

Iteration 1 Report

Accessing and running the system

Running ruby twitter.rb runs a twitter program that runs in timed intervals that fetches incoming tweets and processes them (account checks, sending to database etc).

Run ruby site.rb to access website

Accounts:

Admin:

Username: admin

Password: password

Customer:

Username: customer

Password: password

Customer accounts can also be created by signing up for an account on the website.

Stories

We have not added any extra stories so the story submission is the same as last semester. However the stories we planned to implement this iteration had to be changed due to the knowledge we had acquired this semester on how to implement the features.

Stories we planned to tackle and how they changed

As a user
I want to be able to place an order on twitter
so that I can buy pizza

Priority: 2**Estimated Effort: 40**

Acceptance Criteria:

- When I tweet my order to PizzaShed, I will be able to buy pizza.
- When a tweet is sent, the company will **receive this as an order**

Implemented two databases for orders and non-orders

In order to keep pizza orders and other tweets to the company separate, we had to implement a system to check for the #order in the tweets. This means that those with the hashtag are sent to the orders database and those without sent to the questions database. This clients had not specified that they wanted a table for non-order tweets but we found this would be the simplest and clearest way to do this.

As an order admin
I want to have access to the website
so that I can manage incoming orders

Priority: 3**Estimated Effort: 12**

Acceptance Criteria:

- When I click Orders, I will be able to see a list with all the customer details and there orders.
- When no orders have been placed, a message will show up "**There are no orders**".

Not yet implemented. However this is a small feature that will easily be done in the next iteration

Admins with privilege "order admin" have access to the orders page so that they can read the orders that have been sent in by customers on twitter. The estimated effort was a more as it required the use of sessions which took the group much longer to understand and fully get working than originally thought.

As a customer
I want to be able to see the menu regardless of having an account
so that I know what options I have

Priority: 4 Estimated Effort: 10

Acceptance Criteria:

- When a user with or without an account opens the website, he is able to see the menu.

Once the website was created, this was much easier than anticipated

The menu page is accessible by anyone viewing the website, whether they have an account or not.

As an owner
I want to check where the user lives before letting them make an account
so that only people that are located in Sheffield can sign up

Priority: 5 Estimated Effort: 5

Acceptance Criteria:

- When a user that lives in Sheffield tries to make an account, he will be able to click Sign Up.
- When a user that doesn't live in Sheffield tries to sign up the 'Sign Up' button should be disabled and a message should appear saying "Sorry you have to live in Sheffield to make an account"

We predicted the estimated effort for this story very well as once the stories leading up to this was completed, it was easy to check the sign up form for a non sheffield address. However a more indepth check will be implemented next iteration

Despite the special offers not yet completed, the website runs and works very well and to the clients specification. Even though this story last in the Essential list, this was the second story we tackled as without it the other stories could not be done.

As a user admin

I want to be able to verify customer twitter accounts

so that I know they are a member

Priority: 1 **Estimated Effort: 20**

Acceptance Criteria:

- When a user places an order they are categorised as member or non-member in the queue.

This was more difficult and took longer than first thought as the databases were more complicated

Despite the acceptance criteria not specifying it, a tweet is replied to non-user tweets encouraging them to sign up

We did not implement a queue system for those without account. If they don't have an account on the website when tweeting then they will receive a reply from the company telling them that they need an account.

As a user

I want a tweet from the company once my order has been placed

so that I have some detail about my order and I know that it has been confirmed

Priority: 2 **Estimated Effort: 5**

Acceptance Criteria:

- When I place an order on twitter I get a tweet back confirming that the order has been received.
- If the tweet isn't a valid order then there will not be a confirmation sent

So far we actually send them a tweet if they don't put "#order" at the end of an order

We planned to do this but it's priority for lower than the other stuff, we will do it in next iteration. However, if the tweet sent is not a valid order, then they will receive a tweet telling them so. We thought this would work better as otherwise customers may mis-tweet their order and never know that it was not accepted

As a website admin

I want to be able to access the website

so that I can update the website with special offers and menu

Priority: 7 **Estimated Effort:** 40

Acceptance Criteria:

- The admin uses a password that gives him access to edit the website.
- The admin clicks 'Edit Website' which gives him permission to update the website with special offers and menu.

Accurate estimated effort as we were not able to complete the whole story as we felt other functional stories were more important than getting special offers implemented

Special offers have not yet been implemented but the main admin is able to change the menu from the website. However, we have not yet implemented different levels of admins so only the main admin as yet can do alterations.

As an owner

I want a twitter account that represents the website

so that I can keep orders in the same place and strengthen the brand.

Priority: 1 **Estimated Effort:** 10

Acceptance Criteria:

- The twitter page can be accessed through searching or through a link in the website.
- The twitter account will show some basic and relevant information about the company.

We did not need to change any of this story as it was easy to implement by creating a new twitter account with the name of the business. The twitter page includes a link to the website, and the website a link to the twitter.

As an owner

I want a link to the twitter page on the website

so that users can place orders from there

Priority: 2

Estimated Effort: 5

Acceptance Criteria:

- Given that the website exists the user can click on a link on the website that takes them to the companies twitter account.

This was quickly and easily implemented to the website.

As the owner

I want a website for PizzaShed

so that users can access the menu and the special offers

Priority: 6

Estimated Effort: 35

Acceptance Criteria:

- Given that the website exists the users can access it and they are able to see the menu and the special offers.
- If the url to the website is entered then the website will open

We had hoped to complete this within this iteration but found it wouldn't be feasible to complete this in time as we felt it not essential to the running of the program

As a Public Relations Officer

I want to be able to create competitions on twitter

so that I will attract more business

Priority: 8

Estimated Effort: 17

Acceptance Criteria:

- The Public Relations Officer logs into twitter and can tweet a competition.

We planned to get this set up during this iteration however we felt it a small, not essential feature that could be added at a later date.

As a twitter admin

I want to be able to differentiate tweets placing orders from other types of tweets

so that I can respond to each one appropriately.

Priority: 1

Estimated Effort: 25

Acceptance Criteria:

- When a customer tweets an order he is added to an order list.
- When a customer tweets something to PizzaShed that is not an order he is added to Q/A list.

This was implemented by having seperate tables for different types of tweets (orders and questions). Depending on what # the user tweets decides where it goes

Resubmitted Stories

As a user

I want to be able to log into the website and see special offers specific to my account
so that I will be able to use them

Priority: 3

Estimated Effort: 60

Acceptance Criteria:

- When I log into my website account I will see all the offers specific to my account.
- If I am not logged in then there will be no special offers shown to me

It was not feasible to get this done this iteration however this will be possible for this to be complete by the end of the next deadline

As an owner

I want 4 different admin level accounts
so that I can give different privileges to my employees

Priority: 4

Estimated Effort: 15

Acceptance Criteria:

- The owner can give employees one of 4 different types of account depending on their position in the company.
- Each of these accounts can be logged into
- Each of the accounts will have access to the specific controls necessary: eg the order account will be able to see all of the orders sent

It took longer than anticipated to implement different views for different accounts but as it is now set up it will be easier to implement more admin level accounts during the next iteration

As an order admin

I want a to be able to verify customers orders within an appropriate radius from PizzaShed

so that it applies with our policy.

Priority: 6 **Estimated Effort:** 25

Acceptance Criteria:

- When a customer places an order the program checks if the customer's account is verified as living in Sheffield
- When a customer who doesn't live within the appropriate radius from PizzaShed places an order, he gets a tweet back saying that we can't deliver to them.

So far we have only checked the sign up city for verification that the user is from Sheffield. It may not be possible to create a radius search so we will discuss this with our client

As a Public Relations Officer

I want to be able to create competitions on twitter

so that I will attract more business

Priority: 8 **Estimated Effort:** 17

Acceptance Criteria:

- The Public Relations Officer logs into twitter and can tweet a competition.

This will be done for the second iteration.

As a Public Relations Officer

I want to be able to access twitter accounts
so that I can promote business

Priority: 9 **Estimated Effort:** 14

Acceptance Criteria:

- When I enter the correct password ,I gain access to the account.

This story needs to be discussed with the client as it may not be possible to integrate several twitter accounts into the system

As an owner

I want to be able to view twitter statistics
so that I can make decisions based on them

Priority: 10 **Estimated Effort:** 1

Acceptance Criteria:

- When I click 'Statistics' button, statistics get displayed on the screen.

This is can Be done through twitter. This user story needs to be dicussed with the clients for further details.

As a user

I want to be able to change my details on the site
so that they are always up to date.

Priority: 11 **Estimated Effort:** 25

Acceptance Criteria:

- When I click 'Update Details' I can update my details and save them.
- These details will be viewable instantly so that if the user orders directly afterwards, the new details will be used instead of the old ones

It wasn't possible for this to be completed on the first iteration but it will be done for the second one.

As an order admin

I want to be able to respond to orders for clarification directly from the website

so that I can make sure orders are correct

Priority: 1

Estimated Effort: 50

Acceptance Criteria:

- When a user places an order I can respond to that order for clarification by clicking on it directly from the website

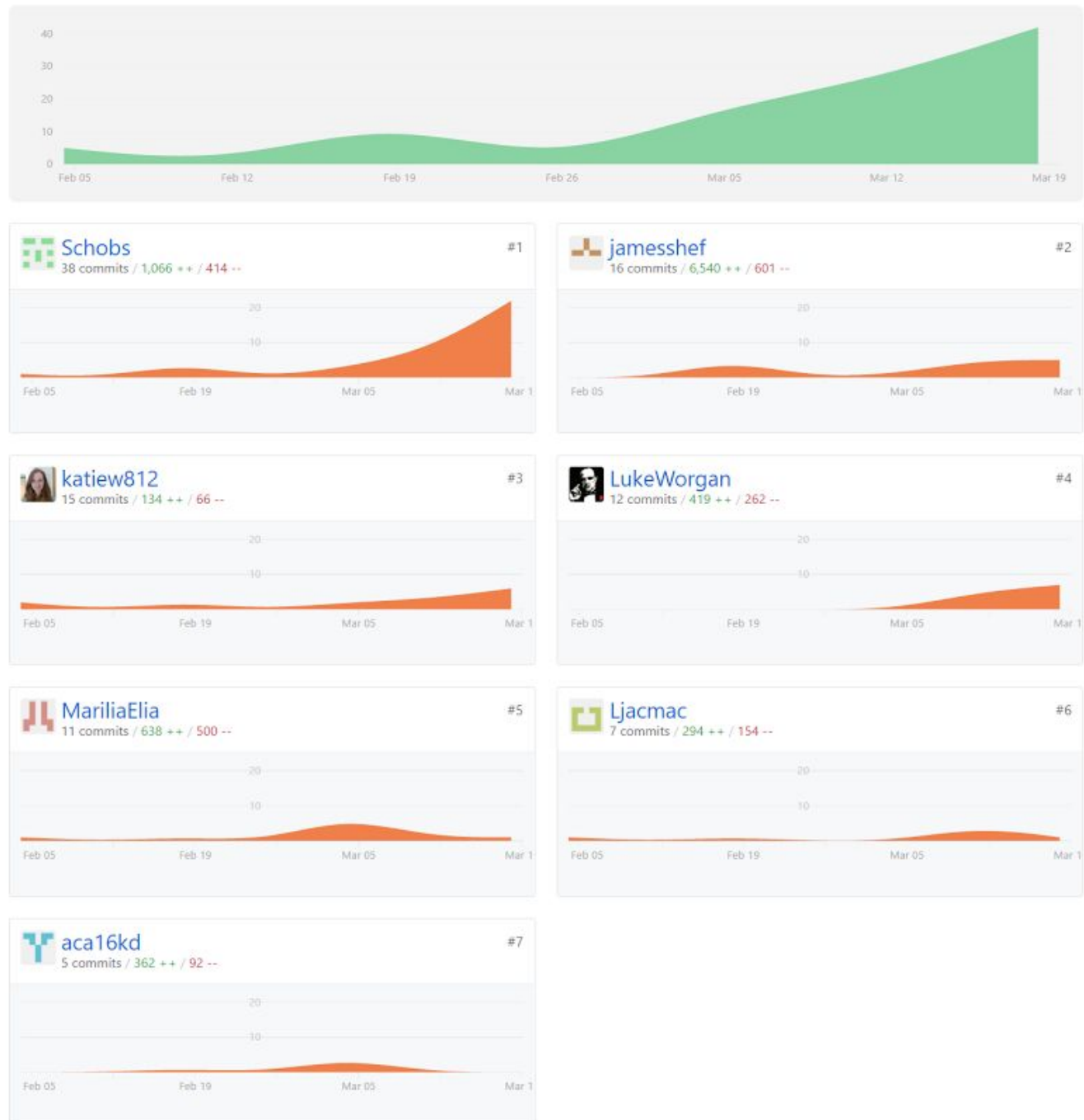
We weren't able to do this for the first iteration but we'll work on it for the final deadline.

Github commit graphs

Feb 5, 2017 – Mar 20, 2017

Contributions: **Commits** ▾

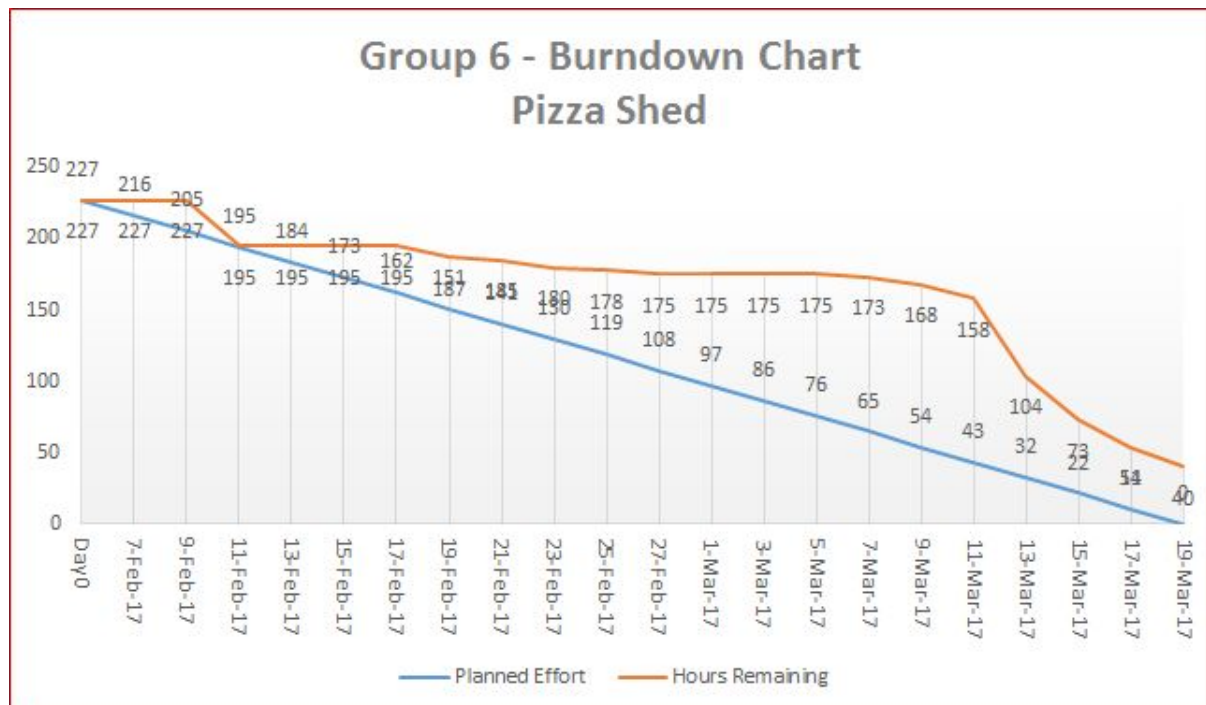
Contributions to master, excluding merge commits



Git graphs showing each team member's commits over time

Burndown chart

Collectively, we had 10 tasks that mean 227 efforts were estimated. Although the tasks have been deducted over time, we still have 4 tasks which are either entirely remaining or partially done.

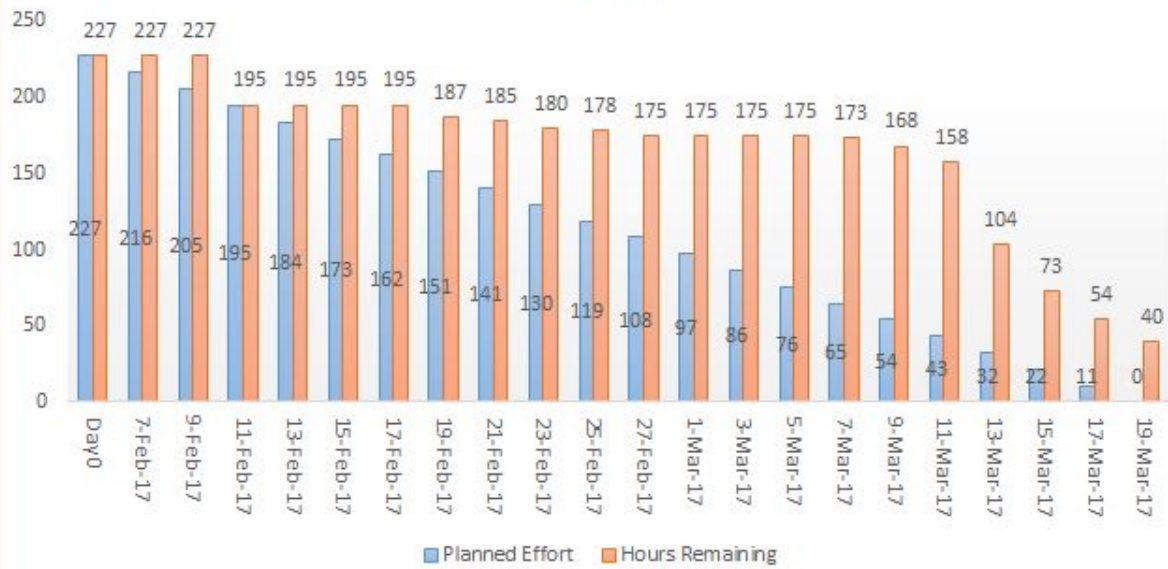


At the beginning of the iteration we were following the planned effort fairly well, however during the middle we did not do as much work as we should have. However, towards the end we pulled it back somewhat and managed to get most of the estimated effort completed.

Next iteration we need to split the work more evenly throughout the weeks to be more efficient.

Group 6 - Burndown Chart

Pizza Shed



Testing

For testing we will use a mixture of Cucumber, mini testing and manual tests. Mini testing isn't yet implemented. Cucumber is used to test the website automatically. We can program it to go to each page and test all elements on it. Eg. forms and buttons. Mini testing will be used to test the software of the system, ie. the ruby and any javascript we use. Manual testing is a simpler faster way to test everything we have missed so far, however it is not as efficient in the long run and so will be gradually changed to the automated tests.

Automated using Cucumber

Cucumber tests can be setup to automatically test many parts of a system. Currently there are test for navigation of the website, and login success and failure.

Navigation Test:

As you can see from the image it runs through a series of scenarios that attempt to navigate to each page of the website as of right now. They are green as they succeeded.

```
Feature: Navigation
Scenario: Navigate to the Homepage # features/nav.feature:3
  Given I am on the menu page # features/step_definitions/web_steps.rb:12
  When I follow "Home" within "nav" # features/step_definitions/web_steps.rb:26
  Then I should be on the homepage # features/step_definitions/web_steps.rb:188
Scenario: Navigate to Menu # features/nav.feature:8
  Given I am on the homepage # features/step_definitions/web_steps.rb:12
  When I follow "Menu" within "nav" # features/step_definitions/web_steps.rb:26
  Then I should be on the menu page # features/step_definitions/web_steps.rb:188
Scenario: Navigate to Twitter # features/nav.feature:13
  Given I am on the homepage # features/step_definitions/web_steps.rb:12
  When I follow "Our Twitter" within "nav" # features/step_definitions/web_steps.rb:26
  Then I should be on the twitter page # features/step_definitions/web_steps.rb:188
Scenario: Navigate to the Signup page # features/nav.feature:18
  Given I am on the homepage # features/step_definitions/web_steps.rb:12
  When I follow "Sign Up" within "nav" # features/step_definitions/web_steps.rb:26
  Then I should be on the signup page # features/step_definitions/web_steps.rb:188
Scenario: Navigate to the login page # features/nav.feature:23
  Given I am on the homepage # features/step_definitions/web_steps.rb:12
  When I follow "Log In" within "nav" # features/step_definitions/web_steps.rb:26
  Then I should be on the login page # features/step_definitions/web_steps.rb:188
```

```
Feature: Login
Scenario: Correct password entered # features/login
  Given I am on the login page # features/step_
  When I fill in "username" with "admin" # features/step_
  When I fill in "password" with "password" # features/step_
  When I press "Submit" # features/step_
  Then I should see "Welcome to the Admin Panel!" # features/step_
Scenario: Wrong password entered # fe
  Given I am on the login page # fe
  When I fill in "username" with "admin" # fe
  When I fill in "password" with "notpassword" # fe
  When I press "Submit" # fe
  Then I should see "Your username or password was incorrect" # fe
2 scenarios (2 passed)
10 steps (10 passed)
0m0.182s
Cucumber report generated for Cucumber Features by @hansluis@wetr
```

Login Test:

For this test it checks what happens upon a success and fail attempt at logging in. First it inputs a correct set of data into the fields and tries to log in, then checks if the website shows the right thing for a successful login. Then does the same but for a failed login.

Manual

These tests were based off of the acceptance criteria but with with added detail so that we could check everything worked perfectly. Green indicates that it works well and as expected, yellow indicates partially working or minor missing feature and red indicates that it does not work entirely or is not implemented.

Website Manual Testing:

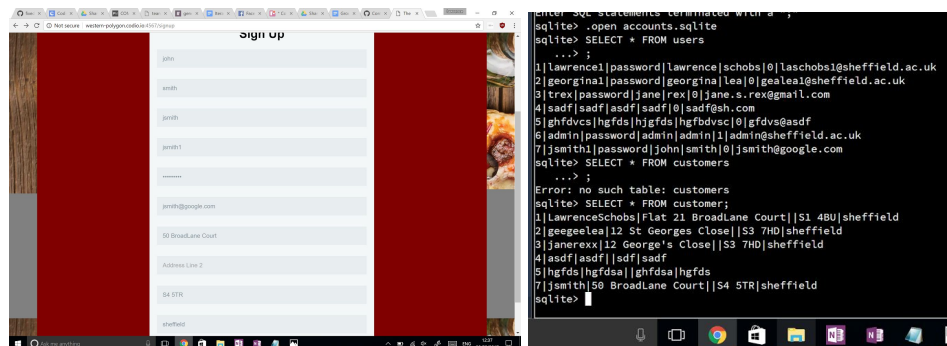
The website can be accessed by the twitter

Once the website's address is made concrete (as currently it changes each time it is run due to codio), it will be very easy to implement a link onto the twitter page to direct to the website's homepage

The Home, menu, log in and sign up pages are all accessible from each other
All the pages can be accessed from one another successfully.

A user can create an account

If the user creates an account with details that pass the validation tests such as valid email, no blank fields and with the city as Sheffield, the user can create an account. It is possible to then check the database and confirm they have correctly made it.



When creating an account, if the location is not set to Sheffield then the account cannot be created

If the city is left blank or set to anything but sheffield, then the user is not allowed to create the account and an error message will show that the user's city must be sheffield before they can make an account.

When signing up no important details can be left blank

It was set up that all the inputs on the form except for address line 2 are important and must be imputed before the form will be accepted by the website. This works successfully for all the sections of the form. However not all validations have been coded.

When signing up or logging in the user cannot inject inputs (eg HTML or sqlite)

You cannot sign up with sqlite commands or use scripts.

(above does not do anything)

A user can sign into the website

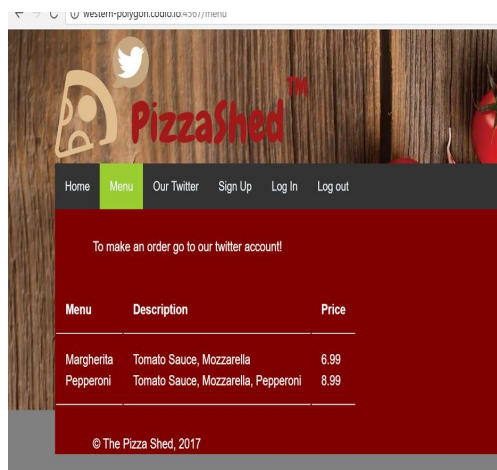
After creating an account with details username: Katie and password: walker, I was able to sign in using these details. **Once logging as a customer there is not yet any change to the website to indicate that you are signed in** but you are unable to log in again to prove that you are logged in. In the next iteration personal special offers will be shown once logged in.

When changing between pages, the user is not logged out until leaving the browser or pressing log out

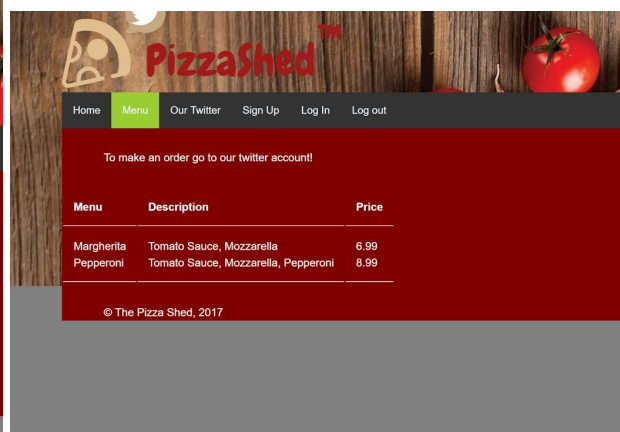
Once the user presses log out, you are then able to log in again that shows that the user has successfully been logged out from the previous session.

Both users and non users can access the menu

Whether or not the user is logged in does not affect the ability to navigate to and to see the menu.



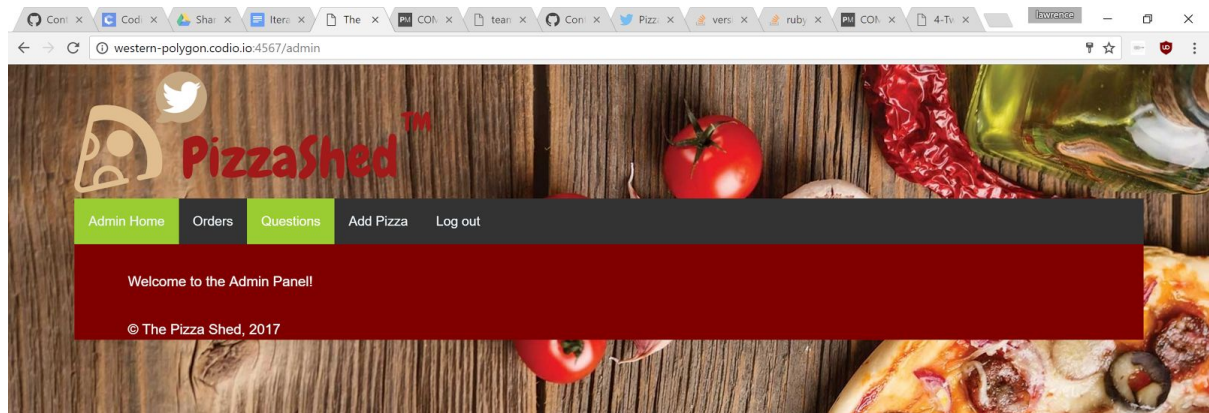
(logged in)



(not logged in)

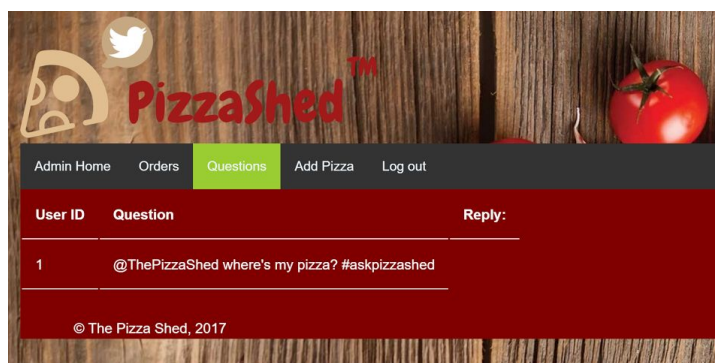
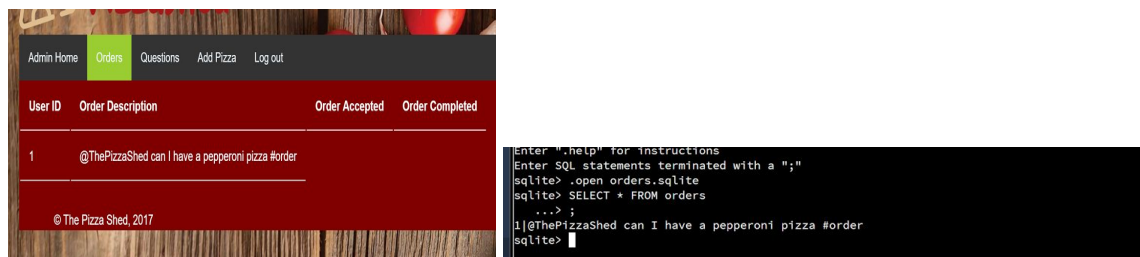
The main admin has access to all pages on the website

Once the admin has logged in, they receive a message saying “Welcome to the admin panel!” and can now see the orders, questions and add pizza pages which are specific to admins. To see the website as a customer, they must the log out.



The orders page shows a database of orders and questions

There are two separate pages for orders and questions but they both show the correct data. However I cannot accept the orders or mark them completed, not reply to questions.



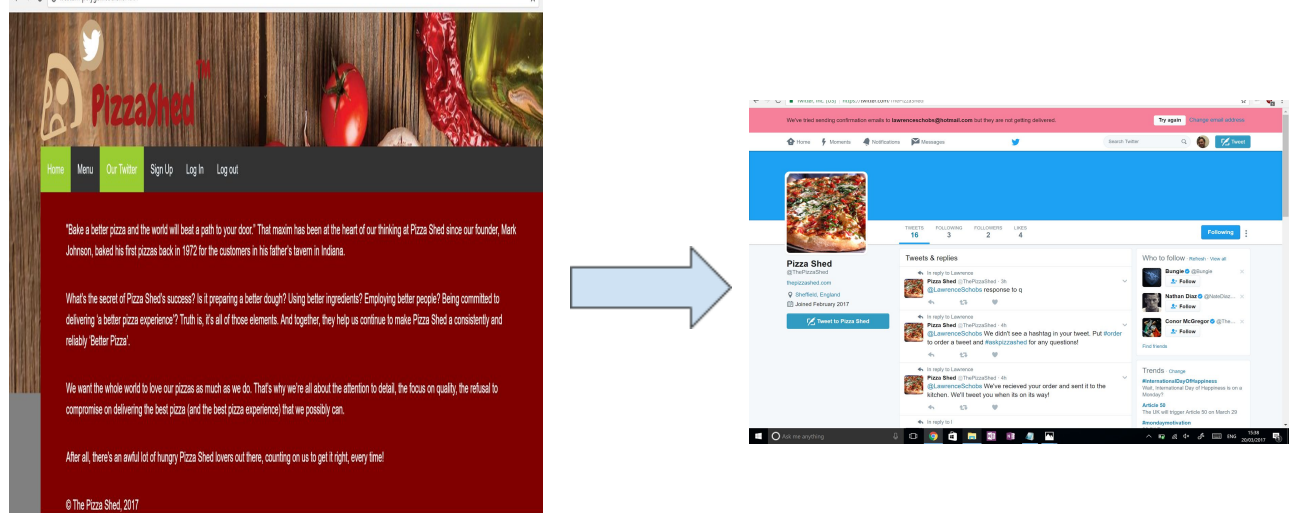
If there are no new orders, then “No orders will be shown”

Not yet implemented. When there are no orders or questions, the table is shown as empty.

Twitter:

The twitter page can be accessed by the website

When pressing the Twitter navigation button, the user is navigated to the ThePizzaShed twitter page.

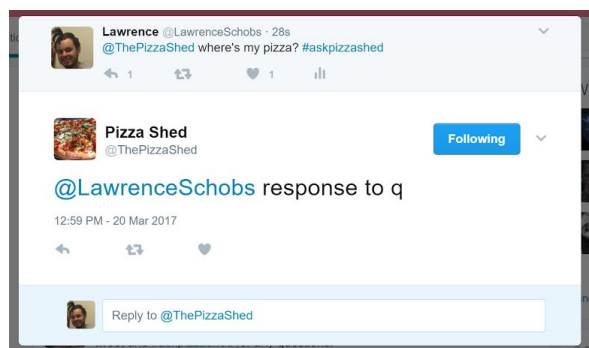


An order is placed on twitter and an automated reply is received

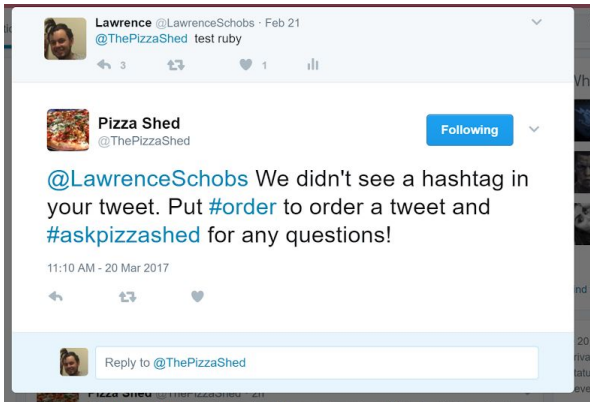
If I tweet with “#order” I will receive confirmation that my order has been sent to the kitchen.



If i tweet with “#askpizzashed” I will not recieve a reply unless someone manually responds to me



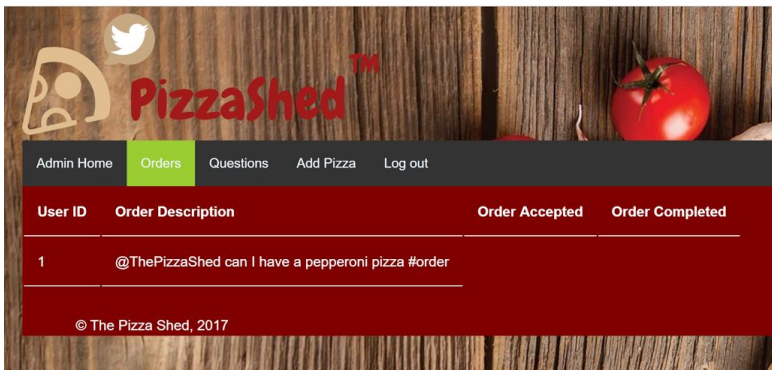
If i tweet without a hashtag I will receive a tweet telling me to resend using one.



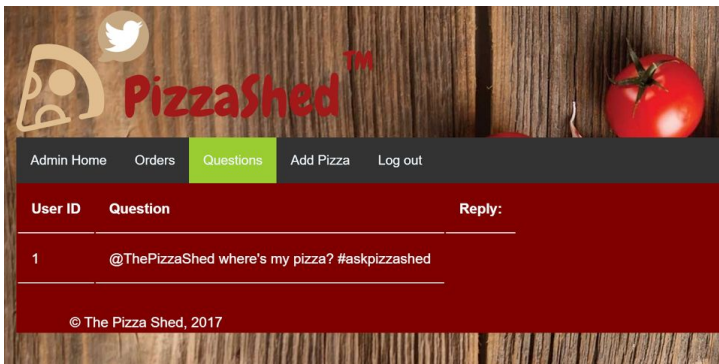
Databases:

Orders using the #order hashtag are placed in the orders database

The orders database printed is correct, clear and up to date as long as i run twitter.rb first



Questions/ non orders are placed in the questions database



The orders and questions are in the correct databases

```
sqlite> .open orders.sqlite
sqlite> SELECT * FROM orders
...> ;
1|@ThePizzaShed can I have a pepperoni pizza #order
sqlite> SELECT * FROM questions
...> ;
1|@ThePizzaShed where's my pizza? #askpizzashed
sqlite>
```

Within the accounts database is the admin, order admin, twitter admin, marketing admin and customers each with different levels of control on the site

Currently the only levels of account are the customer and admin. The admin accounts are the only ones able to change the menu and see the order page.