**Group 1 SRS**

**1/18/2019**

**Tyler Green, Tyler Milan, Bryce di Geronimo, Michael Zhang, Jarvis Dong**

**Table of Contents -Needs Dates and Time**

1. **Problem Statement**
2. **Users**
   1. **Description**
   2. **Expectations**
   3. **Prior Knowledge**
   4. **Current Technology Usage**

**2.5 Use Cases**

**2.6 Example Users**

1. **Description of Requirements**
   1. **Functional**
      1. **Required**
      2. **Suggested**
   2. **Non-Functional**
      1. **Required**
      2. **Suggested**
2. **Problem Statement**

With burgeoning developing of computing and internet, it’s prevailing for people to arrange and track their daily events and activities with their phones and computers. Thus, a calendar application nowadays is no longer a traditional booklet with limited date information, but it also has become an interactive software that allows users to edit the date information with their own wishes. Users can read and write information anywhere and anytime as long as internet and devices are accessible.

Many users have an overwhelming amount of tasks to complete on a daily basis. Without the ability to track their tasks and update what needs to get done, life can become very difficult. A calendar application that allows these users to track their tasks provides organization and a way to measure progress which makes a large amount of tasks much more manageable.

The application will have the initial functionality:

1. Create, update, and delete events
2. View all events for today, tomorrow, and the following day
3. Keep track of all changes and give the user the option to save all data before exiting the program

Throughout the design and development of the application, there will be a high priority on a clean and scalable solution that can evolve to the demands of the users.

1. **Users**
   1. **Description**

The target users are all people who currently use or previously have used electronic calendars or reminders to keep track of their events and obligations. They will use the application to keep track of all events they did on their previous calendars. These users will not have the time to mess around with a slow application and are looking for something quick and easy to navigate.

* 1. **Expectations**

The user will expect a clean, intuitive user interface that makes it easy for them to interact with the application. They will expect to be able to create, modify, and delete events with speed and ease. They expect not to have any lag or bugs in the program and to be able to complete creating an event in a short amount of time without any hassle.

* 1. **Prior Knowledge**

We are assuming users have the ability to use a computer effectively enough to start and navigate an application. Users will understand how a calendar works and how to enter dates and times into a computer. Besides these basic knowledge requirements, users do not need a lot of prior knowledge to use this application.

* 1. **Current Technology Usage**

These users, like most people today, will have familiarity with many forms of technology. They will use computers for many daily activities such as checking emails and using business applications which will help them navigate this application.

**2.5 Use Cases**

* First, the user will access the calendar from the website which will be hosted on our IX server. The user will identify the date in which the event is to be added and will double click in the square box. A pop up box will be generated and a start and end time must be selected as well as a description of the event in order to be submitted. An optional category for priority of event such as low, medium, or high will be available but is not required. Once the event has been submitted to the database, the calendar will show the event until it is deleted.
* Second, a user can modify a pre existing event by clicking on the event. The same popup box that was generated to create the event will appear with the existing information already in it. The user can then change the time, description, or priority of the event to be submitted and saved to the database.
* The user can delete an event by simply clicking on the event which will generate the popup box for the event. A button labeled delete will be available and once clicked, the user will get a confirmation message asking them if they are sure they want to delete this event. If yes is selected, the event will be deleted from the database and will no longer show. If canceled, the event will remain.

**2.6 Example Users**

1. Mary is a busy attorney who is consistently crunched for time and forgets things often due to being overwhelmed. As a way to stay organized and to make sure she completes all his daily tasks, she wants to start using a calendar application to put all of his daily events on. She is looking for a simple and clean application that allows her to add events, update them when needed, and delete them when they have been completed.
2. John is a secretary at a dentist office and needs an application to keep track of all the appointments for this year. He is looking for an application that would allow him to schedule appointments based on the date and time of the appointment. He needs to be able to update and delete appointments to accomodate for cancellations as well as changes in schedule for customers.
3. Sam is a computer science student who needs to keep track of her increasingly hectic project schedule, home life, and work hours. Sam needs a calendar that can help organize her events by category as well as by importance. She organizes her projects by priority, and differentiates between the different obligations in her life.

**4. Description of Requirements**

**4.1 Functional**

**4.1.1 Absolutely Required**

**1)** User can create new event on each day

**2)** User can easily see what is on the calendar for today, tomorrow, and the next day

**3)** Program keeps track of changes and gives the user the option to save the data before exiting the program

**4)** Users are able to edit existing and delete event

**5)** The system should enter and retrieve or exchange properly formatted calendar data without loss or error.

**6)** The calendar software should ensure that data entered is consistent with HH:MM and DD/MM/YYYY date time standard. Optionally, events might be expected to occur on the quarter hour (such as 1:15PM, 1:30PM, etc.)

**7)** It should be possible to edit the name (or other fields) at any time while keeping the associated data.

**8)** It should be possible to make likely changes to the system without extensive re-design. Simple changes should require changes to only a single system component (module).

**4.1.2 Not Absolutely Required**

**9)** Permit users to define categories for events (such as "work", "home", etc.).

**10)** Permit users to define priorities for events (such as "high", "medium" , or "low").

**11)** The application will be responsive (i.e. maintain a clean look across all screen sizes)

**12)** Allow users to add multiple events with repeated pattern on a single popbox.

**4.2 Non-Functional Requirements**

**4.2.1 Absolutely Required**

**13)** The software should respond to user requests at a speed equal to or better than competing applications, in any event not to exceed 500ms.

**14)** Application will work on Linux with kernel 4.0 or newer, OSX(MacOS) with version 10.9 or later and on Windows 10.

**15)** System user interface responds to user interactive at a adequate speed.

**16)** System ensures data display and I/O properly formatted without error and

loss.

**17)** System consists of flexible components that is easy to extend and modify such as add, delete and .

**18)** System is broken down into separate modules that can be tested individually as well as a whole

**4.2.2 Not-Absolutely Required**

**19)** compati browsersuch as FireFox, Chrome, IE

**20)** A clean background picture to add to the design of the application