Quack Time – Productivity Assistant
Project Plan

Neal Kimchi (nk), Nikhar Ramlakhan (nr), Areyan Rastawan (ar), Abie Safdie (as)

6-04-2024 - v1.06

Table of Contents

1. Project Plan Revision History

Date	Author	Description
5-18-2024	nr	Created the initial document.
5-19-2024	nr	Completed the initial document.
5-19-2024	nk	Added the project's Gantt chart.
5-30-2024	nk	Revisions to Project Plan
6-03-2024	nk	Revisions to Project Plan
6-04-2024	nr	Finalized the Project Plan

2. Management Plan

2.1. Team organization

The team consists of Neal Kimchi, Nikhar Ramlakhan, Areyan Rastawan, Abie Safdie who are all computer science students in CS 422. Each member contributes to both technical and organizational aspects of the project. Our team is structured to ensure collaboration and expertise in all aspects of the project; therefore, all members will be iterating on various parts of the project. Nikhar Ramlakhan will also oversee project management and ensure adherence to documentation.

2.2. Work division amongst members

Tasks are allocated based on individual strengths and interests, ensuring a balanced workload and diverse skill set utilization. Nikhar and Areyan focus on system setup and backend development, while Neal and Abie handle user interface design and front-end development. Collaboration and cross-functional involvement are encouraged to foster a holistic understanding of the project.

2.3. Decision making protocols

Decisions are made collaboratively, with input from all team members. Major decisions require a super-majority vote (3 out of 4) to ensure consensus and representation of all perspectives. Major decisions will be documented and signed / initialed by all team members to ensure that a record is kept in case of any future disputes or disagreements.

2.4. Team meetings and communication

Communication is facilitated through a Discord channel and an iMessage group channel for any group discussions outside of meetings. In addition, Discord will serve as the primary platform for providing progress updates, facilitating unscheduled meetings for development assistance, and promptly informing the team of any issues that can be addressed later. iMessage, on the other hand, will be reserved for emergency situations, ensuring immediate attention to urgent matters that require immediate resolution. Members are expected to communicate respectfully and within designated hours (8:00am to 10:00pm). In-person meetings are held at the Allan Price Science Commons Library on the following dates and times:

- Sunday, 19 May | 6:00pm to 8:00pm
- Tuesday, 21 May | 6:00pm to 8:00pm
- Thursday, 23 May | 6:00pm to 8:00pm
- Sunday, 26 May | 6:00pm to 8:00pm
- Tuesday, 28 May | 6:00pm to 8:00pm
- Thursday, 30 May | 6:00pm to 8:00pm
- Sunday, 2 June | 6:00pm to 8:00pm
- Additional meetings can be organized at least 24 hours in advance.

3. Work breakdown schedule

3.1. Milestones

#	Projected Milestone	Projected Date
1.	Set up Project Repository	05/19/2024
2.	Database Design	05/21/2024
3.	Backend Development Setup	05/21/2024
4.	Frontend Development Setup	05/21/2024
6.	User Registration and Authentication	05/25/2024
7.	Task Board and Task Management	05/28/2024
8.	Pomodoro Timer Functionality	05/30/2024
9.	Break and Productivity Rating System	05/30/2024
10.	Productivity Logbook	06/31/2024
10.	Frontend-Backend Integration	06/01/2024
11.	Testing	06/02/2024
12.	Final Review and Documentation	06/03/2024
13.	Deployment	06/03/2024

3.2. Project Schedule

Following the above milestones, the project schedule can be derived as follows:

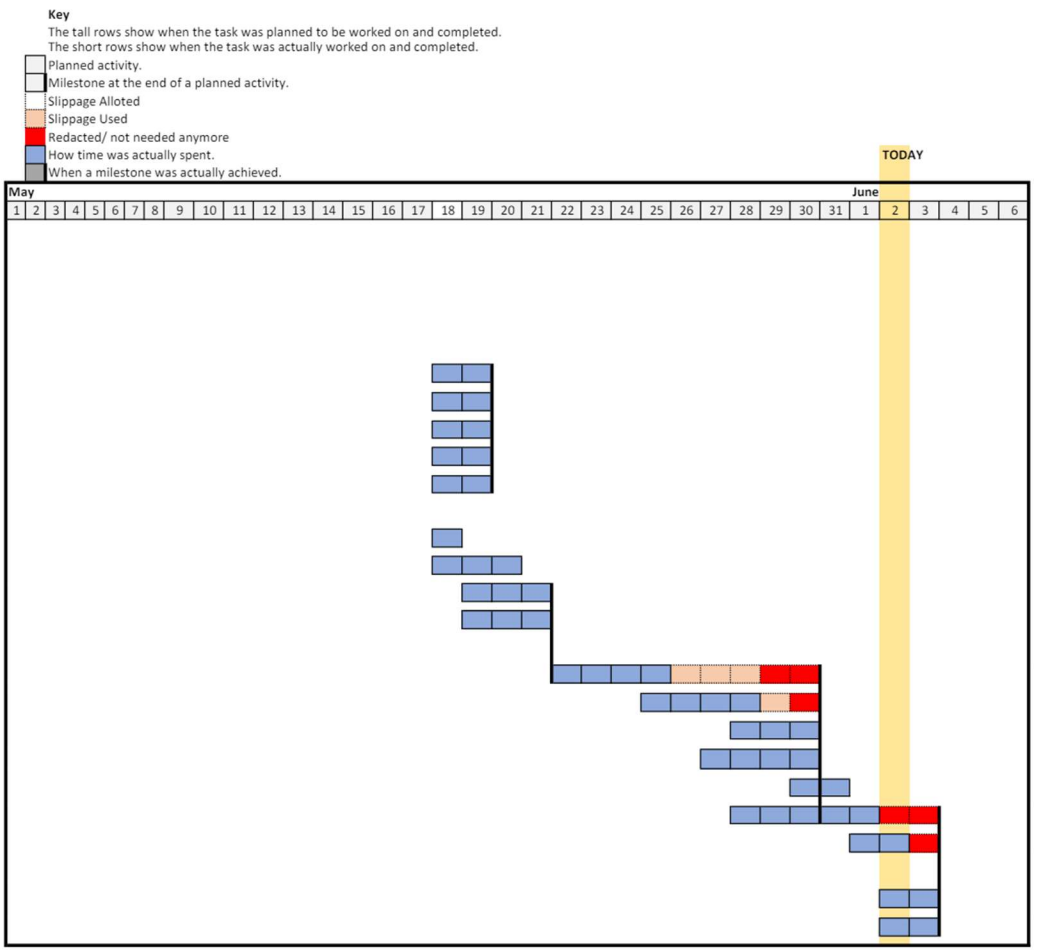
Milestone	Projected Date	Assigned Member(s)
Set up Project Repository	05/19/2024	nr
Database Design	05/21/2024	nr, ar
Backend Development Setup	05/21/2024	nr, ar
Frontend Development Setup	05/21/2024	nk, as
User Registration and Authentication	05/25/2024	nr, ar
Task Board and Task Management	05/28/2024	nk, as
Pomodoro Timer Functionality	05/30/2024	nk, as
Break and Productivity Rating System	05/30/2024	nk, as
Productivity Logbook	06/31/2024	nr, ar
Frontend-Backend Integration	06/01/2024	team
Testing	06/02/2024	team
Final Review and Documentation	06/03/2024	team
Deployment	06/03/2024	team

3.3. Project Gantt Chart

The Gantt Chart was formatted from an Excel file for this document.

Project Timeline as of: 5/18/2024
The timeline is maintained by: Neal Kimchi
Team Members: NK: Neal Kimchi
NR: Nikhar Ramlakhan
AR: Areyan Rastawan
AS: Abie Safdie

ID	Tasks and Milestones	Assigned to...	Predecessor	Expected Days
Project Planning				
	Assemble team.			0
	Initial brainstorm and assigning of tasks.	Entire Team		1
	All team members sign off on initial timeline.	Entire Team		0
Requirements Analysis				
1	Create SRS	AS,AR		2
2	Create SDS	AS,AR		2
3	Create Project Plan	NK,NR		2
4	Research Current Technologies	NK,NR		2
5	Interview "Experts"	AS,AR,NK,NR		2
Implementation				
6	Set up Project Repository	NR		1
7	Database Design	AR,AS		3
8	Backend Development Setup (MySQL)	AR,AS		3
9	Frontend Development Setup (JavaScript)	NR,NK		3
Modules:				
10	User Authentication and Registration	AR,AS		4
11	Task Board (Management System)	NR,NK		4
12	Pomodoro Timer Implementation	NR,NK		3
13	Break/Productivity Rating System	AR,AS		5
14	ProDUCKtivity log module	NR,NK		2
15	Fronend/Backend Implementation	AR,AS		5
16	Testing	NK,NR,AR,AS		2
Project Final				
17	Final Review & Documentation	NK,NR,AR,AS		1
18	Final Deployment	NK,NR,AR,AS		1



4. Monitoring and reporting

4.1. Individual progress monitoring

Individual progress will be primarily monitored through the GitHub repository and the developer log / document log markdown file. Each team member is responsible for regularly updating the developer log / document log with their contributions, changes made, and revisions. This log will serve as a central hub for tracking individual progress and documenting project developments.

Regular team meetings will be held, providing an opportunity for members to report their progress, discuss any challenges or roadblocks, and coordinate tasks. At the beginning of each meeting, members will provide updates on their individual contributions, which will be documented in the developer log / document log.

4.2. Project progress monitoring

Project progress will be monitored through various channels to ensure alignment with the project plan and objectives:

- Documentation Updates: Documentation, including the project plan, SRS, and SDS, will be continuously updated to reflect project progress and any changes or refinements made. During team meetings, documentation will be reviewed and revised as necessary to ensure accuracy and alignment with project developments.
- Communication Channels: iMessage and Discord will serve as additional avenues for minor updates, progress reviews, and ad-hoc discussions among team members. While major updates and progress reports will be documented in the developer log / document log, these communication channels will facilitate real-time collaboration and information sharing.
- Gantt Chart: A Gantt chart will be developed as part of project monitoring, providing a visual representation of project timelines, milestones, and dependencies. The Gantt chart will be updated regularly to reflect progress and any adjustments to the project schedule.

5. Build plan

5.1. System build plan

Phase 1: Setup and Foundation (5/19/2024 – 5/21/2024)

1. Project Setup
    - Set up project repository.
    - Establish version control.
    - Initialize project structure.
2. Documentation
    - Create a Project Plan.
    - Create Software Systems Requirements.
    - Create Software Design Requirements.
3. Design Phase
    - Create a draft of the user interface.
    - Design the database schema.

Phase 2: Core Functionality Implementation (5/22/2024 – 5/31/2024)

4. Backend Development
    - Implement user registration and authentication.
    - Development or managing task boards, tasks, and timed focus sessions.
5. Frontend Development
    - Develop the frontend components based on the drafted UI design.
    - Implement user interfaces for task management, Pomodoro timer, and productivity log.
6. Integration and Testing
    - Integrate frontend and backend components.
    - Conduct unit testing, integration testing, and user acceptance testing.

Phase 3: Refinement and Deployment (6/1/2024 – 6/3/2024)

7. Refinement and Bug Fixes
    - Address any issues identified during testing.
8. Final Documentation
    - Finalize Project Plan.
    - Finalize Systems Requirements Specifications.
    - Finalize Systems Design Specifications.
9. Deployment and Release
    - Prepare system for deployment to production environment.
    - Develop deployment scripts and procedures.
    - Release system to end-users.

5.2. Explanation of system build plan

1. Incremental Development: Breaking the system development into phases allows for incremental progress, reducing the risk of scope creep and enabling early feedback from stakeholders.

2. Parallel Development: Concurrent backend and frontend development accelerates the overall process and facilitates integration testing, ensuring that components work seamlessly together.

3. Testing Throughout: Incorporating testing at each phase mitigates the risk of major defects and ensures that the system meets quality standards before deployment.

4. Risk Reduction: By addressing potential risks proactively, such as technical complexity, scope creep, and resource constraints, the development team can efficiently build and deploy the Quack Time system while ensuring quality, security, and stakeholder satisfaction.

5. Progressive Enhancement: The build plan follows a progressive enhancement approach, starting with essential functionalities and gradually adding more advanced features. This incremental development strategy allows for early feedback from developers and users, enabling iterative improvements based on real-world usage and feedback.

5.3. Risks and risk reduction strategies

1. Technical Complexity: The project may face technical challenges during implementation, potentially leading to delays.
    - Risk Reduction: Conduct thorough research and prototyping to identify and address technical uncertainties early in the project. Maintain a skilled and diverse team capable of addressing a range of technical issues.

2. Scope Creep: There is a risk of expanding project scope beyond initial requirements, affecting timelines and resource allocation.
    - Risk Reduction: Establish clear project objectives and requirements upfront. Implement a change management process to evaluate and prioritize new feature requests. Regularly review project scope and adjust as necessary while considering resource constraints.

3. Resource Constraints: Limited resources such as time, budget, or personnel may impact project delivery.
    - Risk Reduction: Develop a realistic project plan with achievable milestones and allocate resources effectively. Monitor resource utilization and adjust the plan as needed to optimize productivity and mitigate burnout.

6. Acknowledgements

The content of this document is inspired by the Project 1 Evaluation Criteria provided by Prof. Anthony Hornof.

This document template is built and derived from SRS/SDS template provided by Prof. Anthony Hornof. Additionally, it builds on a document developed by Stuart Faulk in 2017, and on the publications cited within the document, such as IEEE Std 1016-2009.

7. Annexures

7.1. Development and Document Log

This log, initially in Markdown format, has been formatted for the purpose of this document.

GUI:

5/11:
- Initial Landing page created.

5/12:
- Login page made.

5/14:
- Landing page was built out with multiple panels for each section.
- Login Interface completed.

5/16:
- Created and completed Sign-Up page.
- Completed timer panel.

5/20:
- Integration with database, allowing the task boards and entries to populate the landing page for returning users.
- Integrate setters, to set new task entries in the database.

5/22:
- Made the landing page dynamically load itself with current logged in user.
- Created a custom timer.

5/23:
- Added HOW-TO tab.

5/24:
- Added the logs page.
- Logs page pulled information from the database and showed it to the user.

5/25:
- Added rating system for users completing a timer.
- Reformatted the logs page.

5/28:
- Redesigned some aspects of UI on landing page.

Server/Database:

5/12:

- Initialized mysql database
- Installed mysql-connector-python.

5/14:

- Created database, and tables for users
- Created functions to change information for users.

5/16:

- Created task board entries (top part of hierarchical structure)

5/17:

- Made user specific entries for tasks and task boards.

5/20:

- Added functions for task board entries.
- Created tasks and functions for tasks
- Integrated with frontend.

5/22:

- Added time to each task, to allow the timer to change the value

5/24:

- Integrated with logs page.

7.2 Meeting Minutes

Saturday, 11 May | 6:00pm to 7:30pm | Allan Price Science Commons

- General design of the ARA
- Setting up a Github repository and means of communication
- Assessment of the team and allocation of roles based on individuals strengths and weaknesses.
- Set up initial parts of project

Sunday, 12 May | 4:00pm to 5:45pm | Allan Price Science Commons

- Reviewing the drafted Project Plans, SRS and SDS.
- Creating static and dynamic models for our components we anticipate using.
- Compiling the documents for our initial submission.

Tuesday, 14 May | 6:00pm to 8:00pm | Discord

- Initialized database
- Created various objects in database
- Completed login interface

Monday, 20 May | 6:00pm to 8:00pm | Allan Price Science Commons

- Integrated database with various UI sections.
- Created tasks object in database, with corresponding functions.

Friday, 24 May | 6:00pm to 8:30pm | Discord

- Created the logs page.
- Integrated database functions to populate logs.

Tuesday, 4 June | 6:00pm to 8:00pm | Allan Price Science Commons

- Finalized tasks needed to be completed.
- Finished project