Bilkent University

Department of Computer Engineering

Senior Design Project

Funravel

Project Specifications Report

Fatma Sena Genç - 21901426
Gülin Yılmaz - 21903057
Hissam Mahmoud Elsayed Faramawy - 21901253
Mustafa Efe Tamyapar - 21902856
Öykü Erhan - 21901541

Supervisor: Fazlı Can
Jury Members: Erhan Dolak, Tağmaç Topal
Innovation Expert: Mustafa Sakalsız

Final Reader: Öykü Erhan

Abstract

Funravel is a mobile application which is a travel planning assistance specific to the concert events. The application proposes an engineering solution with optimizing the time spent on planning for an event by finding the most optimal travel package. This package includes accommodation and transportation according to the user's preferences. Also using this mobile application, users can connect with each other using group chats and plan their travel together. Goals and functionalities of Funravel are discussed in detail in this specification report.

Keywords: Trip Optimization, Concert, Group Travel, Mobile Application, Travel Package

Contents

1. Introduction

Planning a trip for an event outside your city can be time-consuming and hard. According to Ticketmaster's database, passionate fans, which is around 10% of an artist's fans, travel 100+ miles for a concert and even some of them cross the Atlantic [1]. Target users of this application are the type of fans who are willing to travel far away and make a trip. Moreover, the average person plans their trip in 10 hours as surveys claim [2]. Funravel wants to optimise this planning time spent, by suggesting to users a travel plan for an event and calculating the total cost of their travel, including accommodation and transportation. Essentially, Funravel is an event-based travel optimization mobile application. Funravel also aims to connect users with each other who do not want to attend these events all by themselves. Overall, it wants to reduce time spent on the planning of far-away events as much as possible with the possibility of group planning.

The innovation that is planned to implement is an offering type of innovation. The main goal of this application is to assist fans who want to attend a concert far away. In this offering, combining multiple platforms to ease the planning for an event trip is intended. The innovation type of Funravel is incremental innovation because Funravel adds new functionalities to the services which are currently in the market.

The rest of the report includes a detailed explanation of the application, constraints of the project, professional and ethical issues, risks, expected goals, optional future works and lastly functional and non-functional requirements.

1.1 Description

The main goal of Funravel is to be a travel assistant for fans who want to attend an event that can be distant. Concert events are the main focus of the Funravel app. Using the user's preferences Funravel finds the most optimal travel package for the user. Other functionalities also aim to help users with planning trips for events, creating posts, and finding groups to attend with. Users can connect with each other before the event and plan their trip together using group chats. Moreover, users can share posts that include reviews about their experience. In the application, each user has a profile page. On this page, posts and invitations generated by the user, profile picture, and the top 3 singers of the user are shown.

1.1.1 Suggesting a Travel Plan for Concerts

On the homepage of the app, there is a search bar where users can search for the event. While searching, users can filter city and country. After finding the event, the user can directly go to the ticket-buying website using the redirection button. Also, the user can select a package. After selecting a package, the user selects the desired period of stay at the location, hotel preference, starting location and desired transportation type. The hotel preference is the hotel rating and the desired transportation type includes bus and plane. After collecting all this information, Funravel finds the most optimised package for the user in terms of budget and distance between location of the event and hotel. The total calculated cost and redirection buttons for the ticket, and booking sites are shown to the user. After creating a package, if the user wishes to see the package later, they can save the package using the save button. The saved packages with calculated costs are shown in the saved events page.

1.1.2 Creating Review Posts

Funravel helps fans who want to share their experience with other people after the concert. Users can create posts on the profile page about the concert they attended. These posts will be shown on the page of each event. They include text and a rate from 10. Using these, users can share the kind of experience they had in the concert. This way, users can express their feelings using the application and they will be encouraged to use the application more.

1.1.3 Group chats and Invitations

Users can share a group invitation for connecting with others who want to attend the same event. These invitations can be seen through a separate invitations page in the application. The explanation written by the user and information about the concert is shown in each of the group invitations. The user can click the join group chat button by using these group invitations. In the group chat, users can communicate and plan a concert event together. Each user can see the group chat they have joined on the group chats page.

1.2 Constraints

Different constraint types of the project are explained in this part.

1.2.1 Implementational Constraints

● The platforms of the project are planned to be both Android and IOS.
● Flutter will be used as a framework of the application.
● In order to work simultaneously, Git and Github will be used as a version control system.
● Firebase (NoSQL database) and MySQL databases will be used to store and handle the data.
● Dart will be used as a programming language.
● Ticketmaster API, Flight Data API, Flixbus API, Google Hotel Prices APIs and other similar travel APIs will be used to collect information.

1.2.2 Economic Constraints

● Firebase and MySQL database services are free to use.
● The frameworks, implementation tools and APIs used are free.

1.2.3 Language Constraints

● Both Android and IOS parts of the project will support English and Turkish.

1.2.4 Ethical Constraints

● Personal data will not be shared with any third-party application.
● Hashing will be used to encrypt confidential data such as passwords.
● User history trips might be used for future work. To maintain users' privacy, this data should be encrypted before it is stored in the database. And only get decrypted when used for the AI suggestions model.

1.2.5 Social Constraints

● The users will be able to communicate with each other with the help of group chats that have been created via a user who is looking for a group to attend a specific event.

1.2.6 Sustainability Constraints

- Statistics about users' views will be collected anonymously on Funravel. Later this data might be used to update the application for a better user experience.
- The business model of Funravel has two revenue streams: first one is the commissions via each successful booking and ticket purchase. Second one is the advertisements within the app.

1.2.7 Time Constraints

- The full implementation of Funravel will take two semesters, however until the first demo which is around the end of the first semester, the core functionalities will be implemented.

1.2.8 Health & Safety Constraints

- The application will offer travel options to places where there are not any current safety or health problems such as wars or epidemic diseases.

1.3 Professional and Ethical Issues

Funravel utilises users' information and preferences, integrates social environments and provides a platform for those who want to find people to attend events with. These features might bring ethical issues that needed to be handled. Privacy should be protected and the safety of users should be managed in an efficient way with the available tools. These features also must follow the local Turkish laws as well as the General Data Protection Regulation, GDPR [3].

1.4 Risks

- Possibility of losing the data of the users.
- Possibility of leaking the personal data of users.
- Fetching incorrect data from the APIs for event, accommodation and transportation.
- Not being able to sync event status. For instance, events can be rescheduled or cancelled but this does not reflect on API and therefore on the application.

1.5 Expected Goals

- Successfully suggesting travel packages for the user.

● Having a unique, enjoyable user experience.
● Implementing the functionalities correctly that are stated in this report.
● Implementing a bug-free and high performing application.

1.6 Optional Future Works

● First feature that can be implemented later is personalised notifications. The system should be able to detect and send notifications about the events that the user would like based on their preferences and past travels with machine learning algorithms.
● Another feature that could be implemented in the future is to include different types of events in our application apart from concerts, such as football matches.
● For the current version we want to focus on popular cities in Europe. Later we plan to expand to the US and other popular cities as well.
● Lastly, to ensure a safe environment for every user, inappropriate-word filters for posts and to report offending posts and comments feature could be implemented.

2. Requirements

2.1 Functional Requirements

● The system should be able to suggest generated travel plans for an event after receiving the preferred dates and start points from the user.
● The system should be able to show the total cost and redirect the user to the related booking and means of transport reservation website (Booking.com, TripAdvisor etc.) for the trip plan of the concert event.
● The system should display the type of the currency according to the users' preference.
● The user should be able to create their accounts with their email and password.
● The user should be able to choose the accommodations in which they want to stay according to the star rating of that place.
● The user should be able to select the date range for the trip they wanted to attend.
● The user should be able to see the upcoming events on the home page.
● The user should be able to filter the city and country of the upcoming events in order to choose a specific location.
● The user should be able to see details and shared posts related to the events.

- The user should be able to create a review post to write their comment on a specific event.
- The user should be able to create group invites in order to connect with other users. The system will automatically create a group chat regarding that invite.
- The user should be able to see the group invitations other users created and join their chat rooms in order to find a group to attend an event with.
- The user should be able to see their own group invitations and review posts on their profile page.
- The user should be able to see the group invites they attended at their group chats page.
- The user should be able to leave the chat rooms when they wish to do so.

2.2 Non-functional Requirements

User friendliness / Usability: The user interface is intended to be as convenient but uncomplicated as possible so that the users will be able to adapt to it easily. There will be use of components that the majority of internet users are accustomed to. Additionally, a user interface manual will be provided.

Maintainability: The system should be capable of being modified to fix the fault or improve the performance in a cost-effective way. Some technical approaches will be used such as high cohesion, loose coupling and using standard API formats and clear document interfaces. In that way, it will be easy to track the code and see where the errors might be occurring. Also, Flutter's built-in tools will be used to catch errors and write test cases.

Flexibility: The system should be open to changes and adding new features since the market and demands of the customers rapidly change. The software should be easily modified to adapt to different environments and user expectations.

Scalability: The system should be able to handle expansion and adapt its resources to changing demands. Additionally, system architecture should be developed in such a way that it enables for easy change and component upgrade/downgrade.

Portability: Flutter is a cross-platform SDK for mobile applications. So, the system should be working on different mobile platforms such as IOS and Android simultaneously with small time and effort.

3. References

[1] Ticketmaster, "Concert Road Trips: How Far Have Fans Traveled to See the Artists They Love?" *Ticketmaster Blog*, 16-Apr-2020. [Online]. Available: https://blog.ticketmaster.com/concert-road-trips/. [Accessed: 17-Oct-2022].

[2] R. Knight, "Average person spends 10 hours planning their holiday, survey claims" *The Independent*, 28-Feb-2019. [Online]. Available: https://www.independent.co.uk/travel/holiday-booking-planning-travel-survey-tourist-a8801211.html. [Accessed: 17-Oct-2022].

[3] The European Parliament and the Council of the European Union, "General Data Protection Regulation," 27-Sep-2022. [Online]. Available: https://gdpr-info.eu/. [Accessed: 17-Oct-2022].