas para el cliente.

Revisión del Sprint:

Al final de cada Sprint, el equipo realiza una revisión para demostrar los avances y recibir comentarios del Product Owner y los stakeholders.

Retrospectiva:

El equipo realiza una retrospectiva para analizar lo que funcionó bien y lo que podría mejorarse en el próximo Sprint.

Iteración Continúa:

El ciclo de planificación, desarrollo, revisión y retrospectiva se repite en cada Sprint, lo que permite la adaptación continua a medida que se avanza en el proyecto.

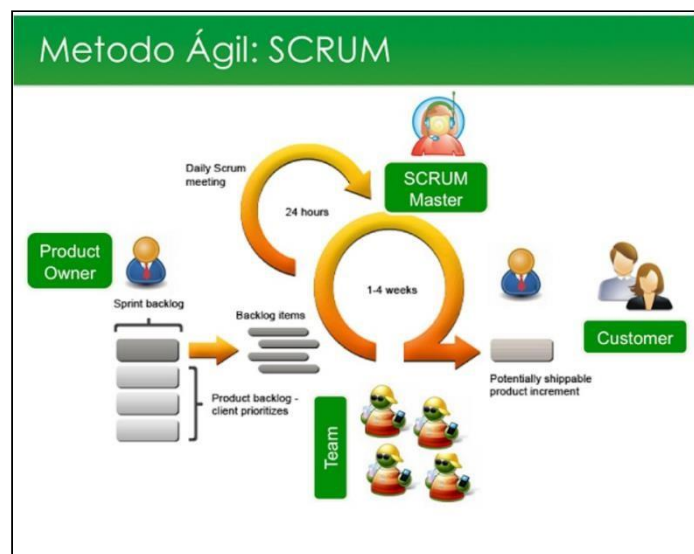
Entrega Incremental:

A lo largo de los Sprints, el sistema se entrega en incrementos funcionales, lo que permite a los usuarios comenzar a beneficiarse de las funcionalidades más importantes antes de que el proyecto esté completo.

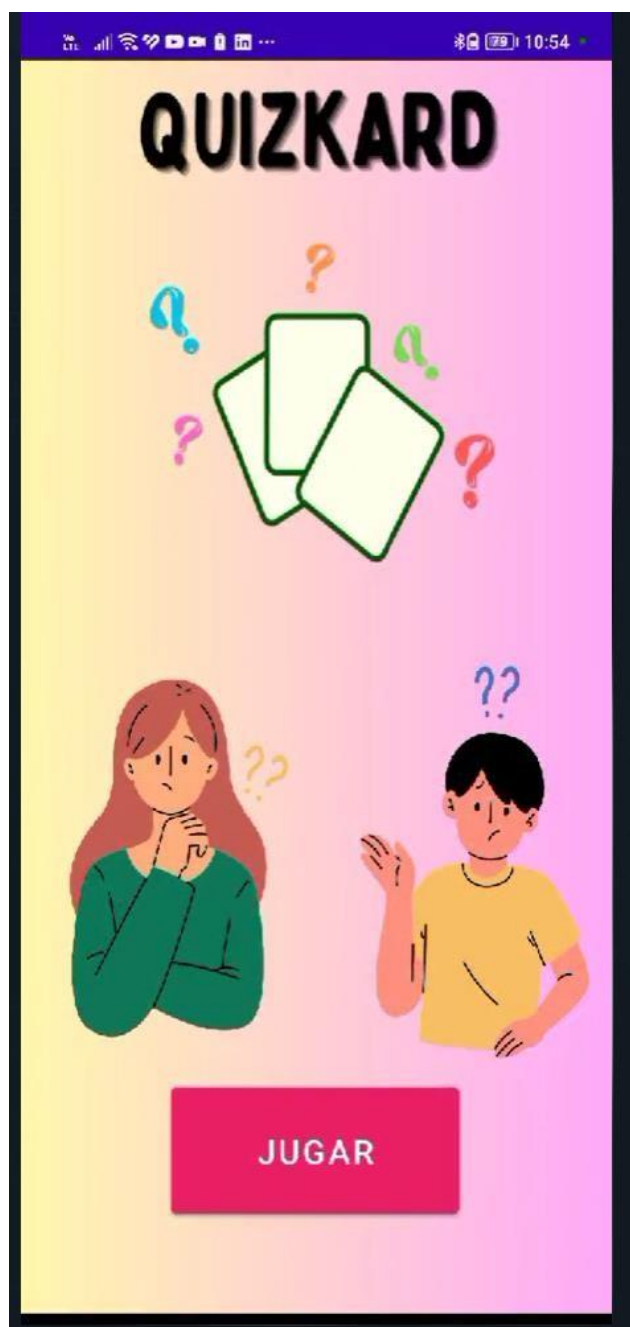
Finalización del Proyecto:

El proyecto se considera finalizado cuando se han completado todas las funcionalidades del Backlog del Producto y se ha logrado el objetivo general del proyecto.

A través de la figura 3, visualizamos la representación de la metodología SCRUM, cómo se organizan equipos de desarrollo, cómo se planifican y ejecutan las iteraciones (Sprints) y cómo se mantiene una comunicación constante con los stakeholders. Cada elemento en la figura representa un aspecto clave de la metodología SCRUM, desde el Product Backlog hasta las Reuniones Diarias y las Revisiones de Sprint.



PRIMER ENTREGABLE



10:54

Crear nuevas flashcards

La pantalla de creación de tarjetas flash te permite agregar preguntas y respuestas a tus flashcards. Una vez que hayas añadido todas las flashcards que desees, debes hacer clic en el botón "Guardar" para guardar todas las tarjetas y comenzar a jugar.

Categoría

Pregunta

Respuesta

AÑADIR OTRA PREGUNTA

GUARDAR

5G LTE

79% 10:54

Crear nuevas flashcards

La pantalla de creación de tarjetas flash te permite agregar preguntas y respuestas a tus flashcards. Una vez que hayas añadido todas las flashcards que desees, debes hacer clic en el botón "Guardar" para guardar todas las tarjetas y comenzar a jugar.

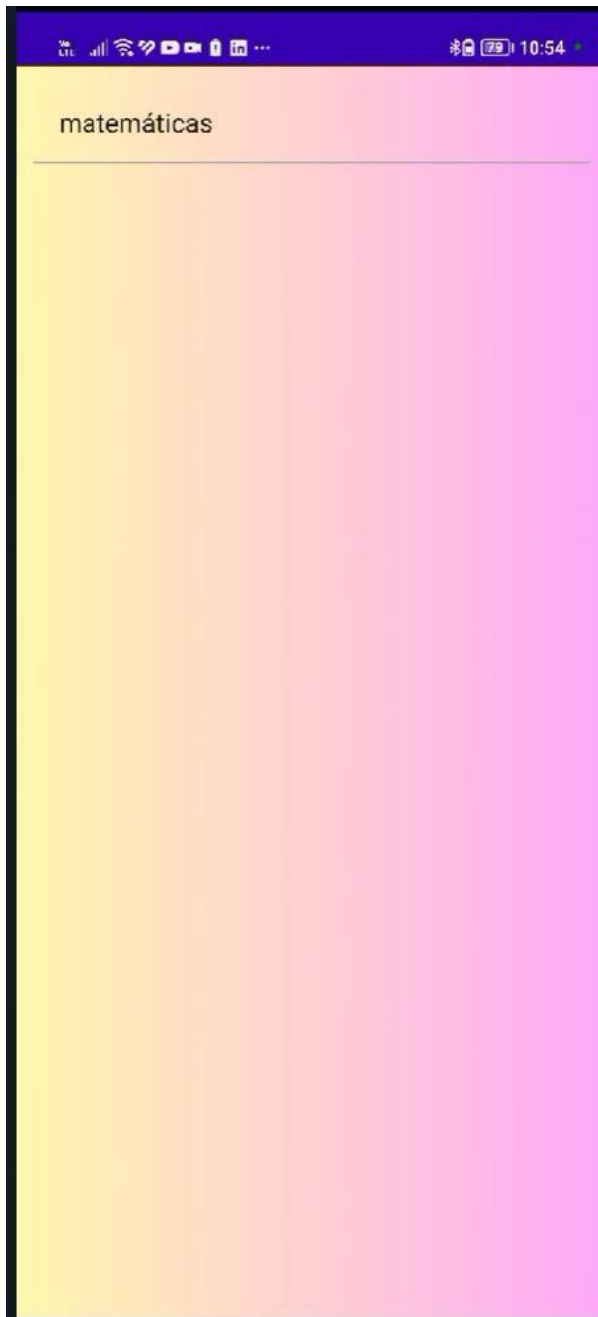
matemáticas

4+4

8

AÑADIR OTRA PREGUNTA

GUARDAR



matemáticas

$$4+4$$

MOSTRAR RESPUESTA





Organización

😊 Add icon 🖼 Add cover

Backlog semana 28 al 4 de noviembre

👤 Responsable

S susana Rivera P Pablo 🧑 Gerardo Espino Espindola
🧑 JORGE SEBASTIAN ALEJANDRO CHIU MOLINA RAMIREZ J Jesus Aranda
M Manuel Guillermo Legorreta Ramirez 🧑 Ana Sofía Espinoza Durán

⚙ Estado

● Sin empezar

☰ Resumen AI

Empty

📅 Fecha límite

Empty


S Add a comment...

- ☐ Crear interfaces(Pablo)
- ☐ Documentación(Ana)
- ☐ Manejo del git y creación de los sprint e historias de usuarios
- ☐ Implementación basica entre la pantalla del main y el quiz
- ☐ crear la acción de barajear las cartas durante el quiz
- ☐ Permitir a los usuarios voltear entre las preguntas y las respuestas cuando este tocando la tarjeta


FICHA TECNICA :


Procesador	11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz 2.42 GHz
RAM instalada	8.00 GB (7.80 GB utilizable)
Id. del dispositivo	33DFACAC-2DAA-44E5-8F73-A95740907FC4
Id. del producto	00327-31085-61112-AAOEM
Tipo de sistema	Sistema operativo de 64 bits, procesador x64
Lápiz y entrada táctil	La entrada táctil o manuscrita no está disponible para esta pantalla

Vínculos relacionados Dominio o grupo de trabajo Protección del sistema Configuración avanzada del sistema

 Especificaciones de Windows

Edición	Windows 11 Home Single Language
Versión	22H2
Se instaló el	09/02/2023
Compilación del SO	22621.2428
Experiencia	Windows Feature Experience Pack 1000.22674.1000.0
Contrato de servicios de Microsoft	
Términos de licencia del software de Microsoft	

 About Android Studio ✕



Android Studio Giraffe | 2022.3.1 Patch 2

Build #AI-223.8836.35.2231.10811636, built on September 14, 2023

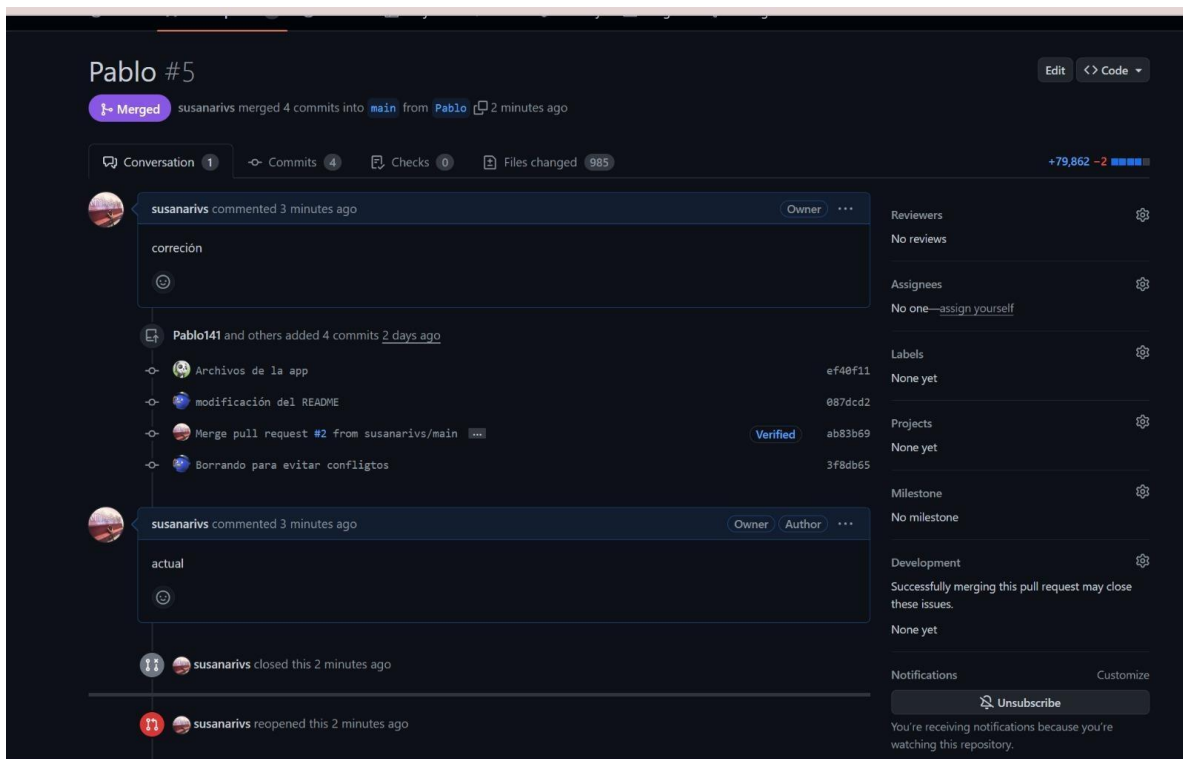
Runtime version: 17.0.6+0-b2043.56-10027231 amd64
VM: OpenJDK 64-Bit Server VM by JetBrains s.r.o.

Powered by [open-source software](#)
Copyright © 2000–2023 Google

CONCLUSIONES

El proyecto logra su objetivo de proporcionar a los usuarios una herramienta efectiva para aprender y repasar contenido importante a través de tarjetas de memoria. Esta aplicación no habría sido posible sin la colaboración y el arduo trabajo en equipo de un grupo de desarrolladores apasionados por la educación y la tecnología.

ANEXOS



LINK: <https://github.com/susanarivs/Kotlin>