Project State for Deliverable 2

Group 1: Robert, Lachezar, Duong, Bethany

User Stories:

1. As a Park Manager I want to submit a new job.
   Extend code is written: Fully implemented. A user logs in with their email and is presented with a menu that allows for this function. They are prompted for all the relevant fields for the job and if it's able to be put in the master calendar, it is.
   What is tested: Unit tests ensure that no more than 30 Jobs can exist in the calendar at a single time.
   Verified via user interface: Yes. Users can verify the new job exists with User stories 2, 3, 8 and 9.
2. As a Park Manager I want to delete a job.
   Extend code is written: Fully implemented. A user logs in with their email and is presented with a menu that allows for this. They are asked for what job they want to delete, and then it is removed from the calendar.
   What is tested: Unit tests ensure that a job is only deleted if it's equal to the requested job deletion.
   Verified via user interface: Yes. Users can verify the job is gone by either trying to delete it again or attempting to edit.
3. As a Park Manager I want to edit the details of a job.
   Extend code is written: Fully Implemented. Park Managers are able to edit any job they have previously created.
   What is tested: Jobs are only edited in the calendar if they are equal to the origin job.
   Verified via user interface: Yes. Users can verify that the job is changed by viewing it again.
4. As an Volunteer, I want to view a summary of all upcoming jobs.
   Extend code is written: Fully implemented. A volunteer logs in with their email and then is able to view all jobs in the calendar.
   What is tested: Volunteer emails are verified through the login method.
   Verified via user interface: Yes. The volunteer can see this through menu commands.
5. As an Volunteer, I want to view details of a selected upcoming job.
   Extend code is written: Fully implemented. Volunteers can pick a job from the master list to view the details of.
   What is tested: Input checked is checked to make sure the correct job's details are put up.
   Verified via user interface: Yes. Volunteers can check this function by their menu.
6. As a Volunteer I want to volunteer for a job.
   Extend code is written: Fully implemented. Volunteers can volunteer for a job after viewing its details.

What is tested: The job is checked to make sure there's still enough room for more volunteers.

Verified via user interface: Yes. Volunteers can check this via User Story 7.

7. As a Volunteer I want to view the jobs I am signed up for.
   Extend code is written: Fully implemented.
   What is tested: Volunteers keep track of the jobs they are signed up for.
   Verified via user interface: Yes. This is one of the functions in their menu.
8. As a Park Manager I want to view a summary of all upcoming jobs in the parks that I manage.
   Extend code is written:
   What is tested:
   Verified via user interface:
9. As a Park Manager I want to view the Volunteers for a job in the parks that I manage.
   Extend code is written: Fully implemented. Park Managers can view volunteers for any particular job through their menu.
   What is tested: Only parks managed by that particular manager are checked for jobs and Volunteers.
   Verified via user interface: Yes. Park Managers can test this with their menu.
10. As an Urban Parks staff member, I want to search volunteers by last name.
    Extend code is written: Fully implemented. An Urban Parks Employee logs in and is able to see the master list of volunteers by iterating over all the jobs.
    What is tested: Volunteers are tested as they are iterated over to make sure the same volunteer is not returned more than once.
    Verified via user interface: Yes. Urban Parks Staff Members can check this function from the menu.
11. As an Urban Parks staff member, I want to view a summary of all upcoming jobs.
    Extend code is written: Fully implemented. An Urban Parks Employee logs in and is able to see the master list of jobs from the Calendar.
    What is tested: JUnit tests ensure the staff member gets all jobs from the master list.
    Verified via user interface:
12. As an Urban Parks staff member, I want to view details of a selected upcoming job
    Extend code is written: Fully implemented. An Urban Parks Staff Member can view the specific details of any job from the master list.
    What is tested: That the park employee is seeing the details of the correct requested job.
    Verified via user interface: Yes.

Business Rules:

1. A job may not be added if the total number of pending jobs is currently 30.
   Extend code is written: Fully implemented. Park Managers cannot create the job if the Calendar is too full.
   What is tested: We have JUnit Tests attempting to fill the calendar with 100 jobs, and then verifying that only 30 are able to be created.

Verified via user interface: Yes. A Park Manager will be warned if there are too many jobs currently in the Calendar.

2. A job may not be added if the total number of pending jobs during that week (3 days on either side of the job days) is currently 5. In other words, during any consecutive 7 day period there can be no more than 5 jobs.
   Extend code is written: Fully implemented. Park Managers cannot create jobs if the job would cause the calendar to exceed 5 jobs per week.
   What is tested: The dates 3 days before and after are calculated and then the calendar is iterated over to check how many jobs are already in that time span. If it's 5 or more, then the Calendar rejects the new job.
   Verified via user interface: Yes. Park Managers can attempt to create 5 jobs on the same day to easily verify this.
3. A Volunteer may not sign up for a work category on a job if the maximum number of Volunteers for that work category has already been reached.
   Extend code is written: Not implemented. This business rule was discussed in class, and it seemed that there was a consensus that this wasn't actually going to be implemented.
   What is tested: Nothing for the reason stated above.
   Verified via user interface: Not verified for the reason above.
4. A job may not be scheduled that lasts more than two days.
   Extend code is written: Fully implemented. Park Managers cannot add jobs if they fail the condition.
   What is tested: We have tests to check that the duration is not longer than two days.
   Verified via user interface: Yes. Attempting to create a job with a greater length will fail.
5. A job may not be added that is in the past or more than three months in the future.
   Extend code is written: Fully implemented. ParkManagers cannot add jobs if they fail either of those conditions.
   What is tested: The presumed date is checked against dates that have already occured and dates 90 days from now. If it fails either test, the job is not created.
   Verified via user interface: Yes. Park Managers will encounter this when attempting to get around either condition.
6. A Volunteer may not sign up for a job that has passed.
   Extend code is written: Fully implemented. Volunteers cannot sign up for jobs that have passed.
   What is tested: A check to make sure that job has not already passed.
   Verified via user interface: Yes. Volunteers will encounter this while attempting to sign up for a job that has already occured.
7. A Volunteer may not sign up for two jobs on the same day.
   Extend code is written: Fully Implemented. Volunteers cannot sign up for a job that is on the same day as another job they had previously signed up for.
   What is tested: The job they attempt to volunteer for has it's starting date checked against all other jobs the volunteer has coming up.

Verified via user interface: Yes. Volunteers will encounter this while attempting to sign up for two jobs on the same day.

8. A Park Manager can create jobs only for those parks that he/she manages.

   Extend code is written: Fully Implemented. Park Managers only have access to their own parks.

   What is tested: Nothing. This is enforced via the interface.

   Verified via user interface: Yes. ParkManagers can only see their own parks.