

ДИПЛОМНА РАБОТА

на тема

“Разработване на мобилно приложение за споделяне на
вдъхновяващи идеи и съдържание”

Дипломант: **Мартин Венелинов Кирилов**

ученик от 12А клас

ЧПГДН "СофтУни Светлина"

Научен ръководител: Симеон Кръстев

Рецензент: Лечо Лечев

Дата: **03 май 2023**

Сесия: май-юни 2023 г.

Съдържание

Увод	3
Проблем	4
Цели на дипломния проект	5
Целта на настоящата дипломна работа е да бъде	7
Глава 1. Проучване (смени заглавието)	8
Глава 2. Кратко Overview на главните компоненти в проекта, езици и инструменти	10
Глава 2.1 Packages и необходимост в проекта	12
Глава 2.1.1 Защита на избраните пакети	14
Глава 3. Проектиране и имплементация (смени заглавието)	17
Глава 3.1 Архитектура на системата	17
Глава 3.2 Имплементация на системата	21
Глава 3.3 Тестове	25
Глава 3.4 Внедряване (deployment)	26
Глава 4. Ръководство за потребителя (смени заглавието)	28
Заклучение	29
Информационни източници	31

Увод

В настоящата епоха на интернет и социалните мрежи, споделянето на вдъхновение и идеи е съществена част от нашия онлайн опит. Милиони хора се стремят да открият нови начини за декорация на дома, мода, готвене и много други области на интерес. Въпреки това, за много потребители е предизвикателство да намерят идеи, които отговарят на техните предпочитания и стил.

Това е мястото, където клонът на Pinterest идва на помощ. Това иновативно приложение създава персонализирано и уникално пространство за вдъхновение, което отговаря на индивидуалните предпочитания на всеки потребител. Вместо да се загубват в морето от безкрайни снимки и информация, потребителите могат да открият точно това, което търсят, и да се насладят на персонализирано преживяване.

Една от най-големите предизвикателства, които срещат потребителите, е натрупването на огромно количество информация, което прави търсенето на конкретни идеи сложно и дълго. Клонът на Pinterest решава тази проблема чрез интелигентни алгоритми, които предлагат точно това, което потребителите търсят, на база на техните предпочитания и предишни дейности. Това намалява времето за търсене и дава възможност на потребителите да открият нови идеи, които съответстват на техния уникален стил.

Освен това, клонът на Pinterest предоставя възможност на потребителите да споделят своите собствени проекти и идеи с голяма общност от хора, които имат подобни интереси. Това създава единствена платформа за вдъхновение, където потребителите могат да получават обратна връзка, да се свързват с други ентусиасти и да разширяват своите знания и умения. Клонът на Pinterest създава общност от потребители, които се вдъхновяват един от друг и сътрудничат заедно.

Основният принос на клонът на Pinterest е, че предоставя решение на проблема със събирането и сортирането на информация. Потребителите вече не са принудени да търсят идеи в различни източници и да губят време в процеса. Вместо това, клонът на Pinterest предоставя единствено място, където потребителите могат да намерят всичко, от което се нуждаят, включително вдъхновяващи снимки, проекти, инструкции и връзки към ресурси.

Като клон на Pinterest, приложението предлага също така лесен начин за запазване на идеи и организиране на колекции. Потребителите могат да създават свои собствени "дъски" и да групират снимки и информация според тема или проект. Това им помага да подредят идеите си, да ги намерят бързо и да ги споделят с други.

Клонът на Pinterest също така предлага възможност за интеграция с други социални мрежи и онлайн платформи. Това позволява на потребителите да споделят идеи от Pinterest с лесно и безпроблемно, както и да имат достъп до вдъхновяващи съдържание от различни източници.

Със своите функции за персонализация, организация и споделяне, клонът на Pinterest представлява решение за проблема на потребителите да открият идеи и вдъхновение, които отговарят на техните уникални предпочитания и стил. Тази платформа създава удобна и ефективна среда за потребителите да се вдъхновяват, да създават и да споделят своите проекти и идеи с други хора.

Проблем

Проблемът разглеждан в текущата дипломна работа:

В днешната цифрова ера социалните мрежи играят важна роля в ежедневието на милиони хора по целия свят. Една от най-популярните и познати социални мрежи е Pinterest. Тя предоставя потребителите със средство за откриване, споделяне и организиране на вдъхновение и идеи в различни области на интерес. Въпреки това, в България все още липсва платформа, която да предоставя подобна функционалност и удобство на българските потребители.

Това е мястото, където нашето приложение, клонът на Pinterest, влиза в игра. Създадено специално за българската аудитория, то решава проблема с липсата на подходяща платформа за вдъхновение и споделяне на идеи. Клонът на Pinterest предоставя на потребителите възможността да откриват идеи, да създават свои собствени колекции и да споделят вдъхновение с други хора в България.

Този проблем е от особена важност и актуалност, тъй като все повече хора в България се интересуват от различни области като интериорен дизайн, мода, кулинария, градинско оформление и други. Но те се сблъскват с предизвикателството да намерят вдъхновяващи идеи, които са подходящи за българския контекст и отговарят на техните специфични предпочитания.

Така че, нашето решение е необходимо за всеки, който търси вдъхновение, идеи и нови начини за изразяване на своя стил и креативност в България. То се насочва към хора, които искат да открият нови идеи и да бъдат вдъхновени, но също така и към онези, които искат да споделят своите проекти и да се свържат с други ентусиасти в българската общност. Клонът на Pinterest е подходящ за хора от различни възрасти и интереси, които искат да бъдат вдъхновени и да се включат във визуалната общност в България.

Това приложение предлага решение не само на индивидуални потребители, но и на професионалисти от различни области. Дизайнери, стилисти, градинари, готвачи и други професионалисти могат да използват клонът на Pinterest като инструмент за изява на творчеството си, споделяне на свои проекти и установяване на връзки с клиенти и колеги.

Така че, клонът на Pinterest решава проблема с липсата на удобна и персонализирана платформа за вдъхновение и споделяне на идеи в България. Той предоставя на потребителите удобство, функционалност и възможността да се включат в общността на креативни идеи в България. Откривайте, създавайте и споделяйте с клонът на Pinterest в България!

Цели на дипломния проект

Целите на дипломния проект са следните:

Разработване на клон на Pinterest: Основната цел е да създадем функционално и изпълнено с възможности приложение, което да предоставя на потребителите в България средство за откриване, създаване и споделяне на вдъхновение и идеи, подобно на оригиналния Pinterest.

Потребителски интерфейс и лесна навигация: Желаяме да проектираме потребителски интерфейс, който е интуитивен, лесен за използване и предлага приятно преживяване на потребителите. Навигацията трябва да бъде плавна и логична, за да се осигури удобство при откриването и организирането на идеи.

Персонализация и интелигентни алгоритми: Целта е да изградим система за персонализация, която да предлага на потребителите съдържание, вдъхновение и идеи, които са специфични за техните предпочитания и интереси. Чрез използване на интелигентни алгоритми, ще се стремим да подобрим преживяването на потребителите и да им предоставим точни предложения.

Възможности за споделяне и взаимодействие: Желаем да разработим функционалности, които позволяват на потребителите да споделят свои идеи, проекти и колекции с други потребители. Възможността за коментари, оценки и създаване на общност ще помогне на потребителите да се свързват, да получават обратна връзка и да се вдъхновяват един от друг.

Адаптиране към българския контекст: Една от основните цели на проекта е да се адаптира приложението към българската култура и специфични нужди на българските потребители. Ще се стремим да предоставим съдържание, свързано с местния контекст, както и функционалности, които са полезни и удобни за българската аудитория.

Изграждане на устойчива инфраструктура: Желаем да създадем стабилна и надеждна инфраструктура, която да поддържа приложението и да осигурява безпроблемно функциониране на услугите. Това включва избор на подходящи технологии, скалируема архитектура и подходящо управление на данните и сигурността.

Тестване и оптимизация: Целта е да извършим изчерпателни тестове на приложението, за да гарантираме неговата стабилност, сигурност и пърформанс. Ще използваме техники за откриване и отстраняване на грешки, както и за оптимизиране на производителността, за да осигурим висококачествено преживяване на потребителите.

Популяризиране и маркетинг: Желаем да популяризираме приложението и да го представим на българската аудитория. Ще изградим маркетингова стратегия, която включва различни канали за реклама и популяризиране, с цел да привлечем и задържим потребители и да утвърдим клонът на Pinterest като водеща платформа за вдъхновение в България.

Целите на дипломния проект са да създадат функционално, персонализирано и лесно за използване приложение, което да отговаря на специфичните нужди на българските потребители. Проектът има за цел да предостави удобство, вдъхновение и възможност за създаване на общност, като така решава проблема с липсата на подобна платформа в България.

Целта на настоящата дипломна работа е да бъде

Designing the User Interface: Трябва да разработим интуитивен и естетично привлекателен потребителски интерфейс, който е лесен за използване и предлага приятно преживяване на потребителите. Проектирането на интерфейса трябва да отразява английската култура и да отговаря на специфичните нужди на англоговорещата аудитория.

Development of Core Functionality: Трябва да създадем функционалности, които позволяват на потребителите да откриват идеи, да създават колекции, да споделят съдържание и да се свързват с други потребители. Това включва разработка на система за търсене, съхранение на данни, система за коментари и оценки и други важни функции.

Development of Personalization System: Трябва да изградим система, която използва алгоритми и данни за персонализация, за да предлага на потребителите съдържание и идеи, които са подходящи за техните интереси и предпочитания. Това включва събиране и обработка на данни, създаване на профили на потребителите и предоставяне на персонализирани препоръки.

Testing and Optimization of the Application: Трябва да извършим изчерпателни тестове на приложението, за да гарантираме неговата стабилност, сигурност и производителност. Това включва функционални тестове, тестове на издръжливост, тестове на сигурността и тестове на производителността.

Adapting to English-Speaking Culture and Language: Трябва да се гарантира, че приложението е насочено към англоговорящата аудитория, като се вземат предвид специфичните културни аспекти и езикови особености на английския език. Това включва превод и локализация на интерфейса, предоставяне на съдържание, свързано с английската култура, и включване на регионални функции и настройки.

Documentation and Demonstration: В крайна сметка, трябва да съставим документация, която описва процеса на разработка, архитектурата на системата и функционалностите на приложението. Освен това, може да бъде необходимо да представим демонстрация на приложението пред преподаватели, колеги и потенциални потребители, за да получим обратна връзка и да демонстрираме постигнатите резултати.

Тези задачи ще ни помогнат да постигнем целите на дипломния проект и да създадем успешно приложение, което решава проблема с липсата на подобна платформа за вдъхновение сред англоговорящата аудитория.

Глава 1. Проучване (смени заглавието)

Дипломната работа се фокусира на проблема с липсата на подобна платформа за вдъхновение в България, като предлага решение във формата на клон на Pinterest, насочен към българската аудитория. Тази част на дипломната работа представя проучването, което е извършено относно наличните решения и техните предимства и недостатъци.

До момента нашето изследване показва, че в България липсва специализирана платформа за вдъхновение, която да предоставя на потребителите възможността да откриват, споделят и създават идеи в различни области като мода, интериорен дизайн, кулинария, пътувания и други. Въпреки наличието на различни социални мрежи и уебсайтове, нито един от тях не предлага комплексна платформа за съхранение и организиране на идеи, както и персонализирани препоръки, специфични за българската аудитория.

Едно от наличните решения е Pinterest самият, който е известен в световен мащаб и предлага подобни функционалности. Предимствата на Pinterest включват голямата му база от потребители, широкия спектър на тематичните категории и възможността за споделяне на идеи с други потребители. Въпреки това, за българската аудитория Pinterest има някои недостатъци. Първо, съдържанието в Pinterest е предимно на английски език, което ограничава достъпа и релевантността за българските потребители. Второ, липсва персонализиране, което означава, че потребителите получават препоръки, които не отговарят напълно на техните интереси и предпочитания.

Друго налично решение е различният уебсайт и приложения, които се фокусират върху конкретни области като мода, интериорен дизайн или кулинария. Предимствата на тези решения включват специализираната им насоченост към определени теми и “нишова” аудитории, както и наличието на специфични функционалности и съдържание, свързани със съответните области. Например, уебсайтове като "Food Network" или "Houzz" предоставят детайлни рецепти и вдъхновение за кулинарни и интериорни проекти. Въпреки това, тези решения са ограничени до конкретни области и не предлагат общоизползваема платформа за съхранение на идеи и откриване на нови интереси.

След проведеното проучване става ясно, че не съществува подобна платформа, която успешно обединява функционалността на Pinterest, като предлага персонализирани препоръки и специализирано съдържание за българската аудитория. Това открива възможност за разработката на клон на Pinterest, насочен към българската аудитория, който ще съчетае гъвкавостта и функционалността на Pinterest със специфичните нужди и предпочитания на българските потребители.

Целта на дипломния проект е да разработи и представи успешно решение, което ще попълни тази празнина на пазара, като предостави на потребителите в България интуитивна, персонализирана и цялостна платформа за съхранение, споделяне и откриване на идеи в различни области на интерес. Предвидените задачи и процеси в дипломния проект ще доведат до създаването на приложение, което ще се отличава с голяма функционалност, персонализиране и вдъхновение, специфични за българската аудитория.

Глава 2. Кратко Overview на главните компоненти в проекта, езици и инструменти

Задание на софтуера:

Целта на нашия софтуерен проект е да разработим клон на приложението Pinterest, което предоставя възможност на потребителите да откриват, споделят и организират вдъхновяващи снимки и идеи в различни категории. Системата ще предлага потребителски профили, възможност за създаване на персонализирани колекции от снимки, функционалност за споделяне на снимки с други потребители и възможност за търсене на снимки по ключови думи или категории.

Компоненти на системата и тяхното предназначение:

Входно-изходен компонент: Този компонент е отговорен за комуникацията с потребителя.

Включва екрани за вход в системата, регистрация.

Компонент за управление на потребителските профили: Този компонент се грижи за създаване, редактиране и изтриване на потребителските профили. Той също така управлява входа и изхода на потребителите в системата и се грижи за проверката на автентикацията.

Компонент за управление на снимките и колекциите: Този компонент управлява операциите, свързани със снимките и колекциите. Той позволява на потребителите да качват, преглеждат, организират и споделят свои снимки. Също така осигурява функционалност за създаване, редактиране и изтриване на колекции.

Компонент за търсене: Този компонент предоставя възможност на потребителите да извършват търсене на снимки и колекции по ключови думи или категории. Потребителите могат да използват търсачката, за да намерят конкретни снимки или да открият нови идеи в определени тематични области.

Компонент за визуализация и интерфейс: Този компонент се грижи за визуализацията на снимките, колекциите и другите елементи на потребителския интерфейс. Той включва различни екрани, списъци, картички и други елементи, които предоставят приятен и интуитивен интерфейс за потребителите.

Аргументиран избор на езика за програмиране и софтуерните средства и технологии:

Избрахме Flutter и Android Studio за разработката на клон на Pinterest поради следните причини:

Ефективност на разработката: Flutter предоставя среда за разработка, която позволява бързо и лесно създаване на красиви и мултиплатформени мобилни приложения. Един код може да се използва за създаване на приложения за Android и iOS, което намалява времето и разходите за разработката.

Привлекателен потребителски интерфейс: Flutter предоставя богата библиотека от готови компоненти и стили, които могат да бъдат използвани за създаване на привлекателен потребителски интерфейс. Това позволява на разработчиците да създават приложения с модерен и интуитивен дизайн, което е важно за успешното потребителско изживяване в приложението Pinterest.

Бързина и производителност: Flutter използва техниката на компилацията на машинен код, която осигурява бързо изпълнение на приложението. Това е от съществено значение за приложенията, като клон на Pinterest, които работят с голям обем от данни и изображения.

Интеграция със сървърни технологии: Flutter предоставя лесна интеграция с различни сървърни технологии чрез API извиквания. Това позволява приложението да комуникира със сървър и да извлича и обработва необходимата информация за снимки, потребители и колекции.

Поддръжка и разширяемост: Flutter има активна общност от разработчици и предлага множество библиотеки и инструменти, които улесняват разработката и поддръжката на приложението. Това предоставя възможности за разширяване на функционалността и бързо реагиране на променящите се изисквания и тенденции.

Техническите решения, които ще приложим в проекта, включват използването на Firebase за управление на потребителските профили, аутентикацията и съхранението на изображения. Също така, ще използваме библиотеката ImagePicker за избор и обработка на снимки, ImageCropper за изрязване на изображенията и Flutter Firestore за съхранение и извличане на данни от базата данни. Тези технически решения ще осигурят функционалността и ефективността на нашето приложение за клон на Pinterest.

Глава 2.1 Packages и необходимост в проекта

```
cupertino_icons: ^1.0.2
firebase_auth: ^4.6.1
firebase_core: ^2.13.0
cloud_firestore: ^4.7.1
firebase_storage: ^11.2.1
uuid: ^3.0.7
fluttertoast: ^8.2.1
flutter_staggered_grid_view: ^0.6.2
transparent_image: ^2.0.1
image_picker: ^0.8.7+5
image_cropper: ^4.0.1
image_downloader: ^0.31.0
intl: ^0.18.1
```

Използваните пакети и техните функции в проекта на клон на Pinterest са следните:

cupertino_icons: ^1.0.2 - Този пакет предоставя икони, специфични за iOS, които са необходими за изграждането на визуалния интерфейс в стила на Cupertino (iOS).

`firebase_auth`: ^4.6.1 - Този пакет предоставя библиотека за аутентикация с Firebase. Използва се за управление на потребителските профили и аутентикацията на потребителите в приложението.

`firebase_core`: ^2.13.0 - Този пакет предоставя основната функционалност на Firebase, като инициализиране и конфигуриране на приложението за работа с Firebase услугите.

`cloud_firestore`: ^4.7.1 - Този пакет предоставя библиотека за работа със сървърна база данни на Firestore от Firebase. Използва се за съхранение и извличане на данни, например снимки, потребители и колекции.

`firebase_storage`: ^11.2.1 - Този пакет предоставя функционалност за съхранение на файлове в Firebase Storage. Използва се за качване и изтегляне на изображения в приложението.

`uuid`: ^3.0.7 - Този пакет предоставя уникални идентификатори (UUID). Използва се за генериране на уникални идентификатори за снимките и потребителите в приложението.

`fluttertoast`: ^8.2.1 - Този пакет предоставя функционалност за показване на тост съобщения (плаващи съобщения) във Flutter приложението. Използва се за показване на съобщения и предупреждения към потребителите.

`flutter_staggered_grid_view`: ^0.6.2 - Този пакет предоставя `Widget` за изграждане на разширяем и разпределен се тип на мозайка във Flutter приложението. Използва се за отражение на мозайчатата галерия на снимки в приложението.

`transparent_image`: ^2.0.1 - Този пакет предоставя виджет за зареждане на прозрачни изображения във Flutter приложението. Използва се за зареждане `image_picker`: ^0.8.7+5 - Този пакет предоставя функционалност за избор на изображения от галерията на устройството или снимане на ново изображение с камерата. Използва се за позволяване на потребителите да качват снимки в приложението.

`image_cropper: ^4.0.1` - Този пакет предоставя възможност за изрязване и редактиране на изображения. Използва се за обработка на изображенията, качени от потребителите, преди да бъдат запазени или показвани в приложението.

`image_downloader: ^0.31.0` - Този пакет предоставя функционалност за изтегляне на изображения от интернет. Използва се за изтегляне на снимки от уеб източници и запазването им в устройството на потребителя.

`intl: ^0.18.1` - Този пакет предоставя локализация и форматиране на данни. Използва се за превод и форматиране на текстови съобщения и дати, съответно за многоезичност и регионални настройки на приложението.

Глава 2.1.1 Защита на избраните пакети

Тези пакети бяха избрани и включени в проекта, за да предоставят необходимата функционалност за клон на Pinterest. Те позволяват на потребителите да се аутентикират, да качват и споделят изображения, да преглеждат и търсят колекции от снимки, както и да изтеглят и запазват изображения. Също така, те поддържат визуалното представяне на данните и улесняват манипулацията и обработката на изображенията.

Изборът на езика за програмиране - Dart и средата за разработка - Android Studio, е основан на факта, че Dart е официалният език на Flutter и предоставя богата библиотека от инструменти и функционалности за разработка на крос-платформени мобилни приложения. Android Studio, от своя страна, е мощна и популярна интегрирана среда за разработка (IDE), която предлага богати възможности за разработка, отстраняване на грешки и тестване на приложения за Android.

Тези технически решения - използването на Firebase услугите (Firebase Auth, Cloud Firestore, Firebase Storage) предоставят надеждна инфраструктура за управление на аутентикацията на потребителите, съхранение на данни и управление на медийни файлове. Използването на Firebase позволява бързо и лесно свързване на приложението с облачни услуги, което е от съществено значение за функционалността на клон на Pinterest, където потребителите качват и споделят снимки.

Пакетът `uuid` се използва за генериране на уникални идентификатори, които се присвояват на снимките и потребителите. Това е важно, за да се гарантира, че в системата няма дублиращи се идентификатори и да се осигури еднозначно идентифициране на ресурсите.

Пакетът `fluttertoast` се използва за показване на визуални съобщения (тостове) в приложението. Това позволява да се информира потребителя за успешни операции, грешки или други важни съобщения по време на използването на приложението.

Пакетът `flutter_staggered_grid_view` предоставя мощен и гъвкав `Widget` за изграждане на мозайчати галерии и разпределението на снимките в тях. Това е важно за представянето на снимките в клон на `Pinterest`, където мозайката е основен елемент на интерфейса.

Пакетът `transparent_image` се използва за зареждане на прозрачни изображения в приложението. Това е полезно, когато потребителите качват или преглеждат снимки с прозрачен фон, като лога или дизайнерски елементи.

Пакетът `image_picker` предоставя функционалност за избор на изображения от галерията или снимане с камерата на устройството. Това е важно за позволяване на потребителите да качват собствени снимки или да изберат снимки за споделяне.

Пакетът `image_cropper` се използва за редактиране и изрязване на изображения, качени от потребителите. Това позволява на потребителите да настройват, режат и обработват своите снимки преди да ги качат в приложението. Това е полезно за подобряване на качеството и визуалния аспект на снимките.

Пакетът `image_downloader` се използва за изтегляне на изображения от интернет. Това е полезно, когато приложението предоставя възможност за изтегляне на снимки от уеб източници, като например от външни сайтове или споделени албуми.

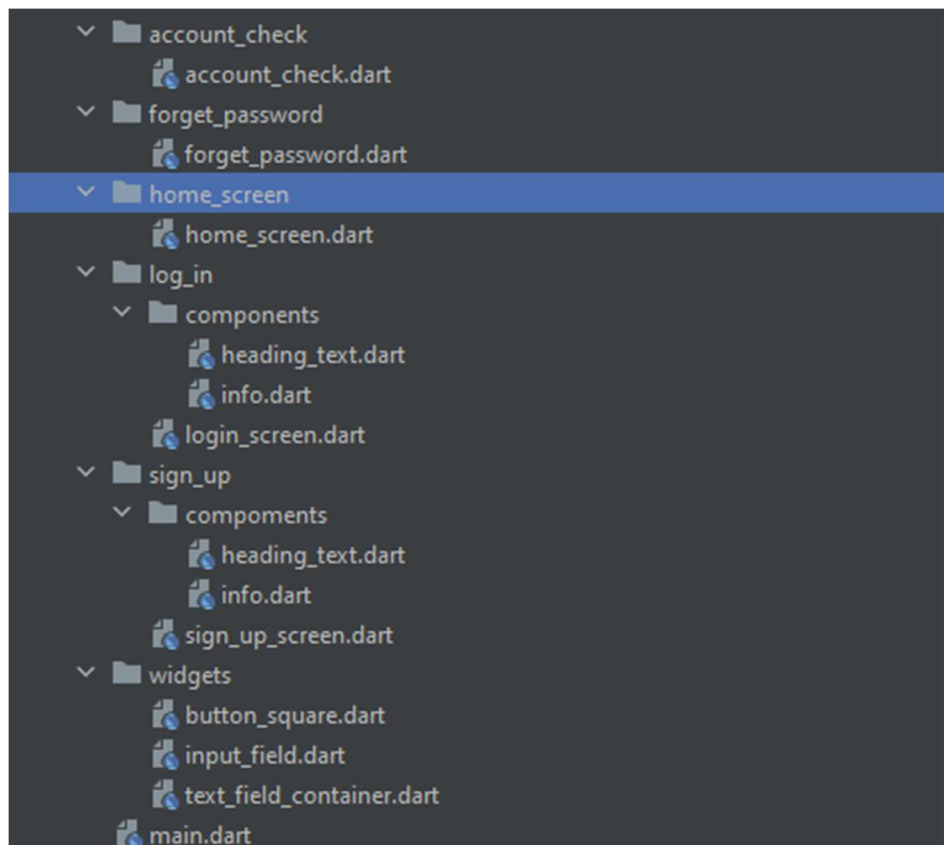
Пакетът intl се използва за локализация и форматиране на данни. Това позволява на приложението да бъде преведено на различни езици и да се представят данни, като дати и време, в съответствие с регионалните настройки на потребителите.

Тези пакети се използват за добавяне на различни функционалности в клон на Pinterest приложението, като управление на аутентикацията, съхранение и извличане на данни, обработка на изображения и подобряване на визуалния интерфейс. Използването на готови пакети е полезно, тъй като те предлагат готови решения и спестяват време и усилия при разработката на приложението.

Глава 3. Проектиране и имплементация (смени заглавието)

Глава 3.1 Архитектура на системата

Архитектура на документите:



```

const SizedBox(height: 15.0,),
ButtonSquare(
  text: "Create Account",
  colors1: Colors.red,
  colors2: Colors.redAccent,

  press: () async{
    if(imageFile == null) {
      Fluttertoast.showToast(msg: "Please select an Image");
      return;
    }
    try
    {
      final ref= FirebaseStorage.instance.ref().child('userImages').child(DateTime.now().toString() + '.jpg');
      await ref.putFile(imageFile!);
      imageUrl=await ref.getDownloadURL();
      await _auth.createUserWithEmailAndPassword(
        email: _emailTextController.text.trim().toLowerCase(),
        password: _passController.text.trim(),
      );
      final User? user=_auth.currentUser;
      final _uid=user!.uid;
      FirebaseFirestore.instance.collection('users').doc(_uid).set({
        'id':_uid,
        'userImage':imageUrl,
        'name': _fullNameController.text,
        'email': _emailTextController.text,
        'phoneNumber': _phoneController.text,
        'createAt': Timestamp.now(),
      });
      Navigator.canPop(context) ? Navigator.pop(context):null;
    }
    catch(error){
      Fluttertoast.showToast(msg: error.toString());
    }
  }
//create Homepage

```

Регистриране на профилски акаунт и изпращане на информацията към Firebase Firestore,където се създава сървър ,в които се създава папка(таблица) users със следните елементи:

-Unique Id

-Username

-Име

-Парола

-Емайл

-Телефонен номер

-Кога е създаден профила чрез Timestamp.now()-

Input полетата, където се въвежда информацията от потребителя:

```

Widget build(BuildContext context) {
  return Padding(
    padding: const EdgeInsets.all(50.0),
    child: Column(
      crossAxisAlignment: CrossAxisAlignment.center,
      children: [
        GestureDetector(
          onTap: ()
          {
            _showImageDialog();
          },
          child: CircleAvatar(
            radius: 90,
            backgroundImage:imageFile==null
              ?
                const AssetImage("images/avatar.png")
              :
                Image.file(imageFile!).image,
          ), // CircleAvatar
        ), // GestureDetector
        const SizedBox(height: 10.0,),
        InputField(
          hintText: "Enter Username"
            , icon: Icons.person
            , obscureText: false,
            textEditingController: _fullNameController,
        ), // InputField
        const SizedBox(height: 10.0,),
        InputField(
          hintText: "Enter Email",
          icon: Icons.email_rounded,
          obscureText: false,
          textEditingController: _emailTextController,
        ), // InputField
        const SizedBox(height: 10.0,),
        InputField(
          hintText: "Enter Password",
          icon: Icons.lock,
          obscureText: true,
          textEditingController: _passController,
        ), // InputField
        const SizedBox(height: 10.0,),
        InputField(
          hintText: "Enter Phone number",
          icon: Icons.phone,
          obscureText: false,
          textEditingController: _phoneController,
        ), // InputField
      ],
    ),
  );
}

```

Ъпдейт на снимката подадена от клиента.

Праща новата снимка към сървъра.

```
void _updateImageInFirestore() async {
  String fileName = DateTime
    .now()
    .microsecondsSinceEpoch
    .toString();
  fStorage.Reference reference=fStorage.FirebaseStorage.instance.ref()
    .child("userImages").child(fileName);
  fStorage.UploadTask uploadTask=reference.putFile(File(imageXFile!.path));
  fStorage.TaskSnapshot taskSnapshot=await uploadTask.whenComplete(() => {});
  await taskSnapshot.ref.getDownloadURL().then((url) async{
    userImageUrl=url;
  });
  await FirebaseFirestore.instance.collection("users").doc(FirebaseAuth.instance.currentUser!.uid).update(
    {
      'userImage':userImageUrl,
    }).whenComplete(){
    updateProfileImageOnUserExistingPosts();
  });
}

updateProfileImageOnUserExistingPosts() async{
  await FirebaseFirestore.instance.collection('wallpaper')
    .where("id",isEqualTo: FirebaseAuth.instance.currentUser!.uid)
    .get()
    .then((snapshot){
      for(int index=0;index<snapshot.docs.length; index++){
        String userProfileImageInPost= snapshot.docs[index]['userImage'];

        if(userProfileImageInPost != userImageUrl){
          FirebaseFirestore.instance.collection('wallpaper').doc(snapshot.docs[index].id)
            .update({
              'userImage':userImageUrl,
            });
        }
      }
    });
}
```

Глава 3.2 Имплементация на системата

Стартиране на приложението:

```
import 'package:firebase_core/firebase_core.dart';
import 'package:flutter/material.dart';
import 'package:project_flutter/log_in/login_screen.dart';

void main() {
  WidgetsFlutterBinding.ensureInitialized();
  runApp(MyApp());
}

class MyApp extends StatelessWidget {

  final Future<FirebaseApp> _inianlization=Firebase.initializeApp();

  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return FutureBuilder(
      future: _inianlization,
      builder: (context, snapshot)
      {
        if(snapshot.connectionState == ConnectionState.waiting){
          return const MaterialApp(
            debugShowCheckedModeBanner: false,
            home: Scaffold(
              body: Center(
                child:Center(
                  child: Text("Welcome to PhotoSharing Clone App"),
                ), // Center
              ), // Scaffold
            ); // MaterialApp
          }
          else if (snapshot.hasError){
            return const MaterialApp(
              debugShowCheckedModeBanner: false,
              home: Scaffold(
                body: Center(
                  child:Center(
                    child: Text("An Error ocured,Please wait"),
                  ), // Center
                ), // Scaffold
              ); // MaterialApp
            )
          }
          return MaterialApp(
            debugShowCheckedModeBanner: false,
            title: "Flutter PhotoSharing Clone App",
            home: LoginScreen(),
          ); // MaterialApp
        }
      ); // FutureBuilder
    }
  }
}
```

Кода има

Кода има за цел да инициализира Firebase в приложението и да покаже начален екран на базата на състоянието на връзката с Firebase.

Основната функция `main()` задейства инициализацията на Flutter и стартира главното приложение `MyApp`.

Класът `MyApp` е наследник на `StatelessWidget` и представлява кореновия виджет на приложението. В метода `build()` се използва `FutureBuilder`, който изчаква завършването на инициализацията на Firebase (`_inianlization`) и базирано на състоянието на връзката (`snapshot.connectionState`) рендира съответния виджет.

Ако връзката все още се установява, се визуализира начален екран с текст "Welcome to Pinterest-2.0 App". В случай на грешка при връзката, се показва съобщение за грешка с текст "An Error occurred, Please wait". Ако връзката е установена успешно, се показва основният екран на приложението - `LoginScreen()`.

Целта на този код е да осигури плавен стартиращ екран на приложението и да се гарантира, че Firebase е успешно инициализирана преди да се покаже основният екран на приложението.

Проверка на акаунта:

```
class AccountCheck extends StatelessWidget {  
  
  final bool login;  
  final VoidCallback press;  
  
  AccountCheck({required this.login,required this.press});  
  
  @override  
  Widget build(BuildContext context) {  
    return Row(  
      mainAxisAlignment: MainAxisAlignment.center,  
      children: [  
        Text(  
          login ? "Do not have an Account" : "Already have an Account?",  
          style: const TextStyle(fontSize: 16.0, color: Colors.white  
            ), // TextStyle  
        ), // Text  
        const SizedBox(width: 10.0,),  
        GestureDetector(  
          onTap: press,  
          child: Text(  
            login ? "Create an Account" : "Log in",  
            style : const TextStyle(fontSize : 16.0,color: Colors.blue, fontWeight :FontWeight.bold ),  
          ), // Text  
        ), // GestureDetector  
        const SizedBox(height: 50.0,),  
      ],  
    ); // Row  
  }  
}
```

Кода представлява клас AccountCheck, който е наследник на StatelessWidget. Този клас съдържа виджет, който показва линк към създаване на нов акаунт или вход в съществуващ акаунт в зависимост от стойността на променливата login.

Конструкторът на класа приема два параметъра: login (булева стойност) и press (функция без връщан тип). login определя дали потребителят е в режим на вход (true) или създаване на акаунт (false). press е функцията, която се извиква при натискане на линка.

Методът build() създава и връща Widget-а Row, който съдържа два текстови елемента: първият текст е условен и зависи от стойността на login, а вторият текст представлява линк, който извиква функцията

press при натискане. Текстовите стилове са зададени чрез TextStyle. Визуалното оформление на компонентите се контролира чрез подравняване (mainAxisAlignment) и отстояния (SizedBox).

Този клас се използва за показване на линк за превключване между формата за вход и формата за създаване на акаунт в приложението.

Визия на текст контейнирите

```
class TextFieldContainer extends StatelessWidget {  
  
  final Widget child;  
  
  TextFieldContainer({required this.child});  
  
  @override  
  Widget build(BuildContext context) {  
    return Container(  
      width: double.infinity,  
      margin: const EdgeInsets.symmetric(vertical: 15.0),  
      decoration: BoxDecoration(  
        gradient: const LinearGradient(  
          begin: Alignment.centerLeft,  
          end: Alignment.centerRight,  
          colors: [Colors.redAccent, Colors.red],  
        ), // LinearGradient  
        borderRadius: BorderRadius.circular(12),  
        boxShadow: const [  
          BoxShadow(  
            offset: Offset(-2, -2),  
            spreadRadius: 1,  
            blurRadius: 4,  
            color: Colors.red,  
          ), // BoxShadow  
          BoxShadow(  
            offset: Offset(2, 2),  
            spreadRadius: 1,  
            blurRadius: 4,  
            color: Colors.redAccent,  
          ), // BoxShadow  
        ],  
      ), // BoxDecoration  
      child: child,  
    ); // Container  
  }  
}
```

Кода представлява клас `TextFieldContainer`, който е наследник на `StatelessWidget`. Този клас представлява контейнер, който обгражда и стилизира даден виджет (подаден като `child`) за полето за текст.

Конструкторът на класа приема един параметър - `child`, който е виджетът, който ще бъде показан вътре в контейнера.

Методът `build()` създава и връща контейнера със следните стилове и настройки:

`width` е зададен като `double.infinity`, което означава, че контейнера ще заема цялата налична ширина.

`margin` задава външни отстъпи от горе и долу на контейнера.

`decoration` представлява стилизацията на контейнера. В този случай, се използва градиент (`LinearGradient`) от два цвята - червен и червено-червен. Също така, има закръглени ъгли на контейнера чрез `borderRadius`. Освен това, се добавят сянки (`BoxShadow`) отстрани на контейнера, създавайки ефект на издигане и вдлъбване на контейнера.

Вътре в контейнера се показва виджетът `child`, който е подаден чрез конструктора.

Този клас се използва за стилизиране на полетата за текст в приложението. Компонентът позволява лесно и единно стилизиране на полетата и може да се използва повторно в различни части на приложението.

Глава 3.3 Тестове

При тестването на приложението за клон на Pinterest могат са били използвани тези техники за уверяване за правилната работа на апликацията.

Ръчно тестване: Използване на ръчни тестове, където тестерите изпълняват различни сценарии на приложението, за да проверят функционалността и да открият грешки и проблеми.

Функционално тестване: Проверка на основните функции на приложението, като се уверим, че те работят правилно и отговарят на очакванията на потребителите.

Интеграционно тестване: Проверка на взаимодействието между различните компоненти и модули на приложението, за да се уверим, че те работят безпроблемно заедно.

Удобство на потребителския интерфейс: Тестване на потребителския интерфейс на приложението, за да се провери неговата функционалност, удобство и навигация. Включва тестове за увереност, валидация на полета, проверка на реакцията на екраните и други аспекти на потребителското изживяване.

Изпитване на производителността: Тестване на производителността на приложението, като се проверява времето за отговор, скоростта на зареждане на данни и използването на ресурси като памет и батерия.

Тези техники се комбинират и допълват в зависимост от конкретните изисквания и цели на тестовия процес.

Глава 3.4 Внедряване (deployment)

За да разгърнете своята Flutter приложение чрез Android Studio, можете да следвате следните стъпки:

Отворете вашия Flutter проект в Android Studio.

Уверете се, че вашият Flutter проект е конфигуриран правилно и всички зависимости са инсталирани.

Свържете вашия Android устройство с компютъра чрез USB кабел или създайте виртуално устройство за Android (AVD).

Отворете "Terminal" в Android Studio, като кликнете върху иконата на терминала в долния десен ъгъл на прозореца.

В терминала изпълнете командата `flutter build apk` за създаване на APK файл за вашето приложение. Ако искате да създадете Bundle файл, използвайте командата `flutter build appbundle`.

След като се генерира APK или Bundle файлът, ще го намерите в папката "build" във вашия проект.

За да инсталирате приложението на своето Android устройство, свържете го към компютъра чрез USB кабел и изпълнете командата `flutter install` в терминала.

Приложението ще бъде инсталирано на вашия Android устройство и ще можете да го използвате.

Това са основните стъпки за конфигурация на приложението в Android Studio. Можете да изследвате допълнителни опции за конфигурация на приложението, като например подписване на приложението и качване в Google Play Store, в зависимост от вашите конкретни изисквания.

Глава 4. Ръководство за потребителя (смени заглавието)

Приложението може да бъде открито в гитхъб на този линк:

https://github.com/martink2004/Clone_Pinterest.git

Има начална страница където ще ви помоли да си направите акаунт и да се регистрирате. В случай че вече имате трябва да се логнвате с регистрираната ви информация.

За качването на снимки и информация. Има два бутона един да изберете от галерията си или да направите снимка. След пращането на снимката тя вече се запазва на сървъра и имате неограничен достъп до тях.

Заклучение

Клонът на Pinterest представлява мощен инструмент за създаване, споделяне и откриване на вдъхновяващи идеи и визуално съдържание. Той предоставя на потребителите възможността да събират и организират снимки, изображения и визуални материали от цялото интернет пространство и да ги споделят с други потребители.

Използването на клонът на Pinterest има множество ползи и приложения. Потребителите могат да използват платформата за инспирация и съхранение на идеи за проекти, мода, интериорен дизайн, кулинарни рецепти, пътувания и много други. Освен това, клонът на Pinterest създава общност, където потребителите могат да се свързват и споделят своите интереси и страсти.

В бъдеще се очаква да видим някои подобрения и иновации в клонът на Pinterest. Някои от тези подобрения могат да включват:

Разширяване на функционалността: Допълнителни функции и възможности, които дават на потребителите по-голяма свобода и гъвкавост при съхранение, споделяне и откриване на съдържание.

Подобрено откриване и препоръки: Постоянно развиване на алгоритмите за препоръки и търсене, за да се предоставят по-персонализирани и релевантни съдържание на потребителите.

Визуални подобрения: Подобрена визуална естетика и потребителски интерфейс, които правят използването на приложението по-приятно и интуитивно.

Интеграция с други платформи: Възможност за интеграция с други социални медии и онлайн платформи, което позволява на потребителите да споделят своето съдържание по-широко и да го взаимодействат с други платформи.

Клонът на Pinterest има голям потенциал за развитие и разширение, като продължава да се адаптира към променящите се потребителски нужди и предпочитания. С подобренията в функционалността, интерфейса и възможностите за откриване на съдържание, клонът на Pinterest може да продължи да бъде предпочитана платформа за вдъхновение и споделяне на идеи.

Бъдещите подобрения могат да се фокусират също така върху подобряването на социалната взаимодействие, създаването на по-богати и персонализирани потребителски профили, както и разширяването на възможностите за съхранение и организация на съдържанието.

Освен това, с развитието на технологиите и нарастващата важност на визуалния контент, клонът на Pinterest може да се възползва от напредъка в областта на компютърното зрение и разпознаването на образи, което ще позволи по-добро разпознаване и етикетиране на изображенията.

В крайна сметка, клонът на Pinterest остава изключително полезен и популярен инструмент за споделяне и откриване на вдъхновяващо съдържание. С развитието на технологиите и отговарянето на потребителските изисквания, приложението има потенциала да продължи да е незаменимо средство за търсене на визуална инспирация и общуване с общности от хора с подобни интереси.

Информационни източници

Flutter Official Documentation - <https://flutter.dev/docs>

Flutter YouTube Channe- <https://www.youtube.com/c/FlutterDev>

Android Studio Documentation- <https://developer.android.com/studio/intro>

Firebase Documentation- <https://firebase.google.com/docs>

Firebase YouTube Channel- <https://www.youtube.com/c/Firebase>

И други.