Project One:

Team Members
Colin Whitcomb
Gabriel Sucich
Jenifer Jayme
Kaitlin Hulsy

Your Project Links

Code Repo URL

https://github.com/GabeSucich/Project1_CGJK

Deployed URL

https://gabesucich.github.io/Project1_CGJK/

Link to Github Projects Board

https://github.com/GabeSucich/Project1_CGJK/projects/1

Project Week Timeline

Date	Subject
Fri 06/26	Project One Introduction
Sat 06/27	Project Work (**No Class**)
Sun 06/28	Project Work (**No Class**)
Mon 06/29	Project Work
Tues 06/30	Project Work || Proposal Due
Wed 07/01	Project Work
Thurs 07/02	Project Work & Mock Presentations
Fri 07/03	Project Work (**No Class**) - Related to July 4th
Sat 07/04	Project Work (**No Class**)
Sun 07/05	Project Work (**No Class**)
Mon 07/06	Project Presentations - **BCS Submission DUE**
Tues 07/07	Mock Interview Day **- Blog & Retrospection Notes DUE**
Wed 07/01	**Updated Portfolio Due**

Project Requirements

- Must use at least two APIs
- Must use AJAX to pull data
- Must utilize at least one new library or technology that we haven't discussed
- Must have a polished frontend / UI
- Must meet good quality coding standards (indentation, scoping, naming)
- Must NOT use alerts, confirms, or prompts (look into modals!)
- Must have some sort of repeating element (table, columns, etc)
- Must use a CSS framework other than Bootstrap

- Must be Deployed (GitHub Pages)
- Must have User Input Validation
- Must be interactive (i.e: accept and respond to user input)
- Must have some repeating element / table

Presentation Requirements

- You will also be responsible for preparing a 10 minute presentation.
- This will be a formal presentation.
- Every member of the group is expected to be able to speak to any section of the application
- One in which you explain in detail:
    - Your overall application's concept
    - The motivation for its development
    - Your design process
    - The technologies you used (and briefly how they work)
    - A demonstration of its functionality
    - Directions for future development

Teamwork & Expectations

Team Effort

Before anything, remember that Projects are a group effort: Working closely with your teammates is a requirement. This both helps teach real-world collaborative workflows, and enables you to tackle more difficult problems than you'd be able to working alone.

In other words, working in groups allows you to work smart and dream big. Take advantage of it!

Note:

- 100% participation and attendance is required in order to receive credit for the Group Project work.
- Participation in all 3 Projects are course requirements.

How will you be using Git?

Git Commit #

20/person = 80 total

Stuck time (How long until you consult a teammate?):

- 30 min

Working Agreements:

Examine these resources:

- http://www.iliokb.com/2012/04/example-working-agreement.html
- http://www.payton-consulting.com/wp-content/uploads/2014/07/WorkingAgreements.jpg

EXAMPLE:

During Sprint Do The Following:

- Team members attend daily stand ups @ 9:40am (M-f))
- We will be expected to be working on Sat, Sun, M-F (is this true?)
- Should a team member have a conflict, s/he/they updates the slack team channel in advance of the meeting
- We believe in in the value of planning collectively as a team
- Every Team Member is engaged and involved.
- Every Team Member is committed to the value of the application over individual recognition.
- Every Team Member will practice active listening
- Every Team Member will be directly engaged with the work - not answer texts or phone calls, social media, or other off-topic material (with the exception of emergencies -which will be communicated to the team)

Your Team Working Agreements

During Sprint Do The Following:

- Team members attend daily huddles @ 9:40am and 3:30pm (M-F)
- We will be working on the project every day including weekend days.
- Should a team member have a conflict, update the team by slack in advance and attempt to reschedule so all can be present.
- We believe in in the value of planning collectively as a team
- Every Team Member is engaged and involved.
- Every Team Member is committed to the value of the application over individual recognition.
- Every Team Member will practice active listening.
- Disagreement is necessary and fine, and should always be done graciously and respectfully.
- Every Team Member will be directly engaged with the work - not answer texts or phone calls, social media, or other off-topic material (with the exception of emergencies -which will be communicated to the team)

Agile Stand-Ups:

For software teams, the stand-up is like the team's huddle. It's even commonly known as the daily scrum, and reinforces "we" to keep everyone aware of the team's landscape and progress.

A stand-up is a daily meeting that involves the core team.
This meeting's flavor is unique to each team, but commonly we use three simple questions to generate structure:

1. What did I work on yesterday?
2. What am I working on today?
3. What issues are blocking me?

These questions highlight progress and help flag team blockers. Also, it strengthens the team when everyone shares the progress they're contributing to the team. The daily reinforcement of sharing individual successes and plans keeps everyone excited about the team's overall progress.

Agile Playbacks:

Teams sometimes have a weekly/bi-weekly meeting called a "playback". This meeting allows team members to explain and demo completed features and the work they did during this past sprint. (normally a work week). Playbacks are important to keep project managers up to date, as well other team members to what has been accomplished during a sprint.

The playback format:

- Tell your team what you worked on and how it went.
- Show the progress you have made on your work.
- Demo any finished work/features.

We encourage you to set up time for playbacks at the end of each sprint.

Application Info:

Project Title:

Project Miles
(play on Miles Morales-Spiderman in alternate Marvel universe, and the fact that it's a health app. I am not married to the idea).

Logo Image

Project Description

Before you start writing any code, your group should outline the scope and purpose of your project. This helps provide direction and prevent scope creep.

Write this as a brief summary of your interests and intent, including:

- Problem / Motivation
- How it addresses the problem
- Possible source for such data

Project Abstract

The nutritional bad guys have all the money at their disposal. McDonald's and the like have marketing budgets that are endless, and it's all going to roping in kids. How do we fight the fast food industry when it comes to kids? is the reality of not knowing when one's next meal is coming. This puts an unfair burden on children when their families are one paycheck away from being homeless. Add to that the reality of the "villains" of fast food restaurants and a litany of misinformation disseminated by marketing and the internet, and you have a recipe for preventable diseases such as depression, diabetes, etc.

How Our App Will Address The Problem:

Comic book characters have long been implemented in education as a means of reducing the affective filter of young learners. We know that our target audience of teenagers have brains which are still forming. Combining active learning and the love of gaming, Project Miles will provide a means of providing education in a way that is interactive and fun. Incorporating characters from the Marvel Universe, interesting destinations that require physical activity(much along the lines of Pokemon Go), we will market and model good nutrition and exercise in a way that is interactive and concise. Included is a link to a fleshed-out stretch goal iteration (lightyears beyond a prototype) of what kids can do, in addition to links to research conducted by the group. The central thesis here is that kids will exercise and read factually accurate nutritional information if it is fun and user-friendly.

For Version 1, we plan to use the following APIs:

- Zomato to return menus of local restaurants

- Google Maps to return distances between start and end points
- Marvel / Avengers characters

We may also include other APIs may include
- calories consumed based on food inputs
- health menu item choices from local fast food restaurants
- Calories burned using X exercise (walk, bike, skateboard) on distance from one location to another
- Mental health tips or symptoms - potential diagnosis - recommended actions?

Opportunities for future expansion could include presenting Avengers characters with challenges which they can better face with improved health, and pitting one character against another in friendly competitions.

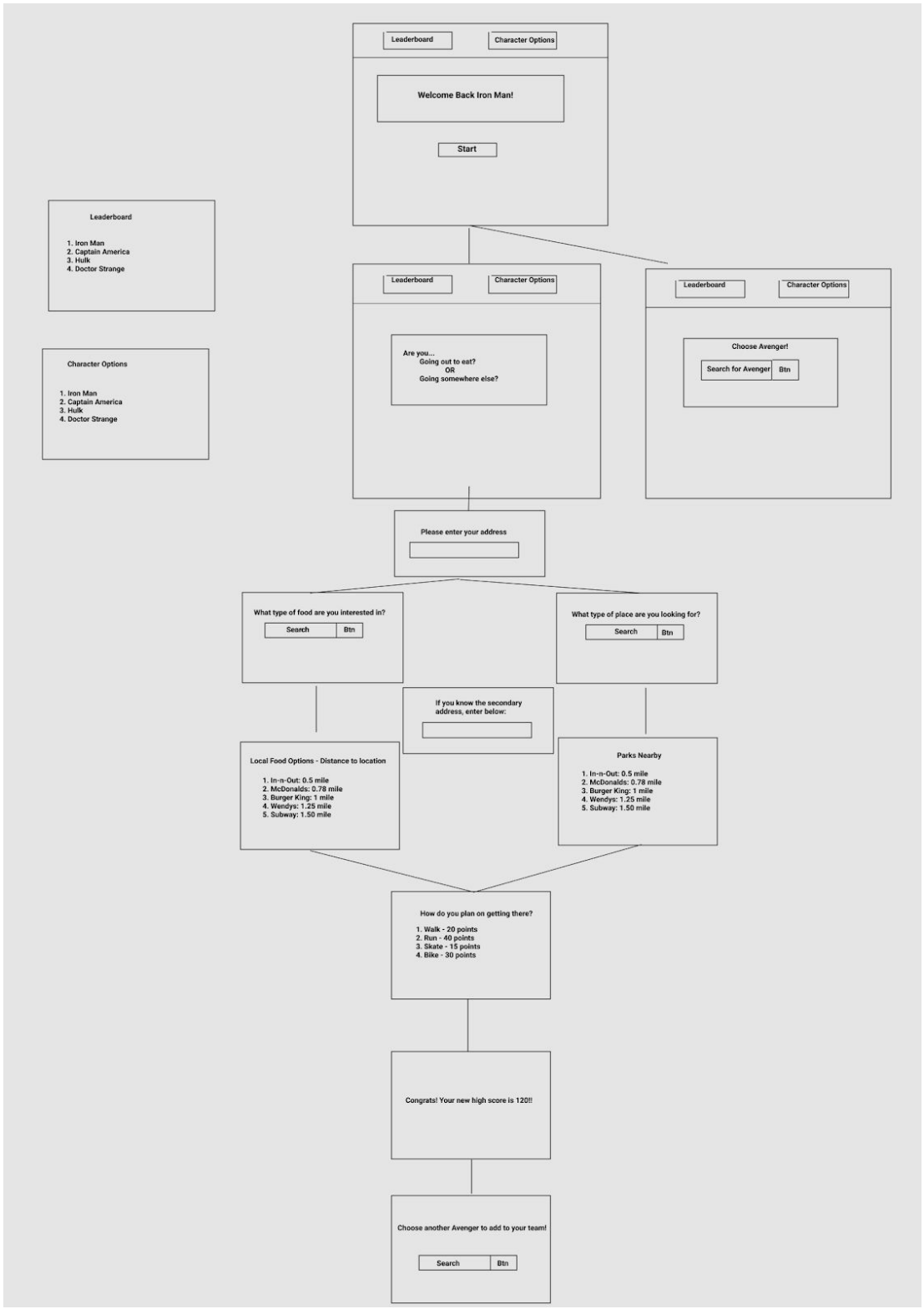
Planning: Design

User Stories

As a <role>,	I want <feature>	so that <reason>.
1 young person*	to search the distance to a secondary location	I can plan how to get there accordingly (i.e. by foot, skateboard, bike)
2 young person*	to see restaurant options in my area	I can choose the location with my favorite food.
3 young person*	to feel rewarded for engaging in physical activity/eating healthier foods	I receive a sense of accomplishment.

** Our target demographic is people of all gender identities between ages 11-15. We choose the general term "young person" to avoid genderizing the user.*

WireFrame Images



MVP Goals

Task	Owner	Completed?
1 Landing Page: Attractive user-friendly page with clear purpose and ways to interact. (Front End)	Kait & Jen	To Be Started
2 User Profile: User can choose a Marvel character as their avatar. Local storage will determine new user vs returning user, and DOM will greet them with their identity in text with simple image(s).	Kait	To Be Started
3 User Interactivity: Input and Outputs work. App will give recommended healthy choices in exercise and diet that are local and directly relevant to users, and give users the chance to earn points through their choices. Points translate to superpower badges.	Jen	Started
4 API Information queries and processing works:	Gabe & Colin	

Stretch Goals

Task
1 Leaderboard shows progress of users
2 Increased activity enables unlocking of new characters
3 Users are presented with challenges for their characters

Rough Breakdown of Tasks

I. Planning Stage: Define the app.

1. Identify criteria for consensus on the project, and create a rubric for evaluating ideas. In our case the we reached general consensus on the following qualities we'd like our project to have:
    - Something socially beneficial, which can include health/wellness, services to marginalized groups, or support for unemployment and career-building

- Ready environment with relevant existing APIs to support the project
- Something that will benefit our professional portfolios

2. Brainstorm ideas, assess them against the rubric and elect one concept to move forward.

3. Finalize the written project concept description (v1) including:
    - target user demographic
    - app purpose
    - planned API and library components
    - rough diagram of relationships between inputs, lookups and outputs
    - MVP (minimum viable product) and stretch goals
    - If for commercial market, research the competitive landscape

II. Blueprint Stage

1. Map out the User Journey. Roughly storyboard the series of prompts or pages the user will encounter, and the input, lookup and output functions needed. This will inform the wireframes and begin to generate a task list.

2. Create Wireframes. Use the user journey storyboard to sketch visuals for each step in the user journey.

3. Plan Code Structure. Translate the user journey into coding structure decisions such as which steps will be achieved through DOM manipulation versus which will need a separate html doc. Determine how many html pages will be needed. Organize major functions into discrete modules that can be worked on somewhat independently.

4. First Building Blocks. Pinpoint the first basic function(s) to build as a starting point from which other functions can be developed.

5. Revisit the project plan. Are revisions needed in view of emerging successes, ideas and challenges? If so, make necessary course corrections.

III. Framing Stage

1) First Function(s). Build and test the simplest Input-to-API Lookup-to-Output function(s). We'll start with the exercise-for-points function.
    a) Prompt user to input their start location, choice of destination and mode of travel (walk, bike, skateboard). Save preferred destinations like Home and Friend's House to user object for future reference.
    b) Query Google maps API for distance between start and end points.

c) Calculate calories burned based on user's mode of travel.
d) Translate number of calories burned into points to add to score.

2) User Account. Build and test the user account function so that users can choose a Marvel character as an avatar (using the Marvel API), and local storage will remember their character name and recognize them as a returning (or new) user.

3) Prototype the Landing Page and Prompt Pages. Produce and test a basic styled index.html with a table, divs, and the above functions and get it to a working demo stage.

IV. Construction

1. Style and polish user input prompt modals.

2. Build out user-friendliness and design of landing page, including text, directions and graphics.

3. Build Meal/Nutrition Function. Add the meal-centered function that enables users to view menus at local restaurants, find healthy options and calculate calories burned travelling to and from by walk, bike, etc..

4. If time permits, consider building stretch goal functions.

V. Beta Test

1. Engage at least three external parties who fit the target demographic to road test the app and provide feedback.
2. Decide on what fixes and improvements to incorporate into version 1, and which enhancements to stage for version 2. Make any version 1 improvements.
3. Finalize the Readme file.

VI. Release Version 1

1. Ensure all final work is incorporated into master and link deploys properly.
2. Write and publish a blog post about the app and include the deployed link as a demo.
3. Finalize and publish instructions for feedback and collaboration on code.

Planning: Technology

APIs to be Used:

API #	API Base URL	Paraments	What is it?
1	api.tomtom.com/search/2/structuredGeocode.JSON	Country Code, street number, street name, city, state, postal code	A geocoder which takes an address and returns the coordinates
2	developers.zomato.com/api/v2.1/search	Q (search query), lat/lon, radius (to search around lat/lon)	Gives restaurants matching the search query in the area.
3	https://api.openrouteservice.org/v2/directions/foot-walking	start=(lat/lon pair) end=(lat/lon pair)	Gives the walking distance between two coordinate pairs.

Screenshot or code Snippet of APIs to be Used:

```
// Zomato API

$.ajax({
    url: 'https://developers.zomato.com/api/v2.1/search?q=Burger&count=4',
    method: 'GET',
    headers: { 'X-Zomato-API-Key': '1dc29c917607ec14f7f9f5309c721b3c' }
}).then(function (response) {
    console.log(response)
})
```

```
// TomTom API

$.ajax({
    url: 'https://api.tomtom.com/search/2/structuredGeocode.JSON?key=L7UIPFqhhWosaSn7oAMjfGZGsRJ9EnPU&countryCode=US&streetNumber=3920&streetName=Seeley&municipality=Chicago&countrySubdivision=Illinoiso&postalCode=60618',
    method: 'GET'
}).t  var console: Console  inse) {
    console.log(response)
})
```

```
// Open Route API

$.ajax({
    // Longitude comes first, then latitude, in each query url
    url: 'https://api.openrouteservice.org/v2/directions/foot-walking?api_key=5b3ce3597851110001cf6248664ece6aa70a4c7dbf8aa68951f471c3&start=-87.68021,41.95303&end=-87.63451,41.90145',
    method: 'GET'
}).then(function (response) {
    console.log(response)
})
```

API key for OR: 5b3ce3597851110001cf6248664ece6aa70a4c7dbf8aa68951f471c3

Screenshot or code Snippet of RESPONSE from APIs to be Used:

Zomato Response:


```
▼Object
  ▼restaurants: Array(4)
    ▼0:
      ▼restaurant:
        ▶R: {has_menu_status: {…}, res_id: 19093313, is_grocery_store: false}
        ▶all_reviews: {reviews: Array(5)}
         all_reviews_count: 5
         apikey: "1dc29c917607ec14f7f9f5309c721b3c"
         average_cost_for_two: 200
         book_again_url: ""
         book_form_web_view_url: ""
         cuisines: "Burger, Fast Food"
         currency: "Rs."
         deeplink: "zomato://restaurant/19093313"
        ▶establishment: []
        ▶establishment_types: []
         events_url: "https://www.zomato.com/ncr/burgery-1-sector-39-gurgaon/events#tabtop?utm_source=api_basic_user&utm_medium=api&utm_campaign=v2.1"
         featured_image: "https://b.zmtcdn.com/data/pictures/2/19086712/e283b071167d33ce3d36231d2a77eca3.jpg"
         has_online_delivery: 1
         has_table_booking: 0
        ▶highlights: (5) ["Breakfast", "Lunch", "Cash", "Delivery", "Dinner"]
         id: "19093313"
         include_bogo_offers: true
         is_book_form_web_view: 0
         is_delivering_now: 0
         is_table_reservation_supported: 0
         is_zomato_book_res: 0
        ▶location: {address: "770, Kilhore Patti, Jharsa Village, Near Sarv Haryana Gramin Bank, Sector 39, Gurgaon", locality: "Sector 39", city: "Gurgaon", city_id: …
         menu_url: "https://www.zomato.com/ncr/burgery-1-sector-39-gurgaon/menu?utm_source=api_basic_user&utm_medium=api&utm_campaign=v2.1&openSwipeBox=menu&showMinima…
         mezzo_provider: "OTHER"
         name: "Burgery"
        ▶offers: []
         opentable_support: 0
         phone_numbers: "0124 4200703"
         photo_count: 33
         photos_url: "https://www.zomato.com/ncr/burgery-1-sector-39-gurgaon/photos?utm_source=api_basic_user&utm_medium=api&utm_campaign=v2.1#tabtop"
         price_range: 1
         store_type: ""
         switch_to_order_menu: 0
         thumb: "https://b.zmtcdn.com/data/pictures/2/19086712/e283b071167d33ce3d36231d2a77eca3.jpg?fit=around%7C200%3A200&crop=200%3A200%3B%2A%2C%2A"
         timings: "10 AM to 11 PM"
         url: "https://www.zomato.com/ncr/burgery-1-sector-39-gurgaon?utm_source=api_basic_user&utm_medium=api&utm_campaign=v2.1"
        ▶user_rating: {aggregate_rating: "3.0", rating_text: "Average", rating_color: "CDD614", rating_obj: {…}, votes: "59"}
        ▶__proto__: Object
```


TomTom Response:

```
▼ results: Array(8)
  ▼ 0:
    ▶ address: {streetNumber: "1034", streetName: "North Wells Street", municipalitySubdivision: "Near North Side", municipality: "Chicago", countr…
    ▶ entryPoints: [{…}]
      id: "US/PAD/p1/24130424"
    ▶ position: {lat: 41.90145, lon: -87.63451}
      score: 9.476884841918945
      type: "Point Address"
    ▶ viewport: {topLeftPoint: {…}, btmRightPoint: {…}}
    ▶ __proto__: Object
  ▼ 1:
    ▶ address: {streetName: "North Wells Street", municipalitySubdivision: "Lincoln Park", municipality: "Chicago", countrySecondarySubdivision: "C…
      id: "US/STR/p0/7572224"
    ▶ position: {lat: 41.91361, lon: -87.6348}
      score: 7.714500904083252
      type: "Street"
    ▶ viewport: {topLeftPoint: {…}, btmRightPoint: {…}}
    ▶ __proto__: Object
  ▶ 2: {type: "Address Range", id: "US/ADDR/p0/37391744", score: 7.319115161895752, address: {…}, position: {…}, …}
  ▶ 3: {type: "Cross Street", id: "US/XSTR/p0/587185", score: 6.7199482917785645, address: {…}, position: {…}, …}
  ▶ 4: {type: "Cross Street", id: "US/XSTR/p1/148309", score: 6.703102111816406, address: {…}, position: {…}, …}
  ▶ 5: {type: "Street", id: "US/STR/p0/7516029", score: 5.618500709533691, address: {…}, position: {…}, …}
  ▶ 6: {type: "Street", id: "US/STR/p0/7516030", score: 5.618500709533691, address: {…}, position: {…}, …}
  ▶ 7: {type: "Street", id: "US/STR/p0/7527360", score: 5.618500709533691, address: {…}, position: {…}, …}
    length: 8
  ▶ __proto__: Array(0)
▶ summary: {query: "wells 1034 chicago 60610 illinois", queryType: "NON_NEAR", queryTime: 58, numResults: 8, offset: 0, …}
▶ __proto__: Object
```


Open Route Response:


```
▼ Object
  ▶ bbox: (4) [-87.679937, 41.901438, -87.634301, 41.953034]
  ▼ features: Array(1)
    ▼ 0:
      ▶ bbox: (4) [-87.679937, 41.901438, -87.634301, 41.953034]
      ▶ geometry: {coordinates: Array(216), type: "LineString"}
      ▼ properties:
        ▼ segments: Array(1)
          ▶ 0: {distance: 8469.9, duration: 6098.2, steps: Array(57)}
            length: 1
          ▶ __proto__: Array(0)
        ▶ summary: {distance: 8469.9, duration: 6098.2}
        ▶ way_points: (2) [0, 215]
        ▶ __proto__: Object
        type: "Feature"
      ▶ __proto__: Object
      length: 1
    ▶ __proto__: Array(0)
  ▼ metadata:
      attribution: "openrouteservice.org | OpenStreetMap contributors"
    ▶ engine: {version: "6.1.1", build_date: "2020-06-29T04:39:10Z", graph_date: "2020-06-15T00:00:01Z"}
    ▶ query: {coordinates: Array(2), profile: "foot-walking", format: "json"}
      service: "routing"
      timestamp: 1593494893446
    ▶ __proto__: Object
    type: "FeatureCollection"
  ▶ __proto__: Object
```


Libraries to be Used:

Library #	Doc Link	What does it do?	How did you use it?
1	Flexboxgrid	Tables	Will help to make interactive and aesthetically pleasing tabulated information
2	Materialize	CSS style Framework	Style components throughout.
3	TBD	Modals? (Jen)	

Project Reflection

Retrospective Notes:

Team Reflection:

- What Went Well
- What Did not go well
- What can **WE** improve for next time

Team Member Feedback (directed at your team member(s)):

- What Went Well
- What Did not go well
- What can **YOU** improve for next time

Team Reflection & Team Member Feedback Google Doc Link

Self Reflection:

- What Went Well
- What Did not go well
- What can **I** improve for next time

Self Reflection Link

Name		
Name		

Name		
Name		

Blog / Medium Post:

Create a medium blog post about (350 Words):

- A new technology and how to use it (tutorial)
- A new concept that you discovered
- Implementation of team working practices

Blog / Medium Link

Team Member #1	
Team Member # 2	
Team Member # 3	
Team Member # 4	

Presentation:

You will be responsible for preparing a formal, 10-minute presentation that covers the following:
Review the Presentation Template. Make a copy; file => make a copy
Project One Presentation Template

Notes: