Sprint 2 Review Document

Mar 14 – Apr 6

# The Flabbergasters

Michael Andrade

Aaron Fine

Abe Gunther

Matthew Lister

# Sprint Overview

During this sprint we are going to continue utilizing pair programming, and incremental prototyping to help mitigate the risks / challenges for this exercise. By choosing specific user stories we are going to begin interacting with the database. The goal by the end of the sprint is to add functionality to the key pages of our website. We intend to have each page interact with the database by reading from and/or writing to it. All remaining increments for the project should no longer use dummy data, it should now be provided by the server.

# Scrum Master

Michael Andrade

# Meeting Schedule

(@330pm unless otherwise noted)

Mar 14: Finish planning document

Mar 16: Standup meeting

Mar 19: Standup meeting

Mar 21: First Merge Party (Sombreros optional)

Mar 23: Standup meeting

Mar 26: Standup meeting

Mar 28: Standup meeting

Mar 30: Second Merge Party (Sombreros required)

Apr 2: Standup meeting

Apr 4: Practice demo / final configurations

Apr 6: Demo presentation

# Risks and Challenges

The challenge for this sprint will be to have the pages successfully interact with the database. The goal will be to convert all dummy variables to actual data provided by the server.

Risks for this sprint include difficulty with version control merging, especially regarding the SQLite database and that we are all modifying the same files. There will certainly be merge conflicts. We will mitigate this by scheduling time to merge while all parties are present.

# Sprint Plan

## User Story A

As a customer, I would like to order with or without giving a name (would be given a table or location instead)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Task Description | Owner | Estimated Time | Actual Time | Done (Y/N) |
| 0 | All members review Django Tutorial | MA | 4 | 3 | Y |
| 1 | Create Django database model for variables | MA | 2 | 1 | Y |
| 2 | Populate model and add to Django Admin | MA | 2 | 1 | Y |
| 3 | Templatize JS functions to fill in HTML with data | MA | 2 | 2 | Y |
| 4 | Create Python within view to query the database | MA | 2 | 5 | Y |
| 5 | Configure Django to use template | MA | 2 | 2 | Y |
| 6 | Load default values and set up event listeners to handle data | MA | 3 | 2 | Y |
| 7 | Set up form submissions/send | MA | 2 | 5 | Y |
| 8 | Reroute or refresh page automatically to avoid duplicate data | MA | 2 | 0 | N |

|  |  |
| --- | --- |
| Test Case A.1 | |
| Title | Django Admin database update |
| Pre-Conditions | Django models are created and database must be migrated. HTML must also be turned into template. |
| Test Steps | 1. Open Django Admin page 2. Navigate to each model 3. Create 1 item in each 4. Verify change in view |
| Expected Results | Added item shows up in view |

|  |  |
| --- | --- |
| Results for Test Case A.1 | |
| Actual Results (outcome) | Adding Database items creates instance on view. |
| Tester Name | Matt Lister |
| Date | 04/04/2018 |

|  |  |
| --- | --- |
| Test Case A.2 | |
| Title | Database Submit Test |
| Pre-Conditions | Django database initialized, Models are created, HTML submit works, Database is migrated |
| Test Steps | 1. Go to view to test 2. Create a sample submission and submit 3. Open Django Admin 4. Check to make sure that tables have incremented |
| Expected Results | Django Admin should reflect submission from template |

|  |  |
| --- | --- |
| Results for Test Case A.2 | |
| Actual Results (outcome) | Upon submitting data from view, admin page populates with expected result |
| Tester Name | Aaron Fine |
| Date | 04/06/2018 |

## User Story B

As a customer, I would like to post a rating and select a predefined statement for a particular dish, leave a custom review, or leave neither a statement or a review.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Task Description | Owner | Estimated Time | Actual Time | Done (Y/N) |
| 0 | All members review Django Tutorial | AG | 4 | 3 | Y |
| 1 | Create Django database model for variables | AG | 2 | 1 | Y |
| 2 | Populate model and add to Django Admin | AG | 2 | 1 | Y |
| 3 | Templatize JS functions to fill in HTML with data | AG | 2 | 4 | Y |
| 4 | Create Python within view to query the database | AG | 2 | 5 | Y |
| 5 | Configure Django to use template | AG | 2 | 2 | Y |
| 6 | Load default values and set up event listeners to handle data | AG | 3 | 1 | Y |
| 7 | Set up form submissions/send | AG | 2 | 6 | Y |
| 8 | Reroute or refresh page automatically to avoid duplicate data | AG | 2 | 0 | N |

|  |  |
| --- | --- |
| Test Case B.1 | |
| Title | Django Admin database update |
| Pre-Conditions | Django models are created and database must be migrated. HTML must also be turned into template. |
| Test Steps | 1. Open Django Admin page 2. Navigate to each model 3. Create 1 item in each 4. Verify change in view |
| Expected Results | Added item shows up in view |

|  |  |
| --- | --- |
| Results for Test Case B.1 | |
| Actual Results (outcome) | Adding Database items creates instance on view. |
| Tester Name | Matt Lister |
| Date | 04/04/2018 |

|  |  |
| --- | --- |
| Test Case B.2 | |
| Title | Database Submit Test |
| Pre-Conditions | Django database initialized, Models are created, HTML submit works, Database is migrated |
| Test Steps | 1. Go to view to test 2. Create a sample submission and submit 3. Open Django Admin 4. Check to make sure that tables have incremented |
| Expected Results | Django Admin should reflect submission from template |

|  |  |
| --- | --- |
| Results for Test Case B.2 | |
| Actual Results (outcome) | Upon submitting data from view, admin page populates with expected result |
| Tester Name | Aaron Fine |
| Date | 04/06/2018 |

# 

# 

# 

# User Story C As a manager, I would like to delete, create, and manage menu items

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Task Description | Owner | Estimated Time | Actual Time | Done (Y/N) |
| 0 | All members review Django Tutorial | ML | 4 | 3.5 | Y |
| 1 | Create Django database model for variables | ML | 2 | 2 | y |
| 2 | Populate model and add to Django Admin | ML | 2 | 4 | y |
| 3 | Templatize JS functions to fill in HTML with data | ML | 2 | 2 | Y |
| 4 | Create Python within view to query the database | ML | 2 | 6 | Y |
| 5 | Configure Django to use template | ML | 2 | 1 | Y |
| 6 | Load default values and set up event listeners to handle data | ML | 3 | 4 | Y |
| 7 | Set up form submissions/send | ML | 2 | 8 | y |
| 8 | Reroute or refresh page automatically to avoid duplicate data | ML | 2 | 0 | N |

|  |  |
| --- | --- |
| Test Case C.1 | |
| Title | Django Admin database update |
| Pre-Conditions | Django models are created and database must be migrated. HTML must also be turned into template. |
| Test Steps | 1. Open Django Admin page 2. Navigate to each model 3. Create 1 item in each 4. Verify change in view |
| Expected Results | Added item shows up in view |

|  |  |
| --- | --- |
| Results for Test Case C.1 | |
| Actual Results (outcome) | New orders are displayed on manager website |
| Tester Name | Abe and Mike |
| Date | 4/4/2018 |

|  |  |
| --- | --- |
| Test Case C.2 | |
| Title | Database Submit Test |
| Pre-Conditions | Django database initialized, Models are created, HTML submit works, Database is migrated |
| Test Steps | 1. Go to view to test 2. Create a sample submission and submit 3. Open Django Admin 4. Check to make sure that tables have incremented |
| Expected Results | Django Admin should reflect submission from template |

|  |  |
| --- | --- |
| Results for Test Case C.2 | |
| Actual Results (outcome) | Menu items can be added and deleted from the manager site |
| Tester Name | Abe |
| Date | 4/4/2018 |

## User Story D

As a cook, I need to be able to select an open order

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Task Description | Owner | Estimated Time | Actual Time | Done (Y/N) |
| 0 | All members review Django Tutorial | AF | 4 | 5 | Y |
| 1 | Create Django database model for variables | AF | 2 | 1 | y |
| 2 | Populate model and add to Django Admin | AF | 2 | 1 | y |
| 3 | Templatize JS functions to fill in HTML with data | AF | 2 | 2: MA/ML | Y |
| 4 | Create Python within view to query the database | AF | 2 | 3 | Y |
| 5 | Configure Django to use template | AF | 2 | 7 | Y |
| 6 | Load default values and set up event listeners to handle data | AF | 3 | 1: MA/ML | Y |
| 7 | Set up form submissions/send | AF | 2 | 1 | y |
| 8 | Reroute or refresh page automatically to avoid duplicate data | AF | 2 | 0 | N |

|  |  |
| --- | --- |
| Test Case D.1 | |
| Title | Django Admin database update |
| Pre-Conditions | Django models are created and database must be migrated. HTML must also be turned into template. |
| Test Steps | 1. Open Django Admin page 2. Navigate to each model 3. Create 1 item in each 4. Verify change in view |
| Expected Results | Added item shows up in view |

|  |  |
| --- | --- |
| Results for Test Case D.1 | |
| Actual Results (outcome) | Adding Database items creates instance on view. |
| Tester Name | Matt Lister |
| Date | 04/04/2018 |

|  |  |
| --- | --- |
| Test Case D.2 | |
| Title | Database Submit Test |
| Pre-Conditions | Django database initialized, Models are created, HTML submit works, Database is migrated |
| Test Steps | 1. Go to view to test 2. Create a sample submission and submit 3. Open Django Admin 4. Check to make sure that tables have incremented |
| Expected Results | Django Admin should reflect submission from template |

|  |  |
| --- | --- |
| Results for Test Case D.2 | |
| Actual Results (outcome) | Upon submitting data from view, admin page populates with expected result |
| Tester Name | Mike |
| Date | 04/06/2018 |

# Product Backlog (this is everything from product backlog)

|  |  |  |
| --- | --- | --- |
| Index | Use Case | Sprint Reference |
| 1 | As a customer, I would like to be able create an account |  |
| 2 | As a customer, I would like to be able to log into or out of my account |  |
| 3 | As a customer, I would like to sort the menu in various ways |  |
| 4 | As a customer, I would like to post a rating and select a predefined statement for a particular dish, leave a custom review, or leave neither a statement or a review. | Sprint 2 |
| 5 | As a customer, I would like to pay a bill |  |
| 6 | As a customer, I would like to order with or without giving a name (would be given a table or location instead) | Sprint 2 |
| 7 | As a customer, I would like to add items to my order |  |
| 8 | As a customer, I would like to request service from a waiter / waitress |  |
| 9 | As a customer, I want to order something after I’ve paid my bill |  |
| 10 | As a customer, I want to be able to change my table / location at any time |  |
| 11 | As a customer, I want to be able to order something for another table / location |  |
| 12 | As a customer, I want to use a phone app compatible with android or ios with the same functionality as the website (As time allows -- Stretch goal) |  |
| 13 | As a customer, I want to request a order-ID |  |
| 14 | As a customer, I want to be able to view the menu without logging in |  |
| 15 | As a cook, I need to be able to select an open order | Sprint 2 |
| 16 | As a cook, I need to view a selected order |  |
| 17 | As a cook, I need to sort open orders by at least one criteria |  |
| 18 | As a cook, I need to indicate that an entire order has finished cooking |  |
| 19 | As a cook, I need to signal that an order is ready for pickup |  |
| 20 | As a cook, I may need to indicate that part of an order has finished cooking |  |
| 21 | As a cook, I need to signal the waiter / waitress that there is a problem or clarification needed with the order |  |
| 22 | As a cook, I need to be able to update inventory counts |  |
| 23 | As a cook, I need to be able to place no-cost meal orders (subject to daily limits) |  |
| 24 | As a cook, I need the arrival of new orders to be clearly signaled |  |
| 25 | As a cook, I need to be able to select sub-components of an overall order |  |
| 26 | As a cook, I need to be able to logout |  |
| 27 | As a cook, I need default prep orders available at the start of the workday (cut up 10 lbs of onions, make 20 gallons of soup, etc.) (If time allows - - Stretch Goal) |  |
| 28 | As a cook, I need default cleanup orders to come up at the end of the shift (clean grill, mop floor, etc.)(If time allows - - Stretch Goal) |  |
| 29 | As a cook, I need the system to interface with external buttons (hit a large button for next order, complete order, etc.)(If time allows - - Stretch Goal) |  |
| 30 | As a cook, I need the next open order to be automatically selected when I complete the current order |  |
| 31 | As a cook, I need to see how long it has been since an order was placed |  |
| 32 | As a cook, I need to see how much more time I have to complete the order to be within my target time (if the admin should be able to say that most order should be completed within 10 minutes of ordering then the cook should see that they only have 5 minutes left to complete the order)(If time allows - - Stretch Goal) |  |
| 33 | As a cook, I need to see how much overage time an order has taken beyond the target time (if the target time is 10 minutes and it took 11 min 20 seconds to complete the system should show -1 min 20 seconds for target time remaining)(If time allows - - Stretch Goal) |  |
| 34 | As a cook, I need to see how much time remaining until my next break or end of shift(If time allows - - Stretch Goal) |  |
| 35 | As a waiter / waitress I need to have the duration of login work as a timecard (If time allows - - Stretch Goal) |  |
| 36 | As a waiter / waitress I need to be able to issue a new customer order-ID |  |
| 37 | As a waiter / waitress I need to place an order for a customer |  |
| 38 | As a waiter / waitress I need to identify which table / location an order relates to |  |
| 39 | As a waiter / waitress I need to be able to pull up an ingredient list for menu items (If time allows - - Stretch Goal) |  |
| 40 | As a waiter / waitress I need to be able to place zero cost orders (subject to approval) |  |
| 41 | As a waiter / waitress I need to get notified when certain orders are available for pickup from the kitchen (i.e. I don’t care about orders not in my section) |  |
| 42 | As a waiter / waitress I need to be able to select or be assigned to a section of locations / tables |  |
| 43 | As a waiter / waitress I need to know what locations / tables have service requests |  |
| 44 | As a waiter / waitress, I would like to switch a bill from one customer to another |  |
| 45 | As a waiter / waitress, I would like to split or combine different bills |  |
| 46 | As a waiter / waitress / cook, I would like to view items that are marked not available |  |
| 47 | As a waiter / waitress / customer, I would like to view the daily special menu items(If time allows - - Stretch Goal) |  |
| 48 | As a manager, I would like to designate particular menu items as a daily special(If time allows - - Stretch Goal) |  |
| 49 | As an employee, I would like to log in and out |  |
| 50 | As a manager, I would like to manage other user accounts |  |
| 51 | As a manager, I would like to delete, create, and manage menu items | Sprint 2 |
| 52 | As a manager, I would like to delete, moderate, and approve user feedback before display to other users |  |
| 53 | As a manager, I would like to be able to manage inventory |  |
| 54 | As a manager, I need moderate payment requirements for some order types. The orders can be set such that the customer must pay before an order is placed, that the customer must pay after they eat but before they leave, or for net 30 (or other terms).(If time allows - - Stretch Goal) |  |
| 55 | As a manager, I need the system to interface with my accounting system (inventory, payroll, etc.)(If time allows - - Stretch Goal) |  |
| 56 | As a manager, I need the system to interface with third party order and delivery systems like ChowHound, PostMates, etc.(If time allows - - Stretch Goal) |  |
| 57 | As a manager, I need to be able to override details about an open order (menu items, price, location, etc.) |  |
| 58 | As a manager, I should be able to pull up a history of any order. |  |
| 59 | As a manager, I need a daily summary report on all activity at the restaurant |  |
| 60 | As a manager, I need an inventory status report |  |
| 61 | As a manager, I need a projected inventory report based on current inventory levels and historical data (I.E. do I have enough inventory for tomorrow?) (If time allows - - Stretch Goal) |  |
| 62 | As a manager, I need to be able to mark a menu item as currently unavailable |  |
| 63 | As a manager, I want to be able to see customer feedback in near real-time. |  |
| 64 | As a manager, I want customer feedback translated into sentiment. (If time allows - - Stretch Goal) |  |
| 65 | Upon first time login, users will be able to login and subscribe to receive notifications of their choice within 60 seconds of the application loading. |  |
| 66 | Application shall be available on the Google Play Store.(If time allows - - Stretch Goal) |  |
| 67 | The app must allow for at least 4+ simultaneous users that are pulling information from the database while others are creating and submitting events to the database at any given time. |  |
| 68 | The app must follow best practices to prevent sql injection, cross site scripting attacks or other common hacking techniques that can harm an application. |  |
| 69 | This project shall utilize a two tier Client/Server architecture. I.e. will not use proxy server | Sprint 1 |
| 70 | The project will have a logical design of hub and spokes. | Sprint 1 |
| 71 | All communications between the clients and server will be encrypted. |  |
| 72 | Aside from the guest/customer view, all user views will be password protected. |  |
| 73 | All views shall use a session cookie (life to not exceed 24 hours) |  |
| 74 | The server will validate all user requests. |  |
| 75 | Unique transaction IDs are generated for every transaction. |  |
| 76 | Party name on orders can be blank |  |
| 77 | Server will provide data to mobile client as JSON (If time allows - - Stretch Goal) |  |
| 78 | Mobile app will handle formatting and presentation of server data. (If time allows - - Stretch Goal) |  |
| 79 | Server will provide formatting through html/css to clients using a web browser. | Sprint 1 |
| 80 | Order and inventory data will be preserved for a user specified amount of time. |  |
| 81 | Changes to orders must be maintained for auditability. |  |
| 82 | Old records must be purged from the system after a user specified amount of time. |  |
| 83 | It should be configurable to allow orders through or not depending on inventory levels. I.E. if the inventory shows you are out of soup then it would depend on the configuration if a customer could order a bowl of soup. |  |
| 84 | The system will support access cards or other tokens for login instead of just username and password (If time allows - - Stretch Goal) |  |
| 85 | The source code will undergo static analysis |  |
| 86 | Within 30 seconds from time of order, new orders will show up in the cook view |  |