

Study Buddy

(Version 0.1 Beta)

- **PROJECT NAME:** Study Buddy
- **PHASE:** Detailed Requirements
- **DATE:** 10/21/2022
- **PREPARED BY:** DNM

Table of Contents

1.0 PROJECT PLAN	03
1.1 PURPOSE/DESCRIPTION OF PROJECT	03
1.2 SCOPE OF PROJECT	03
2.0 PROJECT DESCRIPTION	03
2.1 PROJECT STAKEHOLDERS	03
2.2 PROJECT GOALS	03
2.3 CONSTRAINTS & RESTRICTIONS	04
2.4 ASSUMPTIONS & DEPENDENCIES	04
3.0 PROJECT REQUIREMENTS	05
3.1 FUNCTIONAL REQUIREMENTS	05
3.2 USER INTERFACE REQUIREMENTS	05
3.3 SYSTEM REQUIREMENTS	06
3.4 ERROR HANDLING	06
4.0 USE CASES & SCENARIOS	07
4.1 REMOVE TASK	07
4.2 COMPLETE TASK	07
4.3 ADD TASK	07
4.4 ADD GROUP	08
4.5 SORT	08
5.0 SIGN-OFFS	08
6.0 GLOSSARY	10
7.0 APPENDIX	11
7.1 DIAGRAM UI	11

1.0 - Project Plan

1.1 DESCRIPTION/PURPOSE OF PROJECT:

- To create a visual interface to aid students with scheduling and time management
- To categorize events based on relevant data, in an inclusive-intuitive manner
- To provide the user with multiple sorting options based on relevant data

1.2 SCOPE OF PROJECT:

- Development of a web application
- Web app to run on Google Chrome, Mozilla Firefox, Internet Explorer or any other modern browser
- Work with an online database
- Hosting a simple website

2.0 - Project Description

2.1 PROJECT STAKEHOLDERS

Project Stakeholders include:

- Students enrolled in any post-secondary education program

2.2 PROJECT GOALS

Shared goals for all stakeholders of this project:

- The product should have a user-friendly interface and intuitive controls
- The information displayed needs to be consistent and easy to read

- The final product in its first iteration should have enough features to aid students in scheduling tasks

2.3 CONSTRAINTS & RESTRICTIONS

CONSTRAINT	ISSUE/SOLUTION
User has multiple computers, and does not want to have separate instances of an application on each computer.	Issue -> Stakeholder uses multiple devices. Solution -> Operate database and application through servers, service is accessed through a web browser.
Tier one GUI contains a colour coded legend to categorize groups	Issue -> Stakeholders may have visual impairments. Solution -> Addition of features to adjust colour coded text fields, ensure colours are always labelled.

2.4 ASSUMPTIONS AND DEPENDENCIES

- Users connected to the product are using a modern web browser without plugins that prevent JavaScript from being run.
- Google Cloud Services must be up for the database to work. (A Google cloud based database service called Firebase is used to store, sort and modify data).

- Azure services that run Github to host a website/domain must be up for the application to run. (Github is being used to host the web app.)
- But Client and Server must always have access to the internet for the application to work properly.
- The client must have a modern internet browser to access the website. (IE6 will not be supported)

3.0 - Project Requirements

There is only one primary user group in the system; post-secondary students.

3.1 FUNCTIONAL REQUIREMENTS

- The student should be able to add tasks, or groups to the to-do list.
- Each task needs to have the option to enter a description, type, group, and due date
- The web application will have an option to sort tasks based on their due-date.
- User will be able to select a colour when adding a group to the to-do list
- Default Types will exist as follows: Assignments, Tests, Projects, Labs, and Other
- A checklist field will exist that allows objects to be marked and then sorted as either completed or incomplete.
- A legend, indicating the colours of existing groups will be displayed on the homepage

3.2 USER INTERFACE REQUIREMENTS

- There needs to be holistic error handling for data that is prompted from the user
- Options, controls, defaults, and the display of information must be concise, intuitive, and explicit

3.3 SYSTEM REQUIREMENTS

- User's system must have a modern web browser that supports the latest version of JavaScript, and HTML5.
- User must have an internet connection

3.4 ERROR HANDLING

- User input error handling will consist of;
 - Handling user input of dates that are in the past relative to current date, if they have not entered a date, or the date is not in the specified format
 - If the name of the task is left empty, an error message will be displayed and the task entry to the list will not be processed.
 - If the user tries to add a pre-existing group an error message will be displayed reading "This group name ' group name ' is already in use".
- Functional error handling will consist of;
 - Handling the case that the database is not accessible. An error message will be displayed reading "Sorry, database is offline, please try again later"
 - If user input data is corrupted, an error message "Database contains invalid data - data is corrupted."

4.0 - Use Cases & Scenarios

- Remove task

User	System
User clicks the remove task button	System prompts user for confirmation of remove
User confirms the request	System displays that the task has been deleted

- Complete Task

User	System
User clicks the complete task button.	System displays that the task has been moved to the completed section.

- Add Task

User	System
User clicks the add task button.	Pop-up window is brought up, "add task" is displayed, a text input box is displayed for task type, task group and task due date.
User clicks the ok button.	Displays "Task has been added" and adds the task.

- Add group

User	System
User clicks add group button.	A popup appears, "add group" is displayed, a text box for the name of the group appears, and a colour wheel to select the colour of the group appears.
User clicks the ok button.	"Group has been added" is displayed, system adds group with colour choice to database.

- Sort

User	System
User clicks on the tasks heading.	The tasks column on the main page is sorted according to date due, and any task that has been added since the last sort is added to the display of tasks.

5.0 - Sign-offs

Signature: *David Botero*

Date: 10/21/2022

Print Name: David Botero

Title: Student

Role: Test Lead, UI Co-Lead,

Signature: *Joshua Lade* Date: 10/21/2022

Print Name: Joshua Lade

Title: Student

Role: UI Lead, Test Co-lead

Signature: *Hayden Lister* Date: 10/21/2022

Print Name: Hayden Lister

Title: Student

Role: Contact Person,
Project Management

Signature: *Abhay Parashar* Date: 10/21/2022

Print Name: Abhay Parashar

Title: Student

Role: Version Control Lead,
Dev Lead,
Project Management Co-lead

Signature: *Robert Stewart* Date: 10/21/2022

Print Name: _____ Robert Stewart

Title: _____ Student

Role: _____ Documentation, Dev Co-Lead

6.0 - Glossary

Group: A category that groups different tasks together who share a common goal. (e.g. a course)

Type: A particular kind of action that needs to be accomplished by a task.

Defaults: Assignments, Quizzes, Projects, Exams, Labs

Task: A specific action or activity that needs to be accomplished by a specific time.

7.0 - Appendix

task completion mark

options, but for now it's just **remove**

<div><input type="checkbox"/> Add Task ← Adds new task</div>						User Name
✓	Task	Group	Type	Due Date	...	
					✗	<div>Groups</div> <div><input type="checkbox"/> Add Group ← adds a new group</div> <div>the list of groups</div>
	↑ the task name	↑ the group of the task	↑ the type of task eg: test, assignment...etc	↑ date the task is due		

FIG_UI