Team Aqua

Alac, Calvin, Robert, Tony, & Winnie
Deliverable 4

Agenda

1) Product & Sprint Backlog
2) Team Admin: Task Board, Burndown Chart
3) Team Admin: Git Repo & Code Reviews
4) System Designs
5) Verification & Validation Process
6) Demo of Latest Release

Product & Sprint Backlog

Product Backlog

Product Backlog Part 1

Search Engine

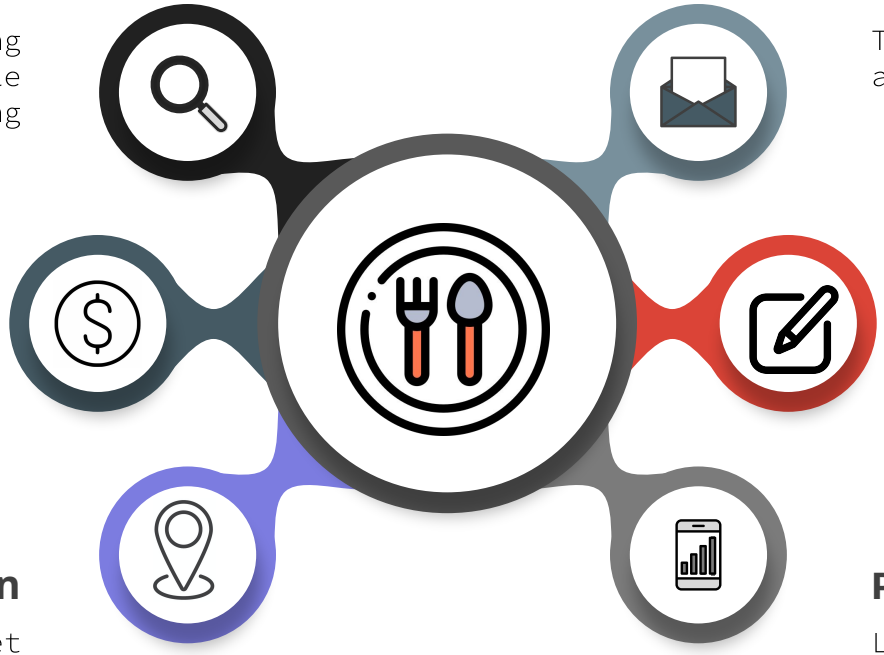
For filtering restaurants available on Scarborough Dining

Specials & Discounts

For restaurant promotions and labelling special dishes

NEW* Geo Location

Things needed to let users look up restaurants by location



Login and Signup

Things needed to login and signup for the page

NEW* Edit Web Page

Things needed to edit restaurant pages such as menus, hours and etc...

Platform Analytics

Let system admins view data such as usage and traffic

Product Backlog Part 2

***NEW Restaurant Setup**

Things needed for restaurant personnel to input relevant info such as restaurant details & menu

Connecting with Customers

Online community to let customers interact digitally

Restaurant Recipe

Let restaurant staff upload recipes, cooking tips and etc...

Restaurant Profile

All things related to the physical page of restaurants on Scarborough Dining

Placing Orders

Features needed to allow customers to order food from restaurants

Accepting Orders

Features needed to let restaurants accept orders

Favourite Restaurant & Reviews

Related to rating restaurants and favoriting them

Sprint Backlogs

- Sprint 2: June 28th to July 5th
- Sprint 3: July 5th to 12th
- Sprint 4: July 12th to 19th

Sprint 2: June 28th to July 5th

Sprint 2

Items Worked On:

- UIs for Restaurants
- Tagging
- Initial Web Page Deployment

Sprint 3

Items Worked On:

- Populate Database
- Endpoints to retrieve data
- Dashboard UI

Sprint 4

Items Worked On:

- Restaurant Setup Forms
- Store Form Informations
- Populate Database

Sprint 3: July 5th to July 12th

Sprint 2

Items Worked On:

- UIs for Restaurants
- Tagging
- Initial Web Page Deployment

Sprint 3

Items Worked On:

- Populate Database
- Endpoints to retrieve data
- Dashboard UI

Sprint 4

Items Worked On:

- Restaurant Setup Forms
- Store Form Informations
- Populate Database

Sprint 4: July 12th to July 19th

Sprint 2

Items Worked On:

- UIs for Restaurants
- Tagging
- Initial Web Page Deployment

Sprint 3

Items Worked On:

- Populate Database
- Endpoints to retrieve data
- Dashboard UI

Sprint 4

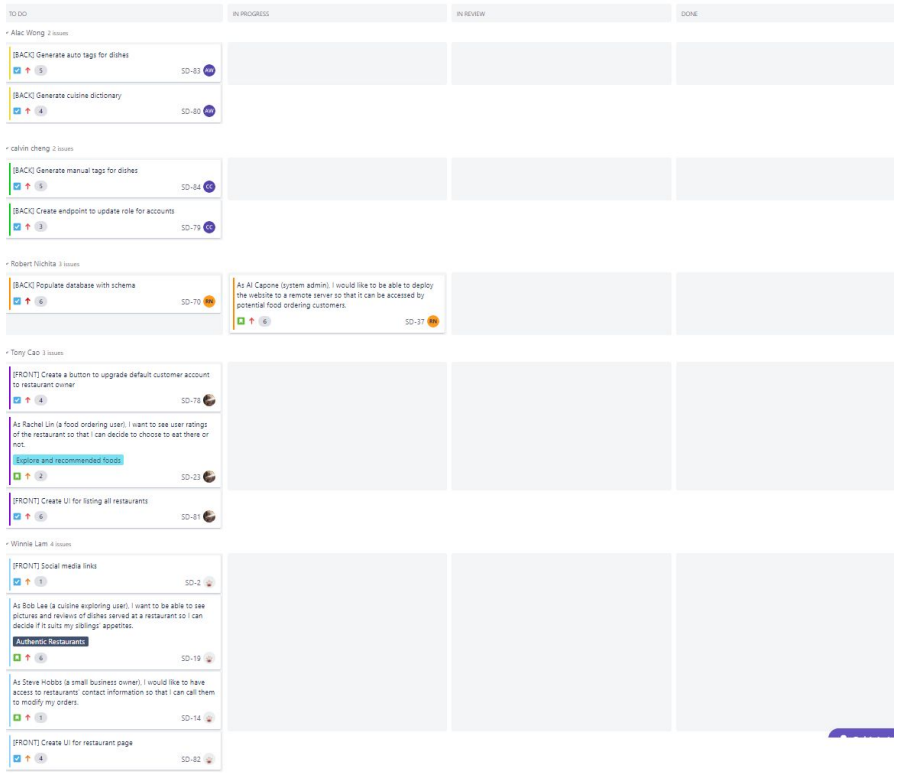
Items Worked On:

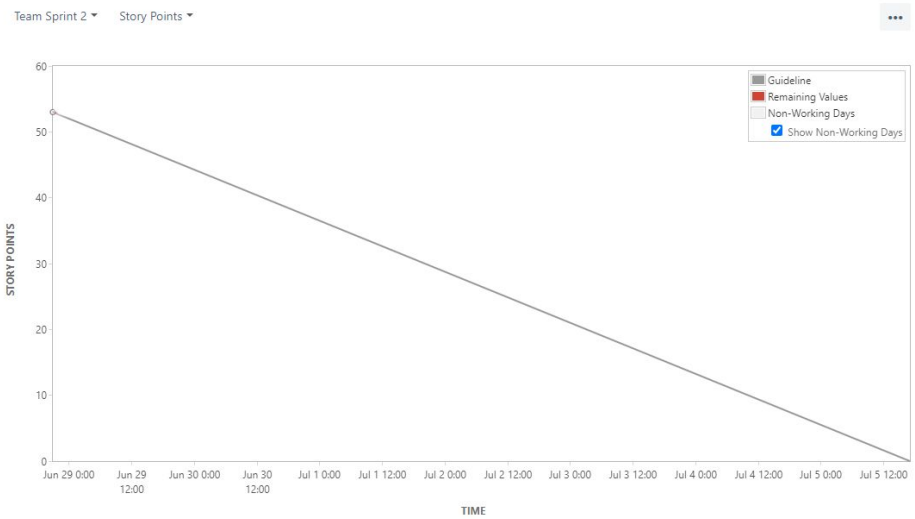
- Restaurant Setup Forms
- Store Form Informations
- Populate Database

Task Board & Burndown Chart

Sprint 2

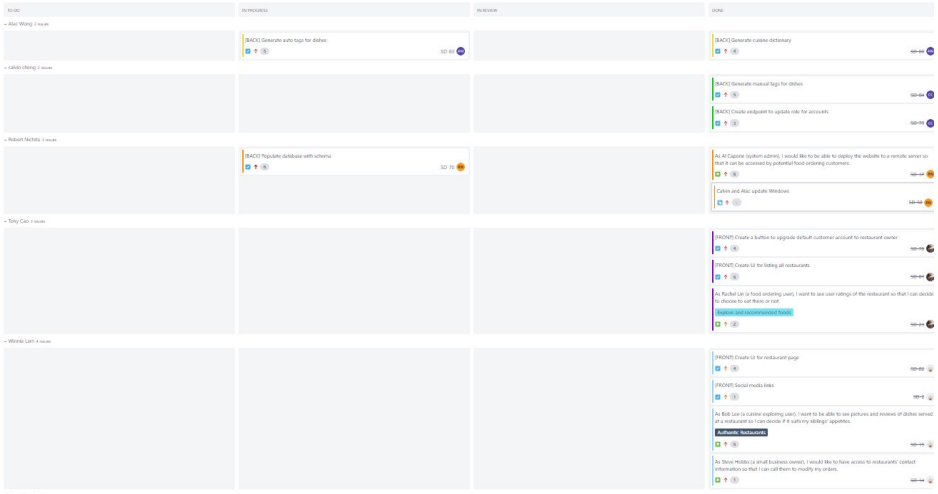
June 28 - July 5 (Initial)

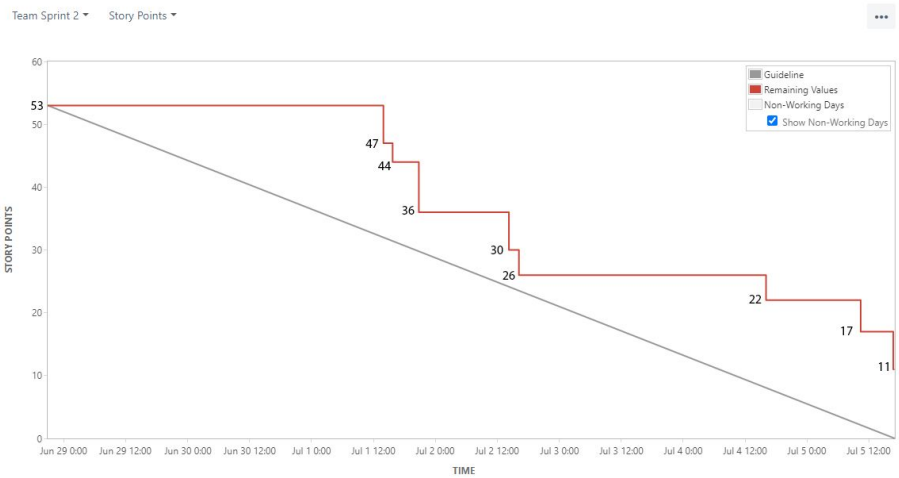




Start: 53 points

June 28 - July 5 (End)

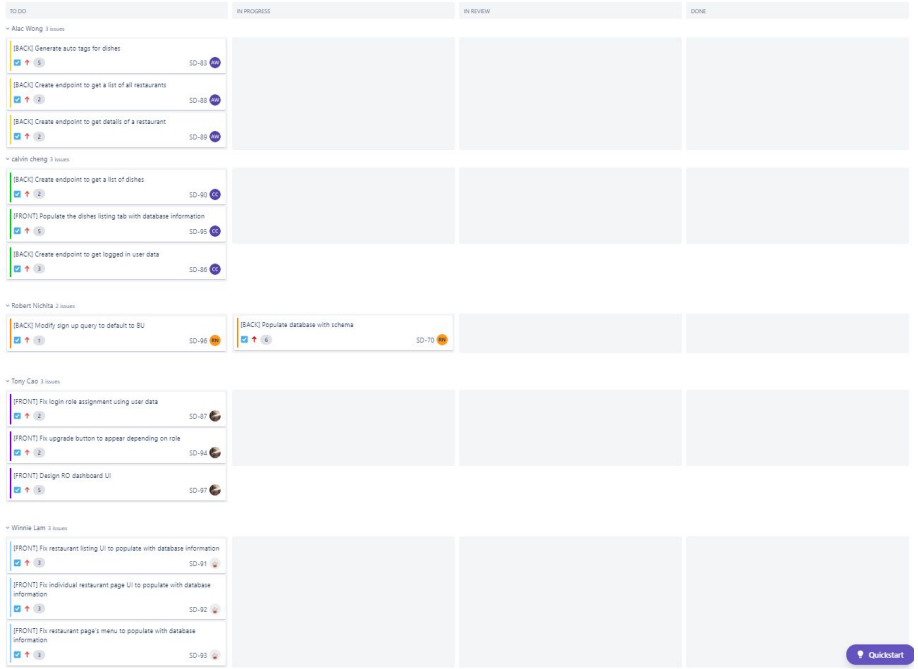


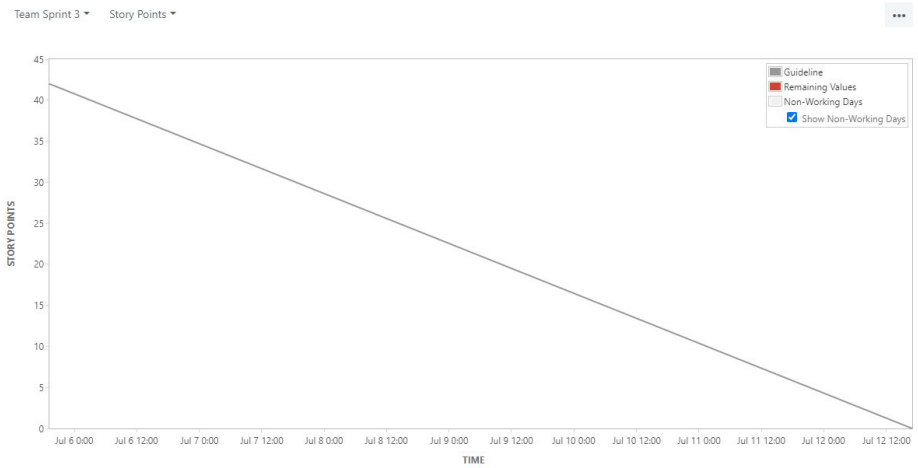


End: 11 points

Sprint 3

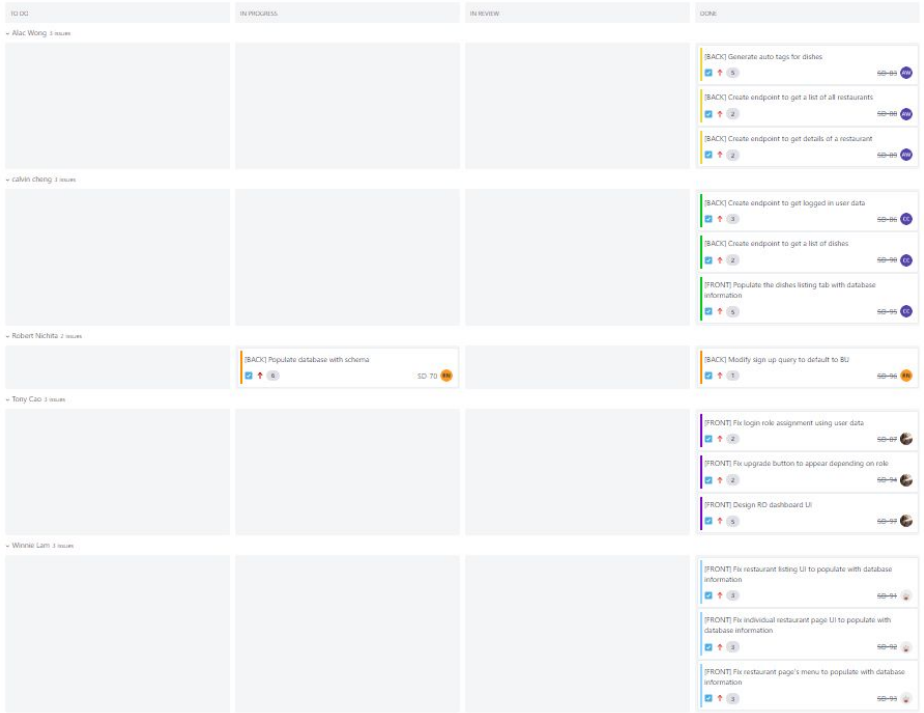
July 5 - July 12 (Initial)

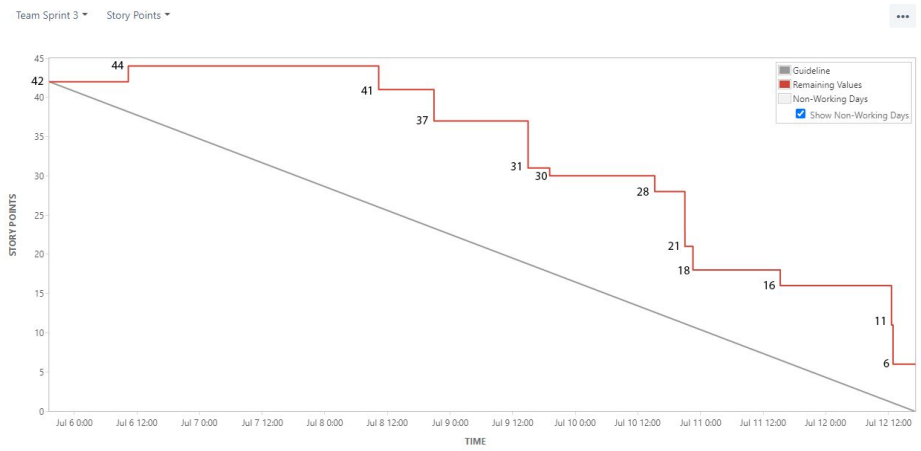




Start: 42 points

July 5 - July 12 (End)

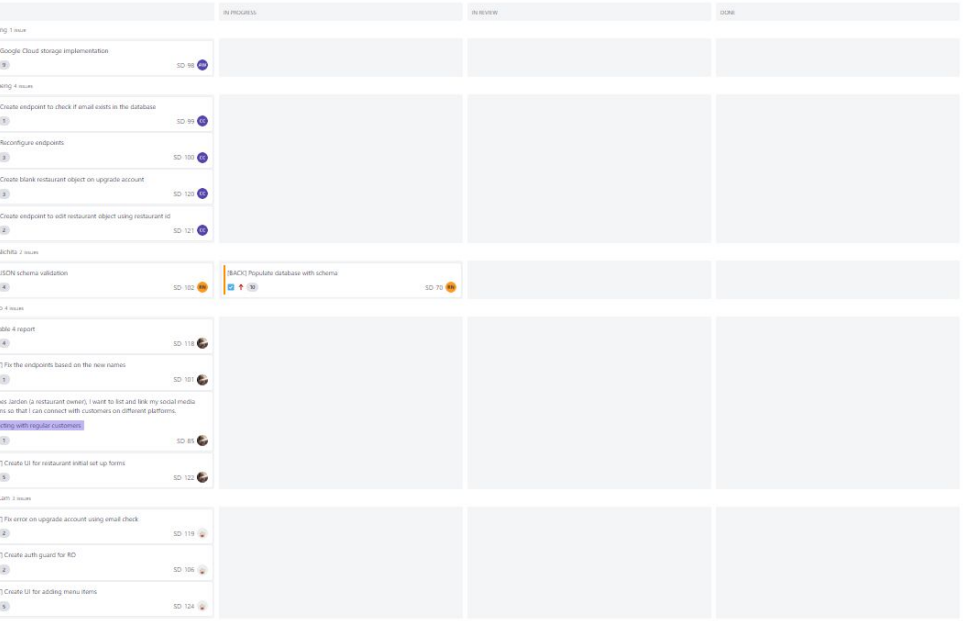


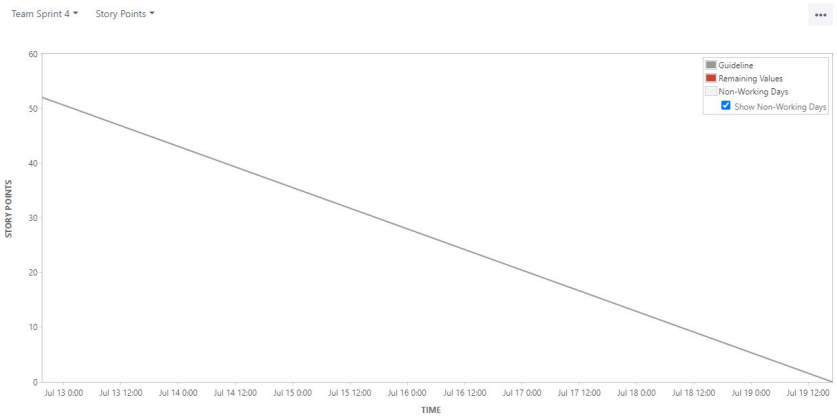


End: 4 points

Sprint 4

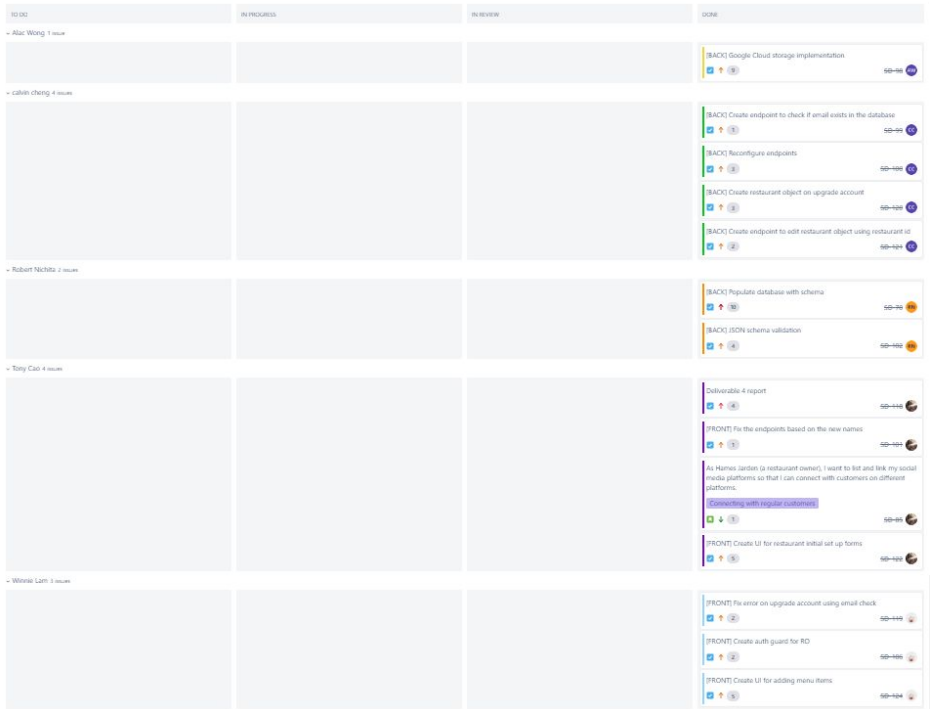
July 12 - July 19 (Initial)

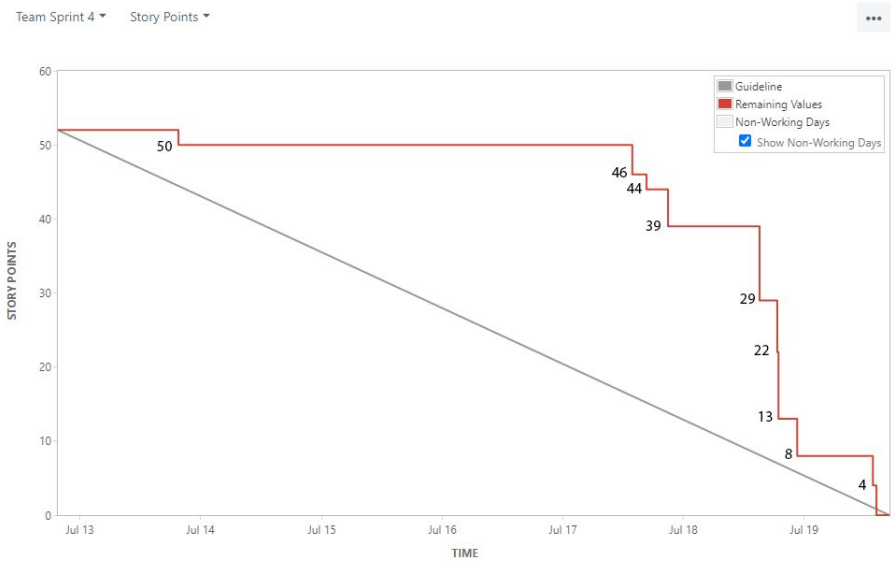




Start: 52 points

July 12 - July 19 (End)





End: 0 points

Git Repo

Git Formats

Branching - Feature -> Development -> Master

Branch Names - Jira-##/Name (Ex: SD-67/Main-Page-UI)

Commit Format - Jira-##:Description (Ex: SD-67:Added Component)

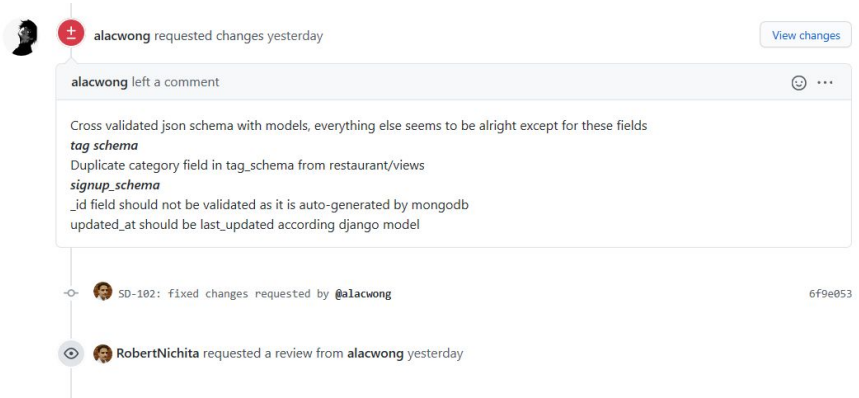
PR Format:
What was changed: Fixed JSON error from the previous merging error.
What was included: N/A (just a bug fix)
How to use: Run the code
How the design changed: The dishes now appear in the dish listing.
Additional Notes: N/A

Code Reviews

Code Reviews

- Done through PR
- Only request reviews from appropriate members
  - Feature -> Development (Requires at least 1 members approval)
  - Development -> Master (Requires all members' approval)

- Reviewer should review all documents changed and test "how to run"
  - In the case of mistakes: Submit a Change Request
    - Coder responsible for amending and submitting new PR
    - Requires Approval from Reviewer who sent Change Request



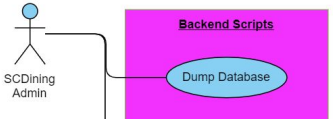
System Designs

Newly Requested Components/Use Cases

Geolocation

- Find address on map
- Locate nearby restaurants
- More research needs to be done before design

Data Dumps



- Export Database contents
- Select arbitrary data
- Command line

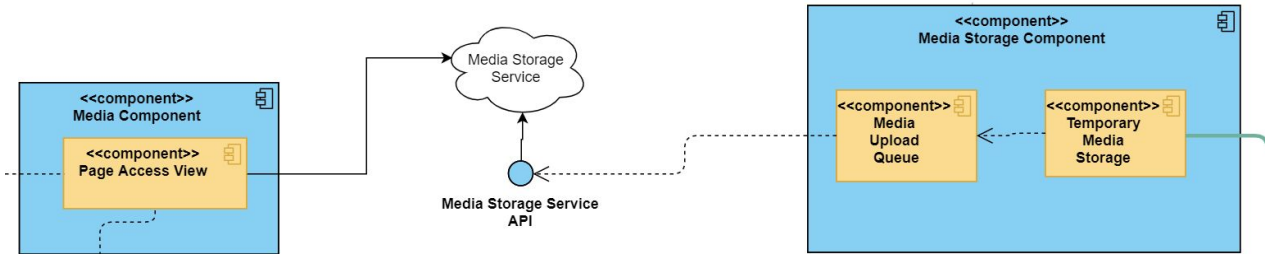
Modified Components / Use Cases

Media Upload

Previous design failed to properly capture the process of a media upload

New Design

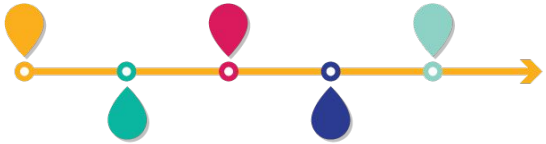
- Multipart Requests
- Temporarily hold file
- Default Assets



Verification and Validation

Client Meetings

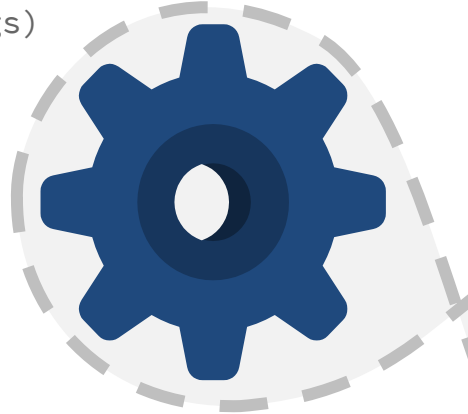
Demo and Feedback



1

Deployed Site:

Show latest progress (design, log in, sign up, restaurant listings)



2

Figma:

Advice on current and future designs for pages

3

Product Backlog:

Insights on which tasks would be the next priority

Additional user stories added

Tests

Testing structure

```
Team_08-project
    Server
        Django_app1
        Django_app2
            Django_file1
            . . .
            tests.py
        . . .
```

```python
def test_find_all_restaurant(self):
    """ Test if all restaurant documents are returned"""
    request = self.factory.get('/api/restaurant/get_all/')
    expected = [self.expected, self.expected2]
    actual = json.loads(view_response.get_all_restaurants_page
                        (request).content)['Restaurants']
    self.assertListEqual(expected,actual)
```

Python manage.py test {app}

Testing Table

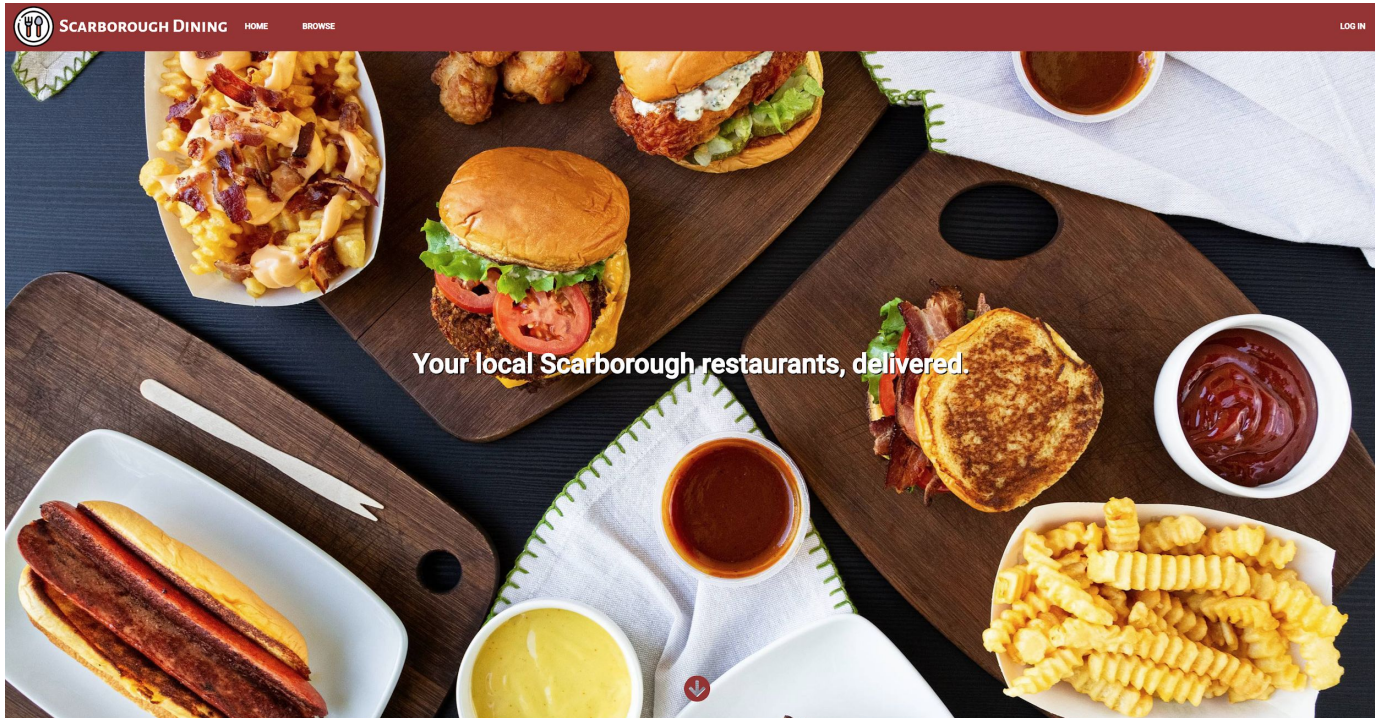
Evaluation Criteria	Criteria function must pass to pass the test case
(Possible Risk) Description	A description of possible risks associated with test case failure
Magnitude	Measurement of how dangerous possible risks associated with test case failure
Probability	Measurement of how likely possible risks may occur associated with test case failure
Priority	Priority of importance for a function to pass test case, priority is influenced by probability, magnitude and the function's priority (found in backlog) it is testing

Test Case Name	App	Test Suite	Evaluation Criteria	(Possible Risks) Description	Magnitude	Probability	Priority
test_signup	user	SDUserTestCases	Checks to see if The User has been inserted into the database	No new Users can be made	High	High	High

Demo of Latest Release

Deploy Link:

https://3.18.84.86



The End