

Florida International University
School of Computing and Information Sciences

Software Engineering Focus

Final Deliverable

Project Title:

Snackability 2.0

Team Members: Carolina Karthik, Christian Canizares, Jeffrey Antoine, Jessiel Benitez, Adrian Serrano, Shafeeque Khan

Product Owner(s): Dr. Cristina Palacios, Lukkamol Prapkree

Instructor: Masoud Sadjadi

The MIT License (MIT)
Copyright (c) 2016 *Florida International University*

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Abstract

This document presents the information necessary to gain a good understanding of Snackability 2.0. Research shows that 98.2% of college students consume snacks daily. Snackability is a mobile application that aims to help people identify healthy snacks by providing a numerical score based on the chosen snack's nutritional information. This application is supported on both Android and iOS mobile operating systems.

This document explains our motivation for developing this application and the solution we implemented for the second release of Snackability.. It further maps out the main features that were implemented as well as fixes done to the previous iteration to make it more efficient and user friendly. In the features described, we list features that were finished, features that we did not have time to finish, and features that would be beneficial for future iterations. Lastly, we have given a manual on how to install and use the application.

Table of Contents

INTRODUCTION	5
CURRENT SYSTEM	5
PURPOSE OF NEW SYSTEM	5
USER STORIES	
IMPLEMENTED USER STORIES	6
PENDING USER STORIES	14
PROJECT PLAN	
HARDWARE AND SOFTWARE RESOURCES	16
SPRINTS PLAN	17
<i>Sprint 1</i>	17
<i>Sprint 2</i>	18
<i>Sprint 3</i>	20
<i>Sprint 4</i>	23
<i>Sprint 5</i>	25
<i>Sprint 6</i>	27
SYSTEM DESIGN	
ARCHITECTURAL PATTERNS	29
SYSTEM AND SUBSYSTEM DECOMPOSITION	30
DEPLOYMENT DIAGRAM	32
DESIGN PATTERNS	33
SYSTEM VALIDATION	34
GLOSSARY	59
APPENDIX	60
APPENDIX A - UML DIAGRAMS	60
<i>Use Case Diagrams</i>	60
<i>Class Diagrams</i>	62
<i>Sequence Diagrams</i>	63
APPENDIX B - USER INTERFACE DESIGN	65
APPENDIX C - SPRINT REVIEW REPORTS	66
APPENDIX D - USER MANUALS, SHORTCOMINGS/WISHLIST DOCUMENT AND OTHER DOCUMENTS	71
REFERENCES	75

INTRODUCTION

83% of US adolescents and 98.2% of US college students consume snack on a daily basis. The majority of these consumed snacks are deemed unhealthy by the USDA guidelines. There are no readily available applications that can be used to differentiate between healthy and unhealthy snacks in accordance to the current guidelines published by the USDA. Many applications rely on unchecked user entries in order to provide nutritional information about snacks. Snackability is a mobile application that aims to tackle this issue by providing a standardized score system that allows users to identify healthy snacks based on the USDA guidelines.

Current System

Snackability 1.0 provided a large in-house database of almost 1000 snacks curated by the product owner, Dr. Cristina Palacios, as well as the entire USDA database for users to search through. This initial release allowed for users to input a portion, unit of measurement for their serving, and an option for highly processed. The user would then receive a numeric score and the nutrition facts of the snack they chose from the search results. This system was not very effective, however, because it required users to have knowledge of nutrition facts information and what is a healthy portion for each of these items.

Purpose of New System

For the second release of Snackability, we focused on fixing crucial bugs, improving the user interface across the entire application and adding gamification elements. The homepage now features a visual representation of the points and leveling system, a graph of the user's average weekly score, and a history of their consumed snacks. The search page simply searches for the name while the additional information needed to get the score is on the snack information page. The overhauled snack information page allowed users to see what part of the snack lacked from nutritional value. This page also gives feedback showing which area of the nutritional information is healthy or unhealthy so that users can make an informed decision about the type of snacks they consume.

USER STORIES

The following section provides the detailed user stories that were implemented in this iteration of the Snackability 2.0 project. These user stories served as the basis for the implementation of the project's features. This section also shows the user stories that are to be considered for future development.

Implemented User Stories

User Story SNAK-5: Scale score

As a user I would like the score to be scaled from 0 to 11 **so that** I can have an accurate numerical representation of how good or bad my snack is nutritionally.

- Acceptance Criteria:
GIVEN I am a user in the snack information page
WHEN I check the score of the snack
THEN I see a correct score scaled from 0 - 11.

User Story SNAK-7: Show how score is calculated

As a user I would like to see how the score is calculated **so that** I have a better understanding of how the snack score works.

- Acceptance Criteria:
GIVEN I am a user on the snack information page
WHEN I get a score
THEN I can see how the snack score is calculated by different components.

User Story SNAK-8: Remove nutrition facts

As a developer I would like to remove the snack nutrition facts from the snack information page **so that** the user could see other features displayed in its place.

- Acceptance Criteria:
GIVEN I pick a snack from the search results page
WHEN I arrive to the snack's information
THEN I will no longer see the snack's nutrition facts.

User Story SNAK-10: Prioritize local database

As a user I want results that best match my search parameters **so that** I can quickly find the snack I am looking for.

- Acceptance Criteria:
GIVEN a search has been made
WHEN the databases are queried
THEN the local database results should be listed first.

User Story SNAK-11: Search history

As a user **I want** my search history saved **so that** I don't have to repeat my searches.

- Acceptance Criteria:
GIVEN I am a user who has made a search
WHEN I tap on the search bar
THEN I should be given a list of previous searches I have made.

User Story SNAK-13: Advanced search

As a user **I want** an advanced search option **so that** I can improve my search with keywords.

- Acceptance Criteria:
GIVEN I am a user who has registered and is making a search
WHEN I use the advanced search option
THEN I can use certain keywords to help me refine my search.

User Story SNAK-14: Consumed button

As a User **I would like** to click the consumed button **so that** it can add a snack to my consumed history.

- Acceptance Criteria:
GIVEN I am a user in the snack information page
WHEN I click the consumed history button
THEN the snack is added to the database.

User Story SNAK-15: Consumed history

As a user **I want** to keep a list of snacks I have consumed previously **so that** I can keep track of what I have eaten that day.

- Acceptance Criteria:
GIVEN I am a user that has previously consumed snacks
WHEN I go to the home page
THEN I see a list of snacks I have consumed.

User Story SNAK-16: Advanced nutrition feedback

As a user **I want** feedback on the nutrition of the snack **so that** I can make better snack choices in the future.

- Acceptance Criteria:
GIVEN I am a user who picked a search result
WHEN I get a score
THEN I get feedback based on the nutrient that got the lowest score.

User Story SNAK-17: Email user feedback

As a user **I want** to be able to send the app creators feedback **so that** they can improve the app.

- Acceptance Criteria:
GIVEN I am a user who has registered
WHEN I click the feedback button
THEN I am able to send the creators feedback on how to improve the app.

User Story SNAK-19: Font errors

As a user **I want** to be able to use the app without errors **so that** the app will run efficiently.

- Acceptance Criteria:
GIVEN I am a user who is loading the app
WHEN I start the app
THEN I can run the app without errors.

User Story SNAK-22: Portioning

As a user, **I want** to specify the portion of what I am eating **so that** the snack score changes accordingly.

- Acceptance Criteria:
GIVEN I am a user on search page
WHEN I enter a portion
AND I press search
THEN the resulting snacks should adjust according to the portion.

User Story SNAK-23: Graph of weekly average score

As a user **I want** to be able to see the scores of snacks I have consumed **so that** I can keep track of how my score changes.

- Acceptance Criteria:
GIVEN I am a user who has registered
WHEN I go to the home page
THEN I am able to see a graph with my average scores from the previous weeks.

User Story SNAK-24: Add a snack

As a user **I want** to suggest a new snack **so that** the applications database has more options.

- Acceptance Criteria:
GIVEN I am a user who is searching for a snack
WHEN I can't find the snack
THEN I can add the snack in the app by suggesting it via a form.

User Story SNAK-25: Deploy to Heroku

As a user I want to deploy to Heroku **so that** I can query the USDA database.

- Acceptance Criteria:
GIVEN I am a user who is searching for a snack
WHEN I search for a snack
THEN I can get snack results from the USDA database.

User Story SNAK-26: Advanced score feedback

As a user I want varied score feedback **so that** I can make better snack choices in the future.

- Acceptance Criteria:
GIVEN I am a user who picked a search result
WHEN I get a score
THEN I get varied general feedback based on the score of the snack.

User Story SNAK-27: Processed button

As a user I would like the processed button to apply to USDA and local results **so that** I can see the change in the score.

- Acceptance Criteria:
GIVEN I am a user who is searching for a snack
WHEN I put the information for processed
THEN I get a varying score for USDA and local results.

User Story SNAK-29: Firebase connection

As a user I want to be able to upload my data into Firebase **so that** I can see my progress.

- Acceptance Criteria:
GIVEN I am a user who has not registered
WHEN I put the information for processed
THEN I am able to create a profile that will have my progress linked.

User Story SNAK-30: Font for iOS and Android

As a user I want to use the Application without errors **so that** I can use the App on Android and ios devices.

- Acceptance Criteria:

GIVEN I am an iOS user on the application
WHEN I access the app
THEN I can use it with no font errors.

User Story SNAK-31: Move portion, units, processed from search to snack page

As a user **I want** the processed button, units, and portion moved to the snack score page **so that** I can search more easily by name only.

- Acceptance Criteria:
GIVEN I am a user on the search page
WHEN I search for a snack
THEN I get the search results and am taken to the score page with the additional information.

User Story SNAK-32: Navigation bar with Home, Search, Settings

As a user, **I want** a simplified navigation **so that** I can quickly navigate through the application.

- Acceptance Criteria:
GIVEN I am a user logged into the app
WHEN the navigation bar is rendered
THEN the navigation bar shows "Home, Search, Settings" icons.

User Story SNAK-33: Score breakdown table

As a User **I would like** the score breakdown to be displayed in a table **so that** I could easily understand the data.

- Acceptance Criteria:
GIVEN I am a user on the snack results page
WHEN I select a snack from the search results
THEN I see a table with a breakdown of the snack's score based on its nutrition.

User Story SNAK-34: Calculate average score

As a user **I want** to have an average of my score **so that** I can assess my diet.

- Acceptance Criteria:
GIVEN I am a user that has previously consumed snacks
WHEN I go to the consumed page
THEN I can see an average of my score on a graph.

User Story SNAK-35: Score breakdown table with visible feedback

As a User I would like the score breakdown table cell to return different colored icons depending on how good or bad a snack component score is **so that** I could easily view feedback on snack searches.

- Acceptance Criteria:
GIVEN I am a user on the snack results page
WHEN I select a snack from the search results
THEN the snack's score breakdown table is displayed with different color icons to show how good or bad each component is.

User Story SNAK-36: Local SQL connection to Heroku backend

As a user I want results not from the USDA database **so that** I can find a snack from the local database.

- Acceptance Criteria:
GIVEN I am a user who is searching for a snack
WHEN I search for a snack
THEN I can find a snack from the local database.

User Story SNAK-37: Highly processed button UI

As a User I would like two buttons "yes" and "no" for processed **so that** I have to choose and it is not defaulted.

- Acceptance Criteria:
GIVEN I am a user on the snack score page
WHEN I choose the processed information
THEN I have to pick either yes or no.

User Story SNAK-38: Snack detail information pop-ups

As a user, I want information about what certain nutrients **so that** I understand how the scoring works.

- Acceptance Criteria:
GIVEN I am a user on a snack detail page
WHEN I click the "i" icon for "processed"
THEN a tooltip shows up with relevant text.

User Story SNAK-39: Delete snacks in consumed history

As a user I want to be able to delete snacks **so that** I can keep my consumed history list accurate.

- Acceptance Criteria:

GIVEN I am a user that has previously consumed snacks
WHEN I go to the consumed page
THEN I can delete a snack from the consumed list.

User Story SNAK-41: Highly Processed information pop-up

As a user, **I want** information about what highly processed is **so that** I understand how the app works.

- Acceptance Criteria:
GIVEN I am a user on a snack detail page
WHEN I click the "i" icon for "processed"
THEN a tooltip shows up with relevant text.

User Story SNAK-44: Points and leveling system

As a user **I would like** to have a leveling and points system **so that** I can use this application as a sort of game.

- Acceptance Criteria:
GIVEN I am a user who has received a snack score
WHEN I consume a snack
THEN my points increase and contribute to the level I am on.

User Story SNAK-45: Progress bar for leveling

As a user, **I want** a visual representation of my levels and points **so that** I can easily see my improvement.

- Acceptance Criteria:
GIVEN I am a user logged in and going to consume a snack
WHEN I consume a snack
THEN a progress bar is updated on the home page with my points and level.

GIVEN I am a user logged in and on the home page
WHEN I delete a previously consumed snack
THEN a progress bar is updated on the home page with my points and level.

User Story SNAK-46: Double points for holidays and weekends

As a user **I want** to get more points for consuming a healthy snack **so that** I can level up and continue using the app.

- Acceptance Criteria:
GIVEN I am a user who has registered and consumes healthy snacks
WHEN I consume a healthy snack

THEN I receive points relative to the health of the snack.

User Story SNAK-48: Settings page

As a User I would like the settings page organized in a list **so that** I easily access the different features.

- Acceptance Criteria:
GIVEN I am a user who is using the application
WHEN I go to the Settings tab on the navigation bar
THEN the settings features are listed clearly.

User Story SNAK-49: Separate USDA and local results

As a user, I want to be able to distinguish between Snackability database and USDA database **so that** I know where the information and score are coming from.

- Acceptance Criteria:
GIVEN I am a user on search page
WHEN I make a search
THEN the search page shows items that indicate Snackability database or usda database.

User Story SNAK-50: Confirmation message when consuming a snack

As a user, I want feedback that I have consumed a snack **so that** I don't over tap.

- Acceptance Criteria:
GIVEN I am a user on the snack detail page
WHEN I tap on the consume button
THEN feedback is shown that a snack has been consumed.

User Story SNAK-51: Clicking on a snack in consumed history takes you to its snack page

As a user I want to be able to click on a snack in my consumed history **so that** I can access the snack page.

- Acceptance Criteria:
GIVEN I am a user that has previously consumed snacks
WHEN I go to the consumed page
THEN I can click on a snack and access its snack page.

User Story SNAK-52: Show no results when an invalid snack name is searched

As a user I want to search for snacks **so that** I can check its health level and consume it.

- Acceptance Criteria:
GIVEN I am a user on search page
WHEN I search for a snack that does not exist

THEN I should see a "No Results" message.

User Story SNAK-53: Update snack database with fruits and vegetables

As a User I would like raw fruits and vegetables in the database **so that** I can easily search for them.

- Acceptance Criteria:
GIVEN I am a user on search page
WHEN I search for snack name that is a fruit or vegetable
THEN I receive the pertinent results.

User Story SNAK-54: Update “About Us” page

As a user I would like the application developers’ contact information **so that** I have some form of contact to whom I could address any issues or feedback.

- Acceptance Criteria:
GIVEN I am a user on the settings page
WHEN I go into the “About Us” section
THEN I will get information on the app and its developers, as well as means to contact them.

Pending User Stories

User Story SNAK-12: Search with barcode scanning

As a developer, I want to add searching by scanning the barcode of a snack so the user does not have to search manually.

- Acceptance Criteria:
GIVEN I am a user with a snack in my hand
WHEN I use the scanning option
THEN I get the correct snack information page.

User Story SNAK-18: Reminder notifications

As a developer, I want to have reminder notifications on the app to remind the user to use it every day.

- Acceptance Criteria:
GIVEN I am a frequent user
WHEN I have not logged into the application in two days
THEN I receive a notification that reminds me to input snacks.

User Story SNAK-20: Build app for iOS

As a developer, I want to build the application for iOS so it could be deployed to the Apple App Store.

- Acceptance Criteria:
GIVEN I am a user who wants to download Snackability app
WHEN I go to the Apple App store
THEN I can download the Snackability app.

User Story SNAK-21: Build app for Android

As a developer, I want to build the application for Android so it could be deployed to the Google Play Store.

- Acceptance Criteria:
GIVEN I am a user who wants to download Snackability app
WHEN I go to the Google Play Store
THEN I can download the Snackability app.

User Story SNAK-40: Consumed History UI

As a developer, I want to further develop the consumed history on the homepage to make it more user friendly.

- Acceptance Criteria:
GIVEN I am a user on the home page
WHEN I go to the consumed history
THEN I can use it easily because it is user friendly.

User Story SNAK-43: Preview of average score before consuming

As a developer, I want to show a preview of what the user's average snack score of the day would look like before consuming.

- Acceptance Criteria:
GIVEN I am a user who selected a search result
WHEN I calculate my snack score
THEN I am shown a preview of what my average score would look like if I were to consume it.

PROJECT PLAN

This section describes the planning that went into the realization of this project. This project incorporated the agile development techniques and as such required the sprints to be planned. These sprint plannings are detailed in the section. This section also describes the components, both software and hardware, chosen for this project.

Hardware and Software Resources

The following is a list of all hardware and software resources that were used in this project:

Hardware

1. Mobile phone running latest version of Android
2. Mobile phone running latest version of iOS
3. Computer: used to run development software
4. Snacks: for testing and eating!

Software

1. Visual Studio Code: Used for coding
2. Android Studio: Used for emulation of Android
3. XCode: Used for emulation of iOS
4. MAMP: Used to instantiate SQL servers on a local machine
5. MySQLWorkbench: Used to develop queries for local database
6. Windows 10: For development
7. macOS: For development and compilation for iOS
8. Jira: Used for agile development and project management
9. Bitbucket: Used for source control management
10. Discord: Used for team communication
11. LucidCharts: Used to create UML diagrams
12. Confluence: Used to store project documentation
13. Hipchat: Used for communication with product owner, mentor, and professor

Sprint Plans

Sprint 1

In this sprint we got ourselves familiar with the code base. We also set up our development environments and started fixing crucial bugs that were preventing the app from starting.

User Story SNAK-5: Scale score

As a user I would like the score to be scaled from 0 to 11 **so that** I can have an accurate numerical representation of how good or bad my snack is nutritionally.

- Acceptance Criteria:
GIVEN I am a user in the snack information page
WHEN I check the score of the snack
THEN I see a correct score scaled from 0 - 11.

User Story SNAK-8: Remove nutrition facts

As a developer I would like to remove the snack nutrition facts from the snack information page **so that** the user could see other features displayed in its place.

- Acceptance Criteria:
GIVEN I pick a snack from the search results page
WHEN I arrive to the snack's information
THEN I will not longer see the snack's nutrition facts.

User Story SNAK-10: Prioritize local database

As a user I want results that best match my search parameters **so that** I can quickly find the snack I am looking for.

- Acceptance Criteria:
GIVEN a search has been made
WHEN the databases are queried
THEN the local database results should be listed first.

User Story SNAK-15: Consumed history

As a user I want to keep a list of snacks I have consumed previously **so that** I can keep track of what I have eaten that day.

- Acceptance Criteria:
GIVEN I am a user that has previously consumed snacks
WHEN I go to the home page

THEN I see a list of snacks I have consumed.

User Story SNAK-17: Email user feedback

As a user I want to be able to send the app creators feedback **so that** they can improve the app.

- Acceptance Criteria:
GIVEN I am a user who has registered
WHEN I click the feedback button
THEN I am able to send the creators feedback on how to improve the app.

User Story SNAK-18: Reminder Notifications

As a user, I want notifications from the app **so that** I am reminded to use the app everyday

- Acceptance Criteria:
GIVEN I am a frequent user,
WHEN I have not used the app in two days
THEN I receive a notification that reminds me to input snacks.

User Story SNAK-19: Font errors

As a user I want to be able to use the app without errors **so that** the app will run efficiently.

- Acceptance Criteria:
GIVEN I am a user who is loading the app
WHEN I start the app
THEN I can run the app without errors.

Sprint 2

In this sprint we prioritized core mechanics that weren't working correctly. We also began adding features.

User Story SNAK-7: Show how score is calculated

As a user I would like to see how the score is calculated **so that** I have a better understanding of how the snack score works.

- Acceptance Criteria:
GIVEN I am a user on the snack information page
WHEN I get a score
THEN I can see how the snack score is calculated by different components.

User Story SNAK-8: Remove nutrition facts

As a developer I would like to remove the snack nutrition facts from the snack information page **so that** the user could see other features displayed in its place.

- Acceptance Criteria:
GIVEN I pick a snack from the search results page
WHEN I arrive to the snack's information
THEN I will no longer see the snack's nutrition facts.

User Story SNAK-11: Search history

As a user **I want** my search history saved **so that** I don't have to repeat my searches.

- Acceptance Criteria:
GIVEN I am a user who has made a search
WHEN I tap on the search bar
THEN I should be given a list of previous searches I have made.

User Story SNAK-14: Consumed button

As a User **I would like** to click the consumed button **so that** it can add a snack to my consumed history.

- Acceptance Criteria:
GIVEN I am a user in the snack information page
WHEN I click the consumed history button
THEN the snack is added to the database.

User Story SNAK-15: Consumed history

As a user **I want** to keep a list of snacks I have consumed previously **so that** I can keep track of what I have eaten that day.

- Acceptance Criteria:
GIVEN I am a user that has previously consumed snacks
WHEN I go to the home page
THEN I see a list of snacks I have consumed.

User Story SNAK-22: Portioning

As a user, **I want** to specify the portion of what I am eating **so that** the snack score changes accordingly.

- Acceptance Criteria:
GIVEN I am a user on search page
WHEN I enter a portion
AND I press search
THEN the resulting snacks should adjust according to the portion.

User Story SNAK-25: Deploy to heroku

As a user I want to deploy to Heroku **so that** I can query the USDA database.

- Acceptance Criteria:
GIVEN I am a user who is searching for a snack
WHEN I search for a snack
THEN I can get snack results from the USDA database.

User Story SNAK-27: Processed button

As a user I would like the processed button to apply to USDA and local results **so that** I can see the change in the score.

- Acceptance Criteria:
GIVEN I am a user who is searching for a snack
WHEN I put the information for processed
THEN I get a varying score for USDA and local results.

User Story SNAK-29: Firebase connection

As a user I want to be able to upload my data into Firebase **so that** I can see my progress.

- Acceptance Criteria:
GIVEN I am a user who has not registered
WHEN I put the information for processed
THEN I am able to create a profile that will have my progress linked.

Sprint 3

In this sprint we began adding features to the app. We added the consume history, better nutrition feedback and a graph of average consumed scores. Most importantly we deployed the app backend to heroku and got a running version into QA.

User Story SNAK-13: Advanced search

As a user I want an advanced search option **so that** I can improve my search with keywords.

- Acceptance Criteria:
GIVEN I am a user who has registered and is making a search
WHEN I use the advanced search option
THEN I can use certain keywords to help me refine my search.

User Story SNAK-15: Consumed history

As a user I want to keep a list of snacks I have consumed previously **so that** I can keep track of what I have eaten that day.

- Acceptance Criteria:
GIVEN I am a user that has previously consumed snacks

WHEN I go to the home page
THEN I see a list of snacks I have consumed.

User Story SNAK-16: Advanced nutrition feedback

As a user I want feedback on the nutrition of the snack **so that** I can make better snack choices in the future.

- Acceptance Criteria:
GIVEN I am a user who picked a search result
WHEN I get a score
THEN I get feedback based on the nutrient that got the lowest score.

User Story SNAK-23: Graph of weekly average score

As a user I want to be able to see the scores of snacks I have consumed **so that** I can keep track of how my score changes.

- Acceptance Criteria:
GIVEN I am a user who has registered
WHEN I go to the home page
THEN I am able to see a graph with my average scores from the previous weeks.

User Story SNAK-25: Deploy to heroku

As a user I want to deploy to Heroku **so that** I can query the USDA database.

- Acceptance Criteria:
GIVEN I am a user who is searching for a snack
WHEN I search for a snack
THEN I can get snack results from the USDA database.

User Story SNAK-30: Font for iOS and Android

As a user I want to use the Application without errors **so that** I can use the App on Android and ios devices.

- Acceptance Criteria:
GIVEN I am an iOS user on the application
WHEN I access the app
THEN I can use it with no font errors.

User Story SNAK-31: Move portion, units, processed from search to snack page

As a user I want the processed button, units, and portion moved to the snack score page **so that** I can search more easily by name only.

- Acceptance Criteria:

GIVEN I am a user on the search page
WHEN I search for a snack
THEN I get the search results and am taken to the score page with the additional information.

User Story SNAK-32: Navigation bar with Home, Search, Settings

As a user, I want a simplified navigation **so that** I can quickly navigate through the application.

- Acceptance Criteria:
GIVEN I am a user logged into the app
WHEN the navigation bar is rendered
THEN the navigation bar shows "Home, Search, Settings" icons.

User Story SNAK-33: Score breakdown table

As a User I would like the score breakdown to be displayed in a table **so that** I could easily understand the data.

- Acceptance Criteria:
GIVEN I am a user on the snack results page
WHEN I select a snack from the search results
THEN I see a table with a breakdown of the snack's score based on its nutrition.

User Story SNAK-34: Calculate average score

As a user I want to have an average of my score **so that** I can assess my diet.

- Acceptance Criteria:
GIVEN I am a user that has previously consumed snacks
WHEN I go to the consumed page
THEN I can see an average of my score on a graph.

User Story SNAK-35: Score breakdown table with visible feedback

As a User I would like the score breakdown table cell to return different colored icons depending on how good or bad a snack component score is **so that** I could easily view feedback on snack searches.

- Acceptance Criteria:
GIVEN I am a user on the snack results page
WHEN I select a snack from the search results

User Story SNAK-36: Local SQL connection to Heroku backend

As a user I want results not from the USDA database **so that** I can find a snack from the local database.

- Acceptance Criteria:
GIVEN I am a user who is searching for a snack
WHEN I search for a snack
THEN I can find a snack from the local database.

Sprint 4

In this sprint we made significant UI changes. We began updating the consumed history UI and also the snack detail page UI. We also began implementing features for deleting a consumed snack.

User Story SNAK-13: Advanced search

As a user I want an advanced search option **so that** I can improve my search with keywords.

- Acceptance Criteria:
GIVEN I am a user who has registered and is making a search
WHEN I use the advanced search option
THEN I can use certain keywords to help me refine my search.

User Story SNAK-16: Advanced nutrition feedback

As a user I want feedback on the nutrition of the snack **so that** I can make better snack choices in the future.

- Acceptance Criteria:
GIVEN I am a user who picked a search result
WHEN I get a score
THEN I get feedback based on the nutrient that got the lowest score.

User Story SNAK-24: Add a snack

As a user I want to suggest a new snack **so that** the applications database has more options.

- Acceptance Criteria:
GIVEN I am a user who is searching for a snack
WHEN I can't find the snack
THEN I can add the snack in the app by suggesting it via a form.

User Story SNAK-26: Advanced score feedback

As a user I want varied score feedback **so that** I can make better snack choices in the future.

- Acceptance Criteria:

GIVEN I am a user who picked a search result
WHEN I get a score
THEN I get varied general feedback based on the score of the snack.

User Story SNAK-35: Score breakdown table with visible feedback

As a User I would like the score breakdown table cell to return different colored icons depending on how good or bad a snack component score is **so that** I could easily view feedback on snack searches.

- Acceptance Criteria:
GIVEN I am a user on the snack results page
WHEN I select a snack from the search results

User Story SNAK-37: Highly processed button UI

As a User I would like two buttons "yes" and "no" for processed **so that** I have to choose and it is not defaulted.

- Acceptance Criteria:
GIVEN I am a user on the snack score page
WHEN I choose the processed information
THEN I have to pick either yes or no.

User Story SNAK-38: Snack detail information pop-ups

As a user, I want information about what certain nutrients **so that** I understand how the scoring works.

- Acceptance Criteria:
GIVEN I am a user on a snack detail page
WHEN I click the "i" icon for "processed"
THEN a tooltip shows up with relevant text.

User Story SNAK-39: Delete snacks in consumed history

As a user I want to be able to delete snacks **so that** I can keep my consumed history list accurate.

- Acceptance Criteria:
GIVEN I am a user that has previously consumed snacks
WHEN I go to the consumed page
THEN I can delete a snack from the consumed list.

User Story SNAK-40: Consumed History UI

As a developer, I want to further develop the consumed history on the homepage to make it more user friendly.

- Acceptance Criteria:
GIVEN I am a user on the home page
WHEN I go to the consumed history
THEN I can use it easily because it is user friendly.

User Story SNAK-41: Highly Processed information pop-up

As a user, I want information about what highly processed is **so that** I understand how the app works.

- Acceptance Criteria:
GIVEN I am a user on a snack detail page
WHEN I click the "i" icon for "processed"
THEN a tooltip shows up with relevant text.

User Story SNAK-43: Preview of average score before consuming

As a developer, I want to show a preview of what the user's average snack score of the day would look like before consuming.

- Acceptance Criteria:
GIVEN I am a user who selected a search result
WHEN I calculate my snack score
THEN I am shown a preview of what my average score would look like if I were to consume it.

Sprint 5

This sprint focused on adding gamification features. We added a leveling and point system, as well as a progress bar to tell the user what level they are on. We also updated the UI of the settings page.

User Story SNAK-13: Advanced search

As a user I want an advanced search option **so that** I can improve my search with keywords.

- Acceptance Criteria:
GIVEN I am a user who has registered and is making a search
WHEN I use the advanced search option
THEN I can use certain keywords to help me refine my search.

User Story SNAK-39: Delete snacks in consumed history

As a user **I want** to be able to delete snacks **so that** I can keep my consumed history list accurate.

- Acceptance Criteria:
GIVEN I am a user that has previously consumed snacks
WHEN I go to the consumed page
THEN I can delete a snack from the consumed list.

User Story SNAK-40: Consumed History UI

As a developer, I want to further developer the consumed history on the homepage to make it more user friendly.

- Acceptance Criteria:
GIVEN I am a user on the home page
WHEN I go to the consumed history
THEN I can use it easily because it is user friendly.

User Story SNAK-43: Preview of average score before consuming

As a developer, I want to show a preview of what the user's average snack score of the day would look like before consuming.

- Acceptance Criteria:
GIVEN I am a user who selected a search result
WHEN I calculate my snack score
THEN I am shown a preview of what my average score would look like if I were to consume it.

User Story SNAK-44: Points and leveling system

As a user I would like to have a leveling and points system **so that** I can use this application as a sort of game.

- Acceptance Criteria:
GIVEN I am a user who has received a snack score
WHEN I consume a snack
THEN my points increase and contribute to the level I am on.

User Story SNAK-45: Progress bar for leveling

As a user, I want a visual representation of my levels and points **so that** I can easily see my improvement.

- Acceptance Criteria:
GIVEN I am a user logged in and going to consume a snack
WHEN I consume a snack

THEN a progress bar is updated on the home page with my points and level.

GIVEN I am a user logged in and on the home page

WHEN I delete a previously consumed snack

THEN a progress bar is updated on the home page with my points and level.

User Story SNAK-46: Double points for holidays and weekends

As a user I want to get more points for consuming a healthy snack **so that** I can level up and continue using the app.

- Acceptance Criteria:

GIVEN I am a user who has registered and consumes healthy snacks

WHEN I consume a healthy snack

THEN I receive points relative to the health of the snack.

User Story SNAK-48: Settings page

As a User I would like the settings page organized in a list **so that** I easily access the different features.

- Acceptance Criteria:

GIVEN I am a user who is using the application

WHEN I go to the Settings tab on the navigation bar

THEN the settings features are listed clearly.

Sprint 6

In the last sprint we focused on small polishing features. We also updated the about us page to include the Snackability 2.0 development team.

User Story SNAK-49: Separate USDA and local results

As a user, I want to be able to distinguish between Snackability database and USDA database **so that** I know where the information and score are coming from.

- Acceptance Criteria:

GIVEN I am a user on search page

WHEN I make a search

THEN the search page shows items that indicate Snackability database or usda database.

User Story SNAK-50: Confirmation message when consuming a snack

As a user, I want feedback that I have consumed a snack **so that** I don't over tap.

- Acceptance Criteria:

GIVEN I am a user on the snack detail page
WHEN I tap on the consume button
THEN feedback is shown that a snack has been consumed.

User Story SNAK-51: Clicking on a snack in consumed history takes you to its snack page
As a user **I want** to be able to click on a snack in my consumed history **so that** I can access the snack page.

- Acceptance Criteria:
GIVEN I am a user that has previously consumed snacks
WHEN I go to the consumed page
THEN I can click on a snack and access its snack page.

User Story SNAK-52: Show no results when an invalid snack name is searched
As a user **I want** to search for snacks **so that** I can check its health level and consume it.

- Acceptance Criteria:
GIVEN I am a user on search page
WHEN I search for a snack that does not exist
THEN I should see a "No Results" message.

User Story SNAK-53: Update snack database with fruits and vegetables
As a User **I would like** raw fruits and vegetables in the database **so that** I can easily search for them.

- Acceptance Criteria:
GIVEN I am a user on search page
WHEN I search for snack name that is a fruit or vegetable
THEN I receive the pertinent results.

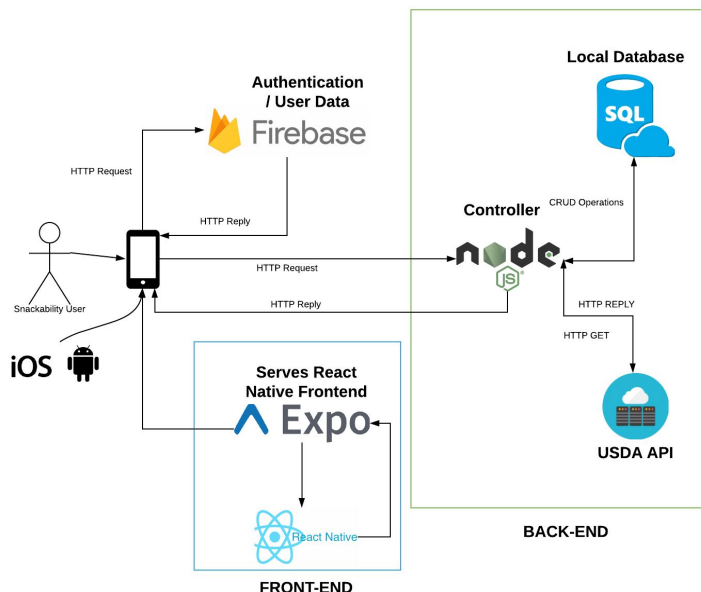
User Story SNAK-54: Update “About Us” page
As a user **I would like** the application developers’ contact information **so that** I have some form of contact to whom I could address any issues or feedback.

- Acceptance Criteria:
GIVEN I am a user on the settings page
WHEN I go into the “About Us” section
THEN I will get information on the app and its developers, as well as means to contact them.

SYSTEM DESIGN

This section contains information on the design decisions that went into this project. The architecture patterns are outlined and explained. The entire system is shown in a package diagram and the subsystems are explained. Finally, the design patterns used in the project are discussed.

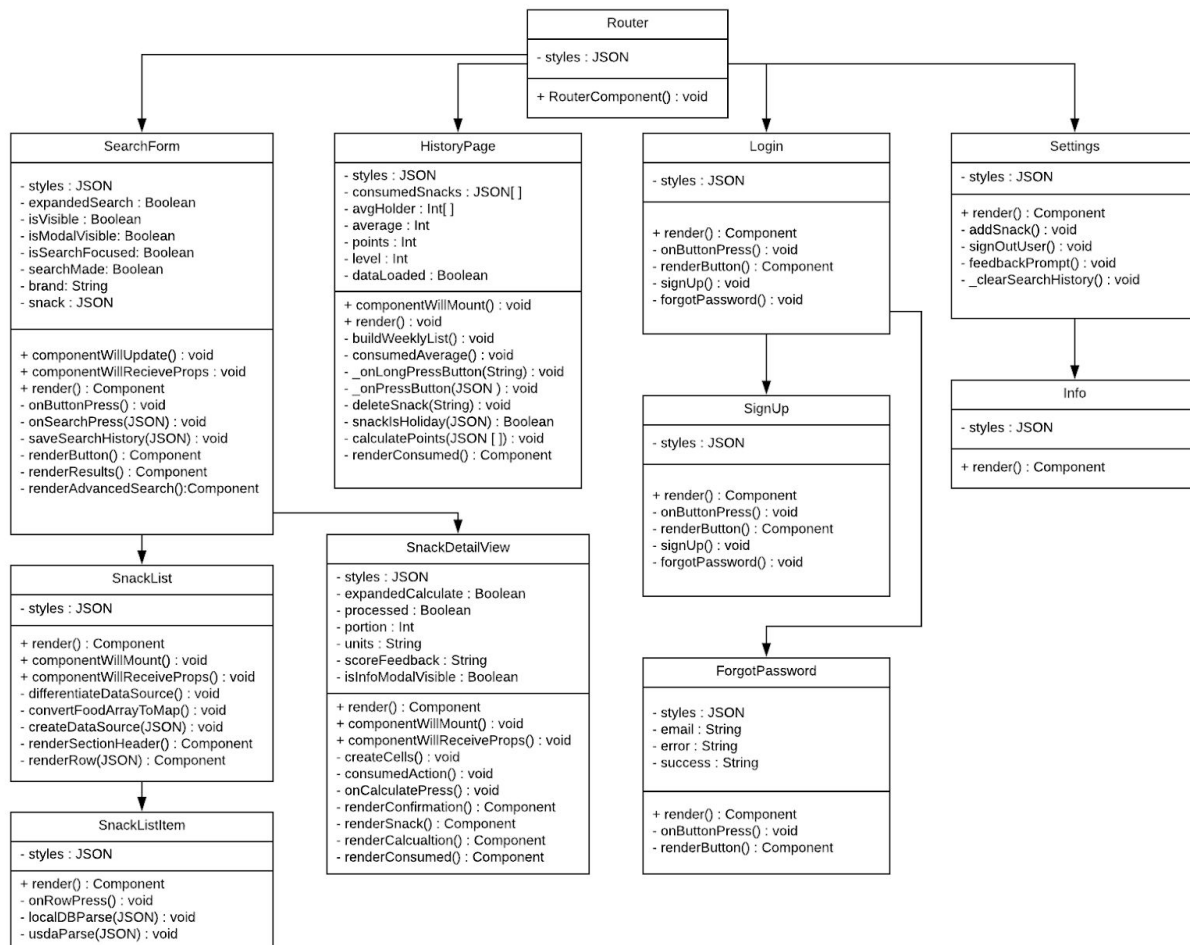
Architectural Patterns



The diagram above describes our architecture at a high level. Our application is served the front-end via Expo which compiles the React Native javascript code to the desired mobile platform. A NodeJS server running on Heroku serves out Snackability API which also communicated with the USDA API and our SQL Database also running on Heroku.

System and Subsystem Decomposition

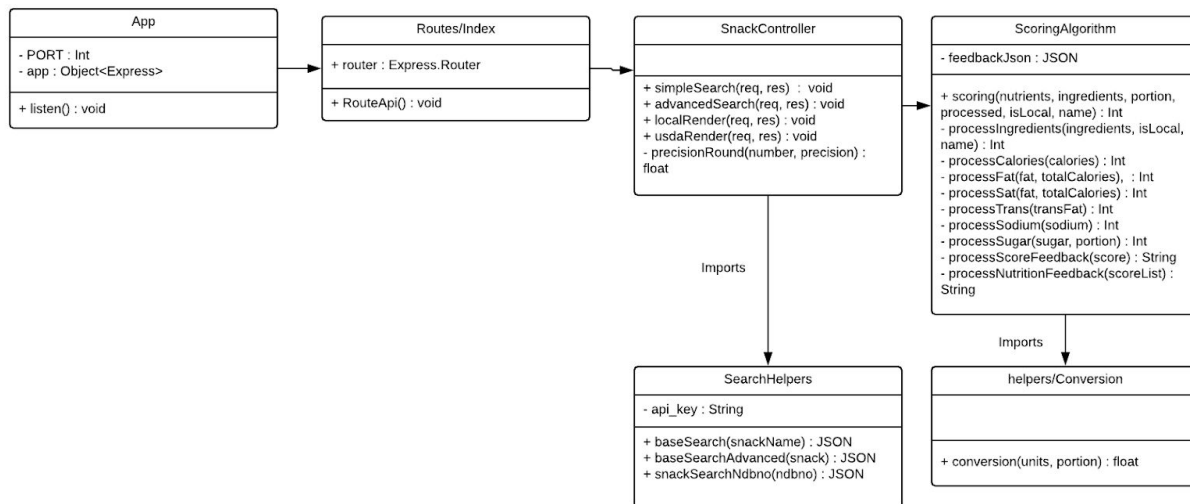
Front-End



The class diagram above illustrates our front-end architecture. At the top we have our Router class. The router class handles the instantiation of our react native scenes and components. The router spawns four scenes: SearchForm, HistoryPage, Login, and Settings. The scenes represent the pages of our application: Search Page, Home Page, Login Page, and Settings Page, respectively. Each page may spawn child components that are indicated by the arrows.

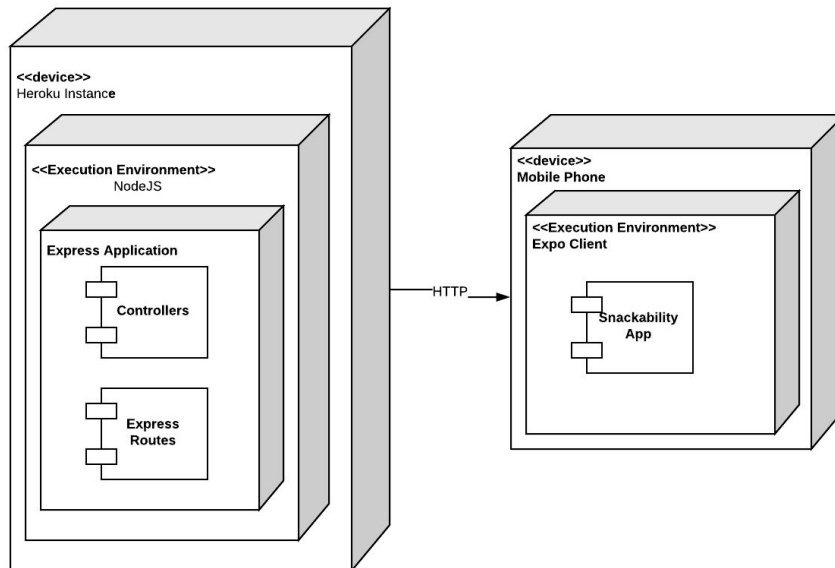
Every object in the diagram is a react native component. Components have optionally implementable methods but every component needs to have an implemented render function.

Back-End



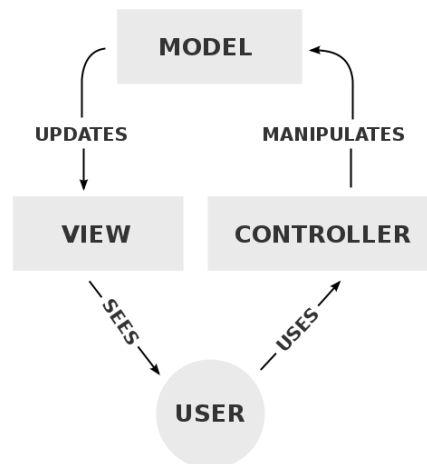
The above graph illustrates the Back-End system. The Node application begins in the App class. The App class starts the web server and points to an API router. The Index object (within Routes/) instantiates our Express router and describes the api endpoints and which functions to call for each. The SnackController object holds all the functions in our API. SearchHelpers holds our calls to the USDA API. Finally ScoringAlgorithm contains functions relating to how the snacks are scored.

Deployment Diagram



The above deployment diagram illustrates the deployed server and where everything is hosted. A heroku instance is hosting a node server. The server also talks to a SQL Server running MySQL. Finally the mobile device is is running Expo which hosts the Snackability App.

Design Patterns



The main design pattern we used is the Model-View-Controller pattern. Our view was a React Native Application which contained all of our buttons, icons, pages, animations, and styling. React Native was chosen since it provides an easy environment to develop for both Android and iOS. The view interacts with the controller via a REST API to a NodeJS server with Express. Finally the controller interacts with the model via a SQL server and the USDA API.

SYSTEM VALIDATION

Unit Tests

User Story SNAK-5: Scaling score

Test Case 1: Scaling score

- **Purpose:** Check that the user receives a snack score scaled from 0 to 11.
- **Precondition:** User has chosen a search result
- **Test Procedure:**
 1. User is taken to the snack detail page
 2. The score is calculated based on the given portion size, processed entry, and units
 3. The score is displayed
- **Expected Result:** User receives a score from 0 to 11.
- **Actual Results:** User receives a score from 0 to 11. (PASS)

User Story SNAK-7: Show how score is calculated

Test Case 1: Show how score is calculated

- **Purpose:** Check that the nutrition information is displayed to show how the score was calculated
- **Precondition:** User is on the search page
- **Test Procedure:**
 1. User makes a search for a snack
 2. User selects the relevant snack from the results page
 3. The score and breakdown of the score are displayed on the snack page

- **Expected Result:** Breakdown of the snack's score is displayed.
- **Actual Results:** Breakdown of the snack's score is displayed. (PASS)

User Story SNAK-8: Remove nutrition facts

Test Case 1: Remove nutrition facts

- **Purpose:** Check that the nutrition facts were removed so that the breakdown could be displayed instead
- **Precondition:** User is on the search page
- **Test Procedure:**
 1. User makes a search for a snack
 2. User selects the relevant snack from the results page
 3. The score and breakdown of the score are displayed on the snack page
- **Expected Result:** Breakdown of the snack's score is displayed instead of nutrition facts.
- **Actual Results:** Breakdown of the snack's score is displayed instead of nutrition facts. (PASS)

User Story SNAK-10: Prioritize local database

Test Case 1: Prioritize local database

- **Purpose:** Check that the local database results are listed before the USDA results
- **Precondition:** User is on the search page
- **Test Procedure:**
 1. User makes a search for a snack
 2. Controller returns search results from database
 3. Results are displayed
- **Expected Result:** Local database results are listed above the USDA results.

- **Actual Results:** Local database results are listed above the USDA results. (PASS)

User Story SNAK-11: Search History

Test Case 1: Search History

- **Purpose:** Check that a search history shows on the search page under the search bar
- **Precondition:** User is on the search page and has previously made a search for “doritos”
- **Test Procedure:**
 1. User clicks on search bar
 2. A list of previously searched items is shown
 3. User selects “doritos” from the search history
 4. “Doritos” search results are displayed
- **Expected Result:** Selecting “doritos” from search history searches for “doritos”.
- **Actual Results:** Selecting “doritos” from search history searches for “doritos” . (PASS)

User Story SNAK-13: Advanced Search

Test Case 1: Advanced Search #1

- **Purpose:** Check that the user can additionally search by the brand name instead of the general search.
- **Precondition:** User is on the search page
- **Test Procedure:**
 1. User clicks on three dots on search bar
 2. Display “brand name” and “snack name” input fields
 3. User searches “oreo” by brand name
 4. Search results are displayed
- **Expected Result:** User gets search results for “oreo”.
- **Actual Results:** User gets search results for “oreo”. (PASS)

Test Case 2: Advanced Search #2

- **Purpose:** Check that the user can additionally search by the snack name instead of the general search.
- **Precondition:** User is on the search page
- **Test Procedure:**
 1. User clicks on three dots on search bar
 2. Display “brand name” and “snack name” input fields
 3. User searches “golden” by snack name
 4. Search results are displayed
- **Expected Result:** User gets search results for “golden”.
- **Actual Results:** User gets search results for “golden”. (PASS)

Test Case 3: Advanced Search #3

- **Purpose:** Check that the user can additionally search by the brand and snack name instead of the general search.
- **Precondition:** User is on the search page
- **Test Procedure:**
 1. User clicks on three dots on search bar
 2. Display “brand name” and “snack name” input fields
 3. User searches “oreo” by brand name and “golden” by snack name
 4. Search results are displayed
- **Expected Result:** User gets search results for “golden oreo”.
- **Actual Results:** User gets search results for “golden oreo”. (PASS)

User Story SNAK-14: Consumed button**Test Case 1: Consumed button**

- **Purpose:** Check that the consumed button adds the snack to the user's consumed history
- **Precondition:** User has chosen a search result
- **Test Procedure:**
 1. User clicks "consume" button
 2. Firebase adds the snack to the user's history
- **Expected Result:** The snack is added to the user's consumed history.
- **Actual Results:** The snack is added to the user's consumed history. (PASS)

User Story SNAK-15: Consumed history

Test Case 1: Consumed history

- **Purpose:** Check that the user can consume a snack and the snack is added to their consumed history.
- **Precondition:** User has chosen a search result
- **Test Procedure:**
 3. User clicks "consume" button
 4. Firebase adds the snack to the user's history
 5. The consumed history updates to display the updated consumed history
- **Expected Result:** The snack is added to the user's consumed history.
- **Actual Results:** The snack is added to the user's consumed history. (PASS)

User Story SNAK-16: Advanced nutrition feedback

Test Case 1: Advanced nutrition feedback #1

- **Purpose:** Check that when the snack score is calculated, the user is given random nutrition feedback specific to the lowest scoring nutrient.
- **Precondition:** User has chosen a search result

- **Test Procedure:**
 1. Snack score is calculated
 2. The nutrition feedback is chosen based on the lowest nutrient score
 3. The score and feedback are displayed
- **Expected Result:** User receives a random nutrition feedback from the list of possible feedbacks.
- **Actual Results:** User receives a random nutrition feedback from the list of possible feedbacks. (PASS)

Test Case 2: Advanced nutrition feedback #2

- **Purpose:** Check that when the snack score is calculated, the user is given random nutrition feedback specific to the lowest scoring nutrient.
- **Precondition:** User has chosen the same search result
- **Test Procedure:**
 1. Snack score is calculated
 2. The nutrition feedback is chosen based on the lowest nutrient score
 3. The score and feedback are displayed
- **Expected Result:** User receives a random nutrition feedback from the list of possible feedbacks.
- **Actual Results:** User receives a random nutrition feedback from the list of possible feedbacks. (PASS)

User Story SNAK-17: Email user feedback

Test Case 1: Email user feedback

- **Purpose:** Check that the user can email feedback to the application creators.
- **Precondition:** User is on the Settings page

- **Test Procedure:**
 1. User clicks on “Email Feedback”
 2. App opens the default email app
 3. User fills out email and sends it
 4. The Snackability email (snackabilityapp@gmail.com) receives the email.
- **Expected Result:** The Snackability email receives the email.
- **Actual Results:** The Snackability email receives the email. (PASS)

User Story SNAK-19: Font errors

Test Case 1: Font errors

- **Purpose:** Check that the user can render the application without font errors..
- **Precondition:** User is on the Snackability “Sign In” page.
- **Test Procedure:**
 1. User signs in to the application
 2. Application renders the home page.
- **Expected Result:** Application renders the home page without errors.
- **Actual Results:** Application renders the home page without errors. (PASS)

User Story SNAK-22: Portioning

Test Case 1: Portioning

- **Purpose:** Check that the user can input a portion that affects the snack score.
- **Precondition:** User is on search page.
- **Test Procedure:**
 1. User searches for a snack
 2. User chooses a search result
 3. User enters a portion in the “Portion” input field

4. The snack score is calculated
- **Expected Result:** The snack score is calculated according to the input portion.
 - **Actual Results:** The snack score is calculated according to the input portion. (PASS)

User Story SNAK-23: Graph of weekly average score

Test Case 1: Graph of weekly average score

- **Purpose:** Check that the user can view a graph of their weekly average score on the home page.
- **Precondition:** User has a consumed history
- **Test Procedure:**
 1. User goes to the home page
 2. Firebase returns the historical data
 3. Home page displays graph with the historical data
- **Expected Result:** The snack score is calculated according to the input portion.
- **Actual Results:** The snack score is calculated according to the input portion. (PASS)

User Story SNAK-24: Add a snack

Test Case 1: Add a snack

- **Purpose:** Check that the user can request the app developers to add a snack to the database
- **Precondition:** User is on the Settings page
- **Test Procedure:**
 1. User clicks “Add a Snack”
 2. App redirects to a Google form
 3. User fills out and submits Google form

- **Expected Result:** The Snackability developers receive the Google form.
- **Actual Results:** The Snackability developers receive the Google form. (PASS)

User Story SNAK-25: Deploy to Heroku

Test Case 1: Deploy to Heroku

- **Purpose:** Check that the user can receive search results from the USDA database.
- **Precondition:** User is on search page.
- **Test Procedure:**
 1. User searches for a snack
 2. Heroku returns USDA results
 3. USDA results are displayed
- **Expected Result:** USDA results are returned.
- **Actual Results:** USDA results are returned. (PASS)

User Story SNAK-26: Advanced score feedback

Test Case 1: Advanced score feedback #1

- **Purpose:** Check that the user can receive a random score feedback from the feedback list based on the score.
- **Precondition:** User has chosen a search result.
- **Test Procedure:**
 1. The snack score is calculated
 2. The score feedback is chosen randomly based on the score
 3. The score feedback and result are displayed
- **Expected Result:** User receives randomly chosen score feedback.
- **Actual Results:** User receives randomly chosen score feedback. (PASS)

Test Case 2: Advanced score feedback #2

- **Purpose:** Check that the user can receive a different random score feedback from the feedback list based on the score.
- **Precondition:** User has chosen the same search result.
- **Test Procedure:**
 1. The snack score is calculated
 2. The score feedback is chosen randomly based on the score
 3. The score feedback and result are displayed
- **Expected Result:** User receives randomly chosen score feedback.
- **Actual Results:** User receives randomly chosen score feedback. (PASS)

User Story SNAK-27: Processed button**Test Case 1: Processed button #1**

- **Purpose:** Check that the snack score changes depending on whether they pick “highly processed” or not.
- **Precondition:** User has chosen a search result.
- **Test Procedure:**
 1. User chooses “highly processed”
 2. The snack score is calculated
 3. User gets 1 point less
- **Expected Result:** User gets 1 point less for picking “highly processed”.
- **Actual Results:** User gets 1 point less for picking “highly processed”. (PASS)

Test Case 2: Processed button #2

- **Purpose:** Check that the snack score changes depending on whether they pick “highly processed” or not.
- **Precondition:** User has chosen a search result.
- **Test Procedure:**
 1. User does not choose “highly processed”
 2. The snack score is calculated
 3. User gets 1 point more
- **Expected Result:** User gets 1 point more for not picking “highly processed”.
- **Actual Results:** User gets 1 point more for not picking “highly processed”. (PASS)

User Story SNAK-29: Firebase Connection

Test Case 1: Firebase Connection

- **Purpose:** Check that the user can create a Snackability account and their user information is stored.
- **Precondition:** User does not have a Snackability account
- **Test Procedure:**
 1. User clicks on “Sign Up”
 2. User submits their email and password
 3. User Firebase creates an account for the user
 4. User is sent to the Home Page
- **Expected Result:** User’s account is created in Firebase.
- **Actual Results:** User’s account is created in Firebase. (PASS)

User Story SNAK-30: Font for iOS and Android

Test Case 1: Font for iOS and Android #1

- **Purpose:** Check that an iOS user can use the application without errors.

- **Precondition:** User is on the “Sign In” page
- **Test Procedure:**
 1. User signs in
 2. The homepage renders without errors
- **Expected Result:** iOS user can use the application without errors.
- **Actual Results:** iOS user can use the application without errors. (PASS)

Test Case 2: Font for iOS and Android #2

- **Purpose:** Check that an Android user can use the application without errors.
- **Precondition:** User is on the “Sign In” page
- **Test Procedure:**
 1. User signs in
 2. The homepage renders without errors
- **Expected Result:** Android user can use the application without errors.
- **Actual Results:** Android user can use the application without errors. (PASS)

User Story SNAK-31: Move portion, units, processed from search to snack page

Test Case 1: Move portion, units, processed from search to snack page

- **Purpose:** Check that the user can search with the snack name and then get the snack score by entering the portion, units, and processed on the snack detail page.
- **Precondition:** User is on the search page
- **Test Procedure:**
 1. User searches the name of a snack
 2. User picks a search result
 3. User enters processed, units, and portion on the snack detail page

4. User calculates the score
- **Expected Result:** User gets a score after inputting units, processed, and portion on the snack detail page.
 - **Actual Results:** User gets a score after inputting units, processed, and portion on the snack detail page. (PASS)

User Story SNAK-32: Navigation bar with Home, Search, Settings

Test Case 1: Navigation bar with Home, Search, Settings #1

- **Purpose:** Check that the user can access the home page from the navigation bar
- **Precondition:** User is not on the home page
- **Test Procedure:**
 1. User clicks on the “Home” icon
 2. User is taken to the homepage
- **Expected Result:** User is taken to the home page.
- **Actual Results:** User is taken to the home page. (PASS)

Test Case 2: Navigation bar with Home, Search, Settings #2

- **Purpose:** Check that the user can access the search page from the navigation bar
- **Precondition:** User is not on the search page
- **Test Procedure:**
 1. User clicks on the “Search” icon
 2. User is taken to the search page
- **Expected Result:** User is taken to the search page.
- **Actual Results:** User is taken to the search page. (PASS)

Test Case 3: Navigation bar with Home, Search, Settings #3

- **Purpose:** Check that the user can access the settings page from the navigation bar.
- **Precondition:** User is not on the settings page
- **Test Procedure:**
 3. User clicks on the “Settings” icon
 4. User is taken to the settings page
- **Expected Result:** User is taken to the settings page.
- **Actual Results:** User is taken to the settings page. (PASS)

User Story SNAK-33: Score breakdown table**Test Case 1: Score breakdown table**

- **Purpose:** Check that the user gets a score breakdown table after they get a score.
- **Precondition:** User has chosen a search result
- **Test Procedure:**
 1. User enters the portion, processed, and units to calculate
 2. The score is calculated
 3. A table displays the score breakdown of the nutritional information
- **Expected Result:** A table is displayed with the breakdown of the snack information.
- **Actual Results:** A table is displayed with the breakdown of the snack information. (PASS)

User Story SNAK-34: Calculate average score**Test Case 1: Calculate average score**

- **Purpose:** Check that the user gets the correctly calculated average of their weekly score.

- **Precondition:** User has a consumed history
- **Test Procedure:**
 1. User consumes a snack
 2. The average score is updated
 3. The graph updates with the new average score
- **Expected Result:** The graph shows the correctly calculated average score.
- **Actual Results:** The graph shows the correctly calculated average score. (PASS)

User Story SNAK-35: Score breakdown table with visible feedback

Test Case 1: Score breakdown table with visible feedback

- **Purpose:** Check that the score breakdown table displays a column with colored icons where the user can easily dissect the feedback.
- **Precondition:** User has chosen a search result
- **Test Procedure:**
 1. User enters portion, units, and processed
 2. The score is calculated
 3. The score breakdown table shows colored icons.
- **Expected Result:** The score breakdown table shows colored icons indicating how well or badly the nutrient scored.
- **Actual Results:** The score breakdown table shows colored icons indicating how well or badly the nutrient scored. (PASS)

User Story SNAK-36: Local SQL connection to Heroku backend

Test Case 1: Local SQL connection to Heroku backend

- **Purpose:** Check that search results from the local database are returned.
- **Precondition:** User is on the search page

- **Test Procedure:**
 1. User searches a snack
 2. Heroku returns results from local database
 3. Display search results
- **Expected Result:** The user gets search results from the local database.
- **Actual Results:** The user gets search results from the local database.(PASS)

User Story SNAK-37: Highly processed button UI

Test Case 1: Highly processed button UI #1

- **Purpose:** Check that when the user enters processed, portion, and units, the “calculate” button is enabled.
- **Precondition:** User is on the snack detail page
- **Test Procedure:**
 1. User enters processed, portion, and units inputs
 2. The “calculate” button is enabled
 3. User presses calculate to get their snack score
- **Expected Result:** The user gets a snack score.
- **Actual Results:** The user gets a snack score. (PASS)

Test Case 2: Highly processed button UI #2

- **Purpose:** Check that when the user does not enter processed, portion, and units, the “calculate” button is disabled.
- **Precondition:** User is on the snack detail page
- **Test Procedure:**
 1. User leaves processed, portion, and units inputs empty
 2. The “calculate” button is disabled

- **Expected Result:** The user cannot get a snack score.
- **Actual Results:** The user cannot get a snack score. (PASS)

User Story SNAK-38: Snack detail information pop-ups

Test Case 1: Snack detail information pop-ups

- **Purpose:** Check that the user wants additional information on the score breakdown table, they can press the info icon.
- **Precondition:** User is on the snack detail page
- **Test Procedure:**
 1. User clicks on info icon by a nutrient on the score breakdown table
 2. A tooltip shows up with information on the chosen tooltip
 3. User presses outside of the tooltip to exit it
- **Expected Result:** The user gets additional information from the tooltip.
- **Actual Results:** The user gets additional information from the tooltip. (PASS)

User Story SNAK-39: Delete snacks in consumed history

Test Case 1: Delete snacks in consumed history

- **Purpose:** Check that the user can delete a snack from their consumed history.
- **Precondition:** User is on the home page
- **Test Procedure:**
 1. User goes to the consumed history section
 2. User holds down on the snack they want to delete
 3. User selects “delete” on the message pop-up
 4. The snack is deleted
 5. The graph is updated with their new average score

- **Expected Result:** The user can delete a snack from their consumed history.
- **Actual Results:** The user can delete a snack from their consumed history. (PASS)

User Story SNAK-41: Highly Processed information pop-up

Test Case 1: Highly Processed information pop-up

- **Purpose:** Check that the user wants additional information on the “highly processed” button, they can press the info icon.
- **Precondition:** User is on the snack detail page
- **Test Procedure:**
 1. User clicks on info icon by the “highly processed” button
 2. A tooltip shows up with information on “highly processed”
 3. User presses outside of the tooltip to exit it
- **Expected Result:** The user gets additional information from the tooltip.
- **Actual Results:** The user gets additional information from the tooltip. (PASS)

User Story SNAK-44: Points and leveling system

Test Case 1: Points and leveling system #1

- **Purpose:** Check that when the user consumes a snack with a score less than 6, they have no points added to their total.
- **Precondition:** User is on the snack detail page
- **Test Procedure:**
 1. User enters the processed, units, and portion
 2. The snack score is calculated
 3. User gets 0 points added to their total points
- **Expected Result:** The user gets 0 points added to their total points and does not move levels.

- **Actual Results:** The user gets 0 points added to their total points and does not move levels. (PASS)

Test Case 2: Points and leveling system #2

- **Purpose:** Check that when the user consumes a snack with a score from 6 to 7, they have 0.5 points added to their total.
- **Precondition:** User is on the snack detail page
- **Test Procedure:**
 1. User enters the processed, units, and portion
 2. The snack score is calculated
 3. User gets 0.5 points added to their total points
- **Expected Result:** The user gets 0.5 points added to their total points and does move levels.
- **Actual Results:** The user gets 0.5 points added to their total points and does move levels. (PASS)

Test Case 3: Points and leveling system #3

- **Purpose:** Check that when the user consumes a snack with a score from 8 to 9, they have 1 point added to their total.
- **Precondition:** User is on the snack detail page
- **Test Procedure:**
 1. User enters the processed, units, and portion
 2. The snack score is calculated
 3. User gets 1 point added to their total points
- **Expected Result:** The user gets 1 point added to their total points and does move levels.
- **Actual Results:** The user gets 1 point added to their total points and does move levels. (PASS)

Test Case 4: Points and leveling system #4

- **Purpose:** Check that when the user consumes a snack with a score from 10 to 11, they have 2 points added to their total.
- **Precondition:** User is on the snack detail page
- **Test Procedure:**
 1. User enters the processed, units, and portion
 2. The snack score is calculated
 3. User gets 2 points added to their total points
- **Expected Result:** The user gets 2 points added to their total points and does move levels.
- **Actual Results:** The user gets 2 points added to their total points and does move levels. (PASS)

User Story SNAK-45: Progress Bar | Visual Representation of scores**Test Case 1: Visual Representation of scores**

- **Purpose:** As a user I want to see my score visual within the app to easily see my progress or improvements made.
- **Precondition:** User is logged in and going to consume a snack.
- **Test Procedure:**
 1. User is logged in
 2. User presses consume on snack that gives ____ points
 3. User goes to home page
 4. Application displays via Progress Bar the users updated progress
- **Expected Result:** Progress bar updates once a snack is consumed
- **Actual Results:** Progress bar updates once a snack is consumed (PASS)

User Story SNAK-46: Double points for weekends and Holidays

Test Case 1: Double points for weekends and Holidays

- **Purpose:** As a user I want some incentive during the weekends and holidays to track my progress
- **Precondition:** User must consume a snack during a weekend or holiday
- **Test Procedure:**
 5. Go to home page
 6. System queries Firebase for historical data
 7. System checks if snacks were consumed during holidays or weekends
 8. System displays total points
- **Expected Result:** during a weekend or holiday Consumed snacks result in double point awarded
- **Actual Results:** during a weekend or holiday Consumed snacks result in double point awarded (PASS)

User Story SNAK-48: Restructure 'Settings' page to list**Test Case 1:** Restructure 'Settings' page to list

- **Purpose:** As a user I want to easily navigate the 'Settings' page in the application
- **Precondition:** User must be in the settings page within the app.
- **Test Procedure:**
 9. After clicking the settings button in the navigation options in the app
 10. The settings page
- **Expected Result:** Upon hitting 'Settings' from nav bar i can see the other pages listed in a list format
- **Actual Results:** Upon hitting 'Settings' from nav bar i can see the other pages listed in a list format (PASS)

User Story SNAK-49: Specify Local/USDA data on the results page

Test Case 1: Specify Local/USDA data on the results page

- **Purpose:** As a user I want to know where the the snack information is coming from: USDA/Local database.
- **Precondition:** User must search for a snack that is within both databases
- **Test Procedure:**
 11. After hitting search and searching for a snack
 12. The results are separated based on the the USDA or local database
- **Expected Result:** Upon hitting search I can see the results separated based on the the USDA or local database
- **Actual Results:** Upon hitting search I can see the results separated based on the the USDA or local database (PASS)

User Story SNAK-50: Feedback to let user know the consumed button has been pressed to avoid over consumption of snacks.

Test Case 1: Feedback to let user know the consumed button has been pressed to avoid over consumption of snacks.

- **Purpose:** As a user I want to click on consumed and know that the app has registered my choice.
- **Precondition:** User must click on the 'consumed' button from within the snack's 'snack detail' page in the application.
- **Test Procedure:**
 13. After searching for a snack and within the snack's page user must hit 'consume' button
 14. The application needs to give feedback once the consume button is pressed
 15. The snack is added onto the users consumed history.
- **Expected Result:** Upon hitting the consumed button from the snacks page the user gets a feedback from the application indicating a successful button press.

- **Actual Results:** Upon hitting the consumed button from the snacks page the user gets a feedback from the application indicating a successful button press.
- (PASS)

User Story SNAK-51: Clicking on consumed history takes you to snack page

Test Case 1: Clicking on consumed history takes you to snack page

- **Purpose:** As a user I want to click on a snack within my consumed history and access that snack's specific page in the application.
- **Precondition:** User must click on the snack from within the 'history' page
- **Test Procedure:**
 16. User goes to the 'history' page
 17. The application queries firebase to return the list of recently consumed snack to the user.
 18. User clicks on a previously consumes snack from within the 'history' page.
 19. Application redirects the user to the snack detail page for the snack selected.
- **Expected Result:** Upon hitting a snack from within the 'history page' I am routed to the snack's detail page within the application allowing me to have the same options as if i searched from the 'search' page.
- **Actual Results:** Upon hitting a snack from within the 'history page' I am routed to the snack's detail page within the application allowing me to have the same options as if i searched from the 'search' page. (PASS)

User Story SNAK-52: Crashes when searching garbage

Test Case 1: Crashes when searching garbage

- **Purpose:** As a user I would like to see no results when the application database doesn't have the snack I am searching for.

- **Precondition:** User must search a snack not in the database or have a spelling error within their input
- **Test Procedure:**
 - 20. User goes to the 'search' page
 - 21. User inputs random letters or text that doesn't equate a snack within the USDA or local database 'garbage'
 - 22. App must return no results to the user.
- **Expected Result:** Upon hitting search of random letters or text that doesn't equate a snack within the USDA or local database 'garbage' I will see 'no results' returned as a user
- **Actual Results:** Upon hitting search of random letters or text that doesn't equate a snack within the USDA or local database 'garbage' I will see 'no results' returned as a user (PASS)

User Story SNAK-53: Update snack database

Test Case 1: Update snack database

- **Purpose:** As a user I would like a way to search for raw fruits and vegetables
- **Precondition:** User must navigate to "search" page in the application
- **Test Procedure:**
 - 23. User goes to the 'search' page
 - 24. User inputs 'raw ____' where ____ can be a fruit or vegetable
 - 25. App returns the 'raw ____' as a snack for the user to consume via results.
- **Expected Result:** Upon hitting search of 'raw fruit/vegetable' on the search page the app returns the raw fruit/vegetable submitted
- **Actual Results:** Upon hitting search of 'raw fruit/vegetable' on the search page the app returns the raw fruit/vegetable submitted (PASS)

User Story SNAK-54: Update about page with contact info**Test Case 1: Update about page with contact info**

- **Purpose:** As a user I would like a way to contact the Admins or developers of the application.
- **Precondition:** User must navigate to “about” page from the settings page
- **Test Procedure:**
 - 26. User goes to the ‘Settings’ page
 - 27. Application view shows options
 - 28. User clicks ‘about us’ option
 - 4. Contact info shows in application of the developers in the application
- **Expected Result:** Upon going to the about us page, the user can see info on the app and its developers.
- **Actual Results:** The user sees info on the app about the developers upon going to the about us page. (PASS)

GLOSSARY

API: Application Program Interface. A set of subroutine definitions, communication protocols, and tools for building software.

Highly Processed: This refers to foods that have been heavily modified from their original form, with addition of salt, sugar, fat, and/or food additives (substances added to food to preserve flavor or enhance its taste, appearance, or other qualities).

JSON: Javascript Object Notation. An open-standard file format that uses human-readable text to transmit data objects consisting of attribute–value pairs and array data types (or any other serializable value).

MVC: Model-View-Controller. An architectural pattern commonly used for developing user interfaces that divides an application into three interconnected parts.

REST: Representational State Transfer. A software architectural style that defines a set of constraints to be used for creating web services.

Snack Score: A score given by Snackability that represents how healthy a snack is. The score is developed from the nutrients of the snack and the USDA recommendations.

UI: User Interface. The means by which the user and a computer system interact, in particular the use of input devices and software.

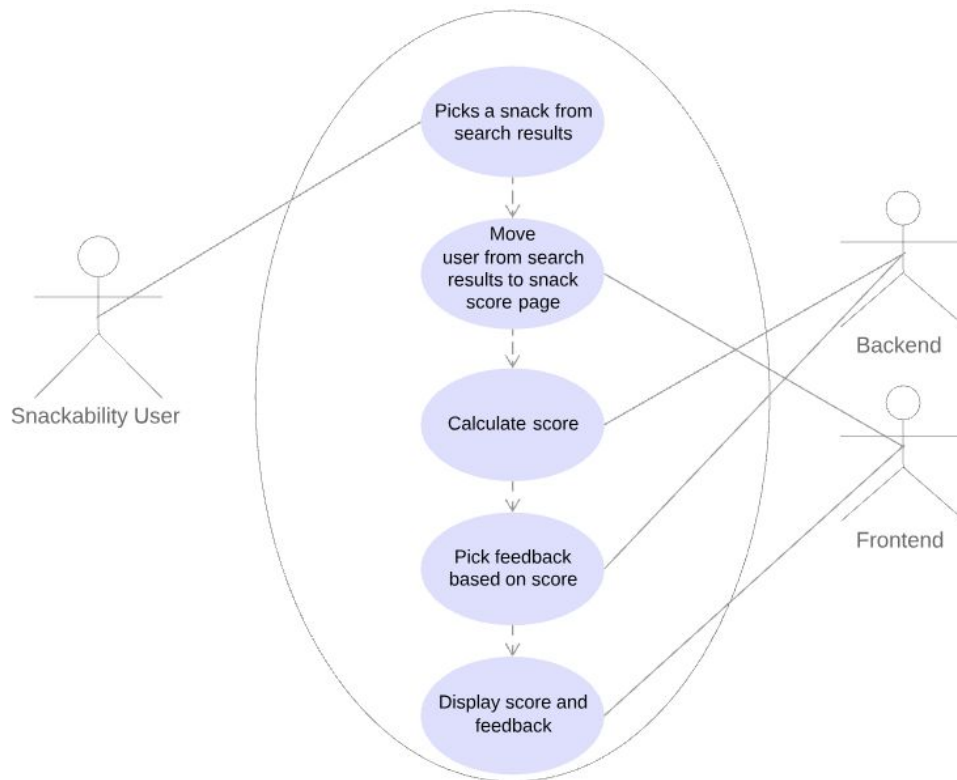
UML: Unified Modeling Language: A general-purpose, developmental, modeling language in the field of software engineering, that is intended to provide a standard way to visualize the design of a system.

APPENDIX

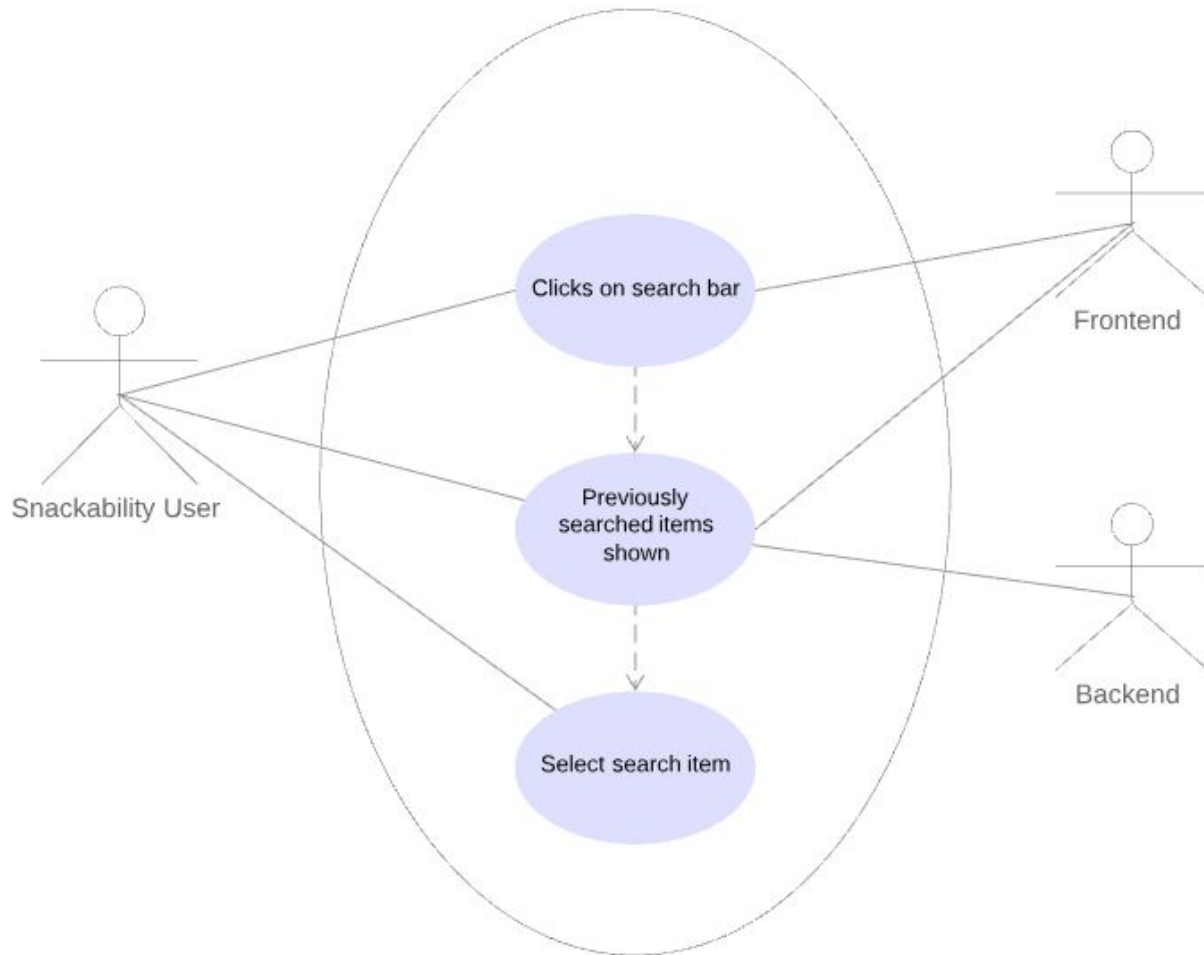
Appendix A - UML Diagrams

Use Case Diagrams

User Story SNAK-16: Advanced nutrition feedback

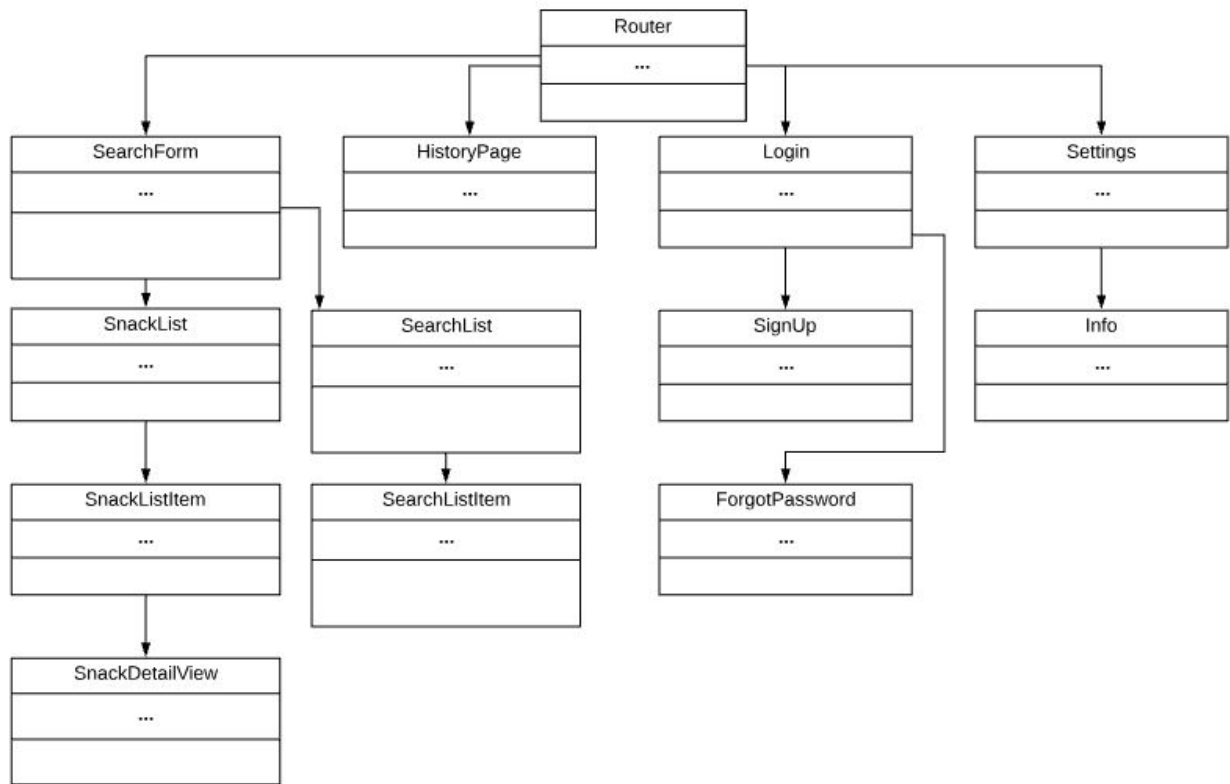


User Story SNAK-22: Search history



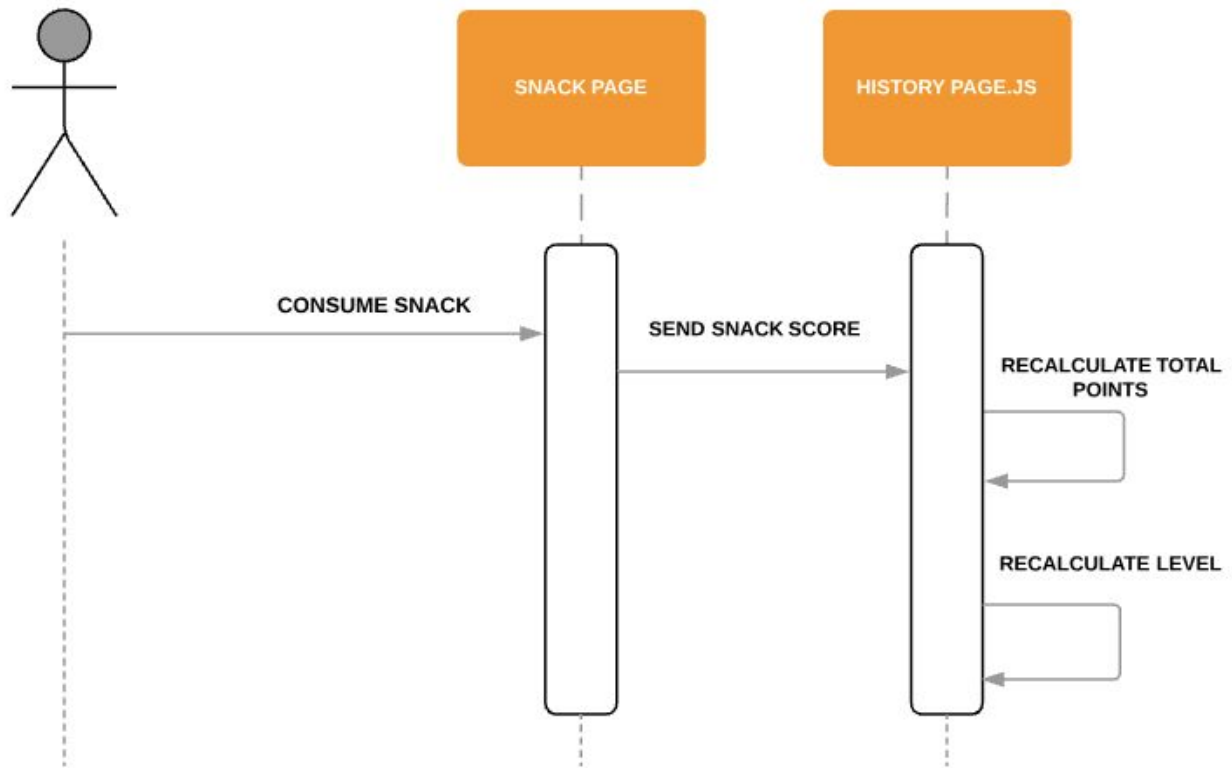
Class Diagram

Frontend Class Diagram

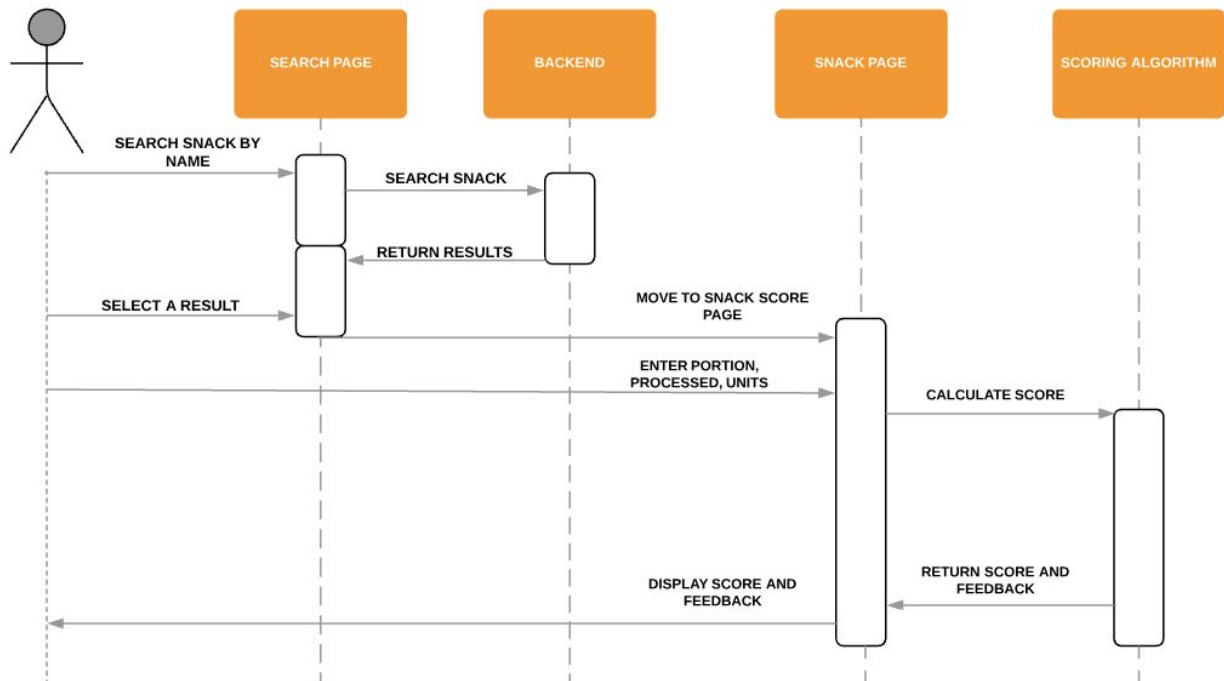


Sequence Diagram

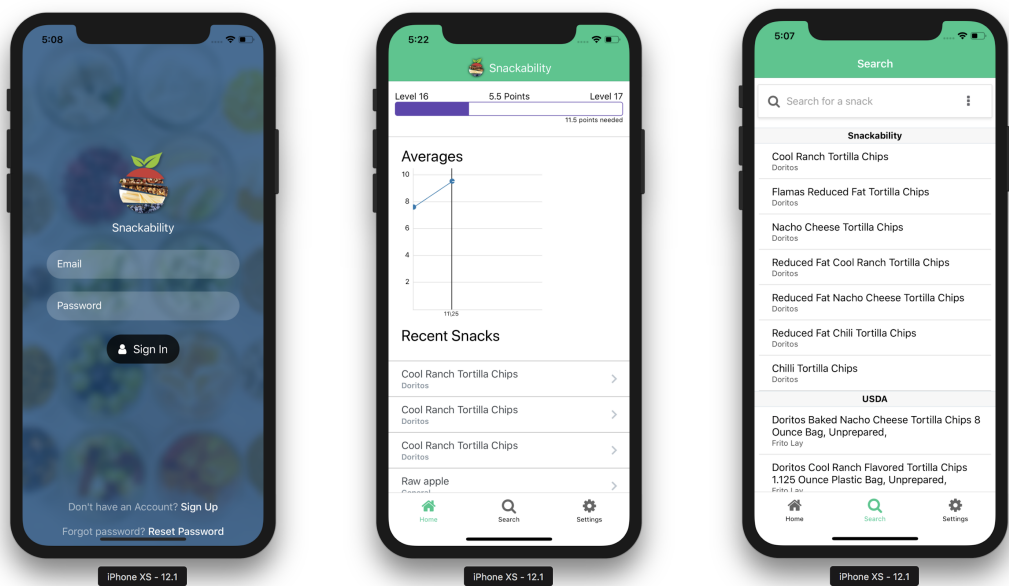
User Story SNAK-44: Points and Leveling System



User Story SNAK-31: Move portion. Units, processed from search to snack page



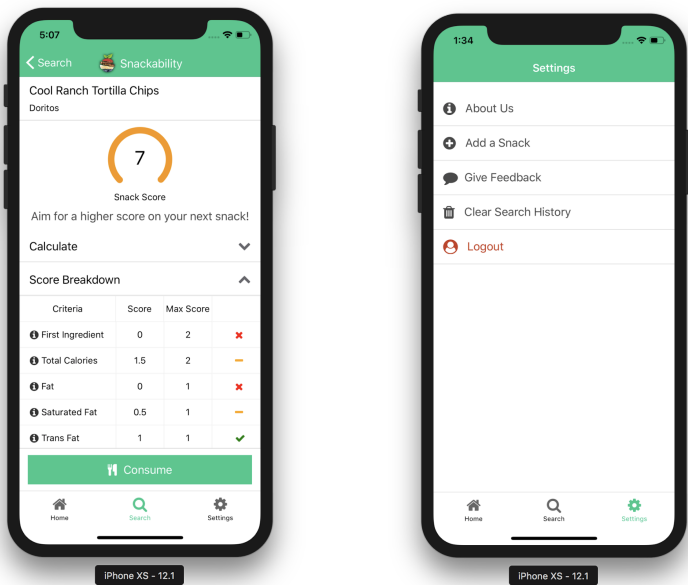
Appendix B - User Interface Design



Login Page

Home Page

Search Page



Snack Detail Page

Settings Page

Appendix C - Sprint Review Reports

Sprint 1

Attendees: Carolina Karthik, Christian Canizares, Adrian Serrano, Shafeeqe Khan, Jessiel Benitez, Jeffrey Antoine

Start time: 12:15 PM

End time: 1:15 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story SNAK-5 “Scaling score from 0 to 10”
- User Story SNAK-9 “Find how score is calculated in the code”
- User Story SNAK-10 “Prioritize local database”
- User Story SNAK-17 “Email user feedback”
- User Story SNAK-19 “Fix fonts”

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

- Not applicable to any of the user stories for this sprint

Sprint 2

Attendees: Carolina Karthik, Christian Canizares, Adrian Serrano, Shafeeqe Khan, Jessiel Benitez, Jeffrey Antoine

Start time: 12:30 PM

End time: 1:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story SNAK-8 - Remove nutrition facts
- User Story SNAK-11 - Keeping the search history
- User Story SNAK-14 - Fix "consume" button
- User Story SNAK-22 - Portioning
- User Story SNAK-27 - Processed button
- User Story SNAK-28 - Env file for IP address

- User Story SNAK-29 - Add Firebase connection

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- User Story SNAK-7 - Show how score is calculated
 - This story is to be improved upon because she gave new specifications on what she would like the snack detail page to look like.
- User Story SNAK-15 - Consumed history
 - Once the feature is completed, the graph representation story can worked on.
- User Story SNAK-25 - Deploy to heroku instance
 - Developer is receiving an error when trying to deploy to heroku instance

Sprint 3

Attendees: Carolina Karthik, Christian Canizares, Adrian Serrano, Shafeeque Khan, Jessiel Benitez, Jeffrey Antoine

Start time: 12:30 PM

End time: 1:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story SNAK-15 - Consumed history
- User Story SNAK-23 - Score/graph/representation of weekly average of scores
- User Story SNAK-25 - Deploy to heroku instance
- User Story SNAK-30 - One Font For iOS & Android
- User Story SNAK-31 - Separate portion, units, and processed into another page
- User Story SNAK-32 - Rearrange UI bar to HOME - SEARCH - SETTINGS and move info to settings
- User Story SNAK-33 - Table for score breakdown
- User Story SNAK-34 - Calculate average of scores (just a number)
- User Story SNAK-36 - Connecting SQL to heroku backend

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- User Story SNAK-13 - Advanced Search
 - This story was put on hold until the Product Owner and the team discussed the necessity of this feature. It will be completed in the following sprint
- User Story SNAK-16 - Advanced nutrition feedback
 - This story required the Product Owner to send a list of all possible feedback messages to display to the user. This was sent to the team at the end of the sprint so it will be completed in the following sprint
- User Story SNAK-35 - Score breakdown table rows dynamically change color
 - This story was not finished in this sprint and will be completed in the following sprint

Sprint 4

Attendees: Carolina Karthik, Christian Canizares, Adrian Serrano, Shafeeqe Khan, Jessiel Benitez, Jeffrey Antoine

Start time: 12:30 PM

End time: 1:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User story SNAK-16 - Advanced nutrition feedback
- User story SNAK-24 - Add a snack
- User story SNAK-26 - Advanced score feedback
- User story SNAK-35 - Score breakdown table rows dynamically change color
- User story SNAK-37 - Highly processed button (yes/no)
- User story SNAK-38 - Snack Detail Redesign info popups
- User story SNAK-41 - Info for highly processed button
- User story SNAK-42 - outline for logo

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting:

- User story SNAK-13 - Advanced Search
 - This story was not finished in this sprint and will be completed in the following sprint
- User story SNAK-39 - Editing consumed history to delete snacks

- This story was not finished in this sprint and will be completed in the following sprint
- User story SNAK-40 - UI of consumed history for home page
 - This story was not finished in this sprint and will be completed in the following sprint
- User story SNAK-43 - preview of what average score would look like before consuming
 - This story was not finished in this sprint and will be completed in the following sprint

Sprint 5

Attendees: Carolina Karthik, Christian Canizares, Adrian Serrano, Shafeeque Khan, Jessiel Benitez, Jeffrey Antoine

Start time: 12:30 PM

End time: 1:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User story SNAK-13 - Advanced Search
- User story SNAK-39 - Editing consumed history to delete snacks
- User story SNAK-44 - Assign levels up to 100
- User story SNAK-45 - progress bar
- User story SNAK-46 - Double points for holidays and weekends
- User story SNAK-47 - Update and merge score breakdown table icons
- User story SNAK-48 - Restructure settings page to list

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting:

- User story SNAK-40 - UI of consumed history for home page
 - There are a few more changes to make that will continue into the next sprint
- User story SNAK-43 - preview of what average score would look like before consuming
 - There are some problems with the front end for this story that will be resolved in the next sprint

Sprint 6

Attendees: Carolina Karthik, Christian Canizares, Adrian Serrano, Shafeeque Khan, Jessiel Benitez, Jeffrey Antoine

Start time: 12:30 PM

End time: 1:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story SNAK-49 - Specify local/usda results on search page
- User Story SNAK-50 - Indication message of consumption
- User Story SNAK-51 - Clicking on consumed history takes you to snack page
- User Story SNAK-52 - Crashes when searching garbage
- User Story SNAK-53 - Update snack database
- User Story SNAK-54 - Update about page with contact info

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- User Story SNAK-40 - UI of consumed history for home page
 - This story was never completed as it was continuous UI changes
- User Story SNAK-43 - preview of what average score would look like before consuming
 - This story was never completed because the developers could figure out how to do it in the allotted time

Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wishlist Document and other documents

USER MANUAL

Installation

1. Download the Expo app from the Google Play Store (Android) or the Apple App Store (iOS).
2. Within Expo, tap on the “Sign in to your account” button.
3. Use E-Mail: Snackabilityapp@gmail.com and password: snackabilityVIP2018
4. Under the “Published Projects” section tap “Snackability” to launch the app.

Signing In to Snackability

1. Press “Sign Up” at the bottom
2. Enter your E-Mail address and Password
3. Press “Sign Up” button

Navigation Bar Pages

The navigation bar at the bottom allows you to move through different pages in the app.

Pages

- The Home page contains information about your current level, your average consumed snack score, and your recently consumed snacks.
- The Search page lets you search for a snack
- The Settings page has buttons for adding a snack, giving application feedback, clearing the search history, and logging out.

Searching for a Snack

In the search page, the bar at the top is used for searching. Simply tap the bar and enter the name of the snack you are looking for and press search. The “recent searches” section shows previously searched items. Once a snack is searched, a list of items is shown. Simply choose your desired item.

Advanced Search

If the desired item cannot be found with a regular search, the advanced search option can be found by tapping on the three vertical dots on the search bar. The advanced search allows you to specify the snacks brand and the snacks name. Once the information is entered simply press the search button.

Calculating a Snack Score

Once a snack has been searched and found, you may calculate the snack score. On the snack detail page, under the “Calculate” section, enter the portion, the units of your serving and whether the item is highly processed or not. If you are unsure about what highly processed means tap the “i” icon for more information. Once entered you may press the “Calculate” button to get a snack score. Note: The calculate button will be disabled until all the fields are populated.

Interpreting your Snackability Snack Score

Once a snack score has been calculated, information about how the score was calculated can be found under the “Score Breakdown” section. For information about what any of the criterion mean tap on the “i” next to the criterion. More feedback is provided under the snack score and under the score breakdown table.

Consuming a Snack

After you have calculated the score for a snack that you would like to consume, press the consume button to indicate that you have consumed the snack. In the home page, you will see the consumed snack reflected on the “Recent Snacks” section.

Deleting a recently consumed snack

If you have accidentally indicated that you have consumed a snack that you didn’t consume you may delete it. To delete a recently consumed snack hold and tap on it. A pop up box will allow you to confirm the action.

SHORTCOMINGS AND WISHLIST

We were not able to complete the following stories in Snackability 2.0. These stories would greatly improve the user experience if added in the next iteration of the application:

User Story SNAK-12: Search with barcode scanning

As a developer, I want to add searching by scanning the barcode of a snack so the user does not have to search manually.

- Acceptance Criteria:
GIVEN I am a user with a snack in my hand
WHEN I use the scanning option
THEN I get the correct snack information page.

User Story SNAK-18: Reminder notifications

As a developer, I want to have reminder notifications on the app to remind the user to use it every day.

- Acceptance Criteria:
GIVEN I am a frequent user
WHEN I have not logged into the application in two days
THEN I receive a notification that reminds me to input snacks.

User Story SNAK-20: Build app for iOS

As a developer, I want to build the application for iOS so it could be deployed to the Apple App Store.

- Acceptance Criteria:
GIVEN I am a user who wants to download Snackability app
WHEN I go to the Apple App store
THEN I can download the Snackability app.

User Story SNAK-21: Build app for Android

As a developer, I want to build the application for Android so it could be deployed to the Google Play Store.

- Acceptance Criteria:
GIVEN I am a user who wants to download Snackability app
WHEN I go to the Google Play Store
THEN I can download the Snackability app.

User Story SNAK-40: Consumed History UI

As a developer, I want to further develop the consumed history on the homepage to make it more user friendly.

- Acceptance Criteria:
GIVEN I am a user on the home page
WHEN I go to the consumed history
THEN I can use it easily because it is user friendly.

User Story SNAK-43: Preview of average score before consuming

As a developer, I want to show a preview of what the user's average snack score of the day would look like before consuming.

- Acceptance Criteria:
GIVEN I am a user who selected a search result

WHEN I calculate my snack score

THEN I am shown a preview of what my average score would look like if I were to consume it.

References

Firebase:

<https://firebase.google.com/docs/database/>

Draw IO:

<https://www.draw.io/>

Lucid Chart:

<https://lucidchart.com>

Screencast-o-matic:

<https://screencast-o-matic.com/screen-recorder>

Heroku Documentation:

<https://devcenter.heroku.com/categories/reference>

React Native Documentation:

<https://facebook.github.io/react-native/>

Expo Documentation:

<https://docs.expo.io/versions/latest/>