Academic Life Management Application

Design Document

Team Number: sddedc22-06
Client: Jewels Academy
Advisors: Rachel Shannon

Team Members/Roles
Adrian Van Der Veer - Team Organization
Olusola Ogunsola - UI Design
Ben Hourigan - File Consistancy/Organization
Christopher Burgos - Client Interaction
Theron Gale - Individual Component Design

Team Email: sddec-06@iastate.edu
Team Website: http://sddec-06.sd.ece.iastate.edu/

Executive Summary

Development Standards & Practices Used

List all standard circuit, hardware, software practices used in this project. List all the Engineering standards that apply to this project that were considered.

- SCRUM / Agile
- ISO/IEC/IEEE 29119-3-2021 standards of software testing
- IEEE 12207-1996 standards for developing and managing software
- IEEE 7002-2022 standards of privacy oriented considerations in software development

Summary of Requirements

List all requirements as bullet points in brief.

- Functional Requirements
    - Access the Dashboard, Tasks List, Calendar, Mental Journal, Notepad pages and their corresponding functions
    - Internet Connectivity
    - Cellular Network Connectivity
    - GPS Connectivity
- Resource Requirements
    - Google Cloud for storage
        - Google Drive for journal entry, feelings chart, and mood chart storage
    - iOS and Android app stores for distribution
    - Finish prototype by early May
- Qualitative Requirements
    - Visually appealing in the Jewels Academy color scheme

- Speedy transitions from page to page
- Economic/Market Requirements
    - None
- Environmental Requirements
    - None
- UI Requirements
    - Easy to navigate
    - Visually appealing in the Jewels Academy color scheme
    - Compact, but open enough for easy navigation
- Other Requirements
    - Login requests shall be processed within 2 seconds. (Constraint)
    - Page transitions shall occur within 1 second. (Constraint)
    - Form submission shall occur within 5 seconds. (Constraint)
    - Google Cloud integration with a Jewels Academy account
    - Development environment setup
    - Canvas approval to get student information

Applicable Courses from Iowa State University Curriculum

List all Iowa State University courses whose contents were applicable to your project.

- Com Sci 227
    - Basic coding knowledge and general structures of code
- Com Sci 228
    - More general structures of code
- Com Sci 309
    - Large group project, potential app design and development
- Com Sci 363
    - Working with databases, SQL
- S E 319
    - Medium group project, potential app design and development
- S E 329
    - Project Management Strategies
- S E 422X
    - Cloud Databasing

New Skills/Knowledge acquired that was not taught in courses

List all new skills/knowledge that your team acquired which was not part of your Iowa State curriculum in order to complete this project.

- Coding in Python
- Coding in React Native
- Cloud Database Management
- Cloud Connector Setup
- Development Environment Setup
- Data Security

Table of Contents

List of figures/tables/symbols/definitions (This should be the similar to the project plan)

Figures:

1. Initial App Design Prototype
2. Detailed Project Plan Schedule
3. Gantt Chart for our Project Plan
4. Annotated Gantt Chart
5. Empathy Map
6. Lotus Blossom Map
7. Use Case Diagram
8. Example User Persona
9. Early Map for Potential Technologies and APIs
10. General List of Modules and Ideas
11. Overall UI Design
12. Page Connectivity
13. Page Examples
14. Page Overlay Examples
15. Interation Cycle

Tables:

1. Risk Management Breakdown
2. Impact Breakdown
3. Work Hours Breakdown
4. Similar Apps Benchmarking
5. Areas of Responsibility Breakdown

1 Team

1.1 Team Members

- Adrian Van Der Veer
- Olusola Ogunsola
- Ben Hourigan
- Christopher Burgos
- Theron Gale

1.2 Required Skill Sets for Your Project

- Mobile App Development
- UI/UX Development
- Requirements Engineering
- Database management
- Graphic design

1.3 Skill Sets covered by the Team

- Frontend Development - Theron, Ben, Chris
- Backend Development - Adrian, Ben, Sola, Chris
- UI/UX Development - Theron, Sola
- Database Management - Adrian, Ben, Theron

1.4 Project Management Style Adopted by the team

- Agile/SCRUM

1.5 Initial Project Management Roles

- Ben: File Consistency/Organization
- Adrian: Team Organization
- Theron: Individual Component Design
- Chris: Client interaction
- Sola: UI design

2 Introduction

2.1 Problem Statement

For many students, school is a stressful time. There are dozens of assignments to keep track of, tests to study for, sports and other extracurriculars to participate in, and much more. This can cause many issues in their day-to-day lives. Our group wants to create an application that will allow students with many responsibilities from school, to work, to social activities to manage their time in a visual and easy-to-use way and to keep tabs on their own mental health.

2.2 Requirements & Constraints

- Functional Requirements
    - User should be able to:
        - Access the Dashboard, Tasks List, Calendar, Mental Journal, Notepad pages
            - These are all interconnected
        - Access Canvas through the app
        - Add/Set/Remove tasks
        - Adjust timescale for the tasks list
        - View calendar
        - Adjust timescale for the calendar
        - Filter which tasks, events, etc. appear on the calendar
        - Add notes (typed & handwritten)
        - View notes
        - Add journal entry
    - View journal entries
    - Emote daily feelings
    - View feelings/mood charts
    - Adjust meditation/break timer
    - Login/Logout/Register
    - Adjust account settings
    - Adjust notifications settings
    - Internet Connectivity
    - Cellular Network Connectivity
    - GPS Connectivity
- Resource Requirements
    - Google Cloud for storage

- Google Drive for journal entry, feelings chart, and mood chart storage
  - iOS and Android app stores for distribution
  - Finish prototype by early May
- Qualitative Requirements
  - Visually appealing in the Jewels Academy color scheme
  - Speedy transitions from page to page
- Economic/Market Requirements
  - None
- Environmental Requirements
  - None
- UI Requirements
  - Easy to navigate
  - Visually appealing in the Jewels Academy color scheme
  - Compact, but open enough for easy navigation
- Other Requirements
  - Login requests shall be processed within 2 seconds. (Constraint)
  - Page transitions shall occur within 1 second. (Constraint)
  - Form submission shall occur within 5 seconds. (Constraint)

2.3 ENGINEERING STANDARDS

ISO/IEC/IEEE 29119-3-2021 standards of software testing

- Standard specification for testing software. This standard is useful for our project because it's important for us to test our application and ensure its reliability before it is deployed to it's primary users.

IEEE 12207-1996 standards for developing and managing software

- This standard provides industry best practices for developing and maintaining software. This is important for our project because we'll need to be able to effectively communicate with Jewels Academy on the best way to maintain the application after we're no longer associated with Iowa State.

IEEE 7002-2022 standards of privacy oriented considerations in software development

- This standard describes the requirements for software that include "privacy-oriented considerations regarding products, services, and systems utilizing employee, customer, or other external user's personal data are defined by this standard", and as our application will require the storage of user data for login and user verification this standard applies to the privacy of that data.

2.4 INTENDED USERS AND USES

The primary beneficiary for this project will be students attending the Jewels Academy organization. Students will be able to use the newly created mobile application to manage their daily lives by integrating their daily tasks, academics, and calendar into one app. With the Jewels Academy Time Management mobile app, students will be able to add, set, and edit tasks as well as enabling task notifications. Students can use the app's calendar in order to integrate canvas, display their daily routines, and plan other life events. And another use case for the app is the implementation of a mental health journal so that students will be able to track their mental health status over time.

3 Project Plan

3.1 PROJECT MANAGEMENT/TRACKING PROCEDURES

Our group decided to use the Agile development management style, because it is the most adaptable to a relatively complex software task, it allows the group to keep updating continually so when there are things the group doesn't like we can easily change them as very little is dependant on the new content each sprint compared to a waterfall approach where replacement of unwanted parts is much more difficult.

Our group intends to use a mixture of Trello, Git issues, Git, for development and planning and discord for communication among the team as well as webex and email for communication to those outside the team.

3.2 Task Decomposition

Our group intends to create 7 total pages with various internal functionalities within them, we will have a dashboard page which will be the landing page when a user logs in to the application, a task list which will show a breakdown of tasks by day, a calendar which will show a monthly or weekly overview of tasks and events, a mental health journal where users can rate their day, and an account settings page. Each page will have various sub-features broken down below. An example of the project is also provided below:

[Figure 1]: Initial App Design Prototype

1.1 Dashboard

1.1.1 General Information
1.1.2 News

1.1.3 Reminders
1.1.4 Positive Quotes
1.1.5 Weather

1.2 Calendar

1.2.1 Calendar
    1.2.1.1 Change Timescale (Daily, Weekly, Monthly)
    1.2.1.2 Include Holidays
1.2.2 Events
    1.2.2.1 Filter Results
1.2.3 Calendar Integration (Outlook, Google, etc.)

1.3 Notes / Reminders

1.3.1 Hand Written Notes
1.3.2 Typed Notes
1.3.3 Create Reminder
1.3.4 Cloud Drive Integration (Google Drive, Microsoft OneDrive, etc.)

1.4 Journal / Mental Health

1.4.1 Diary / Journal
    1.4.1.1 Color Day Rating
    1.4.1.2 Emoticon / Emoji Day Rating
1.4.2 Breaktime / Meditation Timer

1.5 Task List / To-Do List

1.5.1 Add Task
1.5.2 Set Goal / Repeated Event
1.5.3 To-Do List
    1.5.3.1 Change Timescale
1.5.4 Goal Progression Reports
1.5.5 Focus Timer

1.6 Class Assignments

1.6.1 Canvas Integration
       1.6.1.1 Homework Due Dates
       1.6.1.2 Upcoming Assignments
       1.6.1.3 Class Grades
1.6.2 Link to 1.5.3 and 1.2.1

1.7 Account Settings

1.7.1 User Profile
1.7.2 Login / Register New User
       1.7.2.1 Logout
       1.7.2.2 Setup Two-Factor Authentication
       1.7.2.3 Recover Username / Email / Password
       1.7.2.4 Setup Federated Identity Login (Google, Facebook, etc)
1.7.3 Change Display Theme (Dark, Light, etc.)
1.7.4 Change Notification Settings

3.3 PROJECT PROPOSED MILESTONES, METRICS, AND EVALUATION CRITERIA

MileStones:

- Dashboard Page Complete
- Settings Page Complete
- Calendar Page Complete
- Mental Health Page Complete
- To-Do Page Complete
- Canvas Page Complete
- UI complete
- Prototype Complete

We will use the metrics of, response time which we want to keep to a minimum, loading speed which we want to be as fast as possible, and power draw which we want as low as possible for a mobile application.

3.4 Project Timeline/Schedule

• A realistic, well-planned schedule is an essential component of every well-planned project

• Most scheduling errors occur as the result of either not properly identifying all of the necessary activities (tasks and/or subtasks) or not properly estimating the amount of effort required to correctly complete the activity

• A detailed schedule is needed as a part of the plan:

	Task Name	Parent Name	Parent URL		Status	Due Date
3	Dashboard				TO DO	12/30/2022
4	General Information	Dashboard	https://app.clickup.com/t/2hhgtu8		TO DO	03/29/2022
5	Positive Quotes	Dashboard	https://app.clickup.com/t/2hhgtu8		TO DO	09/20/2022
6	Calendar				TO DO	11/30/2022
7	Weather	Dashboard	https://app.clickup.com/t/2hhgtu8		TO DO	09/20/2022
8	Change Timescale (Daily, Weekly, Monthly)	Calendar	https://app.clickup.com/t/2hhgtv0		TO DO	04/15/2022
9	Include Holidays	Calendar	https://app.clickup.com/t/2hhgtv0		TO DO	09/20/2022
10	Events	Calendar	https://app.clickup.com/t/2hhgtv0		TO DO	03/29/2022
11	Filter Results	Calendar	https://app.clickup.com/t/2hhgtv0		TO DO	08/30/2022
12	Integrate External Calendars	Calendar	https://app.clickup.com/t/2hhgtv0		TO DO	09/30/2022
13	Notes /Reminders				TO DO	11/30/2022
14	Hand Written Notes	Notes /Reminders	https://app.clickup.com/t/2hhgtxy		TO DO	10/20/2022
15	Typed Notes	Notes /Reminders	https://app.clickup.com/t/2hhgtxy		TO DO	10/20/2022
16	Create Reminders	Notes /Reminders	https://app.clickup.com/t/2hhgtxy		TO DO	04/25/2022
17	Cloud Drive Integration	Notes /Reminders	https://app.clickup.com/t/2hhgtxy		TO DO	10/15/2022
18	Journal /Mental Health				TO DO	11/30/2022
19	Diary /Journal	Journal /Mental Health	https://app.clickup.com/t/2hhgtze		TO DO	09/15/2022
20	Color Day Rating	Journal /Mental Health	https://app.clickup.com/t/2hhgtze		TO DO	09/30/2022
21	Emoji Day Rating	Journal /Mental Health	https://app.clickup.com/t/2hhgtze		TO DO	10/10/2022
22	Breaktime /Meditation Timer	Journal /Mental Health	https://app.clickup.com/t/2hhgtze		TO DO	11/01/2022
23	Task List /To-Do List				TO DO	11/30/2022
24	Add Task	Task List /To-Do List	https://app.clickup.com/t/2hhgu3z		TO DO	04/20/2022
25	Set Goal /Repeated Event	Task List /To-Do List	https://app.clickup.com/t/2hhgu3z		TO DO	04/25/2022
26	To-Do List	Task List /To-Do List	https://app.clickup.com/t/2hhgu3z		TO DO	04/25/2022
27	Change Timescale	Task List /To-Do List	https://app.clickup.com/t/2hhgu3z		TO DO	04/30/2022
28	Goal Progression Reports	Task List /To-Do List	https://app.clickup.com/t/2hhgu3z		TO DO	05/05/2022
29	Focus Timer	Task List /To-Do List	https://app.clickup.com/t/2hhgu3z		TO DO	09/10/2022
30	Class Asignments	Task List /To-Do List	https://app.clickup.com/t/2hhgu3z		TO DO	11/30/2022
31	Integrate Canvas	Link Canvas to Calendar and Task List	https://app.clickup.com/t/2hhgu8n		TO DO	04/15/2022
32	Link Canvas to Calendar and Task List				TO DO	11/30/2022
33	Homework Due Dates	Link Canvas to Calendar and Task List	https://app.clickup.com/t/2hhgu8n		TO DO	04/15/2022
34	Upcoming Assignments	Link Canvas to Calendar and Task List	https://app.clickup.com/t/2hhgu8n		TO DO	05/15/2022
35	Class Grades	Link Canvas to Calendar and Task List	https://app.clickup.com/t/2hhgu8n		TO DO	04/15/2022
36	Account Settings				TO DO	11/30/2022
37	Reminders	Dashboard	https://app.clickup.com/t/2hhgtu8		TO DO	03/29/2022
38	User Profile	Account Settings	https://app.clickup.com/t/2hhgu8w		TO DO	04/10/2022
39	Login /Register User	Account Settings	https://app.clickup.com/t/2hhgu8w		TO DO	04/05/2022
40	Logout Button	Account Settings	https://app.clickup.com/t/2hhgu8w		TO DO	04/05/2022
41	Two Factor Authentification	Account Settings	https://app.clickup.com/t/2hhgu8w		TO DO	11/10/2022
42	Recover Username/ Password	Account Settings	https://app.clickup.com/t/2hhgu8w		TO DO	11/15/2022
43	Setup Federated Identity	Account Settings	https://app.clickup.com/t/2hhgu8w		TO DO	04/25/2022
44	Change Display Theme	Account Settings	https://app.clickup.com/t/2hhgu8w		TO DO	11/25/2022
45	Change Notification Settings	Account Settings	https://app.clickup.com/t/2hhgu8w		TO DO	08/31/2022
46	News	Dashboard	https://app.clickup.com/t/2hhgtu8		TO DO	09/10/2022
47	Calendar	Calendar	https://app.clickup.com/t/2hhgtv0		TO DO	03/29/2022

[Figure 2]: Project Plan Schedule

– **Start with a Gantt chart showing the tasks (that you developed in 2.2) and associated subtasks versus the proposed project calendar. The Gantt chart shall be referenced and summarized in the text.**

[Figure 3]: Gantt Chart of our Project Plan

• Annotate the Gantt chart with when each project deliverable will be delivered

		ASSIGNEE	DUE DATE	PRIORITY
List ℹ + NEW TASK				✓ SHOW CLOSED
TO DO 7 TASKS				
▾ ✓ Dashboard ⟲ 5 +			Dec 30	⚑
✓ General Information			Mar 29	⚑
✓ News			Sep 10	⚑
✓ Positive Quotes			Sep 20	⚑
✓ Weather			Sep 20	⚑
✓ Reminders			Mar 29	⚑
▾ ✓ Calendar ⟲ 6 +			Nov 30	⚑
✓ Calendar			Mar 29	⚑
✓ Change Timescale (Daily, Weekly, Monthly)			Apr 15	⚑
✓ Include Holidays			Sep 20	⚑
✓ Events			Mar 29	⚑
✓ Filter Results			Aug 30	⚑
✓ Integrate External Calendars			Sep 30	⚑
▾ ✓ Notes /Reminders ⟲ 4 +			Nov 30	⚑
✓ Hand Written Notes			Oct 20	⚑
✓ Typed Notes			Oct 20	⚑
✓ Create Reminders			Mon	⚑
✓ Cloud Drive Integration			Oct 15	⚑

		ASSIGNEE	DUE DATE	PRIORITY
✓ Cloud Drive Integration			Oct 15	⚑
▾ ✓ Journal /Mental Health ⟲ 4 +			Nov 30	⚑
✓ Diary /Journal			Sep 15	⚑
✓ Color Day Rating			Sep 30	⚑
✓ Emoji Day Rating			Oct 10	⚑
✓ Breaktime /Meditation Timer			Nov 1	⚑
▾ ✓ Task List /To-Do List ⟲ 7 +			Nov 30	⚑
✓ Add Task			3 days ago	⚑
✓ Set Goal /Repeated Event			Mon	⚑
✓ To-Do List			Mon	⚑
✓ Change Timescale			Apr 30	⚑
✓ Goal Progression Reports			May 5	⚑
✓ Focus Timer			Sep 10	⚑
✓ Class Asignments			Nov 30	⚑
▾ ✓ Link Canvas to Calendar and Task List ⟲ 4 +			Nov 30	⚑
✓ Integrate Canvas			Apr 15	⚑

		ASSIGNEE	DUE DATE	PRIORITY
✓ Upcoming Assignments			May 15	⚑
✓ Class Grades			Apr 15	⚑
▾ ✓ Account Settings ⟲ 8 +			Nov 30	⚑
✓ User Profile			Apr 10	⚑
✓ Login /Register User			Apr 5	⚑
✓ Logout Button			Apr 5	⚑
✓ Two Factor Authentification			Nov 10	⚑
✓ Recover Username/ Password			Nov 15	⚑
✓ Setup Federated Identity			Mon	⚑
✓ Change Display Theme			Nov 25	⚑
✓ Change Notification Settings			Aug 31	⚑

[Figure 4]: Annotated Gantt Chart

For a more detailed view of the task list[1]

For a more detailed view of the gantt chart[2]

• Project schedule/Gantt chart can be adapted to Agile or Waterfall development model. For agile, a sprint schedule with specific technical milestones / requirements / targets will work.

3.5 Risks And Risk Management/Mitigation

Our group doesn't have much risk regarding hardware or outside factors, our risks are mostly internal regarding communication and internal software bugs. Our group can mitigate this by having multiple lines of communication including discord, email, and in person meetings as well as using a CI/CD software with useful testing to ensure the software is reliable and relatively bug free. Each push to our collective git will have to be integration tested to lower code conflicts, and each sprint our group will hold a planning meeting that will include time for us to share any faults our members have found so they can be resolved.

[Table 1]: Risk Management Breakdown

Risks	Category	Probability	Impact	RMP
Login Security	Technical	45%	2	
New Group	Operational	80%	2	Try and work synergistically with each other to minimize any potential conflicts that arise
Communication Issues	Schedule	35%	2	
GPS/Mapping Issues	Technical	30%	3	
Page Access Speed	Technical	40%	3	
Team Knowledge	Operational	60%	2	Keep each other up to date on what has been learned, as many of us have little to no experience with some of the project's technologies.
Program Task Exporting	Technical	45%	2	
App will not Reach Expectations in Time	Schedule	33%	1	

Unstable Project Scope	Schedule / Operational	40%	1	

Impact Values:

      1 - Catastrophic   3 - Marginal

      2 - Critical   4 - Negligible

3.6 Personnel Effort Requirements

      Given the teams gereral unfamiliarity with the proposed software and development environment, including such factors as new technology in the case of cloud hosting, a new IDE in the case of Intellij Idea, and new lagngages in the case of Python and React Native, the development time of this application may exceed the average of normal application development time for an application of this scope. Our group took into account the unfamliarity of this project, and came up with the following estimates of how long on average it would take the group members to complete each of the necessary tasks based on their previous experience with software development, and estimates of time to become comfortable with the unfamiliar design components. Below is an estimate of the projected work hours for the development of this application.

[Table 2]: Work Hours Breakdown

Task Name	Projected Effort (work hours)
4.1 Dashboard	50
4.1.1 General Information	10
4.1.2 News	5
4.1.3 Reminders	15
4.1.4 Positive Quotes	10
4.1.5 Weather	10

4.2 Calendar	30
4.2.1 Calendar	5
4.2.1.1 Change Timescale (Daily, Weekly, Monthly)	5
4.2.1.2 Include Holidays	5
4.2.2 Events	5
4.2.2.1 Filter Results	5
4.2.3 Calendar Integration (Outlook, Google, etc.)	5
4.3 Notes / Reminders	20
4.3.1 Hand Written Notes	5
4.3.2 Typed Notes	5
4.3.3 Create Reminder	5
4.3.4 Cloud Drive Integration (Google Drive, Microsoft OneDrive, etc.)	5
4.4 Journal / Mental Health	30
4.4.1 Diary / Journal	15
4.4.1.1 Color Day Rating	5
4.4.1.2 Emoticon / Emoji Day Rating	5
4.4.2 Breaktime / Meditation Timer	5
4.5 Task List / To-Do List	30
4.5.1 Add Task	5
4.5.2 Set Goal / Repeated Event	5
4.5.3 To-Do List	5

4.5.3.1 Change Timescale	5
4.5.4 Goal Progression Reports	5
4.5.5 Focus Timer	5
4.6 Class Assignments	40
4.6.1 Canvas Integration	10
4.6.1.1 Homework Due Dates	5
4.6.1.2 Upcoming Assignments	5
4.6.1.3 Class Grades	10
4.6.2 Link to 4.5.3 and 4.2.1	10
4.7 Account Settings	40
4.7.1 User Profile	5
4.7.2 Login / Register New User	5
4.7.2.1 Logout	5
4.7.2.2 Setup Two-Factor Authentication	5
4.7.2.3 Recover Username / Email / Password	5
4.7.2.4 Setup Federated Identity Login (Google, Facebook, etc)	5
4.7.3 Change Display Theme (Dark, Light, etc.)	5
4.7.4 Change Notification Settings	5
Total Work Hours	**240**

3.7 Other Resource Requirements

- Google Cloud integration with a Jewels Academy account

- Development environment setup

- Canvas approval to get student information

4 Design

4.1 DESIGN CONTEXT

4.1.1 Broader Context

Our project seeks to serve female students from challenging economic areas with a tool to make organizing the busy lives and schedules of students easier and giving them methods to assess and better manage their mental health as well as tools to keep track of upcoming activities.

[Table 3]: Impacts Breakdown

Area	Description	Examples
Public health, safety, and welfare	How does your project affect the general well-being of various stakeholder groups? These groups may be direct users or may be indirectly affected (e.g., solution is implemented in their communities)	- Possible improved test scores by encouraging more effective study skills using cognitive psychology research - Better understanding and management of stress to improve health and quality of life - Improves time management to reduce undue stress and ensure important activities are done in the correct time frame.
Global, cultural, and social	How well does your project reflect the values, practices, and aims of the cultural groups it affects? Groups may include but are not limited to specific communities, nations, professions, workplaces, and ethnic cultures.	Development or operation of the solution would violate a profession's code of ethics, implementation of the solution would require an undesired change in community practices

Environmental	What environmental impact might your project have? This can include indirect effects, such as deforestation or unsustainable practices related to materials manufacture or procurement.	• Our group is using Google Cloud which is carbon-neutral, and dedicated to being even more environmentally friendly.
Economic	What economic impact might your project have? This can include the financial viability of your product within your team or company, cost to consumers, or broader economic effects on communities, markets, nations, and other groups.	• This project at its inception should be free to download so as not to cause monetary strain on users who come from challenging economic circumstances • This application should consistently cost a small enough amount for usage that it does not unduly pressure the economic resources of the user base

4.1.2 User Needs

As our users are only students and no other groups, there are only 2 user needs:

- Users need a way to easily manage activities including school, classwork, extra-curricular activities, and work because students especially from low income circumstances have a considerable amount of things to do each day, and without a way to schedule all of that, some activities or assignments may be missed or fall to the wayside.
- Users need a way to manage their stress levels and improve their overall mental health, because stress and poor mental health are incredibly detrimental to overall health, and efficiency and quality of work and schooling.

4.1.3 Prior Work/Solutions

There are some already existing products for scheduling, timekeeping, and time management including the various major calendar applications like google or outlook, as well as applications to assist in mental health such as meditation applications or focus timers such as Pomodoro.

[Table 4]: Similar Apps Benchmarking

Product Name	Product Features	Integration
Google Calendar	- Variable Timeframes - Schedule Goals (scheduled for the user with preferences) - Schedule reminders - Schedule events in advance - Schedule a Task to be completed - Show to do list with tasks, holidays, reminders, and events - Light/Dark Theme - Simple Design - Easy to use	Has API Integration for - Javascript - Java .NET - Node.js - PHP - Python - GO - Android - IOS
Trello	- Task lists (To do, doing, done) - Simple UI - Search feature - Notifications - Workspaces/ Team workspaces - Upload attachments - Scan documents - Comments - Similar to GIT issues, however, there's no code integration. The app's purpose is strictly for task management and organization	API Integration: - cURL - Node.js - Java - Python - PHP
MyState	- Dashboard - Weather - Schedule - Busses - Map - Events - Easy to use - Simple UI	API Integration: Getting in contact with MyState department
OutLook	- Calendar - Meeting reminders - Schedule recurring events	Supported languages for API integration: Node.js

	• Create groups to collaborate with • Communication with microsoft teams for online meetings • Application for taking notes • Email communication • Create to-do lists	• PHP • Python • Ruby • Objective-C • Java • C# (.NET) • cURL
Canvas	• Schedule Goals (both scheduled and not scheduled by user) • Schedule reminders/events • To-Do list view (mobile only) • Calendar and day-to-day views • Course Separation • Each course has its own page & modules • Grouping System • Inbox System • File Upload/Download System (Canvas Studio) • Personalized Accounts System • Grades System • Clean and Mostly Simple Design • Mostly easy to use	Has its own API to integrate Shown integration with: • Node.js • Javascript • Python

4.1.4 Technical Complexity

        The technical complexity of the project is sufficient because of the up-to-date frameworks being used and modern technological standards being implemented. All of the technologies being used in this project are industry standard and meet all of the complexity guidelines currently being implemented in all modern applications.

Relevant list of frameworks and technologies:

    a. Intellij IDE
        i. Modern up to date IDE going to be used for backend and frontend development
    b. Figma
        i. Used for front end design and showcasing page navigation
    c. React Native

       i.   JavaScript Framework used by many modern applications to build stunning UI and easy to navigate apps

d. Python
       i.   Used for backend design and API calling

e. Google Cloud
       i.   Will be used by the backend for hosting our application. Cloud services are what many applications are moving to and away from physical servers.

4.2 DESIGN EXPLORATION

4.2.1 Design Decisions

- Our team decided to use a Google Cloud backend proposing a SaaS solution to the given project
- Our group will be creating a login system including storage of user information as well as federated login with major providers like Google, Facebook, etc.
- Our group has decided to use react native as the development language, Gitlab for version control, and Gitlab CI/CD for integration and testing of proposed changes to the code-base.

4.2.2 Ideation

Our group used brainstorming, user profiles, use case diagrams, as well as the lotus blossom technique and an empathy map.

- Figure 5 is our empathy map that we used to ideate some early options for our app.
- Figure 6 is our lotus blossom technique map that we also used to ideate some early options for our app.
- Figure 7 is our use case diagram to show what all needs to be included in our app.
- Figure 8 is an example user persona for an ideal user for our app.
- Figure 9 is an early visual map for designing our app, including potential technologies and APIs.
- Figure 10 is a general list of ideas and modules for each part of our app.

Think and feel?

Want a way to schedule without hassle

Easy communication

Difficult to use and impractical scheduling apps

Hear?

Teachers recommendations/requirements

Everyone's input

Students

Many calendar apps without canvas integration

See?

Likes apps to be convenient

Likes easy to user UI/UX

Say and do?

Pain — The app will be slow and unresponsive.

Gain — The app will be adopted by the entire JA

[Figure 5]: Empathy Map

[Figure 6]: Lotus Blossom Map
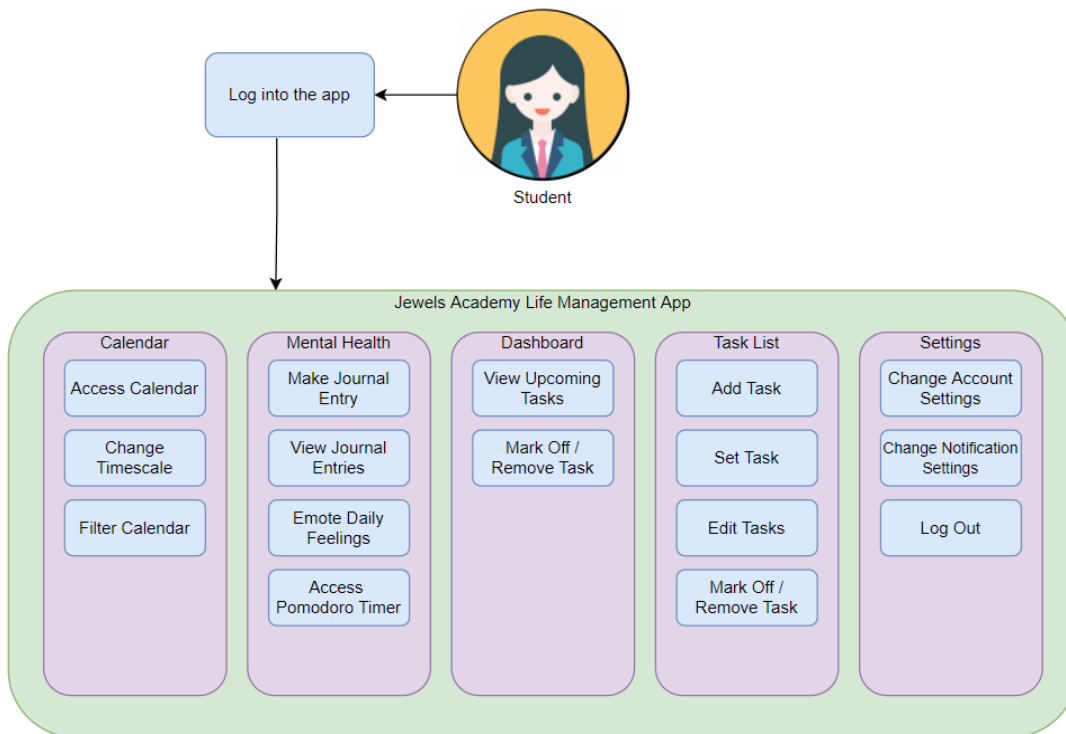
[Figure 7]: Use Case Diagram
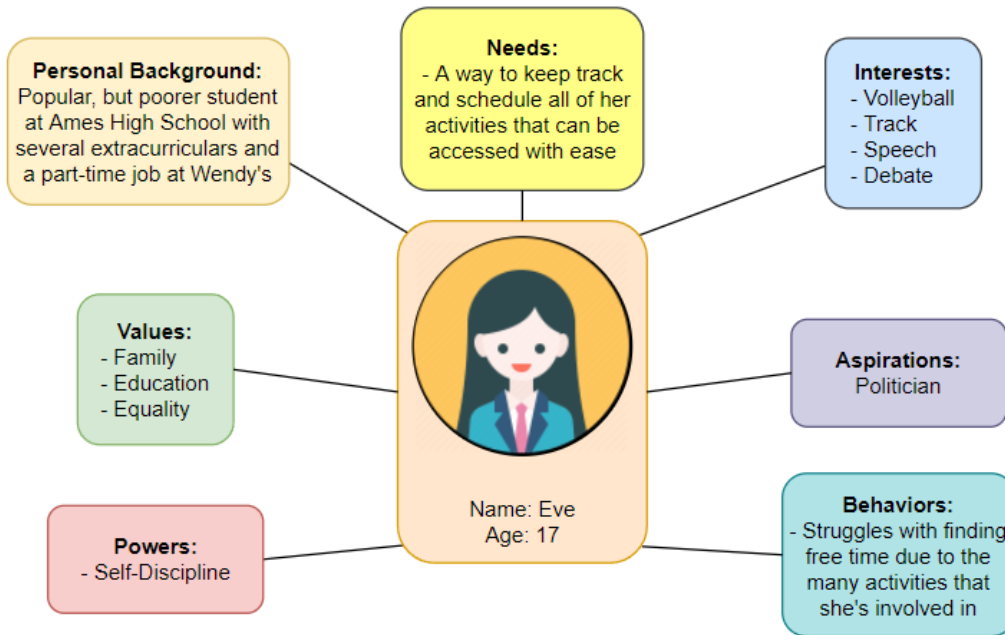
[Figure 8]: Example User Persona
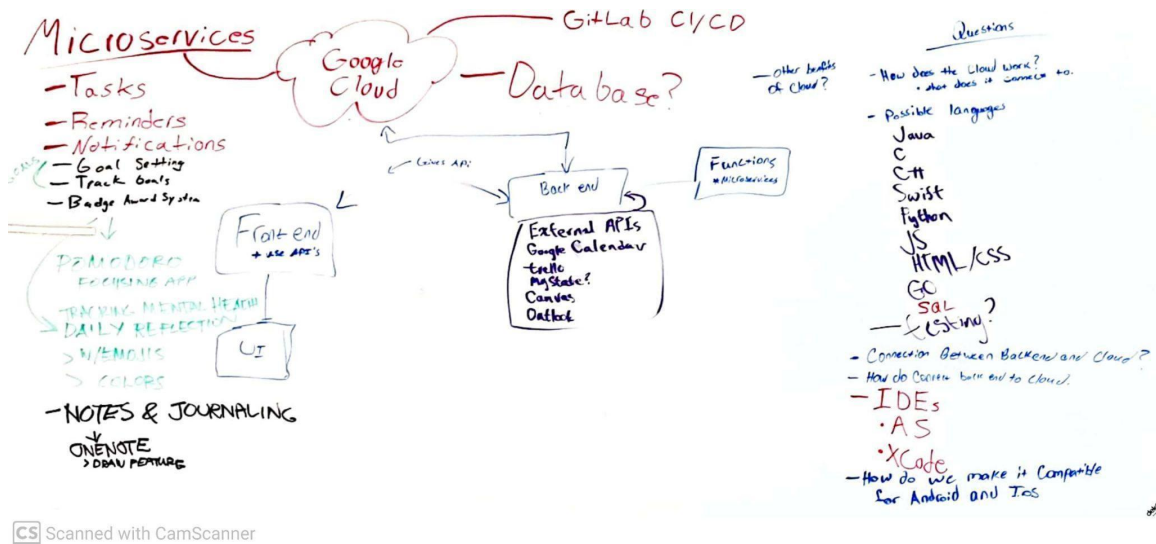
[Figure 9]: Early Map for Potential Technologies and APIs
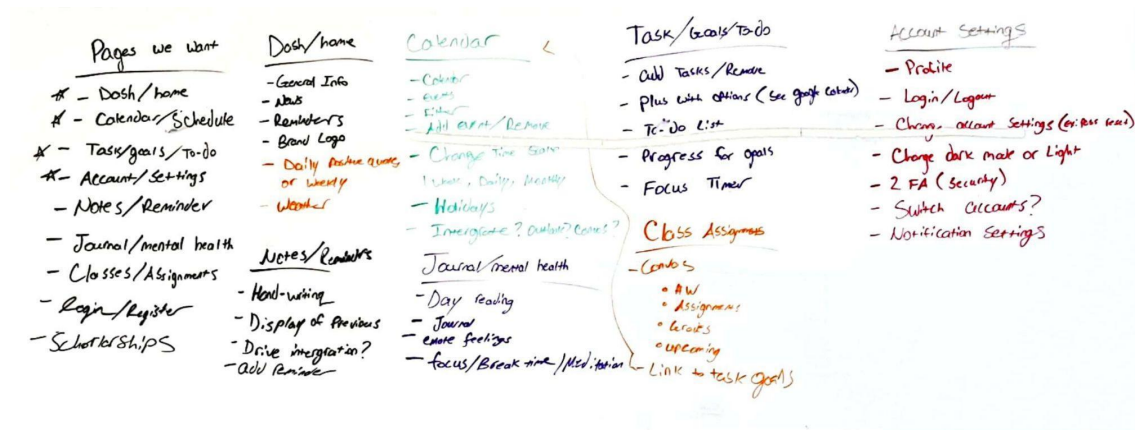
[Figure 10]: General List of Modules and Ideas

4.2.3 Decision-Making and Trade-Off

Our group went through several iterations of design descriptions within the team, including our faculty advisor and client, for the refinement of ideas into a true framework of the final project. We designed multiple iterations of prototypes using the current ideas of the project and refined down the things that we felt did not fit well within the scope of the project, and added in other functionalities given user needs, and stakeholder interests in features that would be good additions.

4.3 PROPOSED DESIGN

4.3.1 Design Visual and Description

Overall Design UI
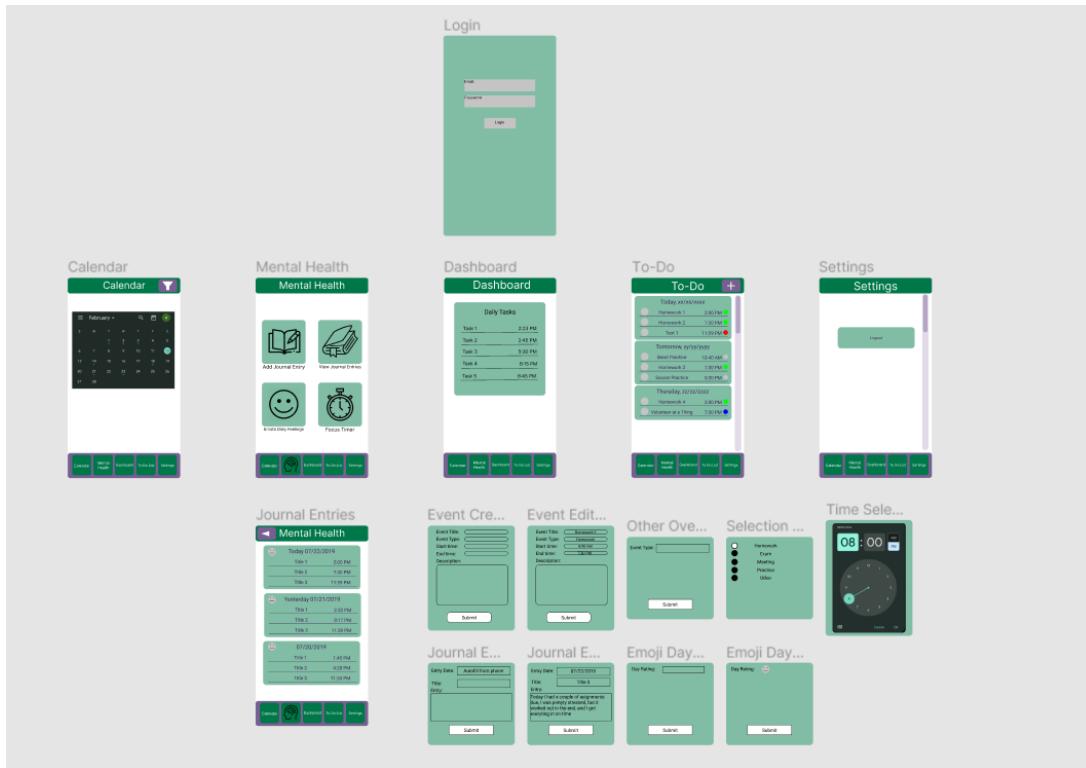
[Figure 11]: Overall Design UI

Design - Layout of Page Connectivity
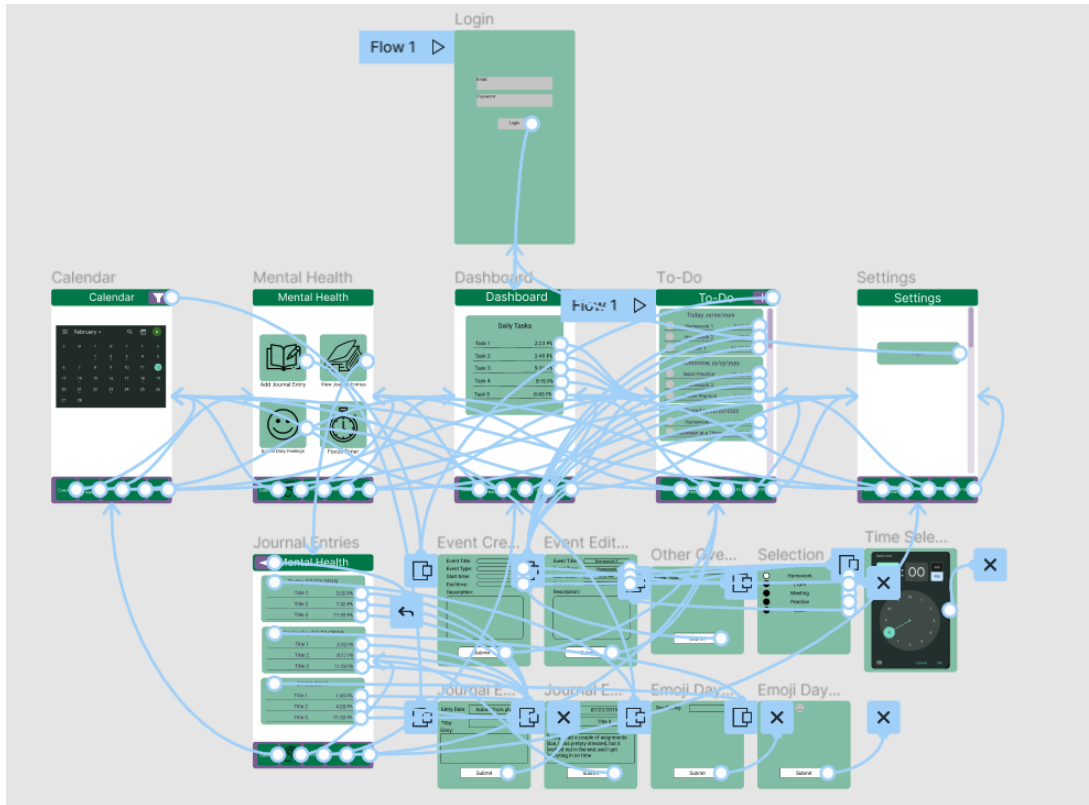
[Figure 12]: Page Connectivity

Design - Page Examples

[Figure 13]: Page Examples

Design - Page Overlay Examples

[Figure 14]: Page Overlay Examples

4.3.2 Functionality

Our current design is intended to operate as an academic life management application. The goal is to help students manage stress from academics through creating tasks/homework reminders as well as improve mental health by providing motivational quotes and goal tracking. Our design is aimed to be used so the user has less buttons to push to create their goals and assignments. We want the application to be easy to use, easy to track, and ultimately a pleasurable experience for the user. The current layout navigation and provisional design of the application can be seen above.

4.3.3 Areas of Concern and Development

One of our primary concerns for delivering our product is that there is a misunderstanding of the requirements between our team, our client, and our users. The main goal of our project is to produce an application that will be used frequently by its users and improve the quality of their everyday lives. As long as our client as well as end users are happy with the product that we deliver, our team will feel like we succeeded in our efforts.

Because our team has meticulously worked through the planning, requirements, and design process for our project, the next immediate plan is to move forward with the execution and development of our application. This includes setting up our CI/CD, setting up an IDE environment, and beginning the process of coding both the frontend and backend components. Throughout the development process, we'll be communicating with our client and faculty adviser in order to address any questions or concerns that arise.

4.4 TECHNOLOGY CONSIDERATIONS

[Table 4]: Similar Apps Benchmarking

Technology	Strengths	Weaknesses	Alternative Technologies
IntelliJ Idea IDE	- High functionality - Able to develop in multiple languages - Albe to run and test many languages	- High Complexity - Power intensive - CPU intensive	- Visual studio code - Android Studio - Xcode

	• Highly modifiable • Has built in GIT functionality		
React Native	• Fast execution for web and mobile applications • Multifunctional • Works well with Python	• Unfamiliar to the group • High learning curve	• Java • C# • Swift • Ionic • Xamarin • JavaScript
Python	• Easy to learn • High functionality • Low complexity • Usable with many APIs including the ones necessary for this project • Usable with React Native and Google Cloud	• Reletively unfamiliar to the group • Low memory management • Very little internal control over the language • Reletively slow in comparison to some other languages	• Java • C# • C++ • C • Ruby • GO • Javascript •
Google Cloud	• Low latency • High uptime • Automatic backup • High functionality • Extremely Adaptable • Automatinc database management • Automatic patching • Automatic updates	• Unfamiliar to the group • External (group has little internal control) • Difficult pricing models • Overwhelming documentation • Difficult to influence with outside applications	• Microsoft Azure • AWS • MySQL • SQL Server • Microsoft Access • ISU Servers
Gitlab	• Easy to use • Linked to ISU • Easy to use visual GIT UI • Automatic record keeping for metrics/contributions • Inclues task list • Includes CI/CD for integration testing	• N/A	• Github • Git console

4.5 DESIGN ANALYSIS

– Did your proposed design from 3.3 work? Why or why not?

That will be discovered in the second semester, as the app in its entirety has not been fully created in code. Although the design in of itself should not change much negatively over the course of its creation.

– What are your observations, thoughts, and ideas to modify or iterate over the design?

We plan to create all that we have shown in the original proposed design first, starting with basic connectivity between screens and progressing to data storage and usage. Once all aspects of the original design are fulfilled, we plan to add on new features on top of the existing design, such as more settings and other bits of information on the dashboard. We will also refine the design aspects of the proposed design as to make it more visually appealing as development continues, and the group gets a firmer and more concrete view of the UI as it is created in software. The functionality will also undergo iteration to increase the speed and efficiency of the application as development continues and issues develop. This development will constantly undergo iteration as it is developed in line with AGILE development practice including the creation of new features.

4.6 DESIGN PLAN

We plan to design starting from the most vital features for the user experience first beginning with the dashboard and login, then moving on to the other main pages including the task list, calendar, mental health page, and user settings last. From these main pages we can further develop into more complexity and functionality, however the plan is to first have a skeletal prototype that can be expanded into a more fully functional application as time progresses.

5 Testing

Testing is an extremely important component of most projects, whether it involves a circuit, a process, power system, or software.

The testing plan should connect the requirements and the design to the adopting test strategy and instruments. In this overarching introduction,

given an overview of the testing strategy. Emphasize any unique challenges to testing for your system/design.

5.1 Unit Testing

What units are being tested? How? Tools?

Our units for this project are our UI pages, along with the backend support for each. For each page, we will test the data entry and retrieval using Postman. We will also be making sure each page, overlay, and all of their internal components are smoothly interconnected through manual usage.

5.2 Interface Testing

What are the interfaces in your design? Discuss how the composition of two or more units (interfaces) are being tested. Tools?

The interfaces used in the design are the UI pages, this includes the login page, dashboard, task list, calendar, and mental health pages. The user interface will go through manual testing through running the application as well as junit tests that will test for expected behavior and function of the application.

5.3 Integration Testing

What are the critical integration paths in your design? Justification for criticality may come from your requirements. How will they be tested? Tools?

The overall most critical path in our software is the communication between the front end application the user will use, and the cloud service that will be handling server duties. Our group will be using postman to test the integration between backend and frontend to simulate data entry, our group will also rigorously test the software through manual usage of the software.

5.4 System Testing

Describe system level testing strategy. What set of unit tests, interface tests, and integration tests suffice for system level testing? This should be closely tied to the requirements. Tools?

Similarly to our method of integration testing, we'll be using Postman to monitor and keep track of entered data and ensure that data is not lost upon leaving the page it was entered on. Rigorous testing of the flow of the UIs will also be utilized to make sure that all pages have continuity between each other.

5.5 REGRESSION TESTING

How are you ensuring that any new additions do not break the old functionality? What implemented critical features do you need to ensure they do not break? Is it driven by requirements? Tools?

App flow and the basic functionalities of the app are vital in ensuring that any other functionality doesn't break. Our group will be using gitlab's CI/CD to ensure there are no issues with adding in new code and regression testing in regards to new contributions.

5.6 ACCEPTANCE TESTING

How will you demonstrate that the design requirements, both functional and non-functional are being met? How would you involve your client in the acceptance testing?

Jewels Academy wishes for a trial of testing the app through some of their students to collect feedback on what does work, what doesn't work, and what other features are or are not needed or utilized.

5.7 SECURITY TESTING (IF APPLICABLE)

Once we set up our server or cloud on the backend we will look for proper strategies into increasing security throughout our application. For example, we are looking to do penetration testing on different within the application, such as wireless, network, firewall, and web application

5.8 RESULTS

What are the results of your testing? How do they ensure compliance with the requirements? Include figures and tables to explain your testing process better. A summary narrative concluding that your design is as intended is useful.

In addition to regular development testing, our team will use GitLab's CI/CD in order to ensure that our application works seamlessly on Google Cloud. By using this feature, we'll be able to make sure that each iteration of our development allows our application to work as intended before being deployed.
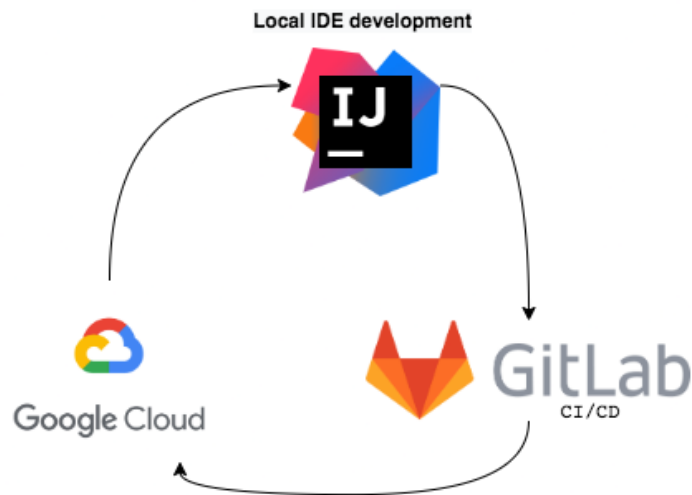
[Figure 15]: Iteration Cycle

6 Implementation

We plan to create all that we have shown in the original proposed design first, starting with basic connectivity between screens and progressing to data storage and usage. Once all aspects of the original design are fulfilled, we plan to add on new features on top of the existing design, such as more settings and other bits of information on the dashboard. We will also refine the design aspects of the proposed design as to make it more visually appealing as development continues, and the group gets a firmer and more concrete view of the UI as it is created in software. The functionality will also undergo iteration to increase the speed and efficiency of the application as development continues and issues develop. This development will constantly undergo iteration as it is developed in line with AGILE development practice including the creation of new features.

7 Professionalism

7.1 Areas of Responsibility

[Table 5]: Areas of Responsibility Breakdown

Area of Responsibility	Definition	NSPE Cannon	SE code of ethics
Work Competence	Perform work of high quality, integrity, timeliness, and professional competence.	Perform services only in areas of their competence; Avoid deceptive acts.	Principle 6: Profession. Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.
Financial Responsibility	Deliver products and services of realizable value and at reasonable costs.	Act for each employer or client as faithful agents or trustees.	Principle 2: Client and employer. Software engineers shall act in a manner that is in the best interests of their client and employer, consistent with the public interest.
Communication Honesty	Report work truthfully, without deception, and are understandable to stakeholders.	Issue public statements only in an objective and truthful manner; Avoid deceptive acts.	Principle 7: Colleagues. Software engineers shall be fair to and supportive of their col-leagues.
Health, Safety, Well-Being	Minimize risks to safety, health, and well-being of stakeholders.	Hold paramount the safety, health, and welfare of the public.	Principle 5: Management. Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.
Property Ownership	Respect property, ideas, and information of clients and others.	Act for each employer or client as faithful agents or trustees.	Principle 3: Product. Software engineers shall ensure that their products and related

			modifications meet the highest professional standards possible.
Sustainability	Protect the environment and natural resources locally and globally.	Adhere to principles of sustainable development to protect the environment for future generations	Principle 8: Self Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.
Social Responsibility	Produce products and services that benefit society and communities.	Conduct themselves honorably, responsibly, ethically, and lawfully so as to enhance the honor, reputation, and usefulness of the profession.	Principle 1: Software engineers shall act consistently with public interest.

7.2 PROJECT SPECIFIC PROFESSIONAL RESPONSIBILITY AREAS

- Work Competence:

    - We agree that this applies to our project because our clients expect professional and organized work. Whether that be the program itself, our communication on the project throughout the team. The professional communication with the client is a High priority when it comes to business in the software industry.

- Financial Responsibility:

    - This criteria applies to our project in a professional context because our client will be providing any necessary expenses that are required for us to create their application. In this area, our team is performing

high because there has not yet been any monetary transactions between our group and the client.

- Communication Honesty:

    - This area of responsibility is vital for our project in a professional context as cohesive, honest communication is needed in order to streamline and clarify the process in which the client wants us to complete the deliverables for this project. This also allows us to discuss potential and needed features clearly without any tensions or doubts.

- Health, Safety, Well-being:

    - This area of responsibility doesn't apply to our group's project because there aren't any health/safety risks that are associated with developing our application. The performance of our team in this aspect of the project is not applicable because the vast majority of our work and communication will be done using a computer.

- Property Ownership:

    - This is something that applies to our project. As we create the application our team wants to take ownership and output the best possible application we can for our client. This means taking responsibility for our product and continuing professionalism with creating the application. Our team is performing this at a High rate due to the fact that we care about our product and want to help out our client as much as possible with the knowledge we have gained over the course of our years.

- Sustainability:

    - This area of professional responsibility does not correlate with our project as much as the other areas. Our project will be entirely set in a virtual environment with no impact to the outside environment. This area is not applicable to our project.

- Social Responsibility:

    - This professional responsibility area is key to our project's success because we will be creating an application that supports a better life-style for the community we are creating it for. This is a high priority for this project because this area will be one of the main focuses and purposes of this application. Our team is performing high in this criteria.

7.3 MOST APPLICABLE PROFESSIONAL RESPONSIBILITY AREA

Property ownership is an important area of responsibility to our project because each of our team members shares the goal of producing a product that is high quality and exceeds the expectations of our stakeholders. We've demonstrated a high level of proficiency in this area by staying continuously engaged with our client, our faculty advisor, and each other. Each member of our team is an aspiring software engineer and therefore we have a mutual understanding that delivering a high quality product will be in the best interest of developing our educational skill set, and a professional responsibility in the future. Our group has spent a considerable amount of time and effort with primary and secondary research in accordance with this professional responsibility area, that research has allowed our group to get to a much more refined product than if the group were to work without considering that we want the final product to reach its highest quality given our skills.

8 Closing Material

8.1 DISCUSSION

The main result of our project will be a completed mobile application, which will not involve the same type of experimentation as a project in the electrical engineering, or computer engineering fields will as this project is completely software based. The experimentation that our group will be doing is impermant and can easily be iterated upon, as there is no physical construction and software development is extremely malleable. If all the requirements are met we will be left with a high efficiency, visually appealing, and multifunctional application with features ranging from time management to mental health and stress management. This product will be easy to use for students as young as 9 years old using visual description and a quality UI/UX and will integrate with popular services to increase usability including Google services and Canvas.

8.2 CONCLUSION

Currently our group has created a plan for the creation of the application, we have also done market research for applications similar, and decided on the technologies and APIs necessary for the initial development of the prototype.The group has also begun the process of development including both frontend UI/UX

development, and backend functionality development for the skeletal prototype of the final product. The development environment has been set up along with a consistent file structure, a MySQL database is currently setup and running in the Google Cloud environment along with the necessary APIs for database management, there is an in progress React Native UI file created, and a backend Python SQL connector with a prototype of some of the necessary backend functionality to connect the UI to the cloud MySQL database with functionality to log and edit users. The major constraint on the group's progress so far has been the unfamiliarity of the group members with the technologies necessary for the development, however as time goes on this issue will be less ans less problematic as the members get more familiar with the process of development using these technologies.

8.3 REFERENCES

[1]   "SQLAlchemy - The Database Toolkit for Python", *Sqlalchemy.org*, 2022. [Online]. Available: https://www.sqlalchemy.org/. [Accessed: 23- Apr- 2022].

[2]"Google Cloud documentation | Documentation", *Google Cloud*, 2022. [Online]. Available: https://cloud.google.com/docs/. [Accessed: 23- Apr- 2022].

[3]"GitHub - GoogleCloudPlatform/cloud-sql-python-connector: Cloud SQL connector for Python", *GitHub*, 2022. [Online]. Available: https://github.com/GoogleCloudPlatform/cloud-sql-python-connector#how-to-use-this-connector. [Accessed: 23- Apr- 2022].

[4]"API Reference | Calendar API | Google Developers", *Google Developers*, 2021. [Online]. Available: https://developers.google.com/calendar/api/v3/reference. [Accessed: 23- Apr- 2022].

[5]"Outlook calendar API overview - Microsoft Graph", *Docs.microsoft.com*, 2021. [Online]. Available: https://docs.microsoft.com/en-us/graph/outlook-calendar-concept-overview. [Accessed: 23- Apr- 2022].

[6]"Using the Trello API", *Trello.org*. [Online]. Available: http://www.trello.org/help.html. [Accessed: 23- Apr- 2022].

[7]"Canvas LMS REST API Documentation", *Canvas.instructure.com*, 2022. [Online]. Available: https://canvas.instructure.com/doc/api/index.html. [Accessed: 23- Apr- 2022].

[8]"Welcome to Python.org", *Python.org*, 2001. [Online]. Available: https://www.python.org/doc/. [Accessed: 23- Apr- 2022].

8.4 APPENDICES

Any additional information that would be helpful to the evaluation of your design document.

If you have any large graphs, tables, or similar data that does not directly pertain to the problem but helps support it, include it here. This would also be a good area to include hardware/software manuals used. May include CAD files, circuit schematics, layout etc,. PCB testing issues etc., Software bugs etc.

8.4.1 Team Contract

Team Members:

1) _____Theron Gale_____   2) _____Adrian Vanderveer_____

3) _____Ben Hourigan_____   4) _____Olusola Ogunsola_____

5) _____Christopher Burgos_____

Team Procedures

1.   Day, time, and location (face-to-face or virtual) for regular team meetings:

- Virtual(Discord) - Wednesday 6-6:30

2.   Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):

- Discord, Webex, Face to face

3.   Decision-making policy (e.g., consensus, majority vote):

- Consensus and discussion

4.   Procedures for record keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):

- Shared google document - Adrian and Theron

Participation Expectations

1.   Expected individual attendance, punctuality, and participation at all team meetings:

Team members are expected to attend and engage in all meetings, it is understood there can be extenuating circumstances, however members who will be unable to attend meetings should let the group know in advance whenever possible.

2.   Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:

All team members are expected to actively participate in the work for assigned work as well as meet deadlines and keep on track with the work planned, if there are issues the member in question should ensure the group is notified that their tasks may not be completed on time so there are no issues that cause overly long delays to project development progression.

3.   Expected level of communication with other team members:

Team members should communicate any time it is necessary for regular updates, questions, concerns, and decisions necessary to the team.

4.   Expected level of commitment to team decisions and tasks:

Team members should be actively engaged in their assigned tasks as well as in the decision making process to ensure smooth progression of the project.

Leadership

1.   Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):

- Ben: File Consistency/Organization
- Adrian: Team Organization
- Theron: Individual Component Design
- Chris: Client interaction
- Sola: UI Design

2.   Strategies for supporting and guiding the work of all team members:

- Regular meetings with updates to the progression of work
- Creating a casual environment where suggestions and questions are encouraged to be shared to improve the overall process and final product
- Dedicated roles for team members with well defined goals and responsibilities

3. Strategies for recognizing the contributions of all team members:

- Regular meetings with updates to the progression of work
- Gitlab space with individual contributions to the code base

Collaboration and Inclusion

1. Describe the skills, expertise, and unique perspectives each team member brings to the team.

Adrian Vanderveer:

- Experience with app development
- Backend Development
- Web development
- Graphic design Experience
- IntelliJ Idea IDE Experience

Theron Gale

- Experience with app development
- Frontend Development
- A Little Web Development
- Some database experience
- C, Java, JavaScript, SQL experience

Ben Hourigan

- Experience in backend development (Java, Spring)
- A ton of front end web development experience
- Good amount of database management experience

Chris Burgos:

- Experienced in front end development (Java, HTML)
- Experience in testing
- Client interactions
- Some SQL experience/knowledge
- App and game development
- Know C,C#, Java, python, SQL,

- Good with android studio

Sola:

- Experience with Android app development
- Github CI/CD
- Backend development (Java, Spring)
- Java, C, PHP, SQL
- Android studio/Intellij IDE

2.   Strategies for encouraging and support contributions and ideas from all team members:

- Praising each other's contributions to the project
- Create a collaborative work environment through positive reinforcement
- Support new ideas with positivity

3.   Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)

- Discuss issue with team in weekly meeting

Goal-Setting, Planning, and Execution

1.   Team goals for this semester:

- Complete a deliverable for a simple version of the Jewels Academy Time Management App

2.   Strategies for planning and assigning individual and team work:

- Sprints
- Agile ideals
- SCRUM
- Trello
- Github

3.   Strategies for keeping on task:

- Team meetings weekly to discuss project progress and create plans to stay on track
- Sprint retrospective meeting

Consequences for Not Adhering to Team Contract

1.   How will you handle infractions of any of the obligations of this team contract?

- Consult faculty advisor with issues and discuss solutions

2.   What will your team do if the infractions continue?

- Set up a 1 on 1 meeting with the professor to decide consequences

****

a) I participated in formulating the standards, roles, and procedures as stated in this contract.

b) I understand that I am obligated to abide by these terms and conditions.

c) I understand that if I do not abide by these terms and conditions, I will suffer the

consequences as stated in this contract.

1) _____ Adrian Vanderveer_ _____ DATE ____04/23/2022____

2) _____ Ben Hourigan_ __ _____ DATE ____04/23/2022____

3) _____ Theron Gale _____ DATE ____04/23/2022____

4) _____ Olusola Ogunsola _____ DATE ____04/23/2022____

5) _____ Christopher Burgos _____ DATE ____04/23/2022____