System and Unit Test Report
Boundless Brilliance Internal App
Team Name: Team Rocket
Date: 12/11/2018

**Feature 1:** Be able to create a unique account.

A. User story 2 from sprint 1: As a member of the organization, I want to be able to create an account where I can access the integrated scheduling platform and presentation data tracker.
    a. Scenario 1: Account can be created
        i. Given: A user wants to create an account
        ii. When:
            1. Password is >= 7 characters
            2. Email is not already in use
            3. Email is valid email
            4. All fields are filled
        iii. Then: Account is created in Firebase and user navigated back to login screen.
    b. Scenario 2: Account cannot be created
        i. Given: A user wants to create an account
        ii. When:
            1. Password is < 7 characters
            2. Email is already in use
            3. Email is not a valid email
            4. Any field is empty
        iii. Then: account cannot be created and a toast notification notifies user of the error.

**Feature 2:** Be able to log into a previously created account.

A. User story 2 from sprint 1: As a member of the organization, I want to be able to use an account where I can access the integrated scheduling platform and presentation data tracker.
    a. Scenario 1: User enters correct login information
        i. Given: A user attempts to login
        ii. When: User enters accurate email and password combination
        iii. Then: The presentation data starts loading and app switches to ListView
    b. Scenario 2: User enters incorrect login information
        i. Given: A user attempts to login
        ii. When: User enters inaccurate email and password combination
        iii. Then: Toast alerts user of failed login attempt

**Feature 3:** View all scheduled presentations

A. User story 5 from sprint 2: As a college student presenter, I want to be able to access a basic integrated scheduling platform, so that I know when and where my next presentation will be.
   a. Scenario 1: User navigates to activity containing list of presentations
      i. Given: A user has presentations assigned to them and their chapter
      ii. When: The user logs in
      iii. Then: The next activity loaded to the screen displays a list of all the presentations pulled from the database
   b. Scenario 2: User searches for specific presentation
      i. Given: A user wants to search for a specific presentation
      ii. When: The user clicks on the search bar and types what they want to search by
      iii. Then: The list view displays the appropriate presentation that the user wanted
   c. Scenario 3: User filters presentations to see only their own
      i. Given: The user wants to see their presentations
      ii. When: The user clicks on the tab with their name in it
      iii. Then: Only their presentations are displayed
   d. Scenario 4: User filters presentations to see presentations only from their own chapter
      i. Given: The user wants to see their chapter presentations
      ii. When: The user clicks on the tab with their chapter name in it
      iii. Then: Only their chapter presentations are displayed.
   e. Scenario 5: Manager filters presentations to see all chapters
      i. Given: The manager wants to view all chapters
      ii. When: The manager clicks on the tab called “All”
      iii. Then: All presentations are displayed

B. User story 12 from sprint 4: As a developer, I want to refine our Firebase queries so that the presentations displayed: in chronological order, with more data.
   a. Scenario 1: Any user looks at the list of presentations
      i. Given: The user can see an amount of presentations
      ii. When: The user looks at the presentations
      iii. Then: The presentations are listed nicely in chronological order

**Feature 4:** View details about each presentation

A. User story 9 from sprint 3: As a user, I want the scheduling logistics for my presentation (date, time, classroom #, etc) to be easily accessible because I often have multiple presentations in one day and need to know exactly where to go and when.
   a. Scenario 1: User has a presentation soon and wants to see what classroom the presentation is in, as well as more details.
      i. Given: The user has an upcoming presentation
      ii. When: The user clicks on the presentation in the list view

iii. Then: A new screen opens with additional information about the presentation that was not in the list view

**Feature 5:** Easily input all post-presentation data (as a user)

A. User story 10 from sprint 4:  As a presenter, I want to be able to access another screen to submit feedback about a presentation after it is over.

a. Scenario 1: Presenter wants to access a screen where they can submit a survey and the presentation date is after the current date

i. Given: The presentation date/time is after the current date/time

ii. When: The presenter attempts to click the button "Feedback"

iii. Then: The "Feedback" button is disabled

b. Scenario 2: Presenter wants to access a screen where they can submit a survey and the presentation date is before or equal to the current date and time

i. Given: The presentation date is before or equal to current date/time

ii. When: The presenter clicks the button “Feedback”

iii. Then: The presentation survey is displayed for the presenter to fill out

**Feature 6**: Send the post-presentation data that the presenters input to an online database that can be uploaded to Salesforce by the product owner (backend to feature 5)

A. User story 11 from sprint 4: As management, I would like to see all the information post presentation saved in a text file to know more about how the presentations went. (10 hours)

a. Scenario 1: Management accesses a file on Storage Firebase with all the information in the form of a csv file

i. Given: Management has access to Storage Firebase

ii. When: Management accesses the Storage Firebase

iii. Then:  management receives the information from the user survey in the form of csv file with

**Testing:** Results of these tests are in “testing” file.