

## **DELIVERABLE 5**



Team\_01(GitHub)  
Anson Feng  
Jiawei Qiao  
Vanessa Pierre  
Yolanda Su

# Table Of Contents

<b>Project Backlog</b>	<b>3</b>
<b>Changes to Project Backlog</b>	<b>8</b>
<b>Release Plan</b>	<b>11</b>
<b>Sprint 5: Sprint Plan</b>	<b>12</b>
<b>Sprint 5: Acceptance Criteria</b>	<b>15</b>
<b>Sprint 5: Team Task Assignments</b>	<b>20</b>
<b>Sprint 5: Task Board</b>	<b>22</b>
<b>Sprint 5: Burndown Chart</b>	<b>23</b>
<b>Sprint 6: Sprint Plan</b>	<b>25</b>
<b>Sprint 6: Acceptance Criteria</b>	<b>30</b>
<b>Sprint 6: Team Task Assignments</b>	<b>34</b>
<b>Sprint 6: Task Board</b>	<b>36</b>
<b>Sprint 6: Burndown Chart</b>	<b>39</b>
<b>Sprint 7: Sprint Plan</b>	<b>42</b>
<b>Sprint 7: Acceptance Criteria</b>	<b>47</b>
<b>Sprint 7: Team Task Assignments</b>	<b>53</b>
<b>Sprint 7: Task Board</b>	<b>56</b>
<b>Sprint 7: Burndown Chart</b>	<b>59</b>
<b>High-level Architecture</b>	<b>62</b>
<b>Changes to High Level Architecture</b>	<b>75</b>
<b>Verification</b>	<b>76</b>
<b>Validation</b>	<b>77</b>
<b>Project Velocity</b>	<b>79</b>
Estimated Velocity	79
Actual Velocity	79
<b>Our Project Throughout The Term</b>	<b>80</b>
<b>Deliverable 5 Retrospective</b>	<b>82</b>
<b>Release</b>	<b>83</b>

# Project Backlog

1 point = 1 working hour

Greyed out rows represent stories that are already done.

## Persona 1: David (Restaurant Owner)

Type	Priority	Story	Back-end Points	Front-end Points	Test Automation Points	Total Story Points
Registration						
Story	Critical	As David (restaurant owner), I want to be able to register my account by using my name, email, password and my restaurant name and address.	6.5	4	3.5	14
Coupons						
Story	High	As David (restaurant owner), I want to create a coupon with a name, redemption cost in points, description, and an optional expiration and begin date.	4	3	2.5	9.5
Story	High	As David (restaurant owner), I want to be able to see all my restaurant's current coupons.	1	6	2	9
Story	High	As David (restaurant owner), I want to be able to delete an existing coupon.	1	1	2	4
Story	High	As David (restaurant owner), I want a graphical interface allowing me to see how many redeemed coupons customers have at my restaurant.	1	2	3	6
Story	High	As David (restaurant owner) I want to be able to see what coupons have been used at my restaurant by customers.	1.5	1.5	2	5
Achievements						
Story	High	As David (restaurant owner), I want to be able to add new achievements to my restaurant by inputting name, description, their rewarding experience and points, and their requirement within	2	2.5	3	7.5

		two types: 1. Spending X dollars in one visit; 2. Purchasing what "Item" X time.				
Story	High	As David (restaurant owner), I want to be able to create achievements in the forms: 1. Visit in a group of x people; 2. Visit x times between date1 and date2	.25	4	3	9.5
Story	High	As David (restaurant owner), I want to be able to see all my restaurant's current achievements.	1	2	2.5	5.5
Story	High	As David (restaurant owner), I want to be able to delete achievements from my restaurant.	1	1	1.5	3.5
Story	Low	As David (restaurant owner) I want to be able to see how many customers have started and how many customers have completed each of the achievements at my restaurant, including expired achievements.	2	2	2	6
Restaurant Settings						
Story	Low	As David (restaurant owner) I want to be able to see the top 3 owned coupons, top 3 used coupons and top 3 achievements at my restaurant on my home page	3	2	3	8
Story	Low	As David (restaurant owner), I want to be able to edit the name and address of my restaurant via my profile page.	1	1.5	1	4.5
Employees						
Story	Critical	As David (restaurant owner) I want to be able to create an employee account for my employee by inputting their name, email, and password.	3	2	2.5	7.5
Story	High	As David (restaurant owner), I want to be able to see all employee accounts associated with my restaurant	2	3	3	8
Story	High	As David (restaurant owner), I want to be able to delete accounts associated with my restaurant.	2	2	1	5
Story	High	As David (restaurant owner), I want to promote or demote employees to/from "manager" type employees, where	1.5	3	1	5.5

		"manager" employees can access all features of an owner except for promotion/demotion.				
Customer Level						
Epic	Low	As David (restaurant owner) I want to be able to set the additional reward points earned by customers based on their user level at my restaurant.	4	4	5	13
Story	Medium	As David (restaurant owner) I want to be able to set the amount of bonus points customers can earn at my restaurant at level thresholds of my choice, e.g. 100 points earned for reaching the level 5 threshold.	3	3	3	9
Story	Medium	As David (restaurant owner) I want to be able to set coupons to be available only to users of a minimum level or higher.	3	2	3.5	8.5

## Persona 2: Daniel (Employee)

Type	Priority	Story	Back-end Points	Front-end Points	Test Automation Points	Total Story Points
QR Code Scanning						
Story	High	As Daniel (employee) I want to be able to scan customer coupon QR codes to mark a customer's coupon used.	4	1	3	8
Story	High	As Daniel (employee) I want to be able to scan customer achievement QR codes to update customer progress on a given achievement.	1.5	1	3	5.5
Story	Low	As Daniel (employee), I want to be able to see the name and address of the restaurant I work at on my home page	0.5	1	0	1.5

### Persona 3: Kevin (Customer)

Type	Priority	Story	Back-end Points	Front-end Points	Test Automation Points	Total Story Points
Registration						
Story	Critical	As Kevin (customer), I want to be able to register my account using my name, email and password.	0.5	0.5	2	3
Search						
Story	Critical	As Kevin (customer), I want a search bar to search for restaurants that are available on the app.	3	2.5	3	8.5
Coupons						
Story	High	As Kevin (customer) I want to be able to see available coupons for exchange at a specific restaurant.	1	1.5	2	4.5
Story	High	As Kevin (customer) I want to be able to purchase coupons by redeeming my points.	2	2.5	2	6.5
Story	High	As Kevin (customer) I want to be able to see what coupons I already redeemed.	1	1	2.5	4.5
Achievements						
Story	High	As Kevin (customer) I want to be able to see what achievements are available.	1	1	3.5	5.5
Story	High	As Kevin (customer), I want a progress bar for each achievement that tracks how close I am to completing the achievement.	1	3	2	6
Story	Medium	As Kevin (customer) I want to be able to see my 3 most recently updated achievements on my home page.	2	1.5	3	6.5
Points						
Story	High	As Kevin (customer), I want a display for each restaurant that tracks how many points I have earned at that restaurant so far.	1	1	2	4

Customer Level						
Story	High	As Kevin (customer), I want to be able to see my current customer level and the experience needed to reach the next level at a given restaurant.	2.5	3	2	7.5
Story	Medium	As Kevin (customer), I want to see user leaderboards for each restaurant on the app so that I can compare my level with other customers and compete to see who has the highest level at a given restaurant.	2	4	3	9
Customer Profile						
Story	Medium	As Kevin (customer), I want a profile page containing my name, email and a list of links to restaurants that I have favoured.	2	2.5	4	8.5

## Changes to Project Backlog

**Story:** As David (restaurant owner), I want to be able to create achievements in the forms: 1. Visit in a group of x people; 2. Visit x times between date1 and date2:

This story was added by ourselves as we originally only had 2 types of achievements, and we decided after team discussion that we should not limit the types of achievement to be only those two, especially as in the client requirements they had more than 2 achievement types as examples. We thus added this story to increase the number of types of achievements, and these types were later approved by the clients.

**Story:** As David (restaurant owner) I want to be able to see how many customers have started and how many customers have completed each of the achievements at my restaurant, including expired achievements:

This story is added and based on one of the previous stories for seeing coupon statistics. According to clients' requirements and our team discussion, we decided to add this task so that we can show more info to the restaurant owner, which is useful for the business side of the product.

**Story:** As David (restaurant owner) I want to be able to see the top 3 owned coupons, top 3 used coupons and top 3 achievements at my restaurant on my home page:

This story was added as we decided to populate the home page of our site so that it did not appear empty, and it was also from one of the previous Epics for allowing the owner to see a general restaurant interface. We didn't have anything showing on the home page before. This feature was approved by the client.

**Story:** As David (restaurant owner), I want to be able to edit the name and address of my restaurant via my profile page.

This story was also split from the Epic for allowing the owner to see a general restaurant interface, and we made it more specific, detailed and testable compared to the epic after team discussion. This feature was approved by the client.

**Story:** As David (restaurant owner) I want to be able to set the amount of bonus points customers can earn at my restaurant at level thresholds of my choice, e.g. 100 points earned for reaching the level 5 threshold.

This is a story added based on clients' requirement that gamifying the app, and was split from the epic "As David (restaurant owner) I want to be able to set the additional reward points earned by customers based on their user level at my restaurant." and written to be more specific.



**Story:** As David (restaurant owner) I want to be able to set coupons to be available only to users of a minimum level or higher.

This is also a story added based on clients' requirement that gamifying the app, and was split from the epic "As David (restaurant owner) I want to be able to set the additional reward points earned by customers based on their user level at my restaurant." and written to be more specific.

**Story:** As Daniel (employee), I want to be able to see the name and address of the restaurant I work at on my home page.

This story was added as we realized the employee home page did not display anything, and there was also nowhere on the site that showed what restaurant the logged in employee account was associated with. This feature enables employees to see this information from the home page.

**Story:** As Kevin (customer) I want to be able to see my 3 most recently updated achievements on my home page.

This story replaced "As Kevin (customer) I want to be able to see a confirmation screen with the number of points earned after completing an achievement." from the previous version of the backlog, as we decided to change the way we implemented the notification that an achievement has been updated and/or completed.

**Story:** As David (restaurant owner), I want to promote or demote employees to/from "manager" type employees, where "manager" employees can access all features of an owner except for promotion/demotion.

This story is based on the client's request. They asked for this feature in the feedback from our sprint 6 demo. In order to meet this requirement, we added this story.

**Story:** As Kevin (customer), I want a profile page containing my name, email and a list of links to restaurants that I have favoured.

We changed part of this story from "list of links to restaurants I visited" to "list of links to restaurants that I have favoured", as we felt the list may eventually become cluttered and difficult to navigate for a long-time user of our app. Having a favouriting system instead allows customers to pick and choose which restaurants they wanted to add to their list of shortcuts.

**Epic:** As David (restaurant owner), I want a menu system allowing me to upload, preview and modify my menu onto the application.

We removed this epic from the backlog, as upon further team discussion we realized that this potential feature did not really fulfill any of the client requirements at all, as it was not related to gamification or any business side needs.

**Epic:** As Kevin (customer), I want to be able to customize my user character with equipable character items earned from levelling up at restaurants.

We removed this epic from the backlog, as upon further team discussion we decided this feature adds very little value in the required "gamification" area, and the client originally did not show that much enthusiasm towards this feature idea.

**Story:** As David (restaurant owner), I want a sample coupon template for customer coupons, so that I can set up coupons more easily.

We removed this story from the backlog, as upon team discussion and based on client feedback we felt our coupon creation feature was easy to use and straightforward enough that this story didn't add much value.

**Story:** As David (restaurant owner), I want to be able to use existing coupons as a template for new coupons.

As in the previous story, we removed this story from the backlog, as upon team discussion and based on client feedback we felt our coupon creation feature was easy to use and straightforward enough that this story didn't add much value.

**Story:** As David (restaurant owner), I want a sample achievement template for customer achievements, so that I can set up achievements more easily

We removed this story from the backlog, as upon team discussion and based on client feedback we felt our achievement creation feature was easy to use and straightforward enough that this story didn't add much value.

**Story:** As David (restaurant owner), I want to be able to modify achievements' name, description, experience, points, and requirements from my restaurant.

After hearing from the client about the possible issues surrounding discontinuing an achievement that customers have already started, we realized this story was troublesome from a business perspective, as being able to change the details of an achievement without warning when a customer has already begun to make progress on that achievement could upset customers and cause disputes. As a result, we removed this story from the backlog.

## Release Plan

During this deliverable, we changed our sprint start and end time to Monday at 3:00PM, as we felt that we had more time to work on weekends and having the sprint end mid-Sunday negatively affected our release quality and the amount of stories we could complete and review. As a result of this changed sprint end time, we moved our combined sprint review, retro, and sprint planning meeting to 4:00-6:00PM on Monday. The length of our sprints remained unchanged at 1 week long.

## Sprint 5: Sprint Plan

Priority	Story	Points	Tasks	Points
High	As David (restaurant owner), I want to be able to add new achievements to my restaurant by inputting name, description, their rewarding experience and points, and their requirement within two types: 1. Spending X dollars in one visit; 2. Purchasing what "Item" X times.	5	T083: Debug feature UI	5
High	As David (restaurant owner) I want to be able to see what coupons have been used at my restaurant by customers.	5	T084: Add column to the redeemed coupon table indicating whether the coupon is used or unused	0.5
			T085: Code backend function for retrieving used or unused coupons	0.5
			T086: Modify existing restaurant redeemed coupon page to only show unused coupons	0.5
			T087: Design UI for listing used coupons	1
			T088: Implement used coupon list UI	0.5
			T089: Automate tests	2
High	As David (restaurant owner), I want to be able to see all my restaurant's current achievements.	5.5	T090: Code backend function for retrieving a restaurant's achievement data	1
			T091: Design UI for listing restaurant achievements	0.5
			T092: Implement achievement list UI for restaurants	1.5
			T093: Automate tests	2.5

High	As David (restaurant owner), I want to be able to delete achievements from my restaurant	3.5	T094: Code backend functions for deleting achievements	0.5
			T095: Design UI for deleting achievements	0.5
			T096: Implement achievement deletion UI	1
			T097: Automate tests	1.5
High	As Daniel (employee) I want to be able to scan customer coupon QR codes and mark the coupon used	8	T098: Code functions for generating QR codes based on redeemed coupon usage URL	1.5
			T099: Code backend functions for marking a coupon as used	0.5
			T100: Code redirects to successful vs unsuccessful scanning based on user's employee and restaurant permissions.	1
			T101: Add 'generate scannable coupon QR code' button to coupons in the customer redeemed coupon list UI	0.5
			T102: Design UI for code scanning success vs failure pages	0.5
			T103: Implement code scanning success vs failure page UIs	1
High	As Kevin (customer) I want to be able to see what coupons I already redeemed.	4.5	T104: Automate tests	3
			T105: Code backend function for retrieving coupons redeemed by a customer	1
			T106: Design redeemed coupon list UI for customers	0.5

			T107: Implement redeemed coupon list UI	0.5
			T108: Automate tests	2.5
High	As Kevin (customer) I want to be able to purchase coupons by redeeming my points.	6.5	T109: Code back-end functions for checking points balance and redeeming coupons	2
			T110: Design UI for redeeming coupons using points at a restaurant	1
			T111: Implement coupon redemption UI	1.5
			T112: Automate tests	2
High	As Kevin (customer) I want to be able to see what achievements are available	5.5	T113: Set up achievement progress table	0.5
			T114: Code functions for filtering achievements based on whether the customer has made non-zero progress	0.5
			T115: Design UI for listing restaurant achievements for customers	0.25
			T116: Implement restaurant achievement list UI for customers	0.75
			T117: Automate tests	3.5

## Sprint 5: Acceptance Criteria

Story	Acceptance Criteria
<p>As David (restaurant owner), I want to be able to add new achievements to my restaurant by inputting name, description, their rewarding experience and points, and their requirement within two types: 1. Spending X dollars in one visit; 2. Purchasing what "Item" X times.</p>	<p><b>Scenario:</b> The restaurant owner tries to add an achievement with negative experience and/or negative points.</p> <p>Given that the restaurant owner has entered a negative experience value and/or negative points value for their new achievement, when the owner clicks the create button, an error message is shown and no new achievement is shown on the achievement list.</p> <p><b>Scenario:</b> The restaurant owner tries to add an achievement with non-negative experience and non-negative points.</p> <p>Given that the restaurant owner has entered a non-negative experience value and non-negative points value for their new achievement, when the owner clicks the create button, a new achievement appears on the achievement list with all values matching the given input values.</p> <p><b>Scenario:</b> The restaurant owner wants to add an achievement of the form "Buy item x times".</p> <p>Given that the restaurant owner wants to add an achievement of the form "Buy item x times", when the owner visits the create achievement page, there is an option to choose an achievement template that takes in an integer x and an item name in the form "Buy item x times".</p> <p><b>Scenario:</b> The restaurant owner tries to add an achievement of the form "Buy item x times" and sets x as a negative value.</p> <p>Given that the restaurant owner has filled in an achievement template of the form "Buy item x times" with a negative x value, when the owner clicks the create button, an error message is shown and no new achievement is shown on the achievement list.</p> <p><b>Scenario:</b> The restaurant owner wants to add an achievement of the form "Spend x dollars in a single visit".</p> <p>Given that the restaurant owner wants to add an achievement of the form "Spend x dollars in a single visit", when the owner visits the create achievement page, there is an option to choose an achievement template that takes in an integer x in the form "Spend x dollars in a single visit".</p>
<p>As David (restaurant owner) I want to be able to see what coupons have been used at my restaurant by customers.</p>	<p><b>Scenario:</b> The owner checks the coupon statistics when the restaurant has no coupons.</p> <p>Given that the user is an owner, when the user visits the coupon statistics page for their restaurant, no coupons are</p>

	<p>listed.</p> <p><b>Scenario:</b> The owner checks the coupon statistics when the restaurant has some active coupons.</p> <p>Given that the user is an owner, when the user visits the coupon statistics page for their restaurant, all coupons at my restaurant are listed by name, description, number of copies held by customers, and number of copies used by customers, with a status field marking the coupon as active.</p> <p><b>Scenario:</b> The owner checks the coupon statistics when some coupons are expired.</p> <p>Given that the user is an owner, when the user visits the coupon statistics page for their restaurant, all expired coupons at my restaurant are included in the coupon list in the same format as the active coupons, except the status field marks the coupon as expired.</p> <p><b>Scenario:</b> The owner checks the coupon statistics when some coupons have been deleted.</p> <p>Given that the user is an owner, when the user visits the coupon statistics page for their restaurant, all deleted coupons at my restaurant are included in the coupon list in the same format as the active coupons, except the status field marks the coupon as deleted.</p> <p><b>Scenario:</b> A single customer has redeemed multiple copies of the same coupon.</p> <p>Given that a single customer has redeemed a coupon multiple times, when an owner checks the statistics of that coupon, each copy redeemed by the same user is counted separately in the number of copies held by customers.</p> <p><b>Scenario:</b> A single customer has used multiple copies of the same coupon.</p> <p>Given that a single customer has used a coupon multiple times, when an owner checks the statistics of that coupon, each copy used by the same user is counted separately in the number of copies used by customers.</p> <p><b>Scenario:</b> A customer has recently used a coupon at a restaurant.</p> <p>Given a customer has just used a coupon at the owner's restaurant, when an owner checks the statistics of that coupon, the number of copies held will be lower by 1 and the number of copies used will be higher by 1.</p>
As David (restaurant owner), I want to be able to	<p><b>Scenario:</b> The owner checks achievements when there are no achievements at the restaurant.</p> <p>Given that the user is an owner, when the user checks the</p>



<p>see all my restaurant's current achievements.</p>	<p>achievement list page of their restaurant, no achievements are listed.</p> <p><b>Scenario:</b> The owner checks achievements when there are some active achievements at the restaurant.</p> <p>Given that the user is an owner, when the user checks the achievement list page of their restaurant, all active achievements at their restaurant are listed with name, description, points reward amount, and experience reward amount information.</p> <p><b>Scenario:</b> The owner checks achievements when there are some expired achievements at the restaurant.</p> <p>Given that some achievements at a restaurant has expired, when the owner checks the achievements list page of their restaurant, no expired achievements are displayed.</p>
<p>As David (restaurant owner), I want to be able to delete achievements from my restaurant</p>	<p><b>Scenario:</b> The owner tries to delete an achievement from the restaurant.</p> <p>Given that there is at least one achievement at a restaurant, when the owner clicks delete on an achievement, the achievement no longer appears on the restaurant achievement list.</p>
<p>As Daniel (employee) I want to be able to scan customer coupon QR codes and mark the coupon used</p>	<p><b>Scenario:</b> A user scans a coupon QR code and they are not an owner or employee.</p> <p>Given that a user is a customer, when the user tries to scan a coupon QR code, the page displays an error message saying they do not have permission to scan QR codes.</p> <p><b>Scenario:</b> An employee or owner tries to scan the QR code of a coupon at a restaurant they do not work at.</p> <p>Given that a user is an employee or owner, when the user tries to scan a coupon QR code for a coupon at a restaurant they do not work at, the page displays an error message saying they do not have permission to scan QR codes at that coupon's restaurant.</p> <p><b>Scenario:</b> An employee or owner tries to scan the QR code of a coupon at their restaurant that has not reached its start date.</p> <p>Given that a user is an employee or owner, when the user tries to scan a coupon QR code for a coupon at their restaurant that has not reached its start date, the page displays an error message saying the coupon cannot be scanned as it is still before the coupon's start date.</p> <p><b>Scenario:</b> An employee or owner tries to scan the QR code of a coupon at their restaurant that has passed its expiry date.</p> <p>Given that a user is an employee or owner, when the user tries to scan a coupon QR code for a coupon at their restaurant that</p>

	<p>has passed its expiry date, the page displays an error message saying the coupon cannot be scanned as it has already expired.</p> <p><b>Scenario:</b> An employee or owner tries to scan the QR code of a valid coupon at their restaurant.</p> <p>Given that a user is an employee or owner, when the user tries to scan a coupon QR code for a coupon from their restaurant that is within its valid date range, the page displays scan successful message.</p> <p><b>Scenario:</b> An employee or owner tries to scan the QR code of a coupon at their restaurant that has already been used.</p> <p>Given that a user is an employee or owner, when the user tries to scan a coupon QR code for a coupon from their restaurant that has already been used, the page displays an error message saying that that coupon no longer exists.</p>
As Kevin (customer) I want to be able to see what coupons I already redeemed.	<p><b>Scenario:</b> The customer has not redeemed any coupons.</p> <p>Given that the user is a customer, when the user visits their coupon page, no coupons are listed.</p> <p><b>Scenario:</b> The customer has redeemed some coupons.</p> <p>Given that the user is a customer, when the user visits their coupon page, all their redeemed coupons are listed, with name, description, issuing restaurant, start and end date, and a link to the QR code for using the coupon.</p> <p><b>Scenario:</b> The customer has redeemed coupons at multiple restaurants.</p> <p>Given that the customer has redeemed coupons at multiple restaurants, when the customer views their coupon page, redeemed coupons from all restaurants are listed.</p> <p><b>Scenario:</b> The customer has redeemed multiple copies of a single coupon.</p> <p>Given that the customer has redeemed multiple copies of a single coupon, when the customer views their coupon page, each copy is displayed separately.</p> <p><b>Scenario:</b> The customer has used some of the coupons they redeemed before.</p> <p>Given that the customer has just used a coupon, when the customer views their coupon page, the used coupons do not appear in the list.</p>
As Kevin (customer) I want to be able to purchase	<p><b>Scenario:</b> The customer tries to purchase a coupon but does not have enough points.</p> <p>Given that the customer does not have enough points to get a</p>

<p>coupons by redeeming my points.</p>	<p>coupon, when the customer views the list of coupon offers at a restaurant, there is a message saying they cannot afford the coupon and the button for purchasing the coupon is greyed out and is not clickable.</p> <p><b>Scenario:</b> The customer tries to purchase a coupon they can afford.</p> <p>Given that the customer has enough points to get a coupon, when the customer clicks the button for purchasing the coupon, the page displays a confirmation message saying the coupon has been purchased and the amount of points in the customer's current point count display has decreased by the amount of points the coupon cost.</p>
<p>As Kevin (customer) I want to be able to see what achievements are available</p>	<p><b>Scenario:</b> The customer checks the achievements at a restaurant with no achievements.</p> <p>Given that a restaurant does not have achievements, when a customer visits the restaurant's available achievement list page, no achievements are displayed.</p> <p><b>Scenario:</b> The customer checks the achievements at a restaurant with some achievements.</p> <p>Given that a restaurant has at least one active achievement, when a customer visits the restaurant's available achievement list page, all of the restaurant's active achievements are displayed with name, description, points reward amount, experience reward amount, and max number of progress steps information, as well as a button linking to the QR code for updating the achievement's progress.</p> <p><b>Scenario:</b> The customer checks the achievements at a restaurant where they have made progress on an achievement before.</p> <p>Given that the customer has made progress on at least one achievement at a restaurant, when the customer visits the restaurant's available achievement list page, achievements with non-zero progress are not shown.</p> <p><b>Scenario:</b> The customer checks the achievements at a restaurant that has some expired achievements.</p> <p>Given that a restaurant has at least one expired achievement, when a customer visits the restaurant's available achievement list page, expired achievements are not shown on the list.</p>

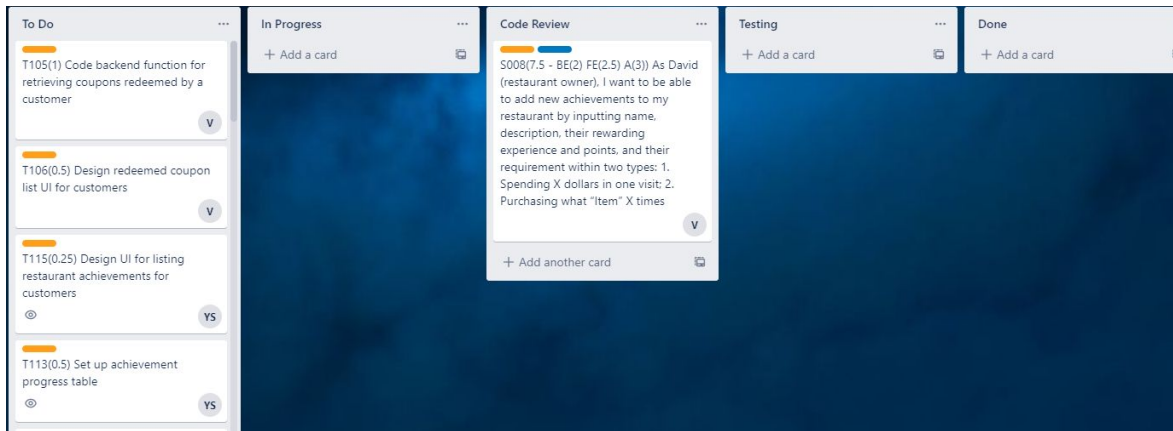
## Sprint 5: Team Task Assignments

Task ID	Cost	Priority	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Assignment
T083	5	High	0	0	0	0	0	5	0	Vanessa
T084	0.5	High	0	0	0.5	0	0	0	0	Vanessa
T085	0.5	High	0	0	0.5	0	0	0	0	Vanessa
T086	0.5	High	0	0	0.5	0	0	0	0	Vanessa
T087	1	High	0	0	1	0	0	0	0	Vanessa
T088	0.5	High	0	0	0.5	0	0	0	0	Vanessa
T089	2	High	0	0	2	0	0	0	0	Vanessa
T090	1	High	0	0	0	0	1	0	0	Yolanda
T091	0.5	High	0	0	0	0	0.5	0	0	Yolanda
T092	1.5	High	0	0	0	0	1.5	0	0	Yolanda
T093	2.5	High	0	0	0	0	1	1.5	0	Yolanda
T094	0.5	High	0	0	0	0.5	0	0	0	Anson
T095	0.5	High	0	0	0	0	0.5	0	0	Anson
T096	1	High	0	0	0	0	1	0	0	Anson
T097	1.5	High	0	0	0	1.5	0	0	0	Anson
T098	1.5	High	0	0	0	0	1.5	0	0	Jiawei
T099	0.5	High	0	0	0	0	0.5	0	0	Jiawei
T100	1	High	0	0	0	0	0	1	0	Jiawei
T101	0.5	High	0	0	0	0	0	0.5	0	Jiawei
T102	0.5	High	0	0	0	0	0	0.5	0	Jiawei
T103	1	High	0	0	0	0	0	1	0	Jiawei
T104	3	High	0	0	0	0	0	0	3	Jiawei
T105	1	High	0	0	0	1	0	0	0	Vanessa
T106	0.5	High	0	0	0	0.5	0	0	0	Vanessa
T107	0.5	High	0	0	0	0.5	0	0	0	Vanessa
T108	2.5	High	0	0	0	2.5	0	0	0	Vanessa

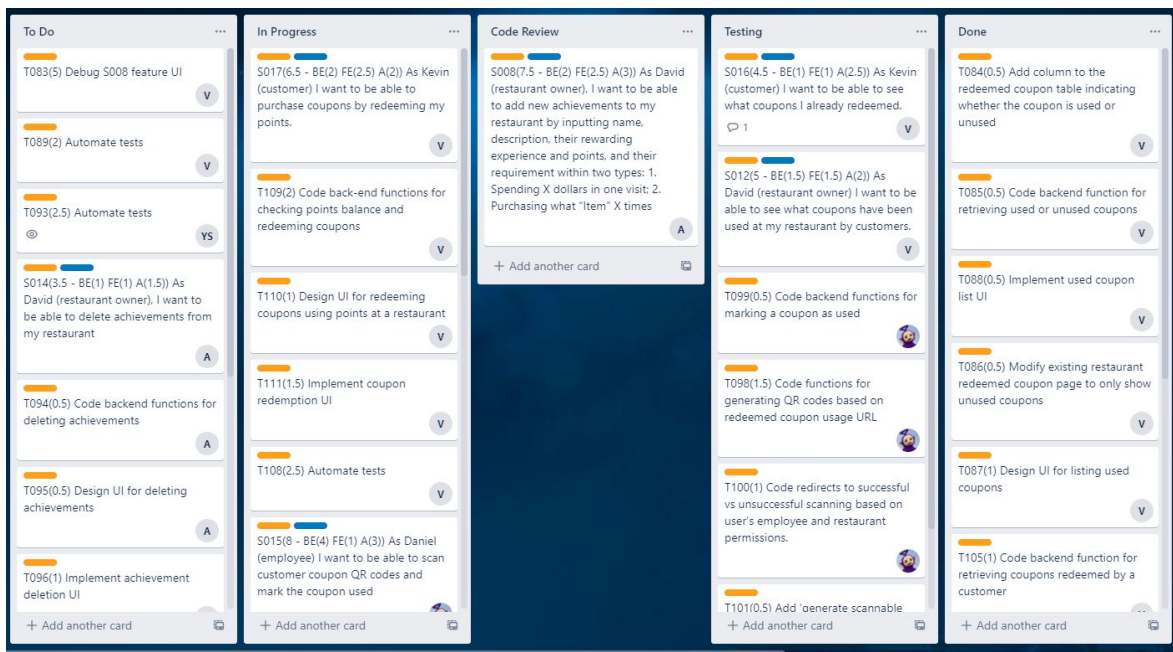
T109	2	High	0	0	0	0	2	0	0	Vanessa
T110	1	High	0	0	0	0	1	0	0	Vanessa
T111	1.5	High	0	0	0	0	1.5	0	0	Vanessa
T112	2	High	0	0	0	0	2	0	0	Vanessa
T113	0.5	High	0	0	0	0	0	0.5	0	Yolanda
T114	0.5	High	0	0	0	0	0	0.5	0	Yolanda
T115	0.25	High	0	0	0	0	0	0.25	0	Yolanda
T116	0.75	High	0	0	0	0	0	0.75	0	Yolanda
T117	3.5	High	0	0	0	0	0	0	3.5	Yolanda
Total Points Completed:			0	0	5	6.5	14	11.5	6.5	
Total Points Left:			43.5	43.5	38.5	32	18	6.5	0	

# Sprint 5: Task Board

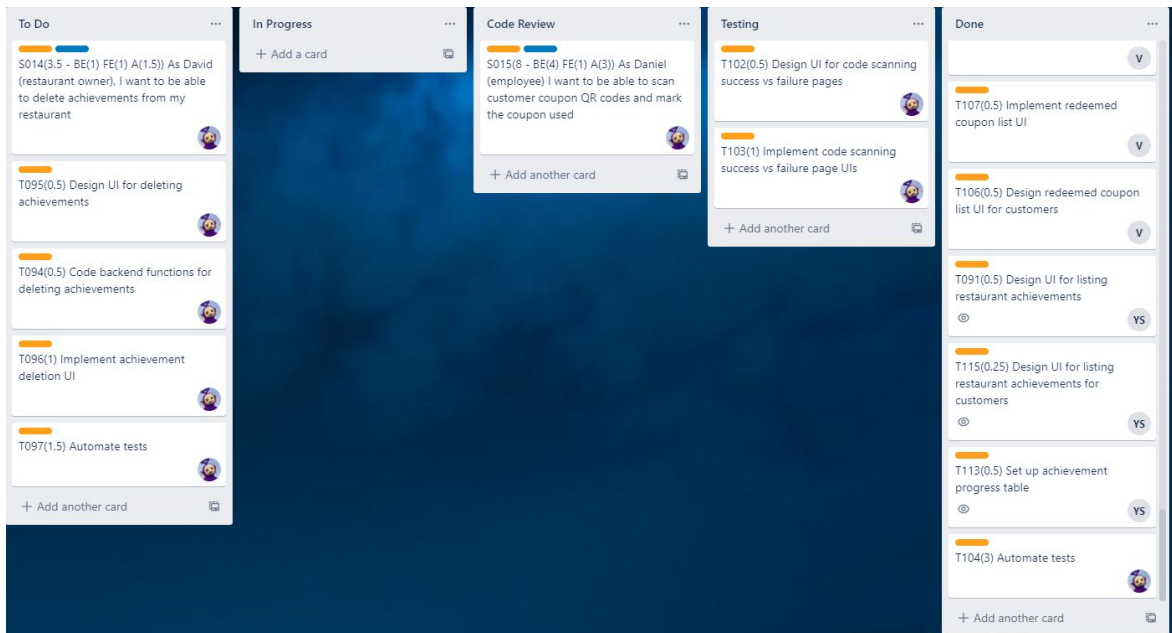
July 20



July 25



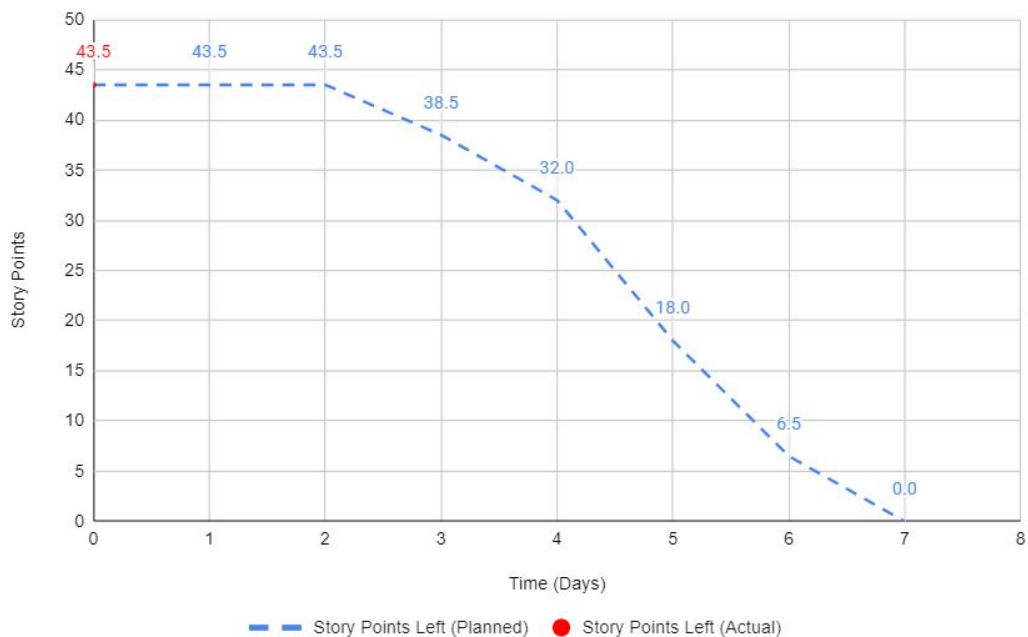
July 27



## Sprint 5: Burndown Chart

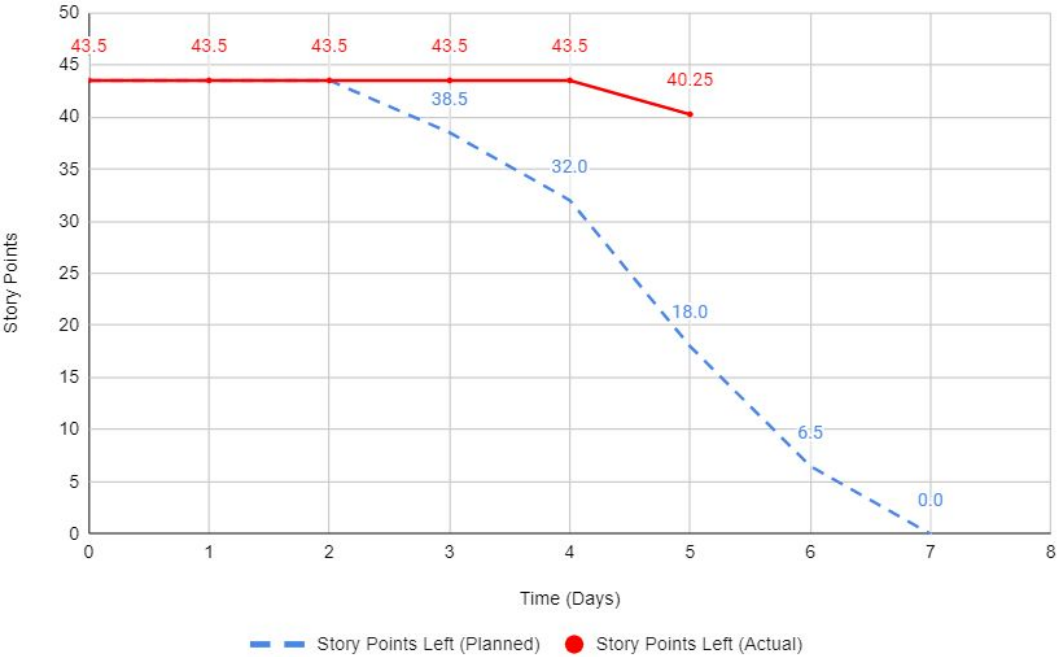
July 20

Burndown Chart



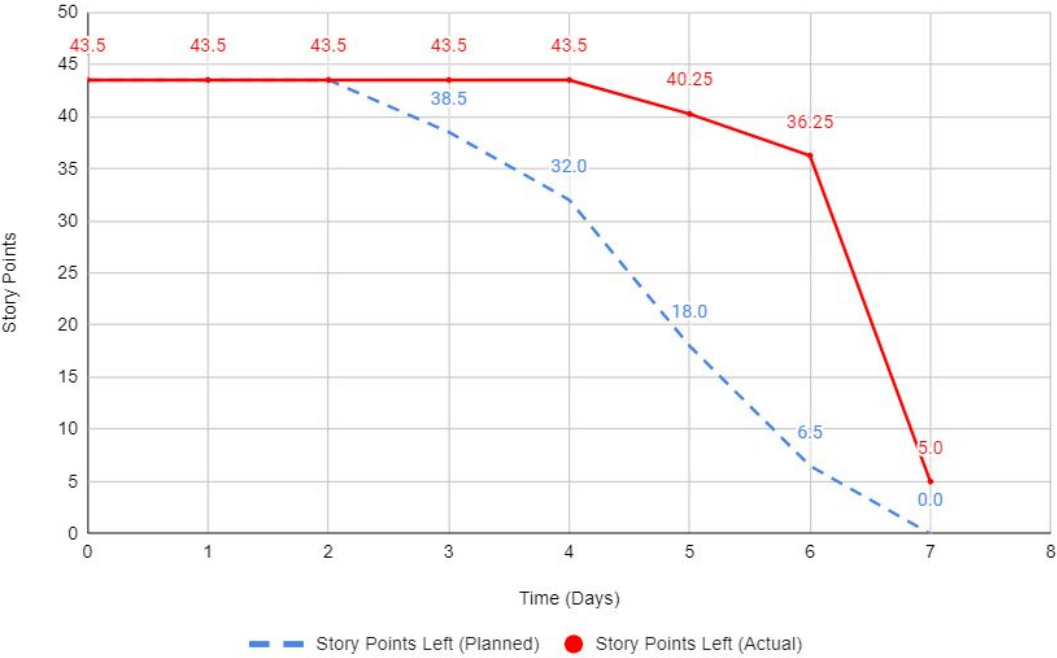
July 25

Burndown Chart



July 27

Burndown Chart





## Sprint 6: Sprint Plan

Priority	Story	Points	Tasks	Points
High	As Daniel (employee) I want to be able to scan customer coupon QR codes and mark the coupon used	1.5	T102: Design UI for code scanning success vs failure pages	0.5
			T103: Implement code scanning success vs failure page UIs	1
High	As David (restaurant owner), I want to be able to delete achievements from my restaurant	3.5	T094: Code backend functions for deleting achievements	0.5
			T095: Design UI for deleting achievements	0.5
			T096: Implement achievement deletion UI	1
			T097: Automate tests	1.5
High	-	3	F003: Fix customer level feature to rely on experience instead of points	3
High	As David (restaurant owner), I want to be able to create achievements in the forms: 1. Visit in a group of x people; 2. Visit x times between date1 and date2	9.5	T118: Update achievement backend functions to accept "Visit in a group of x people" type achievements	0.5
			T119: Update achievement backend functions to accept "Visit x times between date1 and date2" type achievements	0.5
			T120: Code backend for deleting "Visit x times between date1 and date2" type achievements after the deadline is passed	1.5
			T121: Design UI for "Visit in a group of x people" type	1

			achievements	
			T122: Design UI for "Visit x times between date1 and date2" type achievements	1
			T123: Implement UI for "Visit in a group of x people" type achievements	1
			T124: Implement UI for "Visit x times between date1 and date2" type achievements	1
			T125: Automate tests	3
High	As Kevin (customer), I want a progress bar for each achievement that tracks how close I am to completing the achievement.	6	T126: Code backend functions for retrieving achievements that have non-zero progress	1
			T127: Design UI for showing achievements with non-zero progress and the option to filter out completed achievements	0.5
			T128: Implement achievement page UI for a given restaurant	1.5
			T129: Implement 'recent in-progress achievement' UI for a restaurant home page	1
			T130: Automate tests	2
High	As Daniel (employee) I want to be able to scan customer achievement QR codes to update customer progress on a given achievement.	5.5	T131: Code functions for generating QR codes based on achievement updating URL	0.25
			T132: Code backend functions for updating achievement progress	0.5
			T133: Code backend functions for updating	0.75

			customer point and exp count after completing an achievement (i.e. progress = progressMax)	
			T134: Code redirects to successful vs unsuccessful scanning based on user's employee and restaurant permissions and whether the achievement is still valid and incomplete.	0.25
			T135: Add 'generate scannable progress update QR code' button to achievements in the customer achievements list UI	0.25
			T136: Design page for displaying the achievement name and achievement update QR code together	0.25
			T137: Implement code scanning success vs failure page UIs	0.25
			T138: Implement page for displaying the achievement name and achievement update QR code together	0
			T139: Automate tests	3
Medium	As Kevin (customer), I want a profile page containing my name, email and a list of links to restaurants that I visited before.	8.5	T140: Code backend functions for retrieving user info	1
			T141: Code backend functions for visited restaurant info based on whether a customer has made progress on any achievement at a	1

			given restaurant	
			T142: Design profile page UI	1.5
			T143: Implement profile page UI	1
			T144: Automate tests	4
Medium	As Kevin (customer), I want to see user leaderboards for each restaurant on the app so that I can compare my level with other customers and compete to see who has the highest level at a given restaurant.	9	T145: Code backend functions for retrieving a sorted list of customer experience for the top customers at a given restaurant	2
			T146: Design user leaderboard UI that lists the name and level of a restaurant's top 50 customers	1
			T147: Implement user leaderboard UI	3
			T148: Automate tests	3



## Sprint 6: Acceptance Criteria

Story	Acceptance Criteria
<p>As David (restaurant owner), I want to be able to create achievements in the forms: 1. Visit in a group of x people; 2. Visit x times between date1 and date2</p>	<p><b>Scenario:</b> The restaurant owner wants to add an achievement of the form "Visit in a group of x people".</p> <p>Given that the restaurant owner wants to add an achievement of the form "Visit in a group of x people", when the owner visits the create achievement page, there is an option to choose an achievement template that takes in an integer x and an item name in the form "Visit in a group of x people".</p> <p><b>Scenario:</b> The restaurant owner tries to add an achievement of the form "Visit in a group of x people" and sets x as a negative value.</p> <p>Given that the restaurant owner has filled in an achievement template of the form "Visit in a group of x people" with a negative x value, when the owner clicks the create button, an error message is shown and no new achievement is shown on the achievement list.</p> <p><b>Scenario:</b> The restaurant owner wants to add an achievement of the form "Visit x times between date1 and date2".</p> <p>Given that the restaurant owner wants to add an achievement of the form "Visit x times between date1 and date2", when the owner visits the create achievement page, there is an option to choose an achievement template that takes in an integer x and two dates date1 and date3 in the form "Visit x times between date1 and date2".</p> <p><b>Scenario:</b> The restaurant owner tries to add an achievement of the form "Visit x times between date1 and date2" and sets x as a negative value.</p> <p>Given that the restaurant owner has filled in an achievement template of the form "Visit x times between date1 and date2" with a negative x value, when the owner clicks the create button, an error message is shown and no new achievement is shown on the achievement list.</p> <p><b>Scenario:</b> The restaurant owner tries to add an achievement of the form "Visit x times between date1 and date2" and sets date2 in the past.</p> <p>Given that the restaurant owner has filled in an achievement template of the form "Visit x times between date1 and date2" with a past date2 value, when the owner clicks the create button, an error message is shown and no new achievement is shown on the achievement list.</p>

	<p><b>Scenario:</b> The restaurant owner tries to add an achievement of the form "Visit x times between date1 and date2" and sets the date range as indefinite.</p> <p>Given that the restaurant owner wants to create an achievement of the form "Visit x times between date1 and date2" with an indefinite date range, when the owner selects the achievement type, there is a checkbox option for setting date as indefinite, and after creating the achievement the achievement appears in the form "Visit x times".</p>
As Kevin (customer), I want a progress bar for each achievement that tracks how close I am to completing the achievement.	<p><b>Scenario:</b> The customer does not have progress on any achievements.</p> <p>Given that the customer does not have progress on any achievements, when the customer checks the list of their in progress and complete achievements at a restaurant, the lists are empty.</p> <p><b>Scenario:</b> The customer has begun but not completed some achievements.</p> <p>Given that the customer begun but not completed some achievements, when the customer checks the list of their in progress achievements at a restaurant, all incomplete achievements with non-zero progress are listed with information about the achievement's name, description, start and expiry date, and amount of progress done out of the maximum progress steps.</p> <p><b>Scenario:</b> The customer has completed some achievements.</p> <p>Given that the customer completed some achievements, when the customer checks the list of their in complete achievements at a restaurant, all complete achievements are listed with information about the achievement's name, description, start and expiry date, and a full progress bar and completion check mark.</p>
As Daniel (employee) I want to be able to scan customer achievement QR codes to update customer progress on a given achievement.	<p><b>Scenario:</b> A user scans an achievement QR code and they are not an owner or employee.</p> <p>Given that a user is a customer, when the user tries to scan an achievement QR code, the page displays an error message saying they do not have permission to scan QR codes.</p> <p><b>Scenario:</b> An employee or owner tries to scan the QR code of an achievement at a restaurant they do not work at.</p> <p>Given that a user is an employee or owner, when the user tries to scan an achievement QR code for an achievement at a restaurant they do not work at, the page displays an error message saying they do not have permission to scan QR codes at that achievement's restaurant.</p> <p><b>Scenario:</b> An employee or owner tries to scan the QR code</p>

	<p>of an achievement at their restaurant that has not reached its start date.</p> <p>Given that a user is an employee or owner, when the user tries to scan an achievement QR code for a coupon at their restaurant that has not reached its start date, the page displays an error message saying the achievement cannot be scanned as it is still before the achievement's start date.</p> <p><b>Scenario:</b> An employee or owner tries to scan the QR code of an achievement at their restaurant that has passed its expiry date.</p> <p>Given that a user is an employee or owner, when the user tries to scan an achievement QR code for a coupon at their restaurant that has passed its expiry date, the page displays an error message saying the achievement cannot be scanned as it has already expired.</p> <p><b>Scenario:</b> An employee or owner tries to scan the QR code of an achievement at their restaurant that has already been completed.</p> <p>Given that a user is an employee or owner, when the user tries to scan an achievement QR code for an achievement from their restaurant that has already been completed, the page displays an error message saying that that achievement is already complete.</p> <p><b>Scenario:</b> An employee or owner tries to scan the QR code of a valid incomplete achievement at their restaurant.</p> <p>Given that a user is an employee or owner, when the user tries to scan an incomplete achievement's QR code for a coupon from their restaurant that is within its valid date range, the page displays scan successful message, and the progress bar on the achievement has increased by one segment for the customer's achievement list.</p>
<p>As Kevin (customer), I want a profile page containing my name, email and a list of links to restaurants that I visited before.</p>	<p><b>Scenario:</b> The customer visits the profile page.</p> <p>Given that the user is a customer, when the user visits their profile page, their account's name and email are listed.</p> <p><b>Scenario:</b> The customer has not visited any restaurants before.</p> <p>Given that the user has not visited any restaurants before, when the user visits their profile page, no restaurants appear in the 'visited restaurants' section.</p> <p><b>Scenario:</b> The customer has visited at least one restaurant.</p> <p>Given that the user has visited at least one restaurant before, when the user visits their profile page, all restaurants they visited restaurants appear in the 'visited restaurants' section</p>



	with name and address information as well as a button linking to that restaurant's page.
As Kevin (customer), I want to see user leaderboards for each restaurant on the app so that I can compare my level with other customers and compete to see who has the highest level at a given restaurant.	<p><b>Scenario:</b> No customers have earned experience at a given restaurant.</p> <p>Given that no customers have earned experience at a given restaurant, when a user visits the restaurant's leaderboard page, no customers appear on the leaderboard.</p> <p><b>Scenario:</b> Less than 10 customers have earned experience at a restaurant.</p> <p>Given that less than 10 customers have earned experience at a given restaurant, when a user visits the restaurant's leaderboard page, all customers that have some experience at the restaurant appear on the leaderboard ranked by amount of experience, where the customer with the most experience is ranked at 1.</p> <p><b>Scenario:</b> 10 or more customer have earned experience at a restaurant.</p> <p>Given that 10 or more customers have earned experience at a given restaurant, when a user visits the restaurant's leaderboard page, the top 10 customers with the most experience at the restaurant appear on the leaderboard ranked by amount of experience, where the customer with the most experience is ranked at 1.</p> <p><b>Scenario:</b> A customer is in the list of top 10 customers at a given restaurant.</p> <p>Given that a customer is in list of top 10 customers at a given restaurant, when that customer visits the leaderboard page, their position on the board is highlighted.</p>

We did not include acceptance criteria for stories carried over from the previous sprint, as acceptance criteria was already written for those stories in the previous sprint.

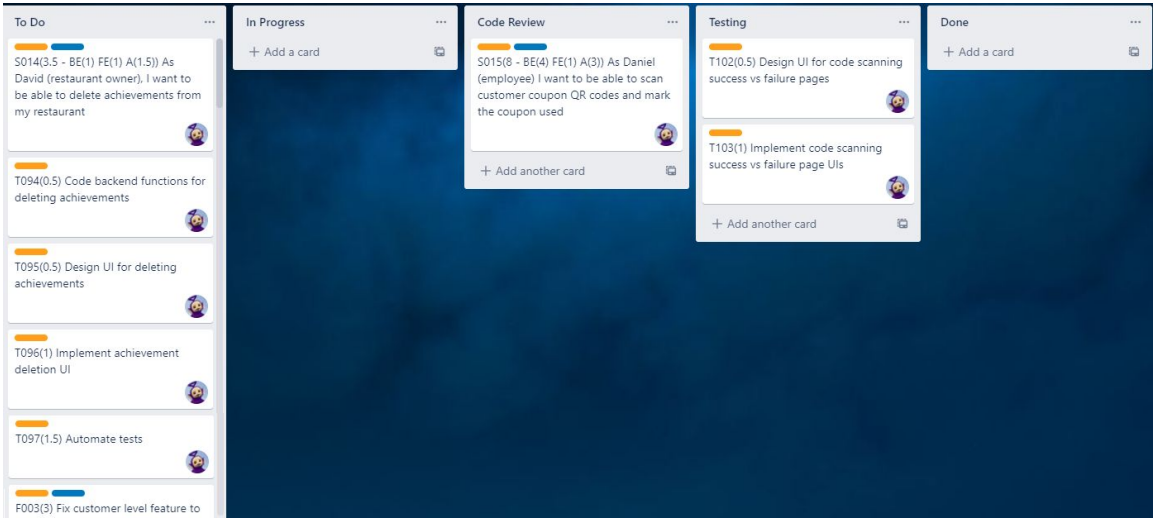
## Sprint 6: Team Task Assignments

Task ID	Cost	Priority	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Assignment
T094	0.5	High	0	0.5	0	0	0	0	0	Jiawei
T095	0.5	High	0	0.5	0	0	0	0	0	Jiawei
T096	1	High	0	1	0	0	0	0	0	Jiawei
T097	1.5	High	0	1.5	0	0	0	0	0	Jiawei
T102	0.5	High	0.5	0	0	0	0	0	0	Jiawei
T013	1	High	1	0	0	0	0	0	0	Jiawei
F003	3	High	0	0	3	0	0	0	0	Yolanda
T118	0.5	High	0	0	0	0.5	0	0	0	Vanessa
T119	0.5	High	0	0	0	0.5	0	0	0	Vanessa
T120	1.5	High	0	0	0	1.5	0	0	0	Vanessa
T121	1	High	0	0	0	1	0	0	0	Vanessa
T122	1	High	0	0	0	1	0	0	0	Vanessa
T123	1	High	0	0	0	1	0	0	0	Vanessa
T124	1	High	0	0	0	1	0	0	0	Vanessa
T125	3	High	0	0	0	3	0	0	0	Vanessa
T126	1	High	0	0	0	0	1	0	0	Yolanda
T127	0.5	High	0	0	0	0	0.5	0	0	Yolanda
T128	1.5	High	0	0	0	0	1.5	0	0	Yolanda
T129	1	High	0	0	0	0	0	1	0	Yolanda
T130	2	High	0	0	0	0	0	2	0	Yolanda
T131	0.25	High	0	0	0	0	0.25	0	0	Jiawei
T132	0.5	High	0	0	0	0	0.5	0	0	Jiawei
T133	0.75	High	0	0	0	0	0.75	0	0	Jiawei
T134	0.25	High	0	0	0	0	0.25	0	0	Jiawei
T135	0.25	High	0	0	0	0	0	0.25	0	Jiawei
T136	0.25	High	0	0	0	0	0	0.25	0	Jiawei

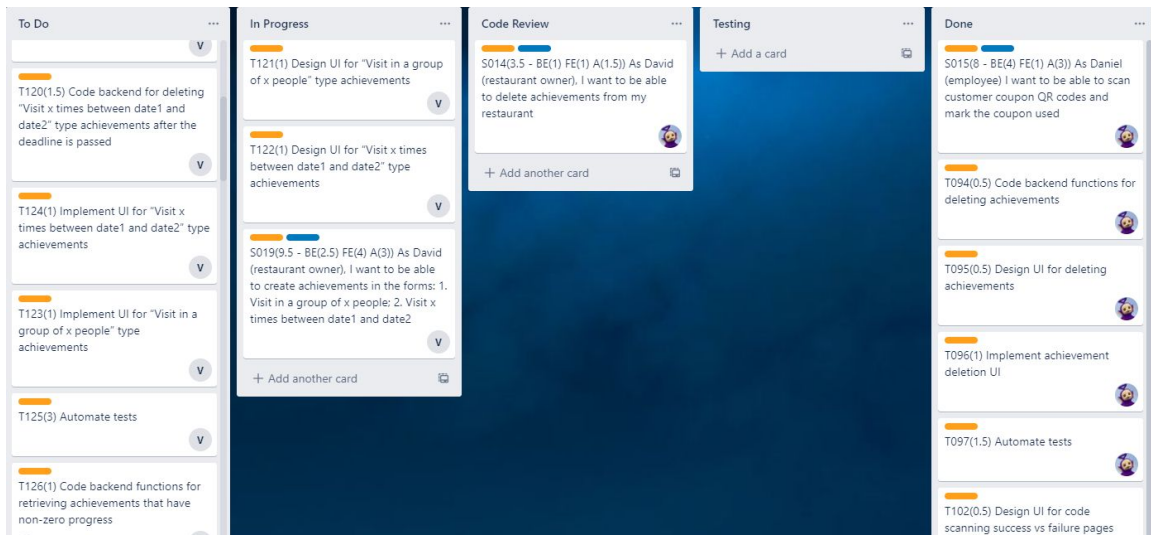
T137	0.25	High	0	0	0	0	0	0.25	0	Jiawei
T138	0	High	0	0	0	0	0	0	0	Jiawei
T139	3	High	0	0	0	0	0	0	3	Jiawei
T140	1	Medium	0	0	0	0	1	0	0	Vanessa
T141	1	Medium	0	0	0	0	1	0	0	Vanessa
T142	1.5	Medium	0	0	0	0	1.5	0	0	Vanessa
T143	1	Medium	0	0	0	0	1	0	0	Vanessa
T144	4	Medium	0	0	0	0	0	4	0	Vanessa
T145	2	Medium	0	0	0	0	0	2	0	Jiawei
T146	1	Medium	0	0	0	0	0	1	0	Jiawei
T147	3	Medium	0	0	0	0	0	0	3	Jiawei
T148	3	Medium	0	0	0	0	0	0	3	Jiawei
Total Points Completed:			1.5	3.5	3	9.5	9.25	10.75	9	
Total Points Left:			45	41.5	38.5	29	19.75	9	0	

# Sprint 6: Task Board

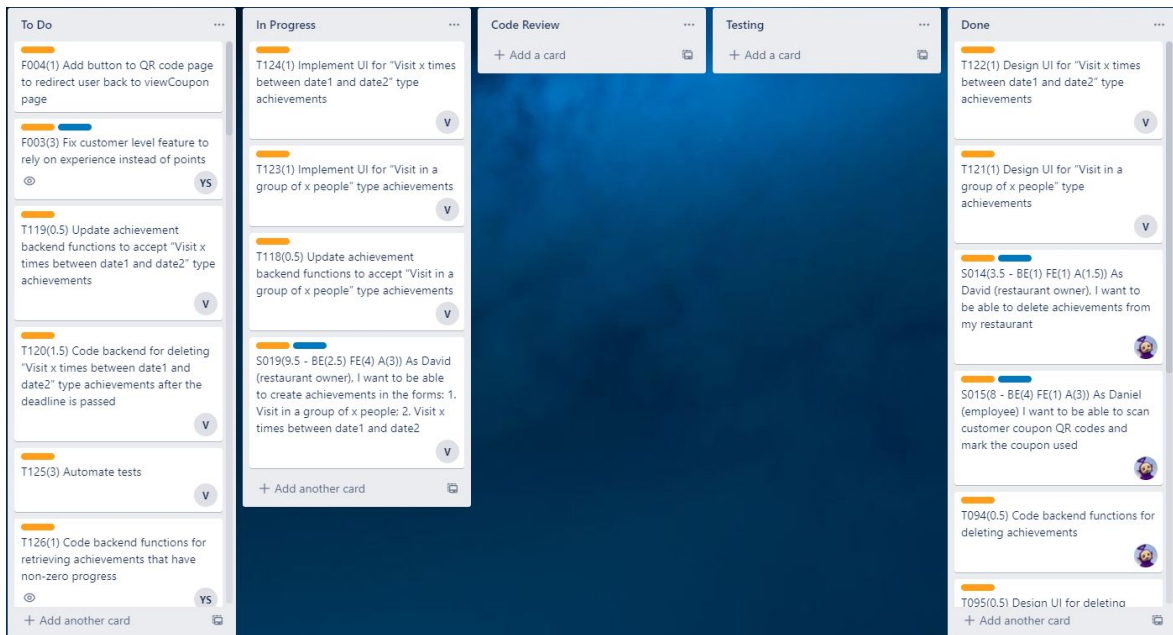
July 27



July 29



July 31



## August 2

**To Do**

- T148(3) Automate tests
- + Add another card

**In Progress**

- S022(8.5 - BE(2) FE(2.5) A(4)) As Kevin (customer), I want a profile page containing my name, email and a list of links to restaurants that I visited before.
- T144(4) Automate tests
- S023(9 - BE(2) FE(4) A(3)) As Kevin (customer), I want to see user leaderboards for each restaurant on the app so that I can compare my level with other customers and compete to see who has the highest level at a given restaurant.
- T147(3) Implement user leaderboard UI
- T125(3) Automate tests
- + Add another card

**Code Review**

- S021(5.5 - BE(1.5) FE(1) A(3)) As Daniel (employee) I want to be able to scan customer coupon QR codes to update customer progress on a given achievement.
- T149(1) Allow employees to see the coupons and achievements offered at their restaurant on the coupon/achievement pages.
- S019(9.5 - BE(2.5) FE(4) A(3)) As David (restaurant owner), I want to be able to create achievements in the forms: 1. Visit in a group of x people; 2. Visit x times between date1 and date2
- S020(6 - BE(1) FE(3) A(2)) As Kevin (customer), I want a progress bar for each achievement that tracks how close I am to completing the achievement.
- + Add another card

**Testing**

- T145(2) Code backend functions for retrieving a sorted list of customer experience for the top customers at a given restaurant
- T146(1) Design user leaderboard UI that lists the name and level of a restaurant's top 50 customers
- + Add another card

**Done**

- T141(1) Code backend functions for visited restaurant info based on whether a customer has made progress on any achievement at a given restaurant
- T143(1) Implement profile page UI
- T142(1.5) Design profile page UI
- T140(1) Code backend functions for retrieving user info
- T130(2) Automate tests
- T128(1.5) Implement achievement page UI for a given restaurant
- + Add another card

## August 3

**To Do**

- T148(3) Automate tests
- + Add another card

**In Progress**

- S022(8.5 - BE(2) FE(2.5) A(4)) As Kevin (customer), I want a profile page containing my name, email and a list of links to restaurants that I visited before.
- T144(4) Automate tests
- S023(9 - BE(2) FE(4) A(3)) As Kevin (customer), I want to see user leaderboards for each restaurant on the app so that I can compare my level with other customers and compete to see who has the highest level at a given restaurant.
- T147(3) Implement user leaderboard UI

**Code Review**

- + Add a card

**Testing**

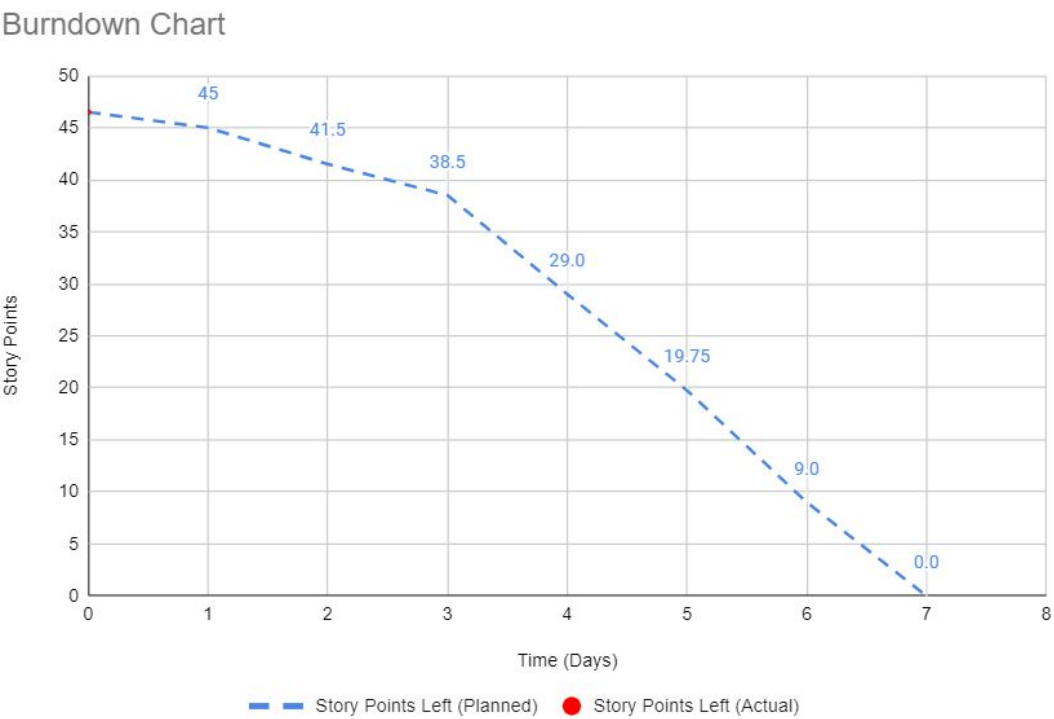
- T145(2) Code backend functions for retrieving a sorted list of customer experience for the top customers at a given restaurant
- T146(1) Design user leaderboard UI that lists the name and level of a restaurant's top 50 customers
- + Add another card

**Done**

- F008(1) Fix QR code scanning page coupon info text to be readable for narrower screens.
- T138(0) Implement page for displaying the achievement name and achievement update QR code together
- T137(0.25) Implement code scanning success vs failure page UIs
- T135(0.25) Add 'generate scannable progress update QR code' button to achievements in the customer achievements list UI
- T136(0.25) Design page for

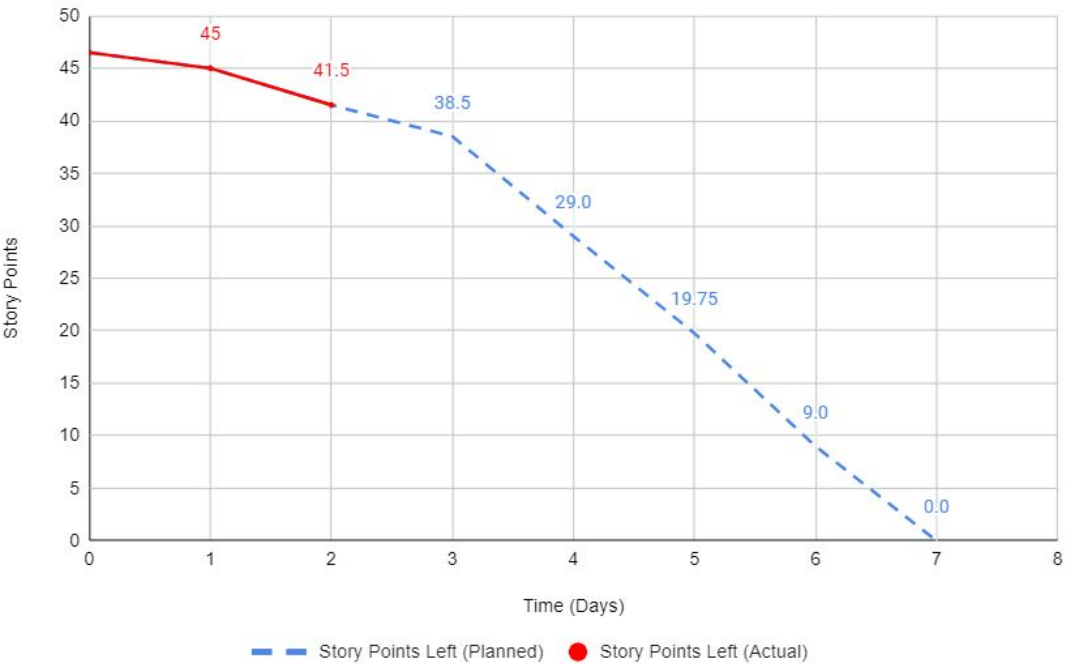
# Sprint 6: Burndown Chart

July 27



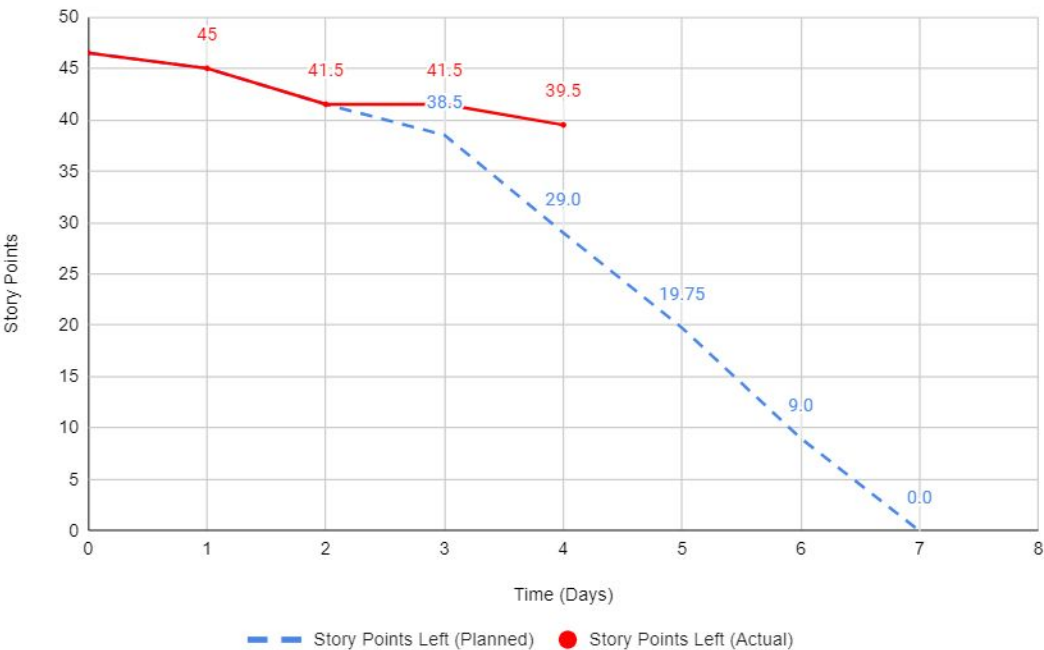
July 29

Burndown Chart



July 31

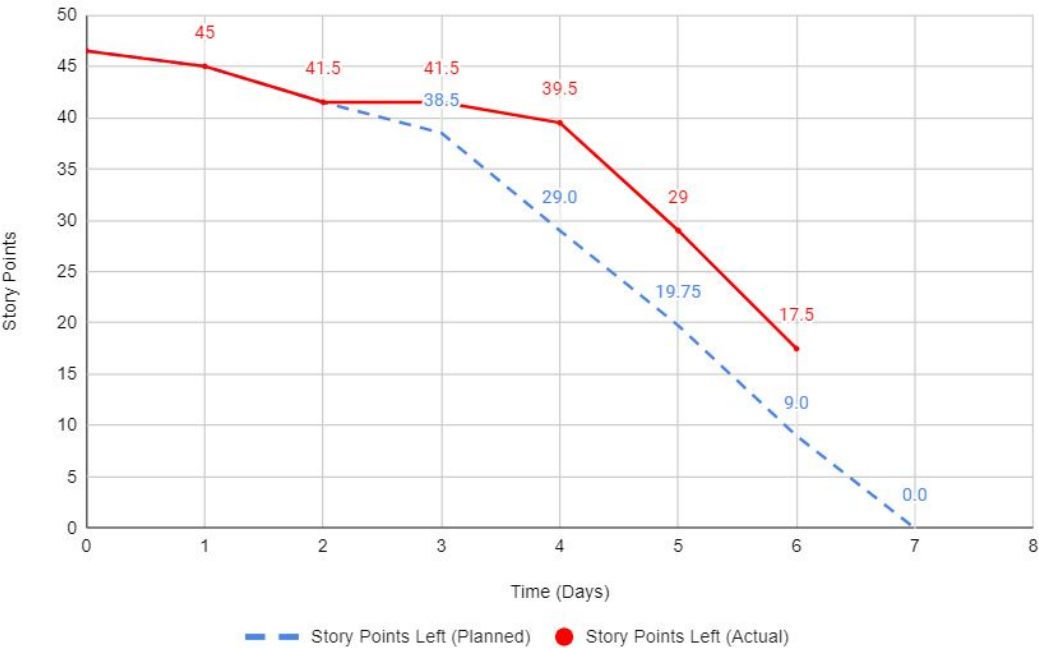
Burndown Chart





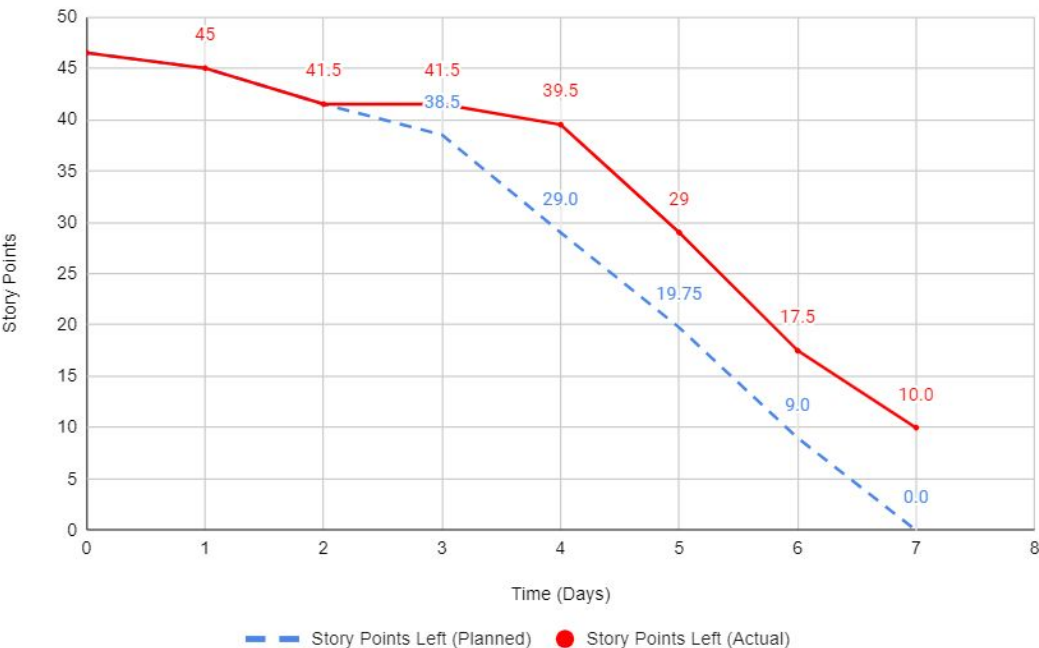
# August 2

Burndown Chart



# August 3

Burndown Chart



## Sprint 7: Sprint Plan

Priority	Story	Points	Tasks	Points
Medium	As Kevin (customer), I want a profile page containing my name, email and a list of links to restaurants that I have favoured.	6	T150: Set up table for favorited restaurants by a given customer	0.5
			T151: Implement UI for favoriting restaurants on a restaurant's home page	1
			T152: Update restaurant list on customer profile to list favorited restaurants only	0.5
			T144: Automate tests	4
Medium	As Kevin (customer), I want to see user leaderboards for each restaurant on the app so that I can compare my level with other customers and compete to see who has the highest level at a given restaurant.	6	T147: Implement user leaderboard UI	3
			T148: Automate tests	3
High	--	9.5	F005: Prevent coupons whose begin date is in the future or whose expiry date is in the past from being scannable	1
			F006: Fix site UI to be mobile-friendly	1
			F007: Add documentation to test suites missing documentation and make sure all documentation is uniform style-wise	2.5
			F009: Allow employees to access and view the viewUserCoupon.html page	1
			F010: Update customer achievement list UI so that achievements that are expired are not shown in the available coupon list and progress updating is not allowed for achievements outside of the date range	2

			F011: Add achievement info the achievement progress QR code page	1
			F012: Fix QR code scanning so that employees and owners cannot scan coupons and achievements that belong to other restaurants besides their own	1
Medium	As David (restaurant owner) I want to be able to set the number of bonus points customers can earn at my restaurant at level thresholds of my choice, e.g. 100 points earned for reaching the level 5 threshold.	9	T153: Set up milestones table for the database, where each milestone has restaurant ID, milestone level, and milestone reward point information	0.5
			T154: Code backend functions for creating, fetching, and deleting milestones	1
			T155: Design UI for letting owners create milestones	0.5
			T156: Design UI for letting owners see the milestones they have created	0.5
			T157: Implement milestone creation UI	1
			T158: Implement milestone list UI, including milestone deletion	1
			T159: Code backend functions for getting the next nearest milestone for a customer	1
			T160: Update achievement progress QR scanning to check if the user has leveled up and hit a milestone after the achievement has been completed, and add points for the user accordingly	0.5
			T161: Update user level UI footer at a restaurant to show the next nearest milestone, e.g. "Earn 100 points for reaching level 5!" when the user level < 5.	0.5

			T162: Automate tests	3
Medium	As David (restaurant owner) I want to be able to set coupons to be available only to users of a minimum level or higher.	8.5	T163: Update coupon table to have a minimum level column	0.5
			T164: Update coupon creation function to include minimum level	0.5
			T165: Code a filter function to filter coupons based on customer level	2
			T166: Update owner's coupon creation UI to take minimum level value	0.5
			T167: Update coupon list UI to show minimum customer level info to owners and employees	0.5
			T168: Update customer coupon purchasing UI so that coupons beyond the customer's level are not shown	1
			T169: Automate tests and update coupon insertion test suite	3.5
Medium	As Kevin (customer) I want to be able to see my 3 most recently updated achievements on my home page.	6.5	T170: Update achievement progress table with a last updated date column	0.25
			T171: Update achievement progress insertion to initialize "last updated" as the creation time	0.5
			T172: Update achievement progress updater to update the last update date with the timestamp of the function call	0.5
			T173: Code functions for fetching the 3 most recently updated achievements	0.75
			T174: Design UI for seeing recently updated achievements	0.5
			T175: Implement UI for	1

			seeing the recently updated achievements on the home page	
			T176: Automate tests and update the achievement progress test suite	3
Low	As Daniel (employee), I want to be able to see the name and address of the restaurant I work at on my home page	1.5	T177: Set up backend for fetching restaurant name and address for an employee using existing functions	0.5
			T178: Design home page UI for showing restaurant information	0.25
			T179: Implement home page UI	0.75
Low	As David (restaurant owner) I want to be able to see how many customers have started and how many customers have completed each of the achievements at my restaurant, including expired achievements.	6	T180: Code functions for fetching the number of in-progress progress entries and the number of complete progress entries for all achievements at a restaurant	2
			T181: Design UI for achievement stats page that lists all achievements at a restaurant and states whether the achievement is expired or not	1
			T182: Implement achievement stats page UI for a restaurant owner	1
			T183: Automate tests	2
Low	As David (restaurant owner) I want to be able to see the top 3 owned coupons, top 3 used coupons and top 3 achievements at my restaurant on my home page	8	T184: Code backend function for fetching the 3 most owned coupons at a given restaurant	1
			T185: Code backend function for fetching the 3 most used coupons at a given restaurant	1
			T186: Code backend function for fetching the 3 achievements with the most progress entries at a given restaurant	1

			T187: Design home page UI for showing the top 3 owned coupons, top 3 used coupons and top 3 achievements at an owner's restaurant	0.5
			T188: Implement owner home page UI	1.5
			T189: Automate tests	3
Low	As David (restaurant owner), I want to be able to edit the name and address of my restaurant via my profile page.	4.5	T190: Code backend functions for updating restaurant name and address	1
			T191: Design UI for editing restaurant name and address	0.25
			T192: Implement restaurant information display UI on owner profile page with an edit button to redirect to editing interface	0.5
			T193: Implement UI for editing restaurant name and address	0.75
			T194: Automate tests	1
High	As David (restaurant owner), I want to promote or demote employees to/from "manager" type employees, where "manager" employees can access all features of an owner except for promotion/demotion.	5.5	T195: Implement new manager user type, linked to a restaurant	0.5
			T196: Code backend functions for promoting/demoting employee type users into manager type users	1
			T197: Implement UI button for promoting/demoting employees on the employee page	1
			T198: Allow manager users to see all features an owner can see except for features related to editing or creating employees	2
			T199: Automate tests	1

## Sprint 7: Acceptance Criteria

Story	Acceptance Criteria
As Kevin (customer), I want a profile page containing my name, email and a list of links to restaurants that I have favourited.	<p><b>Scenario:</b> A customer has not favourited any restaurants.</p> <p>Given that a customer has not favourited any restaurants, when the customer visits their favourites page, no restaurants are listed.</p> <p><b>Scenario:</b> A customer has favourited at least one restaurant.</p> <p>Given that a customer has favourited at least one restaurant, when the customer visits their favourites page, all their favourited restaurants are listed with restaurant name and address information, as well as a button linking to the restaurant's page.</p> <p><b>Scenario:</b> A customer visits the home page of a restaurant they have not favourited.</p> <p>Given that the customer has not favourited a restaurant, when the customer visits the restaurant's home page, the favourite symbol is not selected.</p> <p><b>Scenario:</b> A customer visits the home page of a restaurant they have already favourited.</p> <p>Given that the customer has already favourited a restaurant, when the customer visits the restaurant's home page, the favourite symbol is selected.</p> <p><b>Scenario:</b> A customer clicks on the favourite symbol on the home page of a restaurant they have not favourited.</p> <p>Given that the customer has not favourited a restaurant, when the customer clicks on the favourite symbol on the restaurant's home page, the favourite symbol is selected and the restaurant appears on their favourites page.</p> <p><b>Scenario:</b> A customer clicks on the favourite symbol on the home page of a restaurant they have already favourited.</p> <p>Given that the customer has already favourited a restaurant, when the customer clicks on the favourite symbol on the restaurant's home page, the favourite symbol is deselected and the restaurant no longer appears on their favourites page.</p>
As David (restaurant owner) I want to be able to set the number of bonus points customers can earn at my restaurant at level	<p><b>Scenario:</b> There are no thresholds at a restaurant.</p> <p>Given that there are no thresholds at a restaurant, when the owner visits the thresholds page, no thresholds are listed.</p>

<p>thresholds of my choice, e.g. 100 points earned for reaching the level 5 threshold.</p>	<p><b>Scenario:</b> There is at least one threshold at a restaurant.</p> <p>Given that there is at least one threshold at a restaurant, when the owner visits the thresholds page, all thresholds are listed with level and points reward information.</p> <p><b>Scenario:</b> The owner tries to create a threshold for a new level with a valid points amount.</p> <p>Given that the user is an owner, when the user tries to create a threshold for a new level, the threshold appears on the threshold list.</p> <p><b>Scenario:</b> The owner tries to create a threshold with a negative level or points value.</p> <p>Given that the user is an owner, when the user tries to create a threshold with a negative points value, an error message is displayed and the threshold does not appear on the threshold list.</p> <p><b>Scenario:</b> The owner tries to create a threshold for a level that already has a threshold.</p> <p>Given that the user is an owner, when the user tries to create a threshold for a level that already has a threshold, the old threshold of the same level is replaced with the new threshold.</p> <p><b>Scenario:</b> The owner wants to change the number of points a threshold is worth.</p> <p>Given that the owner wants to change the number of points a threshold is worth, when the owner clicks edit on a threshold, the points field becomes an editable text box, while the level field remains uneditable.</p> <p><b>Scenario:</b> The owner wants to delete a threshold.</p> <p>Given that the user is an owner, when the user clicks on delete on a threshold, that threshold no longer appears on the threshold list.</p> <p><b>Scenario:</b> A customer visits the restaurant's home page</p> <p>Given that the user is a customer, when the user visits a restaurant's home page, the customer can see what the next nearest threshold is based on their customer level and how many points that threshold rewards.</p>
<p>As David (restaurant owner) I want to be able to set coupons to be available only to users of a minimum level or higher.</p>	<p><b>Scenario:</b> An owner wants to create a new coupon.</p> <p>Given that the user is an owner, when the user clicks create a new coupon, the new coupon form includes an integer only minimum level field.</p> <p><b>Scenario:</b> An owner tries to create a new coupon with an</p>



	<p>empty or negative level value.</p> <p>Given that the owner has entered an empty or negative value into the minimum level field on the create coupon form, when the owner clicks create, and error message appears saying that the value must be non-negative and no new coupon appears on the restaurant coupon list.</p> <p><b>Scenario:</b> A customer visits the coupon offers list at a restaurant and is not high level enough for a coupon.</p> <p>Given that the customer is not high level enough to purchase a given coupon at a given restaurant, when the customer visits the restaurant's coupon offers list, the purchase button is greyed out for the coupon and not clickable, with the message 'your level is too low to buy this coupon'.</p>
As Kevin (customer) I want to be able to see my 3 most recently updated achievements on my home page.	<p><b>Scenario:</b> The customer has made progress on less than 3 achievements.</p> <p>Given that the user is a customer, when the user visits their home page, all achievements they have made non-zero progress on appear with name, restaurant, description, progress amount, and reward information.</p> <p><b>Scenario:</b> The customer has made progress on 3 or more achievements.</p> <p>Given that the user is a customer, when the user visits their home page, only the 3 achievements they have updated progress on most recently appear.</p>
As Daniel (employee), I want to be able to see the name and address of the restaurant I work at on my home page.	<p><b>Scenario:</b> The employee visits the home page of the site.</p> <p>Given that a user is an employee, when the user visits the home page of the site, the name and address of the restaurant they work at is displayed.</p>
As David (restaurant owner) I want to be able to see how many customers have started and how many customers have completed each of the achievements at my restaurant, including expired achievements.	<p><b>Scenario:</b> The owner checks the achievement statistics when the restaurant has no achievements.</p> <p>Given that the user is an owner, when the user visits the achievement statistics page for their restaurant, no achievements are listed.</p> <p><b>Scenario:</b> The owner checks the achievement statistics when the restaurant has some active achievements.</p> <p>Given that the user is an owner, when the user visits the achievement statistics page for their restaurant, all achievements at my restaurant are listed by name, description, number of customers who have begun but not completed the achievement, and number of customers that have completed the achievement, with a status field marking the achievement as active.</p>

	<p><b>Scenario:</b> The owner checks the achievement statistics when some achievements are expired.</p> <p>Given that the user is an owner, when the user visits the achievement statistics page for their restaurant, all expired achievements at my restaurant are included in the achievement list in the same format as the active achievements, except the status field marks the achievement as expired.</p> <p><b>Scenario:</b> The owner checks the coupon statistics when some achievements have been deleted.</p> <p>Given that the user is an owner, when the user visits the achievement statistics page for their restaurant, deleted achievements at my restaurant are not shown in the coupon statistics list.</p> <p><b>Scenario:</b> A customer has recently completed an achievement at a restaurant.</p> <p>Given a customer has just completed an achievement at the owner's restaurant, when an owner checks the statistics of that achievement, the number of customers who have begun but not completed the achievement will be lower by 1 and the number of customers that have completed the achievement used will be higher by 1</p>
<p>As David (restaurant owner) I want to be able to see the top 3 owned coupons, top 3 used coupons and top 3 achievements at my restaurant on my home page</p>	<p><b>Scenario:</b> There are less than 3 coupons at a restaurant.</p> <p>Given that the user is an owner, when the user visits their home page, a list of all coupons at the restaurant appears with name, description, and the number of copies held, sorted from greatest to least, as well as a list of all coupons at the restaurant with name, description, and the number of copies used, sorted from greatest to least.</p> <p><b>Scenario:</b> There are 3 or more coupons at a restaurant.</p> <p>Given that the user is an owner, when the user visits their home page, a list of only the top 3 most held coupons at the restaurant appears, sorted from greatest to least, as well as a list of the top 3 most owned coupons, sorted from greatest to least.</p> <p><b>Scenario:</b> There are less than 3 achievements at a restaurant.</p> <p>Given that the user is an owner, when the user visits their home page, all achievements at the restaurant appear with name, description, and the number of customers who have made non-zero progress on the achievement, sorted from greatest to least.</p> <p><b>Scenario:</b> There are 3 or more achievements at a restaurant.</p>

	<p>Given that the user is an owner, when the user visits their home page, only the 3 achievements with the highest number of customers who have made non-zero progress on the achievement are shown, sorted from greatest to least.</p>
<p>As David (restaurant owner), I want to be able to edit the name and address of my restaurant via my profile page.</p>	<p><b>Scenario:</b> The owner tries to update the restaurant information with an empty name or address.</p> <p>Given that the owner has entered an empty restaurant name and/or address, when the owner clicks update, an error message appears and the restaurant's name and address remain unchanged.</p> <p><b>Scenario:</b> The owner tries to update the restaurant information with a nonempty name and nonempty address.</p> <p>Given that the owner has entered a nonempty restaurant name and nonempty address, when the owner clicks update, the owner is redirected to a page showing the new restaurant name and address, and the restaurant's new name and address appear when searched by customers.</p>
<p>As David (restaurant owner), I want to promote or demote employees to/from "manager" type employees, where "manager" employees can access all features of an owner except for promotion/demotion.</p>	<p><b>Scenario:</b> There is a regular employee at a restaurant.</p> <p>Given that an employee is a regular employee, when the owner clicks the gear icon next to their name on the employee list, the promote option appears, and the demote option does not appear.</p> <p><b>Scenario:</b> There is a manager employee at a restaurant.</p> <p>Given that an employee is a manager employee, when the owner clicks the gear icon next to their name on the employee list, the demote option appears, and the promote option does not appear.</p> <p><b>Scenario:</b> An employee has been demoted to a regular employee.</p> <p>Given that an employee has been demoted to a regular employee, when the employee logs in, they can only access the employee home page, profile page, coupon page, and achievement page with no editing permissions.</p> <p><b>Scenario:</b> An employee has been promoted to a manager employee.</p> <p>Given that an employee has been promoted to a manager employee, when the employee logs in, they can access all the same pages as an owner except for restaurant name and address editing and employee account creation pages, and the gear button next to employees on the employee list does not appear.</p>

We did not include acceptance criteria for stories carried over from the previous sprint, as acceptance criteria was already written for those stories in the previous sprint.

## Sprint 7: Team Task Assignments

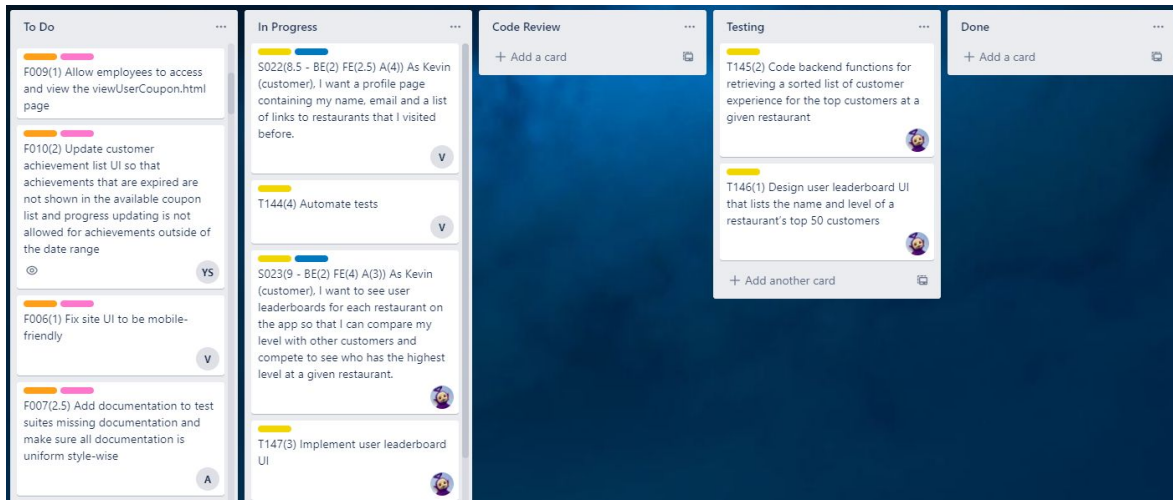
Task ID	Cost	Priority	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Assignment
T150	0.5	Medium	0	0.5	0	0	0	0	0	Vanessa
T151	1	Medium	0	1	0	0	0	0	0	Vanessa
T152	0.5	Medium	0	0.5	0	0	0	0	0	Vanessa
T144	4	Medium	0	4	0	0	0	0	0	Vanessa
T147	3	Medium	0	0	0	0	0	3	0	Jiawei
T148	3	Medium	0	3	0	0	0	0	0	Jiawei
F005	1	High	0	1	0	0	0	0	0	Yolanda
F006	1	High	0	0	0	1	0	0	0	Vanessa
F007	2.5	High	0	0	0	0	0	2.5	0	Anson
F009	1	High	0	1	0	0	0	0	0	Vanessa
F010	2	High	0	2	0	0	0	0	0	Yolanda
F011	1	High	0	1	0	0	0	0	0	Yolanda
F012	1	High	0	0	0	1	0	0	0	Jiawei
T153	0.5	Medium	0	0	0.5	0	0	0	0	Vanessa
T154	1	Medium	0	0	1	0	0	0	0	Vanessa
T155	0.5	Medium	0	0	0.5	0	0	0	0	Vanessa
T156	0.5	Medium	0	0	0.5	0	0	0	0	Vanessa
T157	1	Medium	0	0	1	0	0	0	0	Vanessa
T158	1	Medium	0	0	1	0	0	0	0	Vanessa
T159	1	Medium	0	0	1	0	0	0	0	Vanessa
T160	0.5	Medium	0	0	0.5	0	0	0	0	Vanessa
T161	0.5	Medium	0	0	0.5	0	0	0	0	Vanessa
T162	3	Medium	0	0	0	3	0	0	0	Vanessa
T163	0.5	Medium	0	0.5	0	0	0	0	0	Anson
T164	0.5	Medium	0	0.5	0	0	0	0	0	Anson
T165	2	Medium	0	0	2	0	0	0	0	Anson

T166	0.5	Medium	0	0	0	0.5	0	0	0	Anson
T167	0.5	Medium	0	0	0	0.5	0	0	0	Anson
T168	1	Medium	0	0	0	1	0	0	0	Anson
T169	3.5	Medium	0	0	0	0	2	1.5	0	Anson
T170	0.25	Medium	0	0.25	0	0	0	0	0	Jiawei
T171	0.5	Medium	0	0.5	0	0	0	0	0	Jiawei
T172	0.5	Medium	0	0.5	0	0	0	0	0	Jiawei
T173	0.75	Medium	0	0.75	0	0	0	0	0	Jiawei
T174	0.5	Medium	0	0	0.5	0	0	0	0	Jiawei
T175	1	Medium	0	0	1	0	0	0	0	Jiawei
T176	3	Medium	0	0	3	0	0	0	0	Jiawei
T177	0.5	Low	0	0.5	0	0	0	0	0	Yolanda
T178	0.25	Low	0	0.25	0	0	0	0	0	Yolanda
T179	0.75	Low	0	0.75	0	0	0	0	0	Yolanda
T180	2	Low	0	0	0	2	0	0	0	Vanessa
T181	1	Low	0	0	0	1	0	0	0	Vanessa
T182	1	Low	0	0	0	1	0	0	0	Vanessa
T183	2	Low	0	0	0	2	0	0	0	Vanessa
T184	1	Low	0	0	1	0	0	0	0	Yolanda
T185	1	Low	0	0	1	0	0	0	0	Yolanda
T186	1	Low	0	0	0	1	0	0	0	Yolanda
T187	0.5	Low	0	0	0.5	0	0	0	0	Yolanda
T188	1.5	Low	0	0	1	0.5	0	0	0	Yolanda
T189	3	Low	0	0	1.5	1.5	0	0	0	Yolanda
T190	1	Low	0	0	0	1	0	0	0	Yolanda

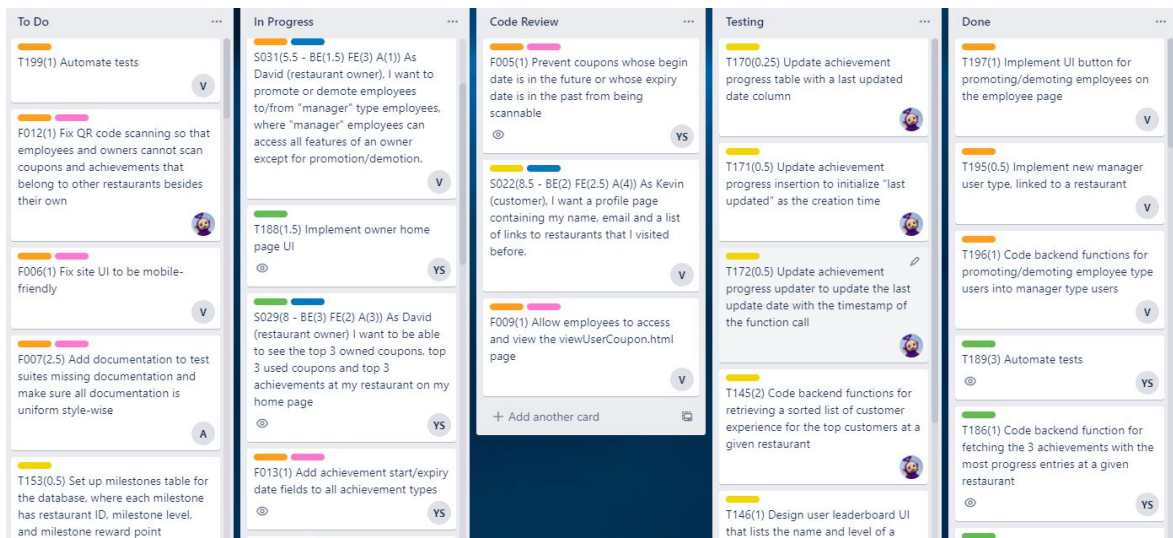
T191	0.25	Low	0	0	0	0.25	0	0	0	Yolanda
T192	0.5	Low	0	0	0	0.5	0	0	0	Yolanda
T193	0.75	Low	0	0	0	0.75	0	0	0	Yolanda
T194	1	Low	0	0	0	1	0	0	0	Yolanda
T195	0.5	High	0	0	0.5	0	0	0	0	Vanessa
T196	1	High	0	0	1	0	0	0	0	Vanessa
T197	1	High	0	0	1	0	0	0	0	Vanessa
T198	2	High	0	0	2	0	0	0	0	Vanessa
T199	1	High	0	0	1	0	0	0	0	Vanessa
Total Points Completed:			0	18.5	21	22	2	7	0	
Total Points Left:			70.5	52	31	9	7	0	0	

# Sprint 7: Task Board

August 3



August 5





August 7

A Kanban board for August 7 with five columns: To Do, In Progress, Code Review, Testing, and Done. The background is a dark blue gradient.

- To Do:**
  - (1) Add/fix documentation for all CSS and HTML files (V)
  - F006(1) Fix site UI to be mobile-friendly (V)
  - (1) Add/fix documentation for all files and functions in databaseHelpers
  - (1) Add/fix documentation for all files and functions in routes
  - (1) Fix all magic values in code base
  - + Add another card
- In Progress:**
  - F007(2.5) Add documentation to test suites missing documentation and make sure all documentation is uniform style-wise (A)
  - (1) Split up test suites so that each .py file in tests only covers 1 function (A)
  - + Add another card
- Code Review:**
  - + Add a card
- Testing:**
  - + Add a card
- Done:**
  - F009(1) Allow employees to access and view the viewUserCoupon.html page (V)
  - S022(8.5 - BE(2) FE(2.5) A(4)) As Kevin (customer), I want a profile page containing my name, email and a list of links to restaurants that I visited before. (V)
  - F005(1) Prevent coupons whose begin date is in the future or whose expiry date is in the past from being scannable (YS)
  - S026(6.5 - BE(2) FE(1.5) A(3)) As Kevin (customer) I want to be able to see my 3 most recently updated achievements on my home page.
  - S031(5.5 - BE(1.5) FE(3) A(1)) As David (restaurant owner), I want to promote or demote employees to/from "manager" type employees.

August 9

A Kanban board for August 9 with five columns: To Do, In Progress, Code Review, Testing, and Done. The background is a dark blue gradient.

- To Do:**
  - F021(1) Add/fix documentation for all files and functions in routes
  - F022(1) Fix all magic values in code base
  - F018(1) In the in-progress tab of restaurant achievement list view for customers, hide achievements that have been expired for more than 6 months.
  - F019(1) In the redeemed coupon list view for customers, hide coupons that have been expired for more than 6 months.
  - + Add another card
- In Progress:**
  - F006(1) Fix site UI to be mobile-friendly (V)
  - F020(1) Add/fix documentation for all CSS and HTML files (V)
  - + Add another card
- Code Review:**
  - F007(2.5) Add documentation to test suites missing documentation and make sure all documentation is uniform style-wise and split up test suites so that each .py file in tests only covers 1 function (A)
  - F008(1) Add/fix documentation for all files and functions in databaseHelpers
  - + Add another card
- Testing:**
  - + Add a card
- Done:**
  - F009(1) Allow employees to access and view the viewUserCoupon.html page (V)
  - S022(8.5 - BE(2) FE(2.5) A(4)) As Kevin (customer), I want a profile page containing my name, email and a list of links to restaurants that I visited before. (V)
  - F005(1) Prevent coupons whose begin date is in the future or whose expiry date is in the past from being scannable (YS)
  - S026(6.5 - BE(2) FE(1.5) A(3)) As Kevin (customer) I want to be able to see my 3 most recently updated achievements on my home page.

August 10

To Do  
+ Add a card

In Progress  
+ Add a card

Code Review  
+ Add a card

Testing  
+ Add a card

Done

F018(1) In the in-progress tab of restaurant achievement list view for customers, hide achievements that have been expired for more than 6 months.

F006(1) Fix site UI to be mobile-friendly

F023(1) Rename all test files so that they are in the form test<TestedFunctionName>.py

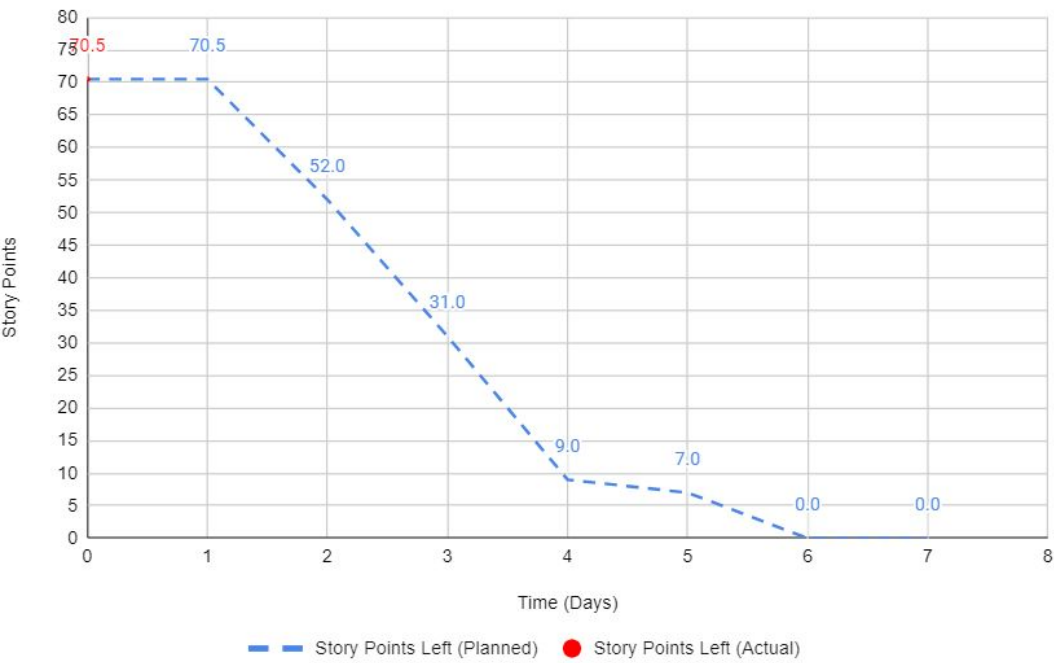
F008(1) Add/fix documentation for all files and functions in databaseHelpers

F007(2.5) Add documentation to test suites missing documentation and make sure all documentation is uniform style-wise and split up test suites so that each .py file in tests only covers 1 function

# Sprint 7: Burndown Chart

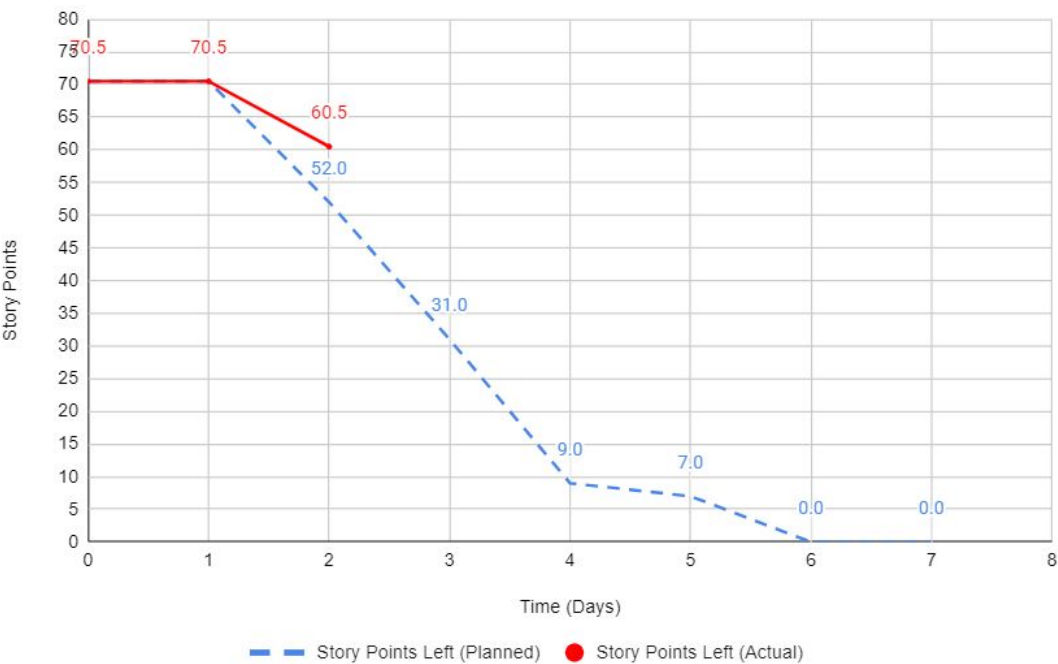
August 3

Burndown Chart



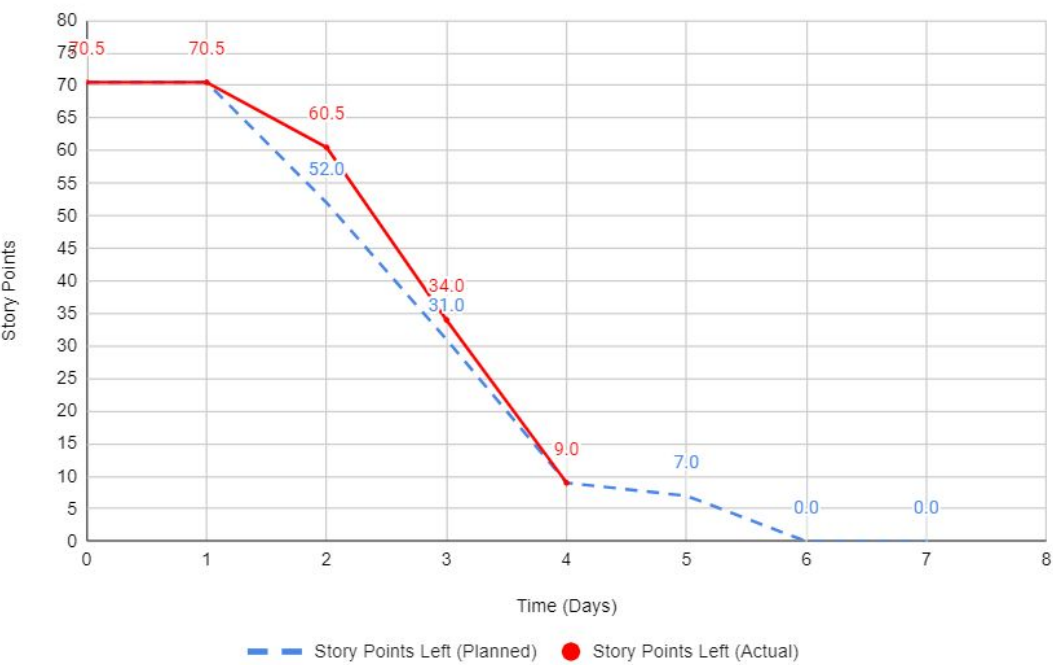
August 5

Burndown Chart



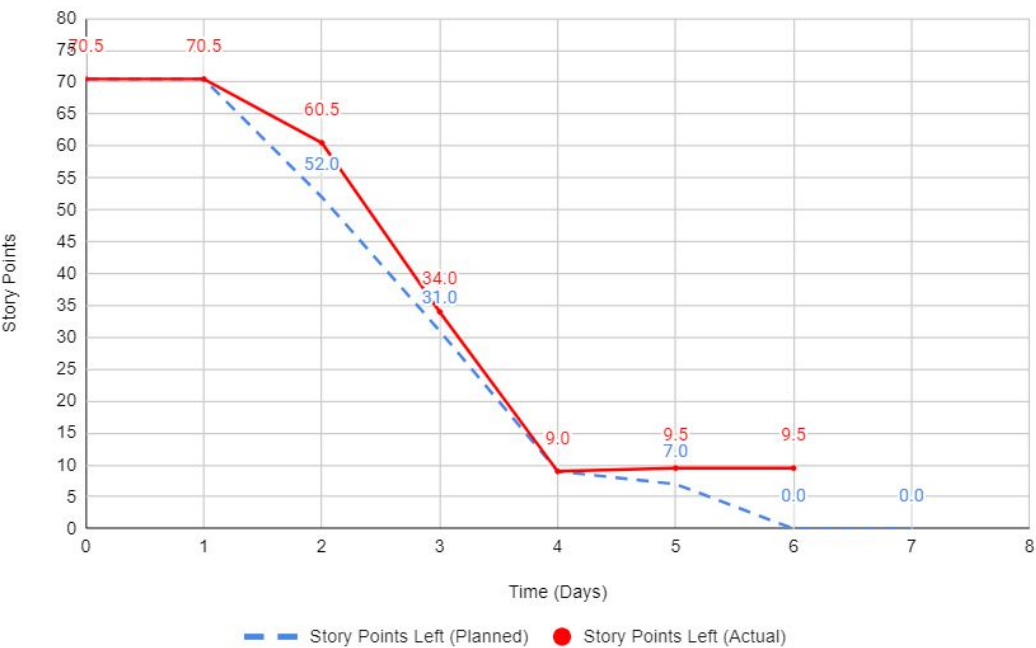
August 7

Burndown Chart



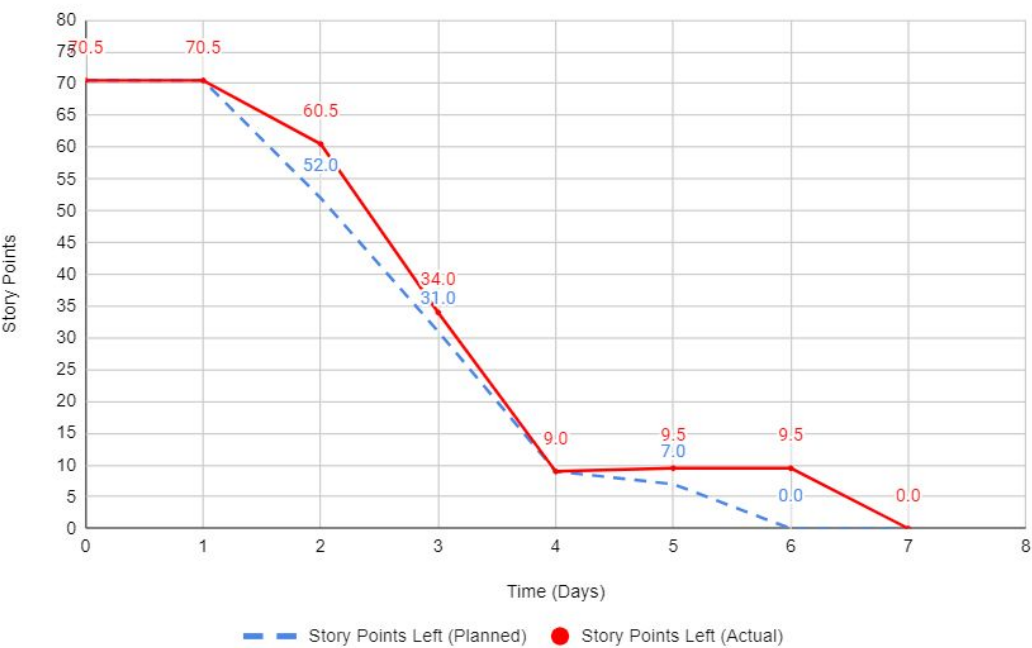
# August 9

Burndown Chart

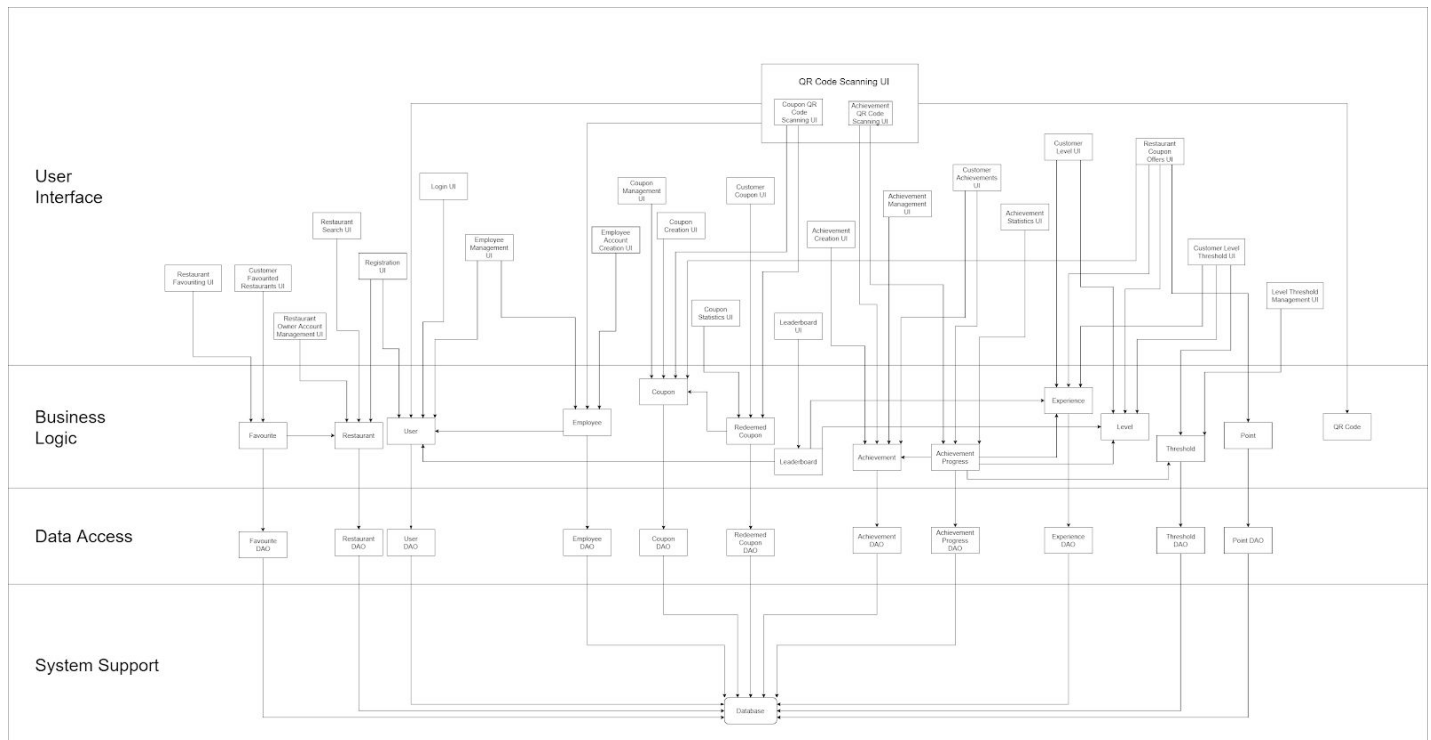


# August 10

Burndown Chart



# High-level Architecture



See the full system architecture diagram at

[https://github.com/CSCC01/team\\_01-project/tree/master/system\\_design.](https://github.com/CSCC01/team_01-project/tree/master/system_design.)

Component	Description	Dependencies
Login UI	This component is responsible for providing an interface for a user to enter user email and password, and displaying an error message if the user email and password do not match an existing account.	This component sends the email and password input to the user component and receives an error message if the input does not match an existing account in the database; otherwise receives an empty error message list.
Registration UI	This component is responsible for providing an interface for a user to enter name, email, password information for creating an account as well as providing an interface for owners to enter restaurant name and address information for setting up a restaurant, and displaying an error message if any of the input is not valid.	<p>This component sends the name, email, and password input to the user component and receives a list of error messages for any invalid input, otherwise an empty list if all input is valid.</p> <p>This component sends the restaurant name and address to the restaurant component, and receives a list of error messages for any invalid input, otherwise an empty list if all input is</p>

		valid and the account has been registered.
Restaurant Search UI	This component is responsible for providing an interface for a customer to enter a search term as input from a user, and displaying a list of restaurants related to that search term.	This component sends the search term input to the restaurant component and receives a list of restaurants in the database whose names match the given search term.
Restaurant Owner Account Management UI	This component is responsible for providing an interface for an owner to enter updated restaurant name and address input, and displaying an error message if the name or address are not valid.	This component sends the new restaurant name and address input to the restaurant component, and receives a list of error messages for any invalid input, otherwise an empty list if all input is valid and the information has been updated in the database.
Employee Account Creation UI	This component is responsible for providing an interface for an owner to enter user name, email, and password input for creating an employee account, and displaying error messages if any of the input is not valid.	This component relies on the employee component in order to check if the name, email, and password information for the created employee, and receives a list of error messages for any invalid input, otherwise an empty list if all input is valid and the employee account has been created.
Employee Management UI	This component is responsible for displaying a list of all employees that work at a restaurant to owners and managers, and providing an interface for an owner to select which employees to delete, promote, or demote.	<p>This component sends the owner or manager's restaurant ID to the employee component and gets a formatted list of employees to display.</p> <p>This component sends an employee's user ID to the user component in order to promote or demote employees from general user type to manager user type and vice versa.</p>
Coupon Creation UI	This component is responsible for providing an interface for an owner or manager to enter coupon name, description, points cost, level requirement, and start and end date input for creating a coupon, and displaying error messages if any of the input is not valid.	This component sends the coupon name, description, points cost, level requirement, and start and end date information for the created coupon to the coupon component and receives a list of error messages for any invalid input, otherwise an empty list if all input is valid and the coupon has been created.
Coupon Management UI	This component is responsible for displaying a list of all coupons at a	This component sends the owner or manager's restaurant ID to the coupon

	restaurant to an owner or manager, and providing an interface for an owner or manager to select which coupons to delete.	component and gets a formatted list of coupons to display.
Coupon Statistics UI	This component is responsible for displaying to an owner or manager the number of copies held and the number of copies used by customers for every coupon at a restaurant.	This component sends the owner or managers restaurant ID to the redeemed coupon component and gets a formatted list of coupons with the number of used and unused redeemed copies for each coupon.
Achievement Creation UI	This component is responsible for providing an interface for an owner or manager to enter achievement name, type, description, experience reward, point reward, and start and end date input for creating an achievement, and displaying error messages if any of the input is not valid.	This component sends the achievement name, type, description, experience reward, point reward, and start and end date information for the created achievement to the achievement component and receives a list of error messages for any invalid input, otherwise an empty list if all input is valid and the achievement has been created.
Achievement Management UI	This component is responsible for displaying a list of all achievements at a restaurant to an owner or manager, and providing an interface for an owner or manager to select which achievements to delete.	This component sends the owner or manager's restaurant ID to the achievement component and gets a formatted list of achievements to display.
Achievement Statistics UI	This component is responsible for displaying to an owner or manager the number of customers who are in progress and the number of customers who have completed an achievement for every achievement at an owner's restaurant.	This component sends the owner or manager's restaurant ID to the achievement progress component and gets a formatted list of achievements with the number of in progress and complete progress instances for each achievement.
Level Threshold Management UI	This component is responsible for displaying to an owner or manager a list of all level thresholds at a restaurant, as well as providing an interface for an owner or manager to enter customer level and points earned for reaching the level, in order to create, edit, or delete a threshold.	<p>This component sends the level and points information for creating a threshold to the threshold component, and receives a list of error messages for any invalid input, otherwise an empty list if all input is valid and the threshold has been created.</p> <p>This component sends the owner or manager's restaurant ID to the threshold component and gets a</p>



		formatted list of thresholds to display.
Customer Level UI	This component is responsible for displaying to a customer their level and total experience at a restaurant.	<p>This component sends the customer's user ID and a restaurant ID to the experience component and gets the customer's experience value at the restaurant.</p> <p>This component sends the customer's experience value to the level component and gets the calculated customer level.</p>
Restaurant Favouriting UI	This component is responsible for allowing a customer to see if a restaurant is favourited and provides an interface for editing its favourite status.	<p>This component sends a restaurant's ID and the customer's ID to the favourite component and gets data on whether a favourite entry exists.</p> <p>This component sends a restaurant's ID and the customer's ID to the favourite component to update the restaurant's favourite status</p>
Restaurant Coupon Offers UI	This component is responsible for displaying to a customer which coupons they are eligible to and/or can afford to buy at a given restaurant.	<p>This component sends the customer's ID and a restaurant ID to the points component and gets how many points the customer has to check if they can afford a coupon.</p> <p>This component sends a restaurant ID to the coupon component and gets the list of available, unexpired coupons offered at a restaurant.</p> <p>This component sends the customer's ID and a restaurant ID to the experience component and gets the customer's experience value at the restaurant.</p> <p>This component sends the customer's experience value to the level component and gets the calculated customer level to check whether the customer is high enough level to buy a coupon.</p>
Customer Achievements UI	This component is responsible for displaying to a customer their current progress on every achievement a given restaurant.	This component sends the customer's ID and a restaurant ID to the achievement progress component and gets the customer's current progress

		on every achievement at a given restaurant.
Customer Level Thresholds UI	This component is responsible for displaying to a customer which level thresholds they have already passed and which they have not passed at a given restaurant.	<p>This component sends the customer's ID and a restaurant ID to the threshold component and gets the list of thresholds at a given restaurant.</p> <p>This component sends the customer's ID and a restaurant ID to the experience component and gets the customer's experience value at the restaurant.</p> <p>This component sends the customer's experience value to the level component and gets the calculated customer level, to check whether the customer has passed each threshold.</p>
Leaderboard UI	This component is responsible for displaying to an owner, manager, or customer a list of the top 10 customers at a restaurant with the highest experience and level.	This component sends a restaurant ID to the leaderboard component and gets the formatted list of top 10 customers.
Customer Favourited Restaurants UI	This component is responsible for displaying to a customer the list of all restaurants that they have favourited.	This component sends the customer's ID to the favourite component and gets a formatted list of all their favourited restaurants.
Customer Coupon UI	This component is responsible for displaying to a customer a list of all unused copies of a coupon that they have redeemed.	This component sends the customer's ID to the redeemed coupon component and gets the formatted list of all unused coupon copies that they have redeemed.

QR Code Scanning UI	This component is responsible for displaying a QR code for an employee or owner to scan, as well either displaying an error message if the code cannot be scanned or displaying a scan success message.	<p>This component sends a scanning URL to the QR code component and gets the path to the generated QR codes to display.</p> <p>This component sends the scanning user's ID to the user component and gets the user's account type, to check if they have permissions to scan QR codes.</p> <p>This component sends the scanning owner or employee's ID to the employee component and gets their restaurant id, to check if the employee or owner has permissions to scan QR codes for the coupon's restaurant.</p>
Achievement QR Code Scanning UI	This component is responsible for displaying an achievement progress update QR code for an employee or owner to scan, as well either displaying an error message if the achievement cannot be scanned or displaying a scan success message.	<p>This component sends the achievement ID to the achievement component and gets the formatted achievement information including expiry status.</p> <p>This component sends the achievement ID and customer ID to the achievement progress component and gets the formatted achievement progress information including completion status.</p>
Coupon QR Code Scanning UI	This component is responsible for displaying a coupon use QR code for an employee or owner to scan, as well either displaying an error message if the coupon cannot be scanned or displaying a scan success message.	<p>This component sends the coupon ID to the coupon component and gets the formatted coupon information including expiry status.</p> <p>This component sends the coupon ID and customer ID to the redeemed coupon component and gets the formatted coupon information if the coupon has not been used and exists, otherwise gets an error message.</p>

User	<p>This component is responsible for validating the given information for creating a new user; if the information is invalid, this component returns an error, otherwise this component creates a new user.</p> <p>This component is also responsible for searching for users by ID in a database, and then formatting and returning their information in the expected format.</p> <p>This component is also responsible for deleting users from the database.</p>	<p>This component relies on the user DAO to insert, retrieve, update, and delete user data in the database.</p>
Restaurant	<p>This component is responsible for validating the given information for creating a new restaurant; if the information is invalid, this component returns an error, otherwise this component creates a new restaurant.</p> <p>This component is also responsible for searching for specific restaurants in a database by ID or by name, and formatting returning their information in the expected format.</p> <p>This component is also responsible for validating the information for updating restaurant name and address; if this information is invalid, the component returns an error, otherwise this component updates the restaurant name and address.</p>	<p>This component relies on the restaurant DAO to insert, retrieve, and update the restaurant data in the database.</p>
Employee	<p>This component is responsible for linking an employee account to the restaurant they work at. It is also responsible for sending new employee data to the user component to create new employee accounts, and returning the error messages returned by</p>	<p>This component sends the user name, email, and password data to the user component in order to create an employee account and gets a list of error messages if there is invalid data, otherwise an empty list when all data is valid and the account has been created.</p>

	<p>the user component.</p> <p>This component is also responsible for searching for employees at a given restaurant, and returning their information in the expected format.</p> <p>This component is also responsible for deleting employees, promoting regular type employees to manager type employees, and demoting manager type employees to regular type employees.</p>	<p>This component sends the user ID of an employee and the new user type of that employee (manager or regular) to the user component to change the user account type from regular employee to manager employee and vice versa.</p> <p>This component relies on the employee DAO to insert, retrieve, and delete the employee data in the database.</p>
Coupon	<p>This component is responsible for validating the given information for creating a new coupon; if the information is invalid, this component returns an error, otherwise this component creates a new coupon in the database.</p> <p>This component is also responsible for searching for coupons in a database by ID or by restaurant, and returning their information in the expected format.</p> <p>Also, this component checks whether a coupon has reached its start date and/or passed its expiry date, and formats the achievement's expiry status accordingly.</p> <p>This component also deletes coupons from the database.</p>	<p>This component relies on the coupon DAO to insert, retrieve, and delete the coupon data in the database.</p>
Redeemed Coupon	<p>This component is responsible for creating a redeemed copy of a coupon offered at a restaurant for a given customer.</p> <p>This component is also responsible for searching for all unused redeemed coupons by customers, and formatting and returning the related coupon</p>	<p>This component sends the coupon ID associated with the redeemed coupon to the coupon component and gets the formatted information of the coupon.</p> <p>This component relies on the redeemed coupon DAO to insert, retrieve, and update the redeemed coupon data in the database.</p>

	<p>information for the redeemed copy in the expected format.</p> <p>This component is also responsible for validating that a redeem coupon has not been used, and if so, marking a redeemed coupon as used.</p>	
Achievement	<p>This component is responsible for validating the given information for creating a new achievement; if the information is invalid, this component returns an error, otherwise this component creates a new achievement in the database.</p> <p>This component is also responsible for searching for achievements in a database by ID or by restaurant, and returning their information in the expected format.</p> <p>Also, this component checks whether an achievement has reached its start date and/or passed its expiry date, and formats the achievement's expiry status accordingly.</p> <p>This component also generates descriptions for achievements based on the achievement's type and values.</p> <p>This component also deletes achievements from the database.</p>	<p>This component relies on the achievement DAO to insert, retrieve, and delete the achievement data in the database.</p>
Achievement Progress	<p>This component is responsible for validating that a progress entry does not exist for a given user on a given achievement into the database, and if it does not, adding a progress entry into the database.</p> <p>This component is also responsible for checking whether the progress on a given</p>	<p>This component sends the ID of the achievement associated with the achievement progress entry to the achievement component and gets the formatted information of the achievement.</p> <p>This component sends the customer ID, achievement restaurant ID, and achievement points reward value to the points component to add to the</p>

	<p>achievement is complete, incomplete, or nonexistent, and formatting this data into a status.</p> <p>This component is also responsible for incrementing progress on an achievement by one, and updating the user's points and experience by the amount set as the achievement's reward if the achievement had been completed.</p> <p>This component is also responsible for checking if completing an achievement has resulted in a user passing a level threshold, and updating points accordingly.</p>	<p>customer points with the reward amount when an achievement progress reaches completion.</p> <p>This component sends the customer ID, achievement restaurant ID, and achievement experience reward value to the experience component to add to the customer experience with the reward amount when an achievement progress reaches completion.</p> <p>This component sends the customer ID and achievement restaurant ID to the experience component and gets the customer's experience data.</p> <p>This component sends an experience value to the level component and gets the calculated customer level data.</p> <p>This component sends the customer's upgraded level to the threshold component and gets the level threshold with a matching level, otherwise no threshold is returned.</p> <p>This component relies on the achievement progress DAO to insert, retrieve, and update the progress data in the database.</p>
Point	<p>This component is responsible for validating that a points entry does not already exist for a given customer at a given restaurant, and if it does not, adding a points entry into the database.</p> <p>This component is also responsible for searching for the number of points a customer has at a given restaurant, as well as updating the customer's points amount by a given increment.</p>	<p>This component relies on the point DAO to insert, retrieve, and update the points data in the database.</p>
Experience	<p>This component is responsible for validating that an experience entry does not already exist for a given customer at a given restaurant, and if does not not,</p>	<p>This component relies on the experience DAO to insert, retrieve, and update the experience data in the database.</p>

	<p>adding an experience entry into the database.</p> <p>This component is also responsible for searching for a customer's experience at a given restaurant, as well as updating the customer's experience amount by a given increment.</p>	
Level	This component is responsible for calculating a level based on a given amount of experience.	This component does not rely on any other component.
QR Code	This component is responsible for generating QR codes based on given URLs.	This component does not rely on any other component.
Leaderboard	This component is responsible for sorting the experience values of all customers at a restaurant, and retrieving and formatting the name, rank, and level information for the 10 customers with the highest experience in the expected format.	<p>This component sends a restaurant ID to the experience component and gets the list of experience values for all customers at a restaurant.</p> <p>This component sends a customer's user ID to the user component and gets the customer's formatting user information.</p> <p>This component sends an experience value to the level component and gets the calculated customer level data.</p>
Favourite	<p>This component is responsible for validating that a favourite entry does not exist for a restaurant, and if not, adding a favourite to the database marking a restaurant as a customer's favourited restaurant.</p> <p>This component is also responsible for searching for favourites by customer, and retrieving and formatting the related restaurant's information for each favourite into the expected format.</p> <p>This component is also responsible for validating that a favourite entry does exist for a</p>	<p>This component sends the restaurant ID saved in the favourite entry and gets the formatted information of each restaurant that has been favourited.</p> <p>This component relies on the favourite DAO to insert, retrieve, and delete the favourite data in the database.</p>



	restaurant, and if so, deleting favourites from the database.	
Threshold	<p>This component is responsible for validating the given information for creating a new threshold; if the information is invalid, this component returns an error, otherwise this component creates a new threshold in the database for the given restaurant.</p> <p>This component is responsible for validating whether the given level information for creating a new threshold matches the level of an existing threshold; if so, the older threshold is updated with the new points information given.</p> <p>This component also searches for thresholds in the database by restaurant, and formats and returns the threshold information in the expected format</p> <p>This component also validates updated points reward data for a threshold, and if it is, it updates the point data of the threshold in the database.</p> <p>This component deletes thresholds from the database.</p>	This component relies on the threshold DAO to insert, retrieve, update, and delete the level threshold data in the database.
User DAO	This component is responsible for providing access to user data insertion, selection/retrieval, updating, and deletion in the database.	This component relies on the database to store user data.
Restaurant DAO	This component is responsible for providing access to insertion, selection/retrieval, and updating restaurant data in the database.	This component relies on the database to store restaurant data.
Employee DAO	This component is responsible for providing access to employee data insertion, selection/retrieval, and deletion in the database.	This component relies on the database to store employee data.

Coupon DAO	This component is responsible for providing access to coupon data insertion, selection/retrieval, and deletion in the database.	This component relies on the database to store coupon data.
Redeemed Coupon DAO	This component is responsible for providing access to redeemed coupon data insertion, selection/retrieval, and updating in the database.	This component relies on the database to store redeemed coupon data.
Achievement DAO	This component is responsible for providing access to achievement data insertion, selection/retrieval, and deletion in the database.	This component relies on the database to store achievement data.
Achievement Progress DAO	This component is responsible for providing access to achievement progress data insertion, selection/retrieval, and updating in the database.	This component relies on the database to store achievement progress data.
Point DAO	This component is responsible for providing access to points data insertion, selection/retrieval, and updating in the database.	This component relies on the database to store points data.
Experience DAO	This component is responsible for providing access to experience data insertion, selection/retrieval, and updating in the database.	This component relies on the database to store experience data.
Favourite DAO	This component is responsible for providing access to favourites data insertion, selection/retrieval, and deletion in the database.	This component relies on the database to store favourites data.
Threshold DAO	This component is responsible for providing access to level threshold data insertion, selection/retrieval, updating, and deletion in the database.	This component relies on the database to store level threshold data.
Database	The database is responsible for storing user, restaurant, employee, coupon, redeemed coupon, achievement, achievement progress, points, experience, favourites and threshold data for the application.	The database does not have dependencies.

## Changes to High Level Architecture

The changes in our final version of our system design were mainly in the User Interface layer, with a few changes in the Business Logic and Data Access Layer.

In the Business Logic layer, we removed the Character Item and Inventory Item components as we decided to remove the story related to custom characters from the product backlog. We also added a QR Code component to encompass the logic needed to generate QR codes, a Leaderboard component to handle data processing for the leaderboard display, and the Favourite and Threshold components due to two new features we added to our backlog, the level threshold feature for restaurants and the restaurant favouriting feature for customers.

For each change in the Business Logic components we changed the Data Access layer accordingly; the Character Item DAO and Inventory Item DAO were removed, and the Favourite DAO and Threshold DAO were added. No QR Code DAO was added, as the QR codes were not stored in the database, and were instead generated as needed based on URLs.

In the User Interface Layer, we removed the Character UI as we decided to scrap that feature, and we added the Restaurant Favouriting UI and Customer Favourited Restaurants UI for favouring restaurants and viewing favourited restaurants respectively in the new favourites feature. We also added the Level Threshold Management UI for owners and Customer Level Thresholds UI for customers for the new threshold feature.

We also split up some components into multiple components so that the scope of each component's responsibilities is not too large, for better cohesion in our system design. We split up the Employee Management UI into the Employee Management UI and the Employee Account Creation UI. We split the Coupon Management UI into the Coupon Management UI, Coupon Creation UI, and Coupon Statistics UI. We also split the Achievement Management UI into the Achievement Management UI, Achievement Creation UI, and Achievement Statistics UI.

We split in Customer Coupon UI into the Restaurant Coupon Offers UI and the Customer Coupon UI as well, one for customers redeeming coupons at a restaurant and one for customers seeing the redeemed coupons they have. We added a distinction for the two types of QR Code Scanning UIs, the Achievement QR Code Scanning UI and the Coupon QR Code Scanning UI, as some of their responsibilities are different. We also added Restaurant Owner Account Management UI as a separate restaurant management UI as we added the ability to edit restaurant name and address as a feature.

Some component dependencies were adjusted as well so that the final design better adheres to our final release of our product. Please see the table above for the descriptions of all dependencies.

## Verification

We decided that for all features, every function involving business logic that is used by the application in the production environment needed to be unit tested. This is to help prevent the possibility of crashing or other unexpected behaviour from occurring on the live site.

The test suite can be found at

[https://github.com/CSCC01/team\\_01-project/tree/master/demo3/test](https://github.com/CSCC01/team_01-project/tree/master/demo3/test).

There is a test folder for each file in demo3/databaseHelpers, and one test suite file in the folder for each function used by the application in the production environment, from the corresponding demo3/databaseHelpers file.

# Validation

In this deliverable, our main method of system validation was through emailing the client with our questions and demo videos.

The screenshots from these emails and the demo videos can be found at [https://github.com/CSCC01/team\\_01-project/tree/master/validation\\_evidence](https://github.com/CSCC01/team_01-project/tree/master/validation_evidence).

The questions asked in the emails include requesting feedback on the features shown in the attached demo videos, as well as questions such as:

- What do you expect from being able to see redeemed coupons? Should each coupon list a number for held and used copies, or should each redeemed copy be listed separately?
- Our current setup has it so that if you've already bought a coupon with points and the owner deletes the coupon afterward, you can still use the coupon, but if you've started making progress on an achievement but it gets deleted before you complete it, you can no longer work on that achievement. Is this okay?
- We added the ability for employees to see a list of the coupons offered at the restaurant they work at. Should employees also be able to see the same statistics page for how many of each coupon has been used / obtained that the owner can see?
- We set the leaderboard for each restaurant to show a maximum of 10 users. Should this number be increased or is it okay?
- Is the current scope of features available to managers okay or should the scope be increased/decreased? Currently managers can access almost all the same features as an owner (e.g. coupon and achievement creation / deletion), EXCEPT for: employee adding/deleting/promoting/demoting, and editing restaurant name and address.
- Is there any information besides what we currently show that you think we need to display on the home page for customers, employees, and/or owners?

For our question about how redeemed coupons should be displayed, the client said they would prefer seeing a number for held and used copies over our existing implementation where each redeemed copy was listed separately, so we added a fix to the coupon statistics page so that each unique coupon was listed once and had numerical statistics information.

For our sprint 5 demo, we received the feedback that our application looked hard to navigate on mobile. As a result, we added scaling on element sizes based on window size for our main pages, and added an additional fix task for the final release so that all pages of the site were adjusted to scale down appropriately for mobile view.

For our sprint 6 demo, we received the feedback that there should be an option to set an expiry date for all coupons, not only in the "Visit x times" type as we previously

implemented. As a result, we added a fix allowing the start date field, end date field, and “make achievement expiry indefinite” checkbox to be visible for all 4 achievement types. We also got the feedback that there should be an option to promote employees at a restaurant to a higher clearance that can access various previously owner-exclusive features. As a result, we added and later implemented a new story for being able to promote employees from a regular employee to a manager employee, and demote employees from a manager to a regular employee.

For our sprint 7 demo, we received the feedback that expired in-progress achievements and redeemed coupons should no longer be shown if they have been expired for more than 6 months. As a result, we added a fix so that these were no longer shown if the expiry date was more than 6 months in the past. We also received feedback that owners should be able to delete expired achievements and coupons. However, we did not have the time to add a new story for this, as this was our last sprint. As a result, we were not able to add this feature into our final release.

# Project Velocity

## Estimated Velocity

Sprint 1: 31 points  
Sprint 2: 31.75 points  
Sprint 3: 39.5 points  
Sprint 4: 40.5 points  
Sprint 5: 43.5 points  
Sprint 6: 46.5 points  
Sprint 7: 70.5 points

For our first two sprints, we estimated our project velocity by dividing the total points in our backlog by the number of total sprints for this project. However, based on feedback in deliverable 3 and our own team discussion, we realized that this was not a good reflection of our team's actual capacity to work during each iteration. As a result, for sprints 3 to 7, we estimated our project velocity for every sprint by summing together the number of points each member of the team was able to commit to the sprint.

## Actual Velocity

Sprint 1: 23.25 points  
Sprint 2: 25.85 points  
Sprint 3: 23 points  
Sprint 4: 40.5 points  
Sprint 5: 38.5 points  
Sprint 6: 37.5 points  
Sprint 7: 78 points

Our velocity varied for every sprint, and was generally higher in later weeks. Some actual velocity exceeded expected as fix tasks were added midway through the sprint.

## Our Project Throughout The Term

In Deliverable 1, we established our team agreement, initial roles, and team meeting times, and did not begin project planning or development. We also got assigned to our client, PickEasy, during this deliverable.

In Deliverable 2, we drafted our personas and user stories for the product based on the client's requirements. These personas and user stories were pretty weak and lacking in many areas. Firstly, we only had two personas, one for a restaurant owner user and one for a customer user. We were missing an employee persona and employee related user stories. We also had several redundant user stories, and the majority of our stories were vague and not testable, and did not give a clear impression of how they should be implemented. We also lost a team member near the end of this deliverable, resulting in our team having to use our contingency plan for the first and only time, as well as requiring us to make changes to the roles of team members for deliverable 3.

In Deliverable 3, we rewrote most of our user stories based on our deliverable 2 feedback and based on our need to estimate the work required for each story, which required us to clarify each user story so that we understood what should be implemented. We also added a third persona, the employee persona, and associated stories. After estimating and assigning priorities to all stories, we completed an initial Product Backlog, and selected stories to implement in our first sprint. We also determined the length of our sprints, the start and end times for each sprint, and the schedule for the team meetings required for each sprint: our daily standup times, and our sprint retrospective meeting time. We created our initial system design, set up our task board and burndown chart, and began our first iteration in the development process, resulting in our very first release. At this point, our Agile development process still had a lot of flaws, namely many steps were missing.

In Deliverable 4, we corrected many of the flaws in our Agile process. We added sprint review and sprint planning meetings to our meeting schedule, and moved our retrospective meeting time to after the end of the sprint. We also again revised many user stories to be more specific about its requirements and expected implementation. We also stopped splitting front-end and back-end work between different developers, as we found having one developer on a single feature was more efficient. However, the biggest change in our process during this deliverable was the addition of product validation and verification.

We formalized our code review process through the use of pull requests on github, and we added unit test suites for all of our backend code. In the process of beginning to write unit tests, we discovered that our code was not testable at all, and it also did not adhere to our system design structure. As a result, we spent an iteration entirely focused on refactoring our whole application and writing tests, which resulted in our second release being much more incomplete than planned. During this refactoring process, we also revised our system design significantly, both so that it more closely adhered to our actual



implementation and so that it was more complete with less coupling based on the deliverable 3 feedback.

We also started to demo new features to the client once every iteration as part of the verification process, and used the feedback from these demos to revise our features, create new stories, and adjust priorities on existing stories. We also created a Development branch on our git in order to accommodate the addition of more refined validation and verification processed; we merged feature branches to Development during the sprint after it has been unit tested and code reviewed via pull request, and we merged Development to master at the end of sprints after receiving client approval.

In Deliverable 5, we did not have significant changes to our iteration process as we had mostly ironed out the kinks in our Agile process in the previous deliverable. We did adjust our code review process so that only one other team member was responsible for reviewing a pull request rather than all 3, in order to be more efficient with our time. We also changed our sprint end/start time to improve productivity productivity, and ended up having to switch from using a mix of emails and live demos in meetings for verification to purely video demos by email, as our new sprint schedule didn't line up very well with the client meeting schedule. We ended up implementing the majority of our product's features during this deliverable, resulting in a final release much more complete than our second one.

Working on this project, we felt that the most important thing we learned was the Agile development process in practice, i.e. how Agile is used when actually developing a software product. We learned the importance of various aspects of Agile, such as the importance of writing testable code in order to prevent bugs in production, the importance of writing good user stories so that we can estimate our work each sprint accurately and implement features correctly, and the importance of frequently validating your product with the client as client's wants and requirements often end up changing. We also gained a better understanding of the tools used to implement these aspects of Agile development.

## Deliverable 5 Retrospective

For the sprints in this deliverable, Sprint 5 and 6 followed our sprint plans pretty closely, though we had one or two stories incomplete at the end of the sprint. Sprint 5 however was the sprint where we decided to change our release plan, so the end of sprint and retro was one day later than originally planned. This change resulted in a later demo time than anticipated, and the client feedback-based fixes for features implemented in sprint 5 had to be added in the middle of sprint 6.

Sprint 7 followed our plan closely in that we managed to complete all stories and tasks in the sprint plan, however the sprint 7 plan itself was a lot different than anticipated prior to our sprint planning meeting, as we found as we discussed our progress that there were a lot of fixes still needed in preparation for the final release. This resulted in our estimated and actual velocity for sprint 7 being significantly higher than the velocities of the previous 6 sprints. We also had to finish stories earlier on in the sprint than usual, so that we had time to get client feedback on the features prior to the final release.

We did not end up needing to use any of our contingency plans during this deliverable.

The majority of our project's features were implemented in the last 3 sprints, i.e. during this deliverable. As a result, our final release is a lot more complete than our deliverable 4 release. Refactoring in Deliverable 4 caused our productivity in the previous deliverable to be much lower than hoped, as well as adjusting to all the changes made during that deliverable to our interaction process.

As we had already gotten used to a more complete Agile process by the end of deliverable 4, for deliverable 5, the iterations went much more smoothly and we got more done. As a result our actual velocities for the deliverable 5 sprint were on average higher than those sprints in the previous two deliverables. Our final product implemented all remaining stories in our backlog. We also made sure that every new feature was approved by the client.

## Release

Our most recent release can be found at <https://pickeasy-beta.herokuapp.com>.