Promodoro Mobile Application

Abanoub George, Ibrahim Fawzy, Mai Mahmoud , Nour Bahaa.
Supervised by: Dr. Essam Eliwa.

October 18, 2020

GitHub: `https://github.com/BonyGeorge/Pomodoro-Mobile-App`

Contents

1 Document Version

SRS Version	Date	Reason for Change
1.0	18-October-2020	SRS First version

Table 1: Document version history

2 Introduction

2.1 Purpose of this document

The point of this paper is to show full data about the task framework. The archive will depict the trademark focuses and portray the use of the Pomodoro. This archive will help the clients of the application how to utilize it and full data about the undertaking capacities and interfaces.

2.2 Scope of this document

This mobile application for Pomodoro technique. The Pomodoro Technique is a profitability framework that encourages you take the correct number of breaks while as yet completing your work. Generally, it separates your day into 25-minute center meetings followed by five-minute breaks.[3] It's the ideal period of time for absorbing information and completing things—without wearing out.

A committed Pomodoro application eliminates the need to split your day physically—rather, it lets you know precisely when to work and when to take a brief break. Here's a short gander at how the procedure functions, alongside our picks for the 10 best Pomodoro clock applications. Therefore, it will be designed to help the users the best way to try to manage their time in the best way they can. We will use flutter to build this application.This application should be ready within the end of this semester.

2.3 Overview

This application aims to help and manage the students or any user to try to make their best in the 25 minutes of studying without any disturb to their focus. Then, it will give them the appropriate break so they can rest. So by that they can use their time in the most usable way. [2]

General rules:

- In the event that an assignment takes more than 5–7 Pomodoros, separate it

- In the event that it takes short of what one Pomodoro, include it up, and consolidate it with another errand

- When a Pomodoro starts, it needs to ring

- The following Pomodoro will go better

- Login to the administration and keep tabs on your development

- The Pomodoro Technique shouldn't be utilized for exercises you do in your extra time. Appreciate leisure time!

3 General Description

3.1 Product Functions

The Pomodoro Technique will give a straightforward apparatus/measure for improving profitability (your own and that of your colleagues) which can do the accompanying[4]:

- Ease tension connected to starting

- Upgrade center and fixation by eliminating interference's

- Increment consciousness of your choices

- Lift inspiration and keep it consistent

- Support the assurance to accomplish your objectives

- Refine the assessment cycle, both in subjective and quantitative terms

- Improve your work or study measure

- Fortify your determination to continue putting forth a concentrated effort even with complex circumstances

3.2 Similar System Information

This application is a stand alone application. The user need internet only for the first time when he/she install it then they can use it offline. Toward the start of every day select the undertakings you have to finish and put them on the TODO list above[1].
Begin working:

1. Start the Pomodoro clock

2. Work until the Pomodoro rings

3. Enjoy a short reprieve (3-5 minutes)

Continue working, Pomodoro after Pomodoro, until the job that needs to be done is done. Each 4 Pomodoros enjoy a more drawn out reprieve, (15–30 minutes).

3.3 User Characteristics

Anyone can download the application. As it's made with Flutter, they can find it in both App Store and Google play. The user need only to have task to do so that the application can manage it.Therefore , it will give the user track of the work that they completed and the work they need to finalize.

3.4   User Problem Statement

Everyone finds a lot of distractions in their daily routine. And do have a lot of deadlines for either assignments or work things that should be delivered in time, so that's when Pomodoro works come it will manage their time and it support for less distraction . Some people forget about it so that comes the work of the Pomodoro, it benefit them to remind their deadlines. Some people feel pessimistic when they plenty of things to do by their day and they end up doing nothing, Pomodoro is the solution because it will give hope when they find themselves finished what they needed of finish in a very organize time in the end of the day. Therefore, they don't feel overworked in the day , as it manage both their work and their break.

4   Functional Requirements

Non-Critical	Non
Less Critical	Less
Critical	Normal
Very Critical	Very
Extremely Critical	Extreme

1. User is able to register and has a choice not to register:

   - Description - Th system will obtain the user's email , username and password. Once user is logged in he will be able to see all his tasks and teams if there are any he belongs to , otherwise there will be an option to create a new team.
   - Criticality - Normal
   - Dependencies - None

2. User is able to create a team and will be the admin:

   - Description - User is able to create a team and give it a name and can add people using username.
   - Criticality - Normal
   - Dependencies - Register

3. User is able to create a task:

   - Description - User is able to create a task and give it a name
   - Criticality - Very
   - Dependencies - None

4. User is able to leave a team:

   - Description - User is able to leave a team so he doesn't see the tasks in the team.
   - Criticality - Normal
   - Dependencies - Register/Log in

5. User is able to start a pomodoro:

   - Description - User can start his pomodoro (25 minutes long) and if he is in a team the other team members can see the count down on their team dashboard and if not he will be the only one to see the count down of his pomodoro.

- Criticality - Extreme
- Dependencies - Create a task

6. User is able to pause pomodoro:

    - Description - When user is in pomodoro , he can pause , whenever it is needed. If he is in a team a pause will be visible to all the team members.
    - Criticality - Extreme
    - Dependencies - start pomodoro

7. User is able to reset pomodoro:

    - Description - User can reset his pomodoro to 25 minutes and if he is in a team all team members can see it.
    - Criticality - Extreme
    - Dependencies - pause pomodoro

8. User is able to make a short break after each pomodoro:

    - Description - User after finishing his pomodoro (25 minutes), the system will show a short 5 minutes break. Other members will see the break count down.
    - Criticality - Very
    - Dependencies - finish pomodoro

9. User is able to make a long break after each four Pomodoros:

    - Description - Instead of a short break the user can take a long 15 minutes break after finishing four Pomodoros. Other members will see the count down.
    - Criticality - Very
    - Dependencies - finish 4 Pomodoros

10. User can define a task in team:

    - Description - User can create a task for his team and give it a name.
    - Criticality - Very
    - Dependencies - create a team

11. User can define how many Pomodoros a task will take:

    - Description - User can control how many Pomodoros he wants to finish his task.
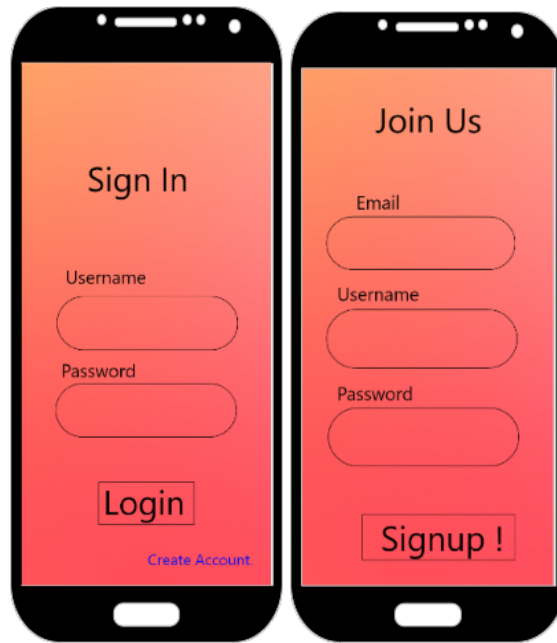   - Criticality - Very
    - Dependencies- None

5 Interface Requirements

5.1 User Interfaces

This application will have a great accommodating UI&UX by that they can both understand and use it in the easiest way.
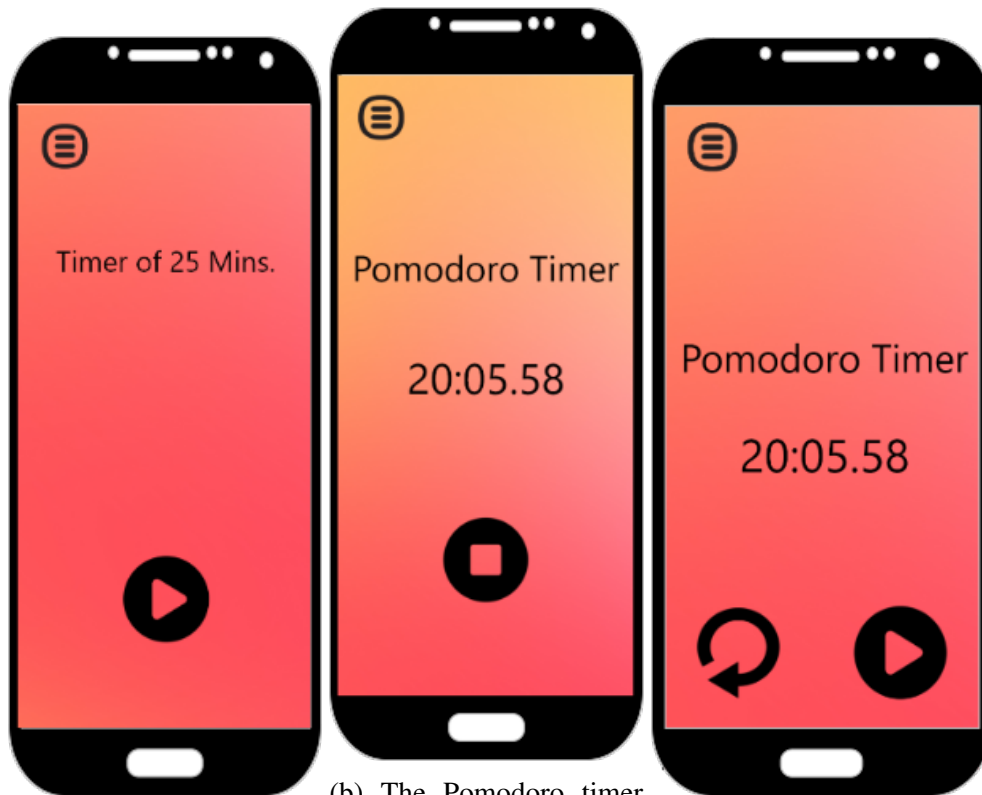

Figure 1: Landing Frame of the application.

(a) User trying to sign in.

(b) User Trying to sign up.

Figure 2: User interface

(a) User starting the timer.

(b) The Pomodoro timer running.

(c) User Stopping the timer.

Figure 3: Timer

<div style="text-align:center">(a) Users' Projects.   (b) Users' Tasks.</div>
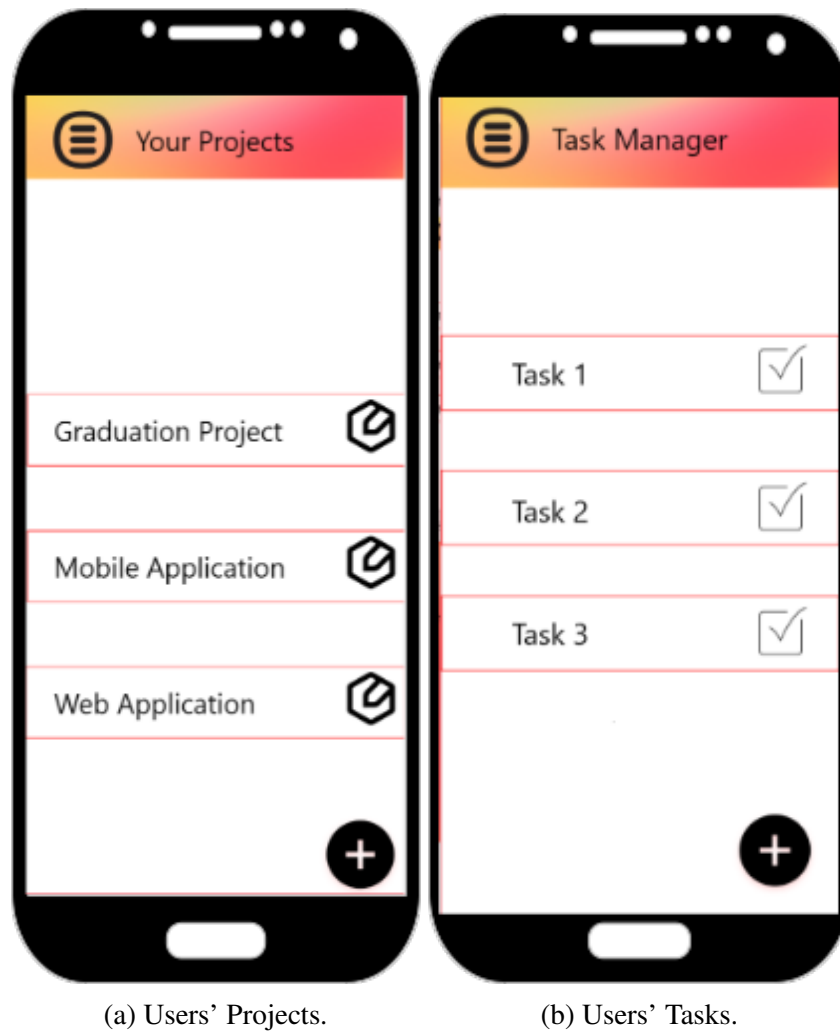
<div style="text-align:center">Figure 4: Task manager</div>

5.1.1 CLI

No Command Line Interface is used.

5.2 API

1. Google Calendar API.

2. Google OAuth API.

3. reCaptcha API.

4. Fire-base.

6 Design Constraints

The mobile application should be accessible by any Android device we operating system (Jellybeans or higher) and on any IOS device.

7 Non functional requirements

7.1 Security

User's data could only be changed and accessed by him, information and each user should not access the database of any other users and each user shall access the system with his username and password.

7.2 Reliability

The database update process must roll back all related updates when any update drops, because it should always be up to date.

7.3 Maintainability

The system is maintained through using a list of design patterns and The system include records in the database including many relations to give the flex-ability to the application.

7.4 Portability

The mobile application should work on cross-platforms devices such as Android, IOS but the user must download it first.

7.5 Scalability

The application shall have high scalability and the attendancy that must be enough to support the large number of users at the same time.

7.6 Usability

This application should be easy to use so anyone can use it and doesn't find any difficulty.

7.7 Performance

The application should be always working properly and on it's highest performance because, we are using a realtime database such as the fire-base because, it's very efficient and it should also be accessible at anytime of the day so that will be affordable for any of the users at any time.

7.8 Availability

The application should be available at anytime and the database shall always be ready to be accessed at anytime of the day and to be secured.
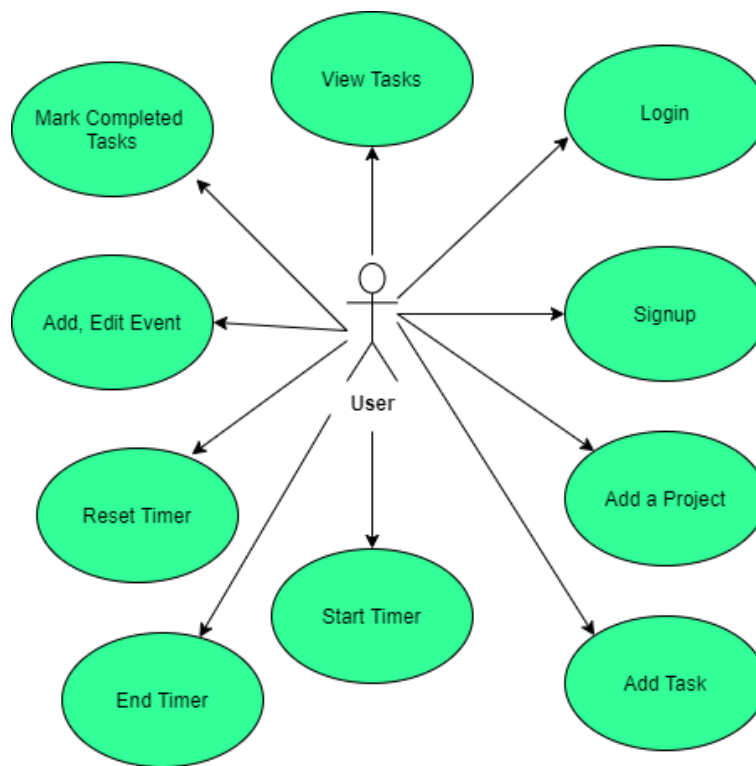
8 Operational Scenarios

Figure 5: Our Scenario case.

1. The user can register by using email to see their task history.

2. User can create a team , add people using their username or can remove them by being the admin.

3. User can make tasks and set a deadline for it for the whole team to do it.

4. User can create a task in the team so all members can see it as well or can make it individually.

5. User can start , pause ,or reset the Pomodoros as he needs.

6. User can define how many Pomodoros in each task.

9 Preliminary Schedule Adjusted.

Project Plan	
Task	Date
SRS Phase 1	18-October-2020
Project Phase 1	8-November-2020
Project Phase 2	Week 6
Project Phase 3	Week 12

References

[1] P. Aminov, N. Bola, D. Shiralkar, and M. Yoganarasimha. Cloud based algorithm for task management. In *2019 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, pages 249–253, 2019.

[2] Francesco Cirillo. The pomodoro technique (the pomodoro). *Agile Processes in Software Engineering and*, 54(2):35, 2006.

[3] Francesco Cirillo. *The pomodoro technique*. Lulu. com, 2009.

[4] Federico Gobbo and Matteo Vaccari. The pomodoro technique for sustainable pace in extreme programming teams. In *International Conference on Agile Processes and Extreme Programming in Software Engineering*, pages 180–184. Springer, 2008.