Lab 1 - Study Buddy Description

Thy Barsell

CS 411

Mr. Thomas Kennedy

8 March, 2019

Version: 2.0
1. Introduction

When it comes to studying for a class, the task can be quite challenging, especially for those who do not particularly excel at note taking or absorbing information. The formation of a study group can greatly alleviate the stress of preparing for a class, as working together with peers means sharing knowledge and motivating each other to stay focused. A study group is an effective method of learning, and it can be the difference between passing or failing a class.

However, there also exist many challenges when trying to form a study group. Understanding learning styles between like-minded peers to form a study group is a challenge to students. Conflicts regarding scheduling, learning styles, and personality clashes can also get in the way of a successful study group. For students who take classes online, or students who have social anxiety or are shy, this is even more difficult. Figure 1 shows the current process for students trying to form a study group.

![Figure 1: Current Process Flow](image)

The solution to this issue is Study Buddy, a web application. Study Buddy will act as a platform that helps students and people pursuing new knowledge find the perfect study groups to achieve academic success. The end result will be a robust, accessible web application that will allow individuals to successfully find a study buddy or form a study group. Figure 2 demonstrates the intended process for students using Study Buddy.
Having a tool such as Study Buddy will greatly enhance the way individuals and students interact with one another in their classes and ultimately, improve the way they learn and study their course material.

2. Study Buddy Product Description

As an application, Study Buddy’s core feature is to match people with others based on set preferences and filters. Individual availability, study subject, learning styles, and group size, among other attributes, will be used to find potential study buddies for someone. The application will have integrated tools for groups to use, such as in-app private messaging and Slack integration to allow study groups to have a platform to communicate effectively.

2.1. Key Product Features and Capabilities

Study Buddy will utilize an Intelligent Buddy Matching algorithm in order to find the optimal study partner for individuals. This algorithm will take into account the learning styles of users, which will be obtained from a brief survey that all new users will have to take. The survey in question will simply query users about how they typically tend to study, and what methods work best for them. The idea is that individuals work best together with those with similar study habits.
The Intelligent Buddy Matching algorithm will use similarity scores to determine how well suited two individuals are for each other. This will be computed by finding the cosine of the angle between two vectors, the vectors being the learning style scores pulled from the user surveys. Users who would make better study buddies will have a cosine similarity closer to 1, while those who would not make a good match will have a cosine similarity closer to -1.

\[
\cos \theta = \frac{W \cdot \hat{c}}{|W| |\hat{c}|} = \frac{2}{\sqrt{2} \times \sqrt{3}} = \frac{2}{\sqrt{6}} \approx 0.8165
\]

Figure 3: Intelligent Buddy Matching Algorithm

Another core feature of Study Buddy will be the use of integrated API for online collaboration tools. While Study Buddy will be intended for use to find a study partner, the application will also serve as a space and platform for users to collaborate on together. Study Buddy will have integration features with:

- Google Drive: a means for users to share documents with each other through Google Docs, Slides, and Sheets.
- Google Hangouts: a platform for users to communicate through video chat and set up meetings in an online space.
- Codeshare: the ability to both video chat and develop programs and code together in real-time.
- Jupyter Notebook: a means for users to develop programs and share code together.
- Slack: a platform to chat with one another, as well as share documents.
- Git: a shared repository in order to work on projects together.
Study Buddy will also have in-app messaging. This will allow users to communicate with one another without having to use an external service.

2.2. Major Components (Hardware/Software)

End users of Study Buddy will be using devices with Internet access in order to access the web application. The web application will be developed using Java Server Faces (JSF), and will interface with a database, where information will be stored. To facilitate online study groups, external services, such as the ones mentioned in Section 2.1, will be integrated.

![Diagram of Major Functional Components](image)

**Figure 4: Major Functional Components Diagram**

Regarding the software end of how the web application will be developed, this can be split up into design patterns and development tools:

2.2.1 Design Patterns

Study Buddy will utilize multiple design patterns for clean, concise code and a well thought out design.

- Model View Controller (MVC) will aim to separate the three different component layers of the application for parallel development. Containing each component’s logic will allow for faster development. For example, one developer can work on the front-end while another can work on the back-end, with no code changes impacting each developer.
- Facade creates a forward facing interface to hide complex business logic. This interface will provide methods that front-end developers can call without getting
into the details of how the business logic is implemented. This will work with both complex database queries and external API calls.

- Data Access Object (DAO) removes the data access layer from the rest of the application, and disguises complex operations and queries in the database. By providing all of the database querying logic, all necessary methods needed to add to or search through a database are present.

### 2.2.2 Development Tools

- Java Server Faces (JSF) will act as the foundation for the Study Buddy web application. As a framework, JSF allows for robust and modern user interface design, and also supports expression language to allow for easy data flow between the application and the server.
- Java Persistence API (JPA) will solve any data persistence issues. JPA will supply the mechanisms required for SQL query construction by using an object-relational mapping technique to achieve object persistence.

### 3. Identification of Case Study

#### 3.1 Customers

Study Buddy will benefit countless individuals and students. However, the main customers for this software would be public schools and universities. By working with educational institutions, Study Buddy can be integrated within these schools’ online portals. For example, with Old Dominion University, Study Buddy could eventually be integrated with myODU for students to be able to access with ease. For public schools, Study Buddy can especially aid students who need to study for standardized testing such as the SAT and the ACT. Similarly, the web application can be used to match students who need to study for more specialized standardized tests, such as the MCAT, GRE, and PCAT at higher education levels.

Besides schools, another main customer would be organizations that issue out certifications and licenses. Oftentimes, individuals trying to obtain these certifications and licenses take online classes in order to learn the information. As such, this is the perfect population of people who would benefit from using Study Buddy.

#### 3.2 End Users

End users will be anyone who is interested in pursuing knowledge. While students will be the main end users for this application, there is no limit to who can use Study Buddy. Within the
student demographic, however, Study Buddy does aim to close the gap between online and shy students with live students. The goal is for students and people of all types to be able to work and study together in an online environment facilitated by the Study Buddy web application.

4. Study Buddy Product Prototype Description

As a prototype application, Study Buddy’s core feature is still to match people with others based on set preferences and filters. The prototype will also feature limited integrated API applications with Google Hangouts.

4.1. Prototype Architecture (Hardware/Software)

The Study Buddy prototype will use the same hardware and software as the proposed final product:

4.1.1 Design Patterns

Study Buddy will utilize multiple design patterns for clean, concise code and a well thought out design.

- Model View Controller (MVC) will aim to separate the three different component layers of the application for parallel development. Containing each component’s logic will allow for faster development. For example, one developer can work on the front-end while another can work on the back-end, with no code changes impacting each developer.
- Facade creates a forward facing interface to hide complex business logic. This interface will provide methods that front-end developers can call without getting into the details of how the business logic is implemented. This will work with both complex database queries and external API calls.
- Data Access Object (DAO) removes the data access layer from the rest of the application, and disguises complex operations and queries in the database. By providing all of the database querying logic, all necessary methods needed to add to or search through a database are present.

4.1.2 Development Tools

- Java Server Faces (JSF) will act as the foundation for the Study Buddy web application. As a framework, JSF allows for robust and modern user interface design, and also supports expression language to allow for easy data flow between the application and the server.
- Java Persistence API (JPA) will solve any data persistence issues. JPA will supply the mechanisms required for SQL query construction by using an object-relational mapping technique to achieve object persistence.

### 4.2. Prototype Features and Capabilities

<table>
<thead>
<tr>
<th>Features</th>
<th>STUDY BUDDY Final Product</th>
<th>STUDY BUDDY Prototype</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GUI</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Login</td>
<td>Allows user entry of authentication credentials</td>
<td>Will be implemented</td>
</tr>
<tr>
<td>New User</td>
<td>Allows a user to create an account</td>
<td>Will be implemented</td>
</tr>
<tr>
<td>Study Preference Setting</td>
<td>Allows users to set study preferences for intelligent buddy matching algorithm</td>
<td>Will be implemented with limited study preference fields</td>
</tr>
<tr>
<td>Search for buddies</td>
<td>Allows user to search for study buddies</td>
<td>Will be implemented</td>
</tr>
<tr>
<td>Create a study group</td>
<td>Allows user to create a study group</td>
<td>Will be implemented</td>
</tr>
<tr>
<td>Set wait time</td>
<td>Allows user to set wait time in case of no matched buddies found</td>
<td>Will not be implemented</td>
</tr>
<tr>
<td><strong>Features</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Web Application</td>
<td>The way in which the user will interact with the Study Buddy application using a web browser</td>
<td>Will be implemented</td>
</tr>
<tr>
<td>Mobile Application</td>
<td>The way in which the user will interact with the Study Buddy application using their smartphone device</td>
<td>Will not be implemented</td>
</tr>
<tr>
<td>Private Message</td>
<td>Allows users to send and receive private messages within the Study Buddy App</td>
<td>Will not be implemented</td>
</tr>
<tr>
<td>Block buddies</td>
<td>Allows users to block buddies with different goals</td>
<td>Will not be implemented</td>
</tr>
<tr>
<td>Partner match by subject of interest</td>
<td>Matching Study Buddies by their own subject interest</td>
<td>Will be implemented</td>
</tr>
<tr>
<td>Intelligent Buddy Matching</td>
<td>Matching Study Buddies with the proprietary algorithm</td>
<td>Will be implemented</td>
</tr>
<tr>
<td>Google Hangout Integration</td>
<td>Allows users to integrate their Google Hangouts accounts for setting meeting times and web conferencing</td>
<td>Will be implemented</td>
</tr>
<tr>
<td>Slack Integration</td>
<td>Allows users to integrate their Slack accounts, allowing channels to be made to aid in communication between Study Buddies</td>
<td>Will not be implemented</td>
</tr>
<tr>
<td>Codeshare integration</td>
<td>Allows users to share their code real time with their Study Buddies</td>
<td>Will not be implemented</td>
</tr>
<tr>
<td>Schedule syncing with &quot;Buddy&quot;</td>
<td>Allows users to make matches with Study Buddies based on the availability they input</td>
<td>Will not be implemented</td>
</tr>
<tr>
<td>Rate your buddy</td>
<td>Allows users to provide feedback on their Study Buddies</td>
<td>Will not be implemented</td>
</tr>
</tbody>
</table>

Table 1: Prototype Features
The Study Buddy prototype will aim to preserve the core features of the final product. As such, much of the features discussed will not be available in the prototype, but will remain to be developed for the final product.

4.3. Prototype Development Challenges

There are three anticipated development challenges:

- Language unfamiliarity: While the prototype will be developed using Java, this may come as a challenge for some team members on the development team who may not have experience programming with Java.
- Framework unfamiliarity: Most team members on the development team have not worked with Java Server Faces (JSF) before. Learning to utilize this framework in an effective manner will be a learning curve to overcome.
- Learning a new development environment: Most team members have not worked with SQL Server 2017 Management Studio. Learning to how to use this interface effectively will be a challenge for team members.
5. Glossary

**Auditory Learner** - Someone who best comprehends information by listening to information rather than reading it or seeing it visually.

**Business Logic** - The programming that manages communication between an end user interface and a database.

**CRUD** - Stands for Create, Read, Update, and Delete. Basic database/application operations.

**Entity Class** - A simple Java Class with member variables and getter and setter methods defined.

**JPA** - Java Persistence Application programming interface is an API for handling all database operations such as storing or retrieve entities from the database.

**JSF** - Java Server Faces is a Java framework that couples the view and servlet into one managed component.

**Kinesthetic Learner** - Someone who best comprehends information by participating in activities or solving problems in a hands-on manner.

**ORM** - Object-relational mapping. Technique for persisting objects into a database table. Tables are modeled after Entity classes.

**Procrastination** - Delaying or postponing a task, which needs to be completed, often to the detriment of the procrastinator.

**Prototype** - A reduced scale version of the final product that will demonstrate the functionality of the completed product in a simulated environment.

**Reading/Writing Learner** - Someone who best comprehends information by reading texts to further absorb information by condensing and rephrasing it in traditional lecture and note-taking environments.

**Study Group** - A small group of students with similar goals who meet regularly to review course material and prepare for exams.
**Visual Learner** - Someone who best comprehends information by visualizing relationships and ideas through maps, charts, diagrams and even essays.
6. References


Codeshare information, Retrieved September 12, 2018, from: https://codeshare.io/


Forgetting All Your Coursework, Retrieved September 16, 2018, from: https://www.blackboard.odu.edu/webapps/discussionboard/do/message?action=list_messages&course_id=_323209_1&nav=discussion_board_entry&conf_id=_351171_1&forum_id=_309142_1&message_id=_7348213_1


Slack Information, Retrieved September 12, 2018, from: https://slack.com/

Google Meet information, Retrieved September 12, 2018, from: https://support.google.com/a/answer/7303775?hl=en
