Wallety Final Project Book

by
Shani Bider
Erez Leviim
Shira Balali

Supervisor: Dr. Igor Rochlin

Submitted to the Computer Science Faculty of College of Management
August 2023

Executive Summary

Wallety, Our Android app, *"Digital Wallet for Families,"* is designed to revolutionize financial management within families. With a focus on enhancing financial transparency and promoting responsible behavior, our app simplifies expense monitoring, secure money transfers, savings management, and parental supervision.

The main objective of our app is to provide families with a comprehensive tool for effortless expense monitoring. By easily tracking and categorizing expenditures, users gain valuable insights into their spending patterns. This allows for informed decision-making to optimize their budget and achieve financial goals.

One of the standout features of our app is its secure and convenient money transfer capability. Whether it's transferring funds for expenses or transferring pocket money to children, users can seamlessly move money between family members. This eliminates the need for physical cash exchanges and promotes financial transparency within the household.

We understand the importance of instilling financial responsibility in children from an early age. Our app allows parents to assign monetary values to specific tasks, incentivizing their kids to complete chores and earn rewards. This not only instills a sense of accomplishment but also teaches children the value of hard work and responsibility.

To further promote financial discipline, our app includes a dedicated savings account feature. Children can set financial goals, track their progress, and watch their savings grow over time. This fosters a habit of saving and teaches them the importance of long-term financial planning.

In addition, our app helps parents actively monitor and supervise their family's financial activities. With customized thresholds, that set predefined limits according to their specific preferences, parents receive notifications when expenditures exceed predefined limits. This proactive approach enables responsible spending and ensures prompt action to address any financial challenges.

Moreover, the app provides a historical overview of expenses, allowing parents to analyze spending patterns over time. This data empowers them to identify trends, pinpoint areas where adjustments can be made, and make informed financial decisions.

Overall, the Digital Wallet for Families app serves as a comprehensive tool to empower families with essential money management skills. With its user-friendly interface and powerful features, it simplifies financial transactions, encourages responsible behavior, and cultivates lasting financial habits. By using our app, families can build confidence in financial decision-making and ensure a brighter financial future for everyone involved.

Table of Contents

Introduction 5

1.1. Background 5

1.2. Problem Statement 7

1.3. Objectives 9

1.4. Scope and Limitations 10

1.5. Methodology 11

1.6. Organization of the Project Book 13

2. Literature Review 15

Overview of Relevant Literature: 15

3. System Design and Implementation 18

3.1 System Architecture 18

3.3. Evaluation Metrics 23

4. Project Results and Analysis 24

5. Additional material: 34

6. Conclusion and Future Work 49

7. References 51

Introduction

This section provides background information, clearly defines the problem, and explain the objectives and scope of your project. It sets the stage for the rest of your project book, providing a foundation for readers to understand the purpose and significance of your work.

1.1. Background

In today's increasingly digital world, financial management within families has become more complex. Traditional methods of handling expenses, transfers, and savings are often cumbersome and lack transparency. Moreover, the need to teach children about responsible money management from an early age has gained prominence.

The advent of smartphones and mobile applications has brought about a paradigm shift in the way we manage our finances. The proliferation of digital wallets and payment apps has made financial transactions more convenient and accessible. However, there remains a gap in the market for a comprehensive solution specifically designed to address the unique financial management needs of families.

This background sets the stage for the development of "Wallety," our Android app, digital Wallet for Families. It is a groundbreaking solution that aims to enhance financial management within families by revolutionizing the way expenses, transfers, and savings are handled.

The context of our project is rooted in the increasing complexity of family finances. In today's fast-paced world, families juggle multiple sources of income, shared expenses,

and individual financial goals. There is a growing need for a centralized platform that provides transparency, simplifies transactions, and fosters responsible financial behavior within the family unit.

Furthermore, societal trends emphasize the importance of teaching children about money management and instilling financial responsibility from an early age. The rapid digitization of financial transactions presents an opportunity to leverage technology to educate and empower children in their financial journey.

Our research and development process were motivated by these industry trends, technological advancements, and societal needs. By analyzing the existing landscape of financial management apps and identifying the gaps, we aimed to create a comprehensive solution tailored specifically for families.

In conclusion, the background of our project highlights the evolving nature of family finances, the opportunities presented by technological advancements, and the societal need for a dedicated solution to enhance financial management within families. "Wallety", aims to address these challenges and provide families with a powerful tool to simplify financial transactions, promote responsible behavior, and cultivate lasting financial habits.

1.2. Problem Statement

The problem that our project aims to solve is **the lack of a comprehensive financial management solution specifically designed for families.** Traditional methods of handling expenses, transfers, and savings within households are often unmanageable, time-consuming, and lack transparency. This poses significant challenges for families in effectively managing their finances and instilling financial discipline in their children.

The specific challenges and issues associated with the problem include:

1. Complex Expense Tracking: Families struggle to track and categorize their expenses effectively. Manual methods such as paper-based tracking or spreadsheets are prone to errors, and it becomes challenging to gain a clear understanding of where the money is being spent. This lack of visibility hampers informed decision-making and budget optimization.

2. Inefficient Money Transfers: Families face difficulties when it comes to transferring money between family members. Conventional methods like physical cash exchanges or bank transfers are often time-consuming, inconvenient, and lack the transparency needed to manage shared expenses effectively.

3. Limited Financial Education for Children: Many families recognize the importance of teaching children about financial responsibility from an early age. However, there is a lack of dedicated tools to support this effort. Children often lack a platform to learn about budgeting, saving, and responsible spending, leading to a limited understanding of financial management principles.

4. Absence of Monitoring and Supervision: Parents face challenges in monitoring and supervising their family's financial activities. Without a centralized platform, it becomes difficult to keep track of expenses, identify overspending, and ensure responsible financial behavior within the household.

5. Including house chores **in one app** alongside saving accounts and regular money transfers, Wallety provides families with a centralized platform where all aspects of financial management, including chores, are easily accessible and transparent. This integration ensures that house chores are upfront and obvious, simplifying the overall financial management process for families.

It is essential to address these challenges and issues to empower families with effective financial management tools and skills. By providing a comprehensive solution tailored for families, we aim to simplify expense tracking, streamline money transfers, instill financial education in children, enable parents to monitor and guide their family's financial activities, and make it all in one easy to use app.

Addressing these problems will not only enhance financial transparency and decision-making within families but also promote responsible spending habits and financial independence.

1.3. Objectives

The objectives of our project are focused on addressing the challenges of fragmented financial management within families. These objectives are clear, measurable, and aligned with the problem statement. They include:

1. Combination: Our first objective is to consolidate various financial management aspects, including chores, saving accounts, and regular money transfers, into one app. By providing a centralized platform, we aim to simplify and streamline the financial management process for families, and group several daily routine operations into one application.

2. Convenience and Accessibility: We strive to enhance convenience and accessibility for families by making all financial management features easily accessible within the app. This includes ensuring that house chores are upfront and obvious, allowing families to effortlessly track and manage their tasks alongside other financial activities.

3. Financial Transparency: Our objective is to promote financial transparency within families. By providing a comprehensive overview of expenses, money transfers, and savings, we empower families to have a clear understanding of their financial situation and make informed decisions, with ease with a friendly user interface.

4. Financial Education: We aim to foster financial education and responsibility, particularly for children. Our objective is to create a platform that encourages children to learn about money management, budgeting, and saving through features such as assigning monetary values to chores and providing dedicated saving accounts.

5. Simplified Transactions: Our objective is to simplify financial transactions within families. By facilitating secure and convenient money transfers between family members, we aim to eliminate the need for physical cash exchanges and make transactions seamless and hassle-free.

6. Monitoring and Supervision: We strive to enable parents to actively monitor and supervise their family's financial activities. By providing customizable alerts and thresholds, our objective is to empower parents to proactively manage expenses, identify potential financial risks, and promote responsible spending habits.

These objectives guide our research and implementation process, ensuring that we focus on creating a comprehensive solution that addresses the challenges families face in financial management. By achieving these objectives, we aim to provide families with a powerful tool, Wallety, that simplifies financial processes, enhances transparency, promotes financial education, and empowers families to achieve their financial goals.

1.4. Scope and Limitations

The scope of our project, "Wallety: Digital Wallet for Families" including the development and implementation of an **Android app that addresses the financial management needs of families**. Our focus is on integrating features such as expense tracking, money transfers, saving accounts, and a chores section within the app. We aim to provide a user-friendly interface and comprehensive functionality within these areas. However, it is important to note that our project does not cover advanced financial

planning or investment strategies. Additionally, while our app is designed for families, it may not cater to every specific family structure or financial scenario. We acknowledge that there may be limitations and constraints, such as technical limitations or data security considerations, which could impact the scope and outcomes of our project. Nonetheless, we are committed to delivering a valuable and user-centric solution within the defined boundaries.

It's important to note that the application serves as a demo app, intended to showcase the capabilities of the concept and its features. After careful consideration, we have opted not to pursue a direct integration with the *postal bank* for the creation of pre-loaded cards or with *credit companies* due to the intricate administrative demands involved. Our commitment remains steadfast in providing a solution that is of value to users and centers around their needs, within the constraints of our project's scope. As a result, all data within the app will remain local.

1.5. Methodology

In carrying out our project, "Wallety: Digital Wallet for Families," we adopted a research and development methodology that involved a series of steps and processes to collect data, implement our solution, and analyze results.

The first phase of our methodology involved extensive research and market analysis. We conducted a thorough examination of existing financial management apps, specifically focusing on features relevant to family financial management. This research allowed us to identify gaps and opportunities within the market, which guided our development process.

Based on the insights gained from our research, we proceeded to the development phase. We followed an iterative approach, starting with designing wireframes and user interfaces. This allowed us to visualize the app's layout and user flow. Once the design was finalized, we moved on to implementing the core features of Wallety, including expense tracking, money transfers, saving accounts, and the chores section.

Throughout the development process, we prioritized rigorous testing and quality assurance procedures. We conducted functional testing to verify the app's functionality and compatibility testing to ensure it worked seamlessly across different devices and platforms. Any bugs or issues identified were addressed promptly to improve the app's stability, security, and overall performance.

Regarding data collection, we gathered user data through various means, including user feedback, app usage analytics, and anonymized user behavior. This data helped us gain insights into user patterns, preferences, and interactions with the app. It also allowed us to make data-driven decisions for further improvements.

Throughout the entire project, we maintained a collaborative and interdisciplinary approach. We worked together closely. Regular team meetings and effective communication channels facilitated seamless coordination, progress tracking, and knowledge sharing.

This comprehensive approach ensured that Wallety met the specific needs of families in terms of financial management and provided a seamless and effective user experience.

1.6. Organization of the Project Book

This is a brief outline of the structure and organization of our final project book:

1. Introduction

- Background: Provides context and background information on the development of Wallety.
- Problem Statement: Clearly defines the problem or challenge the app aims to address.
- Objectives: Outlines the specific goals and objectives of the project.
- Scope and Limitations: Describes the boundaries and limitations of the app.
- Methodology: Explains the approach and methods used in developing Wallety.

2. Literature Review

- Overview of Relevant Literature: Provides a summary of existing literature related to family financial management apps, digital wallets, and similar technologies.

3. System Design and Implementation

- System Architecture: Describes the overall structure and architecture of Wallety.
- Implementation Details: Provides insights into the technical aspects and development process of the app.
- Evaluation Metrics: Outlines the metrics and criteria used to evaluate the app's performance.

4. Results and Analysis

- Experimental Setup: Explains the setup used for testing and evaluating Wallety.
- Presentation of Results: Presents the outcomes and results of the app's performance and features.
- Data Analysis and Interpretation: Analyzes the collected data and interprets the findings.
- Comparison with Existing Approaches: Compares Wallety with other similar apps in the market.
- Discussion of Findings: Provides a discussion of the significance and implications of the results.

5. Additional material

- Includes additional material, such as code snippets, user manuals, our screens and any other relevant information that supports Wallety

6. Conclusion and Future Work

- Conclusion: Summarizes the achievements and contributions of Wallety.

- Future Work: Proposes potential future research directions and areas for further improvement.

7. References

- Lists all the sources and references used throughout the project book.

This outline provides a clear structure and logical flow for our project book, making it easier for readers to navigate through the content and understand the progress and insights gained in developing Wallety.

2. Literature Review

Overview of Relevant Literature:

In today's digital age, the **management of family finances** has become increasingly **crucial**. With the rise of smartphone technology, numerous financial apps have emerged, aiming to revolutionize the way families handle their expenses, transfers, savings, and overall financial management. This section explores the relevant literature surrounding the development and implementation of innovative Android applications like our Digital Wallet for Families.

The Concept of Digital Wallets:

Digital wallets have gained significant traction in recent years, with the growing popularity of mobile payment solutions. These applications offer a convenient and secure way to store payment information, make transactions, and manage financial activities on-the-go. Several studies have highlighted the benefits of digital wallets, emphasizing their potential to streamline financial processes and enhance financial inclusion for diverse user groups (Alam, 2018)[1]. Our Digital Wallet for Families builds upon this concept by tailoring the digital wallet experience to the unique needs of family financial management.

Financial Education for Children:

The importance of early financial education for children has been widely recognized in the academic sphere. **Studies have shown that instilling money management skills at a young age positively impacts financial behavior in adulthood** (Danes et

al., 2017)[2]. Various research works underscore the significance of introducing financial concepts through hands-on experiences, such as incentivizing children with rewards for completing household tasks (Aubert et al., 2020)[3]. Our app addresses this critical aspect of financial education by providing parents with a platform to reward children for their chores, thereby fostering financial responsibility and understanding.

Existing Financial Management Apps:

Literature review reveals the existence of various financial management applications that target different aspects of personal finance. Some apps focus solely on expense tracking, allowing users to manually log their income and expenditures. Others facilitate money transfers between users but lack comprehensive features for family-oriented financial management. Additionally, there are platforms that engage children in managing their allowances and saving goals. However, none of the existing applications appear to offer a holistic solution that combines all these critical features under one user-friendly interface. Our Digital Wallet for Families **bridges this gap** by consolidating various functionalities into a **single**, **comprehensive application**, enabling seamless financial management for parents and children alike.

Challenges in Implementation:

Implementing a multifunctional financial app like ours comes with its own set of challenges. Connecting to credit companies and banking institutions to enable real-money transfers requires robust security measures and adherence to stringent regulations. Additionally, integrating with the postal bank system demands careful consideration of user data privacy and authentication protocols. Nevertheless, recent

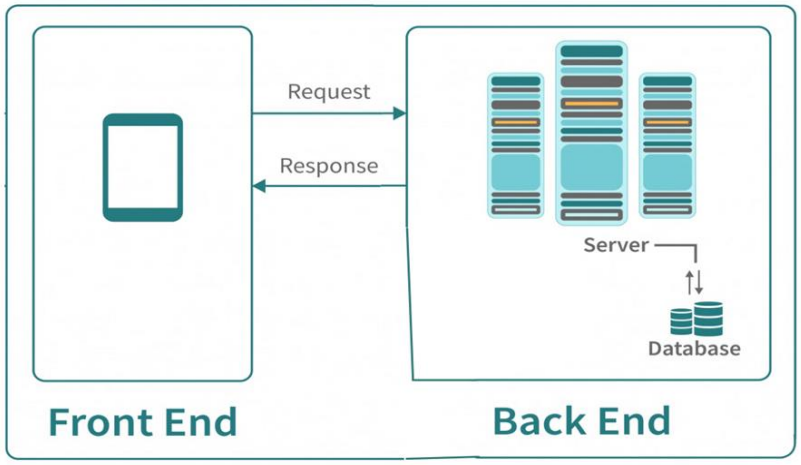
advancements in financial technology and secure APIs offer promising solutions to these challenges, paving the way for innovative applications like ours to thrive.

Conclusion:

The literature reviewed here highlights the significance of digital wallets, the importance of financial education for children, and the existence of various financial management apps with distinct features. Our Digital Wallet for Families emerges as a groundbreaking project that combines these elements into a comprehensive application, addressing the complexities of family financial management. By offering features such as task-based rewards, dedicated savings accounts, secure money transfers, and expense tracking, our app aims to empower families with essential money management skills, foster responsible financial behavior, and pave the way for a brighter financial future for the entire household.

3. System Design and Implementation

3.1 System Architecture



Using Android Studio, we constructed and fashioned both the user interface and the visual aesthetics within the app. The intricate mechanisms and functionalities, including functions like money transfers and task addition, have been skillfully implemented using **Java**. These requests, once initiated, are received by the **server**, a dynamic code developed with **Node.js.**

Upon successful user verification, the server connects to Firebase database. This database serves as a reliable storage space, to store and retrieval user data and information, such as transfer records, his credit card details, saving and tasks tables and more.

Our architectural blueprint embraces a multitude of **design patterns**, each serving a distinct purpose:

- **Singleton:** There is only one class instance which is responsible for making all the requests to the server.
- **Adapter:** When the server retrieves the data from the database, it converts the data to the format that the client can receive.

3.2. Implementation Details

The user interface and appearance of the app are built by **Android Studio framework**, **while the controllers are implemented in Java programming language. The requests to the server are implemented by Retrofit library and received by the server that is implemented in Node.js. The server connects to Firebase database that is used to store data.**

The app can run on Android devices which have at least an android version of 11.0.

During the implementation phase, we decided to create a server that will execute requests instead of performing data actions directly from the client to the database, to verify the user and secure our database.

The server, constructed using **Firebase Firestore DB**, plays a pivotal role in managing user-related data and facilitating essential functionalities within the application. These are some file explanations from our server-

Configuration: The configuration for the server is encapsulated in the **config.js** file. The Firebase initialization and administration are set up here. The server's interactions with Firestore and Firebase services are orchestrated using this configuration.

User Management: The server handles user authentication and registration through the **user.js** file. Functions such as **loginUser** and **signUpUser** manage the user authentication process, utilizing Firebase Authentication services. Upon successful login or signup, user data is accessed and updated in the Firestore database. Users are stored in the "users" collection within the Firestore database, where their details like name, email, phone number, registration tokens, and balances are stored. Passwords are stored securely and not exposed directly.

Transactions and Balances: The **makeTransaction** function allows users to perform transactions, whether it's transferring funds between users or handling savings. The server handles the deduction of funds from the sender's balance and addition to the receiver's balance or savings account. The transaction data is stored in the "transactions" sub-collection within the user's document in the Firestore database.

Linking Cards: The **linkCard** function allows users to link their credit card information to their account. Credit card details such as the holder's name, card number, expiration year and month, and CVV are stored securely in the Firestore database under the user's document.

Child-Parent Relationships: The server accommodates parent-child relationships. Parents can link their children to their accounts and initiate transfers to their children's accounts. Additionally, the **linkCardToChild** function facilitates adding funds to a child's account while linking a credit card for that transaction.

Notifications and Messaging: The server enables the sending of notifications to parents when a child makes an unusual expense. These notifications are facilitated through Firebase Cloud Messaging (FCM) services. Parents are subscribed to specific topics, allowing targeted notifications to be sent.

Formatting and Timestamps: The **format-user.js** file handles the formatting of user data, especially timestamps, making them user-friendly and easily interpretable. This file also manages the formatting of user children's data, ensuring consistent and organized data representation.

This are the algorithms in our app:

1. Detecting Unusual Purchases Algorithm – Z-Score algorithm:

The goal of this algorithm is to identify and notify parents when a child makes an unusual purchase compared to their previous spending habits.

Algorithm Overview:

- Whenever a child makes a new purchase, the algorithm assesses whether the expense is unusual in relation to the child's past spending patterns.
- The algorithm calculates a Z Score for the new purchase. The Z Score is a statistical measure that indicates how far away a data point is from the mean of a dataset, in terms of standard deviations. It provides insights into how unusual or typical a data point is.
- If the Z Score exceeds a certain threshold, the purchase is considered unusual.
- Upon detecting an unusual purchase, a notification is sent to the parent, informing them about their child's atypical expense.

Part of our code:

```
// Z-score algorithm for Irregular Expense (on transactions List)
public boolean isTransactionUnusual(List<Transaction> transactionsList, int transactionAmount) {
    if (transactionsList == null)
        return false;

    // Calculate the mean and standard deviation of the expenses
    double mean = 0.0;
    double stdDev = 0.0;
    for (Transaction transaction : transactionsList) {
        mean += transaction.getAmount();
    }
    mean /= transactionsList.size();

    for (Transaction transaction : transactionsList) {
        stdDev += Math.pow(transaction.getAmount() - mean, 2);
    }
    stdDev = Math.sqrt(stdDev / (transactionsList.size() - 1));

    // Parent set the Z-score threshold
    double zScoreThreshold = 1.4;

    // Calculate the Z-score for each expense and mark irregular expenses
    double zScore = (transactionAmount - mean) / stdDev;
    boolean isUnusual = Math.abs(zScore) > zScoreThreshold;

    return isUnusual;
}
```

2. Get Children Without Parents Algorithm:

Objective: This algorithm identifies and presents parents with the option to choose children who are not yet linked to any parents during registration.

Algorithm Overview:

- During registration, a new parent needs to select their children. The algorithm assists in populating the list of available children for the parent to choose from.
  A "**blacklist**" set is used to store user IDs of parents. As parents are encountered, their children are added to this set.
  Simultaneously, an array is maintained to collect children who do not yet have parents.
  After traversing all users, the algorithm returns the array of children who are not present in the "blacklist" set.
  Pseudocode:
  Traverse all the users.
  If it's a parent, add his children to the Set. else, it's a child, so add it to the array.
  Return children from the array that are not contained in the Set.

These algorithms enhance the functionality and user experience of Wallety application. The "Detecting Unusual Purchases" algorithm helps parents stay informed about their child's spending behavior, while the "Get Children Without Parents" algorithm simplifies the process for new parents to select children during registration. These algorithms contribute to the overall effectiveness and user-friendliness of the app, making it more intuitive and valuable to users.

3.3. Evaluation Metrics

We use criteria of request time to assess the performance or efficiency of our app. The evaluation of app success is made by checking how much time it takes to make a request, like navigating to a different screen, making a transfer, showing profile details, etc.

According to our evaluation results, there are actions which are fast and take quite milliseconds (screen navigation, show profile details), actions that take one second (make transfer, create new task) and actions that are slower (show available children in parent selection take 2.5 seconds).

So our app runs quickly in the total majority of the actions and there are few times it takes quite more than the average.

4. Project Results and Analysis

4.1. Experimental Setup

In this section, we provide a comprehensive overview of the experimental setup and methodology used to evaluate Wallety's performance and functionality. The experimental setup was carefully designed to ensure the accuracy, reliability, and reproducibility of the results obtained.

Methodology and Objectives:

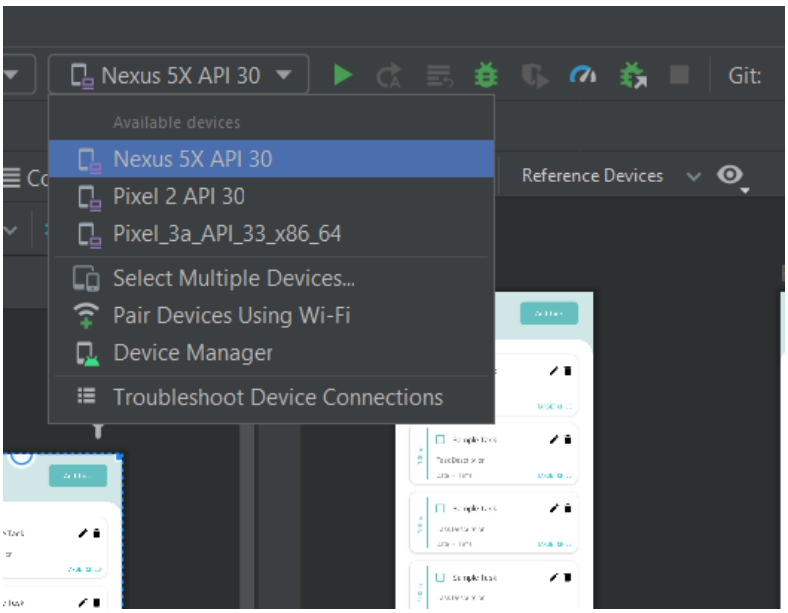
The primary objective of the experiments was to assess Wallety's performance across its core features, including expense tracking, money transfers, chore assignment, and savings goals. To achieve this, a structured methodology was devised, involving both controlled scenarios and real-world usage simulations.

Configurations and Parameters:

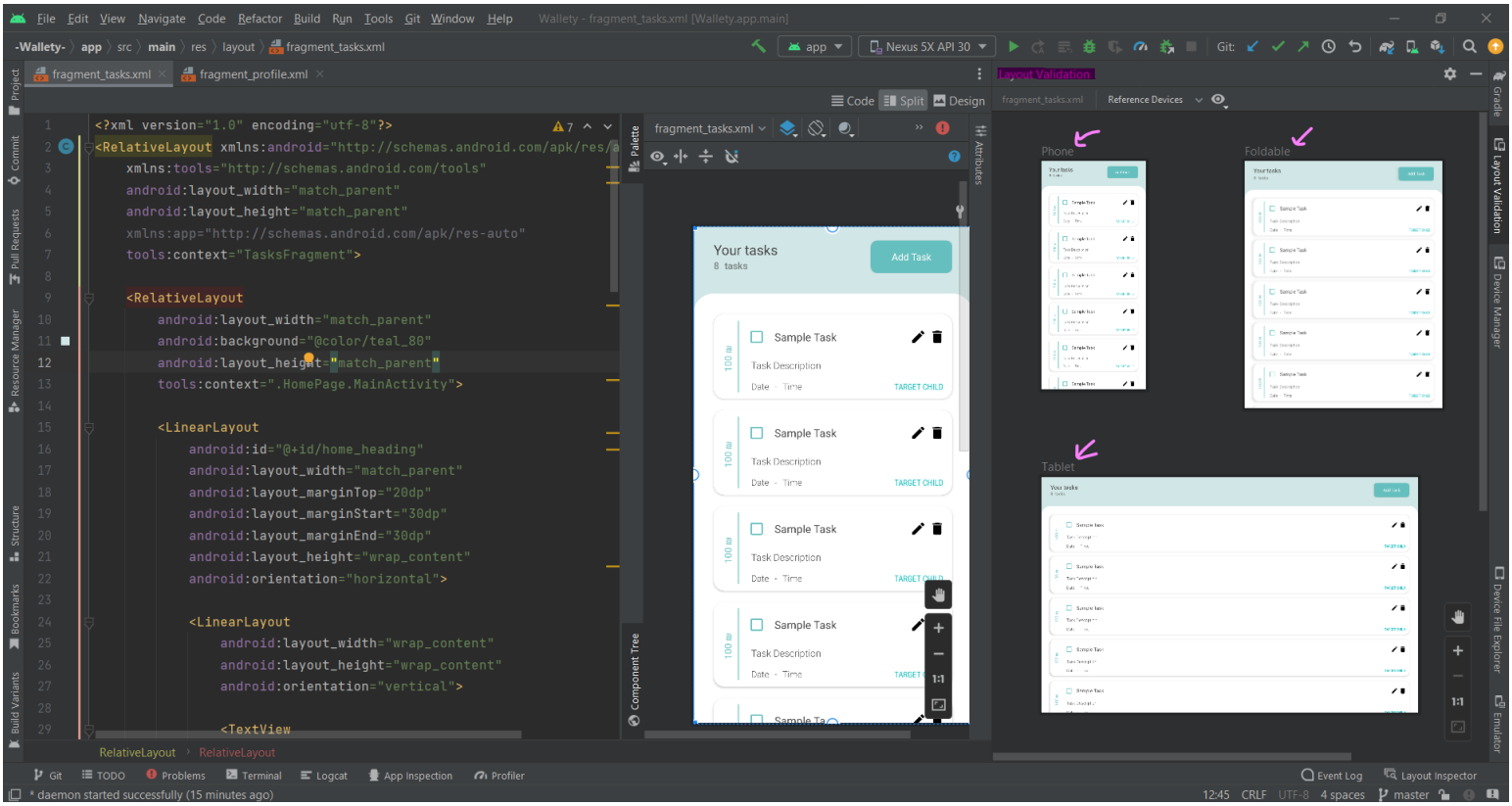
- Device Selection: A range of Android devices, **representing various screen sizes** and performance levels, were selected for the experiments. This allowed us to gauge Wallety's compatibility and performance across different devices.
- User Scenarios: Various user scenarios were defined, such as adding expenses, transferring money between family members, assigning chores, and setting savings goals. Each scenario was designed to simulate real-world usage and cover a wide spectrum of app functionalities.
- Sample Data: A diverse dataset of expenses, chore assignments, and transfers was created for testing purposes. This data encompassed different transaction types, categories, and amounts to emulate real usage scenarios.

Hardware and Software Requirements:

- **Android Devices:** A variety of Android devices, including smartphones and tablets, were used to assess Wallety's performance and user experience on different screen sizes.
- **Android Versions:** The experiments were conducted on multiple Android versions to ensure compatibility with a wide range of operating systems.



Layout validation:



Ensuring Reproducibility:

To ensure the reproducibility of results, the following steps were taken:

- Code Versioning: The source code of Wallety was stored in a version control system-Git, enabling us to track and replicate specific code versions used during the experiments.

This is our Git project repository: https://github.com/ShiraBalali/-Wallety-

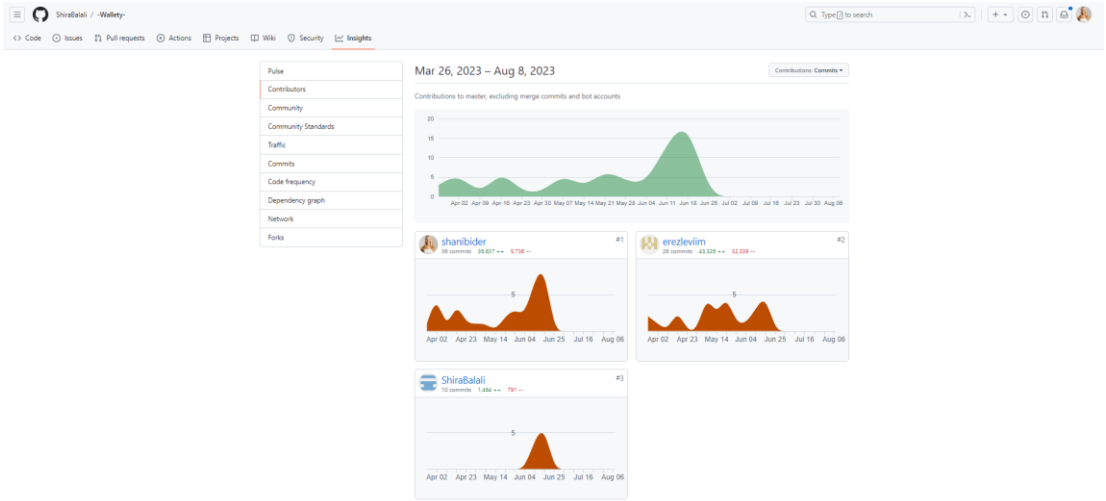
- Backup and Restoration: Regular backups of the app's databases were created before each experiment. In case of unexpected issues, these backups allowed us to restore the app to its original state for further testing.

4.2. Presentation of Results

In this section, we present a comprehensive overview of the outcomes derived from experiments and tests conducted on Wallety's features. The results are not only collected from Git repository data but also include data collected from Wallety's Firebase database. These diverse datasets are carefully organized, utilizing various visualization methods to effectively communicate our findings.

- Git Repository:

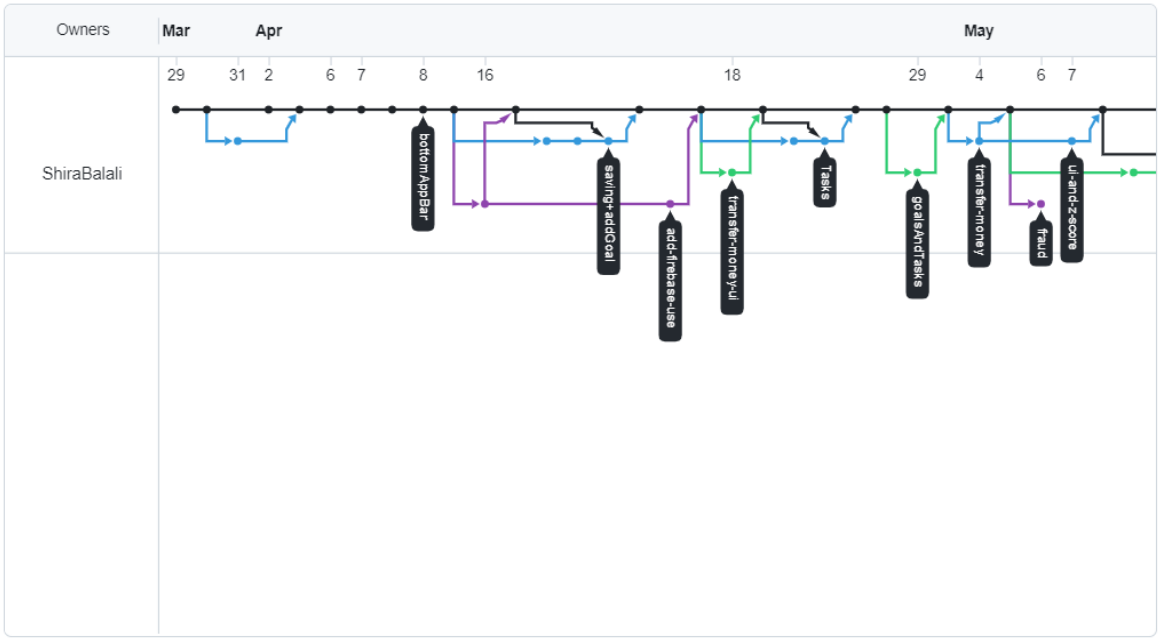
Insights obtained from **Wallety's Git repository** included code versions, changes made during development, and historical records of feature implementation. This allowed us to trace the evolution of Wallety's functionalities over time.



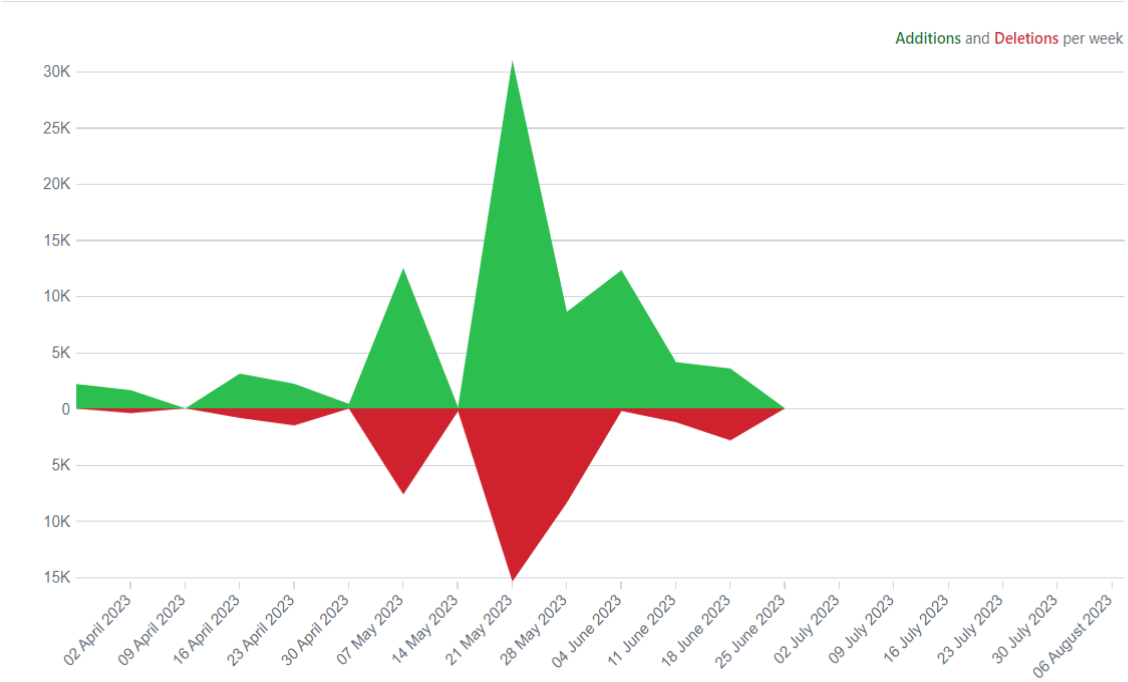
Our collaboration:

Network graph

Timeline of the most recent commits to this repository and its network ordered by most recently pushed to.



Work through the months:



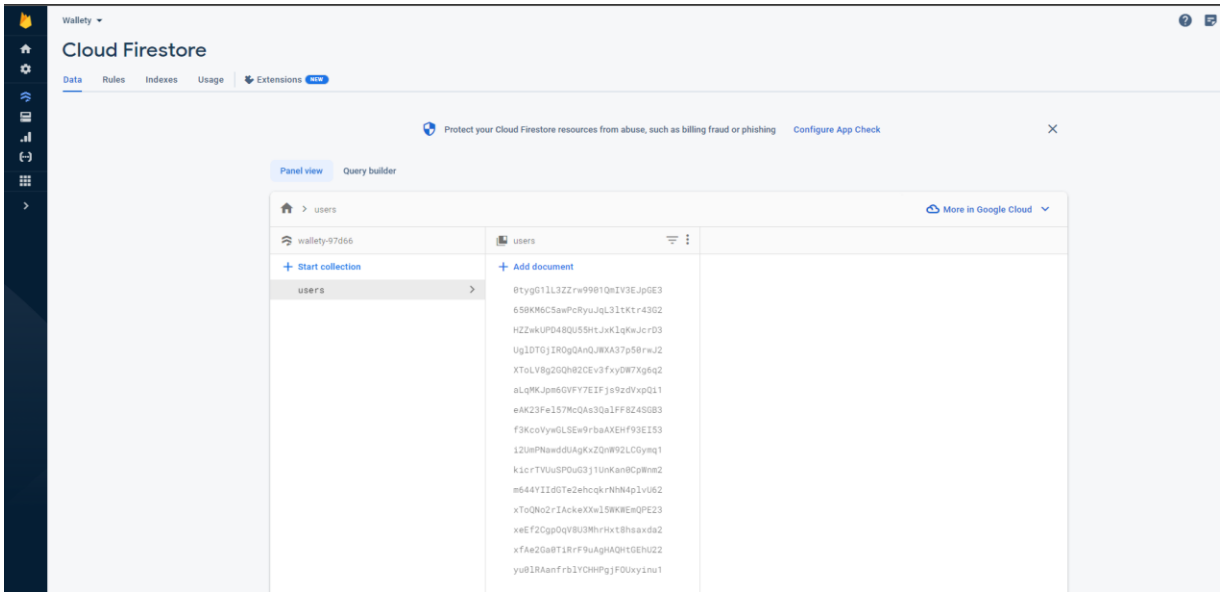
- Firebase Database:

Wallety's **Firebase database** provided invaluable user interaction data, encompassing expense entries, transaction records, chore assignments, savings goals, and user data.
All our data is stored in the Firestore Database, in Firebase, and goes through our own server we developed in javascript.

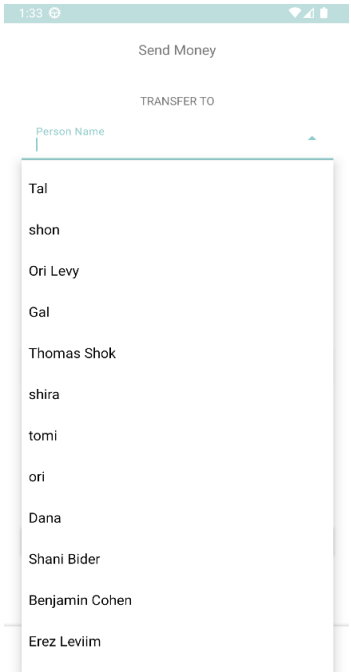
Here are some of our **tables of the data**, and how they appear **on the application screens** to the user:

All the users:

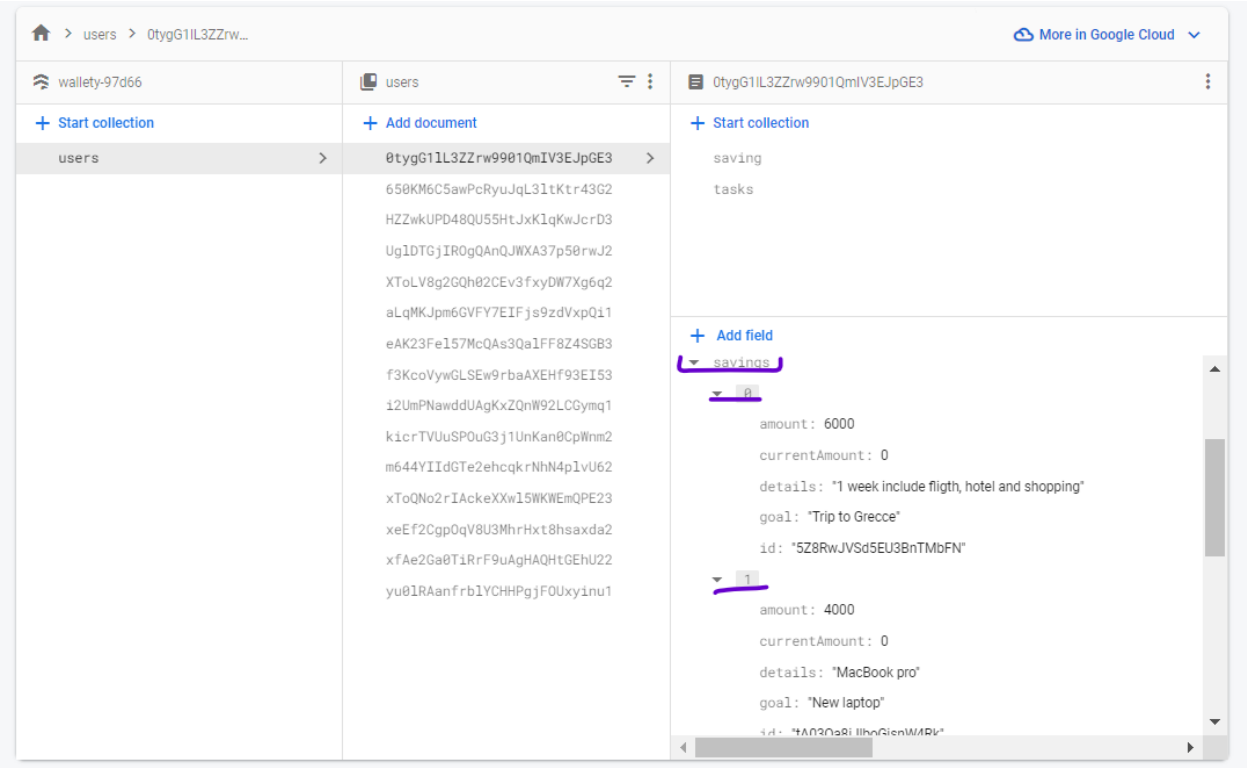
On Firebase:

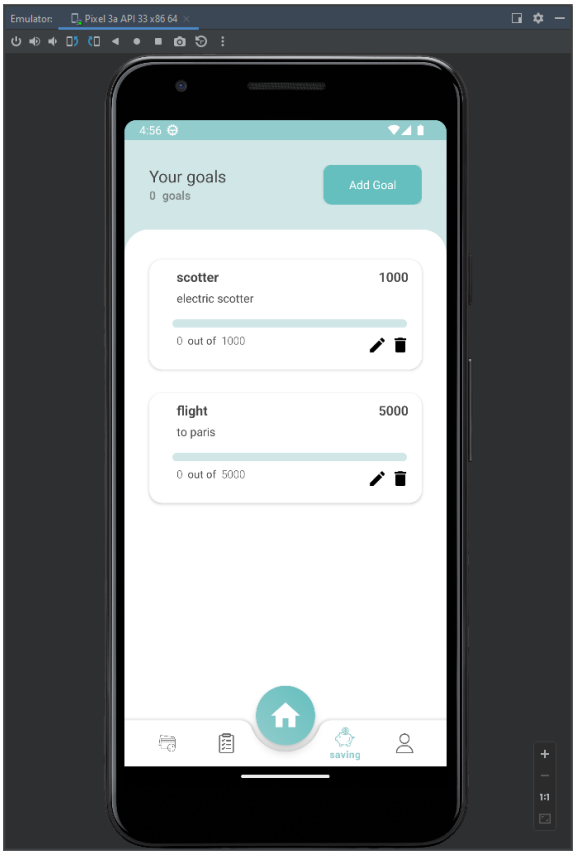


On The app:

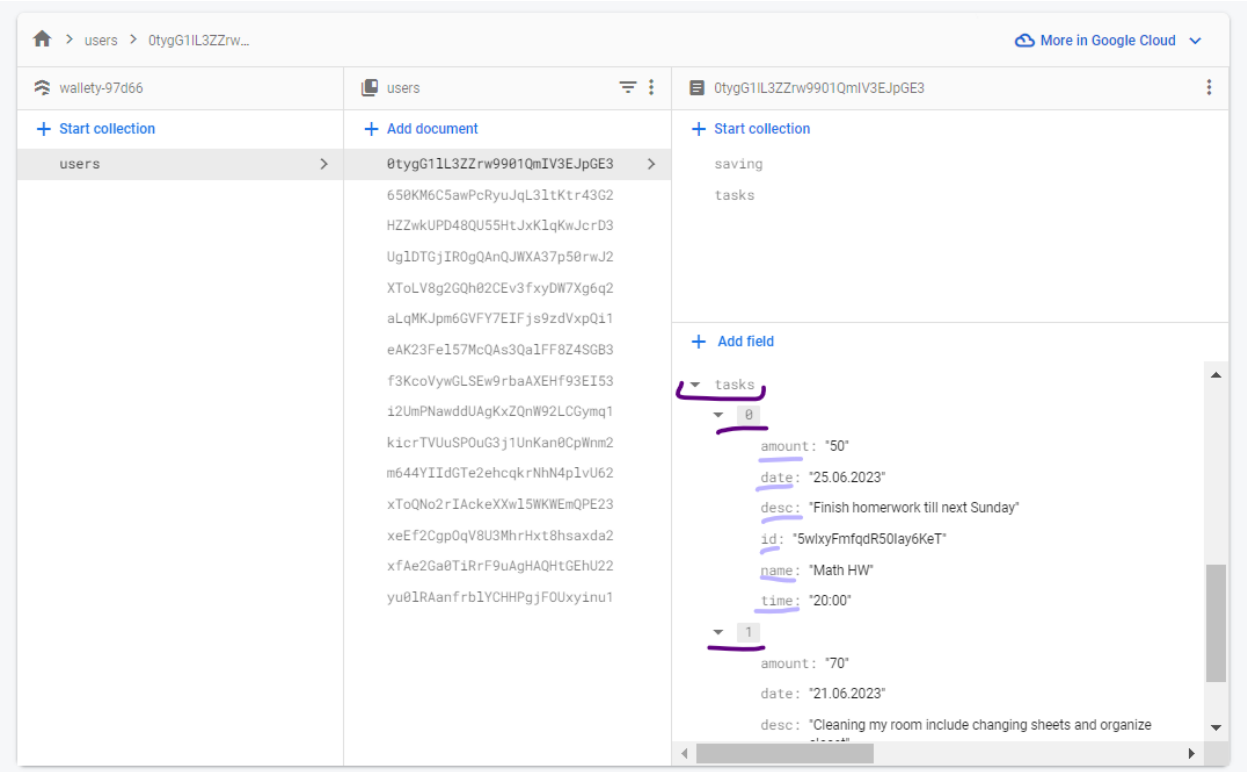


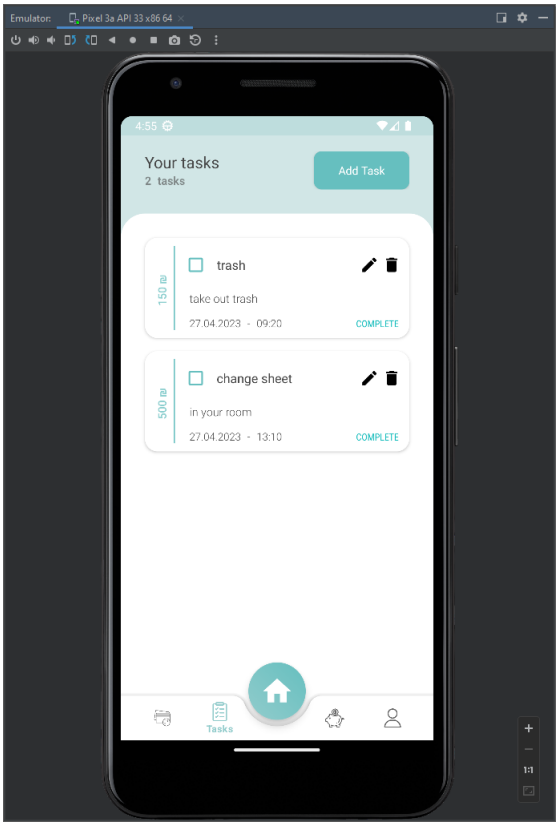
Saving table:



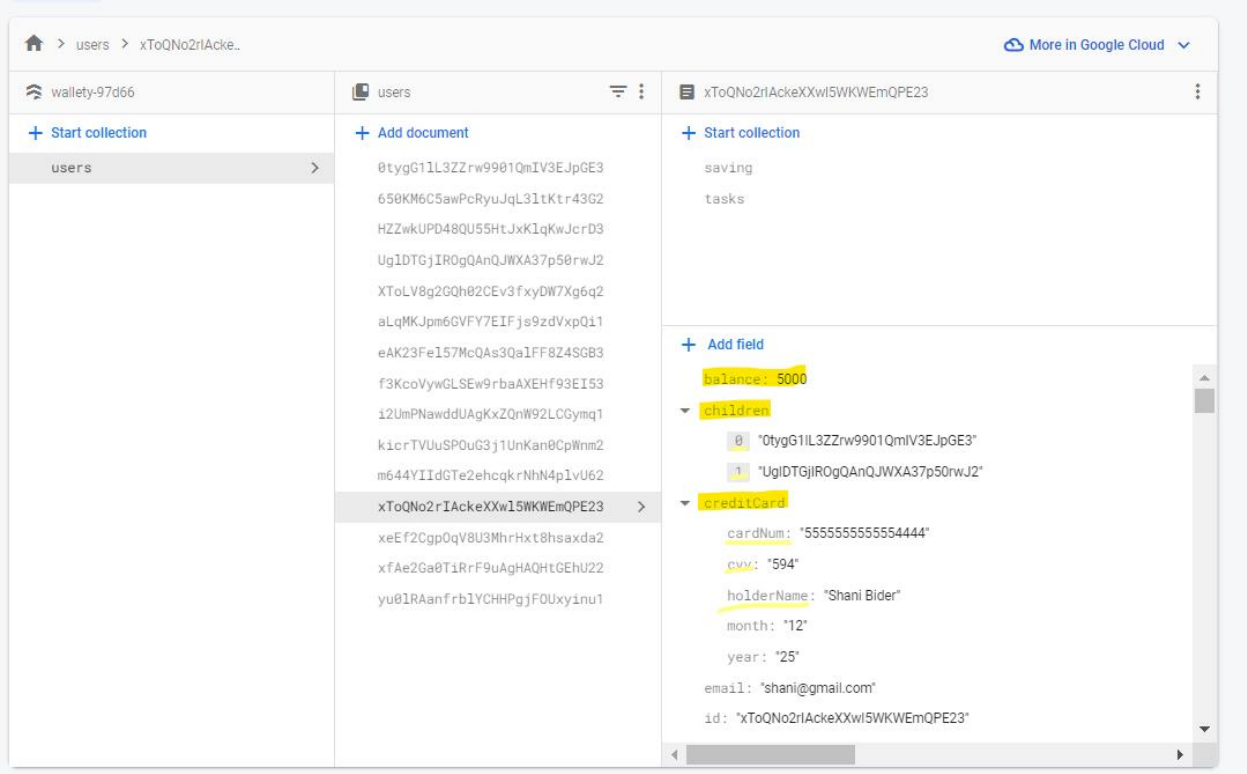


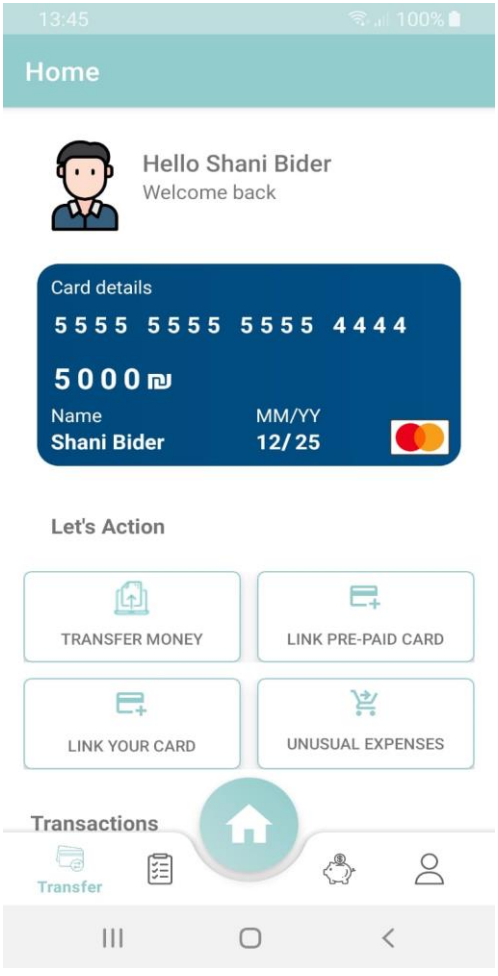
Tasks table:


users > 0tygG1IL3ZZrw...
More in Google Cloud
wallety-97d66
users
0tygG1IL3ZZrw9901QmIV3EJpGE3
Start collection
Add document
Start collection
users
0tygG1lL3ZZrw9901QmIV3EJpGE3
650KM6C5awPcRyuJqL3ltKtr43G2
HZZwkUPD48QU55HtJxKlqKwJcrD3
UglDTGjIROgQAnQJWXA37p50rwJ2
XToLV8g2GQh02CEv3fxyDW7Xg6q2
aLqMKJpm6GVFY7EIFjs9zdVxpQi1
eAK23Fel57McQAs3QalFF8Z4SGB3
f3KcoVywGLSEw9rbaAXEHf93EI53
i2UmPNawddUAgKxZQnW92LCGymq1
kicrTVUuSPOuG3j1UnKan0CpWnm2
m644YIIdGTe2ehcqkrNhN4plvU62
xToQNo2rIAckeXXwl5WKWEmQPE23
xeEf2CgpOqV8U3MhrHxt8hsaxda2
xfAe2Ga0TiRrF9uAgHAQHtGEhU22
yu0lRAanfrblYCHHPgjFOUxyinu1
saving
tasks
Add field
tasks
0
amount: "50"
date: "25.06.2023"
desc: "Finish homerwork till next Sunday"
id: "5wlxyFmfqdR50lay6KeT"
name: "Math HW"
time: "20:00"
1
amount: "70"
date: "21.06.2023"
desc: "Cleaning my room include changing sheets and organize

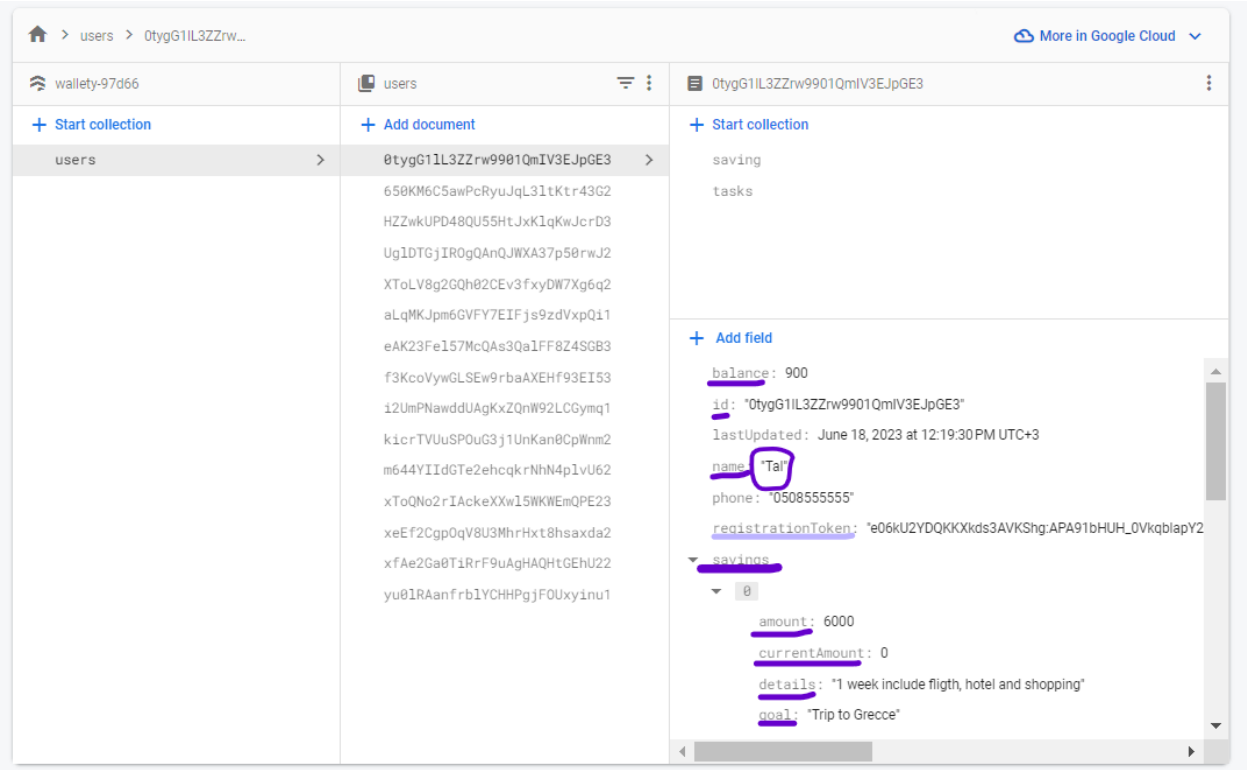

Emulator: Pixel 3a API 33 x86 64
4:55
Your tasks
2 tasks
Add Task
trash
take out trash
27.04.2023 - 09:20
COMPLETE
change sheet
in your room
27.04.2023 - 13:10
COMPLETE
Tasks

Specific user with his data (Initial balance, Children, CreditCard details, id, registerToken):

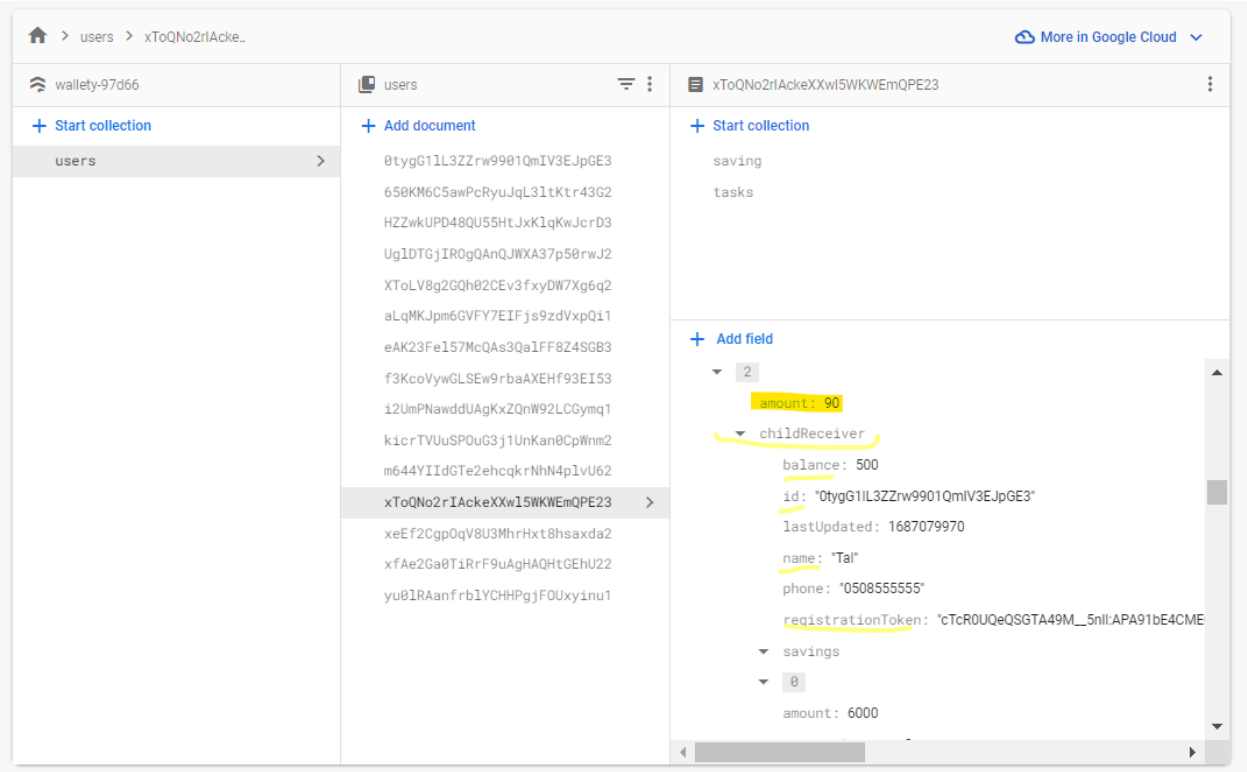


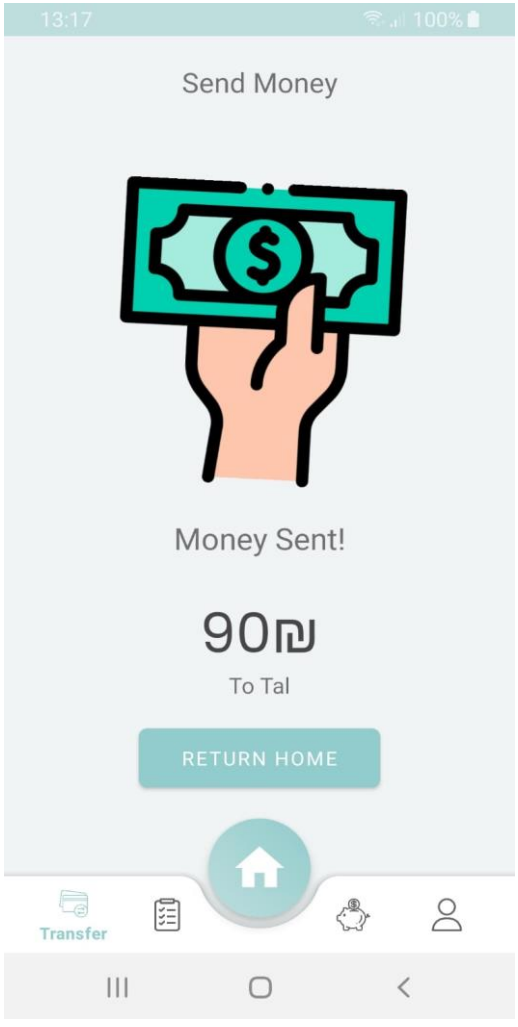


Child user (child initial balance is 900₪):



Transactions:





5. Additional material:

In this section, we provide additional material and relevant information that supports our project, Wallety - the Digital Wallet for Families. Included here are code snippets, existing features, system documentation, and a user manual to aid users in understanding and utilizing the application effectively.

1. Code Snippets:

Below are key **code snippets** from our Android application, Wallety, illustrating the implementation of some of the **core features**:

Code snippets from our APP built in Java using Android Studio:

- **WalletyAPI:**

```
// API endpoints for making different HTTP requests
// @POST/@GET to specify the HTTP method, @Body to indicate the request body
public interface WalletyAPI {

    @POST("loginUser")
    Call<User> loginUser(@Body UserLoginRequest userLoginRequest);

    @POST("signUpUser")
    Call<UserSignUpResponse> signUpUser(@Body User user);

    @GET("childrenWithoutParent")
    Call<List<User>> getChildrenWithoutParent();

    @POST("makeTransaction")
    Call<ResponseBody> makeTransaction(@Body TransactionRequest transactionRequest);

    @POST("linkCard")
    Call<ResponseBody> linkCard(@Body LinkCardRequest request);

    @POST("linkCardToChild")
    Call<ResponseBody> linkCardToChild(@Body LinkCardToChildRequest request);

}
```

- **connection to the server using Retrofit:**

```java
public class UserFetcherCon {
    // initializes the Retrofit instance with the base URL of the server and a Gson converter factory
    private static final Dotenv dotenv = Dotenv.configure().directory("./assets").filename("env").load();
    private static final String BASE_URL = String.format("%s/users/", dotenv.get("SERVER_URL"));
    private static final Retrofit retrofit = new Retrofit.Builder()
            .baseUrl(BASE_URL)
            .addConverterFactory(GsonConverterFactory.create())
            .build();
    private static final WalletyAPI api = retrofit.create(WalletyAPI.class);


    // The methods in this class enqueue the API calls using Retrofit and provide the callback for handling the response
    public static void loginUser(UserLoginRequest userLoginRequest, Callback<User> callback) {
        Call<User> call = api.loginUser(userLoginRequest);
        call.enqueue(callback);
    }

    public static void signUpUser(User user, Callback<UserSignUpResponse> callback) {
        Call<UserSignUpResponse> call = api.signUpUser(user);
        call.enqueue(callback);
    }

    public static void getChildrenWithoutParent(Callback<List<User>> callback) {
        Call<List<User>> call = api.getChildrenWithoutParent();
        call.enqueue(callback);
    }

    public static void makeTransaction(TransactionRequest transactionRequest, Callback<ResponseBody> callback) {
        Call<ResponseBody> call = api.makeTransaction(transactionRequest);
        call.enqueue(callback);
```

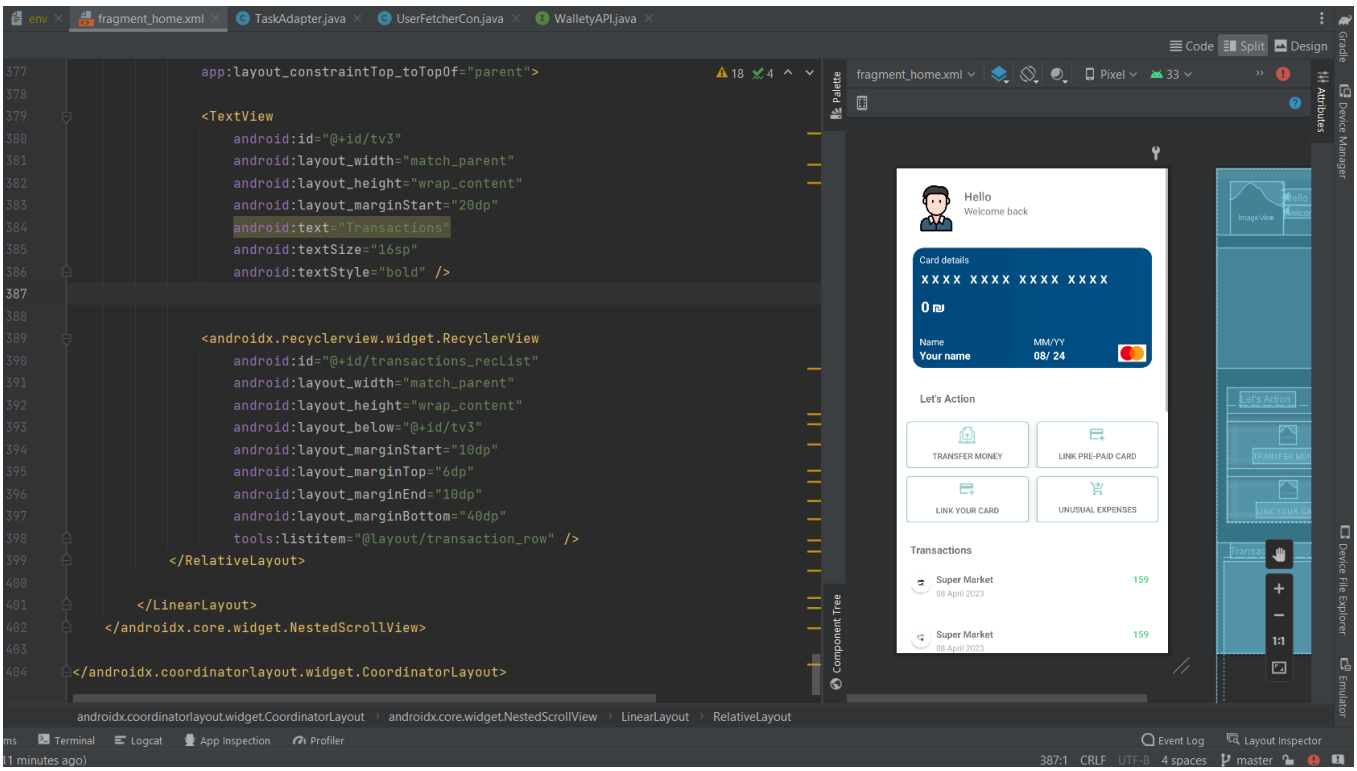
- **Date picker in CREATE TASK Fragment:**

```java
    // Date Picker
    taskDate.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            //The materialDatePicker is shown using the show() method
            materialDatePicker.show(getActivity().getSupportFragmentManager(),  tag: "MATERIAL_DATE_PICKER");
        }
    });
    // When user selects a date, this method called on the materialDatePicker object
    materialDatePicker.addOnPositiveButtonClickListener(new MaterialPickerOnPositiveButtonClickListener() {
        @Override
        public void onPositiveButtonClick(Object selection) {
            Calendar calendar = Calendar.getInstance(TimeZone.getTimeZone("UTC"));
            calendar.setTimeInMillis((long) selection);
            //the selected date is formatted into a string in the format "dd.MM.yyyy" using a SimpleDateFormat
            SimpleDateFormat format = new SimpleDateFormat( pattern: "dd.MM.yyyy");
            String formattedDate = format.format(calendar.getTime());

            // The formatted date stored in the udate array and displayed in the taskDate using setText()
            udate[0] = formattedDate;
            taskDate.setText(udate[0]);
        }
    });
```

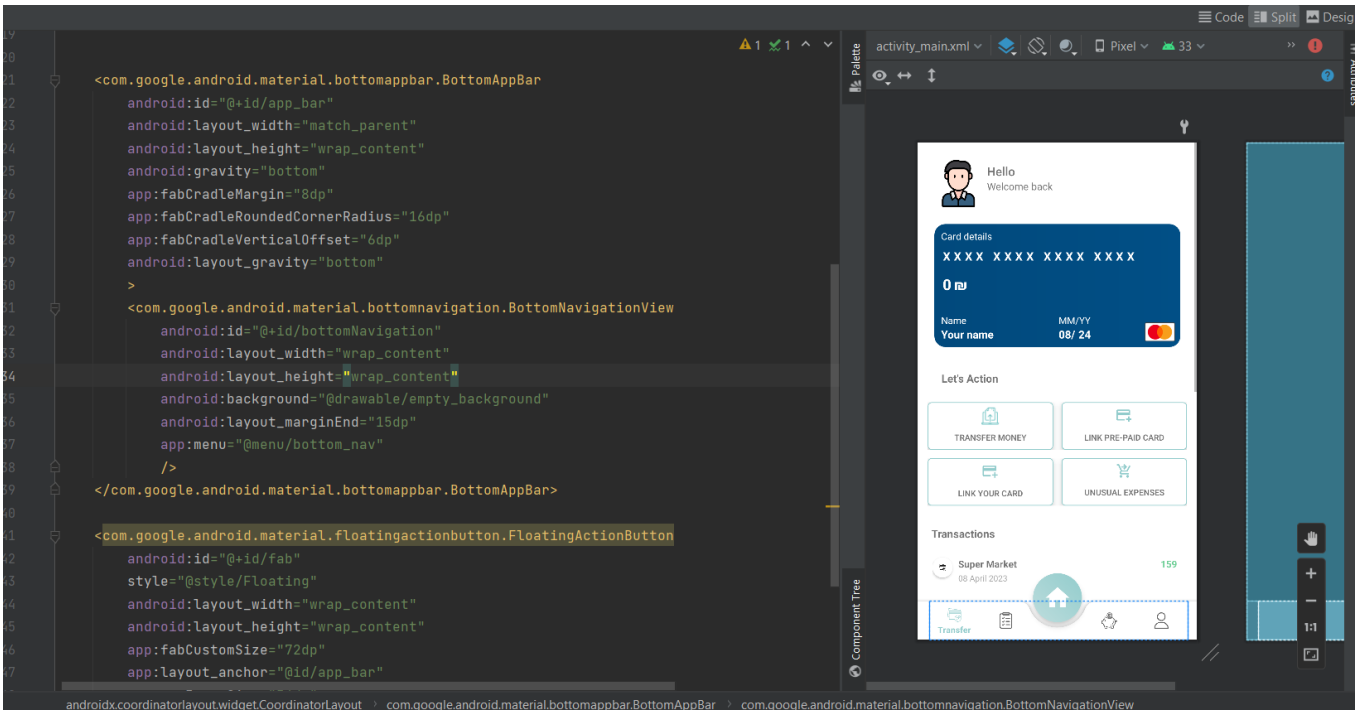
- **Date click handler in “TASK ADAPTER class:**

```
// date click handler
final String[] utime = new String[1];
final String[] udate = new String[1];

final long today = MaterialDatePicker.todayInUtcMilliseconds();
MaterialDatePicker.Builder materialDateBuilder = MaterialDatePicker.Builder.datePicker();
materialDateBuilder.setTitleText("SELECT A DATE");
materialDateBuilder.setSelection(today);
final MaterialDatePicker materialDatePicker = materialDateBuilder.build();
materialDatePicker.addOnPositiveButtonClickListener(new MaterialPickerOnPositiveButtonClickListener() {
    @Override
    public void onPositiveButtonClick(Object selection) {
        Calendar calendar = Calendar.getInstance(TimeZone.getTimeZone("UTC"));
        calendar.setTimeInMillis((long) selection);
        SimpleDateFormat format = new SimpleDateFormat( pattern: "dd.MM.yyyy");
        String formattedDate = format.format(calendar.getTime());

        udate[0] = formattedDate;
        date.setText(udate[0]);
    }
});

date.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        materialDatePicker.show(((FragmentActivity) context).getSupportFragmentManager(),  tag: "MATERIAL_DATE_PICKER");
    }
});
```

- **“fragment_home.xml” layout:**

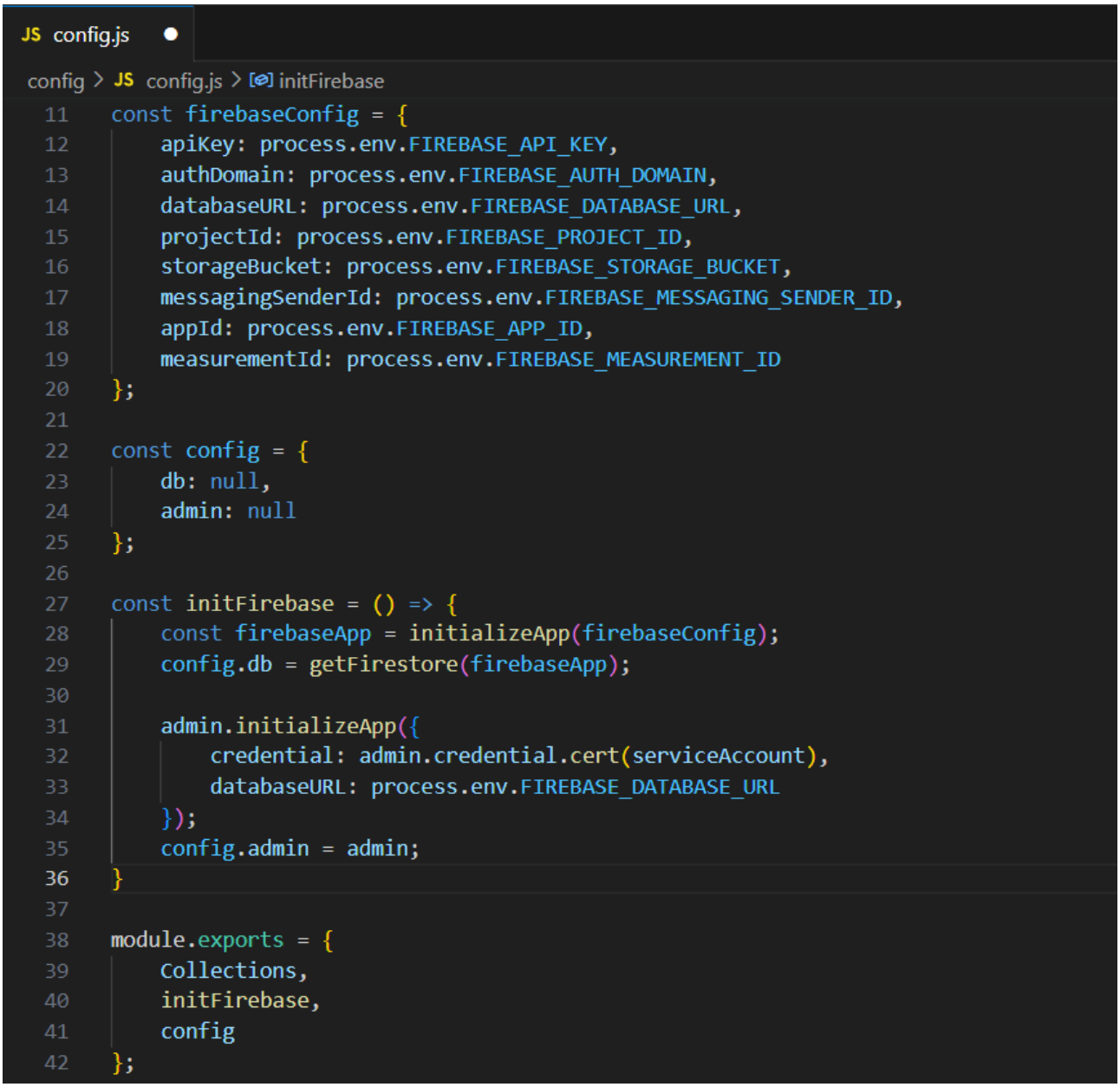


- **Floating Action Button in Home fragment:**



Code snippets from our server built in JavaScript using Visual Studio Code:

- initialize connections and configurations for Firebase DB:



JS config.js ●

config > JS config.js > initFirebase

```
11  const firebaseConfig = {
12      apiKey: process.env.FIREBASE_API_KEY,
13      authDomain: process.env.FIREBASE_AUTH_DOMAIN,
14      databaseURL: process.env.FIREBASE_DATABASE_URL,
15      projectId: process.env.FIREBASE_PROJECT_ID,
16      storageBucket: process.env.FIREBASE_STORAGE_BUCKET,
17      messagingSenderId: process.env.FIREBASE_MESSAGING_SENDER_ID,
18      appId: process.env.FIREBASE_APP_ID,
19      measurementId: process.env.FIREBASE_MEASUREMENT_ID
20  };
21
22  const config = {
23      db: null,
24      admin: null
25  };
26
27  const initFirebase = () => {
28      const firebaseApp = initializeApp(firebaseConfig);
29      config.db = getFirestore(firebaseApp);
30
31      admin.initializeApp({
32          credential: admin.credential.cert(serviceAccount),
33          databaseURL: process.env.FIREBASE_DATABASE_URL
34      });
35      config.admin = admin;
36  }
37
38  module.exports = {
39      Collections,
40      initFirebase,
41      config
42  };
```

- User login:

```
const loginUser = async (req, res) => {
    const {db, admin} = config;
    const auth = getAuth();
    const {email, password, registrationToken} = req.body;
    signInWithEmailAndPassword(auth, email, password)
        .then(async (userCredential) => {
            const docRef = doc(db, Collections.USERS, userCredential.user.uid);
            await updateDoc(docRef, {
                registrationToken
            });
            const docSnap = await getDoc(docRef);

            if (docSnap.exists()) {
                const user = docSnap.data();
                admin.auth().createCustomToken(user.id).then(async (accessToken) => {
                    await formatUserChildren(user);
                    formatUser(user, email, password, accessToken);
                    console.log(email + " logged in");
                    res.status(StatusCodes.OK).send(user);
                }).catch((error) => {
                    console.log(error)
                    res.status(StatusCodes.INTERNAL_SERVER_ERROR).send(error);
                });
            }
        })
        .catch((error) => {
            console.log(error)
            res.status(StatusCodes.INTERNAL_SERVER_ERROR).send(error);
        });
};
```

- Create user:

```
    createUserWithEmailAndPassword(auth, email, password)
        .then(async (userCredential) => {
            const id = userCredential.user.uid;
            const CHILD_INITIAL_BALANCE = 500;
            const PARENT_INITIAL_BALANCE = 5000;
            const balance = children ? PARENT_INITIAL_BALANCE : CHILD_INITIAL_BALANCE;
            const childrenIds = children ? children.map(({id}) => id) : [];
            const user = {
                id,
                name,
                phone,
                balance,
                registrationToken,
                ...(children && {children: childrenIds}),
                lastUpdated: serverTimestamp()
            };
            await setDoc(doc(db, Collections.USERS, id), user);

            // subscribe parent to children unusual expenses
            const {admin} = config;

            if (children) {
                const registrationTokens = [registrationToken];
                childrenIds.forEach(id => {
                    admin.messaging().subscribeToTopic(registrationTokens, id)
                        .then((response) => {
                            // Response is a message ID string.
                            console.log('Successfully sent message:', response);
                        })
                        .catch((error) => {
                            console.log('Error sending message:', error);
                        });
                });
            }
```

2. Our features:

These existing features in Wallety work synergistically to empower families with essential money management tools. The app promotes financial **transparency, responsibility, and independence** among children while **providing parents with a comprehensive solution for overseeing their family's financial activities**.

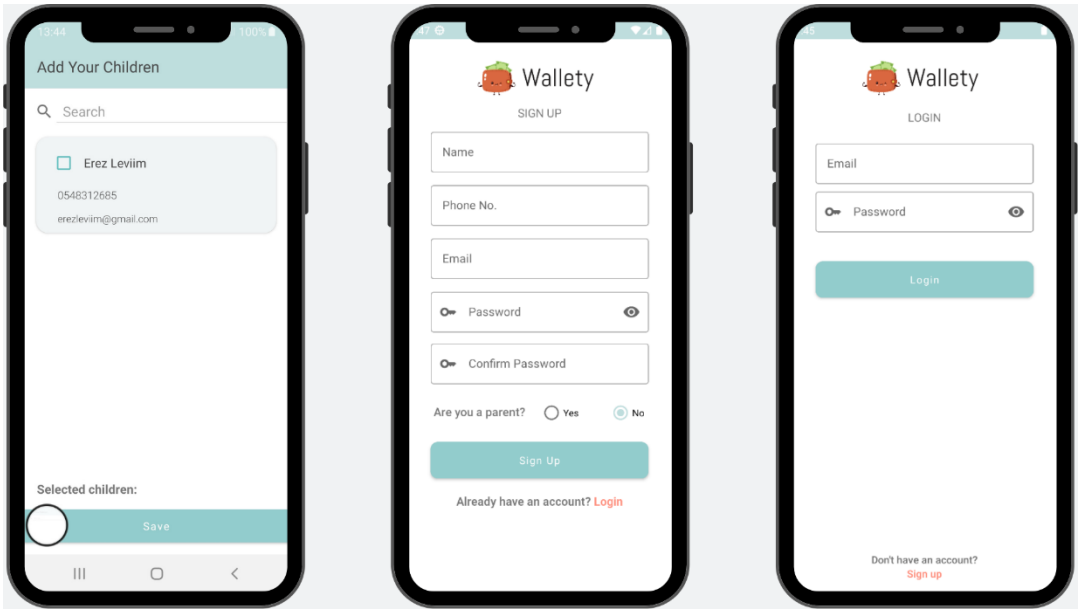
In addition, to help you familiarize yourself with Wallety's user interface, we have included a series of UI screen captures below. These images showcase the design and layout of the app, providing you with a visual guide to navigate its various features and functionalities, alongside our features.

1. Parent User Login / Child User Login:

Wallety provides **separate login interfaces** for parents and children. Parents can log in with their credentials, gaining access to the full suite of administrative features, while children have their own login to view their assigned tasks, expenses, and savings.

2. Parent-Child Linking:

Parents can link their account to all their children within the app. This linkage enables seamless communication and financial management between parents and their children.

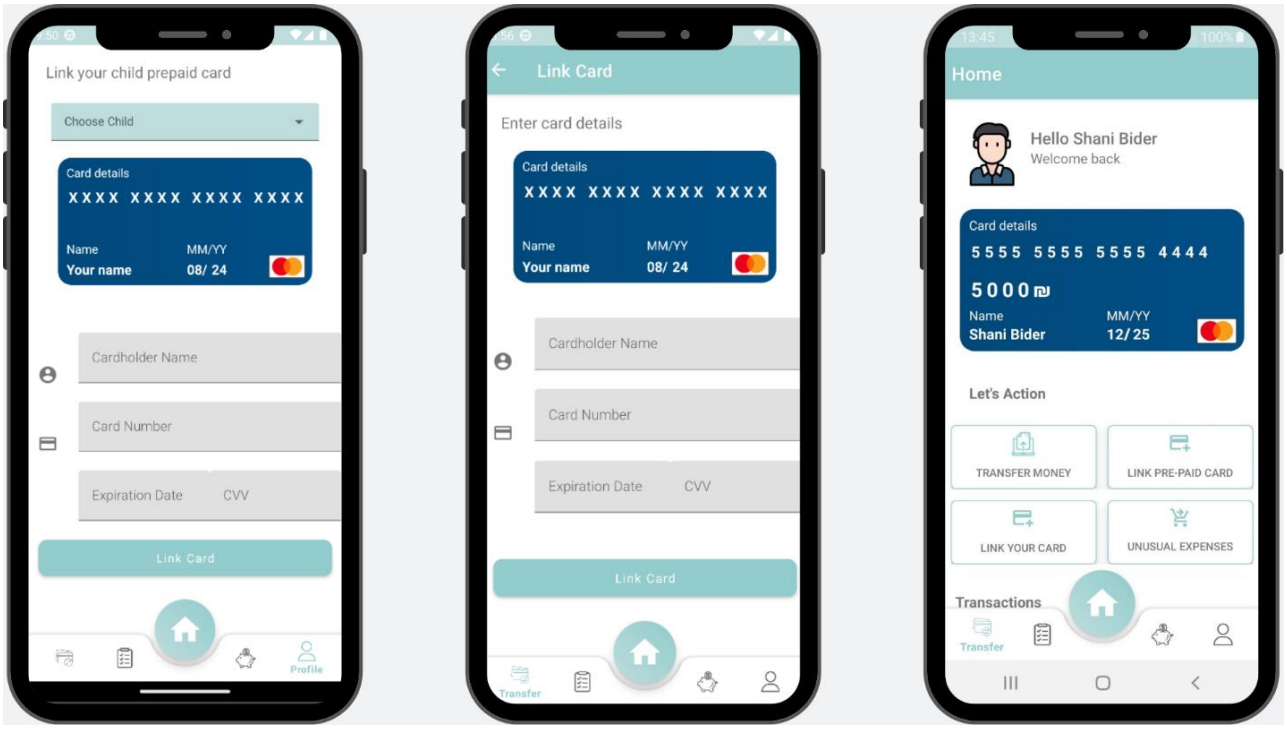


3. Credit Card Linking for Parents:

Parents have the option to link their **credit card** to Wallety, allowing them to initiate money transfers securely. The credit card serves as a source for financial transactions within the app.

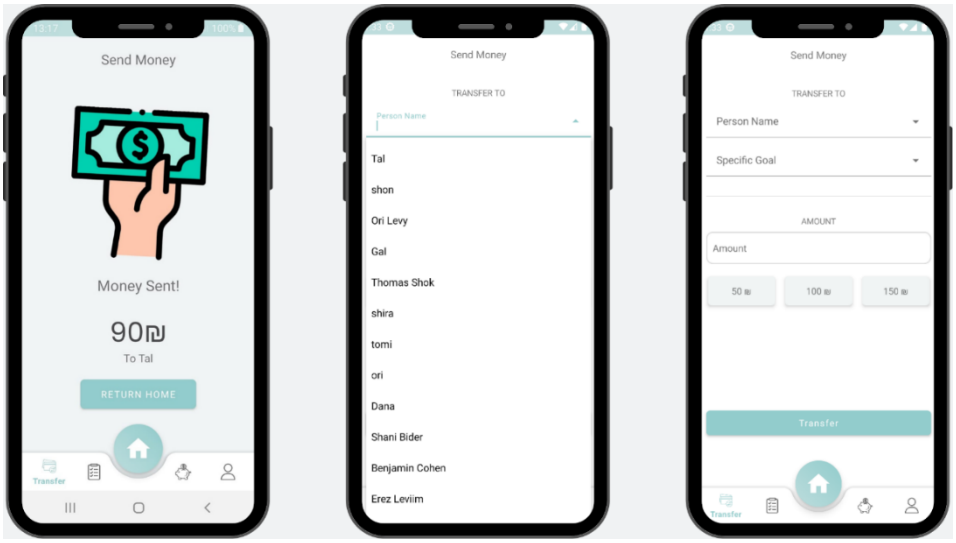
4. Loaded Cards for Children:

Wallety offers a unique feature where parents can create **individual loaded cards** for each of their children. These cards are pre-loaded with an initial amount, providing children with their own digital spending account.



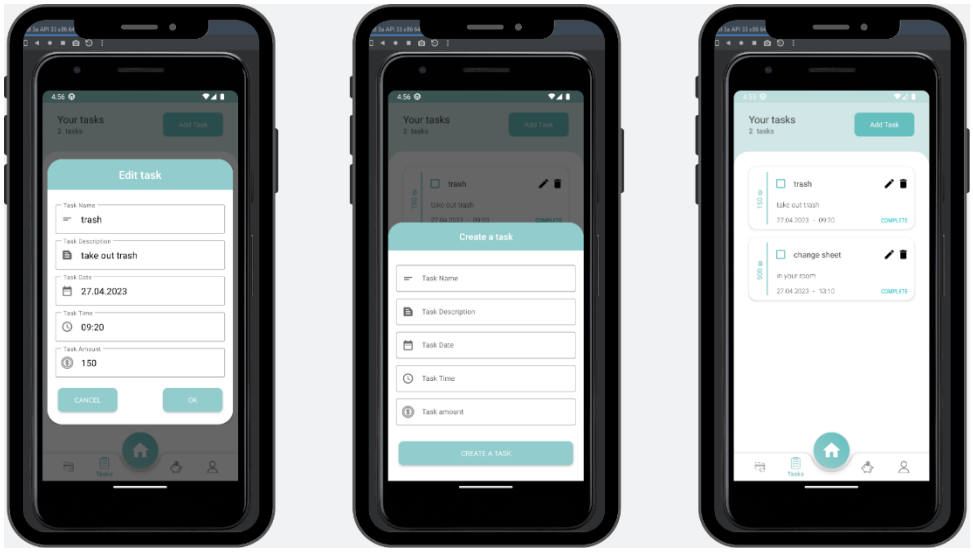
5. Money Transfer Functionality:

Parents can transfer money from their linked credit card to their children's individual loaded cards or to specific financial goals set by each child. This feature promotes financial independence and transparency within the family.



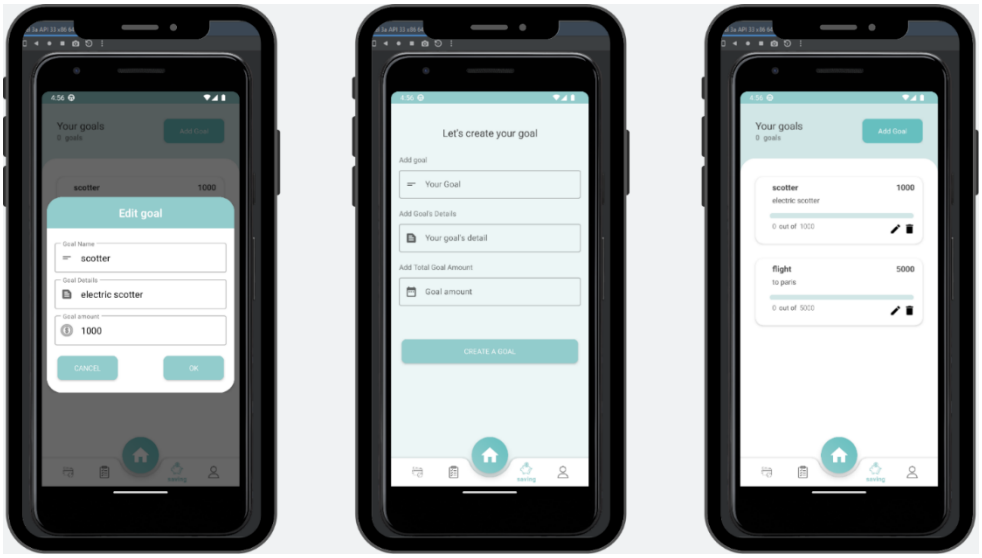
6. Tasks Screen:

The **Tasks** screen displays pre-defined tasks along with their due date, time, and corresponding rewards. Parents can set up and manage these tasks, giving children the opportunity to earn rewards for completing them. The feature also allows parents to edit or delete assignments as needed.



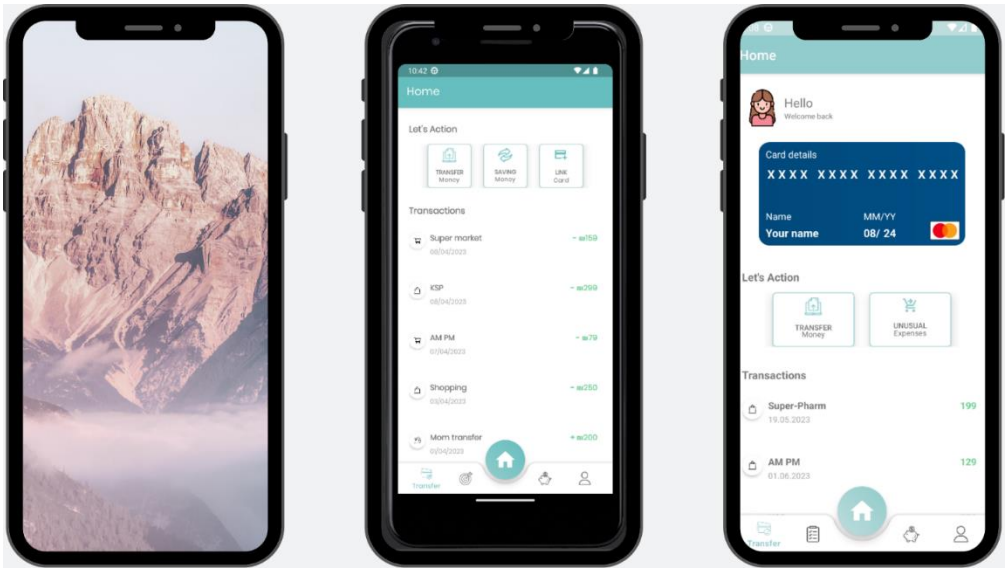
7. Savings Screen:

In Wallety's Savings screen, children can set **financial goals** by specifying a target amount they wish to save. Both parents and children can contribute funds towards any goal from their list of defined goals, encouraging responsible financial planning.



8. Transaction Screen:

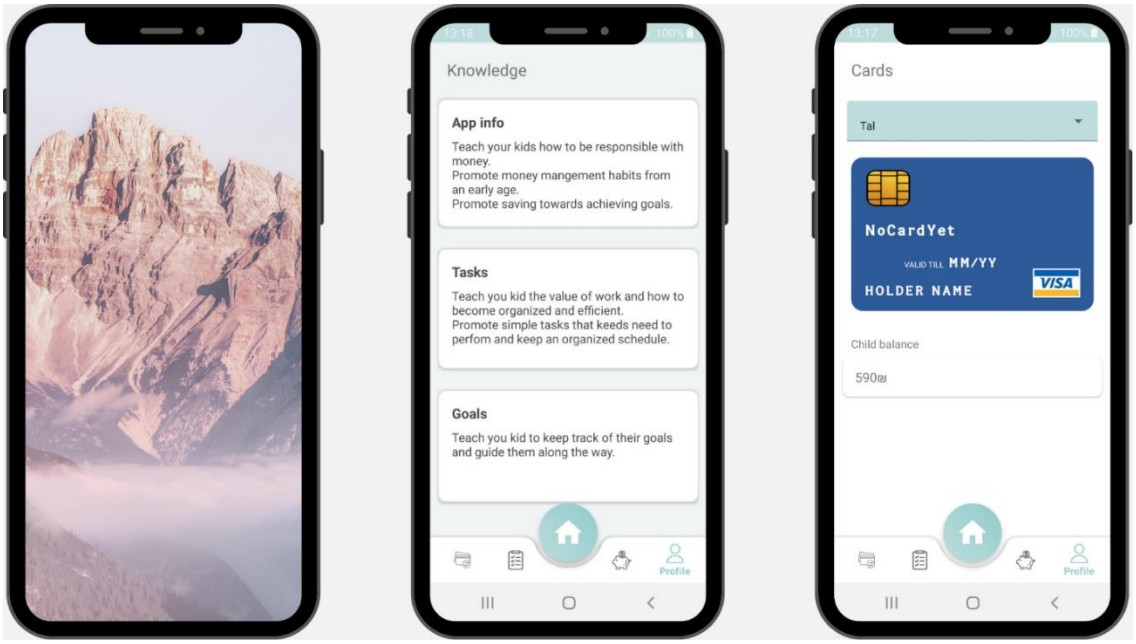
The Transaction screen provides parents with a detailed overview of all expenses incurred in the last month. This feature allows for transparent tracking of family expenditures, aiding in effective budget management.



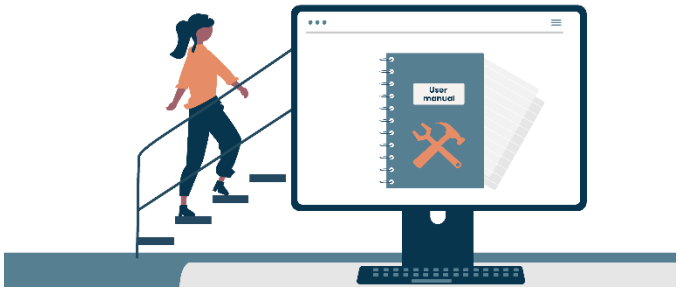
9. My Family Screen:

The My Family screen serves as a centralized hub for parents, displaying all their children's preloaded cards and their current sum of money. This consolidated view allows parents to easily monitor and manage their children's individual spending accounts efficiently. Parents can keep track of each child's spending habits and available balance, fostering financial responsibility and transparency within the family.

As Wallety continues to evolve, our team is dedicated to adding even more features and enhancements to further improve the user experience and strengthen family financial management.



2. Wallety- User Manual:



The user manual for Wallety offers step-by-step guidance on how to use the application. It covers the installation process, registration, login procedures, and detailed explanations of each feature. Additionally, it provides insights into best practices for managing expenses, transferring money, assigning tasks, and monitoring.

Getting Started

1. Download and Install: Search for "Wallety - Digital Wallet for Families" on the Google Play Store and install the app on your Android device.

2. Registration: Launch Wallety and follow the on-screen prompts to create your account. Parents should register as "Parent" and children as "Child."

If you are a parent: Linking Children section: After logging in as a parent, link your account to all your children using their respective login details.

Parent Account

1. Credit Card Linking: To facilitate money transfers, securely link your credit card to Wallety from the main page.

3. Expenses Monitoring: Track your family's expenses by visiting the "Transactions" screen, where you can view detailed expense information for the last month.

Child Account

1. Assigned Tasks: As a child, access the "Houswork Assignments" screen to view pre-defined tasks assigned by your parents, along with their due dates and rewards.

2. Loaded Cards: Each child has their own loaded card with an initial amount. Check your current balance on the main page.

Expense Tracking

1. Income and Expenses Overview: Access the "Expense Tracking" screen to view a comprehensive overview of all income and expenses for the last month.

2. Income and Expense Insights: Gain valuable insights into your family's financial activities, helping you make informed decisions for budget optimization and financial planning.

Money Transfers

1. Sending Money: Parents can transfer money securely to their children's loaded cards or specific savings goals defined by each child.

2. Receiving Money: Children can check their loaded card balances after receiving money transfers from their parents.

Savings Goals

1. Setting Goals: Define your financial goals by adding them in the "Savings" section. Set a target amount and track your progress.

2. Contributing to Goals: Both parents and children can contribute funds to each other's goals to achieve them faster.

Best Practices

1. Budget Management: Monitor your expenses regularly to make informed financial decisions and optimize your budget.

2. Task Completion: Children should complete assigned tasks promptly to earn rewards and cultivate financial responsibility.

The code snippets, system documentation, and user manual in this appendix serve as essential resources to complement the main project documentation. They contribute to a deeper understanding of Wallety's development, functionality, and potential for future improvements.

Note: Due to the size and complexity of Wallety, the complete codebase, system documentation, and user manual are available upon request.

Design and Color Scheme-

Wallety's design philosophy revolves around simplicity, intuitiveness, and a delightful user experience. We understand the importance of user-friendly interfaces, especially when managing finances for the entire family. The color scheme we have chosen for Wallety is "**turquoise**," which is a vibrant and refreshing shade of turquoise. We also ensure that Wallety is both visually appealing and easy to navigate.

The turquoise color scheme embodies a sense of harmony and tranquility, encouraging a stress-free approach to family financial management. The use of turquoise elements throughout the app promotes feelings of security, reliability, and financial transparency, setting a positive tone for financial discussions within the family.

nav_graph.xml | colors.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <!-- in use-->
    <color name="teal_70">#F0F4F4</color>
    <color name="teal_80">#D1E6E6</color>
    <color name="teal_90">#BEDDDD</color>
    <color name="teal_100">#92CCCC</color>
    <color name="teal_150">#71CDCD</color>
    <color name="teal_200">#12BCBC</color>
    <color name="teal_500">#66BFBF</color>

    <!--Basic-->
    <color name="lightGrey">#D8D8D8</color>
    <color name="grey">#474747</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
    <color name="transparent">#00000000</color>

    <!-- Credit Card colors-->
    <color name="blue1">#01497c</color>
    <color name="blue2">#014f86</color>
```

6. Conclusion and Future Work

Throughout the course of our project, we have made **significant steps** in developing the Digital Wallet for Families, an innovative Android application designed to enhance financial management within households. Our app offers a comprehensive solution for monitoring expenses effortlessly, enabling users to gain valuable insights into their spending patterns and make informed financial decisions. Additionally, the ability to transfer money securely between family members promotes financial transparency and simplifies household financial transactions. Moreover, the incorporation of rewarding tasks and dedicated savings accounts fosters financial education for children and instills essential money management skills from an early age.

While our project has achieved notable success, we also encountered limitations and challenges that warrant acknowledgment. One prominent limitation was the complexity of working with credit companies and connecting to the postal bank to facilitate the issuance of loaded credit cards for children. These aspects presented technical and regulatory hurdles that required careful consideration and cooperation from financial institutions. Despite these challenges, we remain committed to addressing these limitations in future iterations of the application.

Looking ahead, our team is enthusiastic about the potential for further development and improvement of our app, with a strong focus on enhancing the user experience. We have identified several exciting ideas for future work, such as *implementing a feature that requires children to attach a picture when completing an assignment*. This additional

data verification step aims to ensure more secure approval by parents and adds an extra layer of transparency and accountability.

Furthermore, we are keen on pursuing collaborations with credit companies and the postal bank to enable real-card connections within the application. By integrating with these financial entities, our app can offer enhanced functionality and provide users with a truly seamless financial experience. This would allow the app to operate at its full potential and offer a more comprehensive suite of financial services.

Looking at the bigger picture, we firmly believe that the field of money transfers within families is shifting towards the digital area. With cash gradually fading from everyday transactions, the world is becoming increasingly digital and interconnected. We envision apps like ours becoming an indispensable part of every household, empowering families to manage their finances efficiently, cultivate responsible financial habits, and promote financial literacy among children.

In conclusion, our Digital Wallet for Families app represents a significant achievement in the area of family-oriented financial management. While we encountered challenges, our commitment to further development and improvement remains strong. With a vision of enhanced user experience, potential collaborations with financial institutions, and the ever-growing digital landscape, our app is considered to make a lasting impact on family financial management. We are excited to continue this journey, **contributing to the evolving field of financial technology** and paving the way for a brighter financial future for families worldwide. We strongly believe that in

the coming years, the paradigm of family financial management will undergo a transformative shift, with an increasing number of families opting to manage their finances through an app like ours.

This is Wallety - The ultimate Digital Wallet for Families! We hope this book helps make the most of our app's powerful features, enabling seamless financial management and fostering responsible financial habits within the family. Happy budgeting and happy saving!

7. References

[1] Alam, S. S. (2018). Adoption of mobile payment technology: an exploratory analysis of the mobile wallet. Journal of Innovation & Knowledge, 3(2), 86-94.

[2] Danes, S. M., Kabbaz, V., & Penwell, J. M. (2017). Adolescent financial capabilities: comparisons to family financial management practices and implications for intervention. Journal of Family and Economic Issues, 38(1), 79-96.

[3]Aubert, B. A., Hamidizadeh, R., & Léger, P. M. (2020). Financial education apps: Empowering users' investment literacy. International Journal of Information Management, 52, 101987.