CEN 4010 Principles of Software Engineering, Spring 2023

Creating an app for the sole purpose of giving time for assignments or other time based tasks based on the due date. A guide for those seeking time management.

Team Procrastination, The Procrastination App. Procrastination.

Group 18

Colby Kerner

[ckerner2020@fau.edu](mailto:ckerner2020@fau.edu)

Matthew Eisenberg

[eisenbergm2019@fau.edu](mailto:eisenbergm2019@fau.edu)

Dawson Stage

[dstage2021@fau.edu](mailto:dstage2021@fau.edu)

Christian Nichols

[cnichols2019@fau.edu](mailto:cnichols2019@fau.edu)

Jason Lockett

[jlockett2020@fau.edu](mailto:jlockett2020@fau.edu)

Documentation date: 3/3/2023

Revision dates:

Executive Summary:

For the average person, a lot of different tasks occur on a day to day basis from different parts of their lives. Finding ways to manage all of these responsibilities is half the battle when it comes to getting everything done on time. Making the perfect scheduler has been attempted by many different companies but there seems to be a lack of user friendliness amongst most of these options along with unrealistic expectations. Our products goal is to provide a very user friendly platform that allows you to schedule out events on a day to day basis while giving proper alerts to make sure you stay on top of your work at the right time. With our team name being “Team Procrastination”, we want to capitalize on this by setting realistic expectations on when you will complete your tasks. The scheduler will allow to pick a category for the type of task you will be working on. This could be things like writing essays, implementing code, and even allotting time to study for an exam. All of these things will very on how much time they will take to complete. You will be able to break down assignments into parts if needed, and give a time estimate for each part to come up with the over all time estimate it will take to complete the assignment. With the scheduler knowing the due date, and the max amount of time it will take the individual to complete the assignment, it will then send out a notification exactly an hour before it would be possible for you to complete the assignment depending on the allotted time you specified. You can add multiple assignments to this scheduler and it will factor in all of these due dates with the max amount of time needed for each to make sure you can wait as long as possible to complete all of your assignments while still getting all of your work done. The target audience for this type of application would be to the average college student with a goal to make them better procrastinators. This scheduler does not have the intention of promoting procrastination but the fact is that a lot of college students do this anyways and need guidance on how to do it better.

Competitive Analysis:

HubSpot:

Sched:

Picktime:

MobiScroll:

Calendly:

The advantages of our scheduler is that it understands that procrastination is a thing that happens to most people. The main difference is that this scheduler capitalizes on this by sending the user an alert exactly when it is their last chance to get their work done. This will allow the user to plan their own destiny by setting times in advance so they know what they are getting themselves into with a plan to finish everything on time.

Data Definition:

The idea will be a calendar based application with the ability to add events to different dates.

* We can have a variable that will deal with over time the user has allotted for assignments that will be adjusted day by day assuming the user does not have over 24 hours of work due on the same day
* We will need a variable for the due date time for any given day. If an assignment is due for example at 12pm, we will assume the day starts at 12am for every day of the week. With this example it would give the user a full 24 hours to work with. If the user puts that their assignments that day will take a total of 5 hours, we will subtract this many hours plus 1 from the due date(in this example 12pm) to give an alert at the perfect time to start working.
* We will need some type of subject function that allows the user to select a type of assignment from a dropdown list. The user will then be able to break this assignment up into smaller sub assignments if needed with each having their own individual amount of allotted time.
* The alert function will be some type of pop up box with some motivational sound that will get the user motivated to grind the assignment out before it’s due.

Overview, Scenario, Use-Cases:

This product would be set up for the user to place dates up to a month in advance. This allows for the user to break their work load down in chunks on a month to month basis, making it more simple to figure out what needs to be done each week. They would be able to click a day of the month, select the type of assignment, break it down into sub assignments if needed, and specify a maximum amount of time needed to complete it. At that point it will be saved on the calendar and allow for the user to go back to dealing with their lives. When a deadline is approaching, the scheduler will send out a notification in order to help motivate the user to finish their work on time. If there are other events that are not work related that may get in the way of the deadline, the scheduler will have an option to add extra time for events to factor this into the time needed to finish on time. The process of adding an event is intended to be as quick and easy as possible.

Initial List of High Functional Requirements:

1. Create an upstanding framework to ensure that the application is not cluttered and is up to par with competing applications alike. Make sure that the application is easy to navigate and allows the user to have an easy, yet beneficial experience with the application.
2. Make sure that the calendar that is linked to the framework is sufficient to enable the user’s to pick the date and time that their task is due.
3. Ensure that the application is synced with the user’s device to guarantee that the time and date that their task is due at will be accurate according to their devices set time and location. This will ensure that the user will be able to complete their tasks accordingly without having to manually adjust due times because of variables such as time zones.
4. Create the notifications that the application will send to the user’s device and ensure that user acknowledges the notification to guarantee that the user knows of the tasks at hand and when they are due
5. Allow the users to have a clear indication of different types of assignments for each day and create a framework to make sure that assignments with higher priority are alerted to them first(green,yellow,red).

List of Non-Functional Requirements:

Performance: This application won’t have a massive performance hit for a user’s device as it is only a simple application with an alert. May have to run in the background to ensure that the user will be able to receive a notification when an assignment is due

Usability: Our team wants to ensure that our product is of use to as many people as possible to ensure that everyone from students to professionals are able to make good use of our application.

Accessibility: In order to ensure that it is able to be accessible to as many people as possible I believe that our team wants to make the application free(with ads) with an option to access a premium paid version that is ad free.

Expected load: Our application will not take that much space or power as it will not feature high quality video(except for advertisements) and will be graphically simple. On the data side, the application will make use of the user’s own data and memory and we will not have to keep the users data on an external server.

Security: For now it will be an offline application that will not house any information that is in need of being secure. In the future we are thinking of implementing user accounts that will allow users to create and link their accounts with their employers to allow employers to set expectations time cutoffs for their employee’s tasks.

Storage Availability: The application will take use of the user’s mobile device and its available storage to store the small amount of data necessary.

Fault Tolerance: As the application will be quite simple and does not require many moving pieces to run sufficiently there is not a fault tolerance in place yet. If needed we will make sure to release updates to ensure the effectiveness of our application and to align it with either IOS or Android updates.

High-Level System Architecture:

Team:

Scrum Master: Matthew Eisenberg

Product Owner: Colby Kerner

Development Team: Dawson Stage

Development Team: Jason Lockett

Development Team: Christian Nichols

Checklist:

* Basic means of communication: DONE
* Time slot to meet: DONE
* Front and back end team leads chosen: DONE
* Github master chosen: DONE
* Team ready for front and back end frameworks: ON TRACK
* Skills of each team member defined and known: ON TRACK
* Team lead ensures each member reads final M1 and agrees/understands before submission: ON TRACK