

**REPUBLIC OF TURKEY
YILDIZ TECHNICAL UNIVERSITY
DEPARTMENT OF COMPUTER ENGINEERING**



GEZİ-YORUM

13011036 – Tarık Nural

13011035 – Murat Baki Yücel

SENIOR PROJECT

Advisor

Assist. Prof. Dr. Ahmet Tevfik İNAN

January, 2018

ACKNOWLEDGEMENTS

We would like to express our sincere gratitude to our advisor "Assist. Prof. Dr. Ahmet Tevfik INAN" for the continuous support to our graduation project, his patience, motivation, and knowledge.

Tarık Nural
Murat Baki Yücel

TABLE OF CONTENTS

LIST OF ABBREVIATIONS	v
LIST OF FIGURES	vi
LIST OF TABLES	viii
ABSTRACT	ix
ÖZET	x
1 Introduction	1
1.1 Literature Review	1
1.2 Objective of the Thesis	2
1.3 Hypothesis	2
2 General Information	3
2.1 Trip Tracker (Android application)	3
2.2 Route Tracker (Android application)	3
2.3 Polarsteps - Travel Tracker (Android and Ios application)	3
2.4 Maps (Android and Ios application)	4
3 Feasibility	5
3.1 Technic Feasibility	5
3.1.1 Software Feasibility	5
3.1.2 Hardware Feasibility	9
3.1.3 Communication Feasibility	10
3.2 Labor Force Feasibility	12
3.3 Legitimate Feasibility	12
3.4 Time Feasibility	13
3.5 Economic Feasibility	16
4 System Analysis	17
5 System Architecture Design	19
5.1 Database Design	20

5.2 Modules	22
5.2.1 User Register Module	22
5.2.2 User Login Module	22
5.2.3 User Password Reset Module	22
5.2.4 Search Module	22
5.2.5 Location Save Module	22
5.2.6 Media Save Module	22
5.2.7 Feature Extraction Module	22
5.2.8 Trip Management Module	23
5.2.9 Notification Sender Module	23
5.2.10 Trip Upload Module	23
5.2.11 Trip Download Module	23
5.2.12 Team Trip Preparing Module	23
5.2.13 Team Content Merging Module	23
5.2.14 Team Members Location Tracking Module	23
5.2.15 Content Complaint Module	24
5.2.16 Shared Content Interaction Module	24
5.2.17 Administrator Module	24
5.2.18 Users Interaction Module	24
5.2.19 Media Supplementation Module	24
6 Application	27
6.1 Web Application	27
6.2 Mobile Application	44
7 Experimental Results And Performance Analysis	56
7.1 Experimental Results	56
7.2 Performance Analysis	56
7.2.1 Result	66
8 Conclusion	67
References	69
Curriculum Vitae	72

LIST OF ABBREVIATIONS

API	Application Programming Interface
AWS	Amazon Web Services
CSS	Cascading Style Sheet
DBA	Database Administrator
FTP	File Transfer Protocol
GPS	Geographic Positioning System
IDE	Integrated Development Environment
iOS	iPhone/iPad Operating System
Js	JavaScript
JSON	Java Script Object Notation
LOOP	Location and Observation Platform by Microsoft
Play Store	Google Play Store
RAM	Random Access Memory
RDS	Relational Database Service
SCP	Secure - Contain - Protect
SDK	Software Development Kit
SFTP	Secure File Transfer Program
SQL	Structured Query Language
SSH	Secure Shell
STS	Spring Tool Suite
VCS	Version Control System
VNC	Virtual Network Computing
GUI	Graphical User Interface
HTTP	Hyper Text Transfer Protocol

LIST OF FIGURES

Figure 3.1 Resource Chart	12
Figure 3.2 Gantt Diagram Drawing	14
Figure 3.3 PERT Diagram Drawing	15
Figure 4.1 System Schema	18
Figure 4.2 User Roles	18
Figure 5.1 Mobile Application Database Diagram	20
Figure 5.2 Server Side Database Diagram	21
Figure 5.3 Level zero data flow diagram	25
Figure 5.4 Level one data flow diagram	25
Figure 5.5 Level two user data flow diagram	26
Figure 5.6 Level two administrator data flow diagram	26
Figure 6.1 Main page of the website	27
Figure 6.2 Register screen for users	28
Figure 6.3 Activation screen for users	29
Figure 6.4 Forgot my password screen for users	30
Figure 6.5 Support screen for users	31
Figure 6.6 Home screen for users	32
Figure 6.7 Responsive home screen for users	33
Figure 6.8 Notification screen for users	34
Figure 6.9 Profile page for users	35
Figure 6.10 Profile page for users	36
Figure 6.11 Profile page for users	37
Figure 6.12 User page	38
Figure 6.13 Creating complaint screen for users, trips and comments	39
Figure 6.14 Trip details screen	40
Figure 6.15 Search page	41
Figure 6.16 Administrator page	42
Figure 6.17 Administrator page	43
Figure 6.18 Mobile application register page	44
Figure 6.19 Mobile application login page	45
Figure 6.20 Mobile application home page	46

Figure 6.21 Mobile application notification page	47
Figure 6.22 Mobile application profile page	48
Figure 6.23 Mobile application search page	49
Figure 6.24 Mobile application trip download module	50
Figure 6.25 Mobile application downloaded trip section	51
Figure 6.26 Mobile application downloaded trip section	52
Figure 6.27 Mobile application trip creating module	53
Figure 6.28 Mobile application trip managing medias while creating trip module	54
Figure 6.29 Mobile application timeline module	55
Figure 7.1 Example HTTP call	56
Figure 7.2 When server is idle position	57
Figure 7.3 Server 30 GET HTTP calls simultaneously	58
Figure 7.4 Server 30 GET HTTP calls simultaneously	59
Figure 7.5 Server 100 GET HTTP calls simultaneously	60
Figure 7.6 Server 100 GET HTTP calls simultaneously	61
Figure 7.7 Server 130 GET and POST HTTP calls simultaneously	62
Figure 7.8 Server 130 GET and POST HTTP calls simultaneously	63
Figure 7.9 Server 180 GET and two POST HTTP calls simultaneously	64
Figure 7.10 Server 180 GET and two POST HTTP calls simultaneously	65
Figure 7.11 Statistical summary report of HTTP calls	66

LIST OF TABLES

Table 2.1 Specifications of Applications	4
Table 3.1 Minimum System Requirement for Mobile Application Development	10
Table 3.2 Minimum System Requirement for Server Side Application Development	10
Table 3.3 Android Hardware Requirements	10
Table 3.4 Communication parameters	11
Table 3.5 Scaleway package options	12
Table 3.6 Personnel cost table	16
Table 3.7 Hardware and software used for development cost table	16

ABSTRACT

Gezi-Yorum

Tarık Nural

Murat Baki Yücel

Department of Computer Engineering

Senior Project

Advisor: Assist. Prof. Dr. Ahmet Tevfik İNAN

The goal of this project is to record travel routes of people's trips using mobile device location and to add and edit media on the route. A system will be developed that can be used to organize trips not only as a person but also as a team, as well as what vehicles are used for transportation. If the trip organized as a team, the users will be able to share their in-team location via the Internet so that team members can follow each other. It is intended that a route created and shared by a user or a team can be examined by other users. Other users may choose the route they are reviewing as their route and they can start trips on this route as personal or team trip. Mobile application will show chosen path on the map different from user's. The system also need to create a social media environment to increase interaction between users. Friendship and tracking system between users will be designed and also personalized news flow which includes shared trips by user's friends and popular trips, will be provided. As a result of the project, a mobile application will be developed that stores route data of a trip and media like photos, videos, audio files tagged on the route, also will provide service to interact people with shared data. Sharing trips is an additional workload for travelers. This application will offer users a practical solution to save the time spent on sharing a trip on any social media environment or on the internet. In addition, the application will generate convenience not only for travelers but also for people who want to share their daily life. It will also provide an open environment for the interaction of people as it is considered to be a social media environment within the application.

Keywords: Trip, Tracker, Advisor, Social Media, Gallery Editor

ÖZET

Tarık Nural

Murat Baki Yücel

Bilgisayar Mühendisliği Bölümü

Bitirme Projesi

Danışman: Yrd. Doç. Dr. Ahmet Tevfik İNAN

Bu projenin hedefi insanların gezilerinin mobil cihaz konum verileri kullanılarak gezi güzergâhının kaydedilmesi ve güzergâh üzerinde medya ekleyip düzenlenmesini sağlamaktır. Gezilerin sadece kişiler olarak değil, takım hâlinde de düzenlenebilmesi, ayrıca yapılacak gezilerde varsa kullanılan ulaşım araçlarının neler olduğunu algılayabilecek bir sistem geliştirilecektir. Takım olarak düzenlenen gezilerde kullanıcılar, takım üyelerinin birbirini takip etmesi amacıyla takım içi konum paylaşımını Internet bağlantısı olması şartıyla yapabilecektir. Bir kullanıcının veya takımın oluşturduğu bir güzergâhı, diğer kullanıcıların da inceleyebilmesi hedeflenmektedir. Diğer kullanıcılar inceleme yaptığı güzerâhi kendi güzergâhi olarak belirleyebilir veya bu güzergâh üzerinde takım gezisi yapmak isteyebilir. Mobil uygulama belirlenen güzergâhi, kullanıcının güzergâhından farklı olarak gösterecektir. Sistemin kullanıcılar arasında etkileşimi artırmak amacıyla bir sosyal medya ortamı oluşturmazı gerekecektir. Kullanıcılar arası arkadaşlık ve takip sistemi tasarlanacaktır. Ayrıca kişiye kendi çevresinden tavsiye edilender, uygulama içinde bulunan popüler güzergâhlar ve arkadaş çevresindeki geziler derlenerek özel bir haber akışı içerisinde sunulacaktır. Proje sonucunda bir gezinin rotasını, rota üzerine etiketlenen fotoğraflarını, videolarını, ses dosyalarını saklamaya imkan veren ve bunları diğer kullanıcıların etkileşimine açabilen bir mobil uygulama geliştirilecektir. Gezilerin herhangi bir sosyal medya ortamında veya Internet üzerinde paylaşılması ve paylaşmak amacıyla düzenlenmesi geziciler için ek iş yükü teşkil etmektedir. Bu uygulama, kullanıcılarına gezi sürecinin paylaşımında sarf edilen eftordan tasarruf ettirecek pratik bir çözüm sunacaktır. Ayrıca uygulama sadece gezicileri değil günlük hayatını paylaşmak isteyen insanlar içinde kolaylık üretecektir. Uygulama içinde bir sosyal medya ortamının da olması düşünüldüğü için kişilerin etkileşimine açık bir

ortam sağlayacaktır.

Anahtar Kelimeler: Gezi, Takip, Öneri Sistemi, Sosyal Medya, Galeri Düzenleyici

1

Introduction

On this project our purpose is to save navigation data which collected from user's mobile device represents latitude, longitude and altitude provided by GPS of mobile device and show the route of a trip on the map. There will be a mobile application and server side application to provide some features, lets users to share trips, review trips, communicate and socialize with each others. Various media records such as photos, videos and sound records recorded from mobile device during the trip will be associated with the route. There will be a team trip option for the trips which includes more than one member. Team trip option lets every team member to see any team member's location simultaneously if there is an Internet connection. In order to determine travelling method, using the temporal data changes of the navigation data taken from the GPS, average and instant speed will calculated. As result, trip type will be labelled as vehicle trip, walking trip, running trip or cycling trip using calculated speed data. In order to provide interaction between users a social media environment will be designed. People will be able to establish their friendship here. In the news flow, there will be a timeline section. Posts which are shared by friends of users will be shown there. The trips can be liked by other users in the news flow, they can leave a comment on trip or the users can follow trips they like by downloading it on their own mobile device.

1.1 Literature Review

Our purpose to realize that project is to simplify planning, recording and sharing traveler's trips. We know that it is not easy to organize photos and media taken on a trip and later on sharing them on any social media or personal blogs. In any marketplaces including play store and appstore there are lots of applications for tracking and planning travel but most of them only saves location data and do not include media or user notes. There are only a few remarkable application which saves location data, media and also provides a social media enviroment. However there is no application that provides trip tracking, organizing media, following other person's

route, sharing memories and also doing all of these with friends at the same time.

1.2 Objective of the Thesis

For travelers it is not easy to organize media and associate media with saved locations. Also if trip has been organized with more than one person it is also difficult to merge media and organize additional material prior to share on any social media environment or on the Internet. It's important to say that every member of team has him/her own memory which is important and it must be saved.

1.3 Hypothesis

To do that it is necessary to store locations data, store media and associate them with their location, and store them with everything in a trip. So that people can remember every single detail of their trip.

2

General Information

The main purpose of this section to review projects that are already developed by others which is related with ours.

2.1 Trip Tracker (Android application)

Trip Tracker[1], a Microsoft Garage project, is a free application that automatically records your drives, runs, walks, and bike rides. This application has developed with LOOP [2]. This application can only track, store user's location and show stored data on the map.

2.2 Route Tracker (Android application)

Route Tracker [3] is a GPS application for tracking/loading routes in your android mobile phone that gives you a real time response on your location. It supports GPX tracks file import, GPX/KML/google map embedded HTML file export, sync workouts with RunKeeper account, auto-Lap for each mile/kilometer, audio reminder for auto lap time reminding, vibration for auto lap, sharing to friends, open street view for record locations, duration, distance, pace and calories indication.

2.3 Polarsteps - Travel Tracker (Android and Ios application)

Polarsteps [4] can provide tracking location of user, can show media as just photo and video on the trip path. They are claiming that it is a low power consuming application. In this application it is not possible to organize team trips. User can use this application only for their personal trips. This application also provides a social media platform. Social media enviroment could only usable for sharing trips and comment on shared content. It is not possible to download and track any shared route.

2.4 Maps (Android and Ios application)

Google Maps [5] can save user's location continuously. In this application there is no start trip option. This application determines trip kind of trip automatically. This application is not developed for tracking, it is developed for navigate users. User can watch his/her activity on the menu named 'TimeLine'. On users timeline any photos and videos are shown, there is no audio or text notes support. In this application there is no social media or sharing trip option.

Gezi Yorum offers an extended trip tracking and organizing system. Gezi Yorum will include all of the features of the applications listed above and in addition Gezi Yorum will be able to organize team trips and follow member on a trip. Table 2.1 shows specifications of applications.

Table 2.1 Specifications of Applications

Specification	Trip Tracker	Route Tracker	Polarsteps	Maps	Gezi Yorum
Record trip	+	+	+	+	+
Show trip on map	+	+	+	+	+
Trip type	+	+	+	+	+
Step counting	-	-	-	-	+
Media on location	-	-	+	-	+
Social media	-	-	+	-	+
Personal trip	+	+	+	-	+
Team trip	-	-	-	-	+
Live tracking	-	-	-	-	+

3

Feasibility

This system requirements are listed below:

- This system needs real time location of a user. The mobile phones have already a GPS sensor so this system needs to run on a mobile device.
- This system must provide interaction within users. So to provide interaction of multiple mobile devices this system needs a server side application.
- In process of development to provide version controlling Git must used as VCS.

3.1 Technic Feasibility

As technical feasibility study, the software, hardware, communication, labor force, legal and economical needs for the project is defined on the following sections.

3.1.1 Software Feasibility

This project depends on web and mobile technologies.

Mobile Side Development

The tools and development environments used for mobile side development of the project are mentioned below.

- Android [6]: Android is a mobile operating system based on the Linux kernel. Its source code is licensed under open source licenses and it is developing by Google and Open Handset Alliance. The top layer of Android's architecture is called The Application Framework layer and it provides many higher-level services to applications in the form of Java classes. Android was chosen over iOS [7] because of publishing problems, restrictions and lack of design guidelines that come with iOS.
- Android Studio: Android Studio is the official IDE for Android application development, based on IntelliJ IDEA. Android Studio offers some advantages over

Eclipse, such as Gradle based flexible build system, advanced layout editor, built-in Git source control and Maven library support [8].

- Android Sqlite Database: SQLite is a opensource SQL database that stores data to a text file on a device. Android comes in with built in SQLite database implementation. SQLite supports all the relational database features [9].
- Android Emulator: Android emulator lets prototype, develop and test Android applications without using a physical device [10].
- Google Map API: Google Maps APIs give developers several ways of embedding Google Maps into web pages or retrieving data from Google Maps, and allow for either simple use or extensive customization [11].
- Operating System: An operating system is required for developing mobile application. Android Studio and Android Emulator can run on Windows, Linux or Mac OS [12]. We prefer to use Linux, because Linux needs low system properties and it's free to use.

Server Side Development

The tools and development environments used for server side development of the project are mentioned below.

- Java EE: Java EE is a collection of technologies and APIs for the Java platform designed to support "Enterprise" Applications which can generally be classed as large-scale, distributed, transactional and highly-available applications designed to support mission-critical business requirements. [13] We have chosen JAVA EE platform because it is widely used by developers, its community is larger than most of its competitors, also it is a better option to use same language which is Java for the server side and mobile side development
- Eclipse: Eclipse platform which provides the foundation for the Eclipse IDE is composed of plug-ins and is designed to be extensible using additional plug-ins. Developed using Java, the Eclipse platform can be used to develop rich client applications, integrated development environments and other tools. [14] Eclipse can be used as an IDE for any programming language for which a plug-in is available. Eclipse has support for JAVA EE and Spring projects within its marketplace there are

few IDEs that have these functionalities but Eclipse is open source and free.

- Sublime Text: Sublime Text is a proprietary cross-platform source code editor with a Python API. It natively supports many programming languages and markup languages, and functions can be added by users with plugins, typically community-built and maintained under free-software licenses. [15] Sublime Text is one of the greatest text editor in the world, almost every programming language is supported, we chosen it because of its Angular JS editing skills.
- MySQL : A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. [16] MySQL is open-source and free. Developers didn't need to pay licence fee, also it gets updates regularly which makes it reliable.
- MySQL Workbench: MySQL Workbench is a unified visual tool for database architects, developers, and DBAs. MySQL Workbench provides data modelling, SQL development, and comprehensive administration tools for server configuration, user administration, backup and many more features. [17] It is best option for MySQL development platform, when we compare to others such as Apache PhpMyAdmin it contains reverse and forward engineering tools, embedded uml diagram chart drawer tools, generating tables from models without complexity.
- Bootstrap: Bootstrap is a front-end development framework that enables developers and designers to quickly build fully responsive websites. The framework contains global CSS settings with built-in components and extensible classes in the form of typography, navigation, buttons and many other html elements. [18] We have chosen bootstrap because bootstrap equals platform in dependency, by developing on Bootstrap developers are able to run their code on every size of device such as phones,tables,computers,laptops and so on.
- Angular JS: AngularJS is a MVVM platform. AngularJS is a structural framework for dynamic web applications. It lets you use HTML as your template language and lets you extend HTML's syntax to express your application's components clearly and succinctly. AngularJS's data binding and dependency injection eliminate much of the code you would otherwise have to write. [19] We have chosen Angular JS because it is officially supported and created by the Google. What that means is that it is more reliable than others. Also it has more libraries than other frameworks its community larger than others.
- Javascript : Javascript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations

allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities. [20] We have chosen it because it is almost only option for front end development.

- Apache Tomcat : The Apache Tomcat software is an open source implementation of the Java Servlet, Java Server Pages, Java Expression Language and Java Web Socket technologies. The Java Servlet, Java Server Pages, Java Expression Language and Java WebSocket specification are developed under the Java Community Process. [21] We have chosen Apache Tomcat because it is platform independent, we can run web server in a Linux machine.
- Spring Framework : The Spring Framework provides a comprehensive programming and configuration model for modern Java-based enterprise applications - on any kind of deployment platform. [22] A key element of Spring is infrastructural support at the application level: Spring focuses on the "plumbing" of enterprise applications so that teams can focus on application-level business logic, without unnecessary ties to deployment environments. Spring framework makes things easy for developers, such as creating Web Services, Controllers, web pages, database integrations, query implementations and so on. It is easy because most of the trivial processes are implemented by the framework, for example you don't need to open and close database connections by yourself when you execute a query in JAVA it is done by Spring it makes code less complex.
- STS: The Spring Tool Suite is an Eclipse-based development environment that is customized for developing Spring applications. It provides a ready-to-use environment to implement, debug, run, and deploy your Spring applications, including integrations for Pivotal tc Server, Pivotal Cloud Foundry, Git, Maven, AspectJ, and comes on top of the latest Eclipse releases. [23] We have chosen STS because it has embedded Eclipse plug-in integrated, there are other options such as Spring Incubator web page, but when you use others you have to implement some extra steps to import those projects in your local computer.
- Postman : API development tool that is a plug-in which comes as a packaged application in Chrome and is used to test the API services. Users can test the JSON REST based Web Services with this and make a API for cross device applications. [24] We have chosen Postman because its perfect GUI. You can save your REST calls and look it from other computers by registering.
- Google Map API: Google Maps APIs give developers several ways of embedding Google Maps into web pages or retrieving data from Google Maps, and allow for either simple use or extensive customization [11]. We have chosen Google Map API because

it is free for the first 100.000 requests.

- PuTTy : PuTTY is a Telnet and SSH terminal software for Unix and Windows platforms that enables any users to remotely access computers over the internet. [25] We have chosen PuTTY because it is platform independent and free.
- VNC : Virtual network computing (VNC) is a type of remote-control software that makes it possible to control another computer over a network connection. Keystrokes and mouse clicks are transmitted from one computer to another, allowing technical support staff to manage a desktop, server, or other networked device without being in the same physical location. [26] We have chosen TightVNC as a tool for accessing remote server. TightVNC is also free and it is recommended by the community.
- WinSCP : This program is an open source free SFTP client, FTP client, Web DAV client and SCP client for Windows. Its main function is file transfer between a local and a remote computer. [27] WinSCP is better option with its GUI for accessing remote file system.
- Jmeter : jMeter is an Open Source testing software. It is Java application for load and performance testing. jMeter is designed to cover categories of tests like load, functional, performance, regression, etc., and it requires JDK 5 or higher. [28] We have chosen JMeter to make HTTP calls in order to get server's client limit. It makes simultaneous calls between defined time periods and shows the summary of call outputs.
- htop : htop is an interactive system-monitor process-viewer and process-manager. It is designed as an alternative to the Unix program top. It shows a frequently updated list of the processes running on a computer, normally ordered by the amount of CPU usage. Unlike top, htop provides a full list of processes running, instead of the top resource-consuming processes. Htop uses color and gives visual information about processor, swap and memory status.[29] We have chosen to monitor our server's stress test with htop because it has perfect GUI and it is really robust and easy to monitor.

3.1.2 Hardware Feasibility

The minimum hardware requirements for each program/IDE and a system requirement compilation for development is shown in 3.1, 3.2, 3.3 based on the requirements.

Table 3.1 Minimum System Requirement for Mobile Application Development

Software	CPU (Ghz)	RAM (MB)	Storage (GB)
Linux OS [30]	1	512	8
Android Studio [31]	1.6	3072	8 (500MB for IDE, 7.5 GB for SDK)
Android Emulator [31]	unknown	2048	1.5
Total	2.6	5632	17.5

Table 3.2 Minimum System Requirement for Server Side Application Development

Software	CPU (Ghz)	RAM (MB)	Storage (GB)
Linux OS [30]	1	512	8
Eclipse [32]	1.5GHz	1024	1
MySQL [33]	1 (2 core)	2048	0.8
MySQL Workbench[33]	unknown	4096	0.2
Total[33]	2.5 (2 core)	7168	2

Table 3.3 Android Hardware Requirements

	Minimum	Recommended
CPU (Ghz)	1	2
RAM (MB)	1024	1536
Storage (GB)	8	32

3.1.3 Communication Feasibility

Media files and path files are separately stored on users android device. It is so hard to send all of trip data one by one to one device to another without any loss. So we thought about that to compressing all files in a zip file before sending from mobile device to web server or vice versa. So we send all data in one file. Also we will use less data bandwidth with compress all files in a zip file. The Internet is the main communication technology used in this project. The anticipated communication variables are shown in Table 3.4.

Table 3.4 Communication parameters

Description	Symbol	Values
Average Path Size	A	50 kB
Average Video Size	B	10 MB
Average Video Number Per User	C	4
Average Photo Size	D	2 MB
Average Photo Number Per User	E	20
Average Sound Record Size	F	1 MB
Average Sound Record Number Per User	G	1
Average Number Of Members in a Trip	H	2
Average Trip Data Size on Mobile(Unzipped)	I	81 MB
Average Trip Data Size on Mobile(Zipped)	J	56,7 MB
Average Trip Data Size on Web(Unzipped)	K	162 MB
Average Trip Data Size on Web(Zipped)	L	113 MB
Average Number of Upload Rate	M	1000/month
Average Number of Download and View Rate	N	7500/month
Average Size of Uploaded Trips	O	55 GB
Average Size of Downloaded Trips	P	827 GB
Supposed Number of Users	R	10000
Average Zip Compression Ratio	S	30% [34]
Server Data Rate Per Month	T	827 GB

$$I = G * F + E * D + C * B + A$$

$$K = I * H$$

$$J = I * (1 - R)$$

$$L = K * (1 - R)$$

$$O = J * M$$

$$P = L * N$$

$$T = MAX(O, P)$$

For this project we selected to rent a cloud computing system for make this project scalable. Scaleway [35] provides cloud computing services. Scaleway can provide multiple datacenters on different locations and developer tools on the machines. We selected scaleway for renting cloud server(s).

Based on the calculations above, the monthly data size will be around 827 GB. Scaleway provide 8 GB RAM and 200 GB storage with expandable options on ARM64 package shown on Table 3.5. So we have to rent at least 5 ARM64 cloud servers from

Table 3.5 Scaleway package options

Specification	Starter	C2	ARM64	C1
CPU	2x86 64bit	8x86 64bit	8xARMv8	4xARMv7
RAM	2GB	32GB	8GB	2GB
Storage	50GB SSD	50GB SSD	200GB SSD	50GB SSD
Number of public IPv4	1	1	1	1
Bandwidth	200Mbit/s	800Mbit/s	200Mbit/s	200Mbit/s
Price	2.99 Euro	5.99 Euro	9.99 Euro	24.99 Euro

Scaleway.

3.2 Labor Force Feasibility

The resource chart for this project shown in 3.1.

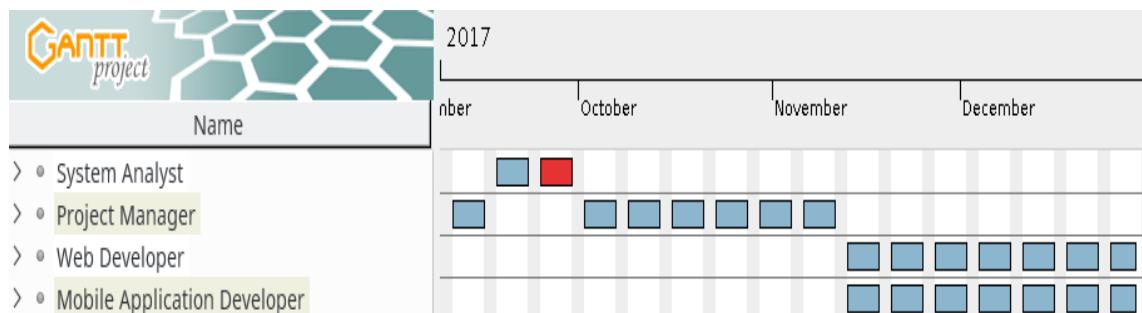


Figure 3.1 Resource Chart

3.3 Legitimate Feasibility

Software which is used within the project does not face any legal issues. All of the software used in the project contain license requirements. Users are responsible for all shared content so any misusing of any sharing content is their risk. Sharing and publishing content is on their own risk.

3.4 Time Feasibility

Gannt diagram shown in Figure 3.2 and PERT diagram shown in figure 3.3.

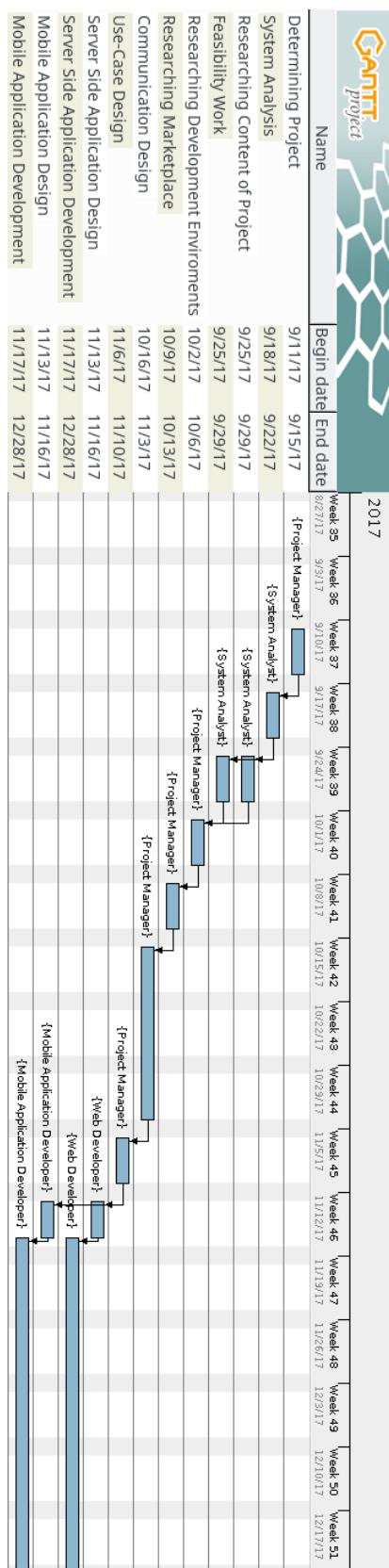


Figure 3.2 Gantt Diagram Drawing

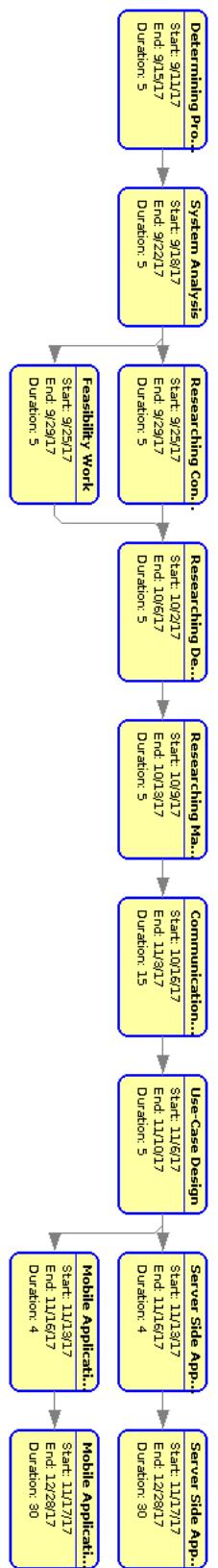


Figure 3.3 PERT Diagram Drawing

3.5 Economic Feasibility

The salary determined by the government of the Republic of Turkey for engineers is 3.500 TL [36]. In a month there are 22 work days on average so an employee's daily salary is $3500 / 22 = 159.10$ TL.

Table 3.6 Personnel cost table

	Time(Day)	Price(TL/Day)	Total(TL)
System Analyst	15	159.10	2.386.5
Project Manager	35	159.10	5.568,5
Full Stack Developer	35	159.10	5.568,5
Mobile Application Developer	35	159.10	5.568,5
General Total:			19.092 TL

Table 3.7 Hardware and software used for development cost table

Components	Price
Eclipse [32]	Free
Android Studio [8]	Free
2* Dell Vostro 5468 [37]	662 TL
General Mobile GM6 [38]	375 TL
2* Github Account	Free
Scaleway Server	310,35
General Total:	1.346 TL

The cost of computers used in the development process in 3181 TL[37]. In the process 2 computer were used. A computer can be used for 2 years so 48 months in average lifetime. From here, we will find the cost of computer as 662TL. General Mobile GM6 used in the development process is 900 TL[38]. It can be used for 12 months in average lifetime. From here we will find the cost of phone as 375 TL. Monthly price of servers is $9.99 * 5 = 49,95$ Euro = 227,09 TL. We need this servers after 17 November 2017 due to 28 December 2017. For this 41 days servers price is $227,09 * (41 / 30) = 310,35$ TL. Considering these expenditures, the cost of hardware and software cost is 1.409 TL. The project cost is $19.092 + 1.346 = 20438$ TL.

4 System Analysis

The project hosts two kinds of roles: admins and users.

The user will start a trip, save location and media, manipulate media share options and share all of contents on social media. Also user can download a shared content which that user has access privileges. User can follow the trip that downloaded. User can search other users; trips by location, name and trip type (walk, run, ride or car). User can organize team trips with other users. During the team trip users can see each other's location on the map.

The admins can view the content of trips or profile of users. They can also evaluate the complaints which has been created by users. They can decide whether they are right or not, if admins think the complaint that is created by users is right they can hide the content of profile,comment or the whole trip. They are also able to view the support messages.

The work model of the system is shown in Figure 4.1.

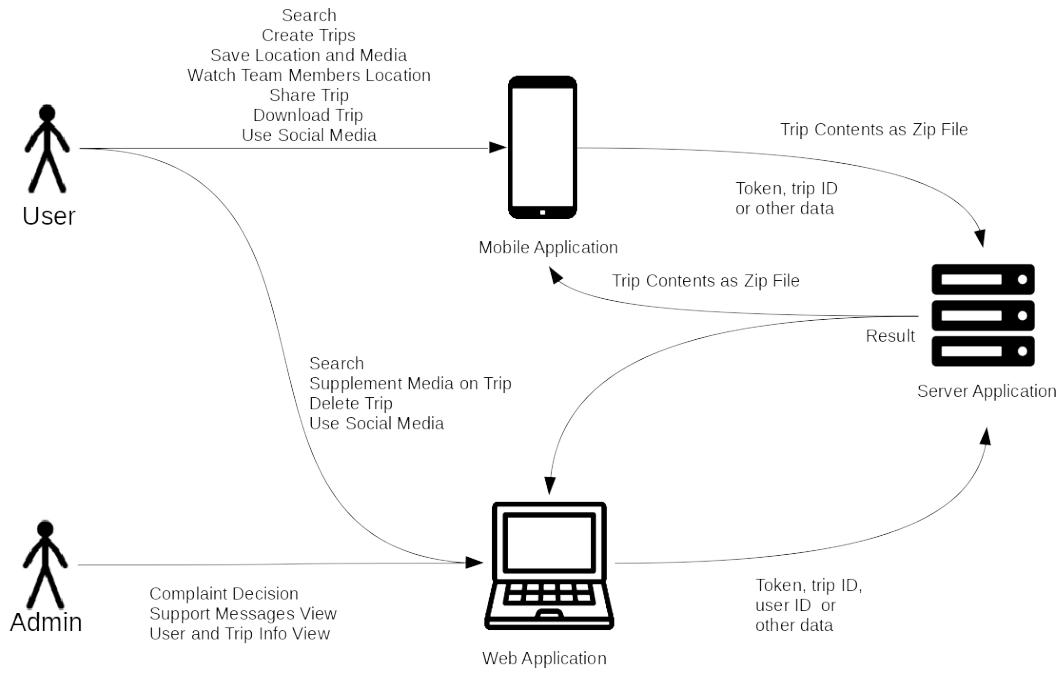


Figure 4.1 System Schema

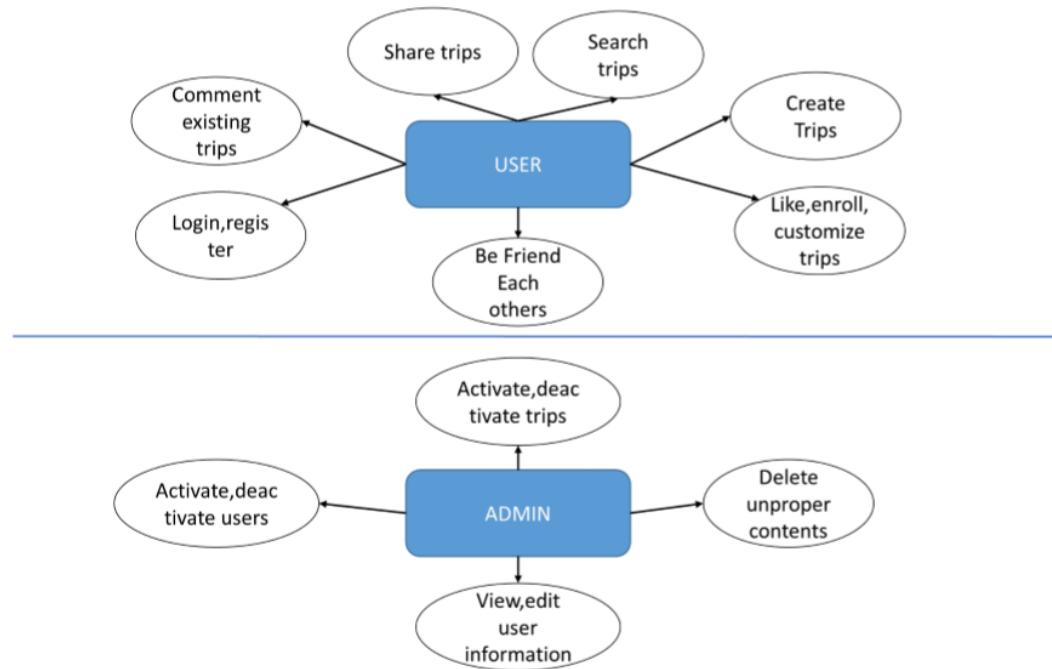


Figure 4.2 User Roles

5

System Architecture Design

Project details and descriptions also architectural design of project are explained in this chapter.

Mobile application will save location, notes and media files also location of media files to storage when user starts trip tracking feature of mobile application. Before starting a trip user can choose some of his/her friends to organize a trip with them when connected to Internet. If a person adds a trip, this person will give an invitation notification which asks user to accept joining or reject joining to trip. If user accepts invitation he/she can join trip as a member. Inviter is accepted as leader of team. During the trip all user's behaviour is saved separately. If location sharing feature is opened any member of this team can watch other members' location. However as we mentioned this feature requires Internet connection. At the end of trip when user wants to share his/her trip data, path data will be provided from team leader but all of media data will be merged. All users can select sharing data separately.

In order to save location in background, Android application will use a location saving service. Using service is a must because, Android services can provide to execute a process for long time in the background. This application must run location save service for long time. User can watch his/her location path on map during trip. If user wants to follow a path which is shared by another user, he/she can follow the path by looking at map.

5.1 Database Design

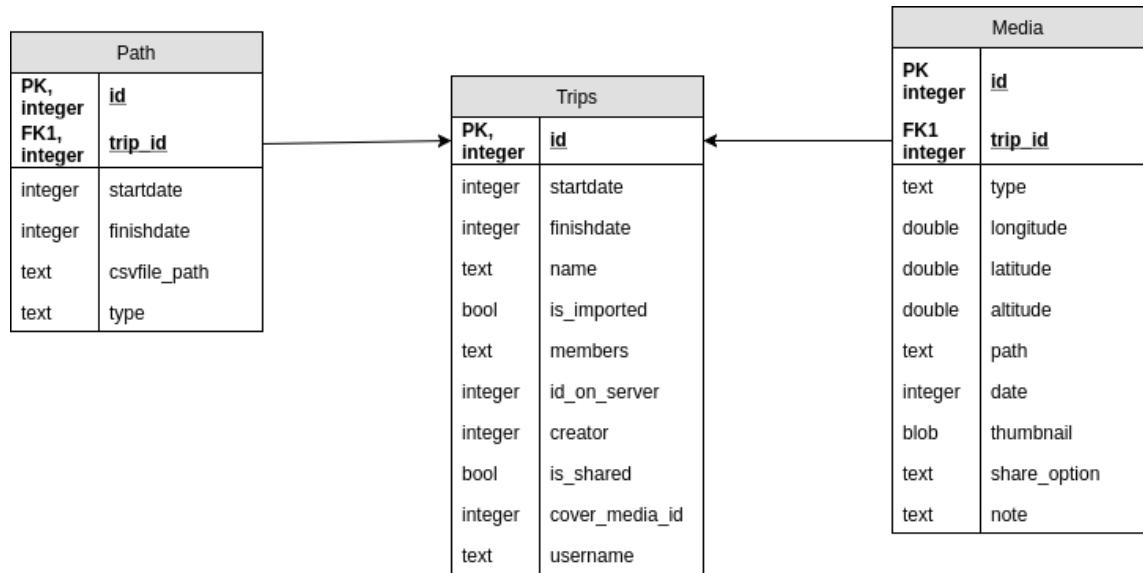


Figure 5.1 Mobile Application Database Diagram

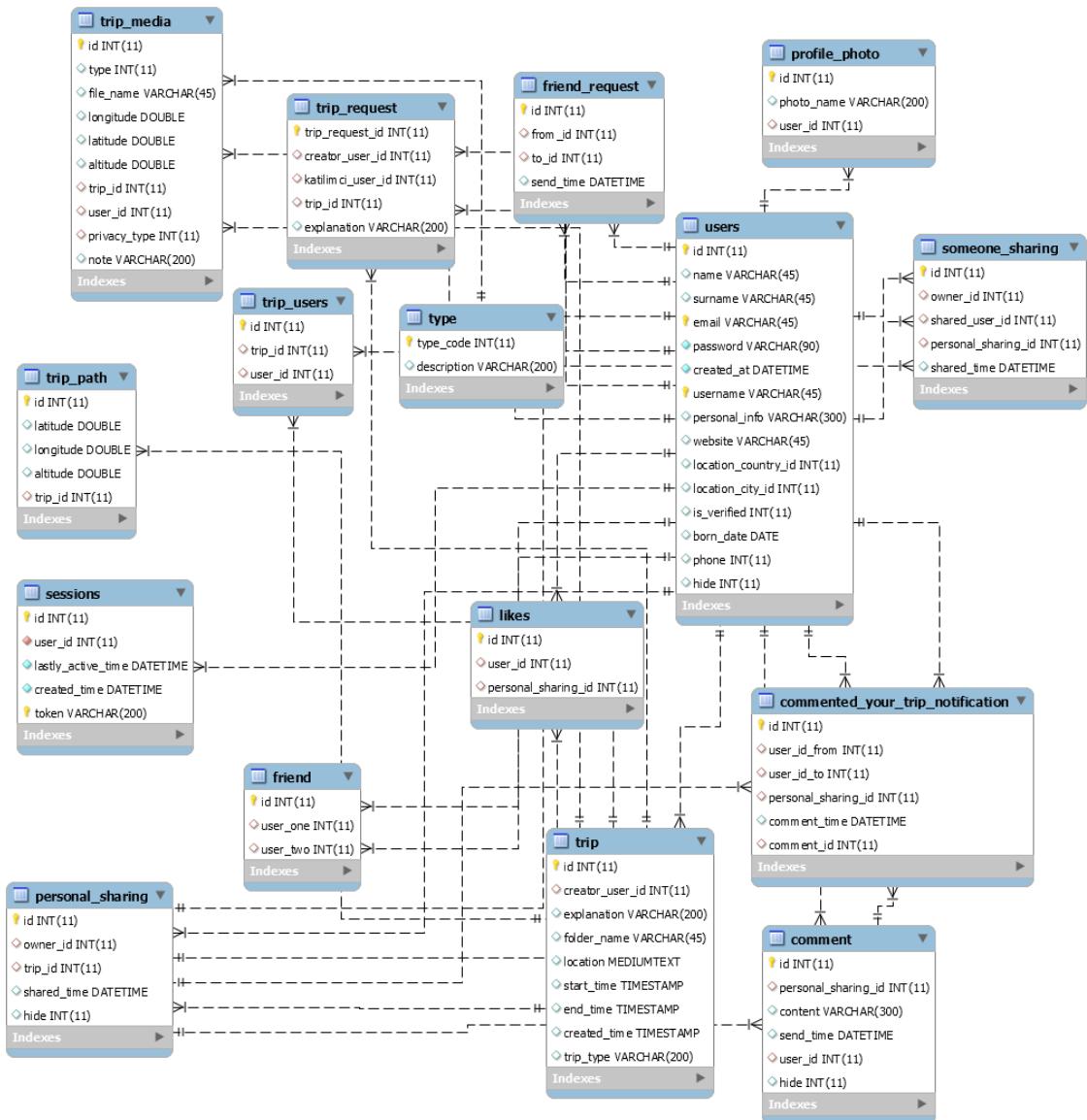


Figure 5.2 Server Side Database Diagram

5.2 Modules

5.2.1 User Register Module

User register module provides creating new accounts, sends activation e-mails and activates user accounts. This module needs user name, password and email from user to create a new account. This module runs on web application but user can use this module on mobile and web application.

5.2.2 User Login Module

User login module provides user to sing in into mobile and web application. This module needs user name and password from user to sing in. This module runs on web application but user can use this module on mobile and web application.

5.2.3 User Password Reset Module

This module provides password reset if the user forgets the password. This module available on both mobile and web application but this module runs on web application.

5.2.4 Search Module

This module provides user to search other users and shared trips by region, user, time and trip type. User can only search on trips which he/she have access rights to see. This module runs on web application but user can use this module on mobile and web application.

5.2.5 Location Save Module

This module runs as a background task on Android device and saves navigation(longitude, latitude, altitude) data provided from GPS to a CSV file. This module runs on mobile application.

5.2.6 Media Save Module

This module saves photo, video and sound record with their location data to SQLite database on Android device. This module runs on mobile application.

5.2.7 Feature Extraction Module

This module tags trips using trip path data by their country, province, distinct, members, time and trip type on the web application. This module runs on mobile

application.

5.2.8 Trip Management Module

This module provides user to manage trip contents like deleting a photo or changing access privileges for contents. This module runs on mobile application.

5.2.9 Notification Sender Module

This module provides sending notifications to users about any topic. This module runs on web application but mobile application use this module as a web service.

5.2.10 Trip Upload Module

This module provides user to share trip content on social media with determined access rights. Mobile application prepares a zip file which includes all of contents belong to trip and upload this zip file to server side.

5.2.11 Trip Download Module

This module provides user to download chosen trip which he/she can access to his/her mobile device using application. Web application prepares a zip file using chosen trip and chosen media files, after that sends it to mobile application.

5.2.12 Team Trip Preparing Module

This module provides users to send team trip requests each others. They can accept or deny these requests, if they accept they can contribute trips by sending their media. This module works on both mobile and web application.

5.2.13 Team Content Merging Module

This module provides trip members to merge trip content easily for share on social media. Web application merges all of sent contents from members in one trip. This module runs on web application.

5.2.14 Team Members Location Tracking Module

This module can only be used on team trips and requires Internet connection, provides team members to track each others locations simultaneously. This module runs on

both mobile and web application.

5.2.15 Content Complaint Module

This module can be used both mobile device and web applications. Users can make complaint about other users, trips, comments. Created complaints are evaluated by the administrators. This module runs on server side but mobile application also interacts with it.

5.2.16 Shared Content Interaction Module

This module can be used both on mobile application and web application. Users can like each others shared trip they can comment on those trips also can share the trips. This module runs on server side but mobile application also interacts with it.

5.2.17 Administrator Module

This module can only be used on web application. Administrator of system are able to interact with it, with the module administrators can evaluate created complaints by the users. Administrators can hide a content,user or comment if he or she thinks that complained content is inappropriate. This module runs on web side application.

5.2.18 Users Interaction Module

This module can be used both web and mobile application. Users can interact with other users by sending friend request, if they accept it they become friends and will be able see contents of each others. This module runs on web side application but can only be used both mobile and web applications.

5.2.19 Media Supplementation Module

This module provides user to add new media files to trip on stated location using Google map on web application. This module only runs on web application.

The dataflow diagrams are shown in Figure 5.2, 5.5, 5.6.

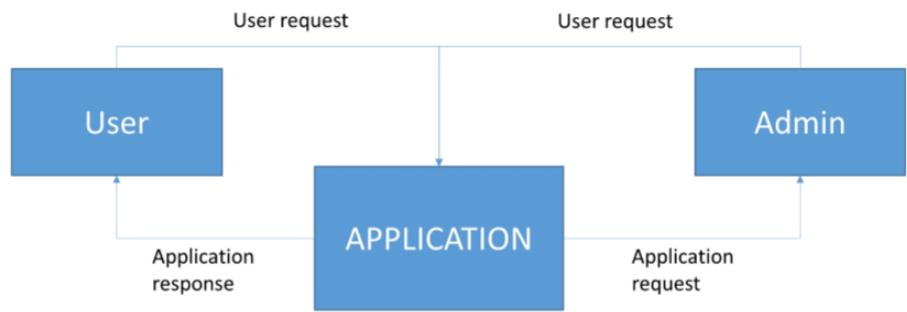


Figure 5.3 Level zero data flow diagram

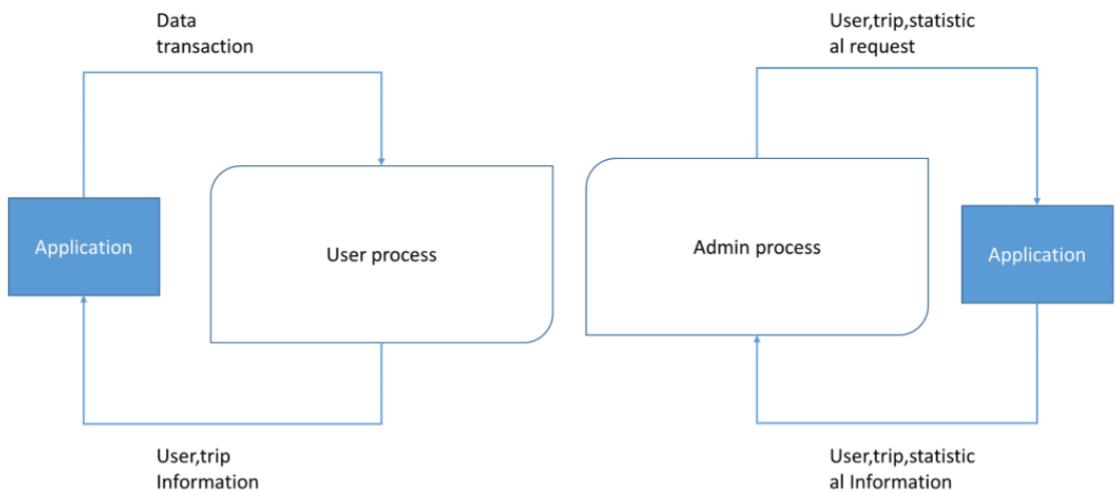


Figure 5.4 Level one data flow diagram

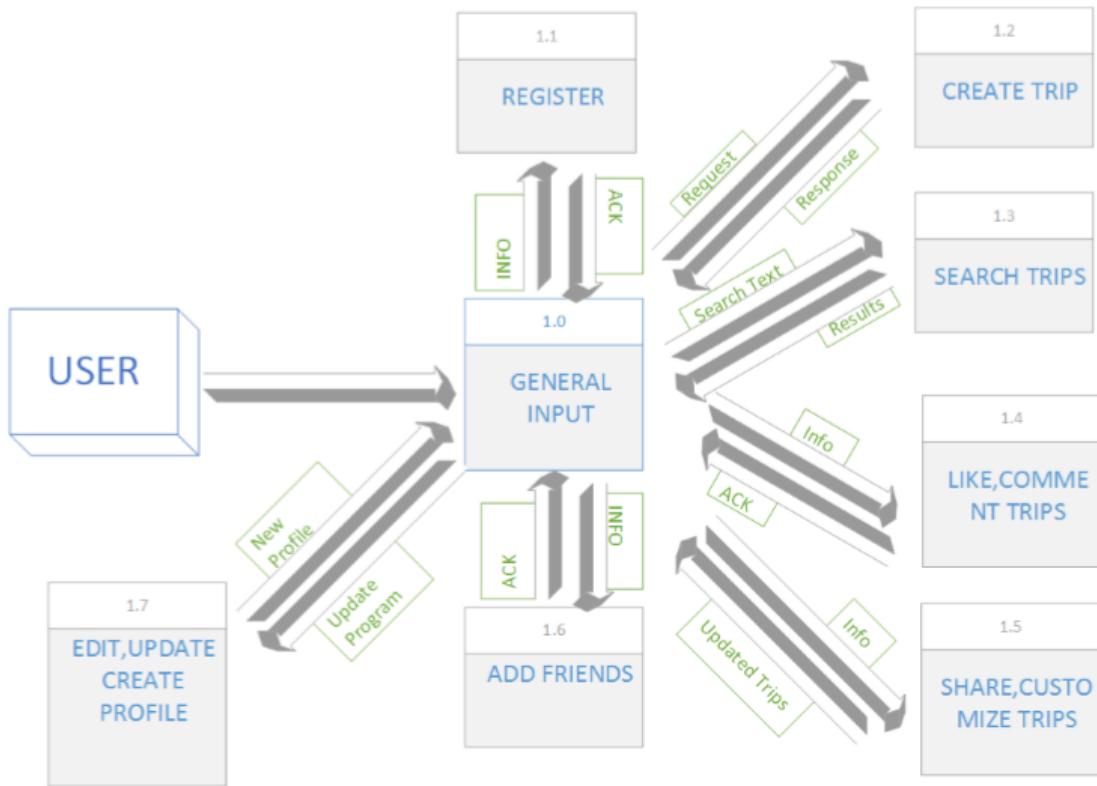


Figure 5.5 Level two user data flow diagram

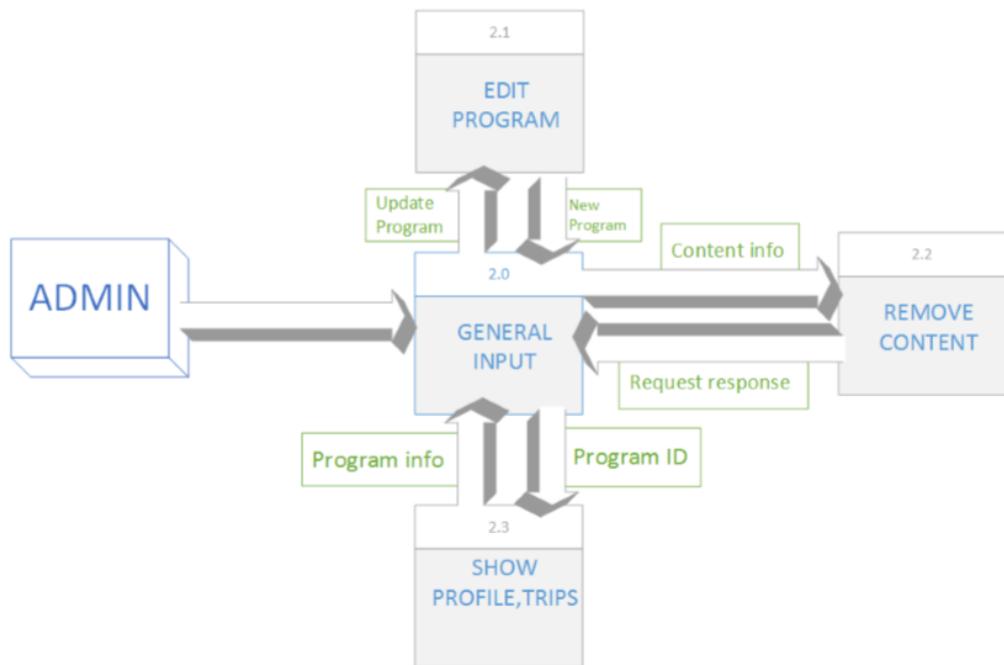


Figure 5.6 Level two administrator data flow diagram

6 Application

In this section screenshots of the system will be shown. Each screenshot represents different module of system.

6.1 Web Application

The main page of website is shown in Figure 6.1. For now it is only for a demo homepage. Only dynamic functions are register and login for now. There are different login pages for administrators and users. The page they are going to see are different from each other based on their roles.

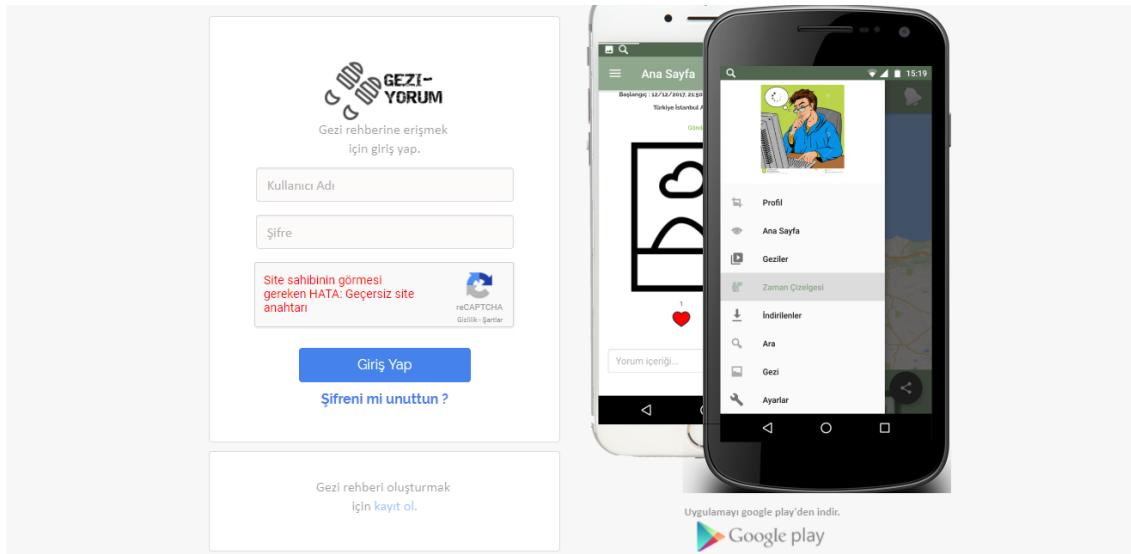


Figure 6.1 Main page of the website

Users can register from register page by typing their personal information. It is shown in figure 6.4.

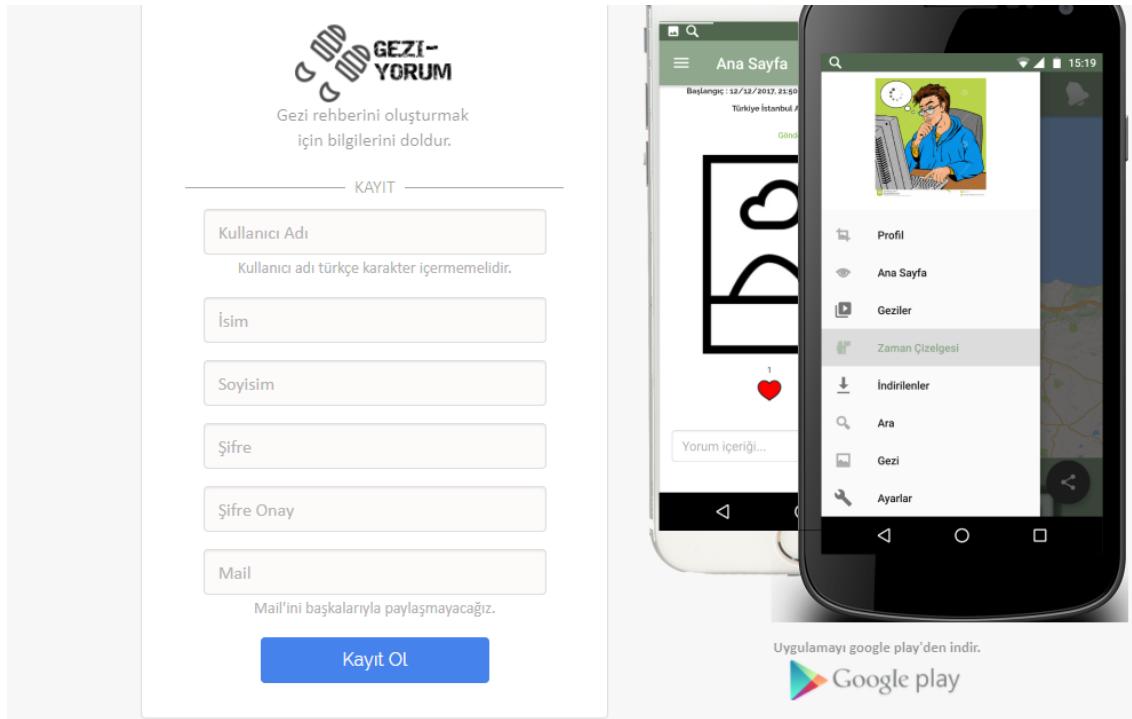


Figure 6.2 Register screen for users

Users can validate their account by redirecting the address that has been sent to them via our SMTP. It is shown in figure 6.4.

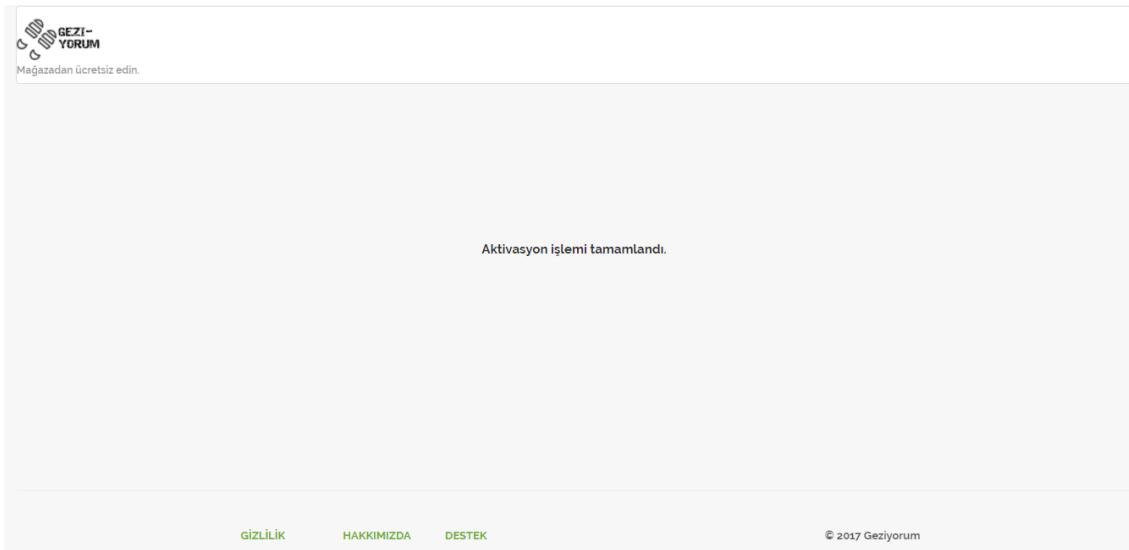


Figure 6.3 Activation screen for users

Users can change their account's password by clicking on the forgot password section and filling the username area in the main page, they can change their password by redirecting the address that has been sent to them via our SMTP. It is shown in figure 6.4.

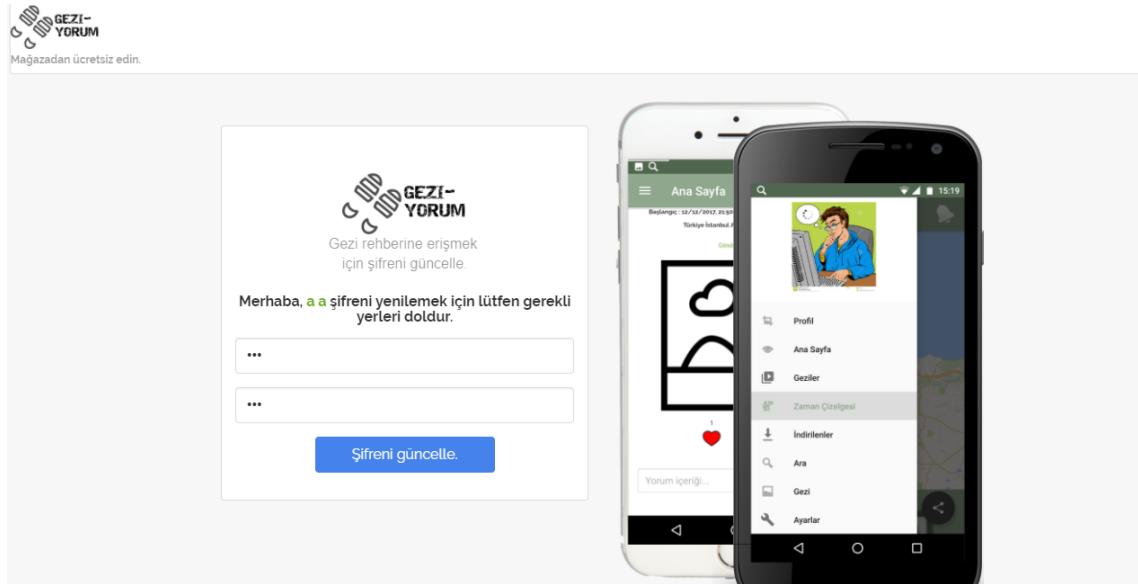


Figure 6.4 Forgot my password screen for users

Users can access to the developers by using support page, and can read the privacy principles for the permission they are going to give site owners about their usage. It is shown in figure 6.5.



Figure 6.5 Support screen for users

After being logged in users are going to see the home page of website. It is shown in figure 6.6. In this page, users are able to search through the site by typing what they looking for which can be not only users but also trips. It will redirect users to detailed search page afterwards. They can also navigate by using header section of the web page to the their personal profile page and notifications page and detailed search page.

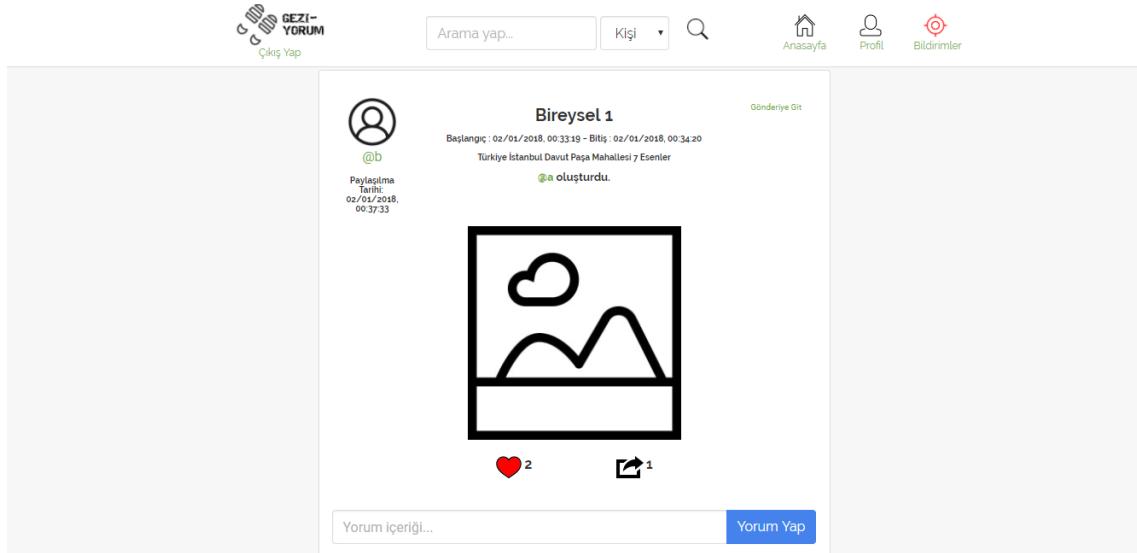


Figure 6.6 Home screen for users

Every page's screen can also turn into dynamic responsive web page no matter what kind of device users use. For example home pages responsive screen is shown in Figure 6.7.



Figure 6.7 Responsive home screen for users

Web page has a notification page which contains users interaction informations. When an user comments on other users trip the other user will be informed by using this screen he or she can hide as he or she prefer, also friend requests can be seen from this screen. Users can accept or deny these requests. The screen is shown in Figure 6.8.

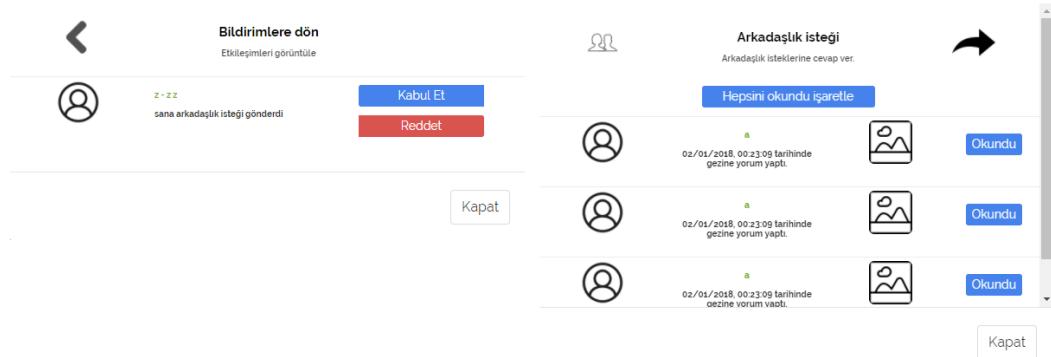


Figure 6.8 Notification screen for users

Users are able to see their personal screen through the profile page. It contains their personal trips list and the ones they shared. Also they can see their friends list they can also unfriend them if they want. The screen is shown in Figure 6.9.

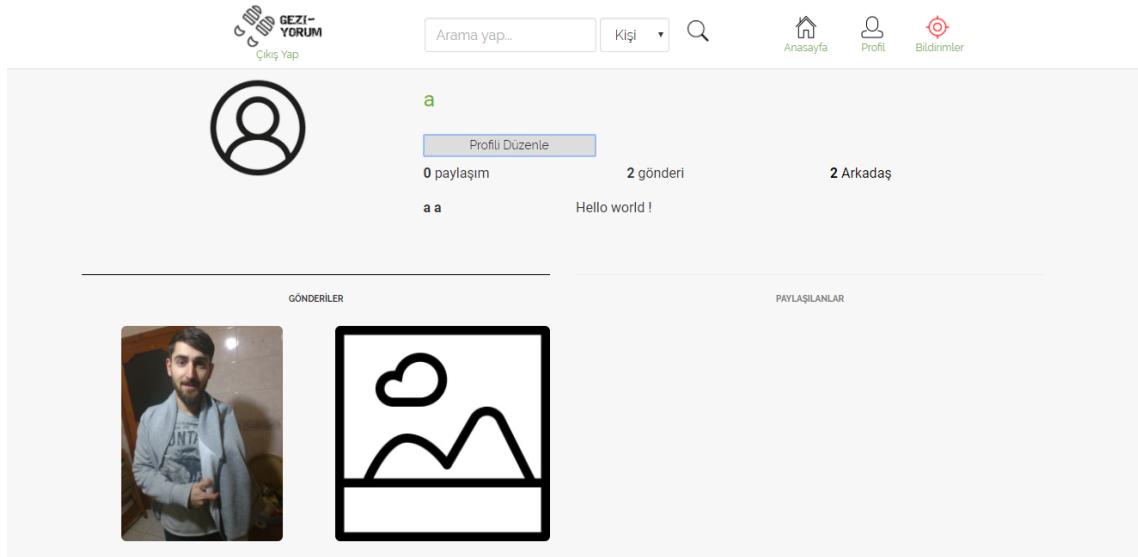


Figure 6.9 Profile page for users

Users can see their personal informations such as name, surname, e-mail, website and so on and they can change them if they want. The screen is shown in Figure 6.10.

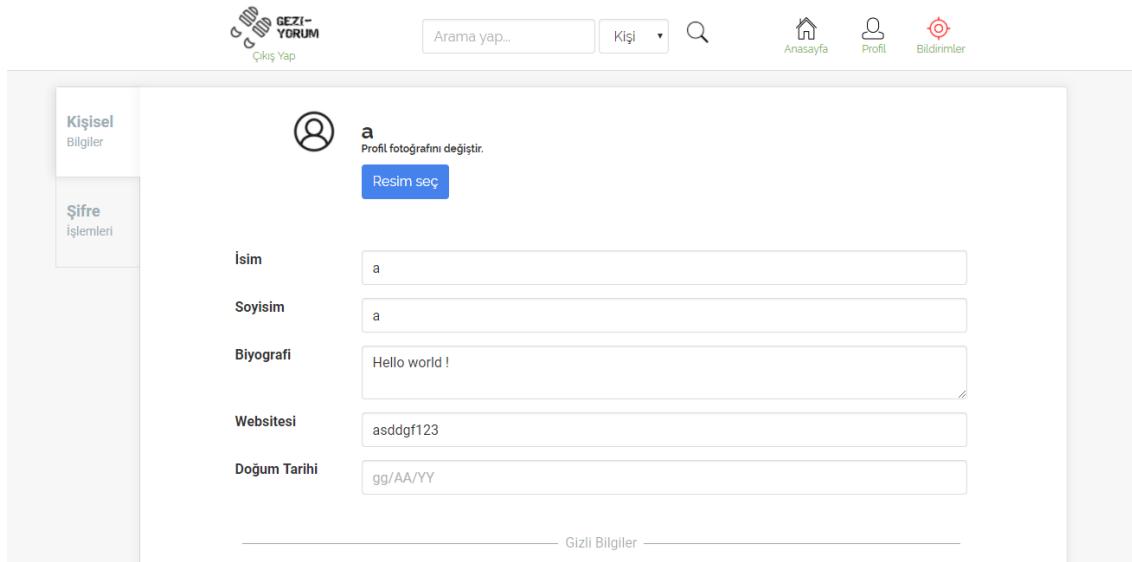


Figure 6.10 Profile page for users

Users can change their password and upload profile picture from this page. The page is shown in Figure 6.11.

The screenshot shows the user profile page of a website. At the top, there is a header with the logo 'GEZİ-YORUM' and a search bar. Below the header, there are navigation links for 'Anasayfa', 'Profil', and 'Bildirimler'. On the left side, there is a sidebar with two tabs: 'KİŞİSEL BİLGİLER' (selected) and 'ŞİFRE İŞLEMLERİ'. The main content area displays a placeholder for a profile picture with the letter 'a' and a note: 'Profil fotoğrafını değiştir.' Below this, there are three input fields for password changes: 'Eski Şifre', 'Yeni Şifre', and 'Yeni Şifre Tekrar'. At the bottom right of the form are two buttons: 'Kaydet' (Save) and 'Vazgeç' (Cancel).

Figure 6.11 Profile page for users

Other users page can be seen through the users page they can send friend requests by using this page, also if they think the owner of the page have some inappropriate content they can create a complaint. The page is shown in Figure 6.12.

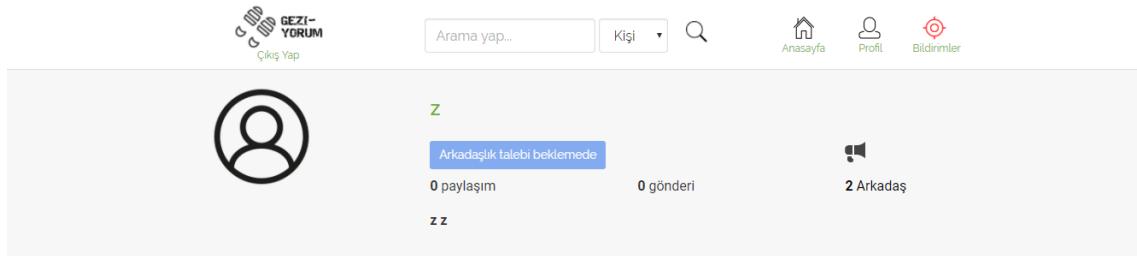


Figure 6.12 User page

Users can create complaints about profiles, shared trips or comments if they think the contents of them contains some harmful information, harassments or insulting. The page is shown in Figure 6.13.

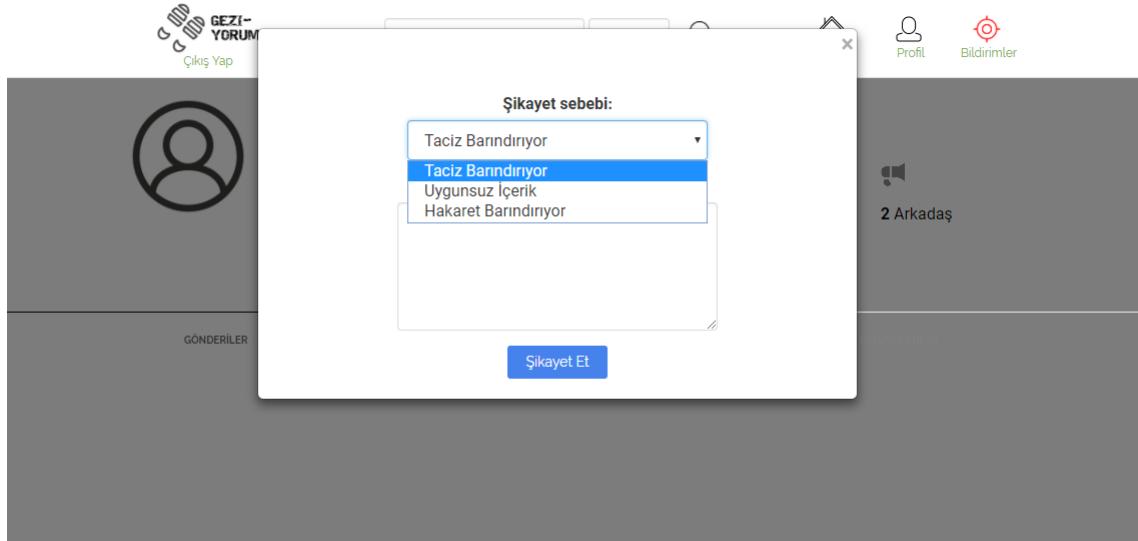


Figure 6.13 Creating complaint screen for users, trips and comments

Trips details can be seen by clicking on the trips cover photos. Its detailed screen contains information about where it is happened, persons who participated, the owner of it, what time it is created, the medias and the comments of medias, comments which the other users writed. It also contains like and share buttons, and writing comment section, lastly complaint creating buttons for both the trip itself and the comments.

6.14.

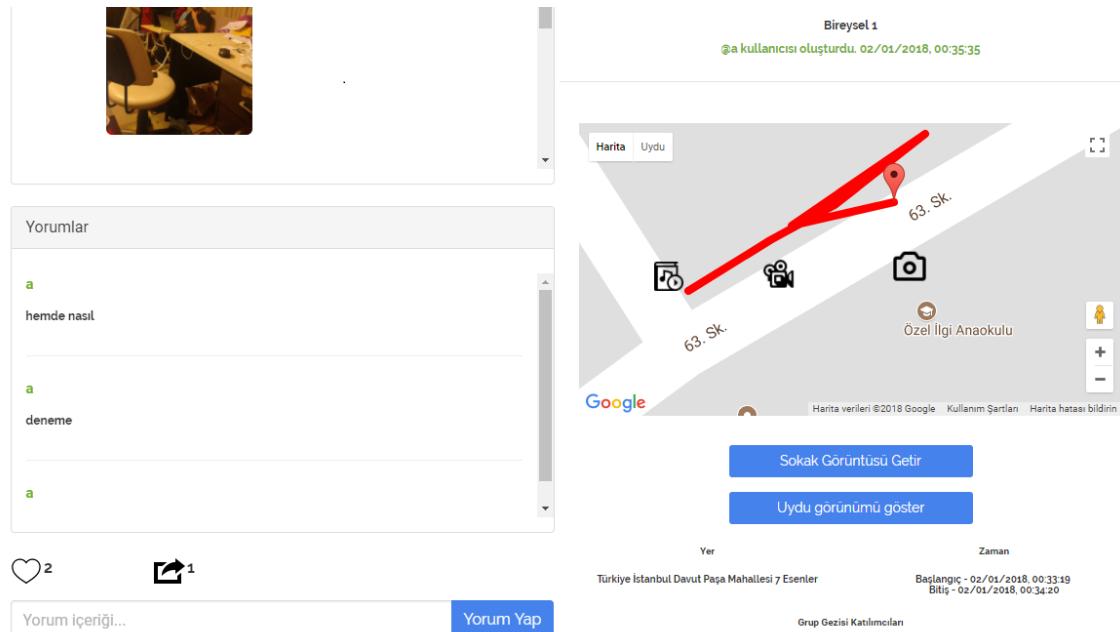


Figure 6.14 Trip details screen

Users can search trips, users from the search page they can also filter the searching by selecting what type of trip they looking for. They can also choose if they want to see trips created only by their friends or by any one on the site. The page is shown in Figure 6.15.

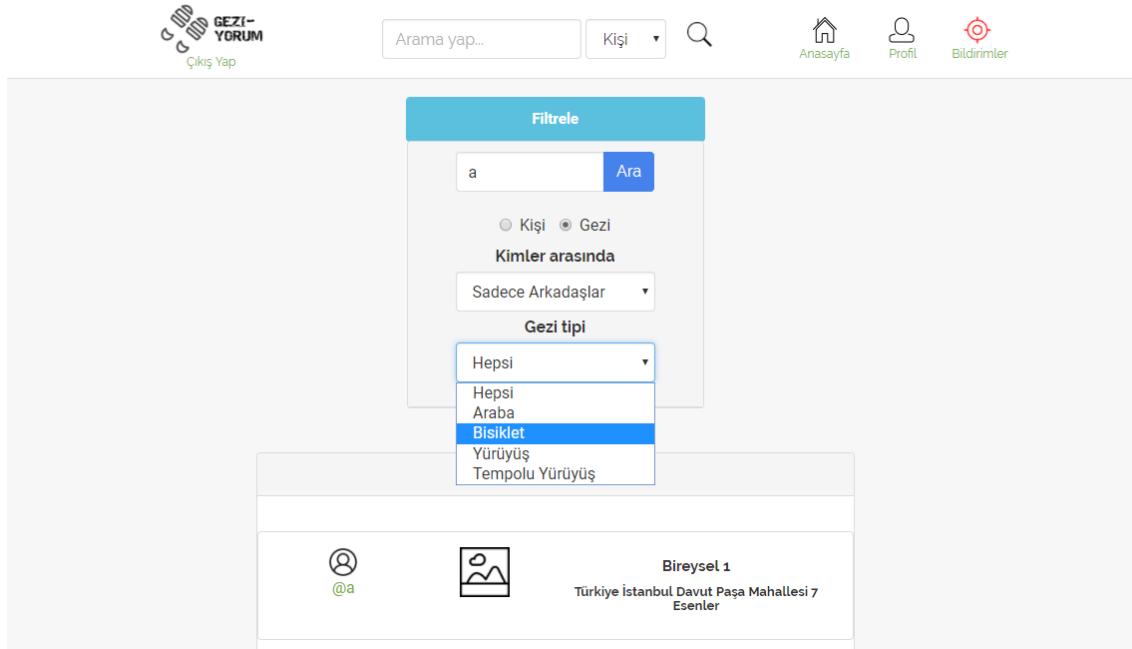


Figure 6.15 Search page

We have administrators management page. In this page administrators can see complaints related with users, trips and comments that are created by users. They can see, search, evaluate and list them with this screen which is shown in Figure 6.16.

The screenshot shows a web-based administrator interface. At the top left is a logo with the text 'GEZİ-YÖRÜM' and 'Yönetici Panelli'. At the top right are welcome messages 'Merhaba yönetici, admin' and a 'Çıkış Yap' (Logout) button. Below the header is a navigation bar with four tabs: 'Yorum Şikayetleri', 'Gönderi İçerik Şikayetleri', 'Profil Şikayetleri' (which is currently selected), and 'Destek Mesajları'. A search section contains a magnifying glass icon and a text input field 'İçerik Ara...'. Below it is a dropdown menu 'Arama Sütunu Seç ▾'. The main content area features a table with columns: ID, Şikayetçi, Şikayet Edilen, Şikayet Nedeni, Şikayet Metini, Şikayet Zamanı, and Değerlendirilme. The table has one row visible.

ID	Şikayetçi	Şikayet Edilen	Şikayet Nedeni	Şikayet Metni	Şikayet Zamanı	Değerlendirilme

Figure 6.16 Administrator page

Administrators can decide whether the content is going to be hidden or not, we have a screen to decide hide the content or not, which is shown in Figure 6.17.

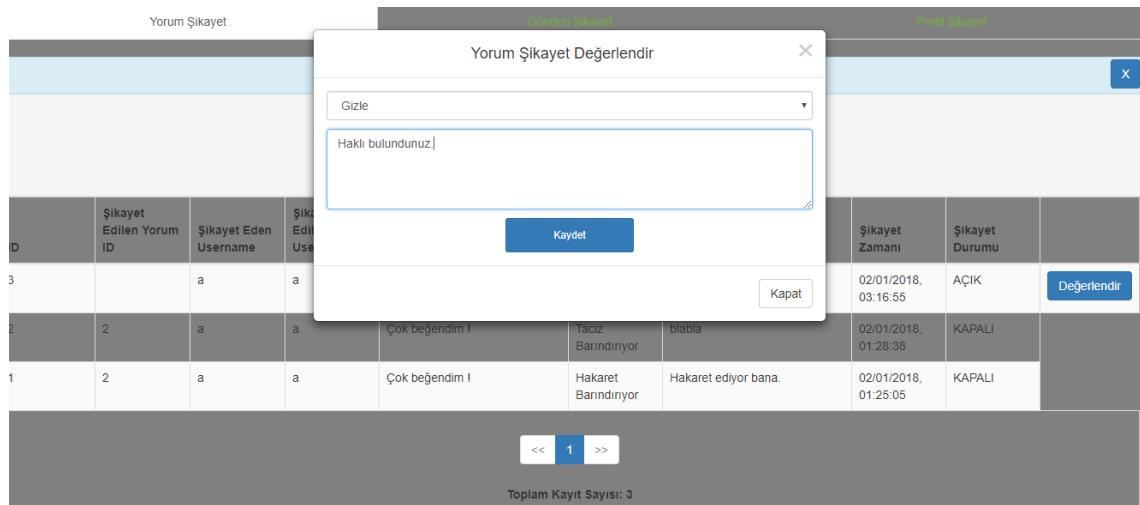


Figure 6.17 Administrator page

6.2 Mobile Application

Our application has a mobile side also. Users can login through mobile application to create, share and look for trips. It also contains the web application's every functionalities. Details of mobile application can be seen figures below.

The mobile application has register screen for users, users can register by using this screen which shown in Figure 6.18.

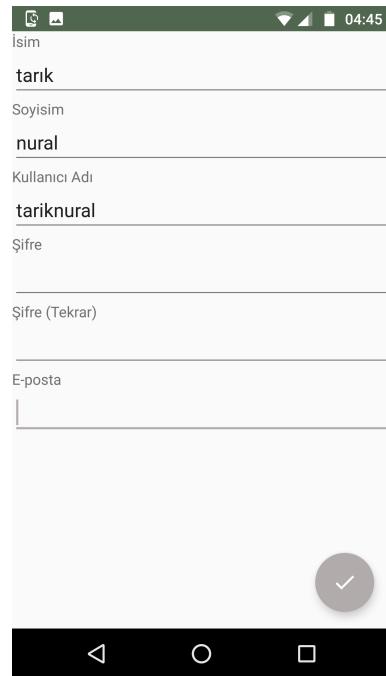


Figure 6.18 Mobile application register page

The mobile application has log-in screen for users, without logged in users are not able to use functionalities of creating trips or interact with other users, the screen is shown in Figure 6.19.

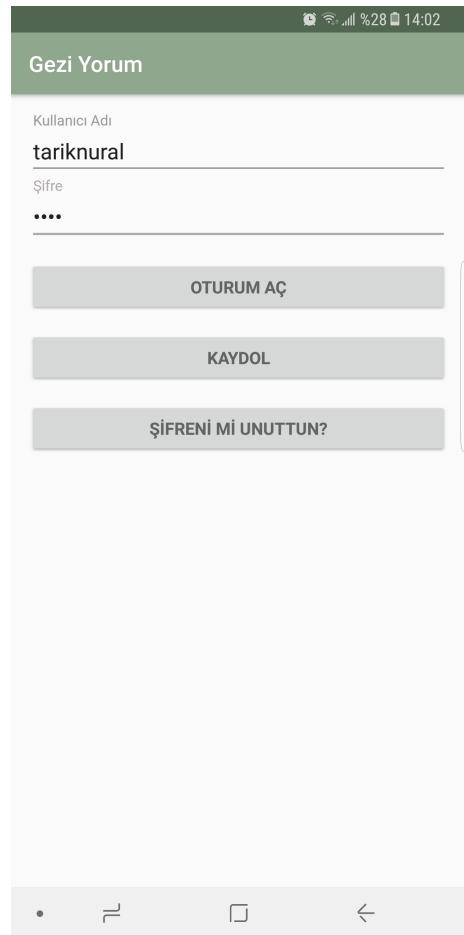


Figure 6.19 Mobile application login page

Mobile application has a main section, this page is similar to the web application home page. In this section, users are able to see their friends shared or created trips listed in here. 6.20.



Figure 6.20 Mobile application home page

Mobile application has a notification section which includes friend requests and trip invitation requests. Users can accept or deny those requests in here. 6.21.

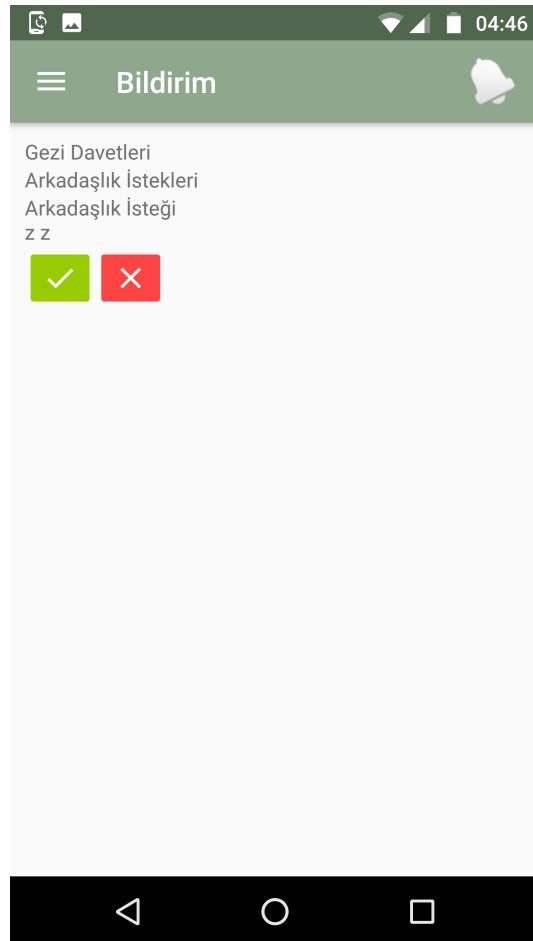


Figure 6.21 Mobile application notification page

Mobile application has a profile section, users are able to see their personal screen through the profile page. It contains their personal trips list and the ones they shared. Also they can see their friends list they can also unfriend them if they want. It has same functionalities as web application's profile page. The screen is shown in Figure 6.22.

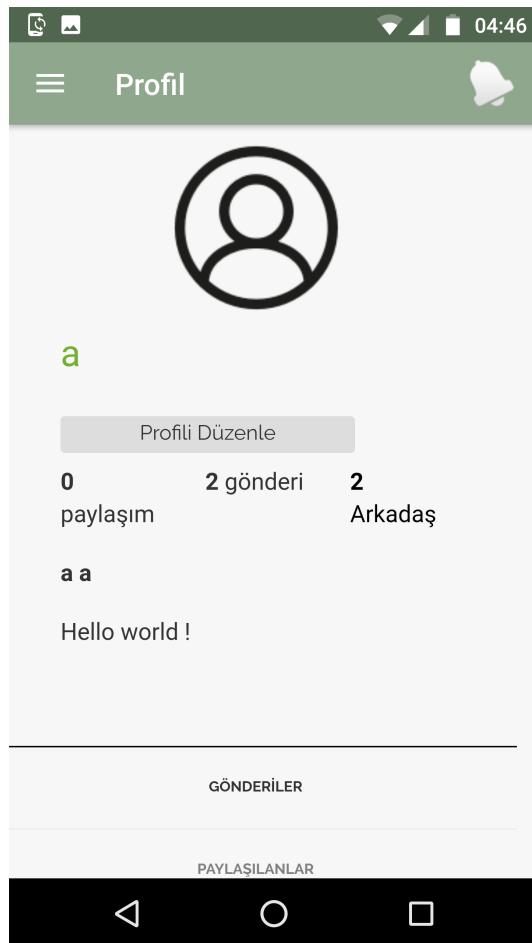


Figure 6.22 Mobile application profile page

Mobile application has trip search module it is same as web application. 6.23.

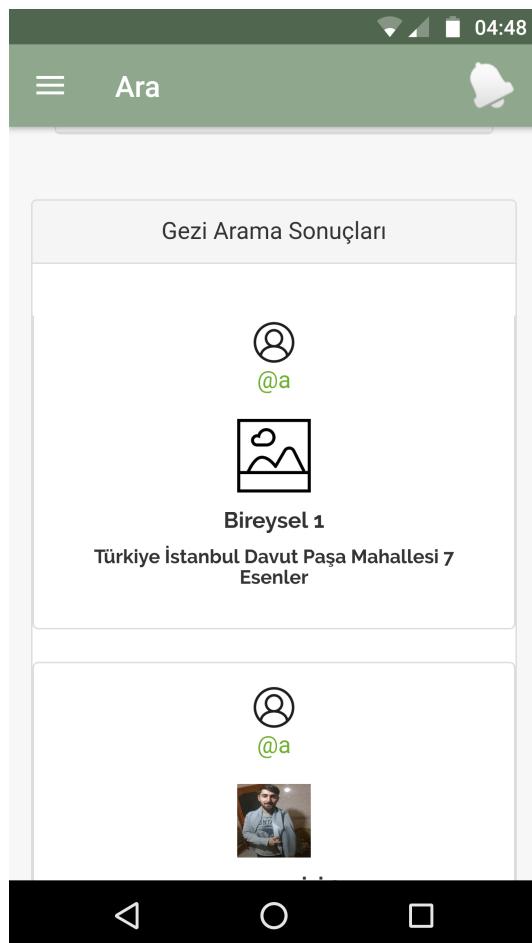


Figure 6.23 Mobile application search page

In mobile application users are able to download trips either from the ones which is shared by their friends or the ones they find in search section. When user click on download text trip will be downloaded to the local storage of mobile application, after downloading it it will be listed in downloaded trips section users can follow the downloaded trip path whenever they want and create a new one by using it. The downloading event is shown in Figure 6.24.

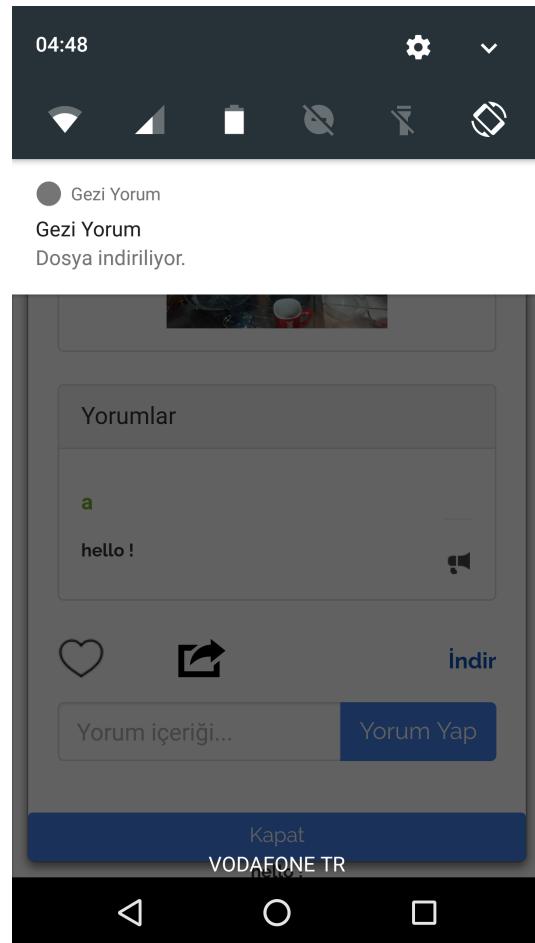


Figure 6.24 Mobile application trip download module

Downloaded trips are shown in the downloaded section, users can see the content of downloaded trip in here. The section is shown in Figure 6.25.

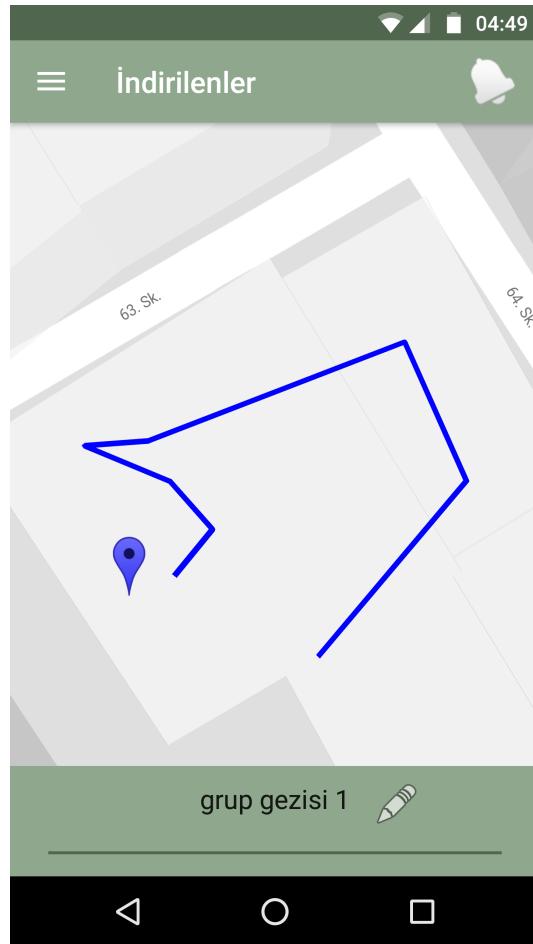


Figure 6.25 Mobile application downloaded trip section

Users can start a new trip by using downloaded trip which can be a single or group one. If they prefer to create a group trip by that they can create a trip request by adding their friends to participant list, this section is shown in Figure 6.26.

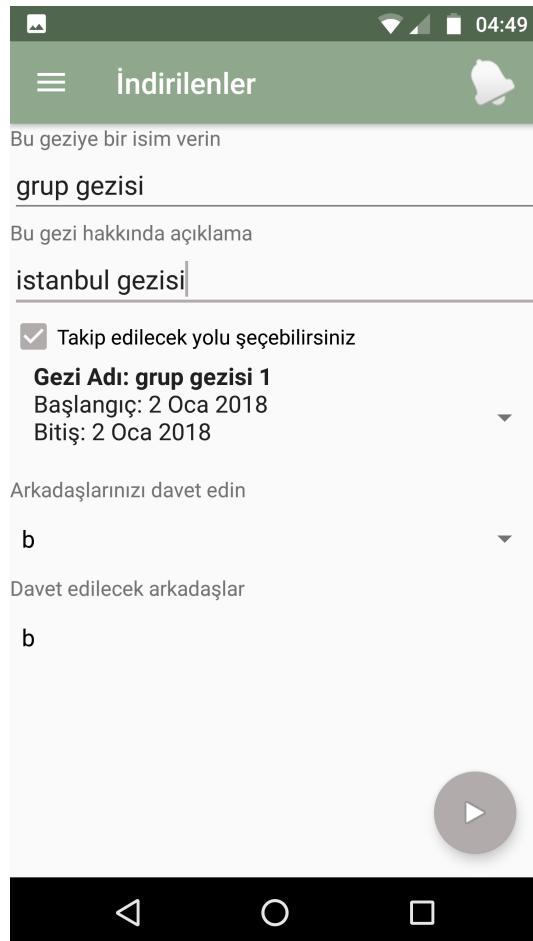


Figure 6.26 Mobile application downloaded trip section

Users can create a trip by using mobile application, poly lines which is the curves on map are generated with tracking the GPS signals of user by the application. If the user following a downloaded trip, downloaded trip's poly line will be different coloured than users own poly line. User can add medias by clicking on little plus sign, then a pop-up is shown. Those pop-ups includes buttons to record audio, to capture picture, or save video footage depending upon the type of the media which user want to save. User can also stops and starts the trip whenever her or she prefers. The section is shown in Figure 6.27.

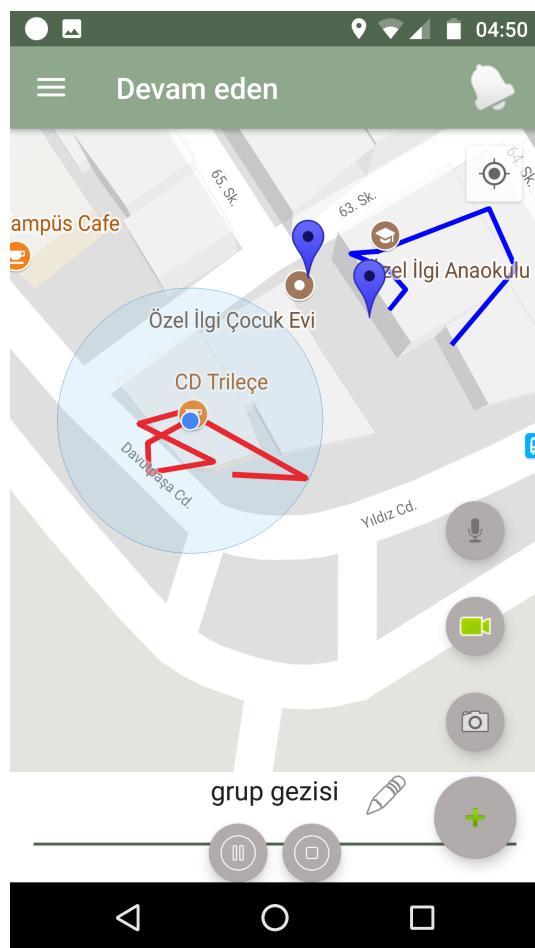


Figure 6.27 Mobile application trip creating module

By using trip creation module user may want to save some medias, if so those are listed in the created medias section, user evaluate them show them or even delete them if he or she didn't like saved medias. This section is shown in 6.28.

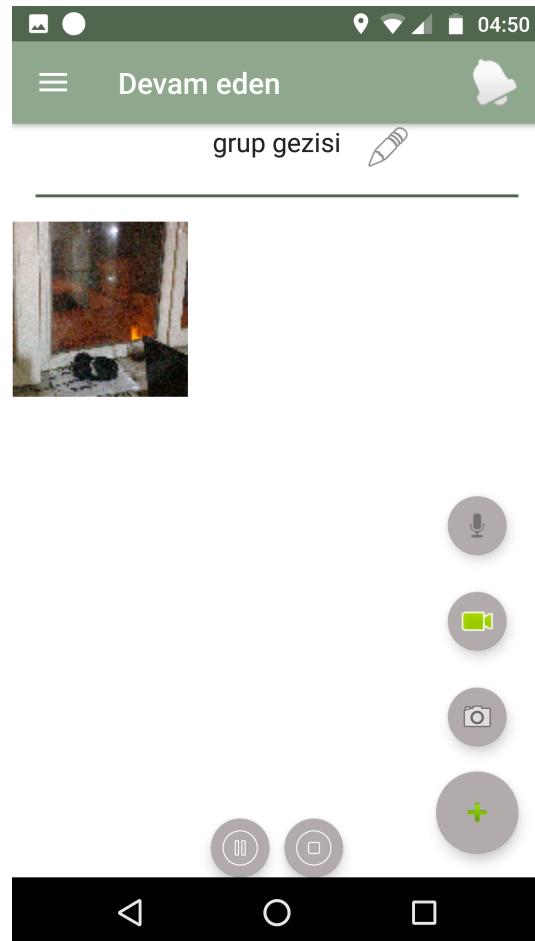


Figure 6.28 Mobile application trip managing medias while creating trip module

If user finishes trips it will be moved to the timeline section to be evaluated whether it is going to be shared or not. This section is like a review section if they don't want to save the trip after reviewing content users may want to delete or if they like they can share by clicking on the share button. This module is shown in Figure 6.29.

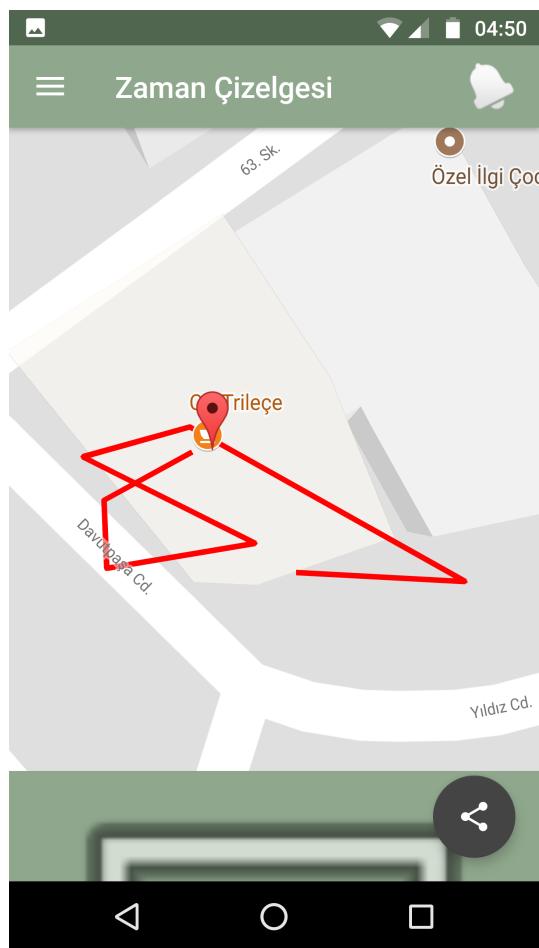


Figure 6.29 Mobile application timeline module

7

Experimental Results And Performance Analysis

7.1 Experimental Results

Web and mobile platform has been tested in our project. Different scenarios were applied with these tests and there was 0.02 percent error observed.

7.2 Performance Analysis

Performance tests were calculated with Apache JMeter tool which makes HTTP request simultaneously depending upon call number that we give. We monitored the CPU and RAM percentage at the server side by using HTOP tool. Different outputs for different number of calls is shown below. HTTP calls loop count is 10 and time interval between loop counts is 10 seconds.

One of the example HTTP call is shown in Figure 7.1.

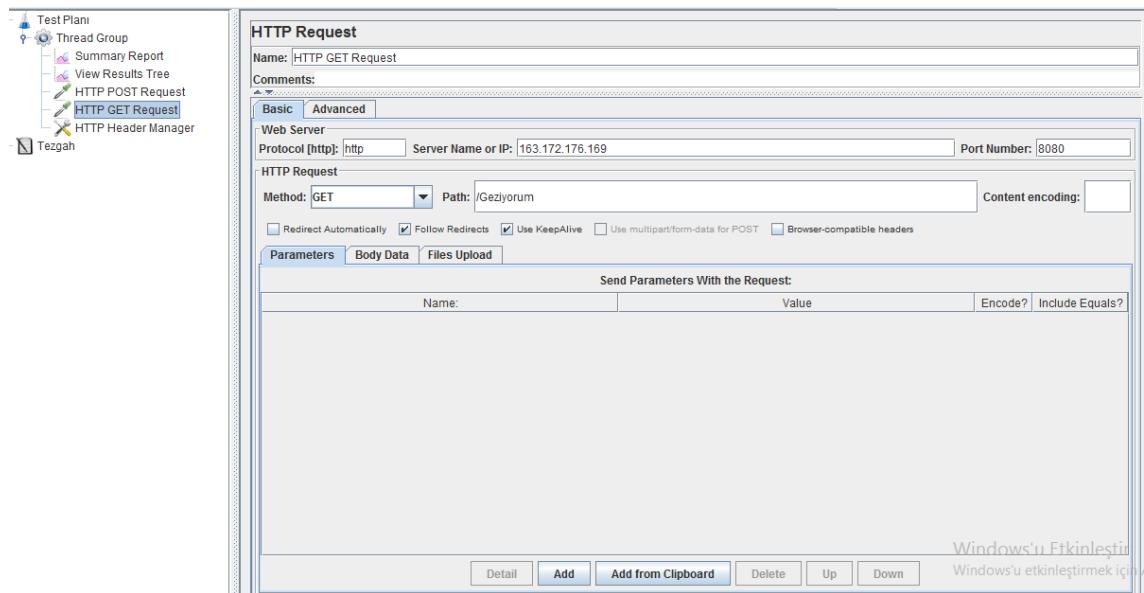


Figure 7.1 Example HTTP call

Server's idle performance is shown in Figure 7.2.

1	[]	1.3%]	Tasks: 71, 143 thr; 1 running								
2	[]	3.2%]	Load average: 0.00 0.07 0.06								
Mem[565/2001MB]	Uptime: 07:28:04								
Swp[0/0MB]									
PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command											
7923	root	20	0	24804	3824	2936	R	1.9	0.2	1:22.90	htop
4827	root	20	0	976M	110M	65588	S	0.6	5.5	2:08.31	/usr/lib/mysql-work
8359	root	20	0	2949M	305M	17836	S	0.6	15.3	0:01.01	/usr/lib/jvm/java-8
4902	root	20	0	2949M	305M	17836	S	0.0	15.3	8:02.55	/usr/lib/jvm/java-8
4913	root	20	0	2949M	305M	17836	S	0.0	15.3	0:24.37	/usr/lib/jvm/java-8
3255	syslog	20	0	249M	4276	2280	S	0.0	0.2	0:00.75	rsyslogd
3865	mysql	20	0	1251M	56288	10888	S	0.0	2.7	0:01.34	/usr/sbin/mysqld
4939	root	20	0	2949M	305M	17836	S	0.0	15.3	0:17.15	/usr/lib/jvm/java-8
8362	root	20	0	2949M	305M	17836	S	0.0	15.3	0:00.81	/usr/lib/jvm/java-8
4480	root	20	0	125M	28844	11000	S	0.0	1.4	0:33.60	Xtightvnc :1 -desk
3799	mysql	20	0	1251M	56288	10888	S	0.0	2.7	0:35.78	/usr/sbin/mysqld
7649	root	20	0	103M	6588	5612	S	0.0	0.3	0:00.64	sshd: root@pts/0
4978	mysql	20	0	1251M	56288	10888	S	0.0	2.7	0:07.08	/usr/sbin/mysqld
8358	root	20	0	2949M	305M	17836	S	0.0	15.3	0:00.97	/usr/lib/jvm/java-8
4968	root	20	0	2949M	305M	17836	S	0.0	15.3	0:00.44	/usr/lib/jvm/java-8
3920	mysql	20	0	1251M	56288	10888	S	0.0	2.7	0:03.35	/usr/sbin/mysqld
4944	root	20	0	2949M	305M	17836	S	0.0	15.3	0:17.66	/usr/lib/jvm/java-8
4916	root	20	0	2949M	305M	17836	S	0.0	15.3	0:11.41	/usr/lib/jvm/java-8
4914	root	20	0	2949M	305M	17836	S	0.0	15.3	0:01.59	/usr/lib/jvm/java-8
4963	root	20	0	2949M	305M	17836	S	0.0	15.3	0:01.33	/usr/lib/jvm/java-8
4555	root	20	0	456M	11988	10360	S	0.0	0.6	0:00.43	zeitgeist-databub
4941	root	20	0	2949M	305M	17836	S	0.0	15.3	0:17.03	/usr/lib/jvm/java-8
4969	root	20	0	2949M	305M	17836	S	0.0	15.3	0:01.76	/usr/lib/jvm/java-8
4948	root	20	0	2949M	305M	17836	S	0.0	15.3	0:06.73	/usr/lib/jvm/java-8
8357	root	20	0	2949M	305M	17836	S	0.0	15.3	0:01.41	/usr/lib/jvm/java-8
4937	root	20	0	2949M	305M	17836	S	0.0	15.3	0:06.08	/usr/lib/jvm/java-8
4942	root	20	0	2949M	305M	17836	S	0.0	15.3	0:17.96	/usr/lib/jvm/java-8
3912	root	20	0	61400	5456	4776	S	0.0	0.3	0:01.58	/usr/sbin/sshd -D
4940	root	20	0	2949M	305M	17836	S	0.0	15.3	0:18.80	/usr/lib/jvm/java-8
4945	root	20	0	2949M	305M	17836	S	0.0	15.3	0:16.76	/usr/lib/jvm/java-8
8367	root	20	0	2949M	305M	17836	S	0.0	15.3	0:00.28	/usr/lib/jvm/java-8
8368	root	20	0	2949M	305M	17836	S	0.0	15.3	0:00.20	/usr/lib/jvm/java-8

Figure 7.2 When server is idle position

When server's performance 30 simultaneous GET call is shown Figure 7.4.

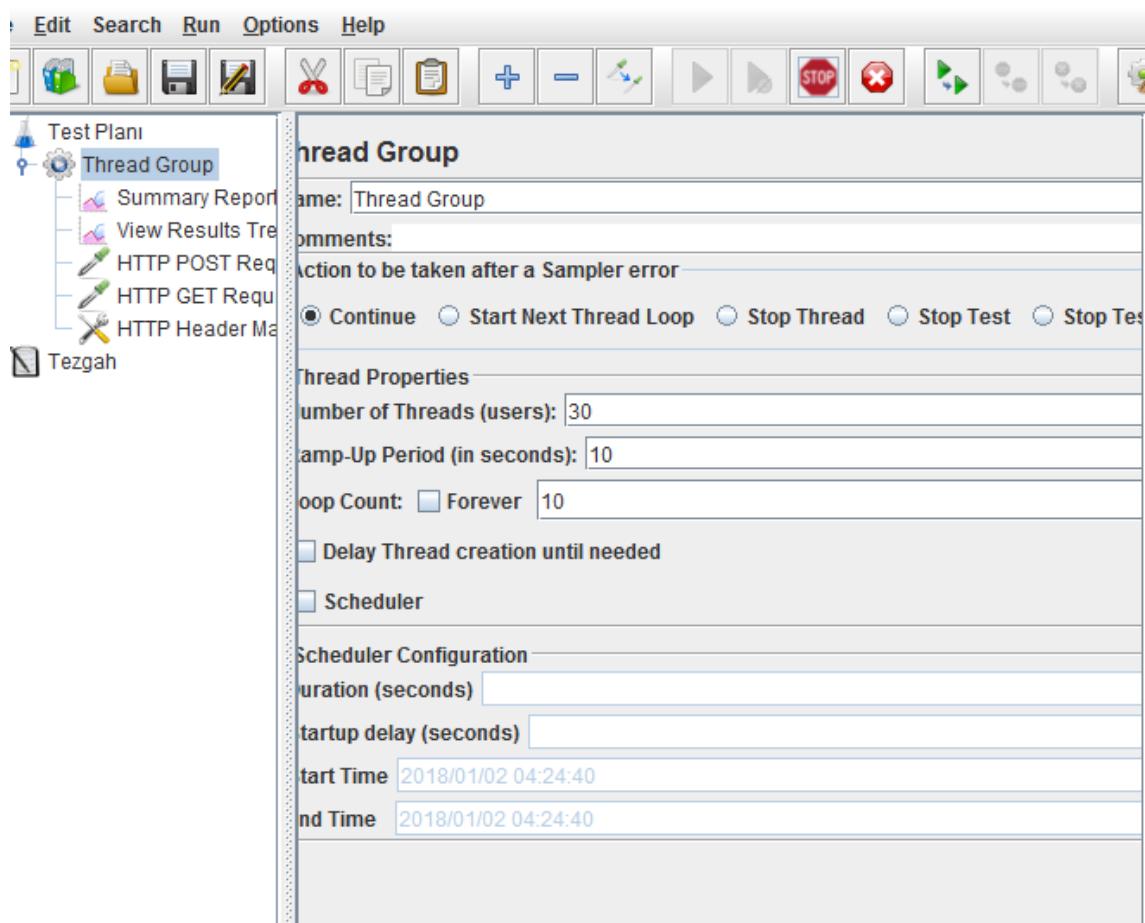


Figure 7.3 Server 30 GET HTTP calls simultaneously

```

1 [|||||||] 18.1%] Tasks: 75, 143 thr; 1 running
2 [|||||||] 15.5%] Load average: 0.38 0.19 0.11
Mem[||||||||||||| 568/2001MB] Uptime: 07:33:30
SWPL 0.0MB]

PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command
4902 root 20 0 2949M 305M 17836 S 18.1 15.2 8:08.46 /usr/lib/jvm/java-8
7923 root 20 0 24804 3824 2936 R 2.6 0.2 1:30.85 htop
4949 root 20 0 2949M 305M 17836 S 1.3 15.2 0:07.19 /usr/lib/jvm/java-8
8372 root 20 0 2949M 305M 17836 S 1.3 15.2 0:00.40 /usr/lib/jvm/java-8
4945 root 20 0 2949M 305M 17836 S 1.3 15.2 0:16.98 /usr/lib/jvm/java-8
4944 root 20 0 2949M 305M 17836 S 1.3 15.2 0:17.85 /usr/lib/jvm/java-8
4947 root 20 0 2949M 305M 17836 S 1.3 15.2 0:16.92 /usr/lib/jvm/java-8
4938 root 20 0 2949M 305M 17836 S 1.3 15.2 0:17.92 /usr/lib/jvm/java-8
8373 root 20 0 2949M 305M 17836 S 1.3 15.2 0:00.39 /usr/lib/jvm/java-8
4916 root 20 0 2949M 305M 17836 S 0.6 15.2 0:11.66 /usr/lib/jvm/java-8
3799 mysql 20 0 1251M 56288 10888 S 0.6 2.7 0:36.27 /usr/sbin/mysqld
8358 root 20 0 2949M 305M 17836 S 0.6 15.2 0:01.18 /usr/lib/jvm/java-8
4942 root 20 0 2949M 305M 17836 S 0.6 15.2 0:18.17 /usr/lib/jvm/java-8
8367 root 20 0 2949M 305M 17836 S 0.6 15.2 0:00.46 /usr/lib/jvm/java-8
8369 root 20 0 2949M 305M 17836 S 0.6 15.2 0:00.41 /usr/lib/jvm/java-8
4940 root 20 0 2949M 305M 17836 S 0.6 15.2 0:19.01 /usr/lib/jvm/java-8
8356 root 20 0 2949M 305M 17836 S 0.6 15.2 0:01.74 /usr/lib/jvm/java-8
4943 root 20 0 2949M 305M 17836 S 0.6 15.2 0:16.94 /usr/lib/jvm/java-8
8368 root 20 0 2949M 305M 17836 S 0.6 15.2 0:00.38 /usr/lib/jvm/java-8
4948 root 20 0 2949M 305M 17836 S 0.6 15.2 0:06.87 /usr/lib/jvm/java-8
4939 root 20 0 2949M 305M 17836 S 0.6 15.2 0:17.35 /usr/lib/jvm/java-8
8370 root 20 0 2949M 305M 17836 S 0.6 15.2 0:00.34 /usr/lib/jvm/java-8
8371 root 20 0 2949M 305M 17836 S 0.6 15.2 0:00.38 /usr/lib/jvm/java-8
8359 root 20 0 2949M 305M 17836 S 0.6 15.2 0:01.19 /usr/lib/jvm/java-8
8362 root 20 0 2949M 305M 17836 S 0.6 15.2 0:00.99 /usr/lib/jvm/java-8
4973 mysql 20 0 1251M 56288 10888 S 0.6 2.7 0:01.92 /usr/sbin/mysqld
4941 root 20 0 2949M 305M 17836 S 0.6 15.2 0:17.21 /usr/lib/jvm/java-8

```

Figure 7.4 Server 30 GET HTTP calls simultaneously

When server's performance 100 simultaneous GET call is shown Figure 7.6.

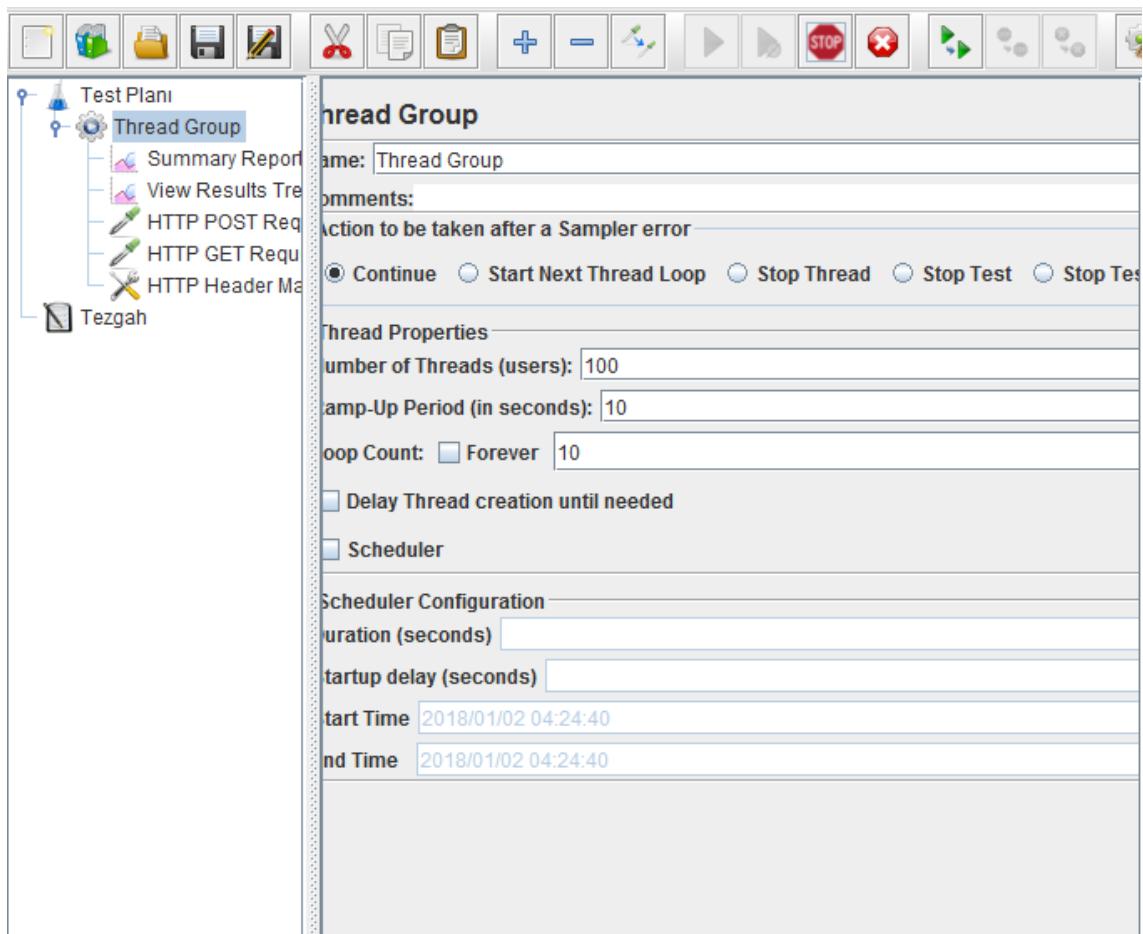


Figure 7.5 Server 100 GET HTTP calls simultaneously

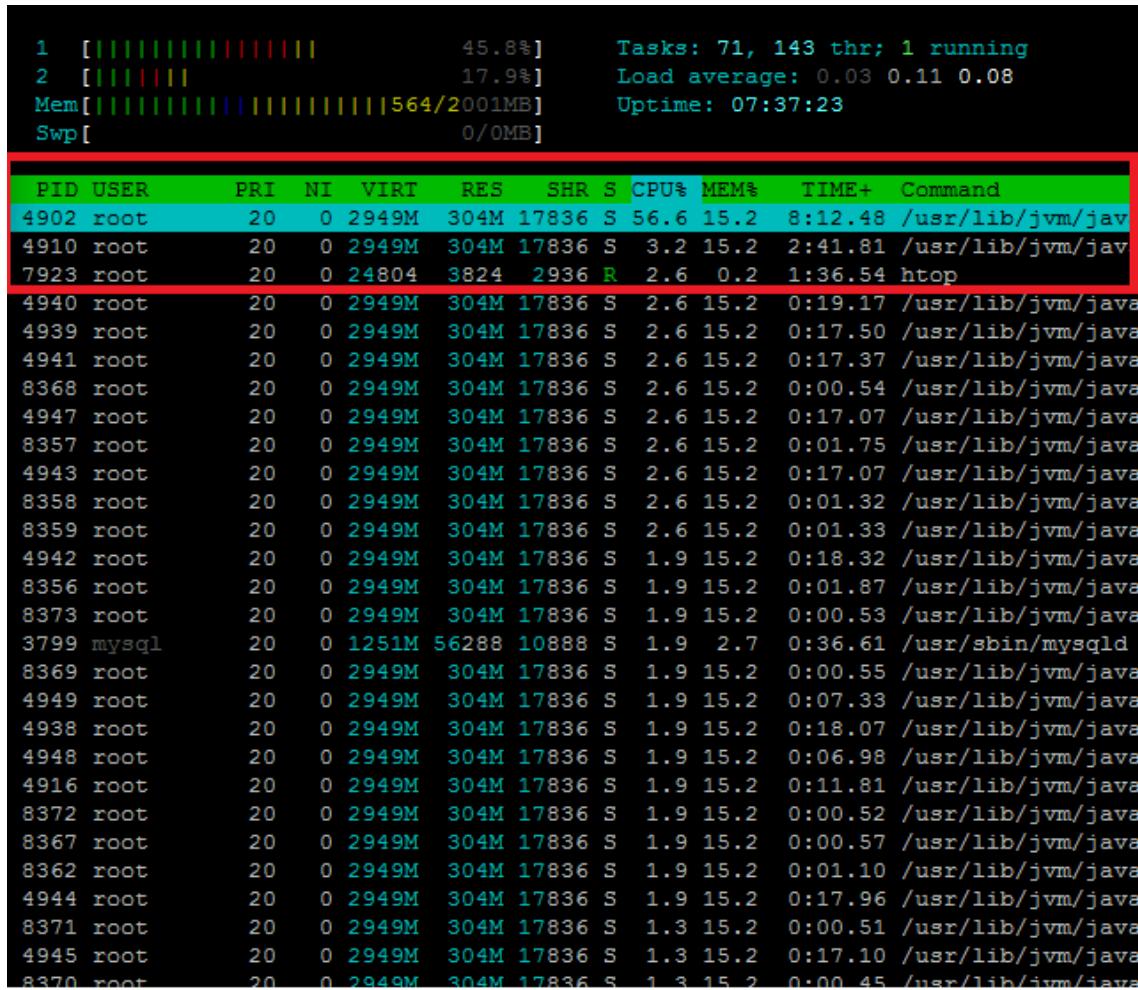


Figure 7.6 Server 100 GET HTTP calls simultaneously

When server's performance 130 simultaneous GET and POST call is shown Figure 7.8.

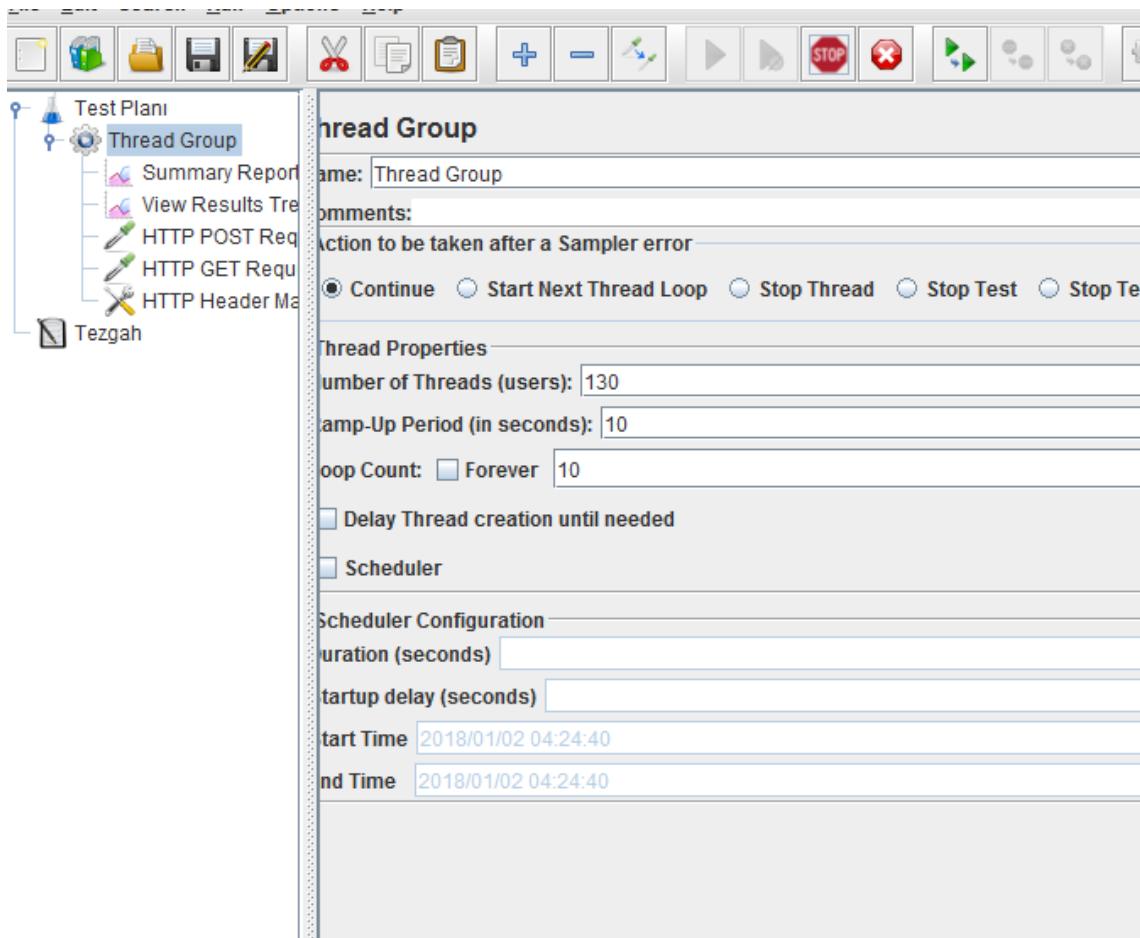


Figure 7.7 Server 130 GET and POST HTTP calls simultaneously

1	[]	38.2%]	Tasks: 71, 143 thr; 5 running								
2	[]	52.2%]	Load average: 0.08 0.11 0.09								
Mem[561/2001MB]	Uptime: 07:38:58								
Swp[0/0MB]									
<hr/>											
FID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
4902	root	20	0	2949M	301M	17836	S	68.8	15.0	8:21.50	/usr/lib/jvm/java
3799	mysql	20	0	1251M	56288	10888	S	3.8	2.7	0:37.10	/usr/sbin/mysqld
4948	root	20	0	2949M	301M	17836	S	3.8	15.0	0:07.39	/usr/lib/jvm/java
8356	root	20	0	2949M	301M	17836	S	3.8	15.0	0:02.28	/usr/lib/jvm/java
4939	root	20	0	2949M	301M	17836	S	3.8	15.0	0:17.87	/usr/lib/jvm/java
4942	root	20	0	2949M	301M	17836	S	3.2	15.0	0:18.71	/usr/lib/jvm/java
4947	root	20	0	2949M	301M	17836	S	3.2	15.0	0:17.44	/usr/lib/jvm/java
8369	root	20	0	2949M	301M	17836	S	3.2	15.0	0:00.92	/usr/lib/jvm/java
8372	root	20	0	2949M	301M	17836	S	3.2	15.0	0:00.88	/usr/lib/jvm/java
8367	root	20	0	2949M	301M	17836	S	3.2	15.0	0:00.97	/usr/lib/jvm/java
4940	root	20	0	2949M	301M	17836	S	3.2	15.0	0:19.53	/usr/lib/jvm/java
4944	root	20	0	2949M	301M	17836	S	3.2	15.0	0:18.33	/usr/lib/jvm/java
4941	root	20	0	2949M	301M	17836	S	3.2	15.0	0:17.69	/usr/lib/jvm/java
4943	root	20	0	2949M	301M	17836	R	2.5	15.0	0:17.46	/usr/lib/jvm/java
8357	root	20	0	2949M	301M	17836	S	2.5	15.0	0:02.10	/usr/lib/jvm/java
4945	root	20	0	2949M	301M	17836	S	2.5	15.0	0:17.46	/usr/lib/jvm/java
7923	root	20	0	24804	3824	2936	R	2.5	0.2	1:38.93	htop
8371	root	20	0	2949M	301M	17836	S	2.5	15.0	0:00.85	/usr/lib/jvm/java
8370	root	20	0	2949M	301M	17836	S	2.5	15.0	0:00.80	/usr/lib/jvm/java
4916	root	20	0	2949M	301M	17836	S	2.5	15.0	0:12.16	/usr/lib/jvm/java
8358	root	20	0	2949M	301M	17836	S	2.5	15.0	0:01.69	/usr/lib/jvm/java
8368	root	20	0	2949M	301M	17836	S	2.5	15.0	0:00.88	/usr/lib/jvm/java
8362	root	20	0	2949M	301M	17836	R	2.5	15.0	0:01.42	/usr/lib/jvm/java
8359	root	20	0	2949M	301M	17836	S	2.5	15.0	0:01.66	/usr/lib/jvm/java
4938	root	20	0	2949M	301M	17836	S	1.9	15.0	0:18.42	/usr/lib/jvm/java
8373	root	20	0	2949M	301M	17836	S	1.9	15.0	0:00.90	/usr/lib/jvm/java
4974	mysql	20	0	1251M	56288	10888	S	1.9	2.7	0:00.47	/usr/sbin/mysqld
4949	root	20	0	2949M	301M	17836	R	1.3	15.0	0:07.54	/usr/lib/jvm/java

Figure 7.8 Server 130 GET and POST HTTP calls simultaneously

When server's performance 180 simultaneous GET and two POST call is shown Figure 7.10.

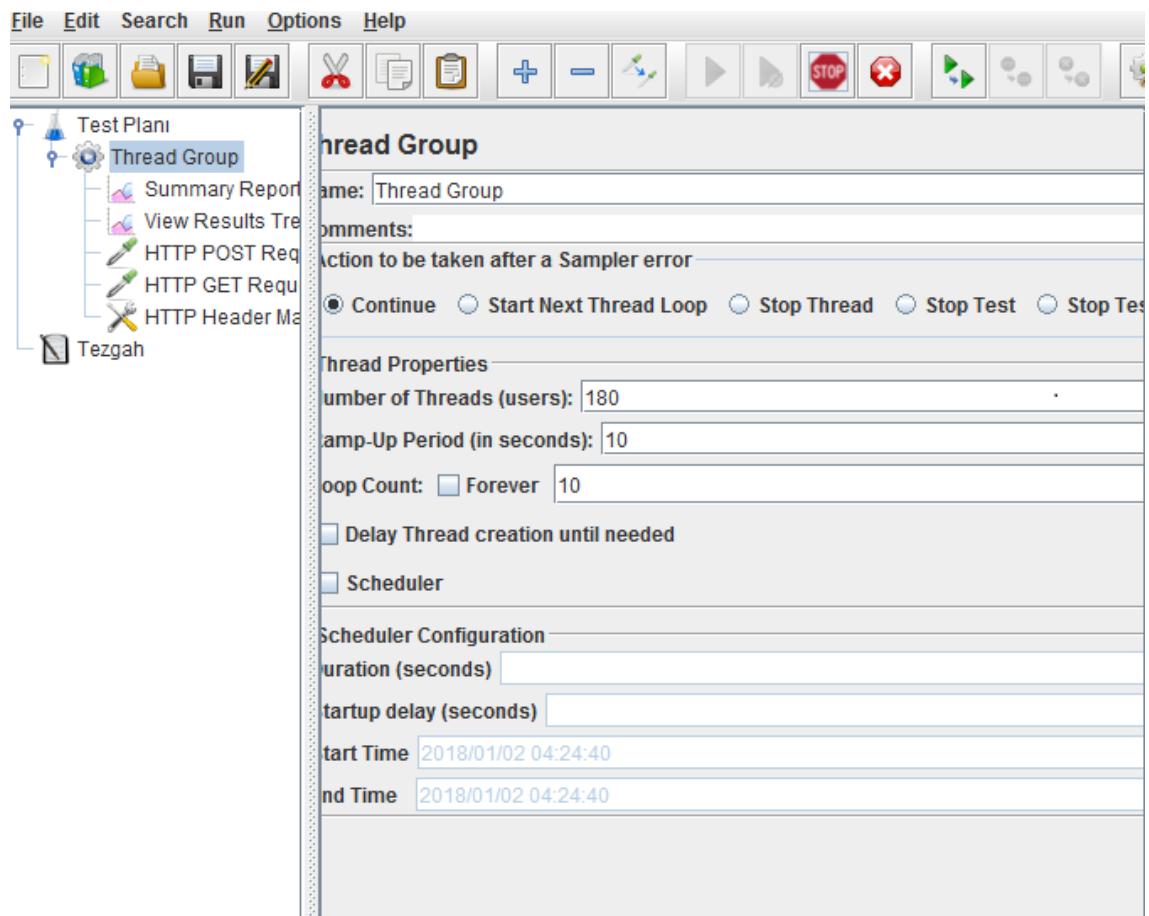


Figure 7.9 Server 180 GET and two POST HTTP calls simultaneously

1	[57.8%]	Tasks: 71, 143 thr; 3 running								
2	[50.3%]	Load average: 0.55 0.18 0.11								
Mem[561/2001MB]	Uptime: 07:41:05								
Swp[0/0MB]									
PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
4902	root	20	0	2949M	300M	17836	S	89.4	15.0	8:32.01	/usr/lib/jvm/java-
3799	mysql	20	0	1251M	56288	10888	S	4.3	2.7	0:37.70	/usr/sbin/mysqld
8358	root	20	0	2949M	300M	17836	S	4.3	15.0	0:02.10	/usr/lib/jvm/java-
4944	root	20	0	2949M	300M	17836	R	4.3	15.0	0:18.74	/usr/lib/jvm/java-
4938	root	20	0	2949M	300M	17836	S	4.3	15.0	0:18.87	/usr/lib/jvm/java-
8371	root	20	0	2949M	300M	17836	S	3.7	15.0	0:01.27	/usr/lib/jvm/java-
8372	root	20	0	2949M	300M	17836	S	3.7	15.0	0:01.32	/usr/lib/jvm/java-
8368	root	20	0	2949M	300M	17836	S	3.7	15.0	0:01.30	/usr/lib/jvm/java-
8367	root	20	0	2949M	300M	17836	S	3.7	15.0	0:01.37	/usr/lib/jvm/java-
4939	root	20	0	2949M	300M	17836	R	3.7	15.0	0:18.28	/usr/lib/jvm/java-
4943	root	20	0	2949M	300M	17836	S	3.7	15.0	0:17.87	/usr/lib/jvm/java-
4941	root	20	0	2949M	300M	17836	S	3.7	15.0	0:18.10	/usr/lib/jvm/java-
8356	root	20	0	2949M	300M	17836	S	3.7	15.0	0:02.68	/usr/lib/jvm/java-
8370	root	20	0	2949M	300M	17836	S	3.7	15.0	0:01.20	/usr/lib/jvm/java-
4945	root	20	0	2949M	300M	17836	S	3.7	15.0	0:17.87	/usr/lib/jvm/java-
8369	root	20	0	2949M	300M	17836	S	3.7	15.0	0:01.34	/usr/lib/jvm/java-
8362	root	20	0	2949M	300M	17836	S	3.7	15.0	0:01.84	/usr/lib/jvm/java-
4916	root	20	0	2949M	300M	17836	S	3.7	15.0	0:12.57	/usr/lib/jvm/java-
4949	root	20	0	2949M	300M	17836	S	3.7	15.0	0:07.89	/usr/lib/jvm/java-
8357	root	20	0	2949M	300M	17836	S	3.7	15.0	0:02.50	/usr/lib/jvm/java-
4947	root	20	0	2949M	300M	17836	S	3.1	15.0	0:17.83	/usr/lib/jvm/java-
4940	root	20	0	2949M	300M	17836	S	3.1	15.0	0:19.92	/usr/lib/jvm/java-
4942	root	20	0	2949M	300M	17836	S	3.1	15.0	0:19.10	/usr/lib/jvm/java-
4948	root	20	0	2949M	300M	17836	S	3.1	15.0	0:07.82	/usr/lib/jvm/java-
8359	root	20	0	2949M	300M	17836	S	2.5	15.0	0:02.06	/usr/lib/jvm/java-
8373	root	20	0	2949M	300M	17836	S	2.5	15.0	0:01.29	/usr/lib/jvm/java-
7923	root	20	0	24804	3824	2936	R	2.5	0.2	1:42.10	htop
4974	mysql	20	0	1251M	56288	10888	S	1.9	2.7	0:00.66	/usr/sbin/mysqld

Figure 7.10 Server 180 GET and two POST HTTP calls simultaneously

7.2.1 Result

As we seen the results above, the server can handle up to 200 simultaneously HTTP calls, its bottleneck is its CPU power, because maximum RAM usage 20 percent even the highest limit of calls. If we assume that average HTTP calls per web page is 20, (the number can vary depending upon the demand of the user) as a result, if server can handle 200 simultaneous calls and user makes 20 calls per page we can say the average number of users that our server can handle is 10.

Comparison between HTTP calls and the result outputs of error rate, throughput, Received KB/sec, Sent KB/sec, Avg Bytes are shown in Figure

7.11.

Summary Report											
Name: <input type="text" value="Summary Report"/>											
Comments:											
Write results to file / Read from file											
<input type="text" value="Filename"/> <input type="button" value="Browse..."/> Log/Display Only: <input type="checkbox"/> Errors <input type="checkbox"/> Successes <input type="button" value="Configure"/>											
Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/...	Sent KB/sec	Avg. Bytes	
HTTP POST ...	12204	112	62	9303	177,93	0,02%	6,2/sec	3,42	1,31	568,0	
HTTP Request	3	0	0	1	0,47	100,00%	1,0/min	0,02	0,00	1032,0	
HTTP GET R...	12201	209	127	6451	232,77	0,00%	7,1/sec	88,55	2,29	12782,0	
TOTAL	24408	160	0	9303	212,85	0,02%	12,3/sec	80,46	3,30	6673,6	

Figure 7.11 Statistical summary report of HTTP calls

8 Conclusion

With this project developed, the trips can be recorded by using mobile application and can be shown by using both mobile and web applications. For users, system is developed to manage trips to save time and effort. Users can save videos, photos, audios along to the route. The trips can be categorized as single and group. For single trips, only one user can create his or her own trip, can save the route and attach medias to the path. For group trips, application allows users same functionality as single trip's in addition to that they can see each others locations simultaneously, trip participants can combine saved trip medias to the creator user's medias. Application can guess what the type of the trip by processing the speed and accuracy of the GPS signals received, there are 4 labels for that, these are walking, running, bicycle and riding. Users can share the created trips with the other users. They can choose privacy of trip content which can be only me, only my friends and everybody. Trips are stored in the server for every trip sharing option except only me, for that selection the content will be stored in the users mobile device in order to make the system efficient. This application will offer users a practical solution to save the time spent on sharing a trip on any social media environment or on the internet, because it contains a social media platform inside. Users can add other users as friends to be able to contact with them and to be able to see the trips they participated, created and shared. Users can interact with other users, they can like or comment each others trips, if they think the trip has inappropriate content they can create a complaint, complaints are evaluated by the administrators of the system. Administrators have right to hide them if they think the complainant is right. Users can also search trips depending upon the type, location or creator of it, they can also search for other users. Users can download the trips to their mobile device that are created by other users if they want, by downloading them they can prefer to follow the route same as downloaded one, or they can see the path of downloaded trip, monitor it but create a totally new one and use the downloaded one as a guide, application allows users every functionality even though user prefer to follow the downloaded one. The project could be improved with more complex social media platform which includes extensive trip search system, trip recommendation

system. Also mobile application could be improved with getting more details about trip and developing a better tracking system. These are evaluated and are among our future works.

References

- [1] M. Garage. (). Trip tracker, [Online]. Available: <https://www.microsoft.com/en-us/garage/profiles/trip-tracker/> (visited on 10/30/2017).
- [2] ——, (). Location and obsevation platform, [Online]. Available: <https://www.loop.ms/> (visited on 10/30/2017).
- [3] PathHackers. (). Route tracker, [Online]. Available: <https://play.google.com/store/apps/details?id=com.kendroid.android.routetracker> (visited on 10/30/2017).
- [4] Polarsteps. (). Polar steps, [Online]. Available: <https://www.polarsteps.com> (visited on 10/30/2017).
- [5] G. Inc. (). Google maps, [Online]. Available: <https://www.google.com.tr/maps> (visited on 10/30/2017).
- [6] Wikipedia. (2017). Android, [Online]. Available: [https://en.wikipedia.org/wiki/Android_\(operating_system\)](https://en.wikipedia.org/wiki/Android_(operating_system)) (visited on 11/28/2017).
- [7] ——, (2018). Ios, [Online]. Available: <http://www.wikizero.net/index.php?q=aHR0cHM6Ly90ci53aWtpcGVkaWEub3JnL3dpa2kvxLBUw> (visited on 01/01/2018).
- [8] Google. (2017). Android studio, [Online]. Available: <https://developer.android.com/studio/intro/index.html> (visited on 11/28/2017).
- [9] T. Point. (2017). Sqlite, [Online]. Available: https://www.tutorialspoint.com/android/android_sqlite_database.htm (visited on 11/28/2017).
- [10] ——, (2017). Android emulator, [Online]. Available: https://www.tutorialspoint.com/android/android_emulator.htm (visited on 11/28/2017).
- [11] Google. (2017). Google map api, [Online]. Available: https://developers.google.com/maps/faq#what_is (visited on 11/28/2017).
- [12] Wikipedia. (2018). Mac os, [Online]. Available: <http://www.wikizero.net/index.php?q=aHR0cHM6Ly90ci53aWtpcGVkaWEub3JnL3dpa2kvTWFjX09T> (visited on 01/01/2018).
- [13] docs.oracle.com/. (2016). Java ee açıklaması, [Online]. Available: <https://docs.oracle.com/javaee/6/firstcup/doc/gkhoy.html> (visited on 12/31/2017).
- [14] eclipse.org/. (2016). Eclipse açıklaması, [Online]. Available: <https://www.eclipse.org/ide/> (visited on 12/31/2017).

- [15] sublimetext.com/. (2016). Sublime text açıklaması, [Online]. Available: <https://www.sublimetext.com> (visited on 12/31/2017).
- [16] mysql.com/. (2016). Mysql açıklaması, [Online]. Available: <https://www.mysql.com/about/> (visited on 12/31/2017).
- [17] ——, (2016). Mysql workbench açıklaması, [Online]. Available: <https://www.mysql.com/products/workbench/> (visited on 12/31/2017).
- [18] stackoverflow.com/. (2016). Bootstrap açıklaması, [Online]. Available: <https://stackoverflow.com/questions/14546709/what-is-bootstrap/> (visited on 12/31/2017).
- [19] docs.angularjs.org/. (2016). Angular js açıklaması, [Online]. Available: <https://docs.angularjs.org/guide/introduction> (visited on 12/31/2017).
- [20] developer.mozilla.org/. (2016). Javascript açıklaması, [Online]. Available: https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript (visited on 12/31/2017).
- [21] tomcat.apache.org/. (2016). Apache tomcat açıklaması, [Online]. Available: <http://tomcat.apache.org/> (visited on 12/31/2017).
- [22] docs.spring.io/. (2016). Spring framework açıklaması, [Online]. Available: <https://docs.spring.io/spring/docs/current/spring-framework-reference/overview.html> (visited on 12/31/2017).
- [23] ——, (2016). Spring tool suit açıklaması, [Online]. Available: <https://spring.io/tools/sts> (visited on 12/31/2017).
- [24] getpostman.com/. (2016). Postman açıklaması, [Online]. Available: <https://www.getpostman.com/> (visited on 12/31/2017).
- [25] en.wikipedia.org/. (2016). Putty açıklaması, [Online]. Available: <https://en.wikipedia.org/wiki/PuTTY> (visited on 12/31/2017).
- [26] ——, (2016). Vnc açıklaması, [Online]. Available: https://en.wikipedia.org/wiki/Virtual_Network_Computing (visited on 12/31/2017).
- [27] winscp.net/. (2016). Vnc açıklaması, [Online]. Available: <https://winscp.net/eng/docs/introduction> (visited on 12/31/2017).
- [28] jmeter.apache.org/. (2016). Jmeter açıklaması, [Online]. Available: <http://jmeter.apache.org> (visited on 12/31/2017).
- [29] en.wikipedia.org/wiki/Htop. (2017). Htop açıklaması, [Online]. Available: <https://en.wikipedia.org/wiki/Htop> (visited on 12/31/2017).
- [30] T. Ulakbim. (2017). Pardus minimum system requirements, [Online]. Available: <http://www.pardus.org.tr/surumler/> (visited on 11/28/2017).
- [31] Google. (2017). Android system requirements, [Online]. Available: <https://developer.android.com/studio/index.html> (visited on 11/28/2017).
- [32] E. IDE. (2017). Eclipse ide, [Online]. Available: <http://www.eclipse.org/downloads/eclipse-packages/> (visited on 11/28/2017).
- [33] MySQL. (2017). Mysql, [Online]. Available: <https://www.mysql.com/support/SupportedPlatforms/Workbench.html> (visited on 11/28/2017).

- [34] Peazip. (2017). Zip compression benchmarks, [Online]. Available: <http://www.peazip.org/peazip-compression-benchmark.html> (visited on 11/28/2017).
- [35] Scaleway. (2017). Scaleway, [Online]. Available: <https://www.scaleway.com/> (visited on 11/28/2017).
- [36] politeknik.org.tr. (2016). Tmmob 2017 yılı mühendis asgari ücretini belirledi, [Online]. Available: <http://politeknik.org.tr/tmmob-2017-yili-muhendis-asgari-ucretini-belirledi/> (visited on 11/28/2017).
- [37] Hepsiburada.com. (2017). Dell vostro 5468 8gb ram 256 gb ssd, [Online]. Available: https://urun.n11.com/dizustu-bilgisayar/dell-vostro-5468-g20f82n-i5-7200-1482564g-940mx-P214420418?cid=604001&gclid=CjwKCAiAr_TQBRB5EiwAC_QCq3CPfU6zxKPfjBguonH33douPlwIr0qv-ZH6Ch67XBopfQ8kmvQAXxoCSrcQAvD_BwE&gclsrc=aw.ds (visited on 11/28/2017).
- [38] Gittigidiyor.com. (2017). General mobile gm6, [Online]. Available: [https://urun.gittigidiyor.com/cep-telefonu-ve-aksesuar/general-mobile-gm-6-4-5g-tek-sim-telpa-garantiliidir-313702023?pkw=&pmt=&sc_gcsp=gg&ggmktg=PaidSearch*PLA*236136807*22544670567*58375879647*%7btype%7d*313702023*pla-387386522094&gclid=CjwKCAiAr_TQBRB5EiwAC_QCq9xKmi0Z1QR9HjtoWk73Nr1FSrNGx0meWF46YyiDv9TP2wfqMU6xwBoCvMgQAvD_BwE&gclsrc=aw.ds&mpch=ads&scxid=14489-145818-2357-0">https://urun.gittigidiyor.com/cep-telefonu-ve-aksesuar/general-mobile-gm-6-4-5g-tek-sim-telpa-garantiliidir-313702023?pkw=&pmt=&sc_gcsp=gg&ggmktg=PaidSearch*PLA*236136807*22544670567*58375879647*%7btype%7d*313702023*pla-387386522094&gclid=CjwKCAiAr_TQBRB5EiwAC_QCq9xKmi0Z1QR9HjtoWk73Nr1FSrNGx0meWF46YyiDv9TP2wfqMU6xwBoCvMgQAvD_BwE&gclsrc=aw.ds&mpch=ads&scxid=14489-145818-2357-0](https://urun.gittigidiyor.com/cep-telefonu-ve-aksesuar/general-mobile-gm-6-4-5g-tek-sim-telpa-garantiliidir-313702023?pkw=&pmt=&sc_gcsp=gg&ggmktg=PaidSearch*PLA*236136807*22544670567*58375879647*%7btype%7d*313702023*pla-387386522094&gclid=CjwKCAiAr_TQBRB5EiwAC_QCq9xKmi0Z1QR9HjtoWk73Nr1FSrNGx0meWF46YyiDv9TP2wfqMU6xwBoCvMgQAvD_BwE&gclsrc=aw.ds&mpch=ads&scxid=14489-145818-2357-0) (visited on 11/28/2017).

Curriculum Vitae

FIRST MEMBER

Name-Surname: Tarık Nural

Birthdate and Place of Birth: 07.01.1995, İstanbul

E-mail: tariknural00@gmail.com

Phone: 0535 261 1429

Practical Training: SSI SCHAEFER
ARÇELİK A.Ş

SECOND MEMBER

Name-Surname: Murat Baki Yücel

Birthdate and Place of Birth: 04.01.1996, Kayseri

E-mail: bakiyucel38@gmail.com

Phone: 0507 915 8686

Practical Training: Pronic Yazılım
Kartaca Bilişim

Project System Informations

System and Software: Linux, Java, Android Studio, MySQL, Android, Spring Boot, Javascript, AngularJS, Bootstrap, CSS, Android Emulator, Postman, JUnit, Google Map API, Mail Service, Android Phone, Git, Eclipse

Required RAM: Mobile Application: 1.5GB

Web Application: 8GB

Required Disk: Mobile Application: 8GB

Web Application: 1TB