



Bilkent University

---

Department of Computer Engineering

# Senior Design Project

*Project Name: StudyB*

## High-Level Design Report

Alara Yaman  
Betim Dođan  
Berfu Deniz Kara  
Mert Özerdem  
Sera Fırıncıođlu

**Supervisor:** Özcan Öztürk

**Innovation Expert:** Aras Bilgen

**Jury Members:** Varol Akman and Eray Tüzün

High-Level Design Report : Dec 31, 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.

<b>1. Introduction</b>	<b>3</b>
1.1 Purpose of The System	3
1.2 Design Goals	4
1.3 Definitions, Acronyms, and Abbreviations	4
1.4 Overview	5
<b>2. Current Software Architecture</b>	<b>5</b>
<b>3. Proposed Software Architecture</b>	<b>6</b>
3.1 Overview	6
3.2 Subsystem Decomposition	6
3.2.1 Android Application	7
3.2.2 Database Manager	7
3.2.3 Web Service	7
3.3 Hardware/Software Mapping	7
3.4 Persistent Data Management	8
3.5 Access Control and Security	8
3.6 Global Software Control	9
3.7 Boundary Conditions	9
3.7.1 Initialization	9
3.7.2 Termination	10
3.7.3 Failure	10
<b>4. Subsystem Services</b>	<b>10</b>
4.1 Database	11
4.2 Model	11
4.3 Controller	12
4.4 View	12
4.5 Service	13
<b>5. New Knowledge Acquired and Learning Strategies Used</b>	<b>13</b>
<b>6. Glossary</b>	<b>14</b>
<b>7. References</b>	<b>14</b>

# 1. Introduction

This is the High-Level Design report for StudyB. In our report starts with an introduction, in which we provide the purpose of the system, design goals, definitions, and an overview of the StudyB. It is followed by the current software architecture of the applications in the field. Afterward, we provide a detailed architecture of our system. We finalize the report with subsystem services, glossary and references.

## 1.1 Purpose of The System

In our project we aims to provide a platform where willing students can find study partners, share course materials and thus help each other. Students will be able to get together after confirming that they are Bilkent students through their bilkent emails. On the course pages, students will be able to share course materials such as course notes and questions and will be able to answer and help each other. Students who are looking for a study partner will be able to create study rooms and will be able to work together with the people who enrolled in these rooms, either face-to-face or in a virtual environment.

There will be a scoring system to encourage students' cooperation and prevent lurkers. After a 2-week trial period, members who do not participate in any way but benefit from the materials of other people on the platform will be removed from the platform. Also a score system is developed according to the amount of participation which will have scoreboards according to the wishes of the users.

Among the members who want to show their participation amounts in the score table, the most participating member will appear on the scoreboard of course page as the master of that course.

In addition, according to the courses students have enrolled, system will have their exam schedules and information, and they will be notified when the exam dates are approaching.

## 1.2 Design Goals

**Usability:** Our project should be easy to use. When the user will open the StudyB, they do not need extra information to understand it. The users want to use our application for studying they do not want to spend extra time to learn to use it. That reason we want to keep it as simple as possible.

**User-friendly Interface:** We will choose a simple design because of our target is students from different class and subject in Bilkent University. We will try to apply Material Design features which are developed by Google to our user interface [1]. It will make our work much easier and our interface will be seen neat.

**Privacy:** In our application, we stored people information in our university. According to laws we have to keep the users information safely and do not share with other authorities.

**Low Cost:** Our application will be a free application. If the student has an Internet connection they do not need to pay anything.

**Performance:** The application will be open for all students at the same time. The application uses web service database, hence the requests and responses should be shorten.

## 1.3 Definitions, Acronyms, and Abbreviations

HTML: Hypertext Markup Language

User: A user of StudyB with a valid account.

JSON: JavaScript Object Notation

XML: Extensible Markup Language

## 1.4 Overview

Our project is a mobile application both for Android systems which is uniquely designed for socializing and helping students to raise their grade. It is especially for the students who do not know anyone from their lectures. For example, if the student did not quarete yet and want to study as a group they can use our application to arrange study time. Exam tips are good before the exam because these tips are based on the other students who took that lesson before.

## 2. Current Software Architecture

The research has been done about the current study application which is similar or related to our project. There are lots of them which are really close to our application but not exactly the same. Some of the applications are described with short information below.

- ❑ StudyBlue has a website and mobile application which helps students to study efficiently. They provide lots of study materials such as notes, flashcards. StudyBlue is acquired by the Chegg web-site which is a platform for students and its features mainly are asking questions or homework and getting solutions to their problems. [1]

- ❑ StudyBuddy is interactive study mobile application. Its features are the closest to our application. Students can chat and meet about their questions or they can take their questions pictures to ask the tutor. [2]

- ❑ MyStudyLife is a web-based and mobile application which helps students to organize their educational programs. MyStudyLife is a remainder also for tasks, homework, quizzes or exams. [3]

- ❑ Eodev is a website and mobile application for generally primary and high school students. Eodev provides to its user to ask questions about their classes. [4]

The applications are really similar to each other in terms of the efficiency of study. But none of them are implemented for any specific university students.

Our project aims to serve for only Bilkent University students. Since the questions which are confronted by the students will be mostly similar and the students will have

a chance to meet with each other in a short time on campus to criticize their questions, the efficiency of our application will be high.

### 3. Proposed Software Architecture

#### 3.1 Overview

This section contains detailed information on StudB’s system. First subsystem decomposition will be explained, then hardware-software mapping will be shown. After that, persistent data management information will follow the access control and security part. Lastly global software control and boundary conditions will be explained in detail.

#### 3.2 Subsystem Decomposition

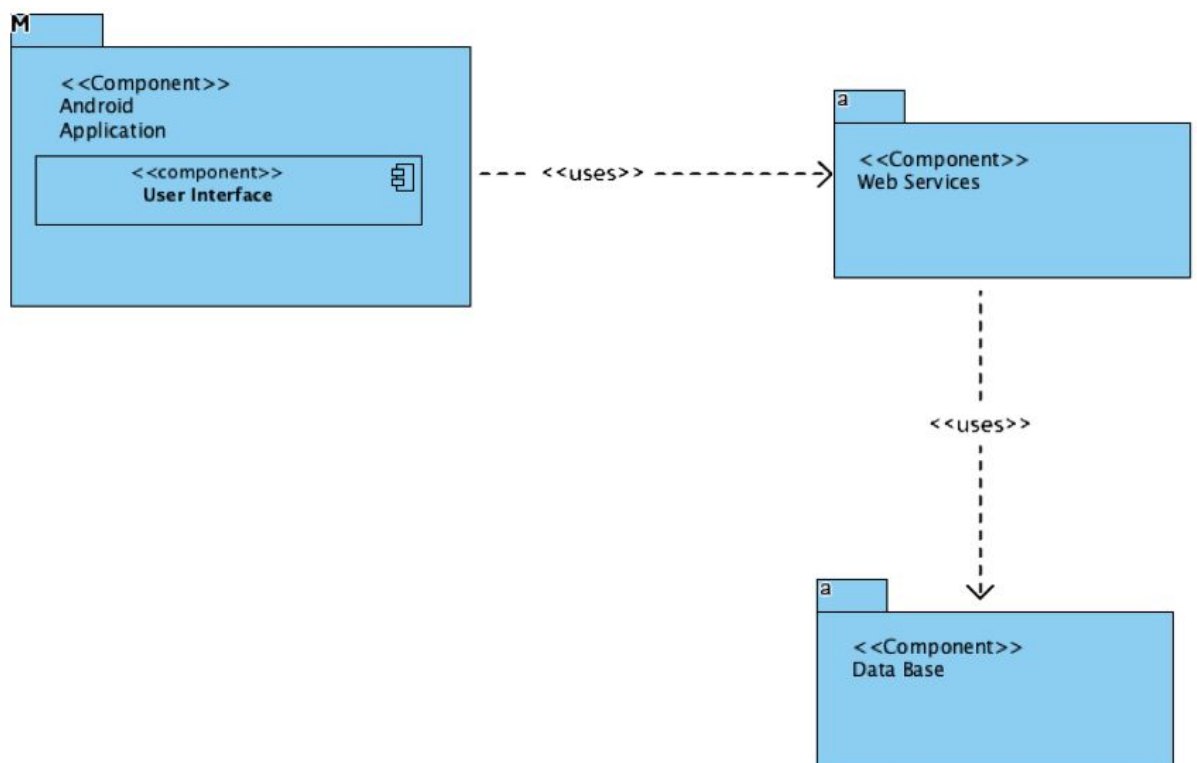


Diagram 1 : Subsystem Decomposition Diagram

### 3.2.1 Android Application

Android Application has two sub components, user and Admin interface. Components are implemented with Angular because it is a robust framework for developing apps and offers users a very engaging experience and easy connection with Web Service.

### 3.2.2 Database Manager

Database Manager is the information manager that is stored in the database. Database stores the information of the users and the courses.

### 3.2.3 Web Service

Web Service links the application to the database server. When a user creates an account or an event is registered, the web service will be used to provide it.

## 3.3 Hardware/Software Mapping

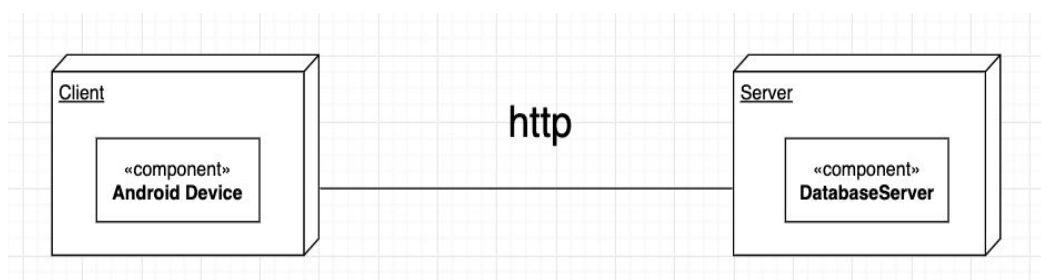


Diagram 2 : Hardware/Software Mapping Diagram

StudyB has two main hardware side: server and client. Client node connects with the users by using StudyB user interface. Server holds the data of the app and hosts the application. Client needs to communicate with server in order to get the data required for the application. This communication will be done by using HTTP.

### 3.4 Persistent Data Management

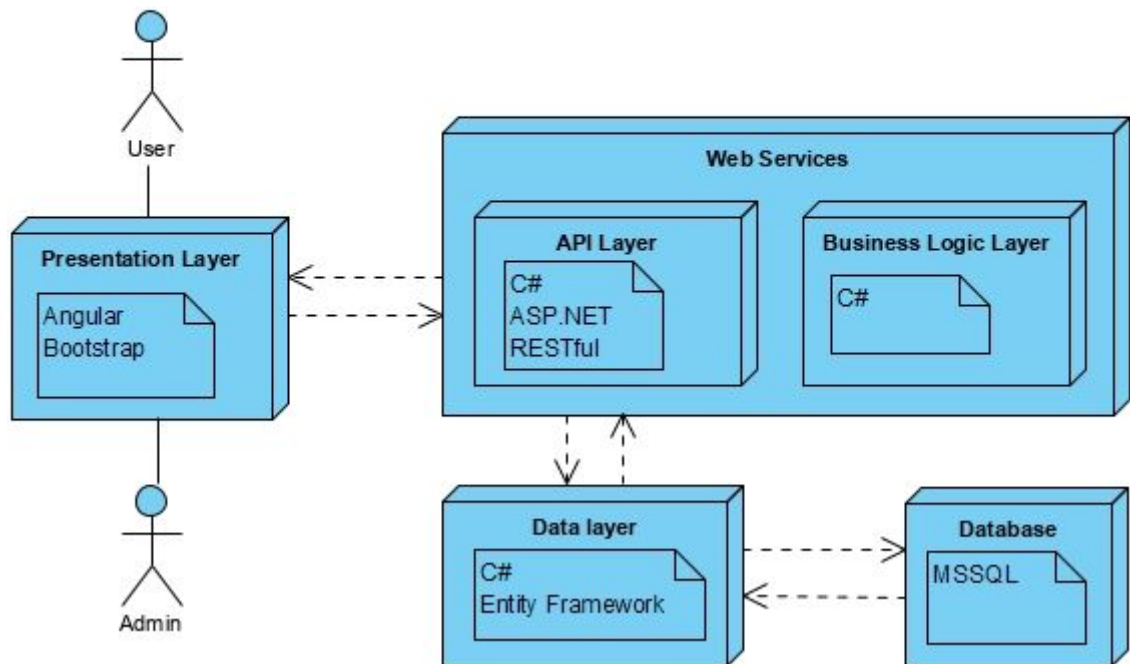


Diagram 3 : Persistent Data Management Diagram

Our project has a large data that needed to be managed; user passwords, emails and nick names will be stored. Chatrooms and shared data in chatrooms will be added to the database too.. Because of this, we chose to use layers with database and web services. We used two layer; presentation layer and data layer which both in connection with web services. Web services are contained of API layer business logic layer. Other than these parts there is also a database which is structured by MSSQL.

### 3.5 Access Control and Security

StudyB is protected via angular guards on the front-end side and authentication which is implemented to back-end server-side.

Guards protect the server from giving unnecessary information to an uncredited person. Guards protect the server from SQL injection, unauthenticated file tracking. Guard accomplishes this by sending the 404 client-side error to the user when he or she asks for uncredited information



from the server. This blocks the user from progressing further by using uncredited requests.

Authentication is the other security measurement of the StudyB. This measurement is checked by the back-end server which compares user's username and password and assigns its credentials if username and password of that user match. The server then allows for further information fetch from the back-end server and write information to the back-end server. In future implementations, passwords of the users will be secured via using SHA-1, MD-5, etc..

### **3.6 Global Software Control**

Our system will be 3-Tier architectural style which are User Interface, Web Services and Database. As soon as the sign up data is taken from the user, it is processed by accessing database to check whether the given email is valid or not. After registration is approved by sending a validation link to the email account, that account will be saved to the database via web services.

Rest of the application is going to be performed to the user through user interface layer. The documents that are uploaded and shared in the application is going to use web services again through user interface. Documents are going to be saved to the database of our application. Whenever the users want to talk with other students privately, they will use the chat feature. In implementation of chat, client side which is the users can communicate through the server. Since server side needs the data of the messages that are sent, requested data will be kept in database tier.

### **3.7 Boundary Conditions**

#### **3.7.1 Initialization**

Accessing and using the StudyB application is going to be done by downloading the application on IOS or Android device from the market. After downloading the application, users can easily register by entering their Bilkent mail and creating a password. All the departments

and courses will be uploaded to the application and can be selected to reach the screens of application only when there is an internet connection.

### **3.7.2 Termination**

After users log in to the program, they will stay active in the application except they do not specifically hit the sign out button. Application will continue to stay open in the background even if they do not sign out. Termination can only be done by clicking the sign out button which is preferred by users.

### **3.7.3 Failure**

It is going to be expected that StudyB might fail in case of two situations. Accessing the application without any internet connection will be resulted as a failure. Users can not connect to the application in this case.

Second case of failure is invalid information to register. If users try to sign up with wrong email accounts which is any account except Bilkent is going to be accepted as a failure. Sign up page will be shown to the users again until getting a valid mail account.

## **4. Subsystem Services**

We used the MVC design pattern which is commonly used for designing for heavy user interaction applications. MVC pattern separates concerns so that every part is specialized in some field while communicating with each other.

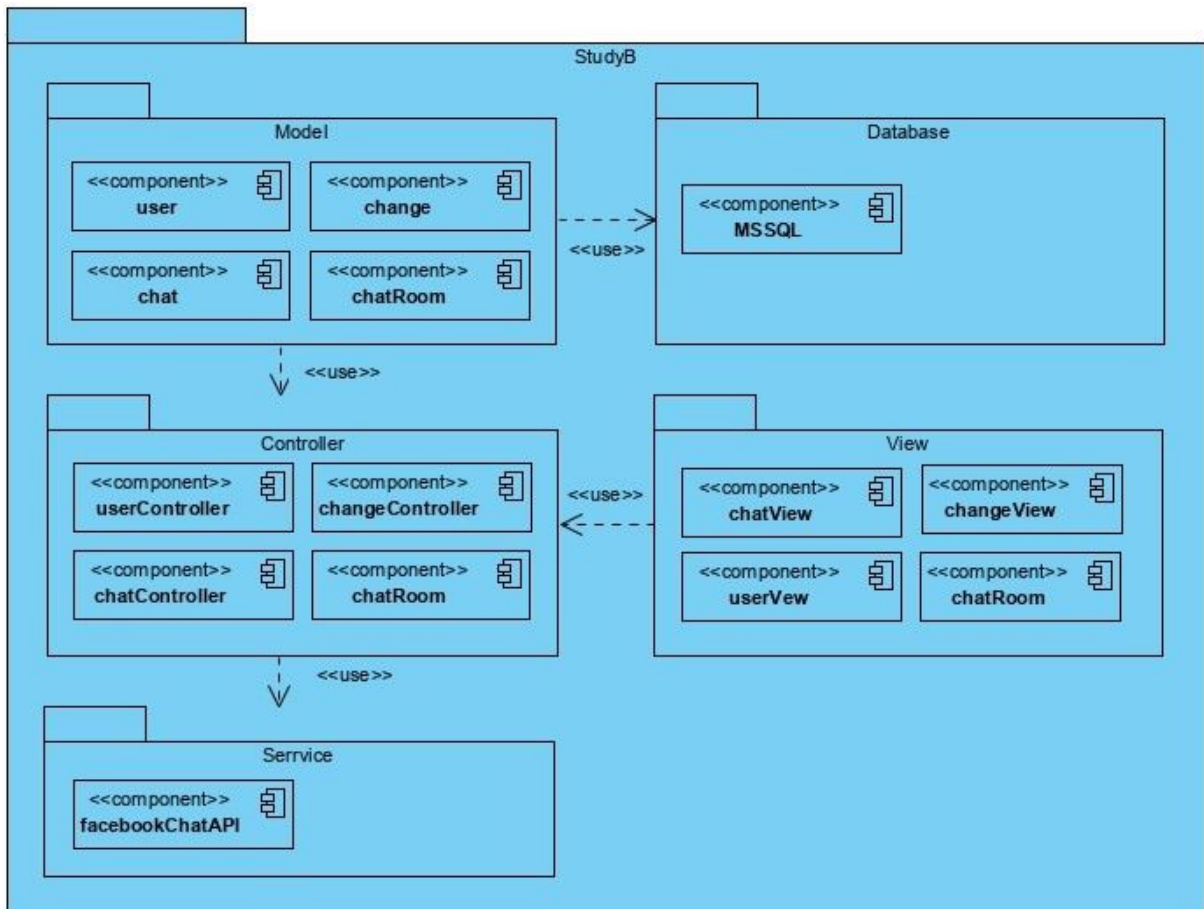


Diagram 4 : Subsystem Services Diagram

## 4.1 Database

Database subsystem is responsible for holding user, chatroom, and chat information. If a change occurs in the that is confirmed by the Controller a model of change is sent to Database for keeping or alteration of the information. For keeping information MSSQL database server will be used.

## 4.2 Model

The User model keeps related information about the user and sets information for CRUD requests.

The Change model keeps the information about when and by which user change has occurred.

The Chat model keeps the information about specific chat rooms and its contents, which can be texts and document's addresses.

The Chatroom model keeps the information about chat rooms but it does not contain any information about chat context. This model only keeps which users participate which chatroom.

### **4.3 Controller**

The Controller classes act as requests come from the View. And does functional operations on them then returns to the model for safekeeping of information in the Database or returns directly to the View.

The UserController class responds to requests that are made by the user and account related interactions.

The ChatController class responds to the reading or writing of information that occurs in a specific chat room that is made by a user via the View.

The ChatroomController class responds to the join or leave requests that are made by a user via using the View.

The ChangeController is activated when a change is occurred in the application whether it is a push text info or leave from chat info, and keeps the responsible user's username and change occurrence time.

### **4.4 View**

The UserView is responsible for displaying information about the user and sending a CRUD request to the server which is related to the user.

The ChatView is responsible for displaying information about the chat room's contents and sending a CRUD request to the server which when a change is requested by a user.

The ChatroomView is responsible for displaying information about the chatrooms and its participants. This view sends a CRUD request to the server whenever a user wants to display chat rooms or its participants.

The ChangeView displays the time and responsible user for the events that occurred in the past.

## **4.5 Service**

The FacebookChatAPI is used to handle simple chat functionalities that can be outsourced.

# **5. New Knowledge Acquired and Learning Strategies Used**

While designing our project, we had the chance to acquire some new knowledge such as gamification. In short, gamification can be defined as the process of taking something that already exists – a website, a business application, an online community – and integrating game mechanics into it to motivate engagement, loyalty and participation. We started to investigate this subject on the direction of our innovative expert and we decided to use it in our project because it has many advantages such as leveraging competitiveness to motivate users, increasing application awareness and motivation, engaging users. The score table and scoring features of our application were added considering the gamification concept.

Secondly, we learned about Angular framework and decided to use it while developing our application because it makes it easy to develop web applications. According to our research, Angular solves challenges in creating a app by combining declarative templates, dependency injection, end-to-end tooling, and integrated best practices.

In addition, when we started this project, we decided to meet at least once a week as a team in order to avoid some of the problems that we had with other previous group activities. Thanks to this situation, we have developed ourselves in team work, because even when there is no document to deliver, we were able to find good ideas in a more comfortable way, not in a busy period, and to strengthen the communication between us.

## 6. Glossary

No glossary is needed to read this report.

## 7. References

[1] "Study Blue" <https://www.studyblue.com> [Accessed: December 24]

[2] "Study Buddy Mobile" <https://www.studybuddymobile.com> [Accessed: December 24]

[3] "My Study Life" <https://www.mystudylife.com> [Accessed: December 24]

[4] "E-ödev" <https://eodev.com> [Accessed: December 24]