

COM1001 Introduction to Software Engineering

Spring Semester, Report for the first iteration

Weeks 1-7

Group 28

Team members:

- Jamie Huddlestone
- Tobias J Heywood
- Huiqiang Liu
- Ziting Zhao
- Kacper T Lodzinski
- Lee Norris
- Michal J Sekulski

1. Instructions on how to run and access the system:

Running the system:

After downloading the whole repository, there should be a file named '*app.rb*' that is not hidden in any folder. Open the console and move to the project's path and then run that file using the '*ruby app.rb*' command.

How to access the system's functions:

The main page/homepage allows the users to create an account or log in if they have created their account already. An account with admin privileges has already been set up with the following credentials:

Email: test@sheffield.ac.uk

Password: test

After logging in to the admin system, a list of actions should be displayed: "Incoming tweets", "Current orders", "Cars" and "Users". Clicking each of the phrases allows to see and partly manage the orders and the fleet. After choosing any of these features, an operator can access all of them again by pressing "Admin system" in top left corner.

In the "Incoming tweets" menu are a list of tweets that mention our Twitter account. From this list, hitting 'Expand' allows you to enter the relevant information for making a booking and simultaneously replying to a tweet. Note that the 'Date' and 'Time' fields are integers representing YYMMDD and HHMM!

Confirmed tweets enter the "Current orders" menu, where you can see the details of each order in a list; clicking on any of these details allows you to edit the content of that field, and any edits will be saved to the order on hitting the 'Update' button. The original tweet is shown under 'Expand', and in future iterations this will show the whole thread of conversation.

User can log out from the system by clicking "Log out" button and this moves the user to the homepage.

2. Resubmission of your stories, with changes annotated and highlighted:

In semester 2, we decided to have another look at user stories. We knew that changes can be made anytime and therefore we took the full advantage of that and improved them according to our vision. We reviewed them before making any changes and rearranged the user's stories, having in mind semester 1's feedback, changes made by client or changes made due to technical reasons. Some of the user stories were finished relatively quickly but some of them require a bit more time. We decided to defer couple of stories till the second iteration and some were finished partially. Generally, our stories did not require that many changes but sometimes it was essential.

Here is the new version of user stories:

0. As a customer I would like to order my taxi through Twitter. Estimated time: 1 Acceptance Criteria: Website properly working with Twitter environment	This user story is considered to be finished. It's essential and related to almost all user stories.
1.As a customer, I would like to order a taxi without a account. Estimated time: 3 Acceptance Criteria: User is required to have a Twitter account in order to order	This user story is going to be started after first iteration. Customer can order a taxi without an account but we encourage creating one.
2. As a customer, I would like to be able to tweet a street address when ordering a taxi so that drivers can pick me up from there. Estimated time: 1 Acceptance Criteria: recognising the street address	We are focusing on postcode, it's not necessary for now. We may add that feature in second iteration.

<p>3. As a customer, I would like to be able to tweet a postcode when ordering a taxi so that drivers can pick me up from there.</p> <p>Estimated time: 1</p> <p>Acceptance Criteria: recognising the postcode</p>	<p>This user story is considered to be finished. This is the basic function we should have done.</p>
<p>4. As an operator, I need to see the tweeted orders appear on the admin section of the website.</p> <p>Estimated time: 2</p> <p>Acceptance Criteria: Tweets must be displayed in order on the admin page</p>	<p>This user story is considered to be finished. By finishing this, operator is able to see all orders.</p>
<p>5. As an operator, I want to be able to reply to tweets as soon as I receive them, so customers know what's happening.</p> <p>Estimated time: 3</p> <p>Acceptance Criteria: reply box for new tweets</p>	<p>This user story is considered to be finished. Operator can decide to accept or reject the orders.</p>
<p>6. As an operator, I want to be able to see the availability of cars: waiting, on its way, with a fare, etc.</p> <p>Estimated time: 3</p> <p>Acceptance Criteria: monitoring system for cars and displaying it on the screen</p>	<p>This user story is progress. We have not decided if we want to include this user story card.</p>
<p>7. As an operator, I want to be able to change the availability of each car in the fleet using the system</p> <p>Estimated time: 2</p> <p>Acceptance Criteria: editing page for fleet cars</p>	<p>This user story is progress. The majority of car availability page/part will be developed by iteration two.</p>

<p>8.As an operator, I want to be able to easily add and remove cars from the fleet list.</p> <p>Estimated time: 2</p> <p>Acceptance Criteria:</p> <p>adding and removing button for fleet cars</p>	<p>This user story is progress. The majority of car availability page/part will be developed by iteration two.</p>
<p>9.As an operator, I need to know that I can get back into the system if I forget my password.</p> <p>Estimated time: 3</p> <p>Acceptance Criteria:</p> <p>forgot password button and forgotten password email system</p>	<p>This user story is going to be started after first iteration.</p>
<p>10.As a customer, I would like to be able to access rewards and special deals through my account on the website.</p> <p>Estimated time: 2</p> <p>Acceptance Criteria:</p> <p>rewards page on the website after logging in</p>	<p>This user story is going to be started after first iteration. We still do not know if we want to implement this feature.</p>
<p>11.As an operator, tweets need to be in conversation threads.</p> <p>Estimated time: 3</p> <p>Acceptance Criteria: group the tweets and replies in threads</p>	<p>This user story is in progress.</p>
<p>12.As a customer, I would like to have confirmation that my taxi has been cancelled when I send a cancellation tweet.</p> <p>Estimated time: 1</p> <p>Acceptance Criteria: send tweet with cancel confirmation</p>	<p>This user story is in progress.</p>

<p>13.As a customer, I would like to receive confirmation that a taxi is on its way, and how long it will take so I know when to be ready.</p> <p>Estimated time: 3</p> <p>Acceptance Criteria: send the tweet with taxi confirmation and with estimated time to the destination</p>	<p>This user story is going to be started after first iteration. We are not expecting any changes to this story card.</p>
<p>14.As a customer, I would like to be able to edit my personal details and choose what happens with my data through my account on the website.</p> <p>Estimated time: 3</p> <p>Acceptance Criteria: personal details page</p>	<p>Account management is planned for second iteration.</p>
<p>15.As a customer, I would like my account on the website to be linked to my Twitter handle.</p> <p>Estimated time: 2</p> <p>Acceptance Criteria: storing the twitter names on the website and linking them with accounts on the website</p>	<p>This user story is considered to be finished. This is a basic function so that twitter user can order a taxi on our website.</p>
<p>16.As a customer, I would like to be able to access my order history through the website.</p> <p>Estimated time: 3</p> <p>Acceptance Criteria: order history page</p>	<p>Vast majority of features that comes with this user story card have been finished and are fully working.</p>

<p>17.As an operator, I would like to have the ability to let the system automatically reply to some customer tweets.</p> <p>Estimated time: 5</p> <p>Acceptance Criteria:</p> <p>chatbot on Twitter for some specific tweets</p>	<p>We might not make a chatbot, but we will use the Twitter reply method to reply tweets automatically.</p>
<p>18.As an operator, I want to be able to set competitions in motion and for the system to manage them automatically.</p> <p>Estimated time: 8</p> <p>Acceptance Criteria:</p> <p>competition management system and page for setting competitions</p>	<p>This user story is going to be started immediately after first iteration.</p>
<p>19.As a customer, if possible I would like my phone to send my location when ordering a taxi so that drivers can pick me up from there.</p> <p>Estimated time: 5</p> <p>Acceptance Criteria:</p> <p>taking a location and adding it to the tweets</p>	<p>Change: Using just postcode but we might consider doing it later.</p>
<p>20.As a customer, I would like to be able to cancel a taxi order by tweet in the same way I ordered it.</p> <p>Estimated time: 2</p> <p>Acceptance Criteria:</p> <p>recognising cancellation tweets</p>	<p>This user story is going to be started after the first iteration.</p>

<p>21.As a customer, I would like to be able to update my location while a taxi is on its way.</p> <p>Estimated time: 2</p> <p>Acceptance Criteria: update location system</p>	<p>This user story is quite challenging, so we are not sure whether we can do this after first iteration.</p>
<p>22.As a customer, I would like to be able to see the different types of cars available (eg. no of seats) to order on the website, and to specify a type when ordering by tweet.</p> <p>Estimated time: 5</p> <p>Acceptance Criteria: displaying the table page with all cars, including type, size and other details.</p>	<p>This user story is considered to be finished. This is essential one which improves user's experience.</p>
<p>23.As a customer, I would like to be able to check the availability of taxis on the website before I order.</p> <p>Estimated time: 2</p> <p>Acceptance Criteria: page with taxi availability</p>	<p>This user story is considered to be finished.</p>
<p>24.As a customer, I would like to be able to specify my 'work' location so that I can easily order a taxi using that keyword instead of specifying an address.</p> <p>Estimated time: 2</p> <p>Acceptance Criteria: being able to add work location on personal details page on the website</p>	<p>This user story is going to be started after first iteration. We are not planning to add any changes.</p>

<p>25.As a customer, I would like to be able to specify my 'home' location so that I can easily order a taxi using that keyword instead of specifying an address.</p> <p>Estimated time: 2</p> <p>Acceptance Criteria: being able to add home location on personal details page on the website</p>	<p>This user story is going to be started after first iteration. Same as 'work' location.</p>
<p>26.As an operator, I need to quickly see a user's details, order history and status so I can interact with them appropriately.</p> <p>Estimated time: 3</p> <p>Acceptance Criteria: admin page with access to all user's details, order and being able to edit it</p>	<p>This user story is considered to be finished.</p>
<p>27.As an operator, I want to be able to easily edit user details/status, including account credit for rides or blocking them from the service.</p> <p>Estimated time: 3</p> <p>Acceptance Criteria: page for altering the user's details</p>	<p>This user story is going to be started after first iteration. We are not planning to make any changes to this story card.</p>
<p>28.As a customer, I would like to be able to get back into my account if I forget my password.</p> <p>Estimated time: 3</p> <p>Acceptance Criteria: forgot password email and button. Resetting password page</p>	<p>This user story is going to be started after first iteration. We are not planning to make any changes to this story card.</p>

<p>29.As an operator, I would like to see customer pickup points and destinations on a map.</p> <p>Estimated time: 8</p> <p>Acceptance Criteria: admin page with map and customers locations</p>	<p>We might not make this done, because we don't have a map yet.</p>
<p>30.As an operator, I need to know that the interactions I have with the system are secure, with an appropriate way of logging in and verifying my identity.</p> <p>Estimated time: 5</p> <p>Acceptance Criteria: special admin login page and setting password and admin name</p>	<p>This user story is considered to be finished. We are not planning to make any changes to this story card.</p>
<p>31.As a customer, I would like to be able to tweet the name of a local landmark when ordering a taxi so that drivers can pick me up from there.</p> <p>Estimated time: 3</p> <p>Acceptance Criteria: system recognise landmark name</p>	<p>Change: Using just postcode but we might consider doing it later.</p>
<p>32.As an operator, I would like to not have to deal with tweets that are clearly not feasible orders, so I can concentrate my time and effort on viable customers.</p> <p>Estimated time: 5</p> <p>Acceptance Criteria: system to filter unreal tweets and enquiries</p>	<p>This user story is considered to be finished. We are not planning to change this story.</p>

<p>33.As a customer, I need the website to be accessible, intuitive and quick to navigate so that I can order a taxi without too much hassle.</p> <p>Estimated time: 3</p> <p>Acceptance Criteria: get everywhere on the website within 3 clicks</p>	<p>This user story is considered to be finished.</p>
<p>34.As a customer, I would like to be able to tweet a what3words code when ordering a taxi so that drivers can pick me up from there.</p> <p>Estimated time: 5</p> <p>Acceptance Criteria: what3words code system</p>	<p>Change:Using just postcode but we might consider doing it later.</p>

3. The stories you planned to tackle in this iteration:

Our aim in the first iteration was to do around half of the user stories, creating a good base for the project which we can then add to. We have left out little features which we plan to add to the website which were not needed, but would be added to improve the quality of our system, such as being able to request a taxi from a landmark. Below we have listed all the story cards we planned to tackle in the first iteration along with who worked on each one.

As an operator, I want to be able to reply to tweets as soon as I receive them, so customers know what's happening. - Jamie created the ruby function, connected it to the view and database.

As a customer, I need the website to be accessible, intuitive and quick to navigate so that I can order a taxi without too much hassle. Jamie created the front-end of the website; Kacper created the login and register function, connected it to the view and user details table.

As a customer, I would like to receive confirmation that a taxi is on its way, and how long it will take so I know when to be ready. - we have the confirmation done, Kacper and Jamie did it.

As a customer, I would like to be able to see the different types of cars available (eg. no of seats) to order on the website, and to specify a type when ordering by tweet. - Lee did database side and created sql statements for viewing all cars in the database of a certain type and viewing cars with a minimum number of seats. Kacper created the table for cars and help Ziting with fetching the cars function. Ziting created the page and ruby functions.

As an operator, I want to be able to see the availability of cars: waiting, on its way, with a fare, etc. Kacper created the table. Lee - created an sql function for viewing cars with a chosen status

As a customer, I would like to be able to tweet a postcode when ordering a taxi so that drivers can pick me up from there. - Michal has initialized that and then Jamie updated and improved this feature.

As an operator, I want to be able to easily add and remove cars from the fleet list. -Lee created sql functions for these. Ziting created the page and ruby functions.

As an operator, I want to be able to change the availability of each car in the fleet using the system - Lee created an sql statement for changing the availability of a chosen car. Ziting created the page and ruby functions.

As an operator, I need to quickly see a user's details, order history and status so I can interact with them appropriately. - Kacper created a table and Lee created an sql statement for viewing a user's details and all orders with a chosen userID; Jamie created the orders view and related controllers.

As an operator, I want to be able to easily edit user details/status, including account credit for rides or blocking them from the service. - Lee created sql functions for these

As a customer, I would like to be able to access my order history through the website. Lee created an sql function for this

As an operator, I need to see the tweeted orders appear on the admin section of the website.-Huiqiang and Michal created ruby function to fetch new tweets. Jamie created the fetch orders, update and delete functions and connected them to the views and database.

As a customer, I would like my account on the website to be linked to my Twitter handle. Kacper made the register page, Lee and Kacper designed the database schema so a twitter handle is stored with a user's details.

As a customer, I would like to be able to check the availability of taxis on the website before I order. - Ziting created the page and ruby functions. Lee created sql functions for these

Toby created the navigation bar and integrated the create order function.

As an operator, I need to know that the interactions I have with the system are secure, with an appropriate way of logging in and verifying my identity.

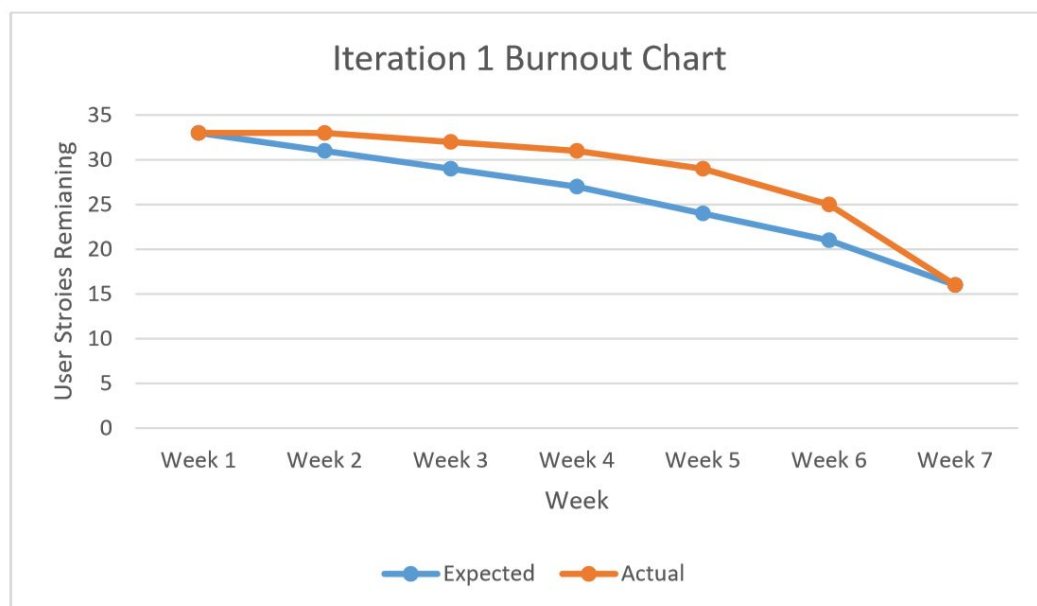
Kacper created the register and login page, connected to the database and view as well as added the SHA256 password encryption.

As an operator, I would like to not have to deal with tweets that are clearly not feasible orders, so I can concentrate my time and effort on viable customers.

Kacper and Jamie created archive and delete function for orders.

4. Your burndown chart, showing progress over all of your stories over the iteration:

User Stories	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7
Expected	33	31	29	27	24	21	16
Actual	33	33	32	31	29	25	16



5. Testing and test coverage:

So far, we have been using miniTest tests to verify correctness of our files. Another method - Cucumber(with capybara) has not been working properly throughout the project as codio seems to use an outdated Ruby version. We managed to obtain a newer version which turned out to be unstable. Having that in mind, we will definitely include Cucumber tests by the next iteration but for now we agreed to use manual tests instead, as can be seen below. We include the screenshots of the code as well as the console's output:

Testing with MiniTest

Test_method.rb :

```
def create_order(text, time, carID)
  return 0 if
  text = text.strip
  time = time.strip.to_i
  carID = carID.strip.to_i
  ary = [text, time, carID]
end
```

miniTesting.rb :

```
require 'minitest/autorun'
require_relative 'test_method'

class TestStringComparison < Minitest::Test
  def test_order
    assert_equal 0, create_order("abc", 123, 1234)
  end
end
```

Console after running miniTesting.rb :

```
Run options: --seed 16593

# Running:

...

Finished in 0.000915s, 3278.6277 runs/s, 3278.6277 assertions/s.

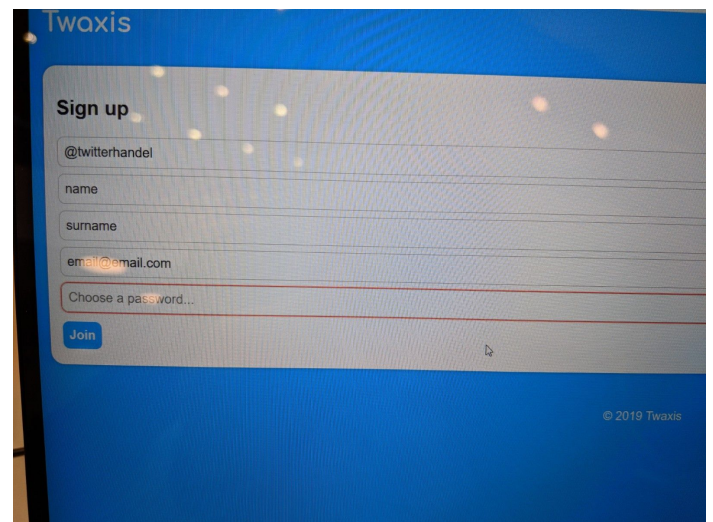
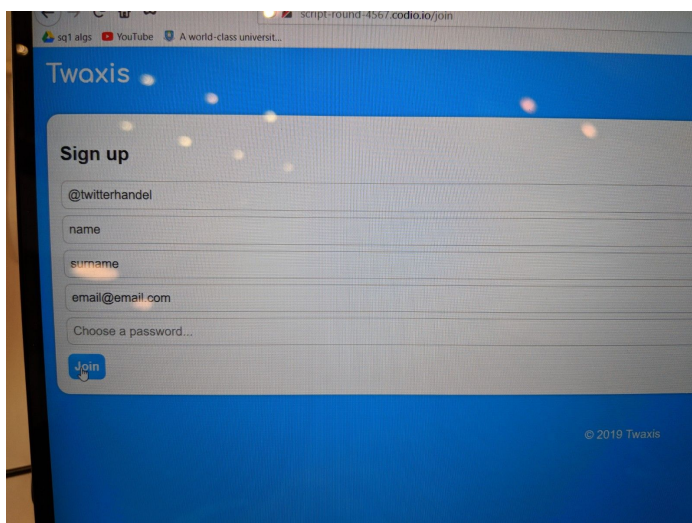
3 runs, 3 assertions, 0 failures, 0 errors, 0 skips
codio@delta-impress:~/workspace/project/controllers$
```

Manual tests

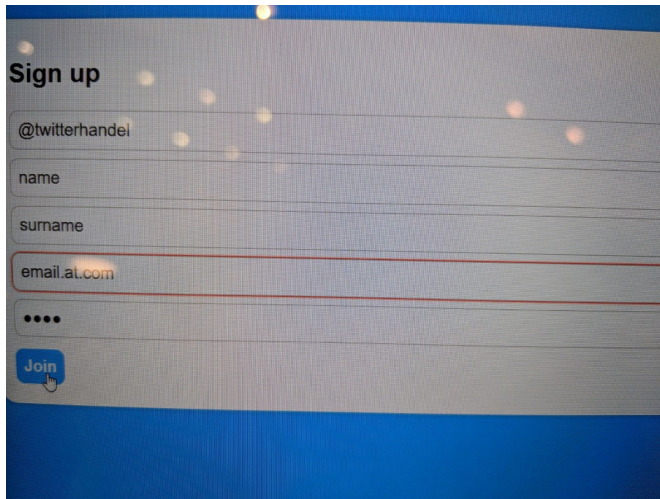
We tested how does the system responds to different user input in different scenarios. Some of the results can be seen below:

Signing in without providing a password --->

The system responds correctly

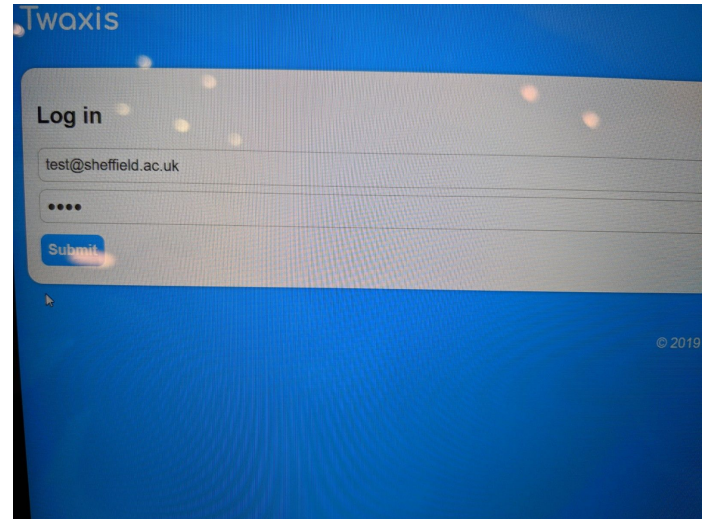


Other spaces work correctly as well



A screenshot of a web browser showing the Twaxis sign-up page. The page has a blue header with the Twaxis logo. Below the header, there is a 'Sign up' section with several input fields: a text field containing '@twitterhandel', a text field for 'name', a text field for 'surname', and a text field for 'email' containing 'email.at.com'. There is a password field with four dots. At the bottom of the form is a blue 'Join' button.

Filling the admin's credentials



A screenshot of a web browser showing the Twaxis login page. The page has a blue header with the Twaxis logo. Below the header, there is a 'Log in' section with two input fields: a text field containing 'test@sheffield.ac.uk' and a password field with four dots. There is a blue 'Submit' button. In the bottom right corner, there is a small copyright notice: '© 2019'.

Order page fully working

