

Deliverable 2

By THCH

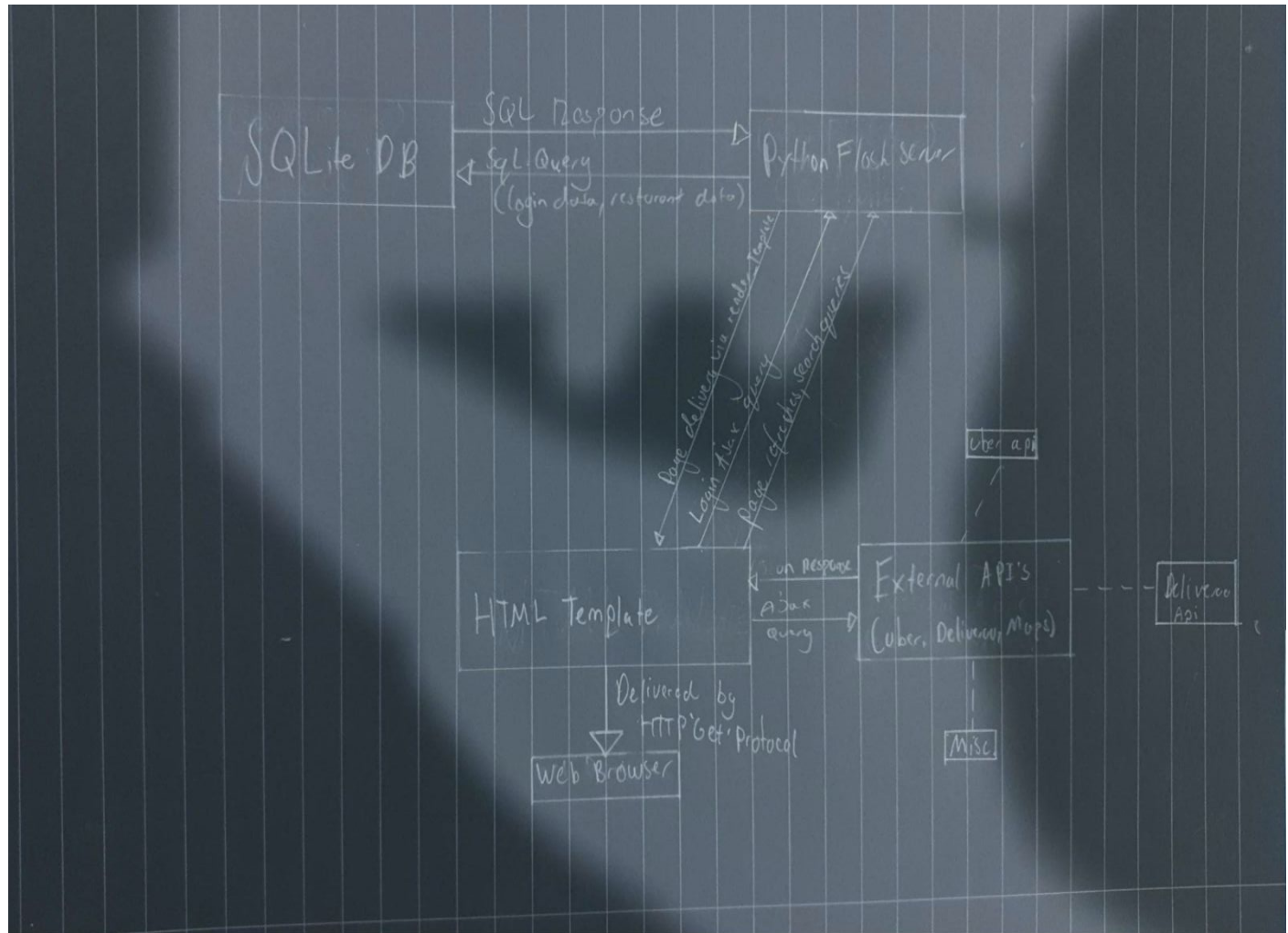
Part 1: Software Architecture

Given the current user stories of what the team has decided to do, what would be the resulting software architecture? The software architecture elements are:

1. What external data sources will your system be accessing?

1. Google Maps
2. Bing Translate
3. Wine (Winestein REST API by winewinewine.com)
4. Deliveroo
5. UberEats
6. Currency conversion ([exchangeratesapi](#) by European central bank)
7. Splashable - backgrounds
8. Food recipe API (Recipe - Food - Nutrition by spoonable/edamam)
9. Search API

2. *Software components: the selected Web stack showing major software components that comprise your solution. These will include both components that need to be developed and third-party components (e.g. web browser).*



In the case of this diagram, Sqlite DB acts as the model, Python Flask Server is the controller and the HTML template is the view. The external Apis will interface directly with the HTML template through javascript ajax scripts as is required to provide a clear and fair comparison between the different gig economy apps.

3. *Relating choices to components: decide which language should be used for which component of the software architecture.*

HTML/View: HTML5/CSS3 have both been used for the UI as they are universally excepted by web browsers on both desktop and mobile. This allows for a large array of users to access the platform and

have the same homogenous experience. With HTML 5 being selected as it allows for complex UIs without the need for proprietary applications such as flash to be installed. JS has been selected as well due to its inbuilt functionality with HTML webpage through its DOM manipulation library. Hence, enabling us to have both asynchronous loading of information and stimulus allowing for a more dynamic webpage to better provide functionality for our users.

Controller/Middleware: Python was selected as it has frameworks which allow for a middleware and server database manager to be written in the same language. It also automates a lot of threading and security vulnerabilities. Python also has a very good json library with similar dictionary/object structures which allow for easy interaction between the front end and middleware. Python also having a very effective sql library installed in python 3.7 by default, allows for a lot more flexibility with SQL queries and more condensed database functions. Overall, these reasons make python perfect for a fast and safe prototype as many memory, security and allocation issues are handled by the included frameworks.

Backend: A SQLite database will be used in the backend to handle data and user authentication with a python database interface being implemented to improve the consistency of the database interactions across the rest of the middleware. SQLite allows for memory only databases free of the client server architecture of postgresql providing a more easily debuggable program and faster load times. It also meets the sql standard making it very easy to move between both a dedicated server environment and local storage. SQLite also does not have unnecessary overheads such as custom types and assertions reducing the overall computational overhead of the system. With SQLite also being easily integrated with python functions it allows for a more manageable link between the database and the python middleware.

4. The choice of an implementation/technology or framework: teams must decide now on a suitable language or combination of languages to implement a prototype. This will be largely determined by the Web stack but at the same time you can make variations.

We selected programming languages that we had experience with, which were popular for use in web development and deployment. These languages' popularity ensures that support is available, and that security flaws will be discovered and patched more quickly. As our team members have experience with them, we will need to spend less time learning them and we can spend more time working on our project.

The Flask API eliminates the need for a dedicated server and server operating system such as Apache/nginx, with Flask instead allowing for the handling of user requests through the middleware, reducing setup time and improving portability.

Using Python as middleware resulted in our selection of SQLite for our DBMS, as it has a built in Python library and is very flexible in its deployable environment. Javascript and its associated frameworks in conjunction with HTML5 and CSS3 were selected due to the wide range of the libraries available for the construction of interactive, responsive and attractive UIs, and that they are some of the most ubiquitous web platforms available, ensuring that users will be able to access our web site on their device's browser. We are using the most up-to-date standards available due to their current support on browsers.

We elected not to use a JS transpiled language, such as Typescript, due to our inexperience with them, and the additional resource strain they put on clients.

5. The choice of a platform: decide on machine or machines requirements (Linux, Windows etc.) for the final system.

As a webapp is being developed all OS platforms will meet the requirements for the implementation of the final system. The website will be designed for primary use with a laptop/desktop computer but mobile devices will still be able to interact with the site in its full capacity.

6. Make a summary of the key benefits/achievements of your architectural choices

Architecture point	Choices	Benefits/Achievements
Frontend	HTML5/CSS3 Javascript	<ul style="list-style-type: none">- HTML 5 is the modern standard for interactive web apps, as Flash has been discontinued. It is fast, widely supported, and functional on Mobile.- JS will improve the user experience and allow for a more interactive, tailored web experience.- JS is extremely widespread, allowing nearly all devices with a web browser to run our web site.- Not operating system dependent, increasing the amount of users.
Middleware	Python3 Flask	<ul style="list-style-type: none">- Flask API reduces the need for a dedicated server to handle user requests.- Allows for the majority of the application to be contained within a single deployable improving debugging and portability.- Allows pre-rendering which reduces the amount of reloads/queries from the user reducing the overall server overhead and hence the need for server resources.- Python 3 is more modern than Python 2, and is the only version still receiving support. It is now common to learn only Python 3.
Backend	SQLite database with Python interface	<ul style="list-style-type: none">- Avoid using a client server architecture improving simplicity and allowing for in memory databases for fast development and debugging.- A python interface also reduces the associated amount of programming required and improves homogeneity over the system design.

Part 2: Initial Software design

1. The updated list of stories or use cases.

Overall these features and scenarios come together and show the complete functionality of a food delivery comparison website with a few extras features.

Name: Compare food delivery services

AS A user

SO THAT I have the best experience when ordering food.

I WANT TO order food in the cheapest and best manner, which means that I can compare and find the cheapest delivery price, compare to other currencies if I am used to a different currency as well as reveal alternative and healthier options, for example cooking at home.

Name: Compare prices

AS A price sensitive user

SO THAT I can save money on my food orders

I WANT TO check the prices of different food delivery services

Scenario: Ordering dinner

GIVEN I am on the homepage
WHEN I type in a search query
THEN I should be shown local restaurants matching my query in at least 10
WHEN I select a restaurant
THEN I should be shown the restaurant's page, where I can see relevant pricing information
WHEN I select a service
THEN I should be taken to that service's website where I can place my order

Name: Compare currencies

AS AN international user
SO THAT I can realise how much I am actually spending
I WANT TO convert the prices to my usual currency

Scenario: Calculating how much an order would be in a different currency

GIVEN I am on the page of a food item
WHEN I select the price of the food item
THEN I should be shown the value in at least usd, aud and yuan different currencies in a modal form

Name: Compare recipes

AS A health conscious user
SO THAT I can become healthier by being encouraged to cook the same food at home
I WANT TO compare the food i am ordering to an actual recipe for it

Scenario: Finding out what may be inside a food item

GIVEN I am on the restaurant's page
WHEN I select a food item
THEN I should be taken to a details page about the food item
WHEN I select "Find Equivalent Recipe" button
THEN I should be taken to a page with different recipe websites that have a recipe for the item
WHEN I select a recipe website
THEN I should be taken to that website

Name: Scroll meals

AS A user

SO THAT I know what is available to order

I WANT TO see many different foods in a categorized and simple to navigate system

Scenario: Finding food I would like to order

GIVEN I am on the homepage

WHEN I type in a food item in the search bar

THEN I should be taken to at least 2 pages showing matching searches

WHEN I click on one of the items

THEN I should be shown the restaurants that have it

WHEN I click on one of the restaurants

THEN I should be taken to the restaurant's page

Name: Ordering in - app

AS A user

SO THAT it saves me time and is convenient not to re-find the order

I WANT TO place my order with the different food delivery apps without going to the other apps

Scenario: Placing an order I have already assembled

GIVEN I am on the restaurant's page

AND GIVEN I have already created an order

WHEN I press the "Place Order" button

THEN I should be sent to the relevant delivery service, with the order already placed so I can order it

Name: Sorting the food by categories

AS A user

SO THAT I can find exactly what food i am feeling like

I WANT TO choose either to sort the food im looking at by cuisine, distance or popular

Scenario: Finding the specific food I like

GIVEN I am on the homepage

WHEN I type in a search query in the search bar

THEN I am taken to a page with search results for that category of food

Name: Personalise by location

AS A user

SO THAT I can find restaurants that will deliver to me and be cheaper

I WANT TO use my location to see what restaurants are nearby

Scenario: Checking restaurants close to me

GIVEN I am on the homepage

AND GIVEN I have allowed the site to access to

WHEN I type in a food item in the search bar

THEN I should be taken to a page showing matches

WHEN I click on one of the items

THEN I should be shown only the restaurants that will deliver to me on at least one service

Name: Recommend by previous

AS A user

SO THAT if I am not completely sure of what I want to eat I can find a range of foods to order

I WANT TO search not just by food names, but also by cuisines

Scenario: Finding restaurants that i would like

GIVEN I am on the homepage

AND GIVEN I am logged in

WHEN I scroll down the homepage

THEN I should be shown at least 4 restaurants based upon my previous orders.

WHEN Clicked on i should be taken to the relevant restaurants page.

Name: See delivery excess

AS A user

SO THAT I can fully compare which company is charging me more so I can find which company is the cheapest

I WANT TO see how much each company costs for the delivery and excess

Scenario: Seeing how much I have to pay additional to my order

GIVEN I am on the restaurant's page

THEN I should be shown a comparison of the delivery excess for different services alongside their menu item prices

Name: Checking reviews

AS A user

SO THAT I can see if the food is good

I WANT TO see how other people rate the food

Scenario: Finding a good restaurant

GIVEN I am on the restaurant's page

WHEN I select the Yelp button

THEN I should be taken to corresponding Yelp review, where I can read the reviews for the restaurant

Name: Pair wines

AS A user

SO THAT I can enjoy a complementing drink with my food

I WANT TO see which wines pair nicely to my meal

Scenario: Find a wine pairing

GIVEN I am selecting my meal

WHEN I select the wine button

THEN I should be taken to a list of wines which would pair with the meal selected

WHEN I click on a wine

THEN I should be taken to another web page so that I can purchase the wine.

Name: Search favourites

AS A user

SO THAT it saves me time and is convenient not to re-find the order

I WANT TO be able to favourite different meals/restaurants for easy access

Scenario: Favourite restaurants

GIVEN I am on the restaurant's page

WHEN I press the heart emoji next to a restaurant

THEN I should be able to find that restaurant in the favourites menu

Scenario: Favourite foods

GIVEN I am on the restaurant's page

WHEN I click a heart emoji next to a meal

THEN I should be able to find that meal in the favourites menu

Name: Find deals

AS A user

SO THAT i can find the best deal which may be cheaper than usually cheaper other apps

I WANT TO be able to see which restaurants/delivery apps have deals on at the time

Scenario: Find a deal

GIVEN I am on the restaurant's page

WHEN I look at the page

THEN I should clearly be able to see which meals/restaurants/delivery apps have deals on at the current time

Overview of non-traditional customer stories

Name: Review competitors

AS A food delivery service

SO THAT I can change my business strategies to improve my company

I WANT TO check out the competition

Scenario: Comparing my service

GIVEN I am selecting meals that I offer

WHEN I select a meal

THEN I should see the prices that my competitors are offering at the current time

Name: Market restaurant

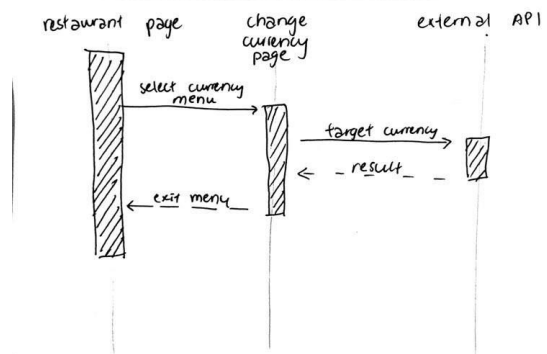
AS A restaurant owner

SO THAT I can come up with better deals for those customers or try and diversify my current market

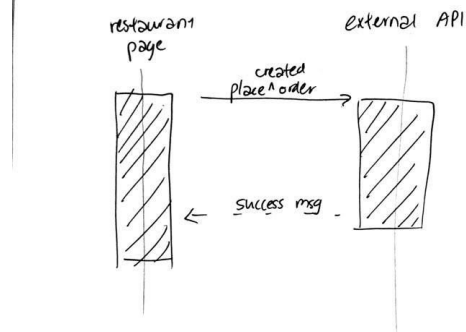
I WANT TO be able to see which applications are giving my customers the best deals

2. Then include one sequence or interaction diagram for each use case. You can use UML sequence diagram definition: http://en.wikipedia.org/wiki/Sequence_diagram. Each box in a sequence diagram should correspond to a component in your architecture.

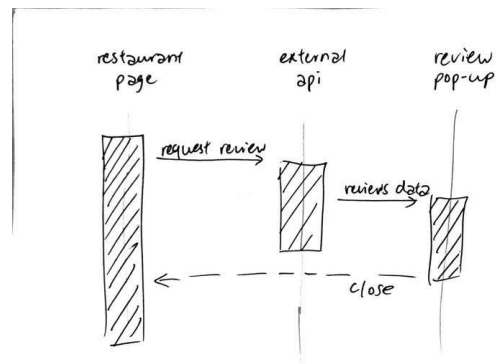
Change currency to foreign currency.



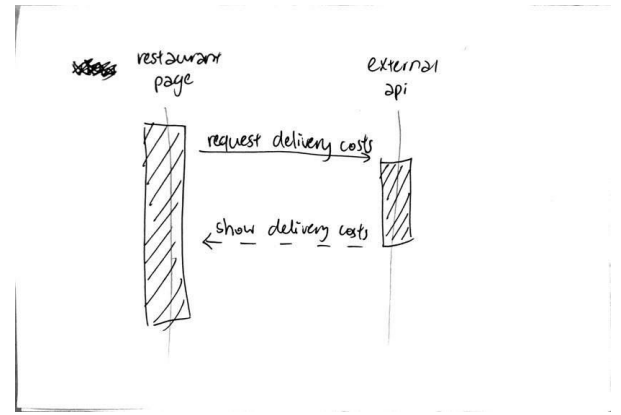
Place order (given existing order)



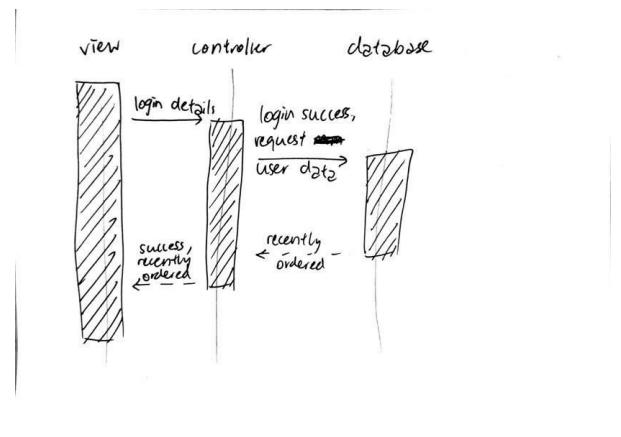
Find good restaurants (see reviews)



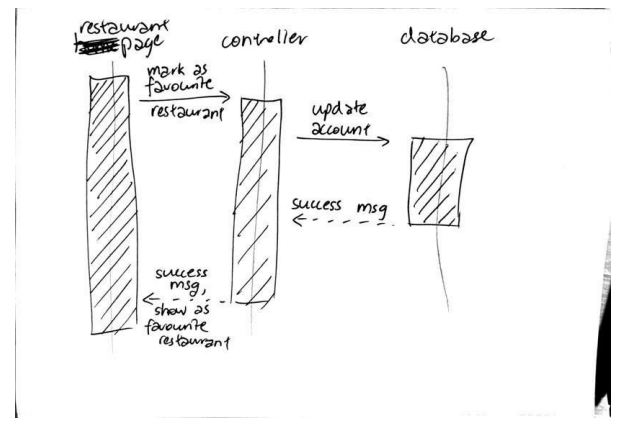
Show additional costs (delivery costs)



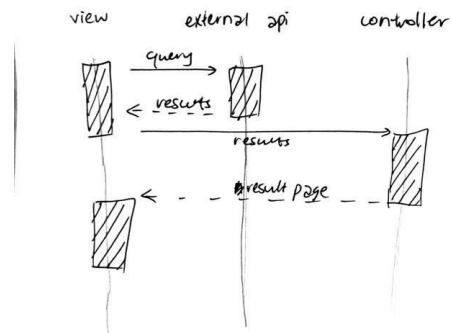
Show recently ordered



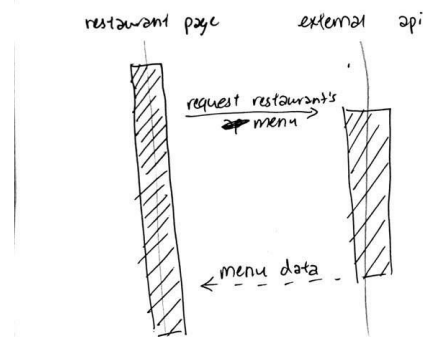
Favourite a restaurant



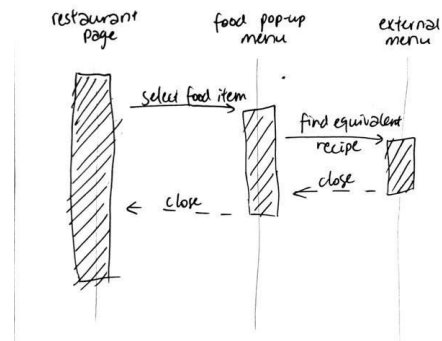
Find food (find specific food matching search query)



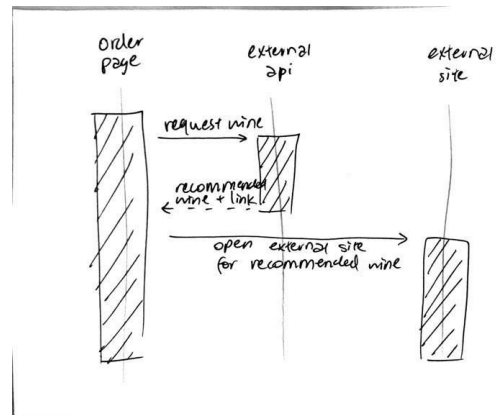
Find a deal
Compare services



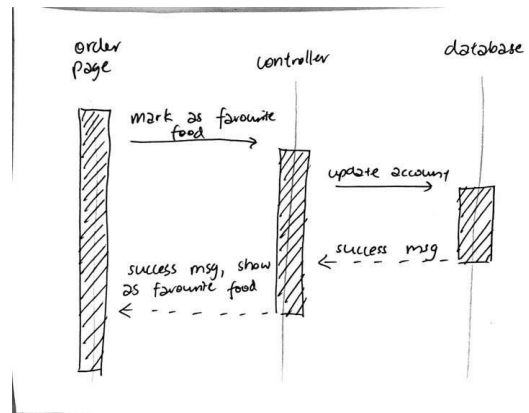
Find out what is in the food (show similar recipe)



Recommend a wine pairing



Favourite a food



Make an order

