Bilkent University

Department of Computer Engineering

Senior Design Project

Snatch

High-Level Design Report

Group Members: Kerem Ayöz
Yasin Balcancı
Alper Kılıçaslan
Mert Saraç
Cansu Yıldırım

Supervisor: Prof. Dr. Özcan Öztürk

Jury Members: Vis. Prof. Dr. Fazlı Can
Asst. Prof. Dr. A.Ercüment Çiçek

Innovation Expert: Prof. Dr. Veysi İşler

High-Level Design Report
May 17, 2019

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2.

Contents

1 Introduction

1.1 Purpose of the System

Events are vital part of a social club since events are indicator of a social club's quality. Due to some economic constraints and increasing inflation in Turkey, arranging events become harder. There are two alternative ways to decrease the cost of an event which are sponsorships with companies and partnership with other social clubs. Also, for services like accommodation, catering and transportation, sponsorships are needed. However, traditional way to find a sponsorship or partnership is trying to contact with every possible candidate even if they are not willing to do. Thus, possibility of having positive responses are very low. On the other hand, there some companies that want to promote their names by sponsorships. Due to the fact that there are numerous social clubs, it is hard to find good social club to be sponsor. In this way, our system is designed as a social platform that brings these two target customers together. In the system, our key point is recommendation system that recommends possible candidates for both partnerships and also sponsorships. For example, a photograph club's possible sponsorship candidates might be companies like Nikon, Canon and Fuji Film, and its possible partnership might be another photography club. As an alternative, if there is an event in Bilkent campus, Bilkent Hotel might be good candidate for accommodation since it is close to Bilkent campus.

Another main purpose is serving messed ways of communication channels such as Whatsapp[1], Google Drive [2] and Survey Monkey [3] to social clubs as a more compact way. Having too many communication channel might causes misunderstandings and creates messy atmosphere among the club about communication. In our system, all of these functions can be easily reachable in one application.

Finally, there are also individual users that can discover new social clubs and new events by their locations as an another and final target customer.

1.2 Design Goals

1.2.1 Reliability

- Recommendation system must be reliable to give the most compatible partner or sponsor advice.
- Geographic based social club and event recommendations need to give reliable results to users.

1.2.2 Availability

- Since it is mobile app and web application, it should be both accessible on markets like Google Play Store and App Store and also on web for 24/7.

1.2.3 Extensibility

- Since there are some possible features for future, our system must be open to add new features.
- Also, there might be needed to some changes according to users' reviews.

1.2.4 Ease of Learning

- Since there are three target customers that have various age range, our system should be easily learnable.
- Our system promises ease of finding sponsorships and partnership, it should not be a hard activity.
- Creating event is one of the key functions so, it should be easy to learn.

1.2.5 Security

- The system need to be capable of protect personal information of individual users, private information about companies and social clubs.
- Since there is a login function, passwords are needed to be well-protected.

1.3 Definitions, Acronyms and Abbreviations

- **HTML :** Hypertext Markup Language
- **CSS:** Cascading Style Sheets
- **React Native:** Javascript framework for writing mobile applications for IOS and Android
- **IOS:** Mobile Operating System
- **Android**: Mobile Operating System
- **API:** A set of functions and procedures allowing the creation of applications that access the features or data of an operating system, application, or other service.

1.4 Overview

Snatch is social platform that brings social clubs which are looking for possible sponsorships, partnerships and services such as accommodation, transportation and catering or other possible social clubs to arrange an event together, companies which are willing to be a sponsor and individuals who are willing to discover new social events and new social clubs. Also, a better and compact way to communication is served to social clubs to prevent mess of many links, files and chats in different platforms such as Whatsapp [1], Google Drive [2] and Survey Monkey [3].

For social clubs, they create events on our application to promote. When they are creating it, they will receive recommendation about the events like sponsor, place of event, need of catering etc. They can also chat among participants of the club, create polls and surveys, share, edit and store files. Since they will receive request from individuals, they can accept or reject these requests.

For companies, they will receive notifications about possible events that are on development process. If they want to collaborate, they can contact with corresponding social club. They can peak old events of social clubs.

For individuals, they can see events and list of social clubs according to their interests. They can join the events on application. Also, they can send requests to social clubs to join them. They can also start new social club with adequate number of board members. They can peak old events of social clubs.

2 Current Software Architecture

In this part, competitors of Snatch will be discussed. Our application mostly depends on the event recommendation. There are lots of applications that does something with the events. Also our application provides an organizer so that events will be organized much faster. Moreover, we will provide users a communication section so that users of the application can communicate with each other.

2.1 Meetup

Meetup[4] is a platform for finding and building local communities. People use Meetup to meet new people, learn new things, find support, get out of their comfort zones, and pursue their passions. Meetup has the following attributes:

- User can create a group and join a group.
- Groups can be found from a location.
- Groups are recommended to the users.
- Users can see past events.
- Users can share photos.
- Users can open polls.

2.2 Nextdoor

Nextdoor[5] is a platform that brings the neighbourhood together. It creates a social network within the neighbourhood. Nextdoor has following attributes:

- User can communicate.
- User can organize a group.
- Sharing and receiving information from local public agencies.
- User can sell their items.

2.3 Nearify

Nearify[6] is a platform that shows users nearest musical or theatrical events. Nearify has following attributes:

- Recommends new events.
- User can buy tickets for the events.
- User can follow their favourite artists.
- User can see the nearby events.

2.4 WhatsApp

Whatsapp[1] is a communication platform for mobile devices. More than 1 billion people uses Whatsapp. Whatsapp has the following attributes:

- Users can communicate with each other.
- Users can send documents or videos or photos.
- Users can share stories.
- Users can create chat groups.

2.5 Google Drive

Google Drive[2] is a file sharing system. Users can share files or create files and collaborate with other users. Google Drive has the following attributes:

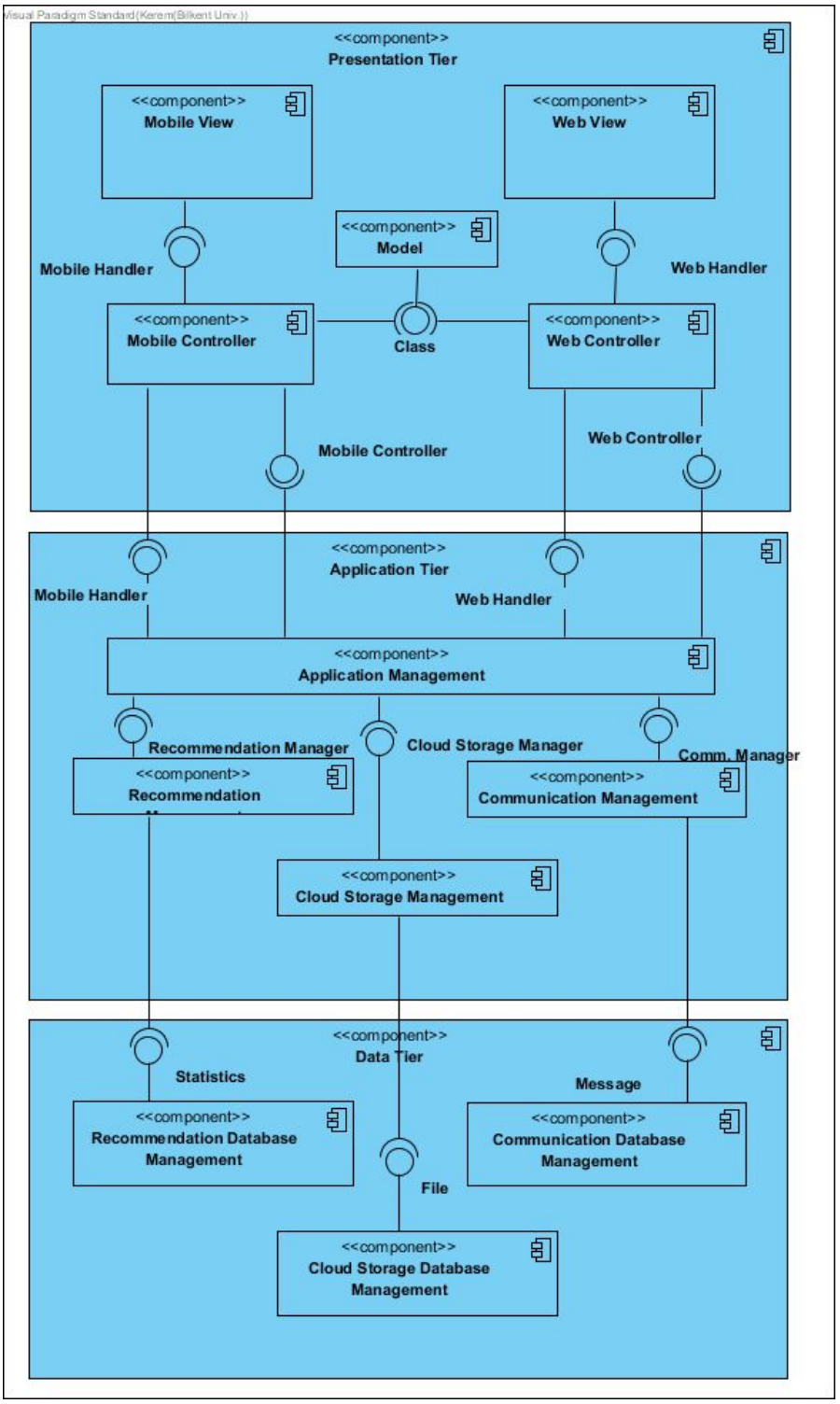
- Users can share files.
- Users can create shared files.
- Users can comment on the files.
- Users can chat on opened documents.

3 Proposed Software Architecture

3.1 Overview

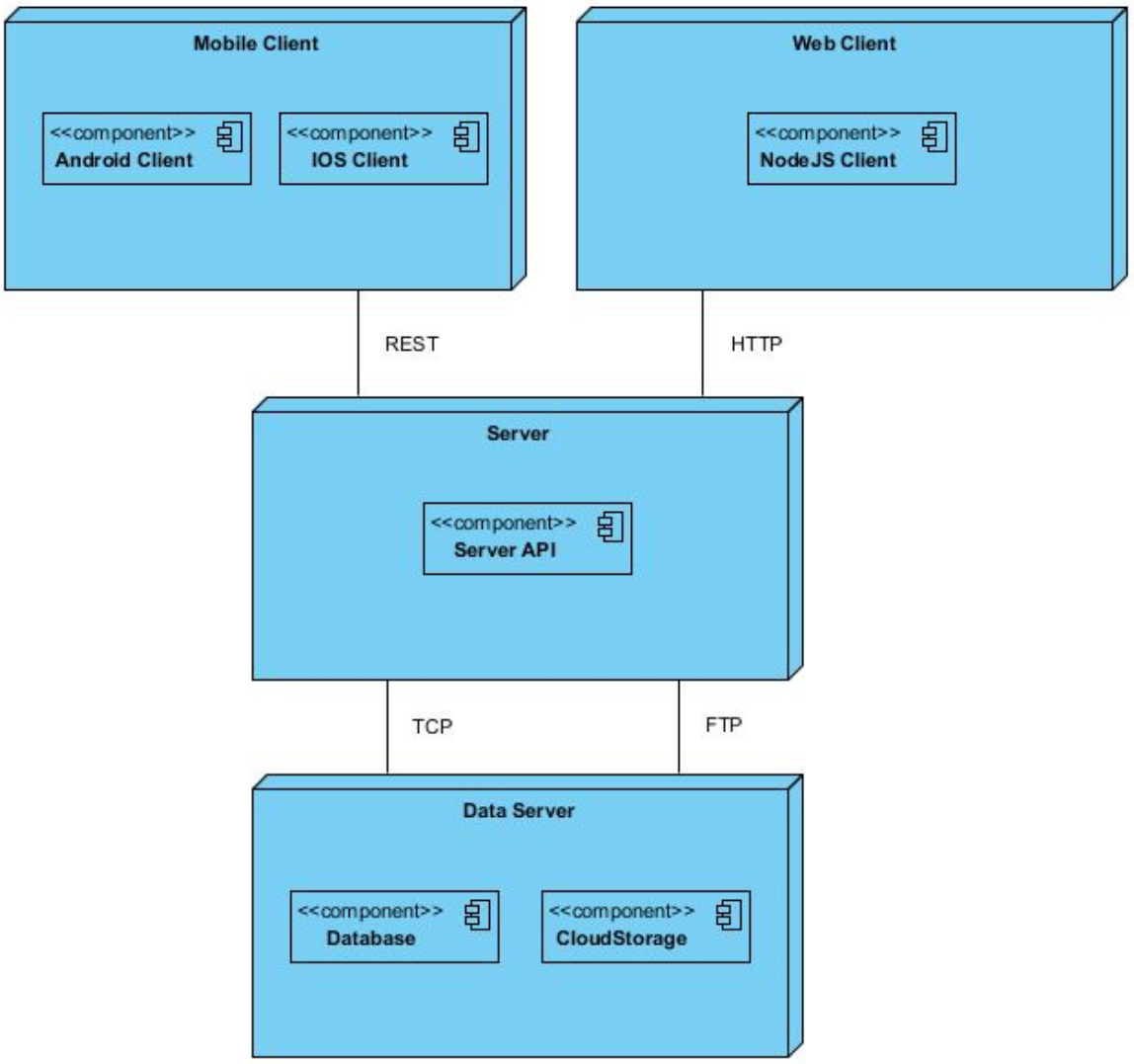
Snatch will have a 3-tier software architecture with each tier have distinct components. This architecture is chosen for our system since we will have multiple different clients and we will serve different type of data storages. Also, our system's data processing work might be computationally expensive and in order to overcome that problem recommendation system and data analysis components will be in application tier which is our server. To achieve simplicity and good performance, we will use 3-tier architecture.

3.2 Subsystem Decomposition



Snatch has a 3-tier architecture as described above. These subsystems are presentation tier, application tier and data tier.

3.3 Hardware/Software Mapping



The tiers of the system will be divided into a client and server software. System have 2 different clients; mobile and web client. These clients will connect to the server with different protocols. Client softwares will be responsible for presentation layer of the system. The server software will contain the logic for data and application tiers such as recommendation engine and cloud storage engine. This software will run on an application server to process and respond to requests sent by clients.

The data tier logic will have 2 different components. One of them will work as a usual database component and stores user data and application data. Second part will work as a cloud storage in order to provide file storage access for users. The data tier components and application tier components could run on the same or different physical machines depending on performance. Also data tier components could be seperated to increase response time of user queries.

The database component and cloud storage component will be developed with MongoDB[7] and FileZilla[8] respectively. Application tier will be developed with Amazon AWS services[9]. The client software will represent the presentation tier and will be the interface to end users. On the mobile client part, React Native[10] technology will be used to develop application on cross platform. On the web client part Node.js[11] technology will be used. These softwares will be used for presentation tier of the system.

3.4 Persistent Data Management

Snatch will store personal data of users and data of communities and enterprises. It will also provide a file storage system which will be stored as well. Cookies etc. will not be used, only persistent data storage will be provided. In order to store all the necessary data, Snatch will use MongoDB, a cross-platform document-oriented database program. File storage will be provided via GridFS[12], which allows storage of binary files. Text files will be stored as regular UTF-8 strings.

3.5 Access Control and Security

Because of the crucial data it stores, access control is an important issue for Snatch. Snatch will use bcrypt[13] library to encrypt user passwords before saving them to database, so that the passwords will be hashed and the system will compare the hashed versions of a newly-entered password to the stored password before authenticating a user. Users will have to enter their passwords besides their emails in order to access both their own data and the data that they are authorized to access, belonging to the community they involve in.

3.6 Global Software Control

User will have to have an account to use the general functionality of Snatch. They will be able to register with unique emails addresses by providing necessary information and start using Snatch.
Community settings and insights will be accessible to authorized users (admins) only. Authorized users can be determined by either the founder of the community or other authorized users of that community. Whereas the chat and file storage of the communities will be accessible to participants of that community.
Enterprise insights and settings will be accessible to the founder of the enterprise and the other admins only, as well.
Event settings will be accessible to the admins of the organizers of the event.

3.7 Boundary Conditions

3.7.1 Initialization

User must be using the web application or the mobile application to use Snatch. Also user must create an account to use the features of our application. The application requires internet to operate itself.

3.7.2 Termination

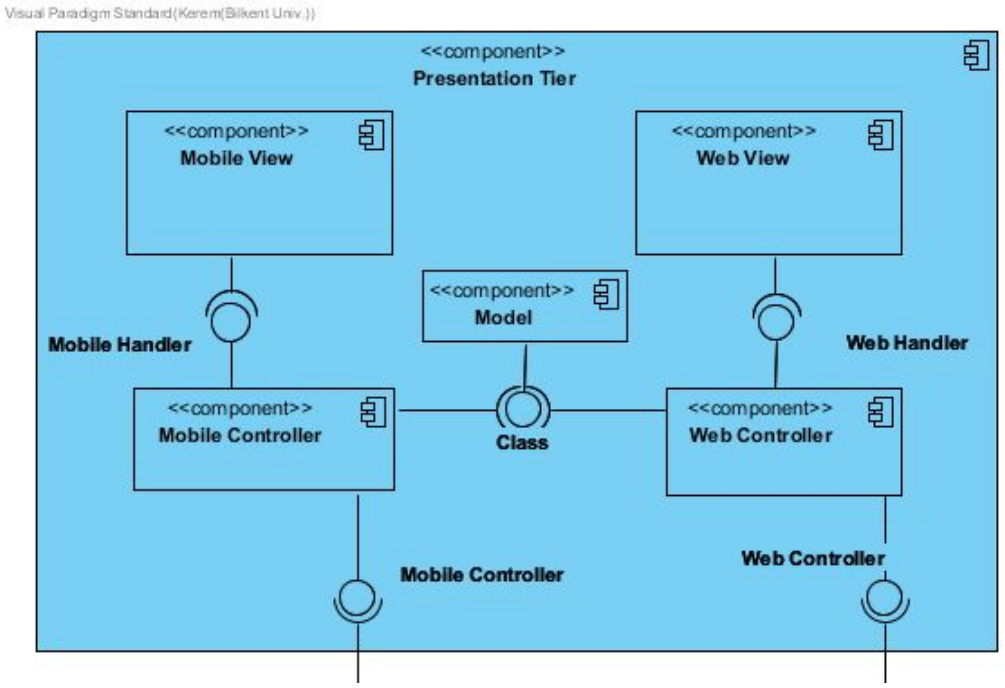
User can terminate the app by logging out or closing the browser window/tab.

3.7.3 Failure

User can get some errors while using the application. For example, if the user enter their credentials in login screen wrong, an appropriate error message will be shown. Also, lack of the internet connection may cause some problems. Lastly, the third party APIs may cause some errors. Our goal is to minimize the errors, and help the users about the errors.

4 Subsystem Services

4.1 Presentation Tier Subsystem



Presentation Tier Subsystem is the front end subsystem that provides the user interface. It displays the information and content to the end user. It interacts with the Application Tier Subsystem and sends the necessary information to it.

It has 5 components which are:

1. Mobile View
2. Mobile Controller
3. Web View
4. Web Controller
5. Model

1. Mobile View Component: It provides interface for views of the mobile application. It can be updated by Mobile Controller component when some changes happen in the Presentation tier or Application tier. Mobile View Component classes have user listeners that listens the user inputs and send them to the Mobile Controller component classes. React Native will be used for this component classes.

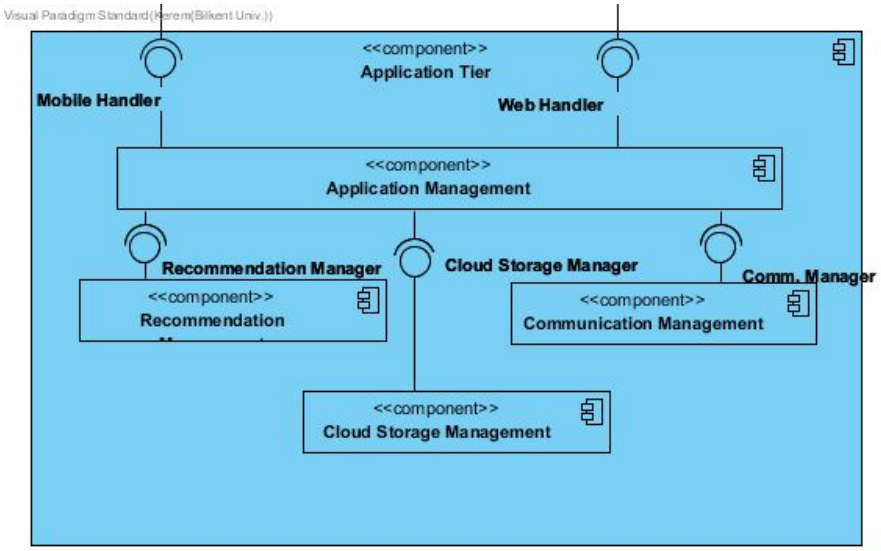
2. Mobile Controller Component: It basically controls the Mobile View Component. Mobile Controller Component classes are responsible for the controls of operations that are done in the mobile view classes and handling the operation that comes from view classes. Operations between Presentation tier and Application tier are done in this component.

3. Web View Component: It provides interface for views of the Web application. It can be updated by Web Mobile Controller component when some changes happen in the Presentation tier or Application tier. Web View Component classes have user listeners that listens the user inputs and send them to the Web Controller component classes. This component classes will be built on HTML, CSS and Javascript.

4. Web Controller Component: It basically controls the Web View Component. Web Controller Component classes are responsible for the controls of operations that are done in the web view classes and handling the operation that comes from view classes. Operations between Presentation tier and Application tier are done in this component.

5. Model Component: It is used by Controller components. It reduces coupling in the presentation tier by separating its classes from the view and controller classes. When there is a need for creation of a model that needs to be done in presentation tier, it is created in Model Component.

4.2 Application Tier Subsystem



Application Tier Subsystem provides the main functionality of the system. It connects the Presentation Tier and Data Tier.

It has 4 components which are:

1. Application Management Component
2. Recommendation Management Component
3. Communication Management Component
4. Cloud Storage Management Component

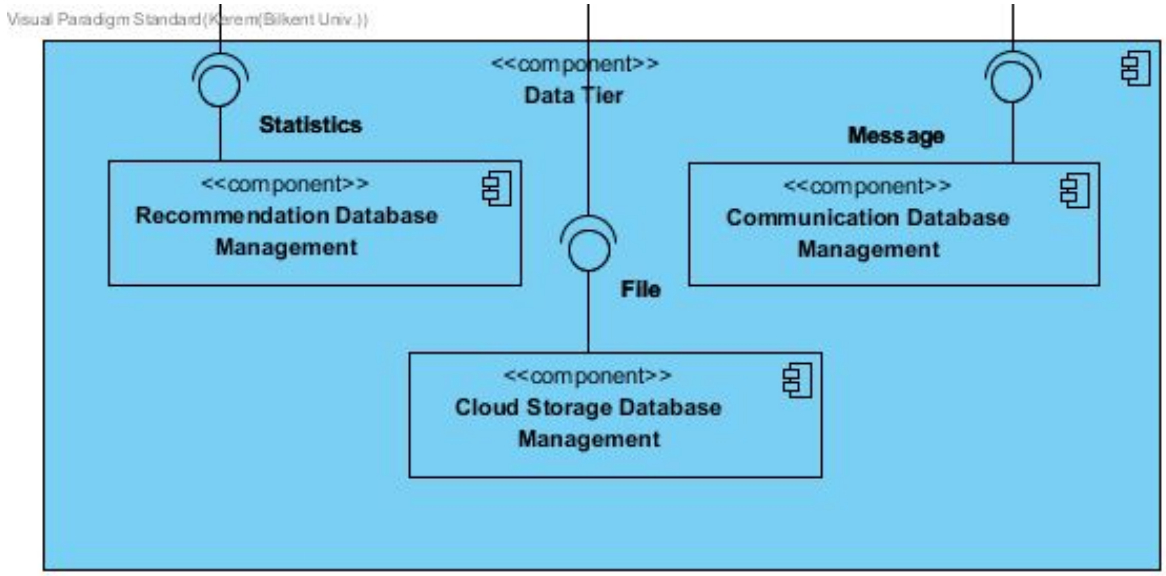
1. Application Management Component: It handles the requests that are coming from the client side (presentation tier).

2. Recommendation Management Component: It provides the recommendation system that will be done to individuals, communities and enterprises. According to the statistics that are obtained from Recommendation Database, it is responsible to manage the appropriate recommendations for the users.

3. Communication Management Component: It provides the communication within the community groups in the system. It is responsible for managing of sending/receiving messages, creating polls and creating checklists.

4. Cloud Storage Management Component: It provides the cloud storage for sending/receiving files within community group environments.

4.3 Data Tier Subsystem



Data Tier Subsystem provides the data management of the system. It cannot directly connect with the presentation tier for the security. It interacts with the Application Tier Subsystem only.

It has 3 components:

1. Recommendation DataBase Management Component
2. Communication DataBase Management Component
3. Cloud Storage DataBase Management Component

1. Recommendation DataBase Management Component: It stores the statistics of the users according to their interests and location. It stores the enterprises' and communities' working area, event types etc. It will be used to make appropriate recommendations for the Snatch users.

2. Communication DataBase Management Component: It stores the incoming/outgoing messages of the users.

3. Cloud Storage DataBase Management Component: It stores the files that are sent via the system.

5 Glossary

- **Community:** A type of group which can be created with the approval of 2-3 Snatch users. It is kind of a social group that organizes social events. It has an environment that enables its members to communicate easily.
- **Enterprise/Company:** A type of group which represents the profit-based companies.

6 References

[1] WhatsApp [Online] Available: https://www.whatsapp.com/. Accessed: May 17th 2019.
[2] Google Drive: Free Cloud Storage for Personal Use [Online] Available: https://www.google.com/drive/. Accessed: May 17th 2019.

[3] "Veri platformumuzla merakınıza güç verin," *SurveyMonkey*. [Online]. Available: https://tr.surveymonkey.com/. [Accessed: 17-May-2019].
[4] "Hakkında," *Meetup*. [Online]. Available: https://www.meetup.com/tr-TR/about/. [Accessed: 17-May-2019].
[5] "We are for neighbors. ," *About*. [Online]. Available: https://about.nextdoor.com/. [Accessed: 17-May-2019].
[6] N. D. Labs, "Discover Events Near You," *Nearify*. [Online]. Available: https://www.nearify.com/. [Accessed: 17-May-2019].
[7] C. to W. projects, "MongoDB," *Vikipedi*, 31-May-2012. [Online]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly90ci53aWtpcGVkaWEub3JnL3dpa2kvTW9uZ29EQg. [Accessed: 17-May-2019].

[8] C. to W. projects, "FileZilla," *Vikipedi*, 29-May-2006. [Online]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly90ci53aWtpcGVkaWEub3JnL3dpa2kvRmlsZVppbGxh. [Accessed: 17-May-2019].

[9]C. to W. projects, "subsidiary of Amazon that provides on-demand cloud computing platforms on a metered pay-as-you-go basis," *Wikipedia*, 16-May-2019. [Online]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvQW1hem9uX1dlYl9TZXJ2aWNlcw. [Accessed: 17-May-2019].

[10] C. to W. projects, "React Native," *Wikipedia*, 16-May-2019. [Online]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvUmVhY3RfTmF0aXZl. [Accessed: 17-May-2019].

[11] N. Foundation, *Node.js*. [Online]. Available: https://nodejs.org/en/. [Accessed: 17-May-2019].

[12] "GridFS" *GridFS - MongoDB Manual*. [Online]. Available: https://docs.mongodb.com/manual/core/gridfs/. [Accessed: 17-May-2019].
[13] "bcrypt," *npm*. [Online]. Available: https://www.npmjs.com/package/bcrypt. [Accessed: 17-May-2019].