Punctual

AWSM

Always Working So Much

Test Cases

Spring ‘15

Caitlin Cagampan
Emily Deng
Jacob Terrado
Jessica Lin
Rajiv Pasricha
Kimberly Ly
Amr Kahhaleh
Harsha Rao
Ryan Milkovits
Merick Markoem
Adina Wollner

Index

Color Coding Legend

Account Credentials
UI Button/TextView/TextField
Screen

Basic Function (BF):

Sign Up for Account

Title: Sign Up for Account **[TCBF1]**

Description: User decides to sign up for a new account. The user is asked to enter new valid credentials (unique Username, and enter an email address and a password). After successfully entering valid credentials, the user is asked to answer a quick quiz that would allow our application to learn their habits.

User Goals: To create a Punctual account and start using the application.

Desired Outcome: User shall have a new unique account to start using the application with. The system should securely save the user's info and credentials on the database.

- **Priority Level**: 1
- **Status:** implemented & tested ✓

Actors:

1. User (Primary)
2. System
3. Parse database

Requirements:

1. Sign up screen [RBF-1]
2. Unique User Credentials [RBF-2]
3. Check Email Format [RBF-3]

Preconditions:

1. User has opened the app and is at the Sign up screen.
2. User has internet connection.
3. Account with username test and e-mail test@test.com does not exist.
4. Account with username awsmtest and e-mail awsmtest@test.com exists. [Alt. Path 1]
5. Account with username test2 and e-mail test2@test.com does not exist. [Alt. Path 2]

Trigger: User has clicked on New? Sign Up Here! from Log in screen

Failed Conclusions:

1. Username/E-mail address already exists
2. User has chosen to reject the Terms of Service

Workflow:

1. The User shall input “test” as the username, “test@testme.com” as the email address, and “test” as the password.
2. The User shall click the SIGN UP button.
3. The System shall verify with Parse to validate that the username/email address are unique.
4. The Parse database shall save the User credentials into the database.
5. The System shall bring the user to the Terms of Service screen.
6. The User shall click the ACCEPT button to agree with the Terms of Service.
7. The System shall verify that ACCEPT has been clicked.
8. The System shall bring the user to the Signup Quiz screens starting with the Question 1 screen.

Result: This test passes when the System displays the Question 1 screen. *Note – Please refer to README file after executing this test.

Alternate Path 1 Workflow [Failed Conclusion 1]:

1. The User shall input “awsmtest” as the username, “awsmtest@test.com” as the email address, and “awsmtest” as the password.
2. The User shall click the SIGN UP button.
3. The System shall verify with Parse to validate that the username/email address are unique.
4. The System shall bring the User to a screen that prompts them to try to Sign up again indicating to try another username or email.
5. The User shall click RETURN TO SIGN UP SCREEN.

Result: This alternate test passes when the System brings the User to the screen described in Alt Path 1 Step 4 and successfully redirects the User to the Sign up screen after executing Alt Path 1 Step 5.

Alternate Path 2 Workflow [Failed Conclusion 2]:

1. The User shall input “test2” as the username, “test2@testme.com” as the email address, and “test2” as the password.
2. Execute Workflow steps 2-5.
3. The User shall click the REJECT button to reject the Terms of Service.
4. The System shall verify that REJECT has been clicked.

5. The System shall bring the User to a screen that prompts them to accept the Terms of Service in order to use the application.
6. The User shall click GO BACK TO TERMS.

Result: This alternate test passes when the System brings the User to the screen described in Alt Path 2 Step 5 and successfully redirects the User to the Terms of Service screen after executing Alt Path 2 Step 6.

Sign Up Quiz

Title: Sign Up Quiz **[TCBF2]**
Description: This is a brief survey that is displayed to the user when sign up for a new account. It basically asks them a few questions to try to learn their habits and how they prefer to use the application to determine the best way to help them stay punctual and arrive on time to their daily Quests (events).

User Goals: Make the application as personal as possible by teaching it their habits.

Desired Outcome: To learn the most about the user's habits and configure the application's notification settings to make the user more punctual than ever.
Details:

- **Priority Level**: 1
- **Status**: implemented & tested ✓

Actors

1. System (Primary)
2. User
3. Database (Parse)

Requirements

1. Initialize Default Settings [RBF-4]
2. Average Tardiness [RNS-1]
3. Default Locations [RNS-2]
4. Preferred Travel Method [RNS-3]

Preconditions

1. User had just finished signing up with username test3, email test3@test.com, and password test3.
2. [TCBF1] has been satisfied with credentials from Precondition 1.
3. User is at the Question 1 screen.

Trigger: User has clicked ACCEPT on the Terms of Services screen.
Failed Conclusions

1. User has decided to leave an answer blank and continue through the quiz.
2. User has decided to not take the quiz and pressed Device Back Button and went back to the Sign up screen.

Workflow

1. The User shall choose Always right on time!
2. The User shall click the NEXT button.
3. The System shall check for the option chosen and saves this information.
4. The System shall bring the User to the Question 2 screen that prompts them with a question.
5. The User shall choose 10 minutes
6. The User shall click the NEXT button
7. The System shall check for the option chosen and saves this information.
8. The System shall bring the User to the Question 3 screen that prompts them with a question.
9. The User shall choose Let’s go! I’m so ready
10. The User shall click the NEXT button.
11. The System shall check for the option chosen and saves this information.
12. The System shall bring the User to the Question 4 screen that prompts them to enter default addresses.
13. The User shall enter “Pepper Canyon Hall” in the Home Address text field.
14. The System shall display a list with the autocompleted address.
15. The User shall click the only address option in the list.
16. The System shall write out the full address in the Home Address text field.
17. The User shall click the NEXT button.
18. The System shall check for addresses entered and saves this information.
19. The System shall bring the User to the Question 5 screen that prompts them with a question.
20. The User shall choose Car.
21. The User shall click the NEXT button.
22. The System shall bring the User to the Welcome screen.
23. The User shall click the I’m Ready! button.
24. The System shall bring the User to the Home screen.

Result: This test passes when the System displays the Home screen after completing the sign up quiz.

Alternate Path 1 Workflow [Failed Conclusion 1]

1. The User shall click the NEXT button.
2. The System shall check that no option was chosen and shall prompt the User with a message, “Please choose an answer.”
3. Execute Workflow Steps 1-24.

Result: This alternate test passes when the System displays the Home screen after completing the sign up quiz.

Alternate Path 2 Workflow [Failed Conclusion 2]

1. The User shall click the Device Back Button.
2. The System shall bring the User to the Sign Up screen.

Result: This alternate test passes when the System displays the Sign Up screen as described in Alt Path 2 Step 2.

Log Into Account

Title: Log Into Account **[TCBF3]**
Description: User decides to sign in. The User shall provide an existing user's credentials. The System verifies the validity of the username/password combination and then either logs the user in and pull their account's information or display an error.
User Goals: To sign into their account on the application.
Desired Outcome: System shall securely sign the user into their right account and pull their personal information or display an error message if login failed.
Details:

- **Priority Level**: 1
- **Status**: implemented & tested ✓

Actors:

1. User (Primary)
2. System
3. Parse database

Requirements:

1. Unique User Credentials [RBF-2]
2. Login Screen [RBF-5]
3. Reset Password [RBF-6]

Preconditions:

1. User has opened the app and no account is currently signed on.
2. User is at the Log in screen.
3. The awsmtest account exists with password awsmtest.
4. User has internet connection.

Trigger: User shall either logout from an existing account or launch the application for the first time ever.
Failed Conclusions:

1. Username / password combination is wrong

Workflow:

1. The User shall input "awsmtest" as the username.
2. The User shall input "awsmtest" as the password.
3. The User shall click LOG IN
4. The System shall verify with Parse that the credentials are valid.

5. The System shall bring the User to their Home screen.

Result: This test passes when the System displays the User’s Home screen.

Alternate Path Workflow [Failed Conclusion 1]:

1. Execute Workflow step 1.
2. The User shall input “awsmtes” as the password.
3. The User shall click LOG IN
4. The User shall prompt the User with a message that displays “Invalid Username/Password Combination. Please Try Again!”

Result: This alternate test passes when the System displays the error message described in Alt Path Step 4.

Reset Password

Title: Reset Password **[TCBF4]**

Description: If user forgets their password, the System shall give user the ability to retrieve their password by entering their email address and receiving an email with reset password link, which re-directs them to update their password directly on the database website (Parse). The System shall return back to the log in screen to allow the User to sign in with the new password.

User Goals: Reset password to a new password that they remember.

Desired Outcome: User shall have a new password and System shall replace old password with new one in the database.

Details:

- **Priority Level**: 1
- **Status**: implemented & tested ✓

Actors:

1. User (Primary)
2. System
3. Parse database

Requirements:

1. Reset password [RBF-6]
2. Verify Email Format [RBF-3]

Preconditions:

1. User is at the Reset Password screen.
2. User has internet connection.
3. The testme account exists with email amistermetoo@gmail.com
4. The User is logged into the account with user testme2 and password testme2 [Alt Path 3]
5. [TCBF3] Preconditions apply. [Alt Path 3]
6. User is at the Settings screen. [Alt Path 3]

Trigger:

1. User clicks Forgot Password? Click Here! from the Log In screen.
2. User navigates to the Settings screen and expands the Account Settings by clicking the respective text view. [Alt Path 3]

Failed Conclusions:

1. No account match is found for provided email address on Parse database
2. User does not enter anything in the email field.

Workflow:

1. The User, testme, shall input “amistermetoo@gmail.com” as the email.
2. The User shall click RESET PASSWORD
3. The System shall verify with Parse that an account is attached to the email.
4. The System shall prompt the User with a success message indicating that a password reset email has been sent.
5. The System shall redirect the User to the Log In screen.
6. The User shall access the indicated email account.
7. The User shall click the link within the email corresponding to the password reset request.
8. The System shall redirect the User to a webpage that prompts them to enter a new password.
9. The User shall enter “testme2” as the new password.
10. The User shall click Change Password
11. The System shall redirect the User to a webpage that displays “Successfully updated your password!”

Result: This test passes when the User has been brought to the webpage described in Workflow Step 11 and the User has returned to the login screen after executing Workflow Step 5.

Alternative Path 1 Workflow [Failed Conclusion 1]:

1. The User shall input “resetpwtest@test.com” as the email.
2. The User shall click RESET PASSWORD.
3. The System shall check with Parse that an account is attached to the email.
4. The System shall prompt the User with an error message that displays “Password Reset E-mail Not Sent. Please check that your E-mail address is valid”

Result: This alternate test passes when the error message described in Alt Path 1 Step 4 is displayed.

Alternative Path 2 Workflow [Failed Conclusion 2]:

1. The User shall leave the email field empty.
2. The User shall click RESET PASSWORD.
3. The System shall prompt the User with an error message that displays "Please enter an email so we can send the password reset link"

Result: This alternate test passes when the error message described in Alt Path 2 Step 3 is displayed.

Alternative Path 3 Workflow:

1. The User, testme2, shall click Reset Password.
2. The System shall prompt the User with a success message indicating that a password reset email has been sent.
3. The User shall access the email account attached with their Punctual account, awsmtest@yahoo.com
4. The User shall click the link within the email corresponding to the password reset request.
5. The System shall redirect the User to a webpage that prompts them to enter a new password.
6. The User shall enter "testme" as the new password.
7. The User shall click Change Password
8. The System shall redirect the User to a webpage that displays "Successfully updated your password!"

Result: This alternate test passes when the User has been brought to the webpage described in Alt Path 3 Step 8.

Edit Settings

Title: Edit Settings **[TCBF5]**
Description: The User shall be able to edit the default settings of the app to customize the different settings and features that the app offers.
User Goals: Customize the app settings in a way that best fits them.
Desired Outcome: Make settings as customizable as possible to best fit User's needs.
Details:

- **Priority Level**: 1
- **Status**: implemented & tested ✓

Actors:

1. User (Primary)

Requirements:

1. Settings Screen [RBF-12]

Preconditions:

1. User is logged into the account with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User has navigated to the Settings screen from the menu bar.

Trigger: From the menu bar, User clicks on Settings.
Failed Conclusions:

1. Email is invalid.

Workflow:

1. The User, awsmtest, shall click Notification Settings or the (›) icon.
2. The System shall expand the notification settings displaying the full options to edit.
3. The User shall click the toggle button for the Enable All Notifications settings to set this to false.
4. The System shall make the button toggle indicating that this setting is set to OFF.
5. The User shall click the spinner associated with the Change Number of Notifications settings.
6. The System shall display the list of options to choose from.
7. The User shall click 4
8. The User shall click Save

9. The System shall save the User’s new notifications settings display a toast message, “Saved.”
10. The User shall click Travel Settings or the (❯) icon.
11. The System shall expand the travel settings displaying the full options to edit.
12. The User shall click the spinner associated with the Change Default Travel Method settings.
13. The System shall display the list of options to choose from.
14. The User shall click Walk
15. The User shall click the spinner associated with the Change Preparation Time settings.
16. The System shall display the list of options to choose from.
17. The User shall click 30
18. The User shall click the spinner associated with the Change Lateness Offset settings.
19. The System shall display the list of options to choose from.
20. The User shall click 15
21. The User shall click Save
22. The System shall save the User’s new notifications settings display a toast message, “Saved.”
23. The User shall click Saved Addresses or the (❯) icon.
24. The System shall expand the saved addresses settings displaying the full options to edit.
25. The User shall clear the text field of the Home setting.
26. The User shall enter “Sixth College Apartments” in the Home Address text field.
27. The System shall display a list with the autocompleted address.
28. The User shall click the only address option in the list.
29. The System shall write out the full address in the Home Address text field.
30. The User shall click Save
31. The System shall save the User’s new notifications settings display a toast message, “Saved.”
32. The User shall click Account Settings or the (❯) icon.
33. The System shall expand the account settings displaying the full options to edit.
34. The User shall clear the text field of the Change Email setting.
35. The User shall enter “awsmtest@testme.com”.

36. The User shall choose the Current working Pal option associated with the Choose which Punctual Pal to display: setting.
37. The User shall click Save
38. The System shall save the User’s new notifications settings display a toast message, “Saved.”

Result: This test passes when every setting in the Settings screen has been successfully changed and updated subject to the Workflow steps 9, 22, 31, and 38.

Alternate Path 2 Workflow [Failed Conclusion 1]

1. The User shall click Account Settings or the (❯) icon.
2. The System shall expand the account settings displaying the full options to edit.
3. The User shall clear the text field of the Change Email setting.
4. The User shall enter “awsmtest@newtest”.
5. The User shall click Save
6. The System shall check the e-mail entered and display to the User, “Failed to save Account Settings”.

Result: This alternate test passes when the System prompts the User with the error message described in Alt Path 2 Step 6.

Notify User of Upcoming Quest

Title: Notify User of Upcoming Quest **[TCNS1]**
Description: The System shall calculate the most optimal times to notify a user for each quest. Notifications include when to get ready and when to leave.
User Goals: The User wants to be reminded at just the right time in order to reach their quests on time.
Desired Outcome: Be able to smartly and consistently notify the User of upcoming quests, when to get ready, and when to leave.
Details:

- **Priority Level**: 1
- **Status**: implemented & tested ✓

Actors: None
Requirements:

1. Send Quest Notifications [RNS-4]
2. Get Travel Time from Google Maps API [RNS-4]

Preconditions:

1. User is logged into the app with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User is at the Home screen.
4. User has global Enable All Notifications setting set to ON
5. User has global Number of Notifications setting set to 4
6. User has global Preparation Time setting set to 30.
7. User has global Lateness Offset setting set to 15.

Trigger: Cloud schedules notifications to be pushed at the calculated time.
Failed Conclusions:

- None

Workflow:

1. The User, awsmtest, shall click the (+) button on the top right corner.
2. The System shall bring the user to the Add Quest screen.
3. The User shall input “Test Notif Quest” in the Quest Title text field.
4. The User shall leave the Start Location option as Home.
5. The User shall enter “Center Hall” in the Quest Location text field.
6. The System shall display a list with autocompleted addresses.

7. The User shall click the address Center Hall, Library Walk, San Diego, CA, United States.
8. The System shall write out the full address in the Quest Location text field.
9. The User shall click the Start Date setting.
10. The System shall prompt a date picker widget.
11. The User shall set the start date to the current date.
12. The User shall click Set/OK.
13. The User shall click the Start Time setting.
14. The System shall prompt a time picker widget.
15. The User shall set the start time to current time + 52 minutes.
16. The User shall click Set/OK
17. The User shall click the End Date setting.
18. The System shall prompt a date picker widget.
19. The User shall set the end date to current date.
20. The User shall click Set/OK.
21. The User shall click the End Time setting.
22. The System shall prompt a time picker widget.
23. The User shall set the end time to current time + 57 minutes.
24. The User shall click Set/OK
25. The User shall enter “5” in the Estimated Travel Time text field.
26. The User shall click the Travel Method setting.
27. The System shall display a list of travel method options.
28. The User shall click Bike
29. The User shall enter “Yay Notifications!” in the Special Notes text field.
30. The User shall verify toggle button is set to ON for the question Would you like to be notified?
31. The User shall click SAVE QUEST.
32. The System shall do its validation checks to make sure the Quest can be added and saves this information into the Parse database.
33. The System shall redirect the User to the Home screen where the newly added Quest is displayed along with other existing Quests.
34. The System shall send the User a notification within the next 6 minutes to remind them of the upcoming Quest titled “Test Notif Quest”.

Result: This test passes when the User has received a push notification that indicates there is an upcoming Quest titled “Test Notif Quest”.

Adapting Notification Times

Title: Adapting Notification Times **[TCNS2]**
Description: Every time the user is late to a Quest, the system shall send future notifications one minute earlier. As the user becomes more punctual, the notifications move towards their original scheduled time. The number of notifications sent depends on the user's answer to sign up quiz question three.
User Goals: To have a smart notification system that calculates the best time/number of notifications to be sent to notify the user of future Quests.
Desired Outcome: The System shall be able to calculate the right time to send notifications to the user depending on how late/early they arrive to their past quests.
Details:

- **Priority Level**: 2
- **Status**: implemented & tested ✓

Actors: None
Requirements:

1. Average Tardiness [RNS-1]
2. Get Travel Time from Google Maps API [RNS-5]
3. Update Quest Notifications [RNS-6]

Preconditions:

1. User is logged in with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User is at the Home screen.
4. User has global Enable All Notifications setting set to ON
5. User has global Number of Notifications setting set to 4.
6. User has global Preparation Time setting set to 30.
7. User has global Lateness Offset setting set to 15.

Trigger: User has navigated to the Home screen
Workflow:

1. The User shall click (+) on the upper right corner from the Home screen.
2. The System shall bring the user to the Add Quest screen.
3. The User shall enter “Test Adapting Notif Quest” for the Quest Title.
4. The User shall leave the Start Location setting as Home.

5. The User shall enter their current location for the Quest Location text field.
6. The User shall click the Start Date setting.
7. The System shall prompt a date picker widget.
8. The User shall set the start date to the current date.
9. The User shall click Set/OK.
10. The User shall click the Start Time setting.
11. The System shall prompt a time picker widget.
12. The User shall set the start time to current time + 4 minutes.
13. The User shall click Set/OK
14. The User shall click the End Date setting.
15. The System shall prompt a date picker widget.
16. The User shall set the end date to current date.
17. The User shall click Set/OK.
18. The User shall click the End Time setting.
19. The System shall prompt a time picker widget.
20. The User shall set the end time to current time + 8 minutes.
21. The User shall click Set/OK
22. The User shall enter “1” in the Estimated Travel Time text field.
23. The User shall leave the Travel Method setting as Walk.
24. The User shall leave the Special Notes text field blank.
25. The User shall click the toggle button for the question Would you like to be notified? to turn this setting off.
26. The System shall set the toggle button for this option to OFF.
27. The User shall click SAVE QUEST.
28. The System shall do its validation checks to make sure the Quest can be added and saves this information into the Parse database.
29. The System shall redirect the User to the Home screen where the newly updated Quest is displayed along with other existing Quests.
30. The System shall reveal an Arrival Button in the Home screen.
31. The System shall display a countdown on the Arrival Button.
32. The System shall display BOOM! once the countdown gets to 0.
33. The User shall click the Arrival Button.
34. The System shall no longer display the Arrival Button after the Quest start time.
35. The System shall log that the User has arrived late to the Quest.

36. The System shall automatically increase the User’s Lateness Offset setting.

Result: This test passes when the System no longer displays the Arrival Button after being clicked and accordingly updates the User’s Lateness Offset for being late to the Quest.

Alternate Path Workflow:

1. Execute Workflow steps 1-31.
2. The User shall click the Arrival Button any time before the Quest start time to indicate the User has arrived to the Quest Location.
3. The System shall log that the User has arrived early to the Quest.
4. The System shall automatically decrease the User’s Lateness Offset setting.

Result: This alternate test passes when the System no longer displays the Arrival Button after being clicked and accordingly updates the User’s Lateness Offset for being early to the Quest.

Obtain Travel Time from Google Directions

Title: Obtain Travel Time from Google Maps **[TCNS3]**
Description: The System shall be able to connect to Google Maps via the Google Maps API to determine the travel time to the upcoming quest. The System takes into consideration the means of travel that the user has selected in that particular quest.
User Goals: The user wants to application to check the travel time from start location to quest location taking into account the method of travel that they specified in the quest.
Desired Outcome: The system shall be able to check with Google Directions API the travel time the next quest in order to employ a smarter notification system in case the user wasn't sure how long will the estimated travel time be.
Details:

- **Priority Level**: 4
- **Status**: implemented & tested ✓

Actors: None
Requirements:

1. Get Travel Time from Google Maps API [RNS-5]
2. Update Quest Notifications [RNS-6]

Preconditions:

1. User is logged into the app with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User is at the Home screen.
4. User has global Enable All Notifications setting set to ON
5. User has global Number of Notifications setting set to 4
6. User has global Preparation Time setting set to 30.
7. User has global Lateness Offset setting set to 15.

Trigger: Cloud schedules notifications to be pushed at the calculated time.
Failed Conclusions:

- None

Workflow:

1. The User, awsmtest, shall click the (+) button on the top right corner.
2. The System shall bring the user to the Add Quest screen.

3. The User shall input “Test Google Time Quest” in the Quest Title text field.
4. The User shall leave the Start Location option as Home.
5. The User shall enter “Geisel Library” in the Quest Location text field.
6. The System shall display a list with autocompleted addresses.
7. The User shall click the address Geisel Library, San Diego, CA, United States.
8. The System shall write out the full address in the Quest Location text field.
9. The User shall click the Start Date setting.
10. The System shall prompt a date picker widget.
11. The User shall set the start date to the current date.
12. The User shall click Set/OK.
13. The User shall click the Start Time setting.
14. The System shall prompt a time picker widget.
15. The User shall set the start time to current time + 52 minutes.
16. The User shall click Set/OK
17. The User shall click the End Date setting.
18. The System shall prompt a date picker widget.
19. The User shall set the end date to current date.
20. The User shall click Set/OK.
21. The User shall click the End Time setting.
22. The System shall prompt a time picker widget.
23. The User shall set the end time to current time + 57 minutes.
24. The User shall click Set/OK
25. The User shall enter “40” in the Estimated Travel Time text field.
26. The User shall click the Travel Method setting.
27. The System shall display a list of travel method options.
28. The User shall click Bike
29. The User shall enter “Yay Notifications!” in the Special Notes text field.
30. The User shall verify toggle button is set to ON for the question Would you like to be notified?
31. The User shall click SAVE QUEST.
32. The System shall do its validation checks to make sure the Quest can be added and saves this information into the Parse database.

33. The System shall redirect the User to the Home screen where the newly added Quest is displayed along with other existing Quests.
34. The System shall send the User a notification within the next 6 minutes to remind them of the upcoming Quest titled “Test Google Time Quest”.

Result: This test passes when the User has received a push notification that indicates there is an upcoming Quest titled “Test Google Time Quest”.

Add Quests to Calendar

Title: Add Quests to Calendar **[TCCS1]**
Description: Users shall be able to manually create quests. Each quest should have a title, start time, end time, and destination. The quest would have a travel method and a start location set to the default values obtained from quiz questions 4 and 5, respectively. The user could also toggle (on and off) the notifications for the quest.
User Goals: To have all their quests information in one place.
Desired Outcome: The System shall securely store all the quests of each user along with the specific information of each quest.
Details:

- **Priority Level**: 1
- **Status**: implemented & tested ✓

Actors:

1. User

Requirements:

1. Add Quest [RCS-1]
2. Save Quest Information [RCS-2]
3. Update Quest Notifications [RNS-6]
4. Per-Quest Notifications [RNS-7]

Preconditions:

1. User has logged in with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User is at the Add a New Quest screen.

Trigger: User clicks the (+) button to add a new Quest.
Failed Conclusions

1. User leaves the Quest Title setting empty.
2. User leaves the Estimated Travel Time setting empty.
3. User leaves the Quest Location setting empty.

Workflow:

1. The User, awsmtest, shall input “Test Add Quest” in the Quest Title text field.
2. The User shall leave the Start Location option as Home.
3. The User shall enter “Center Hall” in the Quest Location text field.

4. The System shall display a list with autocompleted addresses.
5. The User shall click the address Center Hall, Library Walk, San Diego, CA, United States.
6. The System shall write out the full address in the Quest Location text field.
7. The User shall click the Start Date setting.
8. The System shall prompt a date picker widget.
9. The User shall set the start date to the “July 15 2015”.
10. The User shall click Set/OK.
11. The User shall click the Start Time setting.
12. The System shall prompt a time picker widget.
13. The User shall set the start time to “3:30 PM”.
14. The User shall click Set/OK
15. The User shall click the End Date setting.
16. The System shall prompt a date picker widget.
17. The User shall set the end date to “July 15 2015”.
18. The User shall click Set/OK.
19. The User shall click the End Time setting.
20. The System shall prompt a time picker widget.
21. The User shall set the end time to “4:30 PM”.
22. The User shall click Set/OK
23. The User shall enter “5” in the Estimated Travel Time text field.
24. The User shall click the Travel Method setting.
25. The System shall display a list of travel method options.
26. The User shall click Bus
27. The User shall enter “You are Awesome!” in the Special Notes text field.
28. The User shall verify toggle button is set to ON for the question Would you like to be notified?
29. The User shall click SAVE QUEST.
30. The System shall do its validation checks to make sure the Quest can be added and saves this information into the Parse database.
31. The System shall redirect the User to the Home screen where the newly added Quest is displayed along with other existing Quests.

Result: This test passes when the “Test Add Quest” is displayed in the Home screen.

Alternate Path 1 Workflow [Failed Conclusion 1]

1. The User shall leave the text field for the Quest Title empty.
2. The User shall click SAVE QUEST.
3. The System shall check that the Quest Title is empty.
4. The System shall display a red (!) icon in the text field to notify the User, “Your Quest must have a title!”

Result: This alternate test passes when the System displays the error message described in Alt 1 Path Step 4.

Alternate Path 2 Workflow [Failed Conclusion 2]

1. Execute Workflow steps 1-6.
2. The User shall leave the text field for the Estimated Travel Time empty
3. The User shall click SAVE QUEST
4. The System shall check that the Estimated Travel Time text field is empty.
5. The System shall display a red (!) icon in the text field to notify the User, “Your Travel Time must be greater than or equal to 0!”

Result: This alternate test passes when the System displays the error message described in Alt 2 Path Step 5.

Alternate Path 3 Workflow [Failed Conclusion 3]

1. The User, awsmtest, shall input “Test2 Add Quest” in the Quest Title text field.
2. The User shall leave the Start Location option as Home.
3. The User shall leave the text field for the Quest Location empty.
4. Execute Workflow steps 7-29.
5. The System shall check that the Quest Location text field is empty.
6. The System shall display to the User an “Invalid Location” error message.
7. The User shall click OK.

8. The System shall do its validation checks to make sure the Quest can be added and saves this information into the Parse database.
9. The System shall redirect the User to the Home screen where the newly added Quest is displayed along with other existing Quests.

Result: This test passes when the “Test2 Add Quest” is displayed in the Home screen.

Edit Quests

Title: Edit Quests **[TCCS2]**

Description: The System should allow the user to edit the information they provided in any existing quest. A user can change any field they've previously filled, and add to any field that they've previously left blank.

User Goals: To correct the information of an existing quest and make them more accurate

Desired Outcome: Allow the user to update each of their existing quests with the most up-to-date information to help the system use this information to calculate the most efficient time for notifications.

Details:

- **Priority Level**: 1
- **Status**: implemented & tested ✓

Actors

1. User

Requirements

1. Edit Quest [RCS-3]
2. Update Quest Information [RCS-4]
3. Update Quest Notifications [RNS-4]

Preconditions

1. User is logged in with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User has an existing Quest titled "Test Edit Quest" that has yet to happen.
4. User is in the Edit Quest screen for the existing Quest "Test Edit Quest".

Trigger: User has clicked EDIT from expanded Quest description.

Failed Conclusions

1. User leaves the Quest Title setting empty.
2. User leaves the Estimated Travel Time setting empty.

Workflow

1. The User shall clear the text field for the Quest Title.
2. The User shall input "Testing Edit Quest" in the Quest Title text field.
3. The User shall leave the Start Location option as Home.
4. The User shall clear the text field for the Quest Location.

5. The User shall enter “Warren Lecture Hall” in the Quest Location text field.
6. The System shall display a list with autocompleted addresses.
7. The User shall click the address Warren Lecture Hall, La Jolla, CA, United States
8. The System shall write out the full address in the Quest Location text field.
9. The User shall click the Start Date setting.
10. The System shall prompt a date picker widget.
11. The User shall set the start date to the “July 16 2015”.
12. The User shall click Set/OK.
13. The User shall click the Start Time setting.
14. The System shall prompt a time picker widget.
15. The User shall set the start time to “4:30 PM”.
16. The User shall click Set/OK
17. The User shall click the End Date setting.
18. The System shall prompt a date picker widget.
19. The User shall set the end date to “July 16 2015”.
20. The User shall click Set/OK.
21. The User shall click the End Time setting.
22. The System shall prompt a time picker widget.
23. The User shall set the end time to “5:30 PM”.
24. The User shall click Set/OK
25. The User shall clear the text field for the Estimated Travel Time setting.
26. The User shall enter “10” in the Estimated Travel Time text field.
27. The User shall click the Travel Method setting.
28. The System shall display a list of travel method options.
29. The User shall click Car
30. The User shall enter “CSE 110 is Awesome!” in the Special Notes text field.
31. The User shall click the toggle button for the question Would you like to be notified? to turn off notifications.
32. The System shall change the toggle button to OFF.
33. The User shall click SAVE QUEST.

34. The System shall do its validation checks to make sure the Quest can be edited and saves this information into the Parse database.
35. The System shall redirect the User to the Home screen where the newly updated Quest is displayed along with other existing Quests.

Result: This test passes when the newly updated Quest, “Testing Edit Quest” is displayed in the Home screen.

Alternate Path 1 Workflow [Failed Conclusion 1]

1. The User shall clear the text field for the Quest Title.
2. The User shall click SAVE QUEST
3. The System shall check that the Quest Title is empty.
4. The System shall display a red (!) icon in the text field to notify the User, “Your Quest must have a title!”

Result: This alternate test passes when the System displays the error message described in Alt 1 Path Step 4.

Alternate Path 2 Workflow [Failed Conclusion 2]

1. The User shall clear the text field for the “Estimated Travel Time”.
2. The User shall click SAVE QUEST.
3. The System shall check that the “Estimated Travel Time” text field is empty.
4. The System shall display a red (!) icon in the text field to notify the User, “Your Travel Time must be greater than or equal to 0!”

Result: This alternate test passes when the System displays the error message described in Alt 2 Path Step 4.

Alternate Path 3 Workflow [Failed Conclusion 3]

1. The User shall clear the text field for the Quest Location.
2. The User shall click SAVE QUEST.
3. The System shall check that the Quest Location text field is empty.
4. The System shall display to the User an “Invalid Location” error message.
5. The User shall click OK.

6. The System shall do its validation checks to make sure the Quest can be added and saves this information into the Parse database.
7. The System shall redirect the User to the Home screen where the newly added Quest is displayed along with other existing Quests.

Result: This test passes when the “Test Edit Quest” is displayed in the Home screen.

Delete Quests

Title: Delete Quests **[TCCS3]**
Description: The system shall allow the user to delete individual quests from their calendar. When a user deletes a quest, the system will delete it from the database and the user will not receive notifications for it anymore.
User Goals: Delete a canceled quest from their calendar to keep their calendar as up-to-date as possible.
Desired Outcome: Remove the quest from the user's calendar of quests and from the database and to cancel the scheduled notifications for that quest.
Details:

- **Priority Level**: 1
- **Status**: implemented & tested ✓

Actors

1. User

Requirements

1. Delete Quest [RCS-5]
2. Unschedule notifications [RNS-8]

Preconditions

1. User is logged in with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User has an existing Quest titled "Test Delete Quest" that has yet to happen.
4. User is in the Edit Quest screen for the existing Quest "Test Delete Quest".

Trigger: User has clicked EDIT from expanded Quest description.
Failed Conclusion

- None

Workflow

1. The User, awsmtest, shall click (🗑) from the upper right corner.
2. The System shall display to the User a pop question that says "Are you sure you want to delete this Quest?"
3. The User shall click OK.

4. The System shall remove the Quest from existence by deleting it from the Parse database.
5. The System shall redirect the User to the Home screen where the Quest titled “Test Delete Quest” is not displayed anymore.

Result: This test passes when the System does not display the Quest “Test Delete Quest” in the Home screen.

Add Special Notes

Title: Add Special Notes **[TCCS4]**

Description: The System shall give users shall the opportunity to add a special note to each quest. For example, a user may want to include a note to remind them to bring something with them (ex. bring iClicker), or perhaps include some special instructions to arrive to their destination (open gate with code 3123). The user can add a special note to a quest while creating the quest or add it later by editing an existing quest. Those special notes would be also be included in all the notifications that are pushed to the user's screen in regards of this specific quest.

User Goals: To add personal notes to each quest that are special to that specific quest and to include those notes in all notifications for that quest.

Desired Outcome: Store an optional special note with each quest and edit the text of that quest's notification to include the note.

Details:

- **Priority Level**: 3
- **Status**: implemented & tested ✓

Actors:

1. User (Primary)

Requirements:

1. Update Quest Information [RCS-4]
2. Update Quest Notifications [RNS-4]
3. Per-Quest Notifications [RNS-7]
4. Edit Notifications' Text [RNS-9]

Preconditions:

1. User is logged in with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User is at the Add Quest screen.

Trigger: User has clicked (+) on the upper right corner from the Home screen.

Failed Conclusions:

1. User leaves the special notes field blank.

Workflow:

1. The User, awsmtest, shall input "Test Special Notes Quest" in the Quest Title text field.

2. The User shall leave the Start Location option as Home.
3. The User shall enter “Center Hall” in the Quest Location text field.
4. The System shall display a list with autocompleted addresses.
5. The User shall click the address Center Hall, Library Walk, San Diego, CA, United States.
6. The System shall write out the full address in the Quest Location text field.
7. The User shall click the Start Date setting.
8. The System shall prompt a date picker widget.
9. The User shall set the start date to the “July 15 2015”.
10. The User shall click Set/OK.
11. The User shall click the Start Time setting.
12. The System shall prompt a time picker widget.
13. The User shall set the start time to “3:30 PM”.
14. The User shall click Set/OK
15. The User shall click the End Date setting.
16. The System shall prompt a date picker widget.
17. The User shall set the end date to “July 15 2015”.
18. The User shall click Set/OK.
19. The User shall click the End Time setting.
20. The System shall prompt a time picker widget.
21. The User shall set the end time to “4:30 PM”.
22. The User shall click Set/OK
23. The User shall enter “5” in the Estimated Travel Time text field.
24. The User shall click the Travel Method setting.
25. The System shall display a list of travel method options.
26. The User shall click Bus
27. The User shall enter “Wow Team AWSM is so Awesome!” in the Special Notes text field.
28. The User shall verify toggle button is set to ON for the question Would you like to be notified?
29. The User shall click SAVE QUEST.
30. The System shall do its validation checks to make sure the Quest can be added and saves this information into the Parse database.
31. The System shall redirect the User to the Home screen where the newly added Quest is displayed along with other existing Quests.

32. The User shall click the Quest titled “Test Special Notes Quest”.
33. The System shall expand the Quest to display its full description including the special notes.

Result: This test passes when the System displays the full Quest description with the special notes indicated in Workflow Step 27.

Quest Archive

Title: Quest Archive **[TCCS5]**
Description: The user shall be able to view past quests if they need to. All past quests are included in a “Quest Archive” screen that is accessible form the navigation menu. Those quests are sorted from most recent to oldest. The user can’t edit past quests.
User Goals: To have a way to view past quests in case the user wanted to remember any detail about those quests.
Desired Outcome: The system shall keep past quests saved in the database and sorted in a designated page called the “Quest Archive” page in case the user wanted to check them out in the future.
Details:

- **Priority Level**: 3
- **Status**: implemented & tested ✓

Actors None
Requirements

1. Keep Past Quests [RCS-6]
2. Sort Quests Base on Time [RCS-7]
3. Make Past Quests not Editable [RCS-8]

Preconditions

1. User is logged in with username awsmtest and password awsmtest
2. [TCBF3] Preconditions apply.
3. User has completed Quest titled “Past Quest”.
4. User is in Quest Archive screen.

Trigger: After opening the menu bar, the User clicks Quest Archive
Failed Conclusions

- None

Workflow

1. The User, awsmtest, shall click the Quest titled “Past Quest”.
2. The System shall expand the Quest out displaying the full description of this past Quest.

Result: This test passes when the System displays the full details of the past Quest titled “Past Quest”.

Show Weather Recommendations

Title: Show Weather Recommendations **[TCCS6]**
Description: The next upcoming quest on the home screen will include a small weather icon specifying the expected weather conditions at that specific quest. The weather icon could be any of (sun, rain, clouds, snow) along with the temperature in degrees Fahrenheit.
User Goals: The User want to be aware of any unusual weather conditions before leaving to next quest. This would allow them to dress appropriately for the day.
Desired Outcome: For the top quest on the home screen, always include the weather icon indicating the weather condition at the next quest.
Details:

- **Priority Level**: 4
- **Status**: implemented & tested ✓

Actors: None
Requirements:

1. Get Weather Forecast [RCS-9]

Preconditions:

1. User is logged in with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User has Quest titled “Test Weather Quest”.
4. “Test Weather Quest” is the next upcoming event.
5. User is at the Home screen.

Trigger: User has clicked the Quest titled “Test Weather Quest” to expand its full description.
Failed Conclusions:

- None

Workflow:

1. The User shall see 2 icons above the bell icon indicating the forecasted weather and temperature respectively.

Result: This test passes when the System has displayed the appropriate forecasted weather icon in the Quest description.

Toggle Quest Notifications

Title: Toggle Quest Notifications **[TCCS7]**
Description: The System shall give the user a way to indicate if they want to be notified about a certain quest or not. The user can always add quests to their calendar but choose not to be notified about them. This would allow the user to keep their quests in one place and not limit them to adding the ones they would like to be notified about.
User Goals: The user wants to have the ability to choose which quests they want to be notified about and which ones not.
Desired Outcome: To include a notification toggle option for each quest the user can toggle it on or off. It will be set to on by default but the user can always change it per quest.
Details:

- **Priority Level**: 2
- **Status**: implemented & tested ✓

Actors:

1. User

Requirements:

1. Save Quest information [RCS-2]
2. Update Quest [RCS-4]
3. Per-Quest Notifications [RNS-7]

Preconditions:

1. User is logged in with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User has an existing quest titled “Test Toggle Notif Quest” with notifications currently turned on.
4. User is in the Edit Quest screen.

Trigger: User has clicked EDIT from the expanded Quest description for “Test Toggle Notif Quest”.
Failed Conclusions:

- None

Workflow:

1. The User, awsmtest, shall click the toggle button for the question Would you like to be notified? to turn the option off.
2. The System shall display the setting as OFF.

3. The User shall click SAVE QUEST.
4. The System shall save this setting change to the database.
5. The System shall bring the User to the Home screen.
6. The System shall display a strikethrough for the bell icon for this specific quest indicating that notifications have been turned off.

Result: This test passes when the System displays the bell icon with a strikethrough for the Quest titled “Test Toggle Notif Quest” indicating that notifications for that specific Quest have been turned off.

Click Button Upon Arrival

Title: Click Button Upon Arrival **[TCGA1]**

Description: After sending the first notification to the user in regards of a specific quest, an arrival button would become live on the home screen. The arrival button will include a countdown timer that counts the minutes and seconds till the start time of the quest. When the user arrives to their quest, they press on the arrival button to indicate that they've arrived early. The button will then disappear. If they user doesn't arrive early to their quest, the button will stay live until the start time of the quest and then it will automatically disappear.

User Goals: The user wants an easy way to indicate that they've arrived. If they also want to be informed of how much time is left before their quest begins. The arrival button serves both purposes.

Desired Outcome: Give the user an easy and convenient way to be informed of how much time is left before their next quest and to indicate their arrival once they arrive early. By knowing if the user arrives early or late, the system will be able to adjust the Average Tardiness index in order, to adapt and use it when calculating notification times of future quests.

Details:

- **Priority Level**: 1
- **Status**: implemented & tested ✓

Actors:

1. User

Requirements:

1. Arrival Button [RGA-1]
2. Update Average Tardiness Upon Arrival [RGA-2]

Preconditions:

1. User is logged in with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User is at the Home screen
4. User has global Number of Notifications setting set to 4.
5. User has global Preparation Time setting set to 30.
6. User has global Lateness Offset setting set to 15.

Trigger: The User has navigated to the Home screen.

Failed Conclusions:

- None

Workflow:

37. The User shall click (✚) on the upper right corner from the Home screen.
38. The System shall bring the user to the Add Quest screen.
39. The User shall enter “Test Arrival Button Quest” for the Quest Title.
40. The User shall leave the Start Location setting as Home.
41. The User shall enter their current location for the Quest Location text field.
42. The User shall click the Start Date setting.
43. The System shall prompt a date picker widget.
44. The User shall set the start date to the current date.
45. The User shall click Set/OK.
46. The User shall click the Start Time setting.
47. The System shall prompt a time picker widget.
48. The User shall set the start time to current time + 5 minutes.
49. The User shall click Set/OK
50. The User shall click the End Date setting.
51. The System shall prompt a date picker widget.
52. The User shall set the end date to current date.
53. The User shall click Set/OK.
54. The User shall click the End Time setting.
55. The System shall prompt a time picker widget.
56. The User shall set the end time to current time + 10 minutes.
57. The User shall click Set/OK
58. The User shall enter “5” in the Estimated Travel Time text field.
59. The User shall leave the Travel Method setting as Walk.
60. The User shall click the toggle button for the question Would you like to be notified? to turn off notifications.
61. The System shall change the toggle button to OFF.
62. The User shall click SAVE QUEST.
63. The System shall do its validation checks to make sure the Quest can be updated and saves this information into the Parse database.
64. The System shall redirect the User to the Home screen where the newly created Quest is displayed along with other existing Quests.
65. The System shall reveal an Arrival Button in the Home screen.

66. The User shall click the button before the Quest start time indicating that the User has arrived to the Quest location.
67. The System shall make the button disappear and log the time the User has arrived.
68. The User shall click the (≡) icon.
69. The System shall make the menu bar slide over and display the various menu bar options.
70. The User shall click Home.
71. The System shall refresh the Home screen.

Result: This test passes when the System no longer displays the button after the User has clicked it and refreshed the Home screen.

View User Score of Points

Title: View User Score of Points **[TCGA2]**
Description: The User shall have a score based on an accumulation of points. The points increase every time the user arrives early to their quest.
User Goals: Users will be happy to see their achievements and their early habits paying off in the form of points.
Desired Outcome: To show the users their accumulated points in order to motivate them to be more punctual and to use the application more.
Details:

- **Priority Level**: 1
- **Status**: implemented & tested ✓

Actors:

1. User

Requirements:

1. View User Score [RGA-4]
2. Update User Score [RGA-5]

Preconditions:

1. User is logged in with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User is at Home screen.

Trigger: User has navigated to the Home screen.
Failed Conclusions:

- None

Workflow:

1. The User shall click the menu bar icon (☰).
2. The System shall make the menu bar slide over and display the various menu bar options.
3. The User shall click My Planet from the menu bar.
4. The System shall bring the User to the My Planet screen and the menu bar shall slide to the left to become hidden.
5. The System shall display the User’s current total of points in above the User’s current Punctual Pal.

Result: This test passes when the System displays the current total of points at the My Planet screen.

Alternate Path Workflow:

1. The User shall be at the Home screen.
2. The System shall display the User's current total of points in above the User's current Punctual Pal.

Result: This test passes when the System displays the current total of points at the Home screen.

View Current Pal

Title: View Current Pal **[TCGA3]**

Description: Each user has a special avatar (Pal) that they keep updating as they use the application more and more. Every time they arrive early they're allowed to add a feature to their current Pal. The user can add up to 10 features to a current Pal until it's done. Once it's done, it will move gallery. The system shall give the user a way to view their current Pal by navigating to the "Gallery" screen.

User Goals: The User shall be able to view their current Pal and be motivated to complete it by using the app consistently or more

Desired Outcome: Give the user a dedicated screen where they could view their current incomplete Pal with all its current features.

Details:

- **Priority Level**: 1
- **Status**: implemented & tested ✓

Actors:

1. User

Requirements:

1. My Planet Screen [RGA-7]
2. Current Pal's features [RGA-8]

Preconditions:

1. User is logged in with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User is at Home screen.
4. User has set the "Choose which Punctual Pal to display" setting to "Current Working Pal". [Alt Path]

Trigger: User has navigated to the Home screen.

Failed Conclusions:

- None

Workflow:

1. The User shall click the menu bar icon (☰).
2. The System shall make the menu bar slide over and display the various menu bar options.
3. The User shall click My Planet from the menu bar.

4. The System shall bring the User to the My Planet screen and the menu bar shall slide to the left to become hidden.
5. The System shall display the User's current Punctual Pal.

Result: This test passes when the System displays the User's current Punctual Pal at the My Planet screen.

Alternate Path Workflow:

1. The User shall be at the Home screen.
2. The System shall display the User's current Punctual Pal.

Result: This test passes when the System displays the User's current Punctual Pal at the Home screen.

Select Next Feature for Pal

Title: Select Next Feature for Pal **[TCGA4]**
Description: The System shall give the user a way to add new features to their current Pal each time they arrive early to their quests. When the user arrives early to a quest and presses on the arrival button, the system will indicate that they can add new features to their Pal by displaying an “ ! ” button to their home screen. If they click on it, they would be taken to the “Add Feature” page to select a new item for their Pal.
User Goals: The User shall be able to select items to build up and personalize their pals in a way that suits their personality.
Desired Outcome: Reward the user for arriving early to their quests with an opportunity to add new features to their current Pals.
Details:

- **Priority Level**: 2
- **Status**: implemented & tested ✓

Actors:

1. User

Requirements:

1. Update Pal’s PNG File [RGA-6]

Preconditions:

1. User is logged in with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. (!) icon in the User’s My Planet screen is displayed to indicate that a new feature is ready to be added to their Punctual Pal.
4. (!) icon in the User’s Home screen is displayed to indicate that a new feature is ready to be added to their Punctual Pal. [Alt Path]
5. User is at the Add Item screen.

Trigger:

1. The User has clicked the (!) icon from the My Planet screen.
2. The User has clicked the (!) icon from the Home screen. [Alt Path]

Failed Conclusions:

- None

Workflow:

1. The User, awsmtest, shall choose the first option for their Punctual Pal’s new feature.

2. The System shall highlight the first option indicating that it is currently chosen.
3. The System shall update the Punctual Pal to preview the item on it.
4. The User shall click I CHOOSE YOU
5. The System will update the Punctual Pal to have the new item.
6. The System will redirect the User back to the My Planet screen.
7. The System shall no longer display the (!) icon.
8. The System shall display the newly updated Punctual Pal.

Result: This test passes when the System displays the newly updated Punctual Pal with the new item.

Alternate Path Workflow:

1. Execute Workflow 1-8.

Result: This alternate test passes when the System displays the newly updated Punctual Pal with the new item.

View Collected Complete Pals

Title: View Collected Complete Avatars **[TCGA5]**

Description: The User shall be able to view a gallery of the completed Pals. This page will be called the “Gallery” screen. This screen will remain empty until the user has arrived early to enough quests to add all 10 features to a Pal and has one complete Pal. This completed Pal will appear in the gallery and the User will start with a fresh new Pal to build up in future quests.

User Goals: The User wants to view all their beautiful completed pals.

Desired Outcome: The System shall give the user a way to view all their previous pals in one location which is the “Gallery” screen.

Details:

- **Priority Level**: 2
- **Status**: implemented & tested ✓

Actors:

1. User

Requirements:

1. Gallery Screen [RGA-9]
2. Store All Previous Pals [RGA-10]

Preconditions:

1. User is logged in with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User is at Home screen or any screen that has access to the menu bar.
4. User is at My Planet screen. [Alt Path]

Trigger:

1. User has navigated to the Home screen.
2. User has navigated to the My Planet screen. [Alt Path]

Failed Conclusions:

- None

Workflow:

1. The User shall click the menu bar icon (☰).
2. The System shall make the menu bar slide over and display the various menu bar options.
3. The User shall click Gallery from the menu bar.

4. The System shall bring the User to the Gallery screen and the menu bar shall slide to the left to become hidden.
5. The System shall display the User’s completed Punctual Pals.

Result: This test passes when the System displays the User’s collection of Punctual Pals.

Alternate Path Workflow:

1. The User shall click the (🏛) icon.
2. The System shall bring the User to the Gallery screen.
3. The System shall display the User’s completed Punctual Pals.

Result: This test passes when the System displays the User’s collection of Punctual Pals.

View Achievements

Title: View Achievements **[TCGA6]**

Description: The User shall receive achievements for different reasons that show improved punctuality (i.e. being early 5 times in a row, reaching a certain number of points, etc.).

User Goals: The User will want to fulfill the different reasons for getting achievements, which will effectively improve their punctuality.

Desired Outcome: The User will feel accomplished by receiving achievements, which will encourage them to continue to use the app.

Details:

- **Priority Level**: 3
- **Status**: implemented & tested ✓

Actors:

1. User

Requirements

1. Locked Achievements [RGA-11]
2. Unlocked Achievements [RGA-12]

Preconditions:

1. User is logged in with username awsmtest and password awsmtest.
2. [TCBF3] Preconditions apply.
3. User is at Home screen or any screen that has access to the menu bar.
4. User is at My Planet screen. [Alt Path]

Trigger:

1. User has navigated to the Home screen.
2. User has navigated to the My Planet screen. [Alt Path]

Failed Conclusions:

- None

Workflow:

1. The User shall click the menu bar icon (☰).
2. The System shall make the menu bar slide over and display the various menu bar options.
3. The User shall click Awards from the menu bar.
4. The System shall bring the User to the Awards screen and the menu bar shall slide to the left to become hidden.

5. The System shall display the User’s collection of awards.

Result: This test passes when the System displays the User’s collection of awards.

Alternate Path Workflow:

1. The User shall click the (🎖) icon.
2. The System shall bring the User to the Awards screen.
3. The System shall display the User’s collection of awards.

Result: This test passes when the System displays the User’s collection of awards.