Juan Gonzalez  juan_gonzalez@ucsb.edu | Daniel Kang  dkang@ucsb.edu

Iris Moini-Nazeri  azadeh@ucsb.edu | Krishna Nanda  krishnananda@ucsb.edu

Arthur Phan  arthurphan@ucsb.edu

Slack:  https://allstars-space.slack.com/messages/CHK46GYDN/

Sprint 1 Trello Board:  https://trello.com/b/8ok9LoKk/sprint-1

Sprint 2 Trello Board:  https://trello.com/b/qrTOMOBA/sprint-2

Sprint 3 Trello Board:  https://trello.com/b/0yE0mbmi/sprint-3

Github:  https://github.com/dkang1617/cs48ProjectOrganized/tree/master

Travis CI:  https://travis-ci.org/dkang1617/cs48ProjectOrganized

Demo Link:  https://drive.google.com/file/d/1Z0cMcoypX8Fefp58lPYxHm7fqYGRmbon/view?usp=sharing
# Table of Contents

## Draft Project Information:
- Sprint 1: April 15 - 26, Planning April 10-12
- Sprint 2: April 29 - May 10, Planning April 26
- Sprint 3: May 13 - May 24, Planning May 10

## Burn Down Chart

## System Architecture Overview
- Use Case #1: User Can Login
- Use Case #2: Automated Search: Choose Walking
- Use Case #3: Automated Search: Choose Biking
- Use Case #4: Manual Search: Choose Walking
- Use Case #5: Manual Search: Choose Biking
- Use Case #6: Manual Search: Biking and Walking
- Use Case #7: Edit Calendar Entry before putting into Calendar
- Use Case #8: Add All Entries to Calendar
- Use Case #9: Import Map List
- Use Case #10: Export Map List

## UML Diagrams
- Map Optimizer:
- React Portion:
- High Level Sequence Diagram:

## Retrospective info
End of Sprint #1 Consensus: 19
End of Sprint #2 Consensus: 19
End of Sprint #3 Consensus: 20

Challenges Faced/Overcome 20

Missing/Remaining Features or Functionality 20

Test Files and Functions 21

Commit List Contributors 22
  Juan Gonzalez: juangonzales20 23
  Daniel Kang: dkang1617 27
  Iris Moini-Nazeri: irismoini 30
  Krishna Nanda: Krishna-Nanda 31
  Arthur Phan: nahpruhtraton 33

Appendix 42
Draft Project Information:

Organized is an application that is looking to make the UCSB student’s life easier. The application has two main components: automatically adding course information associated with a student into Google Calendar using the Google Calendar API and using Google Maps API to create an optimal route for traversing campus quickly and easily.

| What problem is the project solving (what is innovation, science, and/or new core technical advance)? |
| Organization of multiple classes, focused on scheduling, while also mapping the optimal routes to classes, all done with one login. |

| Why is the problem important? |
| Helps students get a schedule under control, assisting them in performing to the best of their ability. |

| How is the problem solved today? (If it is.) |
| Google Maps does limited work in map optimization. |

| Identify the outcome of the project. |
| The primary outcome is a web app where users can view upcoming assignments and find the location of the optimal route to their classes. Reverse engineering and figuring out a way to replicate how these programs do it is the goal. Each member should be able to learn how to use API’s, the languages required to use them, and implementing and testing them on Travis CI by the end of the project, all while working as a group in industry-style workflow. |

| Specification |
| Use a student’s schedule to create a personalized calendar with a text file upload on a server that stores their information. Also, use the schedule and Google maps, combined with a database that would help in determining the optimal route from class to class. |

| Design |
| Design should not be graphically intensive. There should be (at least) two buttons on the front end: one to add events on a calendar, and one to access the map generation feature. |

| Prototyping |
| ● 1st Prototype: web based schedule to Google Calendar functionality |
2nd Prototype: web design with Google Calendar and Maps functionality

Testing:
We will use local hosting and Travis-CI to test our data structures and functionality.

How do you plan to articulate, design, and implement a solution?
Using publicly available APIs to work with the data and a server.

List the implementation platform and technologies you will use to develop the solution:

- Web App (JavaScript in React)
- Front End and Loading APIs (Using JavaScript)
- Java for Algorithms (Spring)
- Google Calendar and Map API

List initial milestones (w/Sprint/dates) & how you will achieve them:

Sprint 1: April 15 - 26, Planning April 10-12

- Get all the APIs authenticated and working with web app, and understanding how to efficiently use them. (Krishna + Daniel)
- Setting up a basic version of the web app. (K + D)
- Learn how to translate information from APIs, and build off of API authentication and use. (Juan)
- Setting up a way to translate the UCSB Interactive map into something we can use. (Arthur)
- Learn how to input translated information into Google Calendar (Iris)

Retrospective:

Sprint 2: April 29 - May 10, Planning April 26

- Initial test of design, implement revisions as needed.
- Start making sure privacy is ensured.
- Develop previous application further.

Retrospective:
Sprint 3: May 13 - May 24, Planning May 10

- Testing of map routing, using our own schedules.
- Ensuring privacy.
- Test scheduler and accuracy.
- Optimizing client side web app

Retrospective:

Burn Down Chart
System Architecture Overview

Requirements (Functional and Non-Functional)

(Tests in Trello)

Use Case #1: User Can Login

Actors: Web-App, User, React

Precondition: User is a UCSB Student with a valid UCSB gmail account (and is a student in the text file contained on a server meant to simulate the UCSB GOLD API), and inputs their information correctly into the Gmail login using their UCSB email.

Flow of Events:

1. User opens Web-App
2. User sees Gmail login interface on web-app, and enters log-in information
3. This information is used to access their course information contained in a text file on a server and this course information it placed into the Google Calendar.
Postcondition: Web-app has access to information and can use that to build schedule and map routes for classes.

Acceptance Test:

Scenario: Login is successful

1. Given the user is a UCSB student (and their info is in the textfile)
2. And their login information is inputted correctly
3. When the Web-App attempts to access and view user’s schedule
4. Then GauchoSpace will allow it to do so
5. And the Web-App will be able to use the information to input.

Exception Case:

Scenario: Login is unsuccessful

1. The User is not a UCSB Student with valid credentials (is not a student in the textfile)
2. The User incorrectly inputted their information
3. The web-app fails to pull information from the text server using the user’s information.

Trello Card Link:
https://trello.com/c/g3zAUHRH/56-get-google-oauth-working-2-days

| Use Case #2: Automated Search: Choose Walking |

Actors: Web-App, User, React

Precondition: User has clicked on the map component of the website and has their API loaded in, with browser pop-ups allowed

Flow of Events:

1. The user inputs their desired starting location
2. The user clicks submit, as walking is the default method of transportation

Postcondition: Pop-up for routes to their classes (pulled from API) show up in reverse order, in that the second-to-last and last destinations are generated first, working backwards to the first destination to the second destination, while choosing walking for their method of preferred transportation.

Acceptance Test:
Scenario: All maps generated successfully

1. The user presses submit on the automated map search after inputting a starting location
2. Google Maps requests will be opened in browser pop-ups corresponding to the starting location input and class locations from the API info
3. The user will be able to drag and resize those pop-up windows to see their routes for walking.

Exception Case:

Scenario: There is a duplicate in the API info

1. The user presses submit on the automated map search after inputting a starting location
2. However, there was a duplication location, in that a route was sending the user from and to the same building
3. That pop-up will not be requested by Google Maps
4. Google Maps requests will be opened in browser pop-ups corresponding to the starting location input and class locations from the API info
5. The user will be able to drag and resize those pop-up windows to see their routes for walking, without the redundant route.

Use Case #3: Automated Search: Choose Biking

Actors: Web-App, User, React

Precondition: User has clicked on the map component of the website and has their API loaded in, with browser pop-ups allowed

Flow of Events:

1. The user inputs their desired starting location
2. The user checks the box for biking as their method of transportation
3. The user clicks submit

Postcondition: Pop-up for routes to their classes (pulled from API) show up in reverse order, in that the second-to-last and last destinations are generated first, working backwards to the first destination to the second destination, while choosing biking for their method of preferred transportation.

Acceptance Test:

Scenario: All maps generated successfully

1. The user presses submit on the automated map search after inputting a starting location
2. Google Maps requests will be opened in browser pop-ups corresponding to the starting location input and class locations from the API info
3. The user will be able to drag and resize those pop-up windows to see their routes for biking.

Exception Case:

Scenario: There is a duplicate in the API info

1. The user presses submit on the automated map search after inputting a starting location
2. However, there was a duplication location, in that a route was sending the user from and to the same building
3. That pop-up will not be requested by Google Maps
4. Google Maps requests will be opened in browser pop-ups corresponding to the starting location input and class locations from the API info
5. The user will be able to drag and resize those pop-up windows to see their routes for biking, without the redundant route.

Use Case #4: Manual Search: Choose Walking

Actors: Web-App, User, React

Precondition: User has clicked on the map component of the website and has their browser pop-ups allowed.

Flow of Events:

1. The user inputs their desired starting location
2. The user is allowed input up to nine other locations
3. The user checks each box under their desired locations for walking as their method of transportation
4. The user clicks submit

Postcondition: Pop-up for routes to their classes show up in reverse order, in that the second-to-last and last destinations are generated first, working backwards to the first destination to the second destination, while choosing walking for each method of preferred transportation.

Acceptance Test:

Scenario: All maps generated successfully

1. The user inputs any location on UCSB Map, up to ten locations total
2. The user defaults to walking as the preferred method of transportation for each
3. Google Maps requests will be opened in browser pop-ups corresponding to the locations the user has inputted
4. The user will be able to drag and resize those pop-up windows to see their routes for walking.

Exception Case:

Scenario: There is a duplicate in the API info

1. The user inputs any location on UCSB Map, up to ten locations
2. The user defaults to walking as the preferred method of transportation for each
3. However, there was a duplication location, in that a route was sending the user from and to the same building
4. That pop-up will not be requested by Google Maps
5. Google Maps requests will be opened in browser pop-ups corresponding to the locations the user has inputted
6. The user will be able to drag and resize those pop-up windows to see their routes for walking, without the redundant route.

| Use Case #5: Manual Search: Choose Biking |

Actors: Web-App, User, React

Precondition: User has clicked on the map component of the website and has their browser pop-ups allowed.

Flow of Events:

1. The user inputs their desired starting location
2. The user is allowed input up to nine other locations
3. The user checks each box under their desired locations for biking as their method of transportation
4. The user clicks submit

Postcondition: Pop-up for routes to their classes show up in reverse order, in that the second-to-last and last destinations are generated first, working backwards to the first destination to the second destination, while choosing biking for each method of preferred transportation.

Acceptance Test:

Scenario: All maps generated successfully

1. The user inputs any location on UCSB Map, up to ten locations total
2. The user chooses biking as the preferred method of transportation for each
3. Google Maps requests will be opened in browser pop-ups corresponding to the locations the user has inputted
4. The user will be able to drag and resize those pop-up windows to see their routes for biking.

Exception Case:

Scenario: There is a duplicate in the API info

1. The user inputs any location on UCSB Map, up to ten locations
2. The user chooses biking as the preferred method of transportation for each
3. However, there was a duplication location, in that a route was sending the user from and to the same building
4. That pop-up will not be requested by Google Maps
5. Google Maps requests will be opened in browser pop-ups corresponding to the locations the user has inputted
6. The user will be able to drag and resize those pop-up windows to see their routes for biking, without the redundant route.

Use Case #6: Manual Search: Biking and Walking

Actors: Web-App, User, React

Precondition: User has clicked on the map component of the website and has their browser pop-ups allowed.

Flow of Events:

1. The user inputs their desired starting location
2. The user is allowed input up to nine other locations
3. The user checks each box under their desired locations for biking OR walking as their preferred method of transportation
4. The user clicks submit

Postcondition: Pop-up for routes to their classes show up in reverse order, in that the second-to-last and last destinations are generated first, working backwards to the first destination to the second destination, while choosing walking or biking for each method of preferred transportation to each destination accordingly.

Acceptance Test:

Scenario: All maps generated successfully

1. The user inputs any location on UCSB Map, up to ten locations total
2. The user chooses walking or biking as the preferred method of transportation for each location
3. Google Maps requests will be opened in browser pop-ups corresponding to the locations the user has inputted
4. The user will be able to drag and resize those pop-up windows to see their routes with a mix match of walking and biking.

Exception Case:

Scenario: There is a duplicate in the API info

1. The user inputs any location on UCSB Map, up to ten locations
2. The user chooses walking or biking as the preferred method of transportation for each location
3. However, there was a duplication location, in that a route was sending the user from and to the same building
4. That pop-up will not be requested by Google Maps
5. Google Maps requests will be opened in browser pop-ups corresponding to the locations the user has inputted
6. The user will be able to drag and resize those pop-up windows to see their routes with a mix match of walking and biking, without the redundant route.

---

<table>
<thead>
<tr>
<th>Use Case #7: Edit Calendar Entry before putting into Calendar</th>
</tr>
</thead>
</table>

Preconditions: User has already generated at least one entry.

Flow of Events:

Basic path:

1. The entry generated has some mistakes
2. The user edits the entry to improve it
3. The user confirms to add the individual entry to calendar

Alternative Path:

1. The user does not want to edit the entry
2. The user confirms to add the individual entry to calendar

Postcondition: Entries are edited or kept the same, and then are manually approved to be added to the calendar. The entries that are confirmed are removed from the list.

<table>
<thead>
<tr>
<th>Use Case #8: Add All Entries to Calendar</th>
</tr>
</thead>
</table>

Preconditions: User has already generated at least one entry, and we have information to input into Calendar.

Flow of Events:
Basic path:

1. The user clicks on “add all”
2. The website asks for confirmation
3. The user confirms and all entries are added to calendar and removed from the list

Alternative Path: The user cancels adding all entries onto calendar

Postcondition: Every entry is added to the calendar, and there are no more entries on the list to account for, OR no entry is added.

Trello Card Link:

https://trello.com/c/xVee62w2/31-c-figure-out-how-to-use-the-google-calendar-api-1-day
https://trello.com/c/kLZBzQLo/20-c-figure-out-how-to-use-schedule-api-1-day
https://trello.com/c/yvdc20ez/34-c-get-a-simple-version-of-the-webapp-that-shows-us-using-those-api-3-day

Use Case #9: Import Map List

Actors: User, Website Interface, Client storage

Precondition: User is logged into the website and has a text file of their list they want to import

Flow of Events:

Basic:

1. User clicks a “Import List” button
2. The website imports the list, reads it, and generates the list on the website interface
3. The list can be generated if the user wants

Alternative:

1. User clicks a “Import List” button
2. The website imports the list, attempts to read it, but the text file is not in proper format
3. The website writes an error message

Postcondition: User has imported their map from their computer.

Acceptance Test:

Scenario: Text file is uploaded, read from, and not stored in server side

1. User selects “Import List”
2. The website imports the list, attempts to read it, and checks if it is in proper format
3. The list is properly imported
4. “The list file could not be read, please try again”

| Use Case #10: Export Map List |

Actors: User, Website Interface, Client storage

Precondition: User is logged into the website and has a text file of their list they want to download to their machine (to save)

Flow of Events:

Basic:

1. User clicks a “Export List” button
2. The website write the list to a temporary text file
3. The website prompts for the user to download
4. The text file is deleted on website exit

Postcondition: User has downloaded their map to their computer.

Acceptance Test:

Scenario: Text file is downloaded from user list

1. User creates their own list
2. User clicks a “Export List” button
3. The website write the list to a temporary text file
4. The website prompts for the user to download
5. The text file is deleted on website exit
UML Diagrams

Map Optimizer:

- Interface: Database
  - 1

MapUser

- Username: String
- Password: String
- logStatus: boolean
- listIDs: String[7]

- getUsername(): String
- getPassword(): String
- verifyLogin(String, String): boolean
- addList(String): LinkedList
- addNode(LinkedList, DestNode): void

LinkedList

- 1 0-7

DestNode

- Source: String
- Method: String

- getSource(): String
- getMethod(): String

- 1 0-10
Retrospective info

End of Sprint #1 Consensus:

- **What Worked:**
  - Everyone was individually learning technologies and languages to be used for the web application
  - Most of us assigned ourselves to tasks well

- **What Didn’t:**
  - We jumped the gun on actually coding; we need to have spent more time brainstorming how our web application was going to be and how it would look
  - Sticking to daily tasks seemed to not work considering how much work we needed to do for other classes
    - This might have been a result of trying to code right off the bat instead of getting a deeper brainstorming and understanding of how our project wanted to end up
    - This was also due to improper measuring of task timing

- **What Should Change:**
  - We needed to meet up in person more often
  - We needed to participate in daily scrums instead of brushing it off, even if most days we did not work on the project daily

End of Sprint #2 Consensus:

- **What Worked:**
  - Some fruits of labor have come: we figured out how to set up the Google Calendar addition
  - Webpage landing screen and post-login screen was finished up

- **What Didn’t:**
  - GOLD API was unable to be used, and we needed to pivot, and fast
  - The original Java implementation was not well thought out
    - It seemed redundant, and did not work well with the framework the webpage resulting in

- **What Should Change:**
  - Pivoting: no GOLD API means we need to make our own
    - We also needed to make our own JAVA server to host this API, since removing the original Java implementation meant we had no Java component accounted for
  - Accountability: some of us worked too little in Sprint #2
    - Moving forward, we needed to try to keep each other motivated and more accountable for the work we were supposed to do
End of Sprint #3 Consensus:

- **What Worked:**
  - Server implementation working for serve API
  - Map functionality working for implementation
- **What Didn’t:**
  - Testing too shallow
  - It was clear that sticking to Trello cards and tasks was not going to work, since it felt like another hoop to jump through even though we know what we needed to finish
- **What Should Change:**
  - Allocation of work amongst members could be better
  - Increased focus on testing components

Challenges Faced/Overcome

Our group faced numerous challenges throughout the quarter, many of which we were eventually able to overcome. Initially these challenges revolved around our lack of experience with certain languages and with using APIs, which were a central component of our project. Taking advantage of the resources available to us and sharing them amongst ourselves proved to be a valuable method during these early learning stages. As the quarter progressed, the issues we ran into increased in complexity and affected the larger course our project took. Upon contacting the developer of the GOLD API we discovered that gaining access to student schedules would not be manageable due to the API’s lack of functionality. In order to tackle this problem, we decided to create a database in an attempt to essentially replicate the API. Finally, establishing a server in Java and connecting our frontend and backend was an issue we faced, which we solved by speaking to others and working with a TA. The challenges we encountered gave us perspective on the software development process and encouraged us to come together and work closely as a team to help resolve them.

Missing/Remaining Features or Functionality

Our team recognizes that Organized is admittedly a very humble and limited web application, underdelivering on components we have previously wanted to add in.

One missing feature that we would have liked to add in would be a web scraping component that takes your GOLD API, looks up that class on the UCSB Bookstore website, and finds you those books from alternative websites for cheaper prices. Two missing map features that were unable to be added were the print map and export map functionality. Since Organized is about having getting yourself orientated to the new quarter and scheduling, we would have liked to have given users the opportunity to save their maps by printing or exporting their list, to be saved or shared with friends. The problem with implementing this was in how the maps are currently
implemented; window pop-ups instead of the intended idea of embedded maps on the maps page of the application.

Future enhancements include quality of life changes, such as optimizing the automation with Google Calendar and creating a real algorithm for bike routing. These changes would have allowed the user to edit entries before sending those entries to Google Calendar, and creating a real algorithm for biking to send users to bike racks, and then to classes.

Finally, we would have preferred to have actually used GOLD API. The most prized goal for us was to actually be able to pull real user data from the UCSB GOLD API, but we were blindsided by the actual code and its implementation issues. Other UCSB data we would have liked to add would be adding UCSB related events and looking into the DIning Commons menu for other components to our web application.

Test Files and Functions

JsonParser.test.js tests JsonParser.js’ functions that receive input from our database on our Java server and parse it into information we can input into Google Calendar. We test the functions that receive the start time and end time, as well as the getID and getDate functions. We set sample values and check for equivalence.

App.test.js tests the Web App’s login and logout functionalities. We check that each function for logging-in and logging-out executes smoothly without issue. We also check that in the case that the login is inputted incorrectly, the web app handles that issue and outputs the correct error message. render test checks to see if the site renders without crashing.

Travis CI is configured to run these two test files via the “npm test” command that Node.js provides.

The Java server and Map functionality implementations proved too simple, so we did not intensively test their implementations. With the Java server, we would know that it was not running if it did not serve a JSON file and did not input events into Google Calendar. As for the Map functionality, debugging happened with console output statements to track how the pop-ups were being made and how their strings were concatenated.
Commit List Contributors

Mar 31, 2019 – Jun 11, 2019

Contributions to master, excluding merge commits

Arthur Phan

 nahpruhratran
 67 commits 921 ++ 867 --

Juan Gonzales

 juangonzalez20
 25 commits 560 ++ 378 --

Daniel Kang

 dkang1617
 22 commits 248 ++ 133 --

Krishna Nanda

 Krishna-Nanda
 13 commits 584 ++ 260 --

Iris Moini-Nazeri

 irismoini
 7 commits 152,610 ++ 130 --
Juan Gonzalez: juangonzales20

Commits on Jun 10, 2019

Commented out config lines to get Travis working  
juangonzales20 committed 12 hours ago

Commits on May 22, 2019

Fixed some JSX stuff and addArray  
juangonzales20 committed 19 days ago

merge fix  
juangonzales20 committed 19 days ago

Fixed radio stuff  
juangonzales20 committed 19 days ago

Updated gitignore to include maven target folder  
juangonzales20 committed 20 days ago

Merge branch 'juan' of https://github.com/dkang1617/cs48ProjectOrganized
juangonzales20 committed 20 days ago

Merge branch 'master' of https://github.com/dkang1617/cs48ProjectOrganized
juangonzales20 committed 20 days ago

Merge branch 'juan' of https://github.com/dkang1617/cs48ProjectOrganized
juangonzales20 committed 20 days ago

Commits on May 21, 2019

Made new page for map stuff, removed cursor animation, added react-...
### Commits on May 20, 2019

**Trying to fix git stuff**
- juangonzalez20 committed 22 days ago

**Merge branch 'master' of https://github.com/dkang1617/cs48ProjectOrganized...**
- juangonzalez20 committed 22 days ago

### Commits on May 19, 2019

**Got rid of useless getCourse method**
- juangonzalez20 committed 23 days ago

**Fixed JSX stuff in App.js and Spinner.js**
- juangonzalez20 committed 23 days ago

**Added JsonParser test file**
- juangonzalez20 committed 23 days ago

### Commits on May 18, 2019

**Revert "Added getBuilding method to JsonParser"**
- juangonzalez20 committed 23 days ago

**Merge branch 'master' of https://github.com/dkang1617/cs48ProjectOrganized...**
- juangonzalez20 committed 23 days ago

**Added getBuilding method to JsonParser**
- juangonzalez20 committed 23 days ago

**Merge branch 'juan' of https://github.com/dkang1617/cs48ProjectOrganized...**
- juangonzalez20 committed 24 days ago
Deleted Untitled files
Jaangonzalez20 committed 24 days ago

Commits on May 16, 2019

Figured out spinner
Jaangonzalez20 committed 25 days ago ✔

Finished up calendar functionality, and included a java server test
Jaangonzalez20 committed 26 days ago ✔

Commits on May 8, 2019

Had to get rid of references to config
Jaangonzalez20 committed on May 8 ✔

Commented out the import of config to get test to work
Jaangonzalez20 committed on May 8 ✗

Still trying
Jaangonzalez20 committed on May 8 ✗

Playing around with travis.yml again
Jaangonzalez20 committed on May 8 ✗

Merge branch 'master' of https://github.com/dkang/1617/cs48ProjectOrga...
Jaangonzalez20 committed on May 8 ✔

Making something to parse Json
Jaangonzalez20 committed on May 8 ✔
Commits on May 7, 2019

- Hard coded an example of what making events form a json file might lo...
  juangonzalez20 committed on May 7

Commits on May 6, 2019

- Merge branch 'master' of https://github.com/dkang1617/cs48ProjectOrga...
  juangonzalez20 committed on May 6

Commits on May 3, 2019

- Fixing travis stuff
  juangonzalez20 committed on May 3

- Revert "Trying to get Travis Ci working with React stuff" (no changes)
  juangonzalez20 committed on May 3

- Trying to play around with config file
  juangonzalez20 committed on May 3

- Trying to get Travis Ci working with React stuff
  juangonzalez20 committed on May 3

- Wrote test for login and logout functions
  juangonzalez20 committed on May 3

Commits on May 2, 2019

- Can create on an event on Google calendar
  juangonzalez20 committed on May 2
Daniel Kang: **dkang1617**

- **Commits on May 22, 2019**
  
  - Merge pull request #28 from dkang1617/daniel
    - dkang1617 committed 20 days ago

- **Commits on May 19, 2019**
  
  - this is the final one
    - dkang1617 committed 23 days ago
  
  - final update
    - dkang1617 committed 23 days ago
  
  - update
    - dkang1617 committed 23 days ago
  
  - update
    - dkang1617 committed 23 days ago
  
  - update
    - dkang1617 committed 23 days ago
  
  - just a copy of what is on our server for the demo
    - dkang1617 committed 23 days ago
  
  - wrote tests for jsonparser
    - dkang1617 committed 23 days ago
### Commits on Apr 29, 2019

- **Merge pull request #9 from dkang1617/daniel**
  - dkang1617 committed on Apr 29

- **first fake database**
  - dkang1617 committed on Apr 29

- **Merge pull request #8 from dkang1617/daniel**
  - dkang1617 committed on Apr 29

- **daniel branch**
  - dkang1617 committed on Apr 29

- **commit**
  - dkang1617 committed on Apr 29

### Commits on Apr 17, 2019

- **dkang commit**
  - dkang1617 committed on Apr 17
  - 6a42712

- **dkang commit**
  - dkang1617 committed on Apr 17
  - ab746e8

### Commits on Apr 3, 2019

- **Initial commit**
  - dkang1617 committed on Apr 3
  - faf8685
Iris Moini-Nazeri: irismoini

<table>
<thead>
<tr>
<th>Date</th>
<th>Commit Message</th>
<th>Author</th>
<th>Hash</th>
</tr>
</thead>
<tbody>
<tr>
<td>May 27, 2019</td>
<td>changes made to map ui</td>
<td>irismoini</td>
<td>62b1b00</td>
</tr>
<tr>
<td></td>
<td>Merge pull request #34 from dkang1617/iris</td>
<td>irismoini</td>
<td>0c5516</td>
</tr>
<tr>
<td></td>
<td>Merge branch 'master' into iris</td>
<td>irismoini</td>
<td>55beea0</td>
</tr>
<tr>
<td></td>
<td>changes made to ui</td>
<td>irismoini</td>
<td>26b4a77</td>
</tr>
<tr>
<td>May 22, 2019</td>
<td>changes made to map page</td>
<td>irismoini</td>
<td>afeb083</td>
</tr>
<tr>
<td></td>
<td>changes made to messed up interface</td>
<td>irismoini</td>
<td>0269a84</td>
</tr>
<tr>
<td></td>
<td>title change</td>
<td>irismoini</td>
<td>f05b004</td>
</tr>
<tr>
<td></td>
<td>changes made to title (now in js)</td>
<td>irismoini</td>
<td>eb3b3e39</td>
</tr>
<tr>
<td>May 21, 2019</td>
<td>minor changes</td>
<td>irismoini</td>
<td>9be0117</td>
</tr>
</tbody>
</table>


Krishna Nanda: Krishna-Nanda

Commits on May 27, 2019

- final fixes to alignment for demo - w/ arthur
  - Krishna-Nanda committed 14 days ago

- Merge branch 'master' of https://github.com/dkang1617/cs48ProjectOrga...
  - Krishna-Nanda committed 15 days ago

demo configuration
- Krishna-Nanda committed 15 days ago

- Merge branch 'master' of https://github.com/dkang1617/cs48ProjectOrga...
  - Krishna-Nanda committed 15 days ago

merge conflict to be resolved
- Krishna-Nanda committed 15 days ago

Commits on May 22, 2019

- Fixed html tags, and various variable bugs
  - Fix by Krishna and Iris

  - Krishna-Nanda committed 20 days ago

Commits on May 18, 2019

- Corsfilterjava file for Backend Server
  - Krishna-Nanda committed 23 days ago

- Fixed button on homescreen
  - Krishna-Nanda committed 23 days ago

- conflict resolves - duplicate code
  - Krishna and Arthur peer reviewed code to resolve conflicts, but may have not been totally resolved here (resolved later)

  - Krishna-Nanda committed 24 days ago

  - 1 commit
### Commits on May 15, 2019

- **Update Main.java**
  - Krishna-Nanda committed 26 days ago

- **Changes to UI after Login**
  - Krishna-Nanda committed 26 days ago

- **Merge branch 'master' into krishna**
  - Krishna-Nanda committed 27 days ago

### Commits on May 14, 2019

- **Update Main.java**
  - Krishna-Nanda committed 28 days ago

- **Update Main.java**
  - Krishna-Nanda committed 28 days ago

- **Commiting code for Backend**
  - Krishna-Nanda committed 28 days ago

### Commits on May 1, 2019

- **Merge pull request #12 from dkang1617/master**
  - Krishna-Nanda committed on May 1

- **test12**
  - Krishna-Nanda committed on May 1
Arthur Phan: nahpruhtaron

Commits on May 27, 2019

- no zip, just usb tag
  nahpruhtaron committed 15 days ago

- fixes to border w/ iris
  nahpruhtaron committed 15 days ago

- Merge pull request #37 from dkang1617/iris
  nahpruhtaron committed 15 days ago

- Merge pull request #36 from dkang1617/master
  nahpruhtaron committed 15 days ago

- Merge branch 'master' of github.comdkang1617/cs48ProjectOrganized
  nahpruhtaron committed 15 days ago

- god's plan
  nahpruhtaron committed 15 days ago

- zipcode to destination
  nahpruhtaron committed 15 days ago

- Merge pull request #35 from dkang1617/iris
  nahpruhtaron committed 15 days ago

- Update README.md
  nahpruhtaron committed 15 days ago

- manual add tested; the only thing to work on is popup- alignment
  nahpruhtaron committed 15 days ago

- reinvent, working on iterator
  nahpruhtaron committed 15 days ago

- removed mouse follow, continued work on manual function
  nahpruhtaron committed 15 days ago
<table>
<thead>
<tr>
<th>Commits on May 26, 2019</th>
</tr>
</thead>
<tbody>
<tr>
<td>groundwork laid for manual add, writing function now</td>
</tr>
<tr>
<td>naphpuhrat on committed 15 days ago</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Commits on May 23, 2019</th>
</tr>
</thead>
<tbody>
<tr>
<td>Merge pull request #33 from dkang1617/master+Juan</td>
</tr>
<tr>
<td>naphpuhrat on committed 19 days ago</td>
</tr>
<tr>
<td>happy with the output values; more features to be added -- DEMO READY</td>
</tr>
<tr>
<td>naphpuhrat on committed 19 days ago</td>
</tr>
<tr>
<td>wow... I kinda cheesed it so hard</td>
</tr>
<tr>
<td>naphpuhrat on committed 19 days ago</td>
</tr>
<tr>
<td>working on this version of the branch</td>
</tr>
<tr>
<td>naphpuhrat on committed 19 days ago</td>
</tr>
<tr>
<td>Merge pull request #32 from dkang1617/iris</td>
</tr>
<tr>
<td>naphpuhrat on committed 19 days ago</td>
</tr>
<tr>
<td>Merge branch 'master' into iris</td>
</tr>
<tr>
<td>naphpuhrat on committed 19 days ago</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Commits on May 22, 2019</th>
</tr>
</thead>
<tbody>
<tr>
<td>instructions for what we need to finish added to map.js</td>
</tr>
<tr>
<td>naphpuhrat on committed 19 days ago</td>
</tr>
<tr>
<td>Merge branch 'master' of github.com:dkang1617/cs48ProjectOrganized</td>
</tr>
<tr>
<td>naphpuhrat on committed 19 days ago</td>
</tr>
<tr>
<td>action on maps webpage</td>
</tr>
<tr>
<td>naphpuhrat on committed 19 days ago</td>
</tr>
<tr>
<td>Merge pull request #31 from dkang1617/juan</td>
</tr>
<tr>
<td>-------------------------------------------</td>
</tr>
<tr>
<td>Comparison text</td>
</tr>
<tr>
<td>Halpruhratron committed 19 days ago</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Merge pull request #30 from dkang1617/master</th>
<th>Verified</th>
<th>09ad437</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison text</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Halpruhratron committed 20 days ago</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| fix to map website                          |          |         |
|Halpruhratron committed 20 days ago          |          | ac3c3e9 |

<table>
<thead>
<tr>
<th>Merge branch 'master' of github.com/dkang1617/cs48ProjectOrganized</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison text</td>
<td></td>
<td>0cbe4fb</td>
</tr>
<tr>
<td>Halpruhratron committed 20 days ago</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| array method added                           |          |         |
|Halpruhratron committed 20 days ago          |          | e26aa33 |

<table>
<thead>
<tr>
<th>Merge pull request #29 from dkang1617/master</th>
<th>Verified</th>
<th>997b08e</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison text</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Halpruhratron committed 20 days ago</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Merge branch 'iris' into master</th>
<th>Verified</th>
<th>2f7289c</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison text</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Halpruhratron committed 20 days ago</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Commits on May 21, 2019

<table>
<thead>
<tr>
<th>removed some files; started maps webpage</th>
<th></th>
<th>85d56f6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison text</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Halpruhratron committed 20 days ago</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Merge branch 'master' of github.com/dkang1617/cs48ProjectOrganized</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison text</td>
<td></td>
<td>562e73c</td>
</tr>
<tr>
<td>Halpruhratron committed 20 days ago</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>tabs &gt; spaces</th>
<th></th>
<th>11e058f</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison text</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Halpruhratron committed 20 days ago</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>yml config</th>
<th></th>
<th>27c17f1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison text</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Halpruhratron committed 20 days ago</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Merge pull request #27 from dkang1617/master</th>
<th>Verified</th>
<th>167571c</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison text</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Halpruhratron committed 20 days ago</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Merge branch 'juan' into master

<table>
<thead>
<tr>
<th>Merge branch 'juan' into master</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison text</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Halpruhratron committed 20 days ago</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Merge checked by Juan and Arthur</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison text</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Commits on May 21, 2019

<table>
<thead>
<tr>
<th>Comment</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image.png" alt="GitHub Commit" /></td>
<td>commented on App.js - removed deprecated funcs</td>
</tr>
<tr>
<td><img src="image.png" alt="GitHub Commit" /></td>
<td>Merge pull request #26 from dkkang1617/master</td>
</tr>
<tr>
<td><img src="image.png" alt="GitHub Commit" /></td>
<td>Merge branch 'iris' into master</td>
</tr>
<tr>
<td><img src="image.png" alt="GitHub Commit" /></td>
<td>Merge pull request #25 from dkkang1617/juan</td>
</tr>
<tr>
<td><img src="image.png" alt="GitHub Commit" /></td>
<td>removed make</td>
</tr>
</tbody>
</table>

### Conflict/Code discrepancies checked by Juan and Arthur

## Commits on May 20, 2019

<table>
<thead>
<tr>
<th>Comment</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image.png" alt="GitHub Commit" /></td>
<td>fix travis - not sure if maven testing will happen</td>
</tr>
</tbody>
</table>

## Commits on May 18, 2019

<table>
<thead>
<tr>
<th>Comment</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image.png" alt="GitHub Commit" /></td>
<td>maven please work</td>
</tr>
<tr>
<td><img src="image.png" alt="GitHub Commit" /></td>
<td>changed order of script</td>
</tr>
<tr>
<td><img src="image.png" alt="GitHub Commit" /></td>
<td>fuck it, try it out</td>
</tr>
<tr>
<td><img src="image.png" alt="GitHub Commit" /></td>
<td>Merge pull request #23 from dkkang1617/master</td>
</tr>
</tbody>
</table>
Conflicts resolved by Arthur and Juan

Finally resolved conflicts w/ Juan, Krishna, and Arthur

Merge pull request #21 from dkang1617/master

update to read2

removed bad files from master; updated readme

total merge #3

Merge branch 'master' into krishna

total merge #2

Merge branch 'master' into juan

total merge #1

Commits on May 6, 2019

Merge pull request #15 from dkang1617/iris
Merge pull request #14 from dkang1617/juan
nahpruhratron committed on May 5

Delete implementation.class
nahpruhratron committed on May 6

Delete DestNode.class
nahpruhratron committed on May 6

Delete dkang.java
nahpruhratron committed on May 6

update to remove .class files
nahpruhratron committed on May 6

Commits on Apr 30, 2019

Code reviewed by all members before Sprint #1 Merge
Finalized merges from Sprint #1
nahpruhratron committed on Apr 30

Merge pull request #10 from dkang1617/arthur
nahpruhratron committed on Apr 30

Commits on Apr 29, 2019

update
nahpruhratron committed on Apr 29

Commits on Apr 25, 2019

intended
nahpruhratron committed on Apr 25

bug "fixed"
nahpruhratron committed on Apr 25

one last change actually
nahpruhratrons committed on Apr 25

last commit for implementation for now; will work on testing file
nahpruhratron committed on Apr 25

removed test statements
nahpruhratron committed on Apr 25
Commits on Apr 25, 2019

implementation might be good - now to work on linking to APIs
  narshpruhalten committed on Apr 25

changed conditional
  narshpruhalten committed on Apr 25

debugging
  narshpruhalten committed on Apr 25

edits to travis and makefile to prep for java testing
  narshpruhalten committed on Apr 25

travis change, and implementation change
  narshpruhalten committed on Apr 25

test
  narshpruhalten committed on Apr 25

small error in logic
  narshpruhalten committed on Apr 25

didn't know booleans in java SUCK
  narshpruhalten committed on Apr 25

changes to implementation.java
  narshpruhalten committed on Apr 25

need to figure out console input problem
  narshpruhalten committed on Apr 25

missing semicolon
  narshpruhalten committed on Apr 25

stop conditions
  narshpruhalten committed on Apr 25
<table>
<thead>
<tr>
<th>Branch</th>
<th>Author</th>
<th>Commit Message</th>
<th>Date</th>
<th>SHA</th>
<th>Verified</th>
</tr>
</thead>
<tbody>
<tr>
<td>small error in makefile</td>
<td>naphruhtraton</td>
<td>committed on Apr 25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>makefile test</td>
<td>naphruhtraton</td>
<td>committed on Apr 25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>merge</td>
<td>naphruhtraton</td>
<td>committed on Apr 25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>make edit</td>
<td>naphruhtraton</td>
<td>committed on Apr 25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Merge pull request #7</td>
<td>naphruhtraton</td>
<td>committed on Apr 25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>trying makefile</td>
<td>naphruhtraton</td>
<td>committed on Apr 25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Merge pull request #6</td>
<td>naphruhtraton</td>
<td>committed on Apr 25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>small change</td>
<td>naphruhtraton</td>
<td>committed on Apr 25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>optipath delete</td>
<td>naphruhtraton</td>
<td>committed on Apr 25</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Commits on Apr 22, 2019

<table>
<thead>
<tr>
<th>Branch</th>
<th>Author</th>
<th>Commit Message</th>
<th>Date</th>
<th>SHA</th>
<th>Verified</th>
</tr>
</thead>
<tbody>
<tr>
<td>formatting, trying a while loop for input</td>
<td>naphruhtraton</td>
<td>committed on Apr 22</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Commits on Apr 19, 2019

<table>
<thead>
<tr>
<th>Branch</th>
<th>Author</th>
<th>Commit Message</th>
<th>Date</th>
<th>SHA</th>
<th>Verified</th>
</tr>
</thead>
<tbody>
<tr>
<td>Merge pull request #4</td>
<td>naphruhtraton</td>
<td>committed on Apr 19</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
working on linkedlist
  nahphrtration committed on Apr 19

linkedlist sorta working
  nahphrtration committed on Apr 19

Merge pull request #3 from dkang1617/optiPath
  nahphrtration committed on Apr 19
  Verified

node class should be done; need to work on LL + api
  nahphrtration committed on Apr 19
  1d95325

test commit 2
  nahphrtration committed on Apr 19
  3621486

Merge branch 'optiPath' of github.com:dkang1617/cs48ProjectOrganized ...
  nahphrtration committed on Apr 19

Merge pull request #2 from dkang1617/master
  nahphrtration committed on Apr 19
  Verified
  efd9b267

test commit
  nahphrtration committed on Apr 19

fix travis
  nahphrtration committed on Apr 19

Commits on Apr 17, 2019

testing travis
  nahphrtration committed on Apr 17
  346a206

setting up travis
  nahphrtration committed on Apr 17
  ed03509

Merge pull request #1 from dkang1617/optiPath
  nahphrtration committed on Apr 17
  Verified
  9cb48bd

Commits on Apr 17, 2019

branch commit test
  nahphrtration committed on Apr 17

  89def9c

test
  nahphrtration committed on Apr 17
  3ca58de
Appendix

- Front End: React as the Javascript framework
  - HTML + CSS
- API Server: Spark as the Java framework
  - Java server as the API server for website to make calls to
- Google OAuth - login service
- Google API's:
  - Google Calendar
    - Automated entry adding
  - Google Maps
    - Automated and manual path routing