Milestone 5: Test Planning

Team Name: Sweet Victory

Members: Rahul, Ian, Chad, Chris, Jacob, Brooke

Title of Project : MicroManage (μ)

User Acceptance Tests:

1. **Adding/Editing Events:**
   - User should be able to intuitively use the calendar
   - All icons on the app are easy to understand
   - Adding events should work without fail. This means that events can have the same name and time, but should not break anything
2. **Login/Registration:**
   - Users should be able to log in easily
   - Data from forms should be verified such that invalid input doesn't get stored in the DB
3. **Email/timer reminder:**
   - Reminder should work as intended and actually send out an email reminder at the specified time

#	Description	Test Steps	Data Type	Expected Results	Pass/Fail
1	Checks response when events are added for the calendar.	1. Enter data into appropriate form 2. Click Add Event 3. Enter the event name 4. Enter the event date 5. Enter the start/end times for the event 6. Select the urgency and approriate color 7. Click Add Event	Event_name = "Summer Solstice" Event_length = HOUR:MIN:SEC Event_startTime = YEAR-MONTH-DAY Event_endTime = YEAR-MONTH-DAY Event_urgency = 4 Event_color = BLUE	User should be able to sucessfully add events to the calendar.	Pass
2	Checks response when events are edited/deleted from the calendar.	1. Enter data into appropriate form 2. Click Edit Event 3. Edit the event name (option) 4. Edit the event date (option) 5. Edit the start/end times for the event (option) 6. Edit the urgency and approriate color (option) 7. Click Update Event	Event_name = "Winter Equinox" Event_length = HOUR:MIN:SEC* Event_startTime = YEAR-MONTH-DAY* Event_endTime = YEAR-MONTH-DAY* Event_urgency = 5 Event_color = RED*	User should be able to sucessfully edit or delete events from the calendar.	Pass
3	Checking user login with valid data.	1. Enter data into appropriate form 2. Enter User Credential 3. Enter Password Credential 4. Enter Email 4. Click Submit/Enter	User_Name = "user" User_Pass = "password" User_Email = "mayflowers@gmail.com"	User should be able to login successfully.	Pass
4	Checking user login with invalid data.	1. Enter data into appropriate form 2. Enter User Credential 3. Enter Password Credential 4. Enter Email 4. Click Submit/Enter	Case 1: checks for uniqueness User_Name = "user" User_Pass = "password" User_Email = "mayflowers@gmail.com" Case 2: checks for valid email entry User_Name = "unique" User_Pass = "unique" User_Email = "springshowers@gmail..co_"	User login should be unsuccessful.	Pass
5	Check to determine if the email reminder functionality is working.	1) Enter data into appropriate form 2) Set time to be reminded 3) Verify email gets sent when specified time reached (Mailjet API)	User_Email: (private credential) User_EmailPass: (private credential) Set_Time: (according to test conduction time)	Email(s) should be sent to designated address promptly/when expected.	Pass
			An (*) indicates a requirement to input an updated version of data within the paramter. Due to variance in test scheduling, date objects are not specified (left to the discretion of the tester).		