ScheduSmart

Sprint 1 Planning Document

Reece Ausmus, Cassie Chang, Stanley Huang, Bradley Norris, Himanshu Sinha, Gloria Xu

Sprint Overview

During this sprint, we hope to build a website that will hold our entire project with the requirements including viewing the new feature of the website according to what contents we put, creating a new account, safely login with a specific account with username and password, and log out the account safely. Also, we hope to build an effective database that we can add and store multiple accounts information.

Scrum Master:

Reece Ausmus

Meeting Plan:

We will meet weekly on Sunday at 5:00pm with our TA. We also will meet weekly on Saturday at 11:00am with just our team. Each meeting will be on Zoom.

Risks and Challenges

Although the direction to construct the project is pretty clear, most of us are not familiar with the tools we are going to use such as Nginx, React, etc. The time for us to learn and get used to those new tools will potentially delay our project plan. Another challenge is that some of us are going to have exams for this or next week, in which they will need plenty of time to study on other things and are unable to focus on this project.

Current Sprint Detail

User Story #1

As a user, I would like to create a new account.

#	Description	Estimated Time	Owner
1	Create a UI panel that user can input their personal information	3hr	Gloria, Stanley
2	Create a database using firebase with the ability to modify and view the data inside	3hr	Gloria, Stanley
3	Create an algorithm that identifies whether the login data (username and data) is correct or not, and also identifies whether the username is available for a new account.	2hr	Stanley
4	Connect the algorithm with UI to indicate whether the input is available or not	2hr	Gloria
5	Create a unit test for accessing firebase related to accounts.	1hr	Stanley

Acceptance Criteria:

- Given that the UI panel is correctly implemented, when users input any information, then input of any alphabetic, number, or special characters they prefer should be allowed.
- Given that the UI panel is secure, when information is provided, then it should show the user's input but hide the character user input for password.
- Given that the UI and algorithm run properly, when users attempt to input special characters, then all of them should be denied such as
  - Double quote: 34 (")
  - Single quote: 39 (')
  - Backslash: 92 (\)
  - Space: 32 ( )
  - Any control character: 0-31 and 127

- Given that the implementation of the algorithm is correct, when the same username that already exists in the database is inputted, then it should be denied.
- Given that the UI panel is implemented correctly, when the input is not acceptable, then it should clearly show reasons behind it.

User Story #2

As a user, I would like to securely login into and logout of my account

#	Description	Estimated Time	Owner
1	Create a UI panel where the user can input their username and password	3hr	Bradley
2	Create an system that will verify the correctness of the login	2hr	Bradley
3	Create a two-factor authentication system	3hr	Bradley
4	Create a button that will allow users to logout	2hr	Cassie
5	Create a unit test to check login verification	1hr	Cassie

Acceptance Criteria

- Given that the UI Panel runs properly, when users try to input their usernames, then it should include a space for the input.
- Given that the UI Panel is implemented correctly, when the user tries to input their password, then it should include a space and hide the characters, representing them as black dots.
- Given that the settings are implemented properly, when the username and/or password is incorrect, then it should return a single message “Login Information Incorrect”.
    - It should NOT specify which piece of information is incorrect, because that can be viewed as a security flaw.
- Given that the settings are implemented properly, when the username/password are correct, then the system will send an email with a randomly generated code to the user’s email.
    - The UI panel will then display a box for the code

  - If the code is properly placed into this new box, the user will be successfully logged in.
- Given that the logging system is properly implemented, when users want to logout, then the logout button should ask them before allowing them to.

User Story #3

As a user, I would like to edit/update account information.

#	Description	Estimated Time	Owner
1	Create a UI page that displays all user information. This includes learning new technology required for said UI.	2hr	Himanshu
2	Allow users to edit specific pieces of user information. This will include backend functionality for changing user data on the client side.	2hr	Himanshu
3	Update changed information in the database	2hr	Gloria
4	Create a unit test to check if information is properly updated in the database	2hr	Bradley

Acceptance Criteria

- Given that the UI page runs correctly, when displaying all user information is needed, it should be in a neat and organized fashion, with labels for each piece of information.
- Given that the implementation of the UI page is correct, when modifications of information are needed, then the UI page should clearly display which pieces of information can be edited, and require the input of a password to change the password.
- Given that the UI page is implemented properly, when the change should be stored to the database, so that the changes are carried over between logins.

User Story #4

As a user, I would like to view my calendar and assignment tracker.

#	Description	Estimated Time	Owner
1	Create a basic UI display homepage for the assignment tracker and calendar.	3hr	Stanley
2	Create a UI display for the assignment tracker to display all current assignments.	3hr	Reece
3	Create functionality to sort assignments by different factors.	2hr	Reece
4	Create UI displays for the calendar to display all current events in day, week, or month format.	2hr	Himanshu
5	Create unit tests to ensure proper display for calendar and assignment tracker and correct information displayed on both.	2hr	Reece

Acceptance Criteria

- Given that the UI page should display all information in a neat and organized fashion, with clear labels for each piece of information, and a visual design that is in line with the rest of the program
- Given that the Calendar section is implemented properly, when users check their calendars, then it should have clear labels for “hours” on one axis, and date/day-of-the-week on the other axis
- Given that the assignment tracker is implemented correctly, when users check unfinished assignments, then it should have a system that organizes current assignments by priority, putting the most important assignment at the top
- Given that the assignment tracker is implemented correctly, when users check assignments, it should also have a section that displays recently completed items, those items are deleted upon signing out.

User Story #5

As a user, I would like to add events to my calendar.

#	Description	Estimated Time	Owner
1	Create a button that allows users to add events.	2hr	Gloria
2	Create a UI pop-up that allows users to input information and create an event.	2hr	Gloria
3	Create a UI pop-up that shows options to repeat the event including daily, weekly, custom, etc.	1hr	Reece
4	Add the event to the database and update the user's availability.	2hr	Reece
5	Create a function that will access the backend and get all information about all events that have been created and list them in User interface arranged by time	3hr	Stanley
6	Create unit tests for events added to the database and user availability.	2hr	Reece

Acceptance Criteria

- Given that adding functionality is correctly implemented, when users want to add events by clicking the button, then the button should be clear in its purpose, but also small enough that it doesn't take up room. The pop-up should appear over the rest of the site, but not completely cover it up.
- Given that the event-adder popup is correctly implemented, when users add events, it should have space for event-name, date, time, location, repetition, and event-description.
  - There should be a 16 character limit on name
  - Date should be in MM/DD/YYYY format
  - Location will be text for this sprint
  - Repetition will show options for weekly, daily, etc.
  - Event description will have a 50 character limit

- Given that the calendar is correctly implemented, when users check their calendar, then it should cleanly display all events, including the new one, on the correct date and time with the correct information.

User Story #6

As a user, I would like to add assignments to my assignment tracker

#	Description	Estimated Time	Owner
1	Create a button that allows users to add assignments. This also includes time required to learn new technology.	3hr	Himanshu
2	Create a UI pop-up that allows users to input information and create an assignment. This also includes coding back-end functionality for generating assignments.	3hr	Himanshu
3	The tracker should place new assignments at the correct priority. This will include testing for the algorithm required to produce priority.	3hr	Himanshu
4	Add the assignment to the database.	2hr	Stanley
5	Create a unit test for assignments added to the database.	1hr	Himanshu

Acceptance Criteria

- Given that adding functionality is correctly implemented, when users want to add assignments by clicking the button, then the button should be clear in its purpose, but also small enough that it doesn’t take up room. The pop-up should appear over the rest of the site, but not completely cover it up
- Given that the assignment-adder popup is correctly implemented, when users add assignments, it should have space for assignment-name, date-due, work-load, and assignment-description
  - There should be a 16 character limit on name
  - Date should be in MM/DD/YYYY format
  - Work-load will be a float (2 decimals) in hours for units

  - Assignment description will have a 50 character limit
- Given that the assignment tracker is correctly implemented, when users check their assignments, then it should cleanly display all assignments, including the new one, in an organized manner and rank them by priority
  - Priority is determined by work-load/days-till-due-date
  - The resulting value is the amount of hours it should be worked on today
  - The assignment with the highest amount of hours is priority #1, second most priority #2, etc. etc.

User Story #7

As a user, I would like to mark completed assignments as done.

#	Description	Estimated Time	Owner
1	Create a button that allows users to mark completed work as done	2hr	Gloria
2	Update the information of the assignment with completed time	1hr	Reece
3	The assignment should be displayed on the panel showing the recently completed section.	1hr	Himanshu
4	Design a unit test to check whether the completed work is successfully switched to the right section	1hr	Cassie
5	Create a system that moves completed assignments to the completed assignment section	2hr	Reece
6	Create a unit test to check updated assignment status in database	1hr	Cassie

Acceptance Criteria:

- Given that the completed button is implemented correctly, when users click the button of the work, then the work can be marked as done.

- Given that the completed time of work is implemented correctly, when the users click the completed button, then the information of the work should update with the completed time in the MM/DD/YYYY format
- Given that the panel displaying completed sections is implemented correctly, after the users mark the work as done, then the panel for completed work should have the work on the list.

User Story #8

As a user, I would like to change my time zone.

#	Description	Estimated Time	Owner
1	Create a settings button that allows users to select the option of changing their time zone	3hr	Bradley
2	Create a list for user to open and pick up their current time zone	2hr	Bradley
3	All the assignments and works specified with time-related information should be changed to the right time based on the current time zone	1hr	Reece
4	Design a unit test to confirm if the function of changing time zone works in the calendar	2hr	Bradley
5	Design a unit test to confirm if the time zone change is reflected accordingly across the assignments list	2hr	Bradley

Acceptance Criteria:

- Given that the settings feature is created correctly, when users click on the settings button, then they should have a list of choices for them to select, such as changing time zone, changing calendar style, etc.
- Given that the list of different time zones is implemented correctly, when users select the list, then they should have a list of all the time zones to scroll down and choose one of them.

- Given that the function of changing the time zone is implemented correctly, after users change the time zone, then the time zone should be altered to the desired one successfully.
- Given that the function of changing the time zone is implemented correctly, after users change the time zone, then the information related to the time of each work should be displayed correctly, including the due date, and the starting and ending time of works.

User Story #9

As a user, I would like interactive web tours to help me be familiar with the functions of ScheduSmart.

#	Description	Estimated Time	Owner
1	Create interactive web tours to introduce some important features of our application to new users	4hr	Cassie
2	Create an option for users who already familiar with our product to skip the tours	2hr	Cassie
3	Create a unit test to check whether the tours work successfully.	1hr	Cassie
4	Design a unit test to check whether the user can skip the tours	1hr	Cassie

Acceptance Criteria:

- Given that the interactive web tours are implemented correctly, when new users first enter the web, then they should see the tours appearing automatically.
- Given that the interactive web tours are implemented correctly, after users watch the tours, then they can know how to use our product, ensuring usability.
- Given that the function of skipping the tours is implemented correctly, when users who are already familiar with ScheduSmart, then they can skip the tours with the “skip” button.

User Story #10

As a user, I would like to set up reminders for my events and tasks.

#	Description	Estimated Time	Owner
1	Create a setting button for users to turn on/off reminders	2hr	Gloria
2	Create a list for users to choose how much time before the events and tasks they would like to receive reminders	2hr	Gloria
3	Create a UI pop-up to display the information of events and task	2hr	Gloria
4	Create a unit test to check if the reminders are set properly as expected	1hr	Gloria

Acceptance Criteria:

- Given that the reminders setting function is implemented correctly, when users turn on the reminders, then they will receive the message sent by our system.
- Given that the function of setting reminders' time is implemented properly, when the time is up, users can receive reminders as expected.
- Given that the UI pop-up is built correctly, when the time is up, the reminders will be shown on the screen.

User Story #11

As a user, I would like to change the whole displayed text as my native language.

#	Description	Estimated Time	Owner
1	Create a settings button that allows users to change the language of displayed text.	2hr	Stanley
2	Create a list for users to open and pick up their preferred language.	1hr	Stanley
3	Create all files that support the application to change into different languages including English, Spanish, Japanese, Chinese. (Translator will be used for Spanish since I only	3hr	Stanley

	know a little bit)		
4	Design a unit test to confirm if the function of changing languages works in the system	1hr	Stanley

Acceptance Criteria:

- Given that the list of different languages is implemented correctly, when users select the list, then they should have a list of all the languages to scroll down and choose one of them.
- Given that the function of changing languages is implemented correctly, after users change the language to their native one, then the whole displayed texts are shown in it.
- Given that the function of changing languages is implemented properly, after users change the language, pages such as interactive web tours and reminders should be displayed in the chosen language as well.

User Story #12

As a user, I would like to drag and drop events in my schedule.

#	Description	Estimated Time	Owner
1	Create a button that allows users to drag and drop events	2hr	Cassie
2	Create a UI pop-up that double checks if the user wants to drag and drop the event.	2hr	Cassie
3	Drag and Drop the event to the database and update the user's availability	4hr	Cassie
4	Create unit tests for events dragged and dropped from the database and user availability	1hr	Cassie

Acceptance Criteria:

- Given that dropping functionality is properly implemented, when users want to drop the events by clicking the button, then the pop-up for double checking should appear immediately.

- Given that the UI pop-up for double checking is correctly implemented, when users click the dropping button, then the pop-up should disappear and the system begins to do the deleting action.
- Given that the calendar is correctly implemented, when users check their calendar, then it should cleanly display all events with the correct information and the dropped one is not shown anymore.

User Story #13

As a user, I would like to create different calendars that I can turn on and off individually.

#	Description	Estimated Time	Owner
1	Create a button to add another calendar	1hr	Reece
2	Create a visual indicator separating the different calendar styles	2hr	Reece
3	Create a system that allows the user to seamlessly cycle through calendars	2hr	Reece
4	Create a unit test to check if the calendar can turn on/off individually	1hr	Cassie

Acceptance Criteria:

- Given that no calendars are added and the functionality is properly implemented, the display should appear as a default, empty calendar. This means simply showing the dates and times, but with nothing filled in at all.
- Given that there is at least one calendar added and the functionality is properly implemented, the display should show the correct style (colors, fonts, etc.), events, and time-blocks for the currently selected calendar,
- Given that there is at least one calendar added and the functionality is properly implemented, the user should be able to delete calendar data from the dashboard. This deletion should include permanence warnings, as well as update the data in the database.
- Given that there is at least one calendar added and the functionality is properly implemented, users should be able to see which calendar style is currently selected, as well as which calendar styles are available to choose.

User Story #14

As a user, I would like to visualize my calendar in different formats (day, week, month, etc.)

#	Description	Estimated Time	Owner
1	Create a list in settings for users to choose their preferred visualization format	2h	Gloria
2	All the events and tasks should be displayed in the desired format	4hr	Cassie
3	Create a system that allows users to cycle between different formats while on the main dashboard	3hr	Cassie
4	Create a unit test to check if the calendar is in the desired format	1hr	Cassie

Acceptance Criteria:

- Given that settings page is properly implemented, the user should be able to see a list of all the different visual formats, and select a drop down menu that allows them to select the one they wish to set as default
- Given that the dashboard is properly implemented, the user should be able to quickly select the visual format they want, without having to go to the setting page each time
- Given that the dashboard is properly implemented, the user should not have their data erased or altered when the visual presentation is changed quickly,

User Story #15

As a user, I would like to mark a period as “working time” for every week, including commutes, and “out-of-office” time.

#	Description	Estimated Time	Owner
1	Create a system that allows users to separate time slots (different from events, which are overlaid onto of these classifiers)	3hr	Reece
2	Create a system that allows users to turn the broad overlay on and off	2hr	Gloria
3	Create a system that allows users to add new classifiers to the overlay	1hr	Reece
4	Create a unit test to check if the calendar has	1hr	Gloria

	proper classified time		

Acceptance Criteria:

- Given that the classifier is properly implemented, each day in the calendar should be able to be separated between working time, rest time, sleep time, without interrupting the specific events that are scheduled separately from that.
- Given that the classifier is properly implemented, when the day is separated, the partitioned chunks should be clearly displayed with different colors and labels
- Given that the classifier is properly implemented, partitions should not be able to overlap, and the user should be able to add new partitions as well as deleted old classifiers.

User Story #16

As a user, I would like to see the weather.

#	Description	Estimated Time	Owner
1	Create a small window to display the temperature of a day. Includes backend functionality for loading the temperature from an auxiliary source	4hr	Himanshu
2	Create a settings page for users to choose turn on/off weather displaying, as well as set their location for the weather.	2hr	Himanshu
3	Create a pop-up to notify people of the weather when they open the app at certain times of the day (mornings etc.)	2hr	Himanshu
4	Create a unit task to check if the temperature is properly showing	1hr	Himanshu

Acceptance Criteria:

- Given that the weather functionality is properly implemented, the weather information presented should be accurate to the location and time of day that the user is accessing the program from.
- Given that the weather functionality is properly implemented, the weather information should be clearly visible, and easily readable.
- Given that the weather functionality is properly implemented, the weather information should

User Story #17

As a student, I would like to sort my assignments based on due date, creation date, class, type, etc.

#	Description	Estimated Time	Owner
1	Create an algorithm that sorts assignments based on various criteria as efficiently as possible	2hr	Himanshu
2	Create a system that allows users to cycle between these sorting algorithms as easily as possible	2hr	Himanshu
3	Create a system that allows users to set a preferred sorting feature in the settings including domain. Also, implement all sorting algorithm including: greedy algorithm quicksort for starting time quicksort for ending time quickselect base on time interval Max-heap, min heap	5hr	Stanley
4	Create a unit test to check the sorting algorithm works	1hr	Stanley

Acceptance Criteria:

- Given that the algorithm is implemented correctly, the list should be sorted so that it operates in at most O(nlogn) time. With space complexity operating at most at O(N), ideally having no auxiliary space requirement.
- Given that the sorting feature is implemented correctly, the user should be able to switch between different sorted lists without loading very long (less than a second unless the assignment list is quite large)
- Given that the sorting feature is implemented correctly, the user should be able to set their preferred feature to sort by in settings, which should be updated to the database so that it is saved between logoffs and logins.

User Story #18

As a user, I would like to export my schedule to a printable format.

#	Description	Estimated Time	Owner
1	Develop a functionality to export the user's schedule to a printable format (PDF, CSV, etc.).	3hr	Bradley
2	Create a user interface for users to select export options and initiate the export process.	2hr	Bradley
3	Create unit tests to ensure that the selected date range for export is within the valid range of the calendar.	2hr	Bradley
4	Create unit tests to ensure the schedule includes all relevant information, such as event details, times, locations, and any attached documents/files.	2hr	Bradley
5	Implement error handling to notify the user if the export process fails and provide guidance on resolving the issue.	2hr	Bradley

Acceptance Criteria:

- Given that the export process is implemented correctly, the user should be able to generate a file in the selected format (PDF, CSV, etc.) containing the entire schedule for the specified date range and selected calendars.
- Given that the export process is implemented correctly, if a user selects a date range for export that is outside the valid range of the calendar, the system should display an error message, preventing the export process until a valid date range is chosen.
- Given that the export process is implemented correctly, if an error is encountered, the system should display a user-friendly error message, guiding the user on potential solutions.

User Story #19

As a user, I would like to adjust what time my schedule starts and ends, essentially scheduling when I am sleeping.

#	Description	Estimated Time	Owner
1	Implement a feature allowing users to set their preferred daily start and end times for scheduling.	2hr	Reece

2	Create a user interface for users to easily adjust and save their daily schedule start and end times.	2hr	Reece
3	Create a unit test to ensure the scheduler considers the user’s sleeping time accordingly	1hr	Reece

Acceptance Criteria:

- Given that the scheduling algorithm is correctly implemented, users should be able to define and prioritize their sleep hours in the daily schedule.
- Given that the user interface is correctly implemented, users should easily be able to configure their preferred daily start and end times for scheduling.
- Given that the scheduling algorithm is correctly implemented, users should not be able to create events inside of their allocated sleeping time.

Hours Summary

Reece Ausmus:	30
Cassie Chang:	30
Stanley Huang:	30
Bradley Norris:	30
Himanshu Sinha:	30
Gloria Xu:	30

Team 30: ScheduSmart - Updated Product Backlog

Reece Ausmus, Cassie Chang, Stanley Huang, Bradley Norris, Himanshu Sinha, Gloria Xu

02/12/24

CS 307

Project Coordinator: Dominic Damoah

Team Leader: Reece Ausmus

Project Title:

ScheduSmart

Problem Statement

As the world continues to evolve and the average person's life continues to get more and more complicated, the need for a tool that unlocks our capacity for productivity has risen. Our project seeks not only to deliver a tool that allows its users to organize their lives with assignment trackers and a fully functional calendar, but will also include a system that generates efficient schedules automatically, using the power of AI. The project will function as a website, offering a lesser computational load to users as opposed to the typical application approach. The project will offer a fully customizable experience to the user, making this website more user-friendly and creating a more enjoyable experience for the user than other similar applications.

Background Information

In today's fast-paced world, people are constantly juggling multiple responsibilities and tasks, making it increasingly challenging to stay organized and productive. The need for a tool that can effectively manage and streamline our lives has become more crucial than ever. Our project aims to address this need by providing a comprehensive productivity tool that combines assignment trackers, a functional calendar, and an AI-powered scheduling system. The domain of users who could potentially benefit from this service is practically anyone who has an internet connection and device. On a more specific scope, our targeted users are people who require organizational services in their daily lives such as college students, professors, freelance professionals, etc. Students and professors need an integrated calendar and task manager for classes and assignments. Freelance professionals need a calendar to schedule meetings with clients and a task manager to manage billing and other secretarial services.

Many calendar applications already exist, however none have the specific capabilities we plan to implement (AI/Database/Statistics Dashboard) to our knowledge. Some popular examples include Trello, Asana, Todoist, Google Calendar, and Microsoft Outlook. These applications provide users with tools to manage tasks and appointments, but they may not have the same level of automation and AI-powered scheduling capabilities that our project aims to offer. Those will separate our service from others and elevate the product to higher levels.

Other similar applications that are already in existence sometimes have a high learning curve, missing features, or unintuitive design. To address this, we plan to have a user-friendly interface, as well as a suite of features that are not found in many other applications.

Requirements (Backlog)

Functional:

1. Introduction:
2. Login system:
    - 2.1. **As a new user, I would like to create a new account**
    - 2.2. **As a user, I would like to login and logout into my account**
3. Calendar system:

3.1. **As a user, I would like to add events to my calendar.**

3.2. **As a student, I would like to add my homework deadline to the schedule.**

3.3. As a student, I would like to schedule time to study for tests with recurring events.

3.4. As a student, I would like to find the closest available time (15-minute meeting, etc).

3.5. **As a student, I would like to see when I have no class (festival or break).**

3.6. As a student, I would like to manage time schedules before the start of a semester to smooth the course enrollment procedure, such as scheduling class times and breaks.

3.7. As a student, I would like to have the calendar automatically schedule time to complete homework.

3.8. As a student, I would like to type my schedule or tasks into a chat box and have AI schedule my day for me.

3.9. As a student, I would like to be able to list times for exams.

3.10. As a student, I would like to receive suggestions for optimal study plans and breaks.

3.11. As a student, I would like to receive notifications about deadlines and exams.

3.12. As a user, I would like to add locations to my events using text or a map feature (i.e., Google Maps).

3.13. As a student, I would like to attach documents/files to my tasks.

3.14. As a student, I would like to see the weather and have it impact my AI-generated schedule.

3.15. As a student, I would like to share my schedule with teammates to find an available time slot for meetings.

3.16. As a user, I would like to be able to add a repeatable event (sprints, meetings, etc. etc.)

3.17. As a worker, I would like to mark a period as “working time” for every week, including commutes, and “out-of-office” time.

3.18. As a user, I would like to visualize my calendar in different formats (day, week, month, etc.)

3.19. As a user, I would like the calendar home page to be simple and not overwhelming while including pertinent information, such as the date and time.

3.20. As a user, I would like to create different calendars that I can turn on and off individually.

3.21. As a user, I would like to drag and drop events in my schedule.

3.22. **As a user, I would like to change my time zone.**

3.23. As a user, I would like to be able to navigate the schedule with keyboard shortcuts without using the mouse.

3.24. As a user, I would like to adjust what time my schedule starts and ends, essentially scheduling when I am sleeping.

3.25. As a user, I would like to see what time of day it is in relation to my events.

3.26. As a user, I would like to see a quick view of my calendar without being overwhelmed with all features.

3.27. As a team manager, I would like to have a meeting with all the team members when all of them are free.

3.28. As a nurse, I would like to have all patients’ dosing time (with constant intervals)

4. Assignment tracker:

4.1. **As a student, I would like to have an assignment tracker.**

4.2. **As a user, I would like to view my calendar and assignment tracker.**

4.3. **As a student, I would like to be able to prioritize my tasks.**

4.4. **As a student, I would like to add assignments to the assignment tracker.**

4.5. As a student, I would like to track progress on my assignments and see how much time I have planned to work on them.

4.6. As a student, I would like to sort my assignments based on due date, creation date, class, type, etc.

4.7. As a user, I would like to set and track long term goals.

4.8. As a user, I would like my to-do list tasks to synchronize with calendar events if toggled.

4.9. **As a user, I would like to mark work done as completed.**

4.10. As a user, I would like to differentiate events and tasks.

4.11. As a team leader, I would like to assign tasks to employees.

5. Settings:

5.1. As a bodybuilder, I would like to add the daily calories I have.

5.2. As a student, I would like to see statistics on my productivity at certain times.

5.3. As a bodybuilder, I would like a summary of how I work on it (total and average per month).

5.4. As a user, I would like to have a modular, customizable home screen.

5.5. As a user, I would like to set up reminders for my events and tasks.

5.6. As a user, I would like to invite other users to my events and see events I am invited to.

5.7. As a user, I would like to color coordinate my schedule.

5.8. As a user, I would like to change the whole displayed text as my native language.

6. Misc:

6.1. As a student, I would like to be able to take notes.

6.2. As a student, I would like to be able to put a to do list aside for convenience.

6.3. As a student, I would like to have a small notebook to record things I have done in the past as a reminder (finished assignments, meetings, etc.)

6.4. As a student, I would like to be able to import assignments from Brightspace, Canvas, or Blackboard.

6.5. As a user, I would like to be able to connect Google Calendar to this calendar.

6.6. As a user, I would like to export my schedule to a printable format.

6.7. As a user, I would like to track daily habits that I set up, e.g. 5 mins of reading, 10 mins stretching, and cook 3 meals.

6.8. As a user, I would like to see the weather.

6.9. As a boss, I would like to have all workers see the schedule but unable to change it.

6.10. As a team manager, I would like to have my team members drop down what they have done for the project in the past.

6.11. As a team manager, I would like to have a chart (summary) on how each team member processes their job.

6.12. As a team leader, I would like to receive automated progress reports from the team.

6.13. As an employer, I would like to see when my employees will be out of the office.

6.14. As an employer, I would like to have the ability to accept or deny time-off requests that employees submit.

6.15. As an employee, I would like to schedule when I will be out of the office.

6.16. As an employee, I would like to quickly send meeting invites to coworkers.

6.17. As an employee, I would like to integrate my calendar with Zoom meetings, such as creating a meeting from the calendar, joining a meeting from the calendar, and getting notifications from the calendar about my meetings.

6.18. As a dietician, I would like to track my daily calories and water intake.

6.19. As a software engineer, I would like to add a scrum framework to my schedule

6.20. As a software engineer, I would like to upload design pdfs to the notebook section

6.21. As a software engineer, I would like to integrate my notebook with Github, including adding milestones to my calendar and receiving notifications from the calendar when someone pushes to a repository.

6.22. As a writer, I would like to keep my drafts organized with a simple file management system and events on the calendar when I update a draft.

6.23. As a writer, I would like to keep different notebook entries organized with folders/tags

Non-Functional:

Performance

To ensure our users have a positive experience using our website, our server response time must be within 600 ms.

Scalability

Because we have great ambition to make our product available to many people, we employ Nginx, known for its ability to handle large numbers of concurrent connections, and expect that it can handle up to 500 simultaneous requests. What's better, we will also try to enable short-term caching by setting up the Nginx configuration to allow more concurrent requests per second. If time allows, we may try to apply Proxmox to realize horizontal scaling by creating more than one server to support our service.

Security

For ScheduSmart, we place emphasis on customer privacy, thus designing several mechanisms to protect precious information. First of all, we will prevent SQL injection by using parameterized statements, escaping special characters, etc. Also, users' passwords will be encrypted through password encryption functions provided by the database, such as scram with sha256 so that we can shield our consumers from man-in-the-middle attacks for our users. Moreover, even on our server side, we cannot know the passwords since they are stored as ciphertext. Besides, we also need to deal with cross-site scripting when working on our front-end and back-end pages. To achieve this, we would be careful with HTML tags and configure CSP properly for our Nginx web server. What's more, we will also let the Nginx server manage a reverse proxy to protect our server. If time allows, to thoroughly ensure our web security, we will take advantage of OWASP ZAP to do penetration testing.

Usability

To make it easier for our users to get started with ScheduSmart, we have figured out several requirements for the development team to help our consumers. On the one hand, we should design interactive web tours so that users can be more familiar with our functions and know how to use them by the step-by-step instructions. On the other hand, we will design a user-friendly interface and find more than 6 participants to use our applications and share their user experience with us to help us improve our products. If time allows, we can write a step-by-step user guide on Medium to make sure that our users know how to use this application.

Maintainability

As for maintainability, developers have to use Jenkins and Git to do version control and CI/CD allowing us to efficiently maintain our web. Besides, we should pay attention to error handling so that the developer who is responsible for maintaining this application can easily know which part causes errors from the clear and concise error messages at once.