



# Study Buddy

February 11, 2024

---

**Authored By:** HalfTime Company  
Florida Agricultural & Mechanical University  
College of Science and Technology



## Table of Contents

Overview.....	2
Work Breakdown Structure (WBS).....	3
Gantt Chart.....	5
Network	
Diagram.....	6
Requirements.....	7
Use Cases.....	9
Use Case Diagram.....	12
Research & API Descriptions.....	13
References.....	14



## Mission Statement

"At Halftime, our mission is to revolutionize the collaborative learning experience for college students. We are committed to providing a technologically advanced platform that seamlessly connects students based on their courses, schedules, and preferred locations. Through fostering a culture of knowledge-sharing and peer collaboration, we aim to empower students on their academic journey, promoting success, and enriching the overall higher education experience. Our mission is to make collaborative learning accessible, efficient, and enjoyable for students, contributing to their academic achievements and personal growth."

## Product Description

The Study Group Matching App is a mobile application designed to help college students easily find and form study groups based on their enrolled courses, schedules, and preferred meeting locations on campus. The app will utilize advanced algorithms to match students with similar academic needs, fostering a collaborative learning environment.

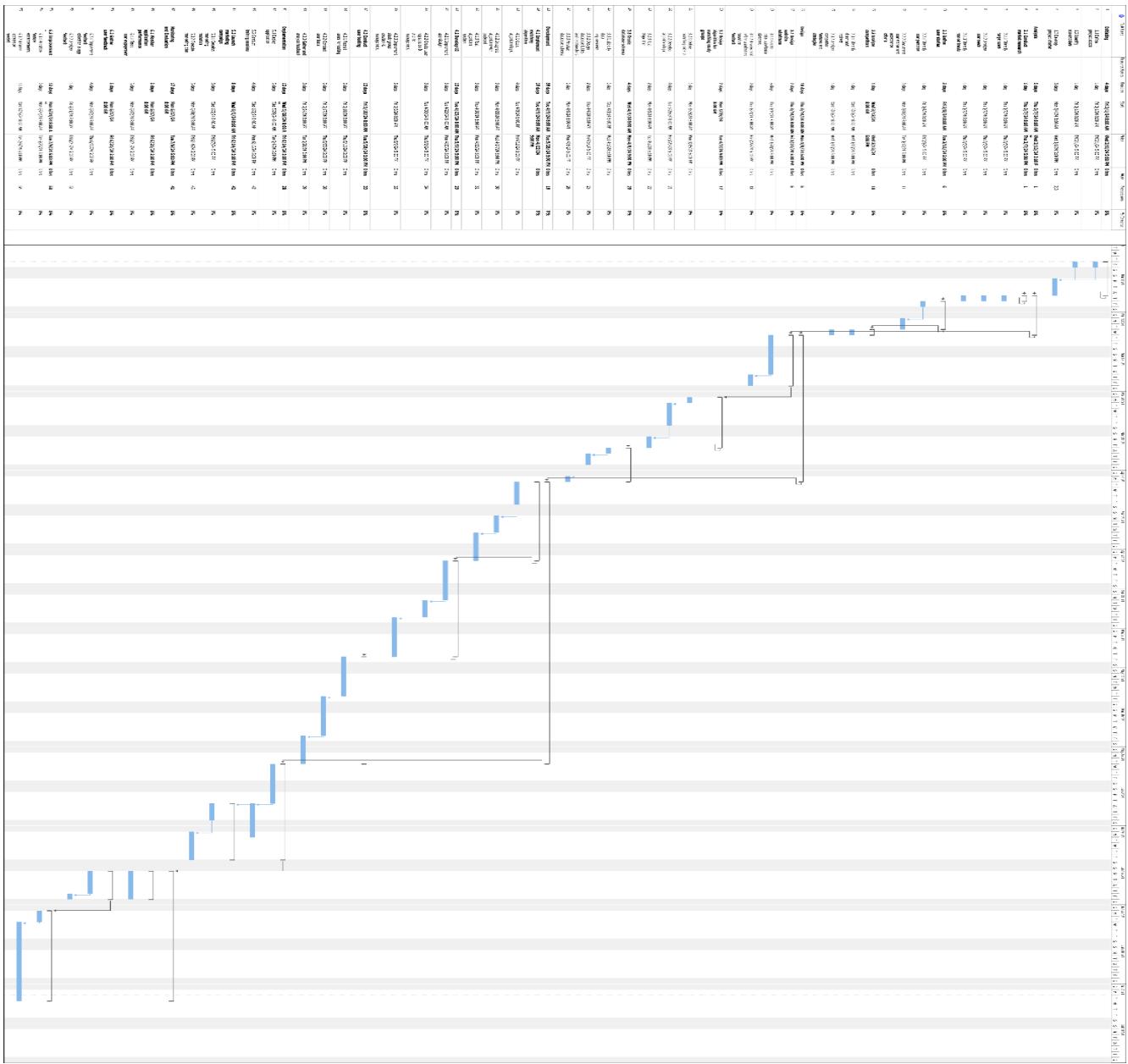
## Work Breakdown Structure (WBS)

	WBS	Task Name	Resource Names	Duration	Start	Finish	Work	Predecessors	% Complete
1	1	<b>Initiating</b>		<b>4 days</b>	<b>Fri 3/1/24 8:00 AM</b>	<b>Wed 3/6/24 5:00 PM</b>	<b>0 hrs</b>		<b>0%</b>
2	1.1	1.1 Define project scope		1 day	Fri 3/1/24 8:00 AM	Fri 3/1/24 5:00 PM	0 hrs		0%
3	1.2	1.2 Identify stakeholders		1 day	Fri 3/1/24 8:00 AM	Fri 3/1/24 5:00 PM	0 hrs		0%
4	1.3	1.3 Develop project charter		3 days	Mon 3/4/24 8:00 AM	Wed 3/6/24 5:00 PM	0 hrs	2,3	0%
5	2	<b>Analysis</b>		<b>5 days</b>	<b>Thu 3/7/24 8:00 AM</b>	<b>Wed 3/13/24 5:00 PM</b>	<b>0 hrs</b>	<b>1</b>	<b>0%</b>
6	2.1	<b>2.1 Conduct market research</b>		<b>1 day</b>	<b>Thu 3/7/24 8:00 AM</b>	<b>Thu 3/7/24 5:00 PM</b>	<b>0 hrs</b>	<b>1</b>	<b>0%</b>
7	2.1.1	2.1.1 Identify target users		1 day	Thu 3/7/24 8:00 AM	Thu 3/7/24 5:00 PM	0 hrs		0%
8	2.1.2	2.1.2 Analyze user needs		1 day	Thu 3/7/24 8:00 AM	Thu 3/7/24 5:00 PM	0 hrs		0%
9	2.1.3	2.1.3 Identify market trends		1 day	Thu 3/7/24 8:00 AM	Thu 3/7/24 5:00 PM	0 hrs		0%
10	2.2	<b>2.2 Define user stories</b>		<b>3 days</b>	<b>Fri 3/8/24 8:00 AM</b>	<b>Tue 3/12/24 5:00 PM</b>	<b>0 hrs</b>	<b>6</b>	<b>0%</b>
11	2.2.1	2.2.1 Identify user personas		1 day	Fri 3/8/24 8:00 AM	Fri 3/8/24 5:00 PM	0 hrs		0%
12	2.2.2	2.2.2 Document user stories and acceptance criteria		2 days	Mon 3/11/24 8:00 AM	Tue 3/12/24 5:00 PM	0 hrs	11	0%
13	2.3	<b>2.3 Analyze competitors</b>		<b>1 day</b>	<b>Wed 3/13/24 8:00 AM</b>	<b>Wed 3/13/24 5:00 PM</b>	<b>0 hrs</b>	<b>10</b>	<b>0%</b>
14	2.3.1	2.3.1 Identify direct and indirect		1 day	Wed 3/13/24 8:00 AM	Wed 3/13/24 5:00 PM	0 hrs		0%
15	2.3.2	2.3.2 Analyze competitor features and strategies		1 day	Wed 3/13/24 8:00 AM	Wed 3/13/24 5:00 PM	0 hrs		0%
16	3	<b>Design</b>		<b>18 days</b>	<b>Thu 3/14/24 8:00 AM</b>	<b>Mon 4/8/24 5:00 PM</b>	<b>0 hrs</b>	<b>5</b>	<b>0%</b>
17	3.1	<b>3.1 Design website wireframe</b>		<b>7 days</b>	<b>Thu 3/14/24 8:00 AM</b>	<b>Fri 3/22/24 5:00 PM</b>	<b>0 hrs</b>	<b>5</b>	<b>0%</b>
18	3.1.1	3.1.1 Create initial wireframe sketches		5 days	Thu 3/14/24 8:00 AM	Wed 3/20/24 5:00 PM	0 hrs		0%
19	3.1.2	3.1.2 Review and refine wireframes based on feedback		2 days	Thu 3/21/24 8:00 AM	Fri 3/22/24 5:00 PM	0 hrs	18	0%
20	3.2	<b>3.2 Design algorithm for matching study groups</b>		<b>7 days</b>	<b>Mon 3/25/24 8:00 AM</b>	<b>Tue 4/2/24 5:00 PM</b>	<b>0 hrs</b>	<b>17</b>	<b>0%</b>
21	3.2.1	3.2.1 Define matching criteria		1 day	Mon 3/25/24 8:00 AM	Mon 3/25/24 5:00 PM	0 hrs		0%
22	3.2.2	3.2.2 Develop algorithm logic		4 days	Tue 3/26/24 8:00 AM	Fri 3/29/24 5:00 PM	0 hrs	21	0%
23	3.2.3	3.2.3 Test Algorithm		2 days	Mon 4/1/24 8:00 AM	Tue 4/2/24 5:00 PM	0 hrs	22	0%
24	3.3	<b>3.3 Create database schema</b>		<b>4 days</b>	<b>Wed 4/3/24 8:00 AM</b>	<b>Mon 4/8/24 5:00 PM</b>	<b>0 hrs</b>	<b>20</b>	<b>0%</b>
25	3.3.1	3.3.1 Identify data requirements		1 day	Wed 4/3/24 8:00 AM	Wed 4/3/24 5:00 PM	0 hrs		0%
26	3.3.2	3.3.2 Design database tables and relationships		2 days	Thu 4/4/24 8:00 AM	Fri 4/5/24 5:00 PM	0 hrs	25	0%
27	3.3.3	3.3.3 Finalize database schema		1 day	Mon 4/8/24 8:00 AM	Mon 4/8/24 5:00 PM	0 hrs	26	0%

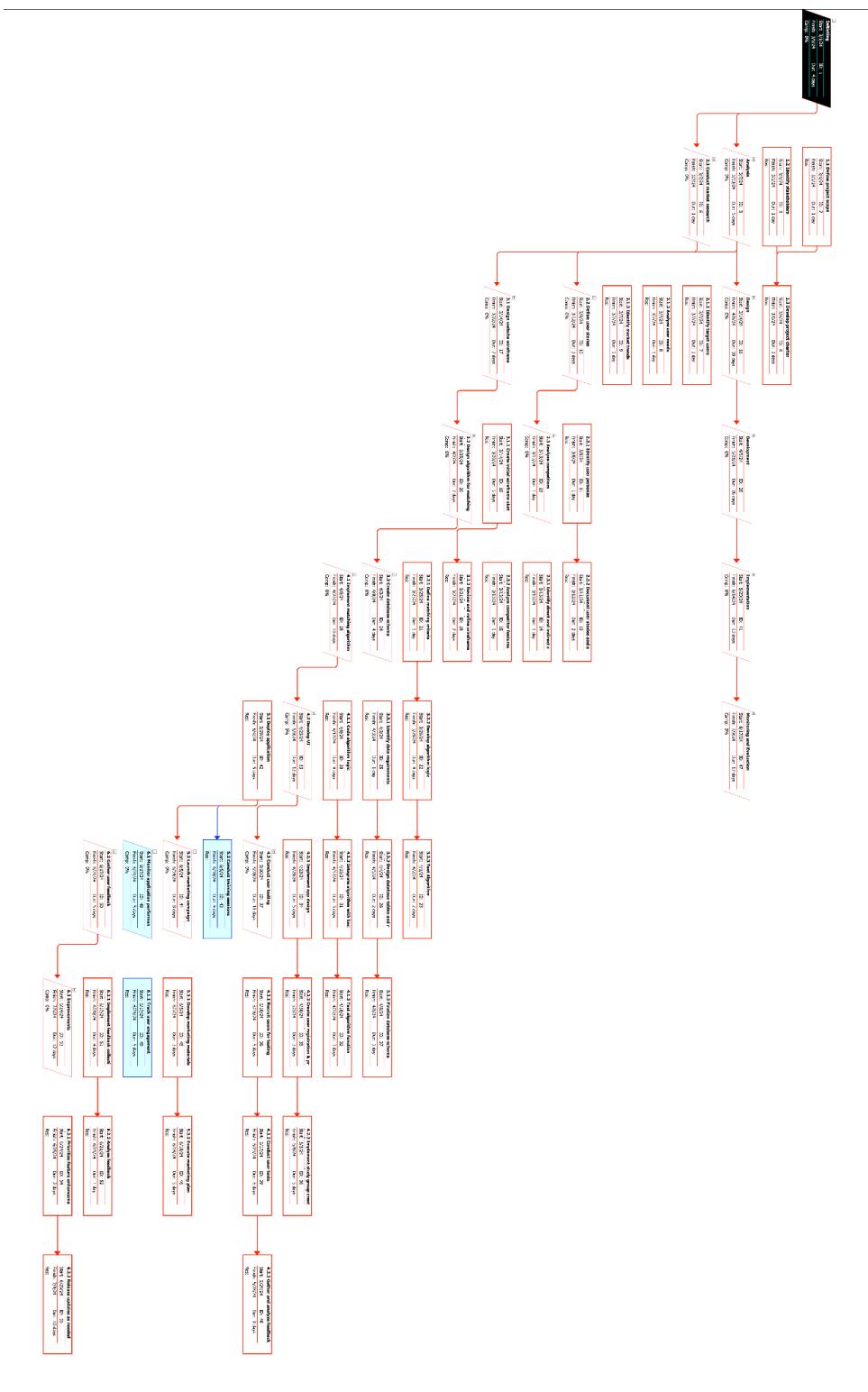


28	<b>4</b>		<b>Development</b>		<b>36 days</b>	<b>Tue 4/9/24 8:00 AM</b>	<b>Tue 5/28/24 5:00 PM</b>	<b>0 hrs</b>	<b>16</b>	<b>0%</b>
29	<b>4.1</b>		<b>4.1 Implement matching algorithm</b>		<b>10 days</b>	<b>Tue 4/9/24 8:00 AM</b>	<b>Mon 4/22/24 5:00 PM</b>	<b>0 hrs</b>		<b>0%</b>
30	4.1.1		4.1.1 Code algorithm logic		4 days	Tue 4/9/24 8:00 AM	Fri 4/12/24 5:00 PM	0 hrs		0%
31	4.1.2		4.1.2 Integrate algorithm with backend		3 days	Mon 4/15/24 8:00 AM	Wed 4/17/24 5:00 PM	0 hrs	30	0%
32	4.1.3		4.1.3 Test algorithm function		3 days	Thu 4/18/24 8:00 AM	Mon 4/22/24 5:00 PM	0 hrs	31	0%
33	<b>4.2</b>		<b>4.2 Develop UI</b>		<b>13 days</b>	<b>Tue 4/23/24 8:00 AM</b>	<b>Thu 5/9/24 5:00 PM</b>	<b>0 hrs</b>	<b>29</b>	<b>0%</b>
34	4.2.1		4.2.1 Implement app design		5 days	Tue 4/23/24 8:00 AM	Mon 4/29/24 5:00 PM	0 hrs		0%
35	4.2.2		4.2.2 Create user registration & profile management		3 days	Tue 4/30/24 8:00 AM	Thu 5/2/24 5:00 PM	0 hrs	34	0%
36	4.2.3		4.2.3 Implement study group creation & management		5 days	Fri 5/3/24 8:00 AM	Thu 5/9/24 5:00 PM	0 hrs	35	0%
37	<b>4.3</b>		<b>4.3 Conduct user testing</b>		<b>13 days</b>	<b>Fri 5/10/24 8:00 AM</b>	<b>Tue 5/28/24 5:00 PM</b>	<b>0 hrs</b>	<b>33</b>	<b>0%</b>
38	4.3.1		4.3.1 Recruit users for testing		5 days	Fri 5/10/24 8:00 AM	Thu 5/16/24 5:00 PM	0 hrs		0%
39	4.3.2		4.3.2 Conduct user tests		5 days	Fri 5/17/24 8:00 AM	Thu 5/23/24 5:00 PM	0 hrs	38	0%
40	4.3.3		4.3.3 Gather and analyze feedback		3 days	Fri 5/24/24 8:00 AM	Tue 5/28/24 5:00 PM	0 hrs	39	0%
41	<b>5</b>		<b>Implementation</b>		<b>13 days</b>	<b>Wed 5/29/24 8:00 AM</b>	<b>Fri 6/14/24 5:00 PM</b>	<b>0 hrs</b>	<b>28</b>	<b>0%</b>
42	5.1		5.1 Deploy application		5 days	Wed 5/29/24 8:00 AM	Tue 6/4/24 5:00 PM	0 hrs		0%
43	5.2		5.2 Conduct training sessions		4 days	Wed 6/5/24 8:00 AM	Mon 6/10/24 5:00 PM	0 hrs	42	0%
44	<b>5.3</b>		<b>5.3 Launch marketing campaign</b>		<b>8 days</b>	<b>Wed 6/5/24 8:00 AM</b>	<b>Fri 6/14/24 5:00 PM</b>	<b>0 hrs</b>	<b>42</b>	<b>0%</b>
45	5.3.1		5.3.1 Develop marketing materials		3 days	Wed 6/5/24 8:00 AM	Fri 6/7/24 5:00 PM	0 hrs		0%
46	5.3.2		5.3.2 Execute marketing plan		5 days	Mon 6/10/24 8:00 AM	Fri 6/14/24 5:00 PM	0 hrs	45	0%
47	<b>6</b>		<b>Monitoring and Evaluation</b>		<b>17 days</b>	<b>Mon 6/17/24 8:00 AM</b>	<b>Tue 7/9/24 5:00 PM</b>	<b>0 hrs</b>	<b>41</b>	<b>0%</b>
48	<b>6.1</b>		<b>6.1 Monitor application performance</b>		<b>5 days</b>	<b>Mon 6/17/24 8:00 AM</b>	<b>Fri 6/21/24 5:00 PM</b>	<b>0 hrs</b>		<b>0%</b>
49	6.1.1		6.1.1 Track user engagement		5 days	Mon 6/17/24 8:00 AM	Fri 6/21/24 5:00 PM	0 hrs		0%
50	<b>6.2</b>		<b>6.2 Gather user feedback</b>		<b>5 days</b>	<b>Mon 6/17/24 8:00 AM</b>	<b>Fri 6/21/24 5:00 PM</b>	<b>0 hrs</b>		<b>0%</b>
51	6.2.1		6.2.1 Implement feedback collection in app		4 days	Mon 6/17/24 8:00 AM	Thu 6/20/24 5:00 PM	0 hrs		0%
52	6.2.2		6.2.2 Analyze feedback		1 day	Fri 6/21/24 8:00 AM	Fri 6/21/24 5:00 PM	0 hrs	51	0%
53	<b>6.3</b>		<b>6.3 Improvement</b>		<b>12 days</b>	<b>Mon 6/24/24 8:00 AM</b>	<b>Tue 7/9/24 5:00 PM</b>	<b>0 hrs</b>	<b>50</b>	<b>0%</b>
54	6.3.1		6.3.1 Prioritize feature enhancements		2 days	Mon 6/24/24 8:00 AM	Tue 6/25/24 5:00 PM	0 hrs		0%
55	6.3.2		6.3.2 Release updates as needed		10 days	Wed 6/26/24 8:00 AM	Tue 7/9/24 5:00 PM	0 hrs	54	0%

# Gantt Chart



## Network Diagram



## Functional Requirements

### User Registration:

- Users should be able to register for an account with their email address or social media accounts.
- The registration process should include verifying email addresses for account activation.

### Profile Creation:

- Users should be able to create and customize their profiles with information such as name, academic interests, availability, and preferred study topics.
- Profiles should be editable to allow users to update their information as needed.

### Group Creation:

- Users should be able to create study groups based on specific subjects, courses, or topics.
- Group creators should have the ability to set privacy settings (public, private, invite-only) for their groups.

### Group Discovery:

- Users should be able to search for and discover existing study groups based on various criteria such as subject, course, location, or study schedule.
- The app should provide recommendations for relevant study groups based on user preferences and academic interests.

### Matching Algorithm:

- The app should feature a matching algorithm that analyzes user profiles and preferences to suggest compatible study partners or groups.
- Matching criteria should include academic interests, availability, study preferences, and location proximity.

### Messaging System:

- Users should be able to communicate with matched study partners or group members through an in-app messaging system.
- The messaging system should support text-based communication, file sharing, and multimedia content.

### Feedback Mechanism:

- Users should have the ability to provide feedback on study sessions or group interactions.
- The app should collect and analyze user feedback to improve the matching algorithm and overall user experience.

### Non-functional Requirements:



#### Performance:

- The app should be responsive and able to handle concurrent user interactions without significant latency.
- Loading times for user profiles, group listings, and messaging should be minimized for a smooth user experience.

#### Scalability:

- The app should be scalable to accommodate a growing user base and increasing data volume.
- Backend infrastructure should be designed to handle potential spikes in user activity during peak times.

#### Security:

- User data should be securely stored and encrypted to prevent unauthorized access or data breaches.
- The app should implement authentication mechanisms such as password hashing and session management to protect user accounts.

#### Reliability:

- The app should be reliable and available for use at all times, with minimal downtime for maintenance or updates.
- Server infrastructure should include redundancy and failover mechanisms to ensure continuous operation.

#### Accessibility:

- The app should be accessible to users with disabilities, following best practices for web and mobile accessibility standards.
- Accessibility features should include support for screen readers, keyboard navigation, and alternative text for images.

## Use Cases

### Use Case: User Registration

Actor: User

Description: User registers for an account on the study group matching app.

Preconditions: User accesses the app for the first time.

Postconditions: User receives a confirmation email for account activation.

Basic Flow:

User accesses the study group matching app for the first time.

User navigates to the registration page and fills out the required fields in the registration form.

User submits the registration form.

System validates the input data and creates a new user account in the database.

System sends a confirmation email to the user's provided email address for account activation.

### Use Case: Profile Creation

Actor: User

Description: User creates and customizes their profile with personal and academic information.

Preconditions: User is logged in to their account.

Postconditions: User's profile is created and saved in the app database.

Basic Flow:

User logs in to their account on the study group matching app.

User navigates to the profile creation section.

User provides personal and academic information in the profile creation form.

User submits the profile creation form.

System validates the input data and saves the user's profile in the app database.

### Use Case: Group Creation

Actor: User

Description: User creates a new study group.

Preconditions: User is logged in to their account.

Postconditions: Study group is created and listed in the app.

Basic Flow:

User logs in to their account on the study group matching app.

User navigates to the group creation section.

User fills out the group creation form, providing details such as group name, subject, and meeting schedule.

User submits the group creation form.

System validates the input data and creates a new study group entry in the app database.

### **Use Case: Group Discovery**

Actor: User

Description: User discovers and explores study groups available on the platform.

Preconditions: User is logged in to their account.

Postconditions: User finds and joins relevant study groups.

Basic Flow:

User logs in to their account on the study group matching app.

User navigates to the group discovery section.

User browses through the list of available study groups or uses search filters to find specific groups.

User selects a study group of interest and views its details.

User joins the selected study group if interested.

### **Use Case: Matching Algorithm**

Actor: System

Description: System matches users with compatible study partners or groups based on their profiles and preferences.

Preconditions: Users have created profiles and provided relevant information.

Postconditions: Users receive suggestions for potential study partners or groups.

Basic Flow:

System analyzes user profiles, including academic interests, availability, and study preferences.

System applies the matching algorithm to identify compatible users or groups based on shared characteristics.

System generates a list of suggested study partners or groups for each user.

Users receive notifications or recommendations for potential matches within the app.

### **Use Case: Messaging System**

Actor: User

Description: User communicates with study partners or group members within the app.

Preconditions: User is a member of a study group or has matched with a study partner.

Postconditions: User engages in discussions, shares resources, or schedules study sessions with other users.

Basic Flow:

User accesses the messaging feature within the app.

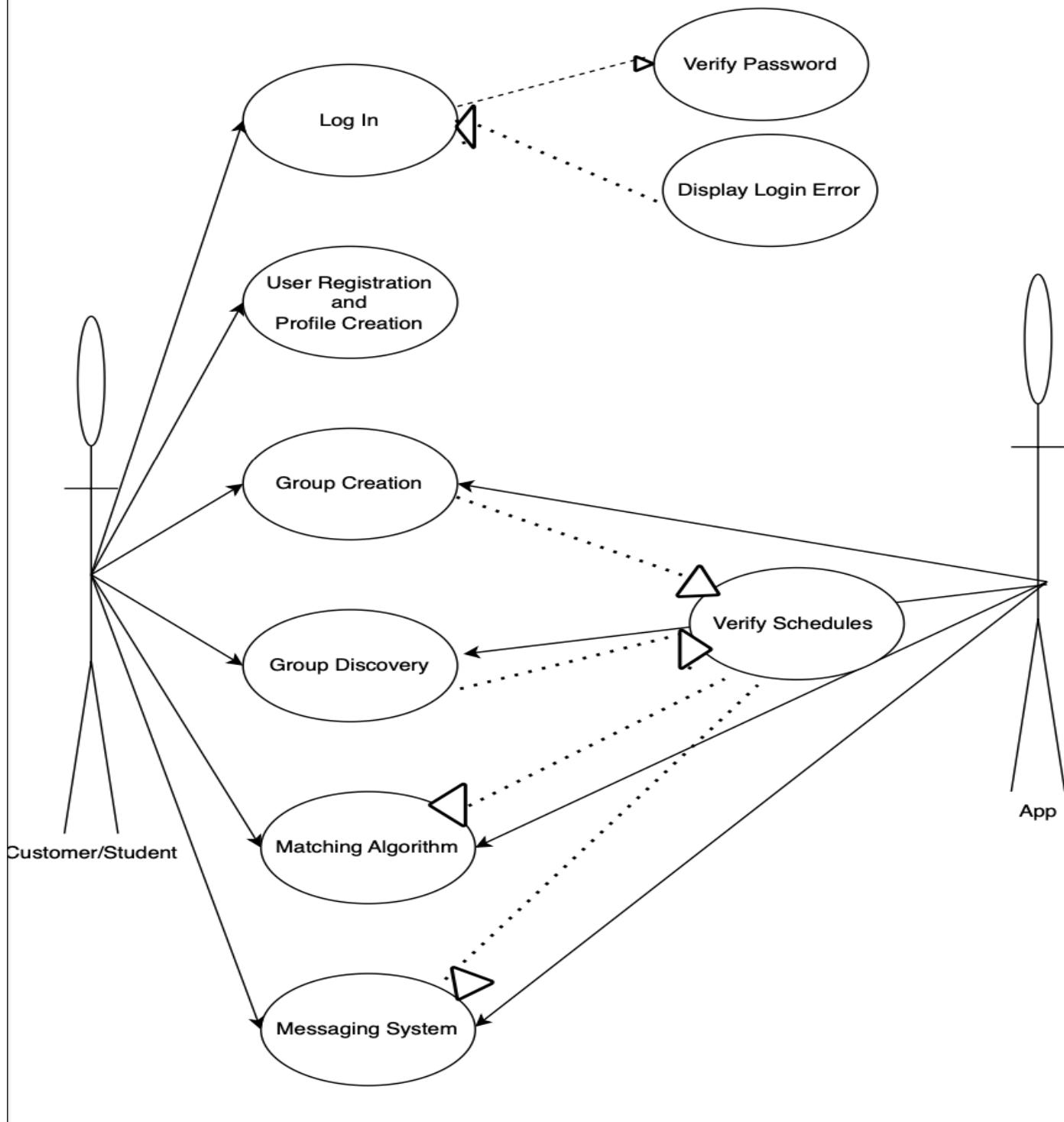
User selects a conversation with a study partner or group.

User composes a message, including text or multimedia content.

User sends the message to the selected recipient(s).

System delivers the message to the recipient(s) within the app.

Recipient(s) receive the message and can view, respond, or engage in further discussion.





## Research on Other Systems

### Tutor Ocean

TutorOcean is a website/service that pairs students with tutors for personal learning, such as study sessions, lectures, homework help, etc. It offers a marketplace where students may hire teachers for various courses, from technical things like computer programming and graphic design to academic subjects like physics, math, and languages. TutorOcean has features for scheduling, payment, video calls for online tutoring sessions, and occasionally other resources like lesson plans or progress monitoring. The difference between TutorOcean and Study Buddy is that Study Buddy's goal is more so for student-to-student use. Another aspect that may be different is that Study Buddy encourages students to meet in groups and come together as a class.

### Varsity Tutors

Varsity Tutors is an online service where students may get individualized instruction in various areas and provide alternatives for in-person and online tutoring. The type of tutoring offered covers an array of courses from regular science and math to standardized SAT and ACT preparation. Students receive personal lesson programs with carefully chosen instructors best suited to their learning preferences and objectives. The tutoring experience is heightened with interactive tools like practice tests and virtual whiteboards, and students can keep track of their progress and receive feedback. Like most other apps and services, the main difference between them and Study Buddy is the medium at which they are trying to help students

### API's

APIs function as links between various software applications. They facilitate communication, allowing for product collaboration and information sharing. Product managers may rapidly add new product features by using APIs. A type of API would be used for StudyBuddy based on scheduling. Services that specialize in scheduling are things such as Picktime, SimplyBook.me, Calendly, and Clara. Another API service that might be beneficial and helpful to StudyBuddy is a service chat feature. Online services with built-in chat features are things like Signal, WhatsApp, TextNow, and even Google Voice.



## References

Weimer, M. (2018, May 16). *The Benefits of Study Groups*. Faculty Focus. Retrieved January 26, 2024, from <https://www.facultyfocus.com/articles/course-design-ideas/what-students-can-learn-from-studying-together/>