College Capital Manual Testing Documentation (Sprint 2)

(Note: User should begin testing logged out of any current College Capital accounts)
(Note: Unless otherwise specified, User should always be logged in during testing)

User Story 9: As a user, I would like to be able to visualize my current financial status.

Test Case 1

- System: chart.js
- Check to make sure pie chart displays correct categorical data
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Visualizations page

- **Expected Result**
  - Users current financial status should be visualized with correct data in a pie chart

Test Case 2

- System: chart.js
- Check to make sure daily expenditures are correctly visualized
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Visualizations page

- **Expected Result**
  - User’s daily expenditures should be correctly visualized

Test Case 3

- System: chart.js
- Check to make sure weekly expenditures are correctly visualized
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Visualizations page
- **Expected Result**
  - User’s weekly expenditures should be correctly visualized

Test Case 4

- System: chart.js
- Check to make sure monthly expenditures are correctly visualized
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Visualizations page

- **Expected Result**
  - User's monthly expenditures should be correctly visualized

User Story 13: As a user, I would like to be able to update my transaction history.

Test Case 1

- System: Transactions.js, CreateEditTransaction.js
- Check to make sure input validation functions correctly
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Transactions page
  - Attempt to create a new expenditure by pressing the NEW TRANSACTION button

- **Expected Result**
  - User should not be able to create a transaction without filling in all fields

Test Case 2

- System: Transactions.js, CreateEditTransaction.js
- Check to make sure user can update their transaction history
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Transactions page
  - Update a present transaction with valid information

- **Expected Result**
  - Updated transaction history should be correctly displayed

Test Case 3

- System: Transactions.js
- Check to make sure user's transaction history is correctly displayed
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Transactions page

- **Expected Result**
  - User's transaction history should be displayed with the correct information

Test Case 4

- System: Transactions.js, CreateEditTransaction.js
- Check to make sure each newly added transaction is listed correctly
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Transactions page
  - Create a new transaction and navigate back to Transactions page

- **Expected Result**
  - Newly made transaction should be listed in the correct order

Test Case 5

- System: Transactions.js, CreateEditTransaction.js
- Check to make sure changes on the front-end are reflected in the database
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Transactions page
  - Update a transaction
  - (As admin) Navigate to College Capital firebase console

- **Expected Result**
  - Transaction values updated in the UI should also be updated in the database

Test Case 6

- System: Transactions.js
- Check to make sure table display functions are functioning correctly
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Transactions page
    - Add a few transactions with differing values
  - Test various table functions (sorting, highlighting, editing)

- **Expected Result**
  - All table functions should operate as expected

User Story 14: As a user, I would like to be able to view my transaction history.

Test Case 1

- System: Transaction.js
- Check to make sure users transaction history is correctly ordered
- Severity: 1

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Transactions page

- **Expected Result**
  - Users transaction history should be displayed in order of transaction date with the correct values

Test Case 2

- System: Transactions.js
- Check to make sure users transaction history is correctly displayed
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Transactions page

- **Expected Result**
  - User's transactions should contain all relevant information (date, amount, and location)

Test Case 3

- System: Transactions.js
- Check to make sure users are given the option to delete transactions
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Transactions.js page

- **Expected Result**
  - Transactions should also give an option to be deleted from the list, should the user decide to do so

Test Case 4

- System: Transactions.js
- Check to make sure transaction history can be reverse sorted
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Transactions page
  - Click on Transaction Date once to reverse sorting order

- **Expected Result**
  - Transactions should be sorted in reverse order, by oldest transaction instead of newest.

User Story 15: I would like to be able to export my transaction history.

Test Case 1

- System: chart.js
- Check to make sure user can export pie chart
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Visualizations page
  - Select file format from drop down menu

- **Expected Result**
  - Page should correctly export pie chart in chosen format

Test Case 2

- System: chart.js
- Check to make sure user can export daily spending chart
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Visualizations page
  - Select file format from drop down menu on daily spending graph

- **Expected Result**
  - Page should correctly export daily spending graph in chosen format

Test Case 3

- System: chart.js
- Check to make sure user can export weekly spending chart
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Visualizations page
  - Select file format from drop down menu on weekly spending graph

- **Expected Result**
  - Page should correctly export weekly spending graph in chosen format

Test Case 4

- System: chart.js
- Check to make sure user can export monthly spending chart
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Visualizations page
  - Select file format from drop down menu on monthly spending graph

- **Expected Result**
  - Page should correctly export monthly spending graph in chosen format

Test Case 5

- System: Transactions.js, exportTransactions.js
- Check to make sure history table exports correctly
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Transactions page
  - Click EXPORT… and choose a file type

- **Expected Result**
  - User should receive a transaction history file of the chosen format

Test Case 6

- System: Transactions.js, exportTransaction.js
- Check to make sure that exported history contains the correct information
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Transactions page
  - Click EXPORT… and choose a file type
  - Open file and check information

- **Expected Result**
  - Exported information should match information on the _ page

Test Case 7

- System:
- Check to make sure user can choose which transactions to export
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to _ page
  - Click EXPORT and select which transactions to export, as well as the file type

- **Expected Result**
  - Exported file should contain only the specified transactions, all of which should have the correct information.

User Story 16: I would like to be able to categorize my expenses.

Test Case 1

- System: Financials.js
- Check to make sure users have the option to categorize their expenses
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Financials page
  - Attempt to add a new expenditure

- **Expected Result**
  - When attempting to add a new expenditure, user should be presented with options for how they can categorize their expenses (PayPal, Boiler Express, etc.)

Test Case 2

- System: Financials.js
- Check to make sure any changes made to expenses are reflected in the database
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Financials page
  - Modify any transaction with values different from what that transaction already contains.
  - (As admin) Navigate to college capital firebase console and check database

- **Expected Result**
  - Changes to any financial accounts should be reflected in the database.

Test Case 3

- System: Transactions.js
- Check to make sure expense sorting works
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to _ page
  - Use the filter to attempt to sort expenses by any of the relevant information

- **Expected Result**
    - Expenses should be displayed sorted by the chosen information

User Story 32: As a user, I would like to be able to message support 24/7 when I need help with the app.

Test Case 1

- System: messages.js
- Check to make sure user messages send correctly
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Support Page
  - Type message and hit SEND NEW TICKET

- **Expected Result**
  - Message should be displayed on page

Test Case 2

- System: messages.js
- Check to make sure sent messages are visible
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Support Page

- **Expected Result**
  - User's active and past support messages should be correctly displayed

Test Case 3

- System: messages.js
- Check to make sure user messages are stored in the database
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Support Page
  - Type message and hit SEND NEW TICKET
  - (As admin) Navigate to college capital firebase console and check support tickets

- **Expected Result**
    - Message should be correctly stored in database

Test Case 4

- System: messages.js
- Check to make sure that support messages are encrypted
- Severity: 2

- **Instructions**
    - Go to www.collegecap.org
    - Navigate to Support page
    - Type message and hit SEND NEW TICKET
    - (As admin) Navigate to college capital firebase console and check to make sure that messages are correctly encrypted

- **Expected Result**
    - Messages should be encrypted

User Story 33: As an administrator, I would like to be able to respond to support messages 24/7.

Test Case 1

- System: messages.js
- Check to make sure administrator receives user support messages
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Support page
  - Type message and hit SEND NEW TICKET
  - As administrator, navigate to firebase console

- **Expected Result**
  - Administrator should receive user's support message

Test Case 2

- System: messages.js
- Check to make sure received messages are correct
- Severity:

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Support page
  - Type message and hit SEND NEW TICKET
  - As administrator, navigate to firebase console

- **Expected Result**
  - Received message should contain the following information:
    - Message contents
    - Sender
    - Time received
    - Options to respond

Test Case 3

- System: messages.js
- Check to make user receives support responses
- Severity: 2

- **Instructions**
  - Navigate to Firebase console
  - Send message to a user

- **Expected Result**
  - User should receive message from sender under the name of Support

Test Case 4

- System: messages.js
- Check to make sure support tickets can be removed
- Severity: 2

- **Instructions**
  - Go to www.collegecap.org
  - Navigate to Support page
    - Send a support ticket if there are none
  - Click the “Close Ticket” button on any open ticket
  - (As admin) Navigate to college capital firebase console

- **Expected Result**
  - Previously active support tickets should be removed from the database if they are closed.

User Story 34: As an administrator, I would like to be able to respond to support messages when the user is offline.

Test Case 1

- System:
- Check to make sure administrator messages can be sent to offline users
- Severity: 2

- **Instructions**
  - Navigate to Firebase console
  - Send message to user that has already sent a message to an administrator

- **Expected Result**
  - Database should store the admin response

Test Case 2

- System:
- Check to make sure messages sent to an offline user are stored and viewable
- Severity:

- **Instructions**
  - Navigate to Firebase console
  - Send a message to an offline user

- **Expected Result**
  - Message sent to offline user should be viewable in the database

Test Case 3

- System:
- Check to make sure offline users are notified of a Support message upon logging back in
- Severity:

- **Instructions**
  - Navigate to Firebase console
  - Send message to offline account
  - Log in to offline account

- **Expected Result**
  - User should be presented with a notification that they were sent a support message while they were offline.

Test Case 4

- System: messages.js
- Check to make sure administrators can differentiate between online and offline users
- Severity: 2

- **Instructions**
    - Navigate to Firebase console
    - Navigate to where user data is stored

- **Expected Result**
    - User data should contain an indicator denoting whether or not they are currently online.