

Software Requirements Specifications

For

Study Tracker

Version 1.0

Prepared By

Kim Pham

Table of Contents

Background	4
Planning of the software project.....	4
Data collection methods	4
Data analysis	5
1. Introduction	7
1.1 Purpose.....	7
1.2 Document Conventions	7
1.3 Intended Audience and Reading Suggestions	7
1.4 Identification and purpose of the software product	8
1.5 References	9
2. Overall Description	9
2.1 Product Perspective.....	9
2.2 Product Functions.....	10
2.3 User classes and characteristics	10
2.4 Operating environment	11
2.5 Potential security vulnerabilities	11
2.6 User documentation.....	11
2.7 Assumptions and dependencies.....	11
3. Scope.....	13
3.1 Items within scope.....	13
3.2 Items not in scope	13
4. Constraints	15
4.1 Legal factors	15
4.2 Social factors	15
4.3 Technical factors.....	15
4.4 Usability factors.....	16
5. Requirements	17
5.1 User register – REQ1.....	17
5.2 User login – REQ2	17
5.3 User logout – REQ3.....	18
5.4 Functional timer – REQ4.....	18
5.5 Timer and break notifications – REQ5	19
5.6 Add subjects or tasks – REQ6	19
5.7 Delete subjects or tasks – REQ7	19
5.8 Recording length of time – REQ8.....	20
5.9 Report of time spent on subjects or tasks – REQ9	20
5.10 Tutorial – REQ10	21
5.11 Settings – REQ 11	21
6. External Interface Requirements	26
6.1 User Interfaces	26
6.2 Hardware interfaces	31
6.3 Software Interfaces	31

6.4	Communication interfaces	31
7.	Other Non-functional Requirements	32
7.1	Performance requirements	32
7.1.1	Reliability and responsiveness – NREQ1	32
7.2	Safety requirements	32
7.2.1	Safety – NREQ2	32
7.3	Security requirements	32
7.3.1	Security – NREQ3.....	32
7.4	Software quality attributes.....	32
7.4.1	Legibility – NREQ4	32
7.4.2	Intuitiveness and consistency – NREQ5	33
7.4.3	Accessibility and usability – NREQ6.....	33
7.4.4	Attractiveness – NREQ6	33
7.4.5	Affordance – NREQ7.....	33
7.4.6	Tolerance and robustness – NREQ8	33
7.4.7	Maintainability and portability – NREQ9.....	34
Appendix A:	Research results	35
Appendix B:	Context diagram	39
Appendix C:	Data flow diagram	39
Appendix D:	Use case diagram.....	40
Appendix E:	Pseudocode	41
Appendix F:	Flow chart.....	46
Appendix G:	User interface mockups	47
Appendix H:	Sitemap	52
Appendix I:	Data dictionary and Object descriptions	53
Appendix J:	Glossary	57
Acknowledgements:	58

Background

Planning of the software project

‘Study Tracker’ is a software solution that aims to solve the problem of students overworking and losing track of the time. Prior to the commencement of this project, many students have shown the tendency to leave tasks to the last minute, and then rushing to complete it. To try fix this problem, I have done research into time management strategies, focusing specifically on strategies used by students to help them utilize their time more efficiently. One particular technique that seems to be popular is the concept that individuals are able to work more efficiently and stay productive if they only focus on one task and work in shorter intervals with frequent breaks (see *Acknowledgements* for more information). I’ve personally tried using it and found that while it was effective at preventing myself from overworking and procrastinating, at times I would’ve liked for more statistics and data regarding the time tracking process and for the process to be automated and simplified. Before and during the planning of the software project, I have asked others on their thoughts of the concept and any suggestions on how to improve it.

Data collection methods

Several data collection methods were used but the dominant collection methods used were surveys and interviews.

Surveys are less expensive and time-consuming than other data collection techniques and allows for the collection of data on a broad range of things from a range of different people. By collecting data from a broad range of people, this reduces the frequency of biased data and increases the range of different opinions and responses. Surveys can also have a target audience which allows for the data collected to be relevant to the topic.

In surveys and questionnaires, questions can be structured to expect quantitative data by using multiple choice or scaled responses and can also be structured to collect qualitative data by using open-ended questions which allows for a wide range of possible questions to be asked. The possibility for quantitative data allows for responses to be processed and analyzed easily, while the qualitative data allows for more detailed responses.

The second method of data collection used are interviews. Interviews, while more time-consuming than surveys, will provide more detailed data than other forms of data

collection. Similar to surveys, a wide range of questions can be asked, and a target audience can be selected. By being present with the participant, a previous answer to be further explained and expanded on, resulting in more detailed questions to ask and more rich responses from real, potential users. They also are more reliable than online surveys and questionnaires as the anonymous nature of surveys can result in some individuals not filling it in properly or can result in responses from individuals that are not part of the target audience.

A minor data collection method that was used were polls on social media. Similar to surveys, these polls allowed for an easy way to quickly gain the perspective of potential users on a particular topic due to its simple one question format. Although the social media platform used (Amino) had a restriction on the length and number of choices, it was easy to get many responses from the target audience (students) as the polls were posted in a community about school life and studying.

The data gathered from these data collection methods will be analyzed to figure out the more about the target audience - what problems they encounter when trying to study (procrastination, feeling overworked etc.), their personal preferences regarding studying, how effective their study habits are for them etc. From this, the solution will be adjusted to suit their needs.

Data analysis

From the surveys and polls, it was found that from of those who have used the Pomodoro® technique from which Study Tracker takes inspiration from (286 respondents), 72% of respondents agreed that the study technique had worked for them. Out of these 72%, 28% said that they used the original 25 minutes working period and 5 minute breaks, while the other 44% said that they use a modified version with either longer/shorter working periods and/or longer/shorter breaks. This suggests that the technique does work for most students, but an adjustable timer is preferred.

Through interviews, the main reasons why using a timer doesn't suit some users was found to be the strictness of the work and break periods. A suggestion that was proposed was that instead of a countdown timer, a count up timer could be used instead so that the pressure of the timer is relieved but the main function of timing the length of the subject/task is still present. This suggests that both a countdown and a count up timer must be implemented in the solution to cater to all users.

Regardless of if they used the timer or not, out of 328 respondents from the surveys and polls, the most popular length of time spent studying before taking a break was from 30 minutes to 1 hour (42%), followed by under 30 minutes (26%) and from 1 hour and 30

minutes to 2 hours (13%). Out of the 365 respondents from the surveys and polls, the most popular break length was 10-15 minutes (24%), 5-10 minutes (20%), followed by 15-20 minutes (13%). This data will determine the default or recommended timer lengths the software solution will use as well as determine the acceptable ranges of these times.

Out of the 152 respondents from the survey, 82% said that they use a laptop while studying. Given that Study Tracker is a web-based software solution that is designed to run on computers, this proves that the software will be suitable for most users. 91.4% of respondents also said that they felt overworked or tired when studying to any degree (13.9% always felt overworked, 24.5% felt overworked most of the time, and 53% sometimes felt overworked). This proves that the problem of feeling burnt out after studying for too long that the software is trying to solve is an existing problem that many face. 59.9% of respondents stated that they feel like not enough time is available to complete tasks, with mainly procrastination and being distracted as being major causes of time lost.

40.1% of respondents stated that they work on a single task at a time, 25.7% stated that they multitask, and the other 33.6% stated that they use a combination of both. This could suggest that it is important to cater to those who multitask and those who work on tasks one at a time, making it easy for both groups to do so within the software. 75.7% of respondents also stated that they don't balance their workload, prioritising or spending more time on certain subject/tasks than others. This suggests that the software solution must make users aware of the time spent on subjects/tasks so that the user can evenly distribute their time for each subject/task.

The implemented timer within the software solution is intended to instill an urgency within the users to get them to focus and concentrate when working. 83.6% of respondents stated that having a time pressure such as a deadline or time limit helps them to work better (21.7% always worked better, 31.6% worked better most of the time and 30.3% worked better sometimes). This suggests that the timer going to be implemented in the solution will be helpful to most.

61.9% of respondents stated that they needed to be reminded of due dates of tasks (13.2% always, 19.1% most of the time, 29.6% sometimes). This suggests that a due date reminder may need to be implemented within the software.

For the processed data that is referred to in this section, refer to *Appendix A: Research Results*.

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to provide a detailed description of the requirements needed for the software solution, Study Tracker. This document will describe the purpose of the solution and each of its intended features. It will also cover the constraints placed onto the solution, its scope and how the solution will process data. This document is intended to be used as a reference by both stakeholders and developers of the software as the project is being developed, tested or maintained.

This software requirements specification is based on the version 1.0 of the software.

1.2 Document Conventions

This document contains some terminology that may be unfamiliar to readers. Refer to *Appendix J: Glossary* for a list of these terms with their definitions.

1.3 Intended Audience and Reading Suggestions

This document is intended for all individuals involved in the Study Tracker project such as users, developers, testers, project managers and documentation writers.

Readers who are wanting to view a brief overview of the software and its features should refer to *Section 1 – Introduction, Section 2 – Overall Description, and Section 3 – Scope*. They may also want to refer to *Background* for further information about the origin and planning of the software solution.

Readers interested in the details of the features presented in the software should refer to *Section 5 – Requirements*. They may also be interested in *Section 3 – Scope and Section 4 – Constraints*. Readers interested in the software's features should also refer to *Section 6 – External Interface Requirements* for further technical details of the requirements such as the user interface and may be interested in *Section 7 – Non-Functional Requirements* for non-technical aspects of the software such as performance and software qualities that are important to users such as attractiveness and usability.

Overview:

Background: Provides the origins and planning of the software solution as well as providing information on the data collection methods used.

1. Introduction: Provides a brief overview of the software solution and describes the structure of this document.
2. Overall description: Introduces the software solution including some of its functions, its user characteristics, the environment it will operate in, and lists any assumed factors used within this document.
3. Scope: Provides information about what the final software solution will and will not do as well as listing any features that may provide opportunities for further development.
4. Constraints: lists any factors such as legal, social, technical, usability factors that limit the project scope.
5. Requirements: Provides detailed descriptions of Study Tracker's features.
6. External Interface Requirements: Provides a visualization of the program and requirements related to hardware, software and networking.
7. Other nonfunctional requirements: Provides other requirements that apply to factors such as performance, safety and security.
8. Appendix: all diagrams, tables and other information that was used or referred to in this document

1.4 Identification and purpose of the software product

The project Study Tracker is a software solution that aims to provide a simple way to help users stay organized and maximise their productivity based on the concept that individuals work more effectively in short bursts rather than in long sessions. By tracking the time spent on tasks or subjects whilst encouraging the habit of taking breaks through the usage of a count up or count down timer, this prevents users from being overworked and increases their awareness about the time spent in order for the user to concentrate on the task at hand and to help them develop better time management skills.

As students are increasingly utilizing portable devices when studying (see *Background: Data Analysis*), this solution provides a modern take on the traditional time tracking method, switching the physical timer and paper for a digital, automated version instead. A report of the time tracked made by the software solution also saves the individual's time by calculating and compiling these statistics automatically allowing the user to analyze their statistics themselves if they wish.

Through these features, the software solution will make studying more effective and efficient for users, meeting their needs, whilst remaining easy to understand and use.

1.5 References

This document is adapted from the given VCE Software Development SRS template and from the IEEE Standard for Software Requirements Specification (IEEE 830-1993).

2. Overall Description

2.1 Product Perspective

Study Tracker is a new, self-contained software solution that aims to help users keep track of the time to avoid overworking and procrastination whilst working or studying. The software solution is intended to be run on a web-based environment with access to a web and database server.

Refer back to *Section 1.4 – Identification and Purpose of the Project* for more information about the software solution.

Although this solution is not a component that is part of a larger system, it features the usage of both client-side and server-side functionalities, both of which are covered in this document.

On the client-side of the solution, being the user interface, users are able to interact with the software and use its features such as recording time spent on individual subjects or tasks with the assistance of timers. This data regarding the time spent on each user's individual subjects and tasks will need to be stored in a database, which relates to the solution's server-side functionalities. The server-side component of the solution will be responsible for the interactions with the database and managing the storage of data within it. Both the client and server-sides of the solution will need to interact with each other for the software solution to function properly.

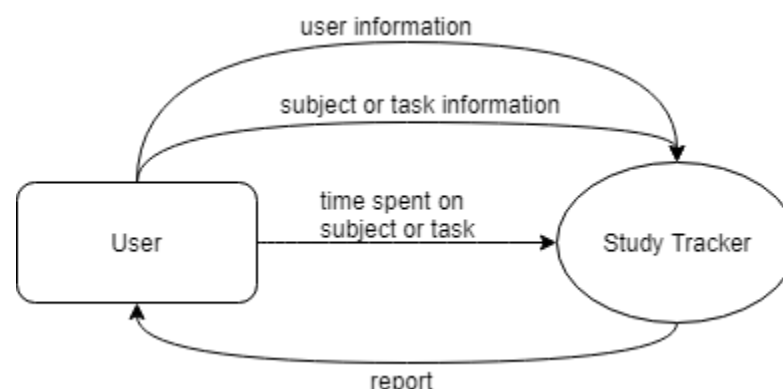


Figure 1: Context diagram

2.2 Product Functions

The main function of this solution is to allow users to record the length of time spent on a subject or task, and to summarise the data in a report.

To do this, an onscreen timer is used to record the length of time spent on a subject or task, make the users aware of the time spent and to remind the user to take breaks. The user should be able to choose from a count-up or a count-down timer and the timers should be adjustable with start, pause and reset buttons. Along with this, the user should be able to input their own subjects and tasks and choose the subjects or tasks they want to work on. The length of time recorded on these subjects and tasks should be displayed to the user in the form of a report page.

For more details into these functions, refer to *Section 3.1 – Items within Scope* and *Section 5: Requirements*.

2.3 User classes and characteristics

Although this solution is intended for students who live in Australia and study VCE subjects, other students who don't live in Australia and/or study VCE and others who are not students but are looking for a simple way to develop time management skills are also suitable for this solution. Within the project scope, the solution is designed to support only one type of user, therefore there is only one user class. Due to this, every user will have access to all the solution's features to track time spent on subjects or tasks.

As the intended users are VCE students, the solution will appeal to adolescent users living in Australia who are somewhat knowledgeable and experienced in using technology. Even so, the solution will still be as intuitive as possible to cater to users who are unexperienced. The solution will also feature simple Australian English that most will be able to understand, as some students, such as non-native English speakers or individuals with impairments such as those with dyslexia, have trouble reading and understanding difficult and complex words. The solution presented also will minimize the usage of small print wherever possible and will ensure that all of its features are easy to use and access. This will cater to those who are visual or motor impaired. Alongside this, there will also be a help page implemented within the solution to ensure that all users are able to understand and use all of Study Tracker's features effectively.

2.4 Operating environment

Study Tracker is a web-based application that will use HTML, PHP, MySQL databases and some JavaScript, coded using Notepad++ and being tested on a XAMPP local server. The web-based nature of the solution allows for the software solution to be able to run on any device (although the solution is designed for usage on computers and portable computers only), on any web search engine and on any operating system provided that there is a web and a database server available.

Due to the scope of the solution and the small databases, the solution will most likely be run on desktop or portable computers using local servers only.

2.5 Potential security vulnerabilities

Since the final solution will be run on local web servers accessing a database that is not password protected due to the scope of the solution, this could pose some potential security vulnerabilities. An individual with access to the database could easily view, compromise and/or exploit the data stored within the databases. These deliberate threats could be prevented by implementing more security features on the software such as password protection, encryption of data, using intrusion detection such as audit logs and using secure firewalls. Due to the scope of the solution, only usernames and user passwords have been established to prevent against deliberate threats.

2.6 User documentation

Whilst the solution is designed to simplify the time tracking process and to be as intuitive as possible, some users will still require additional information about the solution and how to use it. To approach this problem, Study Tracker will provide users this information through a simple tutorial. The Study Tracker tutorial (help page) will take the user through the functions of the solution and teach them how to effectively utilize them.

For more detail on the tutorial (help page) see *Section 5.10: Tutorial – REQ10*.

2.7 Assumptions and dependencies

An assumption about the product is that it will only be used on desktop or portable computers as the solution is not designed to be optimized for other devices with smaller screen sizes such as tablets or mobile phones. Another assumption is that all desktop and portable computers have the same screen ratios. If the assumed screen sizes are different from the actual screen sizes, the appearance of the final solution would not match the designs specified in this document.

It is also assumed that the computer used by the user has access to a web and database server and has the hardware resources available to run the software solution. The also database is assumed to have the capacity to hold the data. If these assumptions are not met, the solution would not work as intended or would not work at all.

The development of this solution depends on the available resources given. A change in the period of time allocated for development or the lack of online resources such as tutorials that could assist with the development of the solution could result in the specified requirements not being met.

3. Scope

3.1 Items within scope

The following list provides a brief description of Study Tracker's main features and functions that are essential to the functionality of the software solution:

The solution should feature:

- A user register, login and logout page
 - o Allows the user to create a new account
 - o Enables the user to access their personal account and access to the solution's features
- A functional and adjustable timer
 - o Users can select between a count down or a count up timer
 - o Allows users to track their time on a subject or task
- Break notifications
 - o Alerts user of timer running out
 - o Reminds the user to take breaks
- A way to add and delete user inputted subjects or tasks
- A way to record the length of time spent on subjects or tasks
- A way to view the length of time spent on subjects or tasks for a day
 - o Summarized in a report
- A tutorial/help page
- A settings page
 - o Allows users to manage their timer preferences, account and subject/tasks

For more details into these functions, refer to *Section 5: Requirements*.

3.2 Items not in scope

Due to project constraints or irrelevance, the current solution should not:

- suggest what subject or task the user should do next
- suggest how users should manage their time
- suggest when the user should do tasks
- suggest how long a user should spend on a task
- assume that a user has completed tasks and automatically record the length of time spent on them without the user's validation. The timer should just be used as a guide

or reference that allows users to be aware of the time spent working. All recorded times should be confirmed/determined by the user.

- implement a report containing statistics that extend beyond the current date (no past records, no weekly, monthly or yearly reports)
- include password protected databases - in real world situations this would have to be implemented but because of technical constraints, placing a password on MySQL databases is not recommended.
- Be run on other servers besides local web and database servers

Although the final solution created will not have these functions due to time and technical constraints, these following functions may be added later as an opportunity for further development:

- Implementing a timer that automatically and accurately records the time spent on subjects/tasks
- Implementing individual to-do-lists for each subject or task
- Synching users so that users can collaborate and/or share subject/task lists or timers
- Being able to organize tasks on the to-do-list by moving them around and prioritizing them
- Implementing a calendar
- Implementing personal study schedules so that users can see their progress through the day
- Implementing inspirational quotes of the day
- Being able to synch school time tables or calendars to the solution to include important date reminders
- Being able to set date reminders as well as a date countdown for due dates, test/exam dates etc.
- Being able to view detailed, weekly, monthly or even yearly reports, complete with graphs that describe the user's productivity
- Implementing different timer notification sounds and ambient music
- Implementing a break timer that runs after the regular work timer has finished
- Implementing better security features such as password protected databases and the encryption of data
- Including more than one user type such as an administrator user who can view other user's subject/task lists and reports

4. Constraints

4.1 Legal factors

Although it is not within the scope of the project, in real world situations, all data collected from users would need to be secured and protected, done through the usage of password protected databases and other security features to prevent unauthorised access to private information and to not breach any legal laws regarding the privacy of individuals. Implementing these features would take time and would increase the cost associated with the project.

4.2 Social factors

Social factors that limit the scope of the solution include the level of expertise of the users have with technology, the level of expertise of the developers have with developing software, the time available to develop the solution, the availability of equipment and resources and making sure that users with special needs, such as impairments or disabilities, are catered to. These social factors will determine the number of and the complexity of the features the solution will contain.

The level of expertise users and developers have will decide the complexity of the features as users are not capable of using features that are too complex and developers are not capable of implementing them. Ensuring that users with special needs will be catered to will also cost some time.

The time available is a major constraint. The shorter time available for the development of the solution's features, the less likely it is for that feature to be complex in nature. Time also dictates the number of features able to be implemented.

The availability of equipment and resources also links back to the time constraint. If equipment and resources such as the device used to develop the software, or the internet used to access online resources were not available, time could be lost as the development of the software solution relies on these resources being available.

4.3 Technical factors

Technical factors that will limit the scope of the solution include the speed of processing required, the capacity of data transfer and memory storage, the availability of equipment, the type of equipment used and the compatibility with existing operating systems or operating environments. These technical factors will also determine the number and the complexity of the features the solution will contain.

The speed of processing and capacity of data will determine the responsiveness of the solution. If the responsiveness is slow, the project scope must be restricted and some features of the solution must be cut out to ensure that the solution will still function effectively.

The capacity of memory storage, the availability of equipment, the type of equipment used and its compatibility with existing operating systems or operating environments will determine the types of features the solution will have. For example, the lack of memory will restrict the project's scope and certain types of operating environments will have different features which can be used in the development of the solution.

4.4 Usability factors

Usability factors include the usefulness of the solution - the user's needs for specific requirements in the solution, and the ease of use of the solution – the requirements of the user interface to ensure that it is intuitive, simple to use, responsive and reliable.

For example, from the data collection (refer to *Background – Data Analysis* for more information) most potential users (44%) would prefer to use a modifiable timer rather than to stick to the standard 25 minute work time as suggested by the Pomodoro® technique from which Study Tracker takes inspiration from. A count up timer should also be used to cater to those who dislike the strictness or pressure from using a countdown timer. The data collection also suggests that most users multitask (59%). Ensuring that users have the option of a countdown or a count up timer, that the timer implemented in the software is fully adjustable, that users are able to easily switch between subjects or tasks, and between pages and that all of these features are easily accessible to all users will cost time.

5. Requirements

5.1 User register – REQ1

Priority: medium

When opening the software solution, the user should be presented with a login page. On the login page, there will be an option for the user to create a new account located alongside the main user login prompt if they do not have an account already. After clicking on this link, the user will be redirected to the register page where they are prompted to enter in their details through text inputs, such as a username and a password. After submitting their details, a new account will be created, and their details will be stored in a 'users' database.

5.1.1 Stimulus/Response Sequences

Preconditions: Software solution is launched and user does not have an account

1. User is prompted to enter in user details – a username and a password
2. User details sent to the server and stored in the database
3. Registration is completed and user is taken to the main page.

Postconditions: A new user account is created and user is able to access the software solution

5.2 User login – REQ2

Priority: high

When opening the software solution, the user should be presented with a login page which prompts them to enter their username and password through text inputs. After entering their username and password, the data will be compared to the user details in the database. If their username and password match the records in the database, they will be allowed access to the software solution. If not, the user will be prompted to try again.

5.2.1 Stimulus/Response Sequences

Preconditions: Software solution is launched.

4. User is prompted to enter in user details – a username and a password.
5. Inputted user details are checked with the user records stored in the database.
6. If user details match, they are taken to the main page. Otherwise, they are prompted to try again.

Postconditions: User is able to access the software solution.

5.3 User logout – REQ3

Priority: low

Users should be able to logout on any page of the software solution, accessed through the logout link. When logged out, the user should not be able to access the software solution without logging back in.

5.3.1 Stimulus/Response Sequences

Preconditions: The user is already logged in.

7. User clicks the logout link.
8. User is no longer allowed access to the software solution and is redirected to the login page.

Postconditions: The user is not allowed access to the software solution without logging back in.

5.4 Functional timer – REQ4

Priority: medium

An adjustable, live timer should be implemented in the software solution to allow the user to see how long they have spent on a task or to see how long they have left until a break. Due to technical and time constraints, this timer will be just will not record the time users spend on their subject/task, it will just be used as a guide for users.

The timer preferences should affect the type of timer presented to the user. If the countdown timer's length is adjusted manually via the settings page, the number input will have to be validated. The user should be presented either with a count down or a count up timer, depending on the user's preference, with functional start, pause and reset buttons.

5.4.1 Stimulus/Response Sequences

Preconditions: The user has selected a subject or task to work on.

1. User is presented with a timer with a start, pause and reset button.
2. User presses the start button to start the timer which starts to count up/down depending on the user's preference.
3. User confirms that they are finished using the timer.

Postconditions: The user is able to use a timer.

5.5 Timer and break notifications – REQ5

Priority: low

When a timer runs out, a notification should be displayed or a sound should be played depending on the user's preferences.

The notification will continue to be displayed or the sound will continue playing until the user stops it through clicking a button.

5.5.1 Stimulus/Response Sequences

Preconditions: A timer has run out of time.

1. A sound plays or a notification is displayed.
2. User confirms that they are aware of the notification/sound.

Postconditions: The user is reminded to take a break.

5.6 Add subjects or tasks – REQ6

Priority: high

Each user should be able to have their own personal set of subjects and tasks. Each subject or task should be inputted by the user through a text input box. Each manual input should be validated. When the user adds a new subject or task, it should be added to the 'subjects' table in the database.

5.6.1 Stimulus/Response Sequences

Preconditions: User is on the main page or on the settings – manage subjects/tasks page.

1. User enters in a subject or task through a text input.
2. User input is validated before being inputted into a database.
3. User inputted subject or task is shown on the main page.

Postconditions: The user able to add their own subjects or tasks

5.7 Delete subjects or tasks – REQ7

Priority: medium

When a user deletes a subject or task through a button click, a notification should be displayed to alert the user of the action. If the user confirms that they wish to proceed to delete the selected subject or task, the selected subject or task should be removed from the database. Users should be able to cancel this action.

5.7.1 Stimulus/Response Sequences

Preconditions: The user has at least one subject or task.

1. User selects a subject or task to delete.
2. A notification alerts the user of their action.
3. If the user confirms that they wish to delete the subject or task, the subject or task is removed from the database.

Postconditions: The user is able to delete a subject or task.

5.8 Recording length of time – REQ8

Priority: high

After adding their own personal set of subjects or tasks, the user should be able to view these subjects or tasks and should have the ability to select the subject or task they want to focus on.

When the user has finished working, the user gives a confirmation that they have worked on a subject or task for a certain period of time using the implemented timer (REQ4) as a guide. Any manual number inputs will need to be validated to check for reasonability before being converted to minutes and stored in a database. The length of time recorded should be associated with the individual user and the subject or task worked on. This will allow for the solution to display the total length of time worked to individual users later.

5.8.1 Stimulus/Response Sequences

Preconditions: The user has at least one subject or task.

1. User selects a subject or task.
2. Timer is displayed.
3. After working, the user confirms that they have finished.
4. Manual inputs of time spent on the subject or task are validated before being stored in a database.
5. User is redirected to the main page.

Postconditions: Time spent on a subject or task is recorded.

5.9 Report of time spent on subjects or tasks – REQ9

Priority: high

Given that the user has recorded the length of time spent working, the user will be able to view the total length of time spent working and will be able to view the length of time spent on individual subjects or tasks in a day.

A report of these statistics will be displayed on a separate page. These statistics will be retrieved from the database, summarizing all the database entries associated with the user and their subjects and tasks for one date.

5.9.1 Stimulus/Response Sequences

Preconditions: The user has recorded the time spent for at least one subject or task.

1. User clicks on the 'view statistics' link.
2. User is presented with a report containing statistics regarding the time spent working.

Postconditions: The user is able to view a report of their time spent on subjects or tasks.

5.10 Tutorial – REQ10

Priority: low

The Study Tracker tutorial link should be accessible on the main page of the software solution. When clicked, the user is redirected to a tutorial page that should provide users with the information needed to effectively utilize the software's features if they are unsure about how to use the software. This should be done through simple annotated screenshots of the software accompanied by short descriptions.

5.10.1 Stimulus/Response Sequences

Preconditions: User is on the main page.

1. User clicks on the tutorial link and is redirected to the tutorial page.
2. User is presented with the Study Tracker tutorial.

Postconditions: The user is able to view a tutorial of how to use the software.

5.11 Settings – REQ 11

Priority: medium

The settings link should be accessible from the main page of the software solution. When clicked, the user should be redirected to the settings page where they are able to view and make changes to their current settings. These settings could include changing the

user's name, username, password, preferred timer type and timer length, timer notification preferences, as well as including the ability to edit and add/delete subjects or tasks.

Changes regarding timer length preferences should be done through a select dropdown list, changes regarding timer type preferences should be done using radio buttons and changes regarding timer notification preferences should be done using a checkbox. If the user intends to change their name, username or password, or to edit or delete specific subjects or tasks, the user should be redirected to a separate page to make these changes. On this page, multiple text inputs should be used. All inputs must be validated.

At the bottom of all settings pages, a final 'save changes' button should be implemented for the user to confirm their changes. These preferences will be stored a database.

5.11.1 Stimulus/Response Sequences

Preconditions: User is on the main page.

1. User clicks on the settings link and is redirected to the settings page.
2. User is presented with their settings and makes necessary changes.
3. User clicks on the 'save changes' button and their preferences are updated in the database records.

Postconditions: The user is able to change timer preferences and manage their account or subject/tasks.

Below in *figure 2*, the data flow diagram visually represents the movement of data throughout the system.

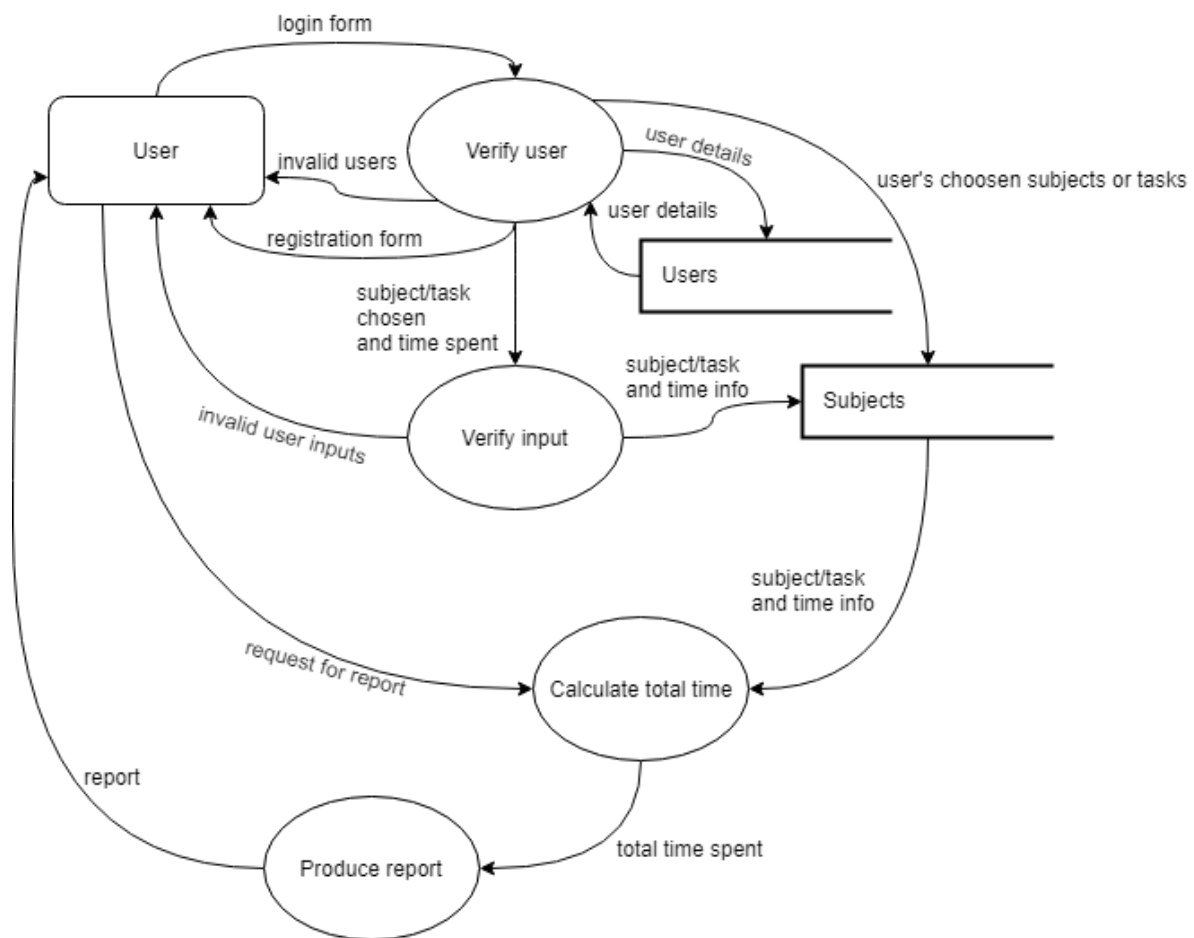


Figure 2: Data Flow Diagram

In *figure 3*, the use case diagram represents the functions of the software solution and defines how actors interact with the system.

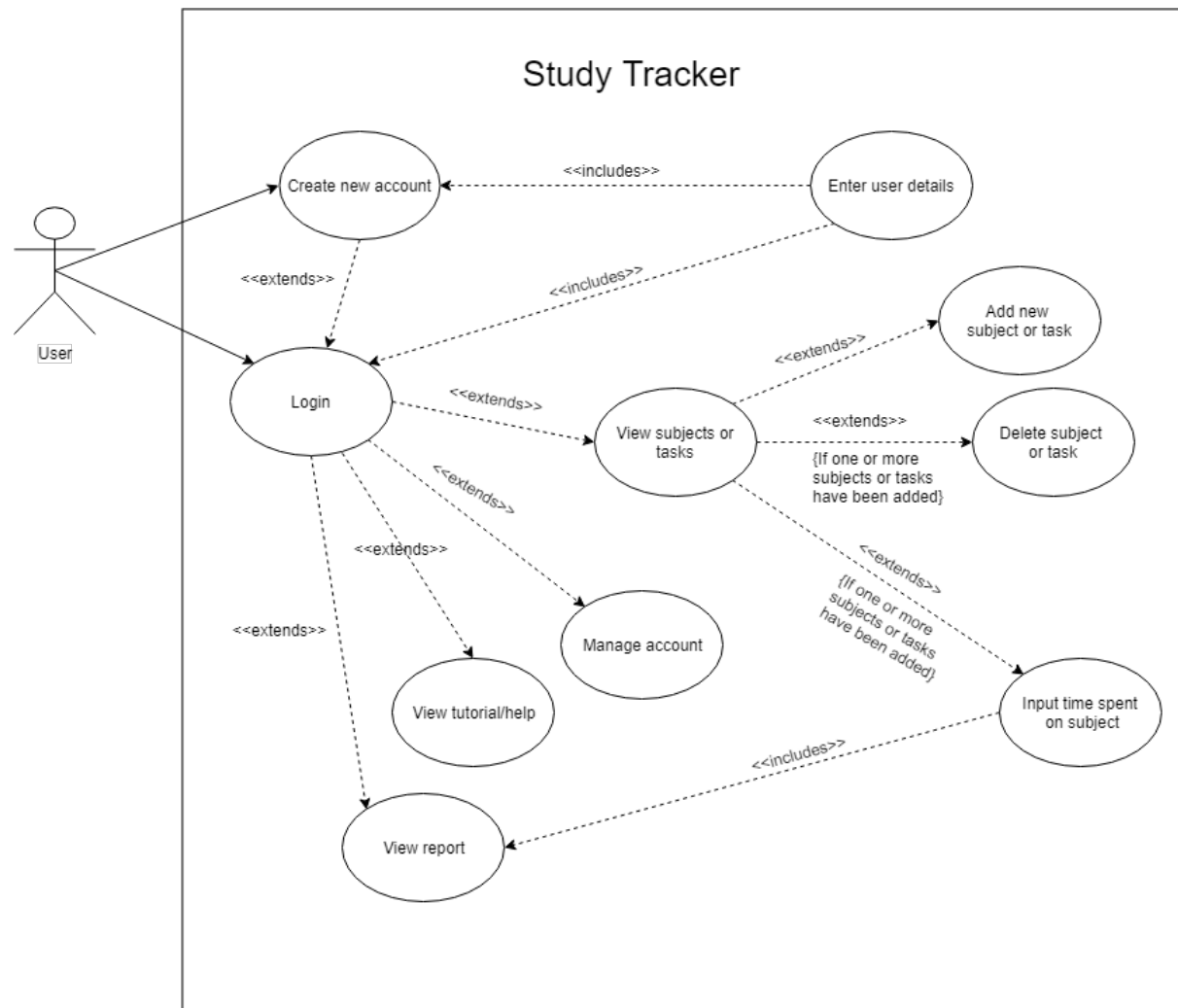


Figure 3: Use Case Diagram

For the written algorithm representation of the logical sequence of processing steps required for the solution, refer to *Appendix E: Pseudocode*

In *figure 4*, the flow chart visually represents the sequence of processes undertaken within the software solution.

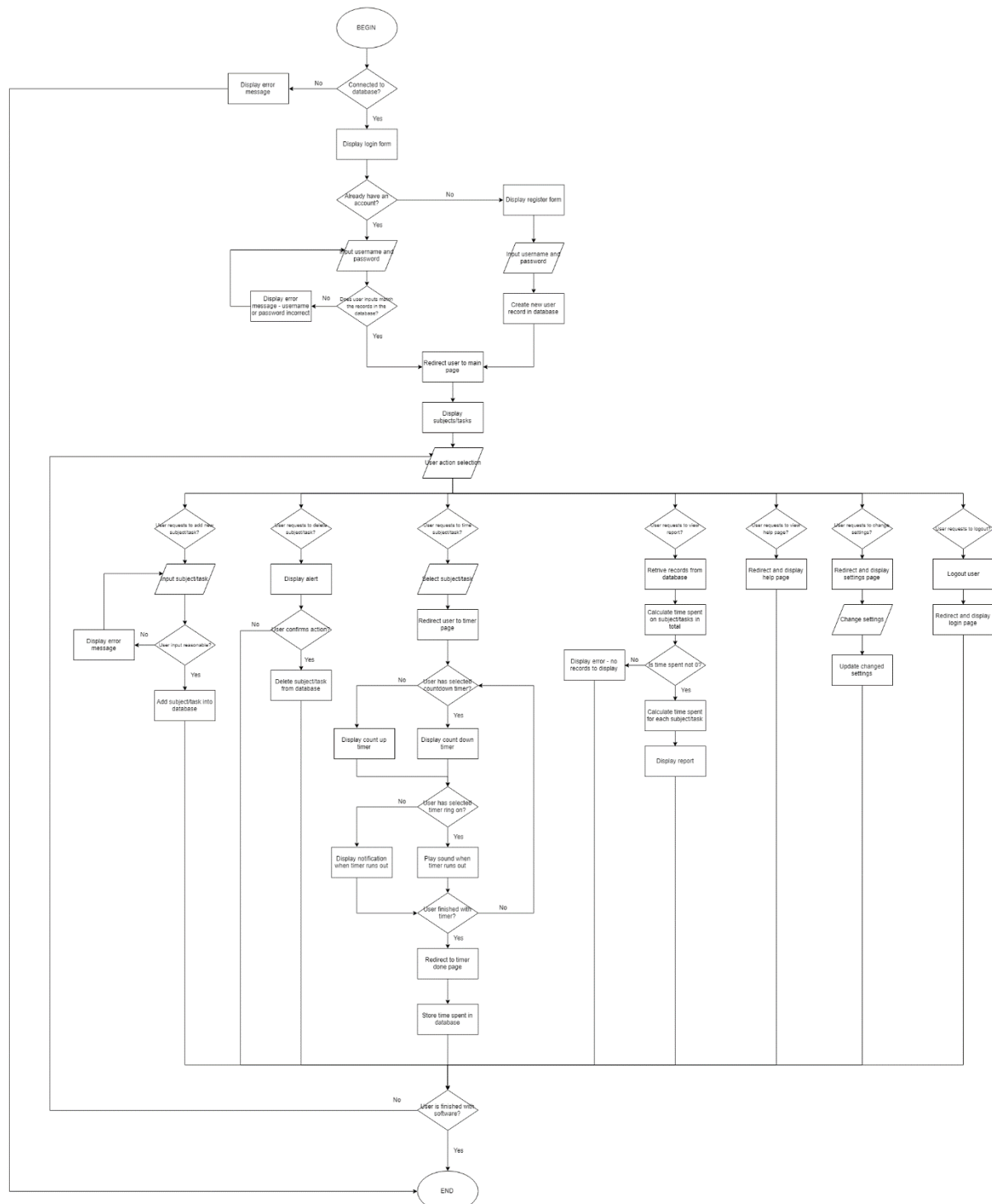
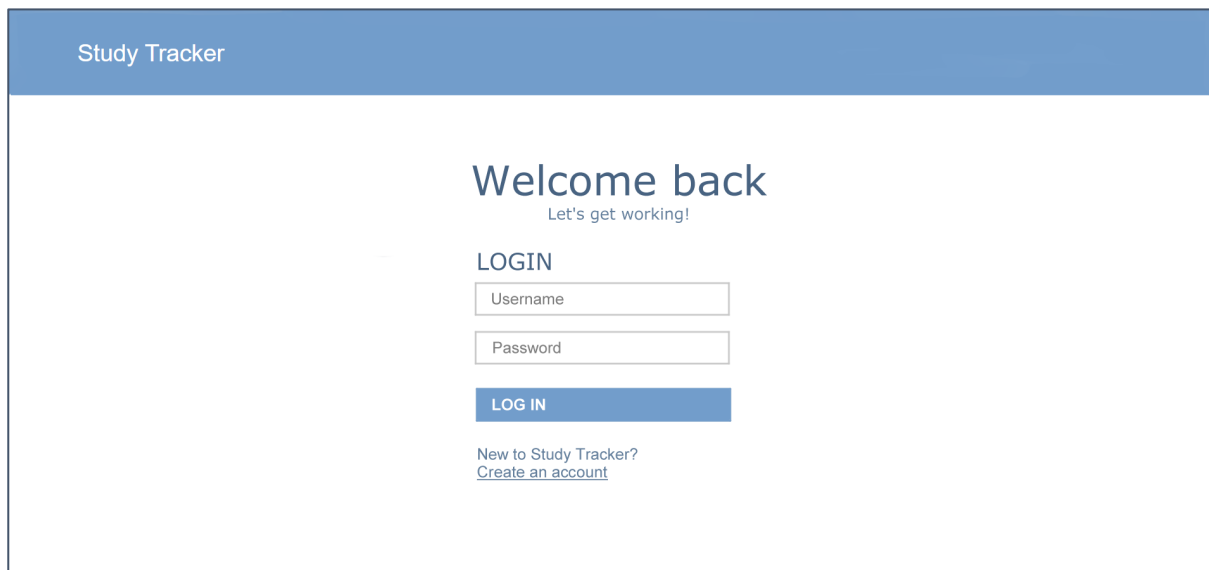


Figure 4: Flow chart

6. External Interface Requirements

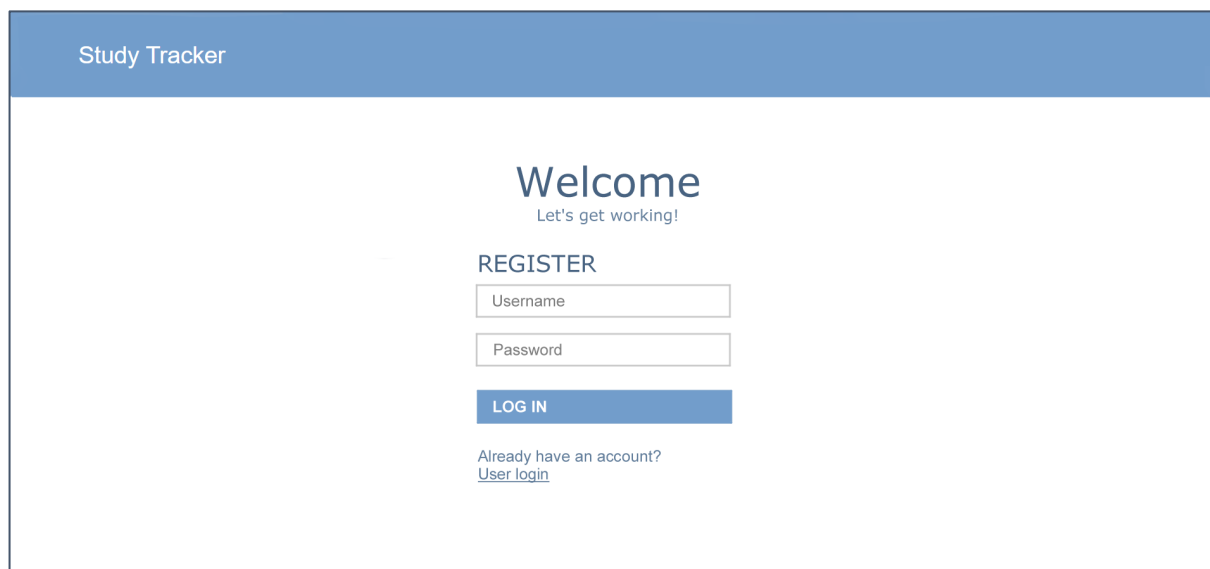
6.1 User Interfaces

When opening Study Tracker, the user should be presented with the login page and prompted to input in a username and password (*figure 5*). If they have not registered an account, they should click on a link under the login prompt which redirects them to the register page (*figure 6*).



The screenshot shows the 'Study Tracker' login page. It features a blue header with the text 'Study Tracker'. Below the header, the text 'Welcome back' is displayed in a large, dark blue font, followed by the subtitle 'Let's get working!' in a smaller, lighter blue font. A 'LOGIN' section contains two input fields: 'Username' and 'Password'. Below these fields is a blue button labeled 'LOG IN'. At the bottom of the login section, there is a link that reads 'New to Study Tracker? Create an account'.

Figure 5: Login page



The screenshot shows the 'Study Tracker' register page. It features a blue header with the text 'Study Tracker'. Below the header, the text 'Welcome' is displayed in a large, dark blue font, followed by the subtitle 'Let's get working!' in a smaller, lighter blue font. A 'REGISTER' section contains two input fields: 'Username' and 'Password'. Below these fields is a blue button labeled 'LOG IN'. At the bottom of the register section, there is a link that reads 'Already have an account? User login'.

Figure 6: Register page

After registering or logging in, the user is redirected to the main page (*figure 7*). On the main page, the user is able to add, delete subjects and tasks, choose a subject or task to work on, view their report, access their settings or log out.

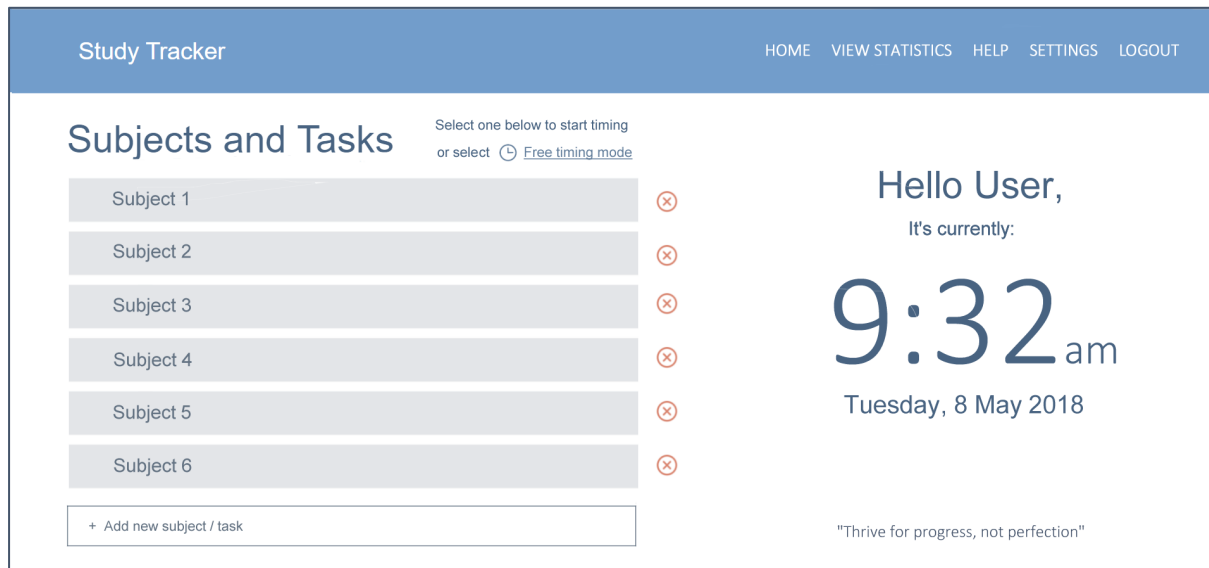


Figure 7: Main (home) page

When the user chooses a subject or task to work on, they should be redirected to a new page where a timer is displayed (*figure 8*). When they are finished they should confirm and manually input the time spent on the subject or task if necessary (*figure 9*).

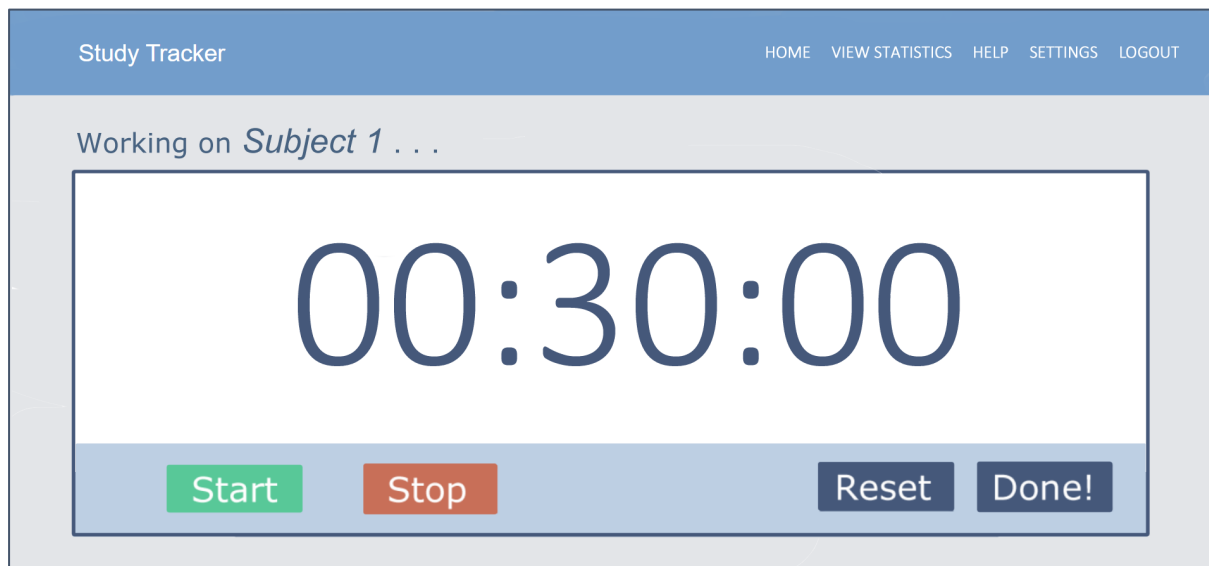


Figure 8: Timer page

The screenshot shows the 'Study Tracker' application interface. At the top is a blue navigation bar with the title 'Study Tracker' on the left and links for 'HOME', 'VIEW STATISTICS', 'HELP', 'SETTINGS', and 'LOGOUT' on the right. Below the navigation bar, the main content area has a light gray background. It starts with the text 'Working on *Subject 1* . . .'. Below this is a white rectangular box with a thin blue border. Inside this box, the question 'Did you work on *Subject 1* for 30 minutes?' is followed by a green button labeled 'Yes'. Below the question, the text 'If not, please input the time spent:' is followed by an empty text input field. At the bottom of the white box, there is a light blue horizontal bar containing two buttons: 'Go back to timer' and 'Done!'.

Figure 9: Timer done page

When the user views their report containing statistics regarding the length of time spent on subjects or tasks, they should be redirected another page to view it. (*figure 10*).

The screenshot shows the 'Study Tracker' application interface. At the top is a blue navigation bar with the title 'Study Tracker' on the left and links for 'HOME', 'VIEW STATISTICS', 'HELP', 'SETTINGS', and 'LOGOUT' on the right. Below the navigation bar, the main content area has a light gray background. It starts with the text 'View statistics'. Below this, the text 'Today you worked for: 2 hours and 30 minutes (150 min) in total' is displayed. Below that, the text 'You spent:' is followed by a list of items: '30 minutes on Subject 1', '60 minutes on Subject 2', '15 minutes on Subject 3', and '45 minutes on Subject 4'. At the bottom right of the main content area, there is a blue button labeled 'Go back'.

Figure 10: Report (view statistics) page

Users are also able to view a help page that provides information about Study Tracker and how to use its functions. (figure 11)

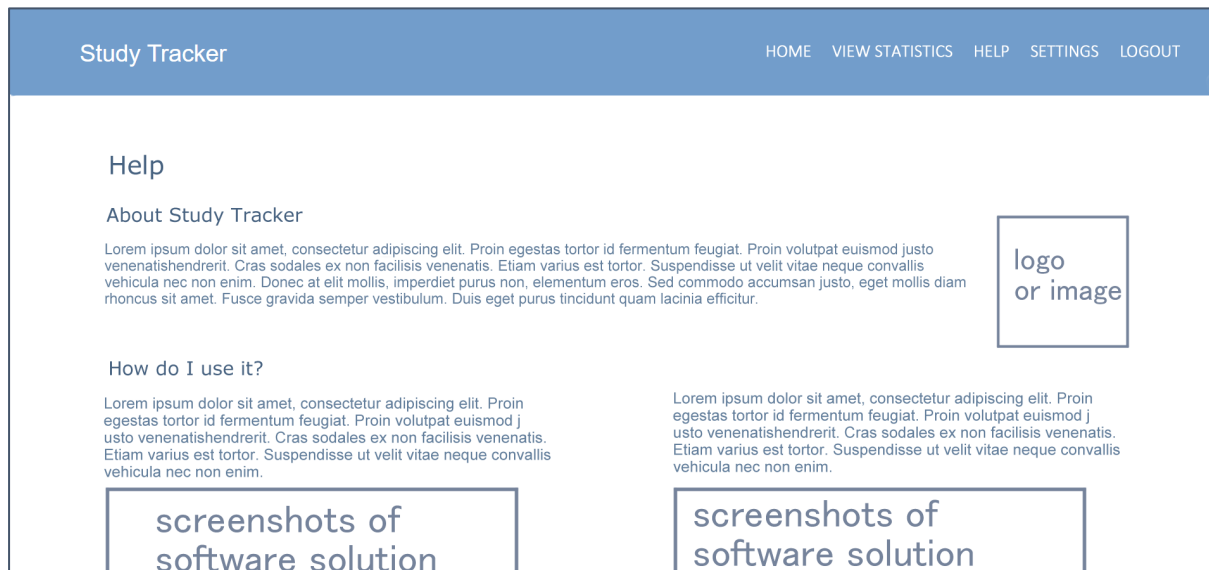


Figure 11: Help page

When the user accesses their settings, they should be redirected to the settings page where they are able to change their user details and timer preferences (figures 12, 13 and 14).

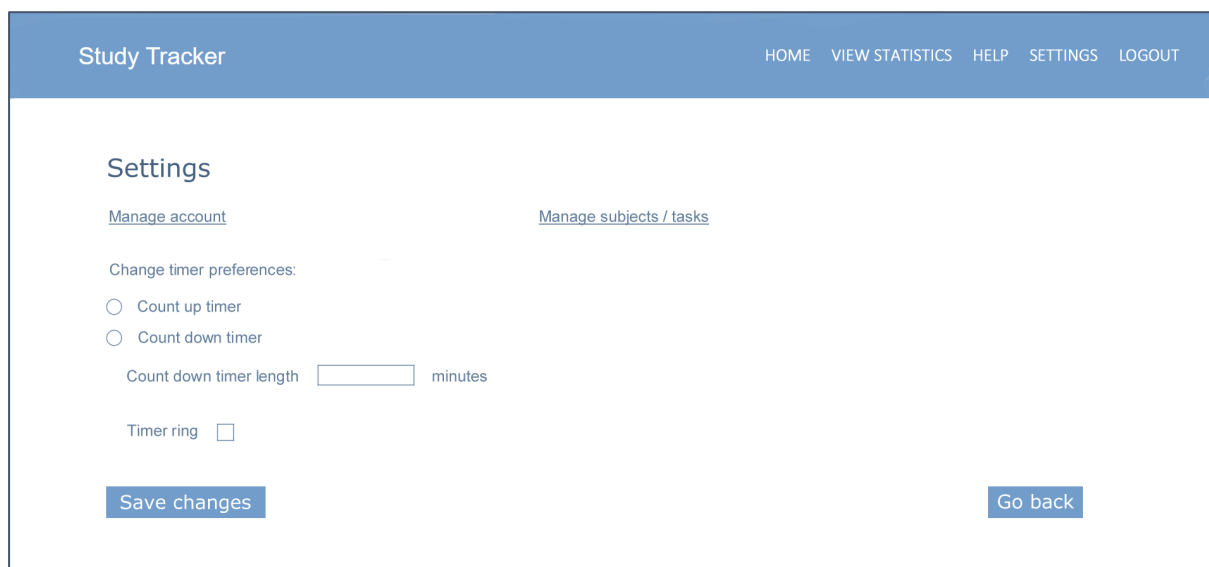


Figure 12: Settings page

The screenshot shows the 'Settings - manage account' page. At the top is a blue header with 'Study Tracker' on the left and navigation links 'HOME', 'VIEW STATISTICS', 'HELP', 'SETTINGS', and 'LOGOUT' on the right. The main content area has the title 'Settings - manage account'. Below the title are four input fields: 'Username:', 'Password:', 'Confirm new password:', and 'Display name:'. The 'Display name:' field has a small note below it: 'This the the name that appears on your home page.' To the right of the input fields is a red link 'Delete my account'. At the bottom are two blue buttons: 'Save changes' and 'Go back'.

Figure 13: Settings – manage account page

The screenshot shows the 'Settings - manage subjects/tasks' page. At the top is a blue header with 'Study Tracker' on the left and navigation links 'HOME', 'VIEW STATISTICS', 'HELP', 'SETTINGS', and 'LOGOUT' on the right. The main content area has the title 'Settings - manage subjects/tasks'. Below the title is a instruction: 'Enter in the textbox to change the subject/task's name and use the delete button to delete the subject/task.' Below this is a list of six subjects, each with a text input field and a red 'X' delete button. The subjects are labeled 'Subject 1' through 'Subject 6'. To the right of the list is a text input field with the placeholder '+ Add new subject/task'. At the bottom are two blue buttons: 'Save changes' and 'Go back'.

Figure 14: Settings – manage subjects/tasks page

In *figure 15*, the sitemap below shows how these pages are linked to each other.

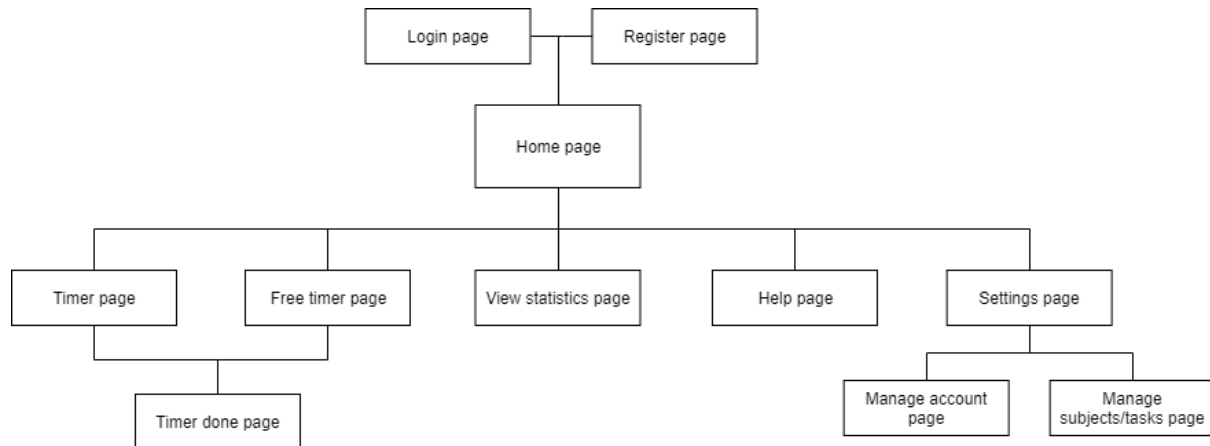


Figure 15: Sitemap

6.2 Hardware interfaces

Study Tracker is a web-based software solution and thus does not require any hardware interfaces. However, the software does require for a web and a database server to be installed for it to run.

6.3 Software Interfaces

Study Tracker communicates with a database to access and store information gathered from its users. This information could include user details such as usernames, passwords, and user preferences. Other information stored in the database includes subject or task details such the name of the subject or task and the length of time spent on it.

Refer to *Appendix H: Data dictionary and Object descriptions* for more information.

6.4 Communication interfaces

Study Tracker requires a web browser to launch the software solution on. The communication that occurs between the clients (from the web browser) and the web server uses the HTTP (Hypertext Transfer Protocol) to exchange or transfer hypermedia. The communication that occurs between the webserver and the MySQL database uses the MySQL protocol.

7. Other Non-functional Requirements

7.1 Performance requirements

7.1.1 Reliability and responsiveness – NREQ1

The software solution should meet user's expectations and process/perform all of the requirements specified in Section 5 in under 3 seconds (not counting the time taken for the user to interact with the solution). There should be little to no delay between the user's actions and the software's responses.

7.2 Safety requirements

7.2.1 Safety – NREQ2

There should be no possible loss, damage or harm that could come from the usage of the software solution. It should not affect data stored outside of its servers and any other software installed on the desktop or portable computer and should not damage the device and any of its internal components.

7.3 Security requirements

7.3.1 Security – NREQ3

It is assumed that only users with a valid username and password are able to access their account and view their information. Besides that, there is no further user identification implemented on the software solution.

As there is no password protecting the database, an individual could view, compromise and/or exploit the data stored within it if they had access to the database. However, Study Tracker doesn't store any personal information from their users (except for a name, which is optional) so it shouldn't be a major issue.

7.4 Software quality attributes

7.4.1 Legibility – NREQ4

All text should be clear and easy to read for the users with a good contrast between the background colour and the text colour. The font and size of the text must be easy to

read and appropriate for the solution. All text must also be as concise as possible to not overwhelm user interface. This will allow for those with visual impairments to use the solution with more ease.

7.4.2 Intuitiveness and consistency – NREQ5

All icons used in the user interface should be self-explanatory and universally understood. The placement of elements within the user interface must be consistent, logical, intuitive and must cohere to standard web design conventions (eg. Always having the menu at the top of the screen). A few select text fonts and a single colour scheme should be used throughout the solution.

7.4.3 Accessibility and usability – NREQ6

All features implemented in the software solution should be as accessible (easy to locate) and intuitive (easy to use) as possible. All of these features must cater to the user's needs; software developers should be aware of the skill level of users and any impairments they may have. These features should also function as expected by the users with little to no delays.

7.4.4 Attractiveness – NREQ7

All elements presented within the user interface must be visually appealing, with a simple set colour scheme used. The user interface must be pleasant to use and look at and must cater to the target users.

7.4.5 Affordance – NREQ8

Any important decisions that could be made by the user such as deleting subjects or tasks (REQ7, Section 5.7) must be confirmed twice by the user. The first confirmation occurs when the users clicks a button to perform the action and the second time occurs after the software alerts the user about their actions. During the alert, the users have the ability to cancel their action.

7.4.6 Tolerance and robustness – NREQ9

When the software solution is used badly, occurring from incorrect usage of the software or incorrect inputs into the software, the software must not crash. This could be achieved through the validation of all user inputs and through ensuring that the software continues to run even when used poorly.

7.4.7 Maintainability and portability – NREQ10

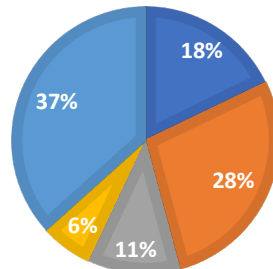
The software solution's code should be written in a way that allows for easy maintenance, further development and implementation of new features.

Study Tracker should be portable with all operating systems and web browsers, given that there is a web and a database server for the software to run on.

Appendix A: Research results

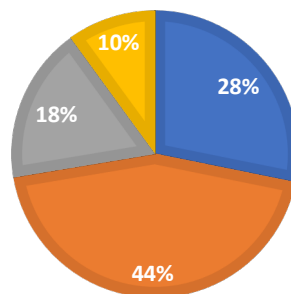
TOTAL THOUGHTS TOWARD TIMER

■ Works ■ Works, uses modified version ■ Doesn't work ■ Unsure ■ Never tried

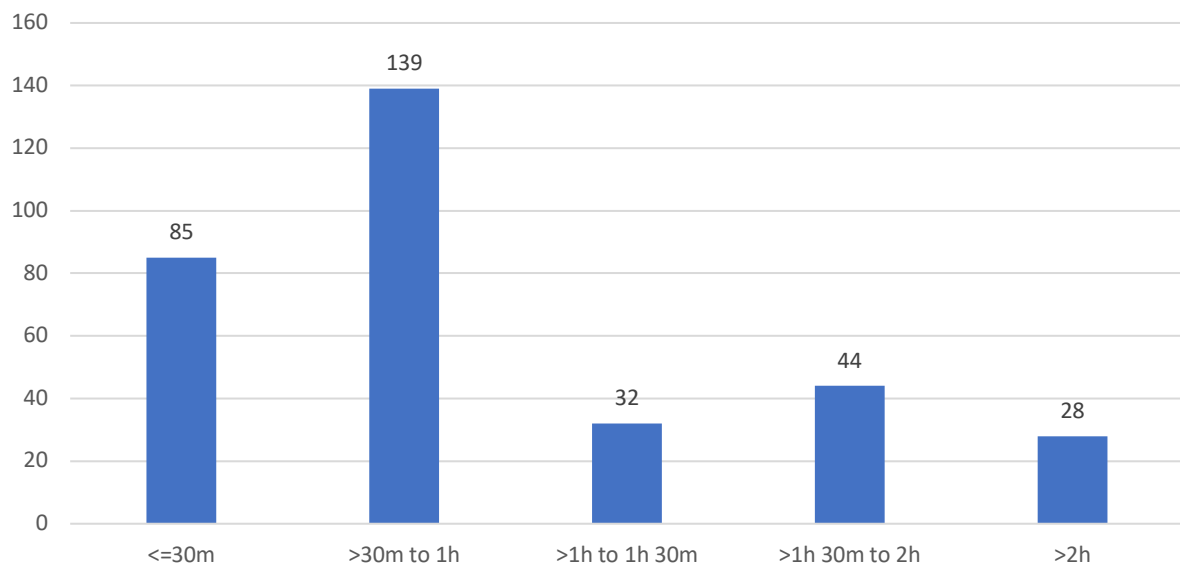


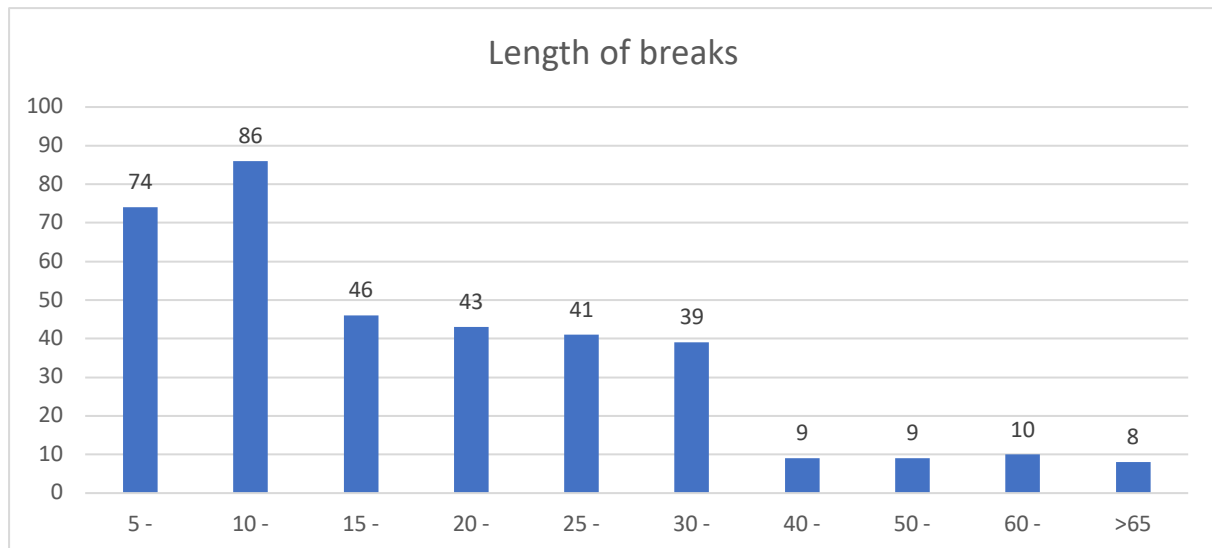
THOUGHTS FROM THOSE WHO HAVE USED THE TIMER

■ Works ■ Works, uses modified version ■ Doesn't work ■ Unsure



Length of study before break

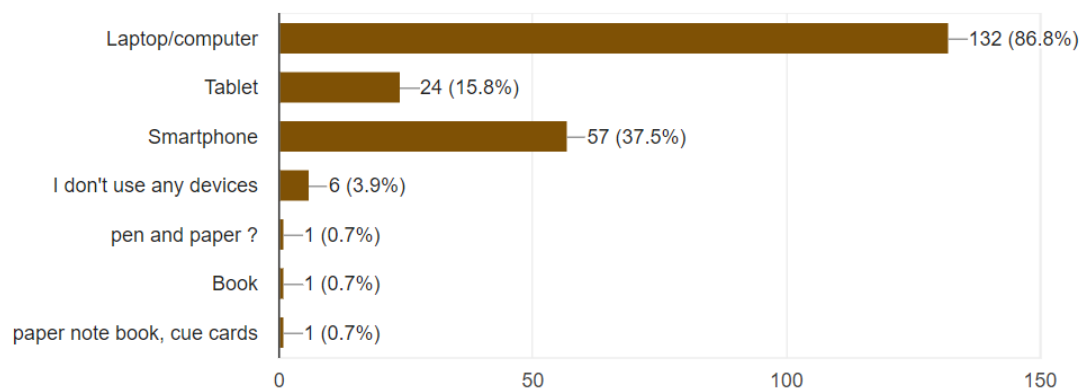




When studying, what devices do you use?

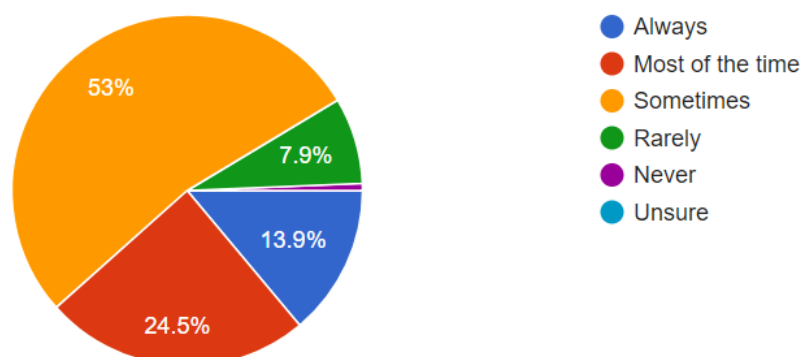


152 responses



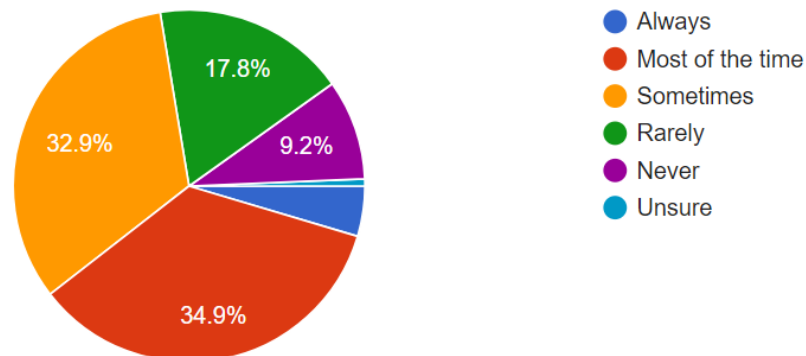
Do you feel overworked/tired when studying?

151 responses



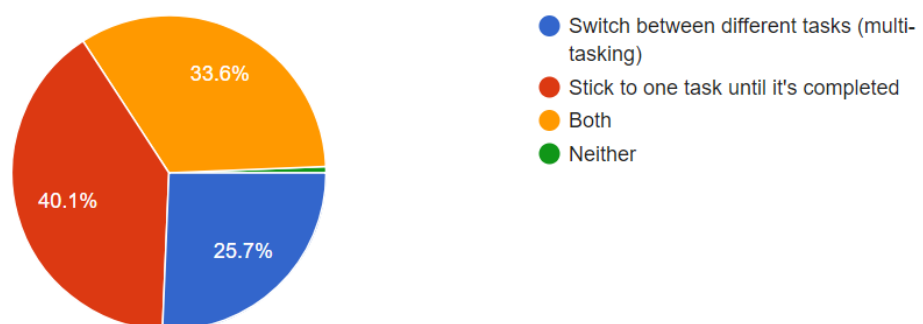
Do you feel like you have enough time to complete tasks?

152 responses



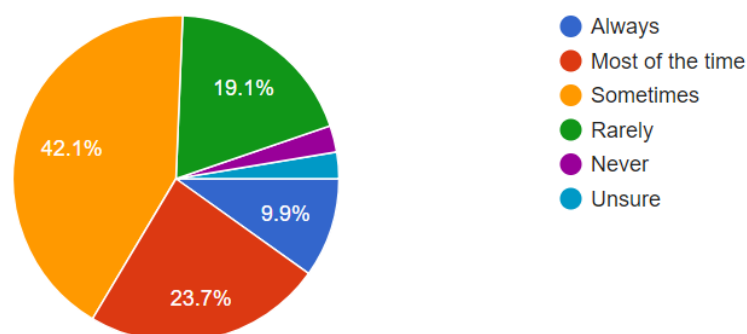
Do you tend to switch between different tasks or do you stick to one task until you've completed it?

152 responses



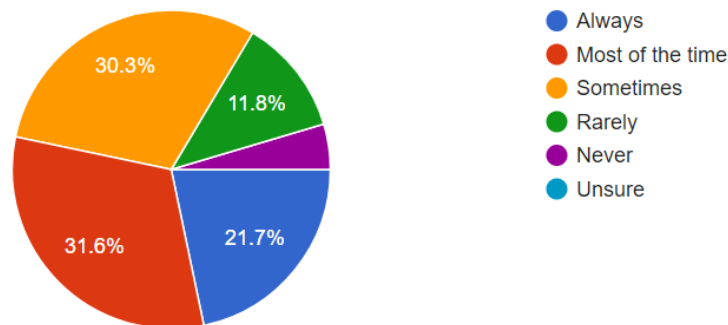
Do you tend to work for too long on a particular area/subject and neglect other areas of study? (Not balancing the work load)

152 responses



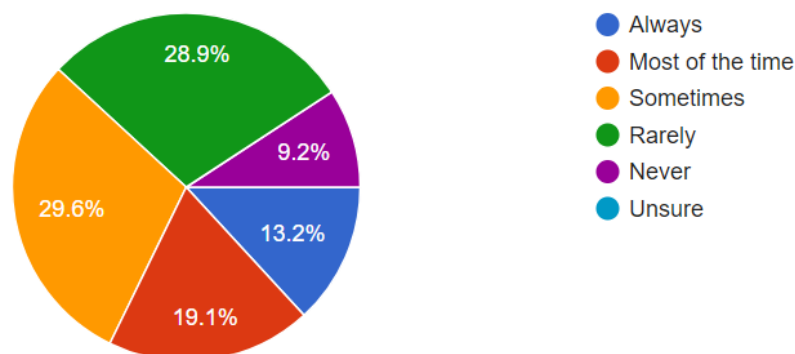
Do you work well under time pressure? (eg. being more focused when having to finish a task within a time limit or when having a upcoming deadline)

152 responses



Do you have to be reminded of the due dates of tasks?

152 responses



This processed data was used in Background: Data Analysis.

Other data was also used to assist the development of the software solution. For the raw data collected, refer to the raw data zip files submitted on Compass.

Appendix B: Context diagram

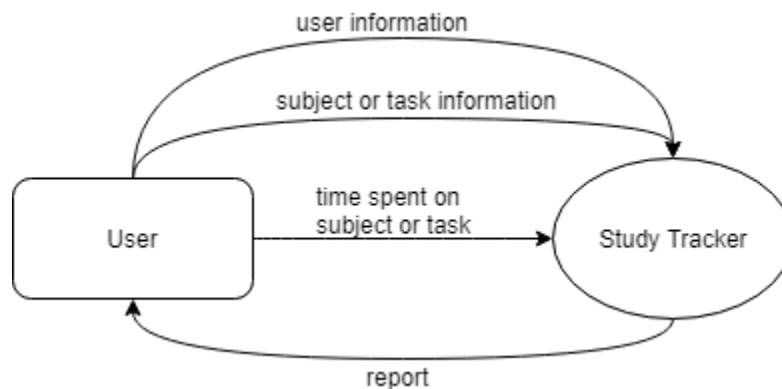


Figure 1. Used in Section 2.1 - Product perspective

Appendix C: Data flow diagram

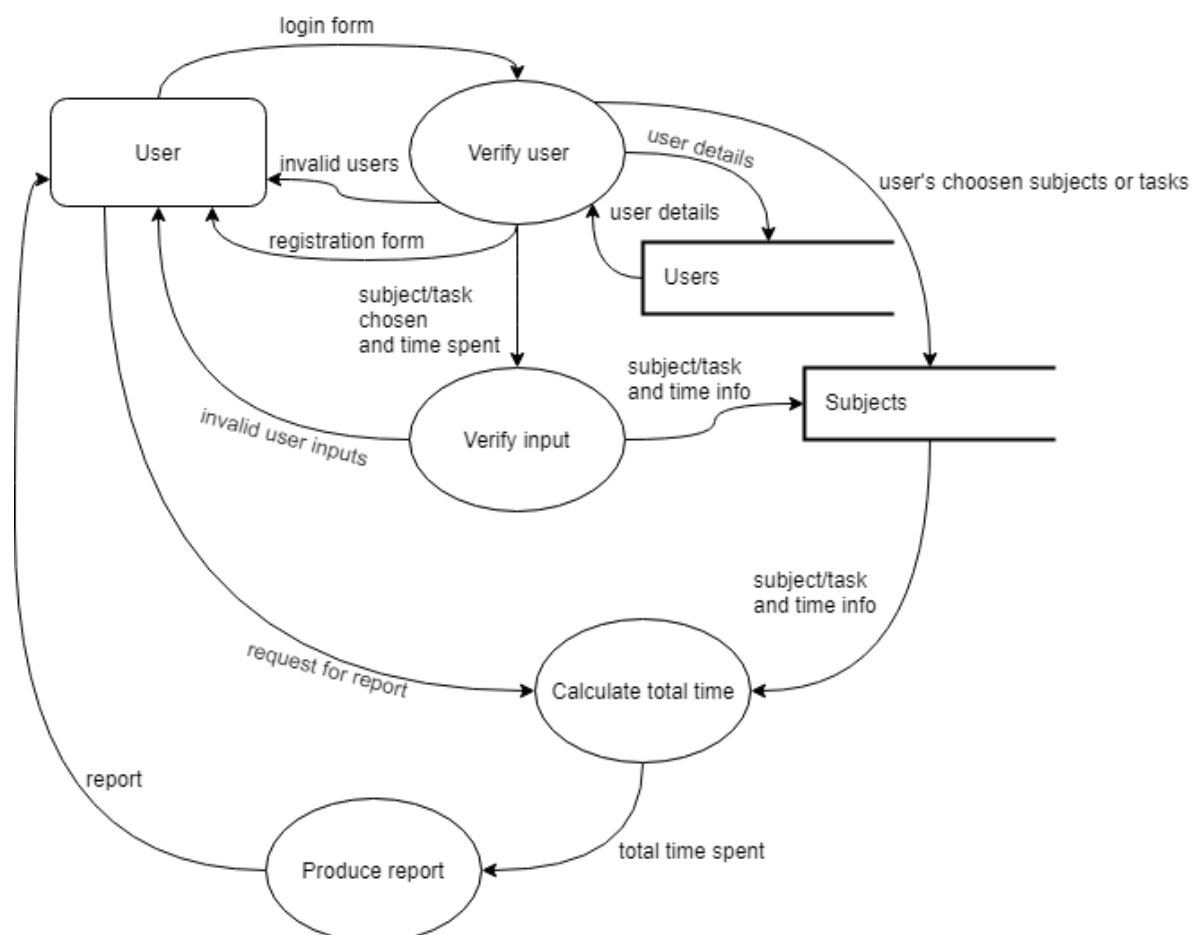


Figure 2. Used in Section 5 - Requirements

Appendix D: Use case diagram

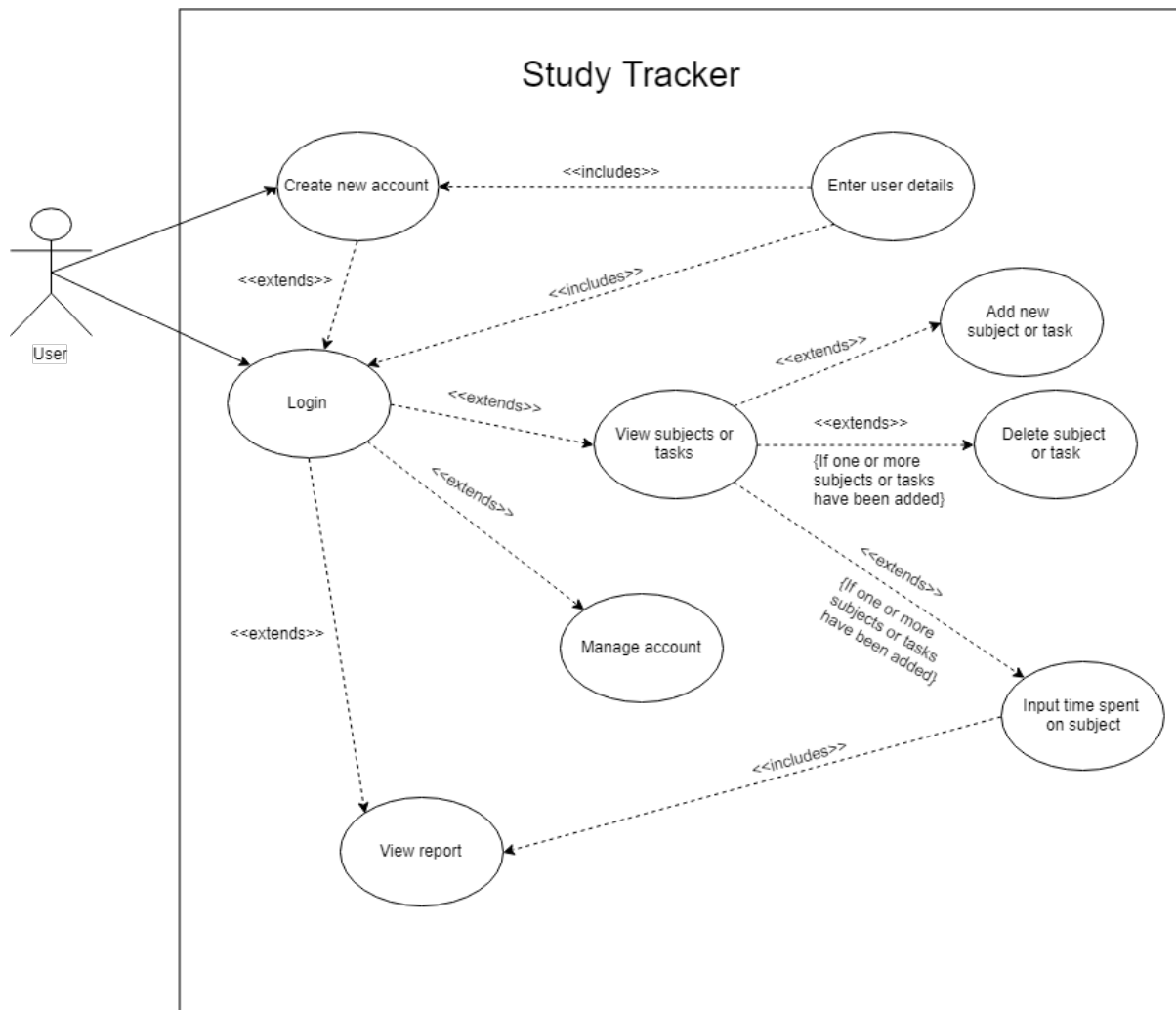


Figure 3. Used in Section 5 - Requirements

Appendix E: Pseudocode

User register – REQ1

```
BEGIN
    INPUT INusername
    INPUT INpassword

    Validate user inputs
    Add new record into user database with username ← IPusername, password ← IPpassword, name ←
IPusername, timerType ← 1, timerNotif ← 1, timerLen ← 30
    Session started
    User is redirected to main page
END
```

User login – REQ2

```
BEGIN
    GET username
    GET password
    INPUT INusername
    INPUT INpassword

    IF username = INusername AND password = INpassword
        Session started
        User is redirected to main page
    ELSE
        Display "Username or password is incorrect. Please try again."
    ENDIF
END
```

User logout – REQ3

```
BEGIN
    Session ended
    User is redirected to the login page
END
```

Timer and break notifications – REQ5

```
BEGIN
    GET timerType
    GET timerNotif

    IF timerType = 1
        IF countdown timer reaches 0
            IF timerNotif = 1
                Play sound
            ELSE
                Display alert "Timer's up!"
            ENDIF
        ENDIF
    ENDIF
END
```

Adding subjects/tasks – REQ6,**Settings – Manage subjects/tasks - REQ11**

```
BEGIN
    INPUT subject
    Validate user input
    Add subject into database
```

```

i ← 1
GET ID
GET number of subjects/tasks in database where userID = ID

REPEAT
    GET subject where SubID = i
    IF subject has userID = ID
        Display subject
    ENDIF
    i ← i + 1
UNTIL i > number of subjects/tasks in database where userID = ID
END

```

//displays current subjects in database

**Deleting subjects/tasks – REQ7,
Settings – Manage subjects/tasks - REQ11**

```

BEGIN
    GET subject selected
    Display alert "Are you sure you want to delete 'subject selected'?"
    IF user confirms
        Delete subject selected from database
        i ← 1
        GET ID
        GET number of subjects/tasks in database where userID = ID

        REPEAT
            GET subject where SubID = i
            IF subject has userID = ID
                Display subject
            ENDIF
            i ← i + 1
        UNTIL i > number of subjects/tasks in database where userID = ID
    ENDIF
END

```

//displays current subjects in database

**Functional timer – REQ4,
Recording time spent on subject/task - REQ8**

```

BEGIN
    GET subject selected
    GET button pressed
    GET timerLen
    GET timerType
    started ← 0
    timerOn ← 0
    timeSpent ← 0

    IF started = 0
        IF timerType = 1
            Set timer to timerLen
        ELSE
            Set timer to 0
        ENDIF
        started ← 1
    ENDIF

    IF start button pressed AND timerOn = 0
        startTime ← time now
        timerOn ← 1
        IF timerType = 1
            Continually decrease timer
        ENDIF
    ENDIF
END

```

```

        ELSE
            Continually increment timer
        ENDIF
    ENDIF

    IF stop button is pressed AND timerOn = 1
        Stop timer
        timerOn  $\leftarrow$  0
        endTime  $\leftarrow$  time now
        timeDiff  $\leftarrow$  endTime – startTime
        timeSpent  $\leftarrow$  timeSpent + timeDiff
    ENDIF

    IF reset button is pressed
        IF timerOn = 1
            Stop decreasing/incrementing timer
            timerOn  $\leftarrow$  0
            endTime  $\leftarrow$  time now
            timeDiff  $\leftarrow$  endTime – startTime
            timeSpent  $\leftarrow$  timeSpent + timeDiff
        ENDIF
        Started  $\leftarrow$  0
    ENDIF

    IF done button pressed
        GET time
        endTime  $\leftarrow$  time now
        timeDiff  $\leftarrow$  endTime – startTime
        timeSpent  $\leftarrow$  timeSpent + timeDiff
        todayDate  $\leftarrow$  current date

        IF timeSpent IS NOT EQUAL TO 0
            Redirect user to timer done page
            Display "Did you work on 'subject selected' for 'timeSpent'?"
            IF "yes" button pressed
                time  $\leftarrow$  time + timeSpent
            ELSE
                Display "Please input the time spent on the subject/task"
                INPUT timeLen
                Validate user input
                time  $\leftarrow$  time + timeLen
            ENDIF
            Update time in database as time for selected subject
            Update today in database as todayDate for selected subject
        ELSE
            Display alert "You haven't used the timer! Would you like to manually input the time spent instead?"
            IF user confirms
                Redirect user to timer done page
                INPUT timeLen
                Validate user input
                time  $\leftarrow$  time + timeLen
                Update time in database as time for selected subject
                Update date in database as todayDate for selected subject
            ENDIF
            Redirect user to main page
        ENDIF
    ENDIF
END

```

Report – REQ9

```

BEGIN
    i ← 1
    todayDate ← current date
    totalTime ← 0
    GET ID
    GET number of subjects/tasks in database where userID = ID

    REPEAT
        GET subject where SubID = i
        IF subject has userID = ID
            GET time for subject where date = todayDate
            totalTime ← totalTime + time
        ENDIF
        i ← i + 1
    UNTIL i > number of subjects/tasks in database where userID = ID
    IF totalTime > 0
        Display "Today you worked for: 'totalTime' minutes in total"
        REPEAT
            GET subject where SubID = i
            IF subject has userID = ID
                GET time from subject where date = todayDate
                IF time > 0
                    Display "You have spent 'time' minutes on subject 'subject'"
                ENDIF
            ENDIF
            i ← i + 1
        UNTIL i > number of subjects/tasks in database userID = userID
    ELSE
        Display "No records available to show. Record the time on a subject/task today then check back here!"
    ENDIF
END

```

Tutorial – REQ10

```

BEGIN
    Display tutorial page
END

```

Settings - Change timer preferences – REQ11

```

BEGIN
    GET timerType
    GET timerLen
    INPUT INTimerType
    INPUT INTimerLen
    INPUT INTimerNotif

    Validate user inputs
    IF timerType IS NOT EQUAL TO INTimerType
        Update timerType in database as INTimertype
    ENDIF
    IF timerLen IS NOT EQUAL TO INTimerLen
        Update timerLen in database as INTimerLen
    ENDIF
    IF timerNotif IS NOT EQUAL TO INTimerNotif
        Update timerNotif in database as INTimerNotif
    ENDIF
END

```

Settings – Manage account – REQ11

```

BEGIN

```

```
    GET name
    GET username
    GET password
    INPUT INname
    INPUT INusername
    INPUT INpassword
    INPUT INsecondPass

    Validate user inputs
    IF INname is inputted AND IS NOT EQUAL TO name
        Update name in database as INname
    ENDIF
    IF INusername is inputted AND IS NOT EQUAL TO username
        Update username in database as INusername
    ENDIF
    IF INpassword is inputted AND IS NOT EQUAL TO password
        IF INpassword = INsecondPass
            Update password in database as INpassword
        ELSE
            Display "Passwords do not match!"
        ENDIF
    ENDIF
END
```

Referred from Section 5 - Requirements

Appendix F: Flow chart

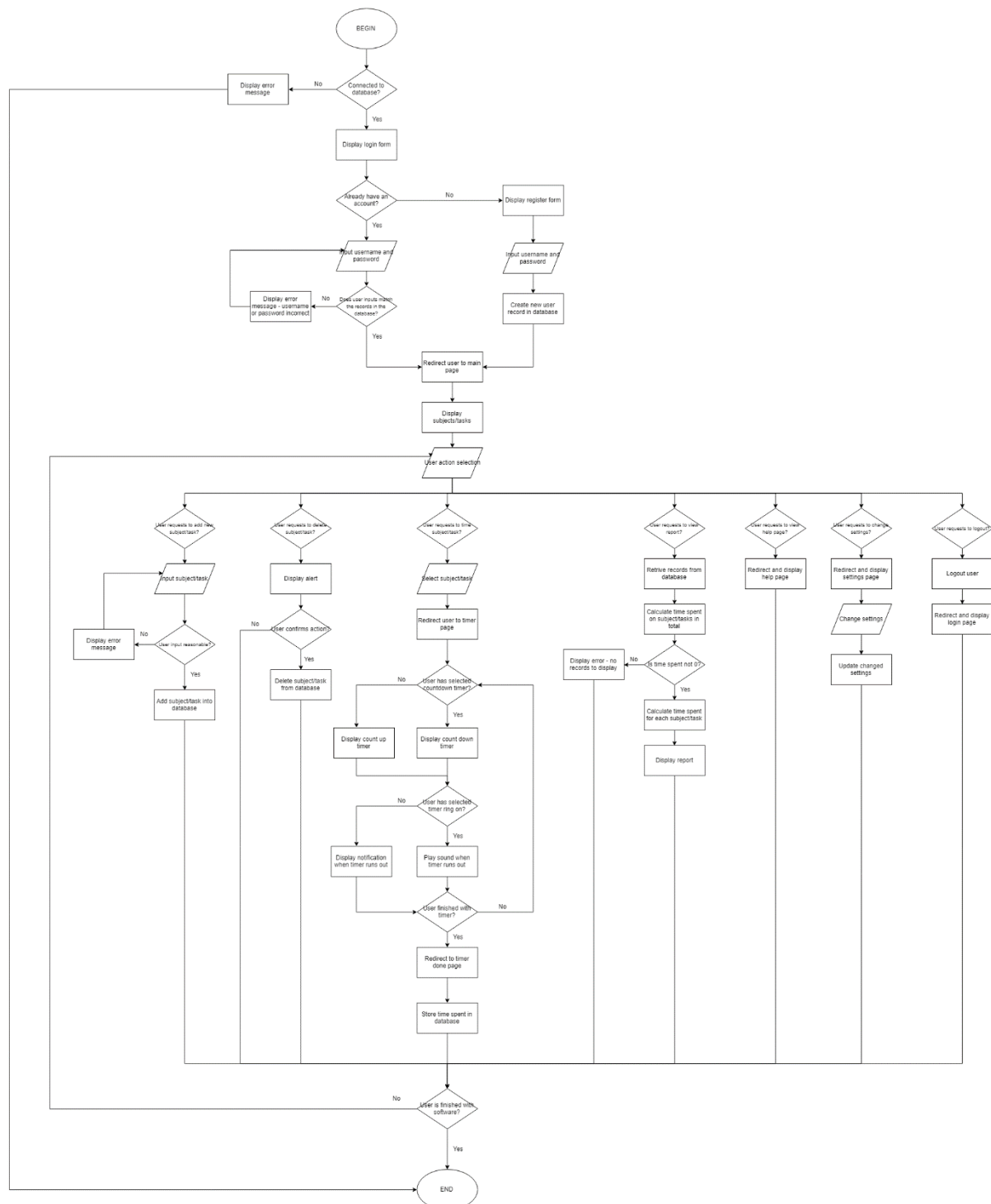
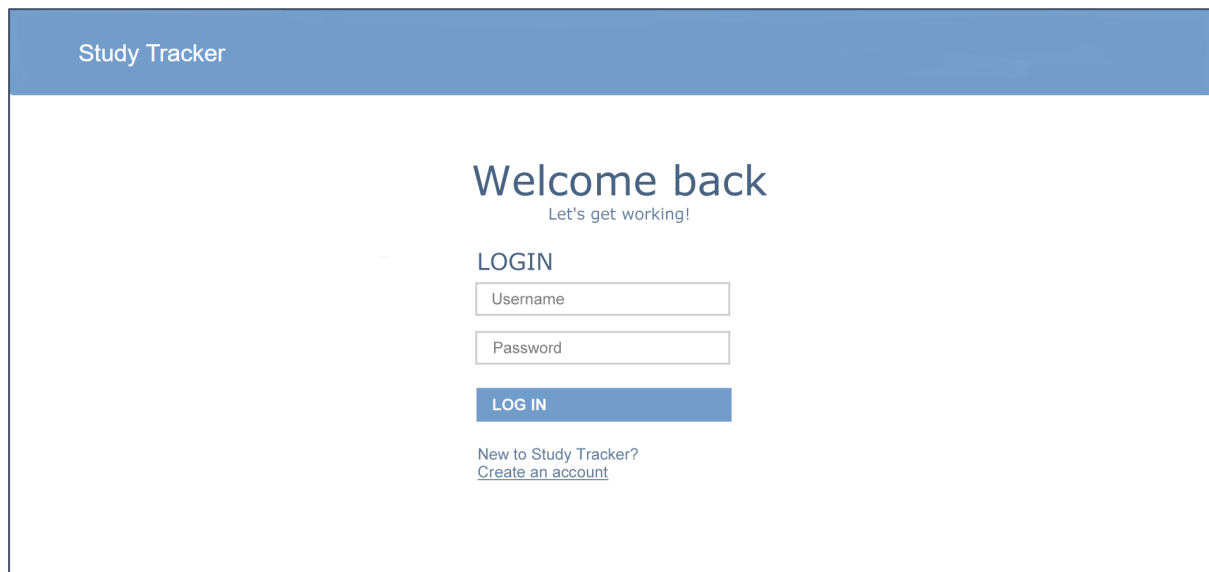


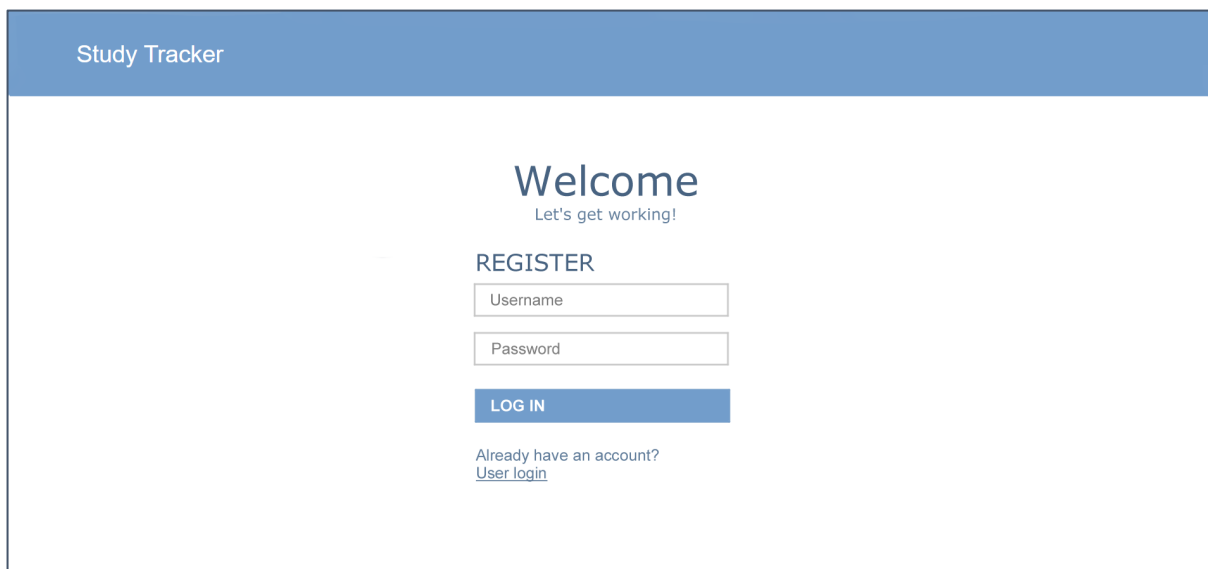
Figure 4 – flowchart. Used in Section 5 - Requirements

Appendix G: User interface mockups



The mockup shows a web page for 'Study Tracker'. It has a blue header bar with the text 'Study Tracker'. The main content area is white and contains the following elements: a heading 'Welcome back' with the subtext 'Let's get working!'; a section header 'LOGIN'; two input fields labeled 'Username' and 'Password'; a blue button labeled 'LOG IN'; and a link 'New to Study Tracker? Create an account'.

Figure 5 – login page mockup. Used in Section 6.1 – User interfaces



The mockup shows a web page for 'Study Tracker'. It has a blue header bar with the text 'Study Tracker'. The main content area is white and contains the following elements: a heading 'Welcome' with the subtext 'Let's get working!'; a section header 'REGISTER'; two input fields labeled 'Username' and 'Password'; a blue button labeled 'LOG IN'; and a link 'Already have an account? User login'.

Figure 6 - register page mockup. Used in Section 6.1 – User interfaces

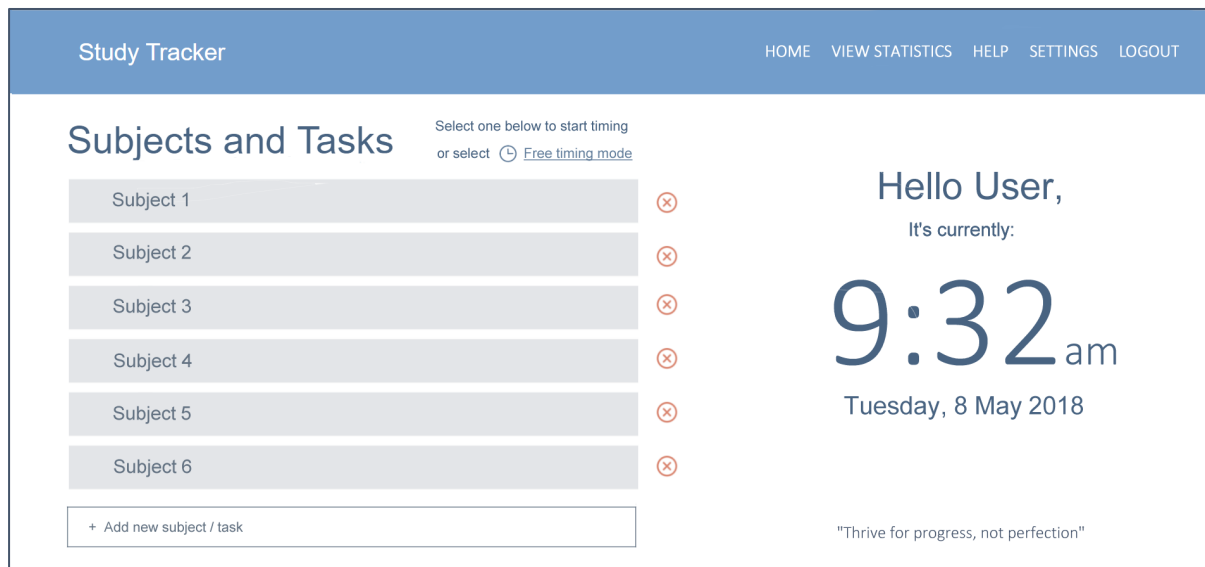


Figure 7 - main page mockup. Used in Section 6.1 – User interfaces

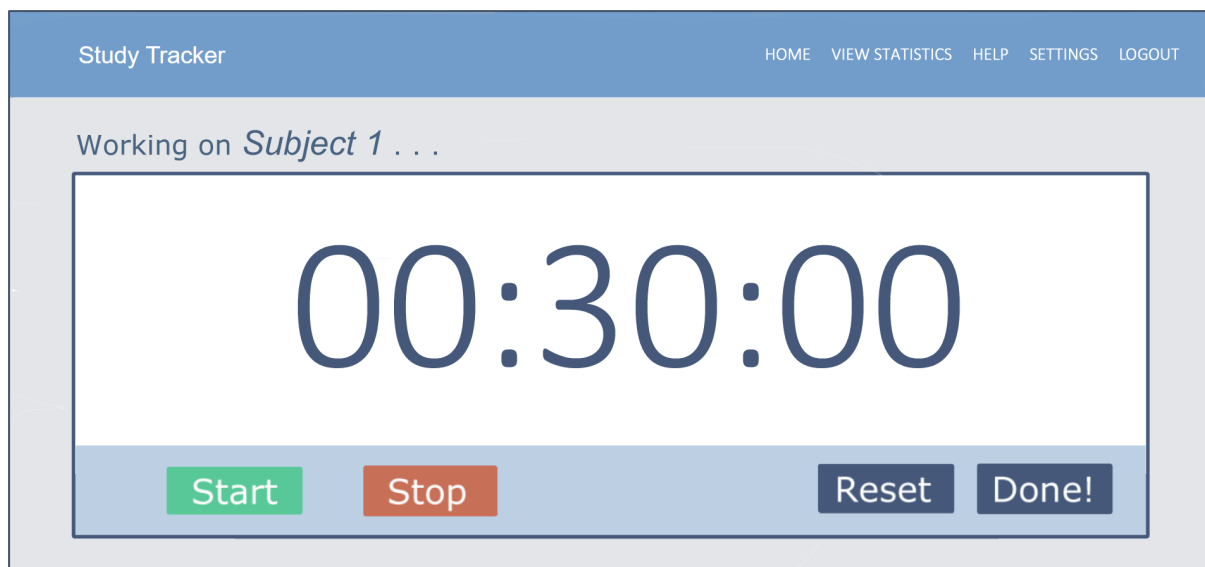


Figure 8 - timer page mockup. Used in Section 6.1 – User interfaces

The mockup shows a web interface for a 'Study Tracker'. At the top is a blue header bar with the title 'Study Tracker' on the left and navigation links 'HOME', 'VIEW STATISTICS', 'HELP', 'SETTINGS', and 'LOGOUT' on the right. Below the header, the main content area has a light gray background. It starts with the text 'Working on *Subject 1* . . .'. Below this is a white rectangular box with a thin blue border. Inside this box, the question 'Did you work on *Subject 1* for 30 minutes?' is followed by a green button labeled 'Yes'. Below the question, the text 'If not, please input the time spent:' is followed by an empty text input field. At the bottom of the white box, there is a light blue horizontal bar containing two dark blue buttons: 'Go back to timer' and 'Done!'.

Figure 9 – timer done page mockup. Used in Section 6.1 – User interfaces

The mockup shows a web interface for a 'Study Tracker'. At the top is a blue header bar with the title 'Study Tracker' on the left and navigation links 'HOME', 'VIEW STATISTICS', 'HELP', 'SETTINGS', and 'LOGOUT' on the right. Below the header, the main content area has a light gray background. It starts with the title 'View statistics'. Below the title, the text 'Today you worked for: 2 hours and 30 minutes (150 min) in total' is displayed. Below this, the text 'You spent:' is followed by a list of items: '30 minutes on Subject 1', '60 minutes on Subject 2', '15 minutes on Subject 3', and '45 minutes on Subject 4'. At the bottom right of the main content area, there is a blue button labeled 'Go back'.

Figure 10 – report (view statistics) page mockup. Used in Section 6.1 – User interfaces

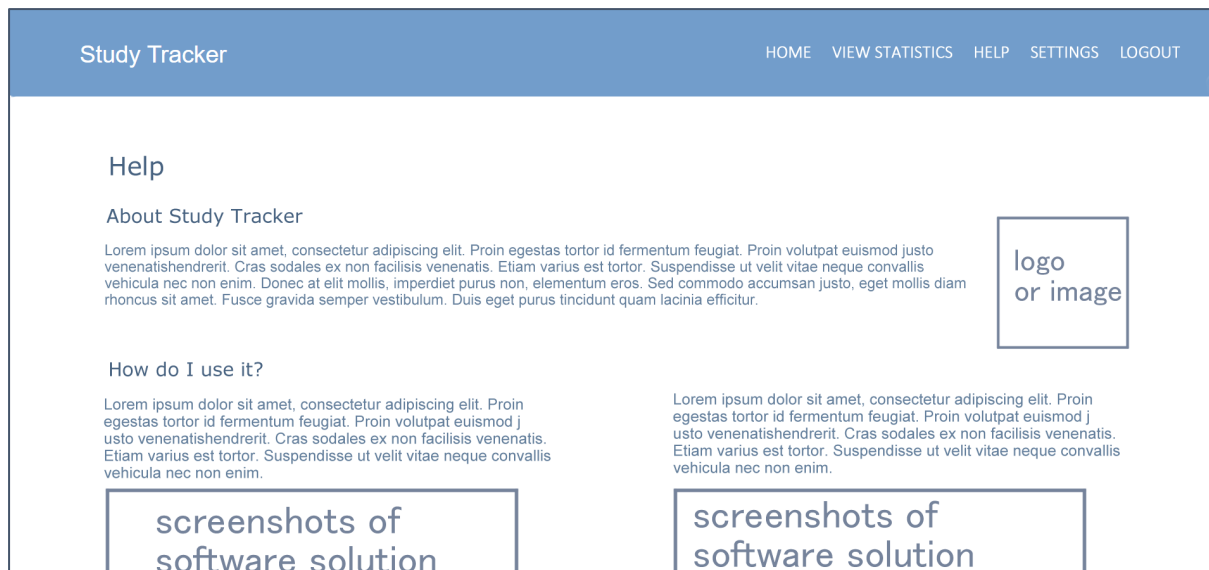


Figure 11 – help page mockup. Used in Section 6.1 – User interfaces

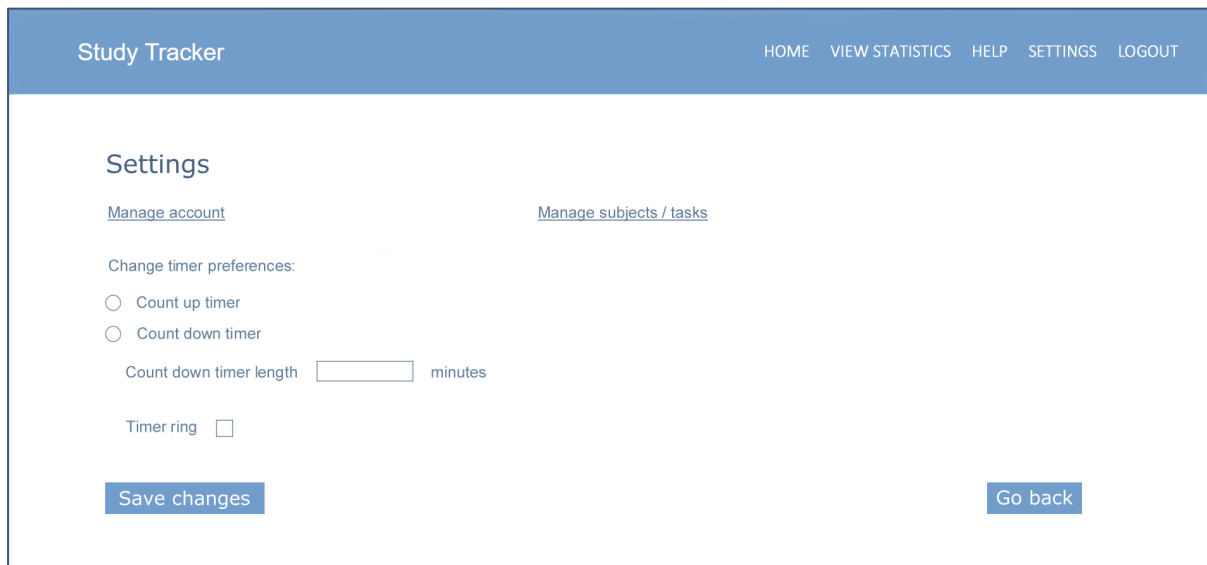


Figure 12 – settings page mockup. Used in Section 6.1 – User interfaces

The mockup shows a web interface for 'Study Tracker'. The top navigation bar includes links for HOME, VIEW STATISTICS, HELP, SETTINGS, and LOGOUT. The main heading is 'Settings - manage account'. Below this, there are three input fields: 'Username:', 'Password:', and 'Confirm new password:'. To the right, there is a 'Display name:' field with a subtext 'This the the name that appears on your home page.' Below the input fields, there is a red link 'Delete my account'. At the bottom, there are two buttons: 'Save changes' and 'Go back'.

Figure 13 – settings – manage account page mockup. Used in Section 6.1 – User interfaces

The mockup shows a web interface for 'Study Tracker'. The top navigation bar includes links for HOME, VIEW STATISTICS, HELP, SETTINGS, and LOGOUT. The main heading is 'Settings - manage subjects/tasks'. Below this, there is a text instruction: 'Enter in the textbox to change the subject/task's name and use the delete button to delete the subject/task.' Below the instruction, there are six input fields labeled 'Subject 1' through 'Subject 6'. Each input field has a red 'X' button to its right. To the right of the input fields, there is a button labeled '+ Add new subject/task'. At the bottom, there are two buttons: 'Save changes' and 'Go back'.

Figure 14 – settings – manage subjects/tasks page mockup. Used in Section 6.1 – User interfaces

Appendix H: Sitemap

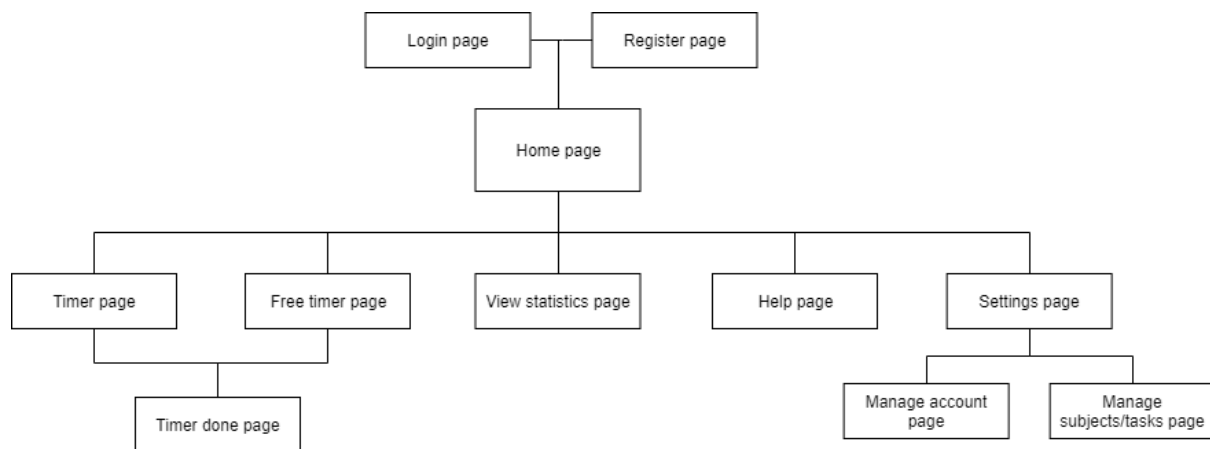


Figure 15. Used in Section 6.1 - User interfaces

Appendix I: Data dictionary and Object descriptions

Data dictionary for users database

Name	Data type	Length	Default value	Key	Required?	Validation	Description	Sample
ID	Integer	8		Primary key	True		User's ID	10
username	VARCHAR	15			True		Username	Alexiscool
password	VARCHAR	15			True	≥ 5 characters long	Password	Abcde
name	VARCHAR	15			False		User's given name	Alex
timerType	Boolean	1	1		True	0 or 1	Use a countdown timer? 1 = true, 0 = false (if false, a count up timer is used instead)	1
timerNotif	Boolean	1	1		True	0 or 1	Timer sound on? 1 = true, 0 = false	1
timerLen	Integer	3	30		False	>0	Length of countdown timer in minutes	30

Data dictionary for subject/tasks database

Name	Data type	Length	Default value	Key	Required?	Validation	Description	Sample
subID	Integer	8		Primary key	True		Subject/task's ID	1
subject	VARCHAR	25			True		Subject/task's name	English
userID	Integer	8		Foreign key (ID from users table)	True		ID of user with the subject/task	10
date	date	10	Current date		True	≤ current date	Date when subject is created or when time is recorded	2018-05-15
time	VARCHAR	5	0		True	≥ 0	Time spent on subject/task in minutes	45

Object description

Name	Caption/display	Location	Visible?	Colour	Background colour	Size	Font	Description
Username text input	Username	Login page, register page	True	Grey		Medium sized	Arial	Allows users to input their username
Password text input	Password	Login page, register page	True	Grey		Medium sized	Arial	Allows users to input their password
Log in button	LOG IN	Login page, register page	True	White	Blue	Medium sized	Arial	Checks if username and password matches the database records, creates a new session, redirects users to main page If the button is on the register page, it first makes a new record in the database.
Register link	Create an account	Login page	True	Navy blue		Medium sized	Arial	Links to login page
Login link	User login	Register page	True	Navy blue		Medium sized	Arial	Redirects to register page
Free timing mode link	Free timing mode	Main page	True	Navy blue		Medium sized	Arial	Redirects to timer page but doesn't record the length of time for a specific subject/task
Subject/task buttons		Main page	True	Blue-grey	Light grey	Medium sized	Arial	Selects a subject/task and links to timer page
Add new subject/task text input	+ Add new subject/task	Main page	True	Grey		Small sized	Arial	Allows users to input new subjects/tasks
Add new subject/task button		Main page, settings – manage subjects/tasks page	False					Adds a new subject/task to database
Delete subject/task image button	(image of red cross)	Main page, settings – manage subjects/tasks page	True			Medium sized		Deletes a subject/task from database
Clock	(current time)	Main page	True	Dark navy blue		Extra large sized	Calibri Light	A functional (animated) clock
Current date	(current date today)	Main page	True	Dark navy blue		Medium-large sized	Arial	Displays the current date
Logo image link	(image of logo)	All pages except for	True	White		Medium-large sized		Redirects to main page

		login/register page						
Home (main) page link	Home	All pages except for login/register page	True	White		Medium sized	Calibri Light	Redirects to main page
Report (view statistics) page link	View statistics	All pages except for login/register page	True	White		Medium sized	Calibri Light	Redirects to report (view statistics) page
Help page link	Help	All pages except for login/register page	True	White		Medium sized	Calibri Light	Redirects to help page
Settings page link	Settings	All pages except for login/register page	True	White		Medium sized	Calibri Light	Redirects to settings page
Logout link	Logout	All pages except for login/register page	True	White		Medium sized	Calibri Light	Destroys session, redirects user to login page
Timer	(timer)	Timer pages	True	Dark navy blue		Extra-extra-large sized	Calibri Light	A functional count up/count down timer
Timer start button	Start	Timer pages	True	White	Green	Medium-large sized	Arial	Starts or resumes the timer
Timer stop button	Stop	Timer pages	True	White	Red	Medium-large sized	Arial	Pauses the timer
Reset timer button	Reset	Timer pages	True	White	Dark navy blue	Medium-large sized	Arial	Pauses and resets the timer back to its default value
Timer done button	Done!	Timer pages, timer done page	True	White	Dark navy blue	Medium-large sized	Arial	Redirects to timer done page
Timer confirm button	Yes	Timer done page	True	White	Green	Medium-large sized	Arial	Saves the time spent on subject/task to database
Length spent number input		Timer done page	True	White	Dark navy blue	Medium sized	Arial	Allows user to manually input the time spent on subject/tasks
Length spent button		Timer done page	False					Saves user inputted length of time in database
Go back to timer button	Go back to timer	Timer done page	True	White	Dark navy blue	Medium-large sized	Arial	Redirects back to the timer page
Go back button	Go back	Report (view statistics) page, help page, all settings pages	True	White	Blue	Medium sized	Arial	Redirects to main page (or to the settings page if the button is located on the

								settings – manage account page or on the settings – manage subjects/tasks)
Timer type radio button		Settings	True	Grey		Medium sized		Allows user to select between the timer types
Timer length number input		Settings	True	Grey		Medium sized	Arial	Allows user to select the countdown timer length
Timer notification checkbox		Settings	True	Grey		Medium sized		Allows user to select whether they want a sound to be played when the timer runs out.
Manage account link	Manage account	Settings page	True	Navy blue		Medium sized	Arial	Redirects to settings – manage account page
Username text input		Settings – manage account	True	Grey		Medium sized	Arial	Allows users to input their username
Password text input		Settings – manage account	True	Grey		Medium sized	Arial	Allows users to input their password
Confirm password text input		Settings – manage account	True	Grey		Medium sized	Arial	Allows users to reinput their password for confirmation
Display name text input		Settings – manage account	True	Grey		Medium sized	Arial	Allows users to input their display name
Manage subjects/tasks link	Manage subjects/tasks	Settings page	True	Navy blue		Medium sized	Arial	Redirects to settings – manage subjects/tasks page
Subject/task text input		Settings – manage subjects/tasks page	True	Blue		Medium sized	Arial	Allows user to change the name of the subject/task
Add new subject/task text input	+ Add new subject/task	Settings – manage subjects/tasks page	True	Blue		Small sized	Arial	Allows users to input new subjects/tasks
Save changes button	Save changes	All setting pages	True	White	Blue	Medium sized	Arial	Saves changes to database

Appendix J: Glossary

SRS - Software Requirements Specification.

User - An individual who interacts with the software solution.

Stakeholder - Any individual involved with the software solution such as users and developers.

Subjects - A branch of knowledge taught in school such as Mathematics or English. Note that in this current solution, the terms 'subjects' and 'tasks' are of equal importance.

Tasks - Activities or pieces of work to do. Note that in this current solution, the terms 'subjects' and 'tasks' are of equal importance.

HTML – Hypertext Markup Language. The standard markup language used to create web pages.

PHP – Hypertext Preprocessor. A server-side scripting language.

MySQL – Structured Query Language. A database management system.

JavaScript – A high-level interpreted programming language.

REQ – requirement. What the software solution should do.

NREQ – Non-functional requirement. How the software solution should perform.

Precondition – A condition that must be met for the requirement to be run.

Postcondition – A condition that is met after the user performs the requirement.

Acknowledgements:

W3Schools – The world's largest web developer site, W3Schools

Retrieved 20 April 2018 from

<https://www.w3schools.com/>

Herbert Wertheim College of Engineering – Computer & Information Science & Engineering,
SRS Example – Spilt Pay 2011, University of Florida

Retrieved 22 April 2018 from

https://www.cise.ufl.edu/class/cen3031sp13/SRS_Example_1_2011.pdf

Wiegers, K, 1999, *Software Requirements Specification Template*

Retrieved 27 March 2018 from

https://view.officeapps.live.com/op/view.aspx?src=http%3A%2F%2Fwww.uni-obuda.hu%2Fusers%2Fboraros-bakucz.andras%2F2013%2Fsrs_template.doc

iTest – Computerized examination made easy, *Software Requirements Specification iTest*,
iTest

Retrieved 27 March 2018 from

http://itest.sourceforge.net/documentation/developer/Software_Requirements_Specification-on-iTest.pdf

Francesco Cirillo, *The Pomodoro Technique*, Cirillo Consulting GMBH

Retrieved 26 April 2018 from

<https://francescocirillo.com/pages/pomodoro-technique>

MySQL, *Client/Server Protocol*, MySQL 8.0.11 Source Code Documentation

Retrieved 7 May 2018 from

https://dev.mysql.com/doc/dev/mysql-server/latest/PAGE_PROTOCOL.html