**Dream Team**

**Smart Calendar**

**Analysis Class Report**

***Revision History***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Authors** | **Description of Change** | **Sections** | **Rev** | **Date** |
| Henry Y | Initial Document |  | O | 2/28/18 |
| Michael P | Updated Document |  | 1 | 2/28/18 |
| Henry Y, Michael P,  Jesus R | Changed block functionality, updated class diagram |  | 2 | 3/6/18 |

**Table of Contents**

[1 Team Description](#_30j0zll) 4

[2 Project Description](#_1fob9te) 4

[2.1 Analysis Class Diagram](#_3znysh7) 5

[2.2 Analysis Class List](#_2et92p0) 6

[2.2.1](#_tyjcwt) Calendar Class 6

[2.2.2](#_3dy6vkm) Block Class 6

2.2.3 Primary Schedule Class 7

2.2.4 Task Class 8

# Team Description

|  |  |
| --- | --- |
| **Team Member Name** | **Email Address** |
| Eric Guzman | eg70889@csu.fullerton.edu |
| Adam Weesner | klaviam@gmail.com |
| Jesus Rodriguez | jrjssrdgz@csu.fullerton.edu |
| Michael Perez | mperez980089@yahoo.com |
| Henry Yang | hkyang@csu.fullerton.edu |
|  |  |

# Project Description

Statement of Purpose: To make it convenient for users to schedule and keep track of their progress on activities.

Detailed Description: This application will allow the user to input their free time for the week. The free time is the user’s schedule that represent the times they are available to perform tasks. The activities are represented as “blocks” on the calendar’s UI. After inputting their free time, the user can create “tasks” to be scheduled: activities that the user needs to complete. Tasks have a name, description, due date, list of start sessions, and estimated amount of time it takes. When a task is created, it is put into the “unassigned task” list. All tasks can have the system sort them into the freetime on the user’s calendar. Two algorithm types will be available: “Slow and steady”, which will cut the task into small segments and spread the segments out evenly, and “Ripping the bandaid”, which will assign the tasks in large segments. Users will be reminded an hour before each task start session that it is approaching. Users are able to add or subtract the time it takes to complete a task, as well as delete a task entirely.

## Analysis Class Diagram

## Analysis Class List

|  |  |
| --- | --- |
| **Sequence Number** | **Analysis Class** |
| 1 | Calendar |
| 2 | Block |
| 3 | Task |

### Calendar Class

Description: UI element that contains the user’s Free Time and Tasks, both of which are represented as blocks. This is where Tasks can be created, and the Primary Schedule can be edited. The Calendar is also what keeps tracks of all assigned/unassigned tasks.

Methods:

* EditFreeTimeBlocks
* EditTaskBlocks
* CreateTask

Attributes:

* List of free time blocks
* List of unassigned tasks
* List of assigned tasks

### 2.2.2 Block Class

Description: Represents the segments of time that is available for the user to perform tasks.

Methods:

* SetStartTime
* SetEndTime
* GetStartTime
* GetEndTime

Attributes:

* Start Time
* End Time

**2.2.3 Task Class**

Description: Tasks are the activities the user wants to complete. Created tasks are placed in the “Unassigned Tasks” list. Unassigned tasks can be individually viewed, where they can be assigned to the Calendar. How they are assigned to the calendar is determined by their Scheduling Algorithm. Assigned tasks can also be individually viewed, and can be unassigned from the Calendar.

Methods:

* SetName
* SetDescription
* SetDueDate
* ShiftForward (Shift forward the time to completion)
* ShiftBackward (Shift backward the time to completion)
* SetAlgorithm
* SetColor
* AssignTask
* UnassignTask
* DeleteTask
* DisplayReminderAlert (Function is called to alert the user a due date is coming up)
* ToggleAlerts

Attributes:

* Name
* Description
* Due Date
* Time Needed to Complete the Task
* Due date reminder intervals
* Scheduling Algorithm (“Slow and Steady” or “Rip the Bandaid”)
* Color
* Assigned (True/False)
* Alerts On (True/False)