UrbanPark

Ionut Robert Cojocaru 1949045 – Yusupha Juwara 1936515 - Reeva Gulzar 1958711 - Davide di Trocchio 1940108- Mohammad Sorooshmehr 1947890- Atay Kar 0000000

Topic and Analysis

We evaluated all the possible topics and we chose the **parking** topic.

The main apps we took in considerations are **EasyPark** and **MyCicero:**

EasyPark

1. **Pay Parking:** EasyPark allows users to pay for public parking spaces in various Italian cities.
2. **Extend Parking Time:** EasyPark typically provides the option to extend your parking time remotely if you need more time than initially paid for. This feature can be helpful to avoid parking fines.
3. **Receive Notifications:** The app often sends notifications to remind users when their parking time is about to expire, helping them manage their parking duration effectively.
4. **Check Rates and Information:** Users can view information about parking rates and other details related to on-street parkings.

MyCicero:

1. **Pay Parking:** MyCicero allows users to pay for public parking spots, pay for parking, and access information about parking facilities in various Italian cities.
2. **Urban Services Integration:** MyCicero aims to integrate various urban services into a single app, making it a comprehensive solution for urban mobility. Users can access a variety of services, including parking and transportation, in one place.
3. **Payment Options:** Similar to EasyPark, MyCicero typically offers multiple payment options for parking and other services, including credit/debit cards and mobile payment methods.
4. **Unified Platform:** The app aims to simplify the user experience by providing a unified platform for various urban services, making it convenient for travelers who need access to multiple services.
5. **Check Rates and Information:** Users can view information about parking rates and other details related to on-street parkings.

Need Finding

We started our Need Finding process by creating first the right questions for an interview:

1. Did you commute today, If yes, with which vehicle did you commute and how was your parking experience today?
2. Can you describe the common challenges you face when trying to find available parking spaces in urban areas?
3. What factors influence your decision to choose a parking area, such as proximity, security, cost, or availability?
4. Tell us about your typical parking experience, how easily do you find parking on the road and would you park in a private lot or in a garage?
5. Do you have any experience with parking lots or garages, if yes tell us how easy do you find them while searching for parkings and how difficult is the payment process?
6. How important is real-time availability information for parking spaces in your opinion?
7. Do you think payment through app can be easier for users to pay for the parking in advance?
8. How important is the ability for users to filter parking options based on factors like distance, cost, or available amenities?
9. Should the app provide information about the nearest available parking spots as a priority?
10. What information from the parking facilities would be most important for you to know before choosing a parking spot?

We conducted interviews with individuals, both online and in person, using the aforementioned questions. After analyzing the interview responses (documented in **Interviews.docx**), we discovered that on-street parking presents a significant challenge. To avoid this challenge, many people turn to private garages or parking lots. However, upon examining the previously mentioned apps, we noted a lack of features related to private parking solutions.

In order to gain a deeper understanding of the users' needs, we administered two questionnaires, one in English and one in Italian, using the same set of questions.

English Questionnaire

Questionario Italiano

In summary, our findings confirmed that on-street parking is a common challenge for many individuals, and they often resort to private parking facilities as an alternative. As a result, we have decided to develop an application that enables people to effectively manage their experiences with private parking solutions.

Storyboarding

We identified the main tasks of the application and we did one storyboard for each main task (documented in **Storyboards.pdf**). Our aim was to depict the most common and expected scenarios related to parking within the app. Our storyboard features a straightforward sequence, portraying a typical user's interaction with our application: first, they access the app and reserve a parking spot, and then they extend the reservation when it's about to expire. When presented to others, the storyboard's natural flow perfectly aligns with the concept we are developing, which prompted us to proceed with the prototyping phase.

Tasks: Login – Search – Booking – Manage Bookings

Prototyping

Drawing inspiration from the previously mentioned apps, EasyPark and MyCicero, we have developed an initial prototype with a user-centric approach. In this prototype, users can effortlessly select a parking spot and complete the payment process in just 10 taps.

First prototype

We decided to use Material Design as our main design philosophy, and thoroughly studied the colors we were going to use. Using the “Primary 500” color combination, we chose a vibrant green as our primary color, #4CAF50, and a slightly lighter tone for the secondary color, #C8E6C9.

Using these colors we were able to recreate a modern UI people would feel comfortable with, reusing common known elements such as lists and cards to convey as much information as possible while keeping everything clean and readable. The evaluation processes would confirm we were going in the right direction, aside from a couple of features people would totally overlook. Initially we put a vehicle selection slider in the home, but we found out, through lab-based tests, that people would either totally overlook the function or stop for a few seconds to understand what the screen was trying to tell. As a result, we decided it was best to remove the vehicle slider in the home screen, achieving a final product that felt linear and intuitive even to casual users.

Second prototype

Evaluation

The evaluation was carried out using a lab-based evaluation method on multiple participants who were asked to conduct a single task on our second prototype. Through the use of a review -based model we were able to identify single design patterns which didn't work and simplify them for the user.

Users: All of the users were in the 21-46 age group, had a driver's license and a fair share of experience with the "parking" theme. Some of them had prior experiences using a third party app to look for parking spots, but none of them were actively using any.

Scenario: It's Friday night, you have to go to a city event and you expect it would be difficult to find parking spots. You will go there with your bike, so you open your app and reserve a spot in one of the nearest parking spots. You arrive at the parking you booked and show the booking to a manager. **Tasks**: The evaluation is carried out by checking how these tasks are carried out:

1. Login through google

2. Vehicle choice

3. Parking spot choice

4. Parking spot booking

5. Parking spot payment

6. Booking review Tests

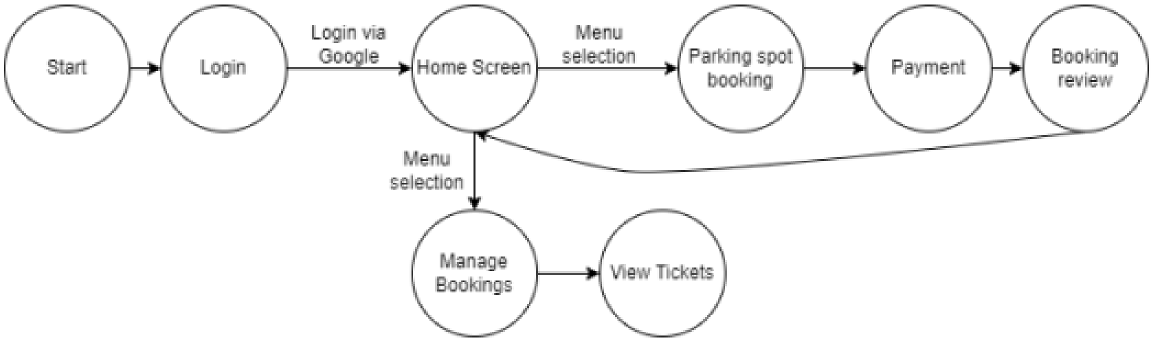
User 1: The first test user to go through the tasks was lost at task point 2. They have not seen the vehicle slider and have failed to select the correct vehicle, therefore continuing the booking with a car instead of a bike. Although the user had difficulties spotting application specific components such as the vehicle selector, they successfully went through all the other tasks in the correct order.

User 2: The second user had taken into consideration the vehicle slider but had to stop for about 10 seconds to understand all the elements they had in front of the screen. The search slider, the profile picture and the vehicle slider presented themselves as main distraction points which had to be understood before continuing with the test. This time, the user correctly selected the vehicle and continued the parking spot booking as normal without any problems.

User 3: The third user intuitively pressed on the search bar to search for the address which he needed booking for. This was different from the previous users who just pressed on the markers on the map and were distracted by other elements. The user then continued the parking spot booking as normal without any problems.

User 4: The fourth user followed the tasks without any problems but was lost when managing the booking. They weren't able to find where exactly they could find the ticket after doing the booking process and were unable to go further for about a minute or two. The significant time loss brought uncertainty in the user, who felt disoriented while doing the task to the point of opening the profile section and looking for information in there. They eventually were able to find the section and conclude the test.

After processing what every user did we decided to remove the vehicle select slider and add it to the form in the booking process, where it couldn't be ignored. Ignoring the slider and booking for a totally different vehicle would pose too much of a problem, since different vehicles amount to different rates and would put users at fault for a badly designed component. The dialog model, as observed from some of the users, takes this form: We can see that the prototype is incredibly linear and presents no branches at all for a typical user. As soon as they understand the parking selection process they are naturally guided through the application flow.



Based on the feedback and evaluation findings, the following recommendations are suggested to improve the Parking App:

- Streamlined Navigation: Simplify the app's menu structure to enhance user navigation.

- Location Accuracy: Improve the accuracy of location services to ensure accurate parking spot recommendations.

- User Education: Provide clearer instructions and onboarding tutorials to guide users through the reservation process.

- Additional Features: Consider adding features like loyalty programs, discounts, and user rewards to enhance user engagement.

App Development

The Urban Parking app primarily utilizes Java, a versatile and widely-used programming language for Android app development. Java offers a strong foundation for developing efficient and reliable Android applications, making it an ideal choice for our project.

Our application relies on key API integrations to deliver its core functionalities:

1. **Google Maps and Places API**: We incorporated Google Maps and Places API to render maps, provide navigation features, enable route finding, and implement an autocomplete search bar. This API significantly enhances the user's ability to locate parking spaces, plan routes effectively, and access real-time location-based information.

2. **Google Directions API**: The Google Directions API powers our smart route navigation feature. It enables users to find optimal routes to their chosen parking spaces, providing real-time directions and estimated arrival times.

3. **Firebase Realtime Database**: Firebase Realtime Database serves as the backbone of our data management system. It facilitates data storage, retrieval, and synchronization in real-time. User details, booking history, and parking information are efficiently managed through this platform.

5. **Google Pay Integration**: Google Pay integration streamlines the payment process for parking reservations, offering users a secure and convenient method to complete transactions.

The initial step involved crafting the login page, and we chose to exclusively offer Google-based login. Given that our app is exclusive to Android users, it's worth noting that virtually all Android users possess a Google account.

Implementing sign-in with Google in an Android app is a straightforward process. First, we set up a project in the Google Developers Console and configure the OAuth 2.0 client ID for your app. Then, in our Android app, we use the Google Sign-In API to handle the sign-in process. This involves integrating the Google Play services SDK, adding a Google Sign-In button to our UI. All the user information are sent to the a real-time database on Firebase. Once a user selects the Google Sign-In option, they'll be prompted to choose their Google account, and our app will receive a user ID token, which we can use to identify the user within our app. With this setup, we can offer a seamless and secure sign-in experience for our Android users using their Google credentials.

In the app, we added a profile activity to allow the user to:

- View About Us
- Change name and surname
- Contact Us
- Manage the garages (AVAILABLE ONLY FOR ADMINS)

The **Manage garage** fragment, we implemented a set of sub-fragments that provided administrators with versatile options, including "Add Garage," "Delete Garage," and "Modify

Garage." These functionalities were designed to efficiently handle various aspects of garage management within our application.

Here's an overview of how each sub-fragment was implemented.

Add Garage: To enable administrators to add new parking garages, we utilized Firebase's database capabilities. We employed a method that added child nodes under the "Parking" node in the Firebase Realtime Database. The unique address of each garage served as the key for these child nodes. Each child node contained essential information about the garage, such as its address, name, and price per hour.

Delete Garage: To facilitate garage removal, we used the garage's address as a key to identify and delete specific garages. Administrators could select a garage for deletion, and the application would use the garage's address to locate and delete its corresponding data from the Firebase Realtime Database.

Modify Garage: For modifying garage information, Administrators could use the key to fetch the data from the database and update the information related to the garage and push them on the database. The administrators could then make changes to attributes like name or price per hour. After modifications were made, the updated information was pushed back to the Firebase Realtime Database, ensuring that the changes were reflected in the application's database.

The **MapFragment** offers an intuitive map interface integrated with Google Maps and Places API, providing users with real-time features:

- **Real-Time Map**: Users can view their current location, available parking spaces, and interactive markers on the map.

- **Navigation and Route Finding**: The app provides seamless navigation with route finding and directions.

- **Autocomplete Search Bar**: Simplifying location-based queries, this feature enhances user convenience.

The **SearchFragment** designed for efficient browsing and booking, includes essential usercentric features:

- **Efficient Recycler View**: A responsive recycler view renders parking space listings, ensuring smooth scrolling even with large datasets.

- **Autocomplete Search**: The search bar suggests locations, reducing user typing efforts.

- **Booking Flow**: Users can select parking spaces, set start and end dates, times, and proceed to secure online payments seamlessly with Google Pay integration.

The **BookingFragment** empowers users to manage their parking reservations effectively, combining user-centric elements:

- **Detailed Recycler View**: Similar to SearchFragment, this fragment features a recycler view for parking bookings, providing comprehensive information.

• **Booking Control**: Users can extend parking durations or end bookings sooner.

•**Effortless Navigation**: A user-friendly navigation bar allows seamless transitions between parking history and current bookings.

The **ProfileFragment** Focusing on personalization and user management, offers a range of user-centric features:

• **User Details**: Displaying usernames, surnames, and email addresses for user identification.

• **Action Buttons**: Users can perform actions like adding, deleting, or modifying parking spaces and access essential information about the app and its creators.

Future Enhancements

While Urban Parking already offers a comprehensive set of features, there are several potential enhancements that could further improve the app's functionality and user experience. These include:

- Predictive Analytics Implementing predictive analytics could help users estimate the likelihood of finding parking at a particular location and time, based on historical data and real-time factors like events or traffic.
- Payment Wallet Integration Offering a digital payment wallet within the app could simplify the payment process and provide users with loyalty rewards or cashback incentives.
- Community Ratings and Reviews Allowing users to rate and review parking spaces could provide valuable insights to others and improve overall parking space quality.
- Multi-Language Support Expanding language options would make the app accessible to a broader audience, particularly in diverse urban environments.
- IoT Integration Integrating Internet of Things (IoT) technology could enable users to reserve parking spots with physical sensors, ensuring the availability of the selected space upon arrival.

References

In the development of Urban Parking, we relied on a host of APIs and technologies to deliver an exceptional user experience. The following references highlight the key features and technologies that have shaped our application:

1. Google Maps and Places API: The integration of Google Maps and Places API has enriched Urban Parking with real-time maps, navigation, route finding, and an autocomplete search bar. This API has significantly improved the user's ability to discover parking spaces, plan routes efficiently, and access location-based information.

2. Google Directions API: The intelligent route navigation feature of Urban Parking is powered by the Google Directions API. It empowers users to find optimal routes to their chosen parking spaces, providing real-time directions and estimated arrival times.

3. Firebase Realtime Database: Firebase Realtime Database serves as the backbone of our data management system. It offers real-time data storage, retrieval, and synchronization. This platform efficiently manages user details, booking history, and parking information.

4. Google Sign-In for Authentication: User authentication is seamlessly facilitated through Google Sign-In. This feature simplifies the login process, ensuring robust user data security and personalized experiences within Urban Parking.

5. Google Pay Integration: Google Pay integration streamlines the payment process for parking reservations in Urban Parking. It provides users with a secure and convenient method to complete transactions, enhancing the overall payment experience

Conclusions

In conclusion, Urban Parking is a feature-rich Android application designed to simplify the parking reservation process and enhance the overall parking experience for users. By seamlessly integrating various APIs such as Google Maps, Google Sign-In, Google Pay, and Firebase Realtime Database, the app offers users a secure, convenient, and efficient platform for finding and booking parking spaces.

The user authentication process, managed through Google Sign-In, ensures both security and ease of access, while the intuitive app flow takes users through a streamlined booking journey. From selecting parking spaces to setting up payment details and completing transactions with Google Pay, the app aims to provide a hassle-free experience.

The user interface and user experience design principles incorporated into MapFragment, SearchFragment, BookingFragment, and ProfileFragment prioritize user-friendliness and efficiency. These design elements, coupled with interactive map features, autocomplete search bars, and detailed booking histories, aim to create a positive and engaging user experience.

Urban Parking represents a significant step towards modernizing and simplifying the parking reservation process in urban environments. With its robust feature set and user-centric design, the app addresses common parking challenges and offers a practical solution for users seeking hassle-free parking experiences.

As technology continues to advance and urbanization intensifies, Urban Parking remains poised to evolve and adapt, providing users with an indispensable tool for navigating the complexities of urban parking