SENG2021 FINAL REPORT

Alva and the Boys

21/10/16

Contributors: Alva Huynh, Matthew Yu, Charley Wong, Andrew Au, Edwin Choi, Chris Miles

System Scope and Specification

Project Theme

The theme of this project is to display all purchasable properties in a given location. The data will be displayed onto a map and will provide a visual interface to assist home buyers in deciding what property they may purchase.
Due to some limitations and issues of permission from certain API providers, the scope of this project will be limited to properties only found within the UK.

Deliverables:

- Ability to search for purchasable properties in a region
- View housing information (e.g. price, location, images)
- Contact with property dealers
- Listings of purchasable properties based on:
    - Price range
    - Keywords
    - Total time for travel from a point of interest

Limits and Exclusions

- The project will only cover properties in the UK
- The system will not act as an agent to buying home property
- The system will not offer any real estate advice

System Functions

Obtain locations from a search field and display housing properties in the nearby vicinity on a map.
This is the main function of the desired system. The function will retrieve data from a selected real estate API and determine the list of purchasable properties within a selected location.
Used in conjunction with another API, these properties will be displayed on a map in the form of map markers. In addition, users are also able to move the displayed map, zoom in or zoom out. As an example, a user may input and search for any address and then have a cluster of markers displayed onto a map to indicate their locations surrounding the address inputted.

Obtain information about a property based on a search
This function is paired with the function to display housing properties and provides details on the actual listing of a property. The function will retrieve data from a selected real estate API and display purchasable properties onto a map as map markers. Upon the selection of a map marker, data containing information on selected housing is retrieved from a real estate API is retrieved and these details are displayed to a user. Details of the selected house includes price, land, bathrooms, bedrooms and contact information for more information and details. As an example, if the client browses through the properties shown on map and finds interests in one property, it is possible to select the property they found interest in and will provide further data given by the real estate API and a link to the real estate webpage available for purchase.

Filter housing properties based on some specified criteria
This is a complementary feature to the search function. This function provides more flexibility and efficiency for the system by filtering search results. This function supports multiple filters such as price and keywords. Filter options presented appear in the form of a checkbox, a search field or a drop down menu. An example of this function is the desire to specify a minimum and maximum price. To do this, the filters for the Price must be filled in with text. When these text boxes are filled out, the system will again filter out search results according to the applied criteria.

Retrieve list of properties by time, location and travel method
System can create a list of purchasable properties within a specific time range by some travel method. This function will take a location, time and travel method. This function will then return a list of purchasable properties to be displayed. An example of this function would be when the client wants to find a property near a specific location (e.g. workplace) they can enter the address and designated time to get there within the surrounding, and all the markers within this area will show, which can get to the location within the time limit placed.

View all purchasable properties as a list
System can retrieve images of all purchasable properties within an area and display them as a series of paginated images on a panel. An example would be if the client decides to prioritise the view of the property than the actual location then selecting the picture view shows each house’s display and on interest the client can select one of them to see where they are located.

Use Cases

1. Use Case Name	Display map of purchasable properties
2. Brief Description	Display a map of markers showing purchasable properties in a certain location
3. Actors	
3.1. Primary Actor/s	User searching for a house in a location
4. Flow of Events	
4.1. Main Flow	
4.1.1.	DISPLAY HOME PAGE OF WEBSITE Use case begins when home page is displayed to the user
4.1.2.	SELECT “FIND MY HOUSE” BUTTON The page redirects to a map with a search address bar overlaid on top of the map on the upper left
4.1.3.	DISPLAY MAP WITH MARKERS SHOWING PURCHASABLE PROPERTIES Given that a valid address is placed into the search bar and user presses enter
4.2. Alternate Flows	
4.2.1.	DISPLAY HOME PAGE OF WEBSITE Use case begins when home page is displayed to the user
4.2.2.	SELECT “MAP” MENU ITEM ON THE UPPER RIGHT The page redirects to a map with a search address bar overlaid on top of the map on the upper left
4.2.3.	DISPLAY MAP WITH MARKERS SHOWING PURCHASABLE PROPERTIES Given that a valid address is placed into the search bar and user presses enter

5. Special Requirements	
5.1. Business Rules	N/A
5.2. Usability Requirements	Use case requires internet connection and web browser or application installed. Address is required to be a valid one.
6. Pre Conditions	Address entered into search field is valid

7. Post Conditions	Map is updated with markers indicating location of houses that are up for purchase

1. Use Case Name	Display information of a housing property
2. Brief Description	View information linked to a marker that provides details on the house at that location
3. Actors	
3.1. Primary Actor/s	User searching for a house in a location
4. Flow of Events	
4.1. Main Flow	
4.1.1.	MAP SHOWS MARKERS WITH PURCHASABLE PROPERTIES Use case begins when/after a user has placed a searched for nearby properties with a valid address
4.1.2.	USER SELECTS MAP MARKER A small popout appears showing an image of the home and some information about the house.
4.2. Alternate Flows	
4.2.1.	USER SELECTS MAP MARKER Alternate use case that redirects user to a more detailed view
4.2.2.	SELECTS “CLICK FOR DETAILS” User is redirected to a more detailed page

5. Special Requirements	
5.1. Business Rules	N/A
5.2. Usability Requirements	Use case requires internet connection and web browser or application installed.
6. Pre Conditions	System has successfully displayed a list of purchasable properties near a certain address
7. Post Conditions	System returns information on the house (such as price, bedrooms, agent details)

1. Use Case Name	Display map of purchasable properties fulfilling certain criteria
2. Brief Description	Using the side menu containing search filters, display a map of purchasable properties
3. Actors	
3.1. Primary Actor/s	User searching for a house in a location
4. Flow of Events	
4.1. Main Flow	
4.1.1.	USER CLICKS BUTTON TO DISPLAY SIDE MENU The user clicks the button to display the side menu where several filters are displayed. These filters include: checkboxes for specifying housing facilities/type, 2 text boxes to indicate price range and 2 text boxes to compare distances between two properties.
4.1.2.	USER TICKS/SELECTS ONE OF 8 CHECKBOXES AVAILABLE The user is provided with 8 different filters being: house, cottage, studio, apartment, balcony, fireplace, gym and lift. Upon selection, only housing with the specified criteria are displayed on the map. Multiple selections are possible.
4.2. Alternate Flows - Price Range Filter	
4.2.1.	USER ENTERS NUMBERS FOR PRICE RANGE Use case begins from 4.1.1 (see above). User types in a number in the min or max text box (or both)
4.2.2.	MAP UPDATES User is shown results which satisfy the indicated price range
4.3 Alternate Flows - Distance Filter	
4.3.1	USER ENTERS ANOTHER ADDRESS TO COMPARE DISTANCES Use case begins from 4.1.1 (see above). User enters an address into the first text field under Compare Distance (field is for destination)
4.3.2	USER ENTERS A SPECIFIED TIME FOR TRAVEL User enters a number specifying amount of time from the location specified above (4.3.1)
4.3.3	USER SELECTS A TRAVEL TYPE From a dropdown menu, user selects a travel type (either DRIVE, TRANSIT or WALK)
4.3.4	USER SELECTS CALCULATE User selects calculate to find properties within distance and display them as map markers on the map

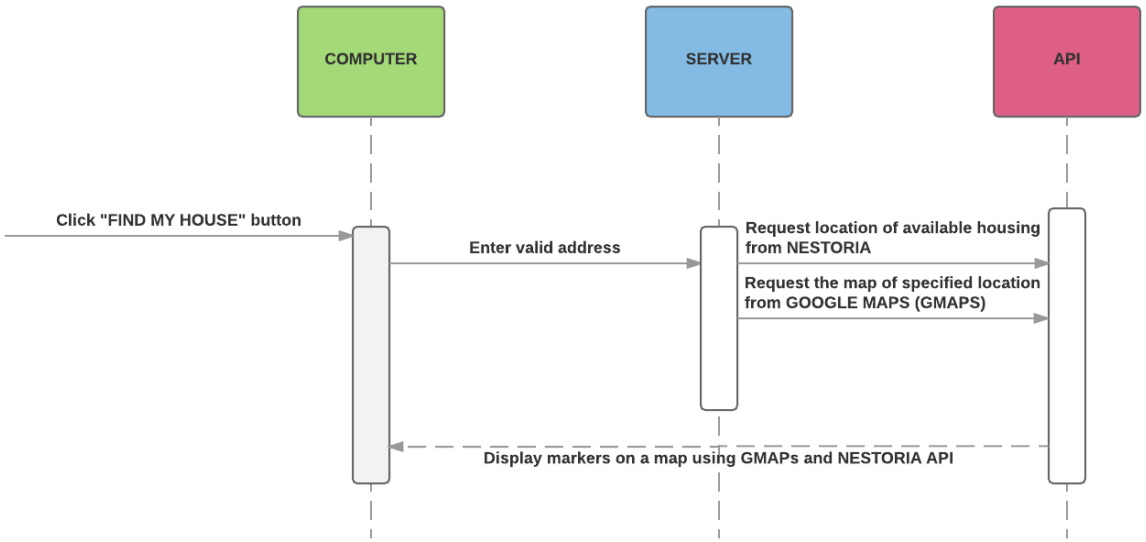
5. Special Requirements	
5.1. Business Rules	N/A
5.2. Usability Requirements	Use case requires internet connection and web browser or application installed.
6. Pre Conditions	Internet connection is required,
7. Post Conditions	System displays available housing with the specific filters selected.

1. Use Case Name	Use Picture View mode to display houses and find information
2. Brief Description	Use the picture view mode to look through the list of purchasable properties.
3. Actors	
3.1. Primary Actor/s	User searching for a house in a location
4. Flow of Events	
4.1. Main Flow	
4.1.1.	USER SELECTS PICTURE VIEW MODE User selects the home view mode button in the upper right hand corner
4.1.2.	A PANEL SLIDES OUT FROM THE TOP User is able to shift through a number of house property in the form of the respective property's image.
4.1.3.	USER SELECTS IMAGE OF PROPERTY IN HOME VIEW MODE Map focus is moved to an expanded view of the map marker representing the property selected

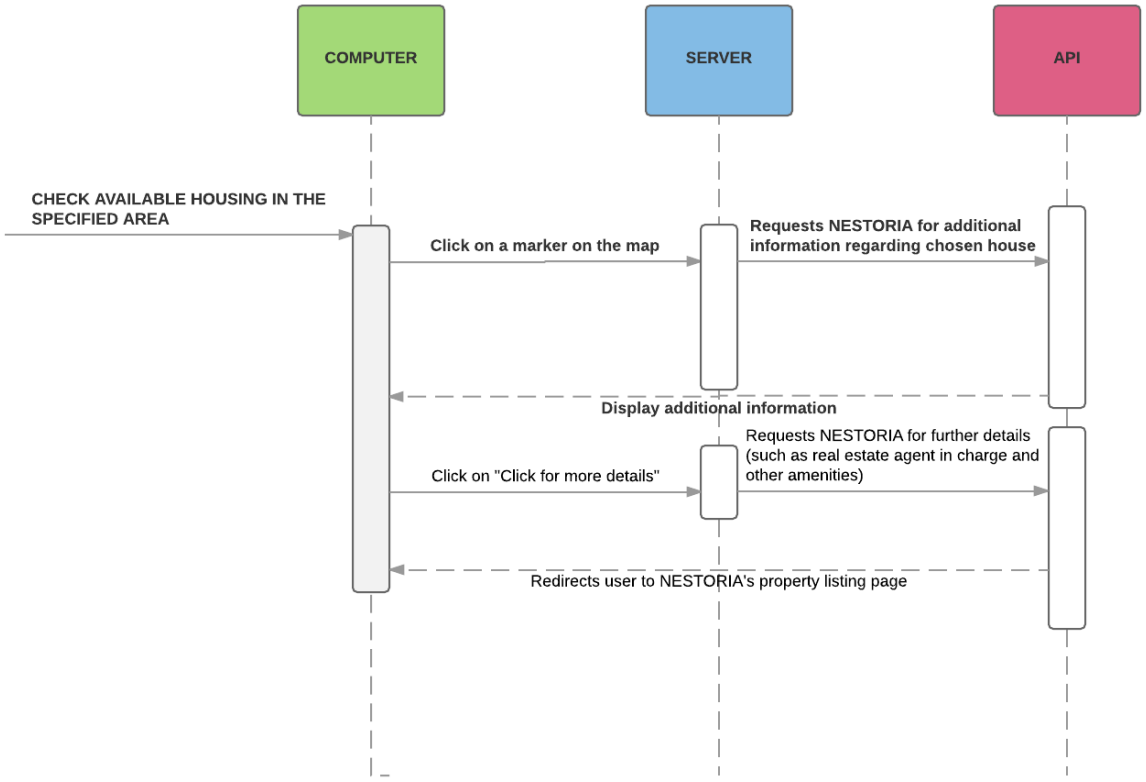
5. Special Requirements	
5.1. Business Rules	N/A
5.2. Usability Requirements	Use case requires internet connection and web browser or application installed.
6. Pre Conditions	User has conducted a valid search previously with at least one result
7. Post Conditions	User is able to search through list of purchasable properties in home view mode

Sequence Diagrams

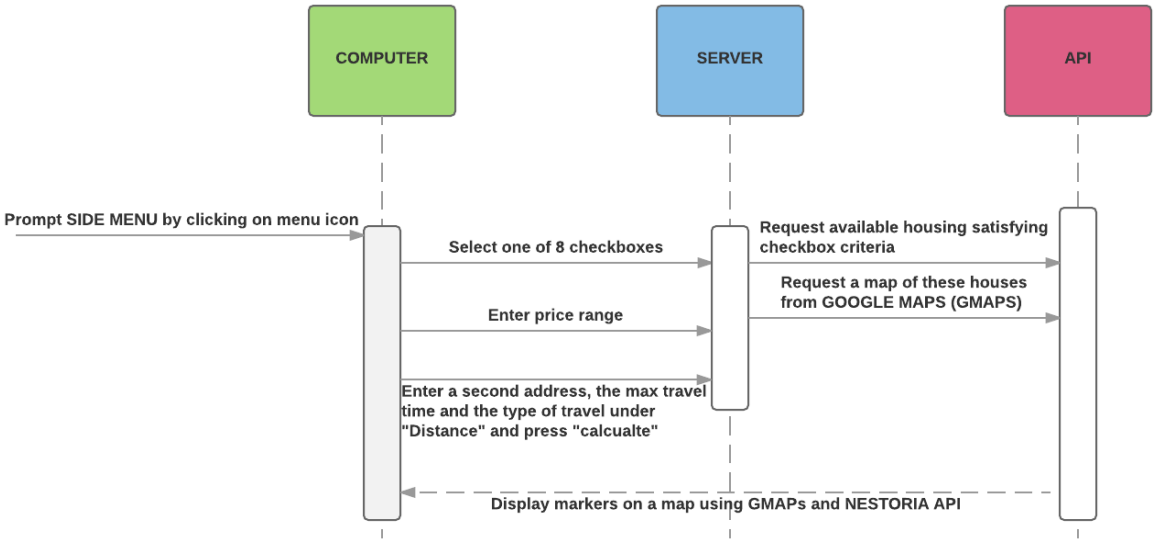
User case: DISPLAY HOMEPAGE



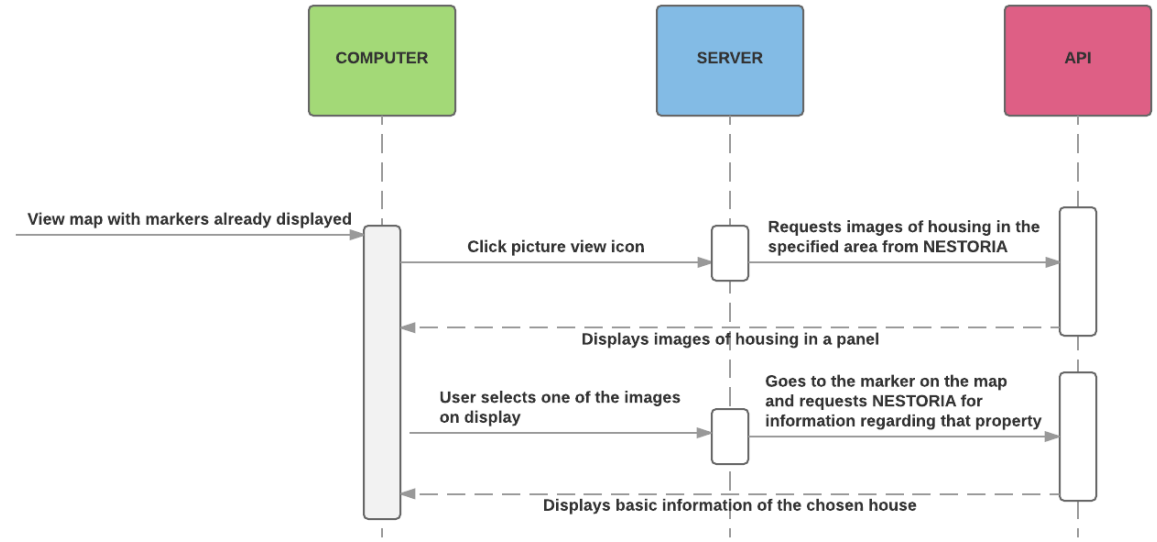
User case: Display information for a selected property



User case: Filters



User Case: select picture view



Software Architecture

Software Components Developed

Real estate information compiler

This component handles the retrieval of real estate information from online sources through the internet, and compiles the data into a universal format that our system will be able to use. This component will heavily make use of the Nestoria API, which will directly gather property information for the UK. This information will then be processed into a common profile (address and location) that can be used by all our additional components.

Data transformation and processing

This component handles the amount of real estate information being displayed through to the user on the mapping interface. The component depends on the processed input of real estate information through our compiler component, as well as requests through our user input requests component which determines the level of data returned. The amount of data transformation and refinement will depend on the search complexity that the user provides through the input requests component. This component does not connect to any API's.

Input requests component

This will handle location searches input and request and will pass requests of the location of property on sale to the data transform and processing for it to process and then be outputted to the mapping interface, based on an entered house number, street name and state. The filters, being an additional feature also within the input request component, will handle the property on sale being outputted depending on keywords, price range, size of land, radius of search and distance to landmarks. This component does not connect to any API's.

Mapping interface

The mapping interface is essentially how the software will be received and used. This component will accept real estate data and display a visual representation of the data by placing markers on a map. To do so, the mapping interface makes use of the Google Maps API developed by Google in order to simplify and streamline the process. Users will be able to view a visual roadmap of the area with location markers to indicate properties on sale - this information will be taken in through the data transformation and processing component. The mapping interface also contains a search bar for location entry (through addresses), which will connect directly to the Google Maps API to utilise location search.

Third party components

Web browser

The web browser acts as the boundary between our software and the user - it displays the mapping interface, and allows for the user to input specific searches and filters that will allow for the location of specific real estate property. Primarily displaying the maps interface component, user navigation takes place using their mouse to drag the map around, as well as click on markers (which represent real estate property on sale) to view additional information (address). Furthermore, the user can enter an address on the search bar to find property on sale around a particular address or region. Our software supports all commonly used browsers, being Google Chrome, Safari and FireFox.

Server - Django inbuilt

In the development process, we used the inbuilt Django development server located on a personal computer. This server acted as the basis for keeping our website running and handled all user requests while in development. These requests will either be through location search, or filters. As such, it is directly connected to two major components we are developing - being the data transformation component as well as the map interface. These requests will be handled by the server in real time when changes to search fields or filter checkboxes are filled in. Location searches will control the location of property on sale being outputted to the mapping interface, based on an entered house number, street name and state. The filters, being an additional feature, will control the property on sale being outputted depending on price range, size of land, radius of search and distance to landmarks.

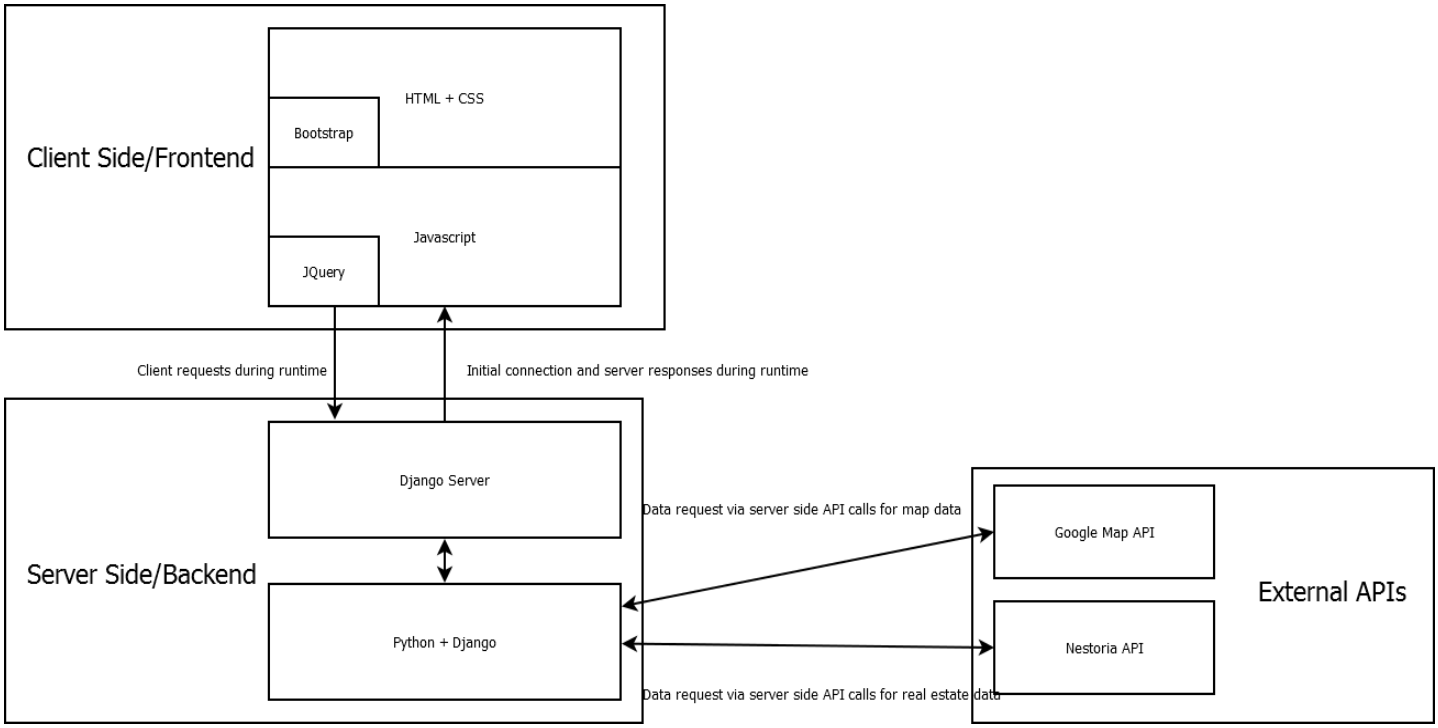
Digital Ocean - Cloud Computing Service

Post development, the web application runs from Digital Ocean, which will store, manage and process data. This occurs through a network of remote servers. Much like the Django inbuilt server, Digital Ocean acts as the basis for keeping the web application running and managing and processing data. The major difference is that in Digital Ocean, the server is now accessible from the internet, rather than only accessible locally on a personal computer.

Architecture

Overall, the front end of our system will use Javascript, which will contain the Mapping interface and Request Handler components. The back end of our system will use Python 3, which will contain the Real estate information compiler and data transformation and processing components.
The Nestoria API will connect directly into our backend real estate information compiler component, which will directly feed in raw real estate data in real time for compilation in our software. The Google Maps API will directly connect into the frontend maps interface component, to visually display the processed information to the user.
Our Django inbuilt server is responsible for the uptime of our web based software, as well as provide the link between user requests and the map interface components.

Relationship between components



Deployment

The web browser component will be client side, and hence will be deployed on the user's machine.
All other four components (real estate information compiler, data transformation and processing, requests handler and map interface) will be server side, and hence will be deployed through our server machine. The server software that will handle this is our chosen Django inbuilt server software.

Implementation consideration

Choice of implementation – technology and frameworks

Technology	Justification of Choice	Alternative
HTML	The common markup language for developing webpages and web applications. Mandatory for developing webpages and web applications.	There are no other alternatives for HTML.
CSS	File used to format and style appearance of HTML files.	There were no other alternatives considered for styling HTML files.
Bootstrap Framework	Front-end framework for designing websites and web applications. Selected for its aesthetic, features across devices and library of scripts. Also chosen due to its community support and popularity for most used front end framework for designing web applications.	Multiple frameworks could have been considered. Some popular frameworks such as Foundation and the new Semantic UI were considered but rejected in favor of Bootstrap for its community support and its popularity.
Javascript	De-facto language used for client side interactions. Also chosen for its excellent community support for developers.	There are no alternatives to Javascript for client side programming at this time for our purpose.
jQuery	Optimised library of Javascript. Used for its simplicity and efficiency. Also complementary to some visual effects with the Bootstrap Framework.	Angular JS is a Javascript based framework for developing front end web applications. Angular JS could have been used over Javascript/jQuery. However due to lack of experience with Javascript, the team decided to use vanilla Javascript. jQuery was also chosen due to its integration with Bootstrap.
Python3	Python3 (CPython) was chosen for its powerful features, readability and community support for developers. Also chosen as some members of the team are well acquainted with the features of Python3.	Python2 (CPython) has shown to be a stable and well supported implementation of Python, especially now it has had 9 major releases with 2.8. Members of the team believed that Python3 was better in terms of speed and modern assets, whilst also being very stable. Java is a powerful language. However, members believed that Python3 was much more readable and would be simpler to code in. Members also found that it could be a worthwhile learning opportunity. Ruby was also considered to be a possible candidate. However, the team lacked experience in languages of Perl roots, and so we opted for languages with syntax similar to C/C++, i.e. Python.

Django	Web frameworks based on Python. Used to manage project and development of web application. Chosen as it is known to be quite diverse and has plenty of community support. Also a very popular framework that is deployed by many other websites (Bitbucket, Instagram, others)	Flask was an option due to its apparent simplicity. It is a framework also implemented in and for Python. However, the team felt that the strong community support and more widespread use of Django would benefit the project and the team members in terms of experience.
Google Maps API	API provided by Google Maps. Provides core functions for the system. API used for navigation of maps, display of markers, public transport and time to travel. Chosen as it is widely used and reliable.	Alternatives include Microsoft Bing Maps. Google Maps was preferred for its popularity and community support.
Nestoria API	Free API providing real estate information. Nestoria covers a fairly large number of property in different countries. Chosen as it was free without restrictions on use. Unfortunately, Nestoria is not available for Australia.	Real Estate API was the initial choice team members decided with as it had access to Australian property. However, permission was not granted to use the API.

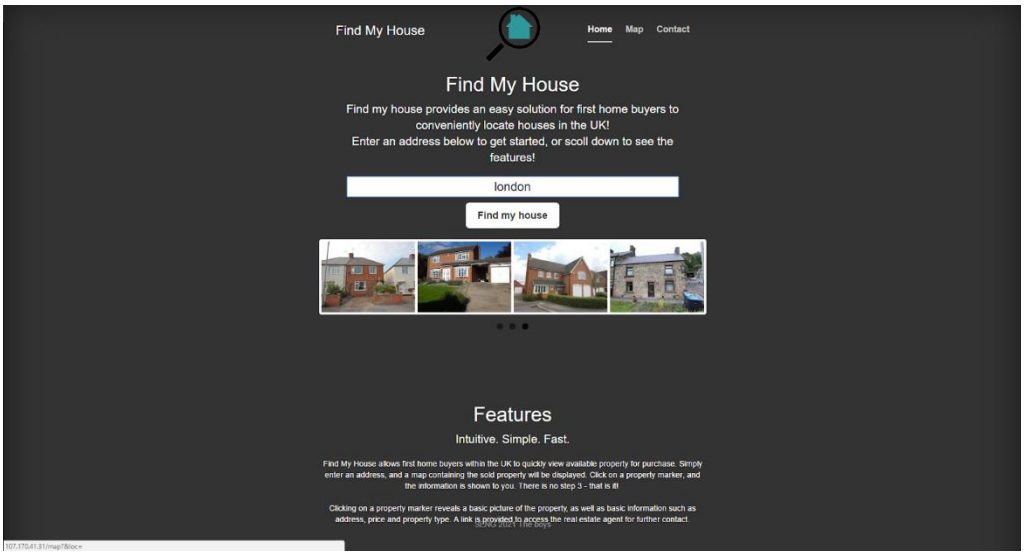
Hosting of web application

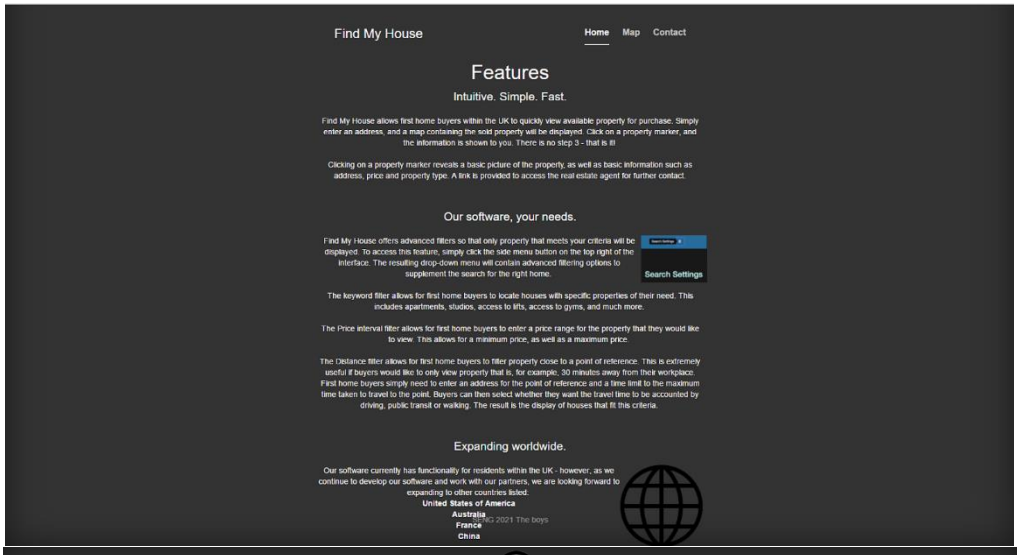
Hosting	Justification of Choice	Alternative
WSGI with Django inbuilt server	The chosen method is to use the WSGI settings for its support for python language and the use of our chosen framework(Django)'s inbuilt provided server, this was chosen mainly due to the time constraint and the ease of setup of the server without having to configure all the static files within the project.	Apache HTTP server can be used for hosting Python based web applications under Apache. Has support for Python 2 and 3. MOD_WSGI was chosen for its simplicity and its use with Python. Apache is also widely used. However due to the setup requirements for the Apache servers and static files, the original inbuilt server by django was used. It is however, interchangeable.
Digital Ocean	A managed cloud hosting infrastructure. Used as it was available on student's discount. Also used because it was simple and robust to use.	Local Machine Alternatively, the web application could have been run on the local machine. However, as digital ocean was available and in comparison to a local machine, Digital Ocean provides much better and more consistent performance,hence Digital Ocean was used.

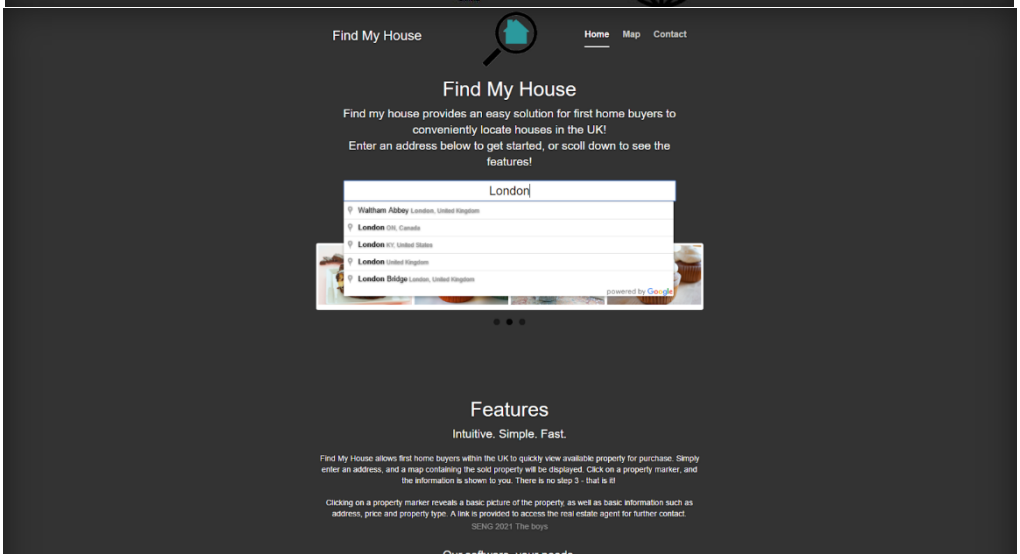
Platform

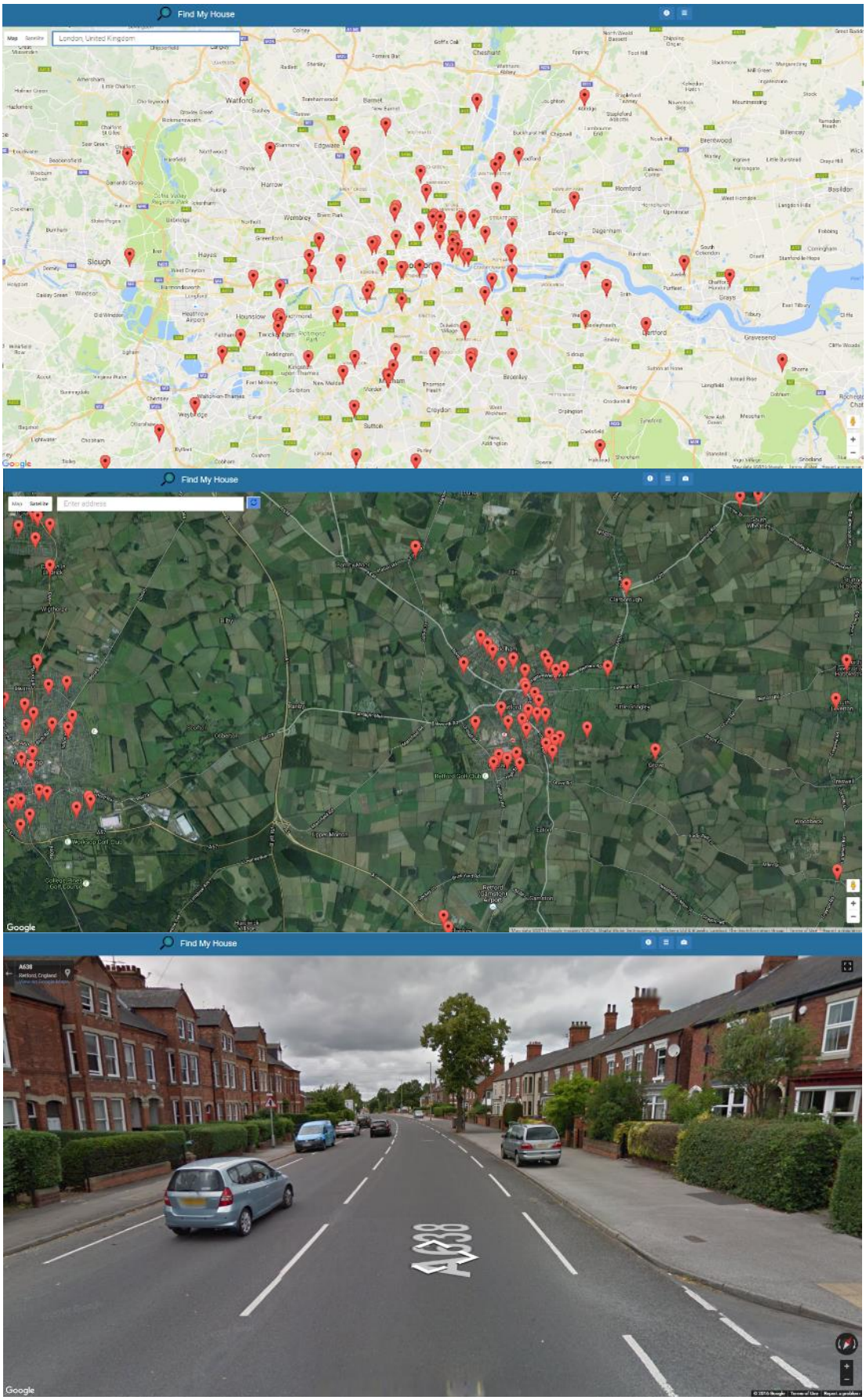
The platform intended is any machine that Python3 supports. We would primarily target Debian Linux, due to ease of maintenance, cost effectiveness, large user base for community support and demonstrated stability. On the client side, the platforms intended are machines that support web browser. However, the interface is designed with desktops and laptops in mind.

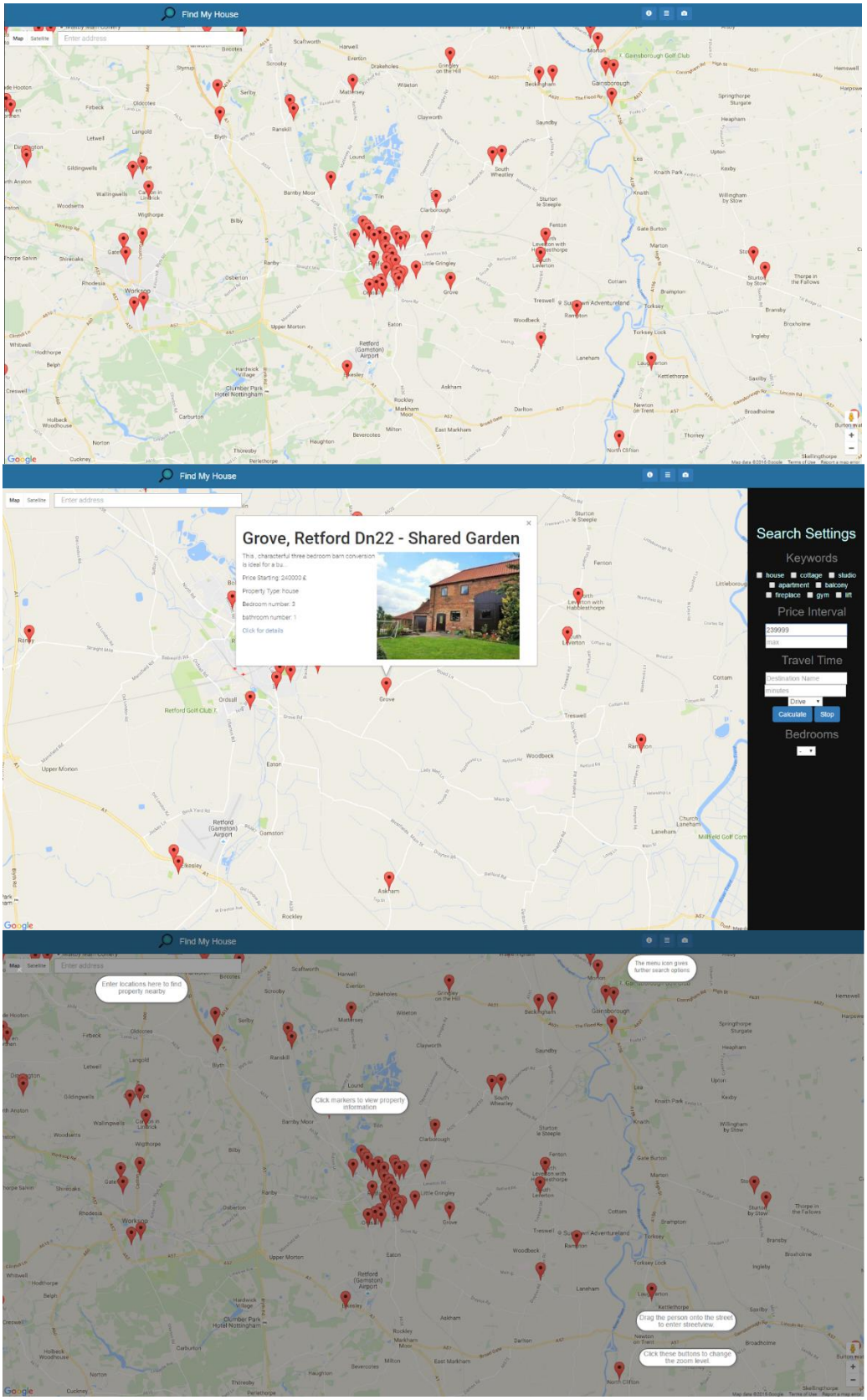
Final Interface Screenshots

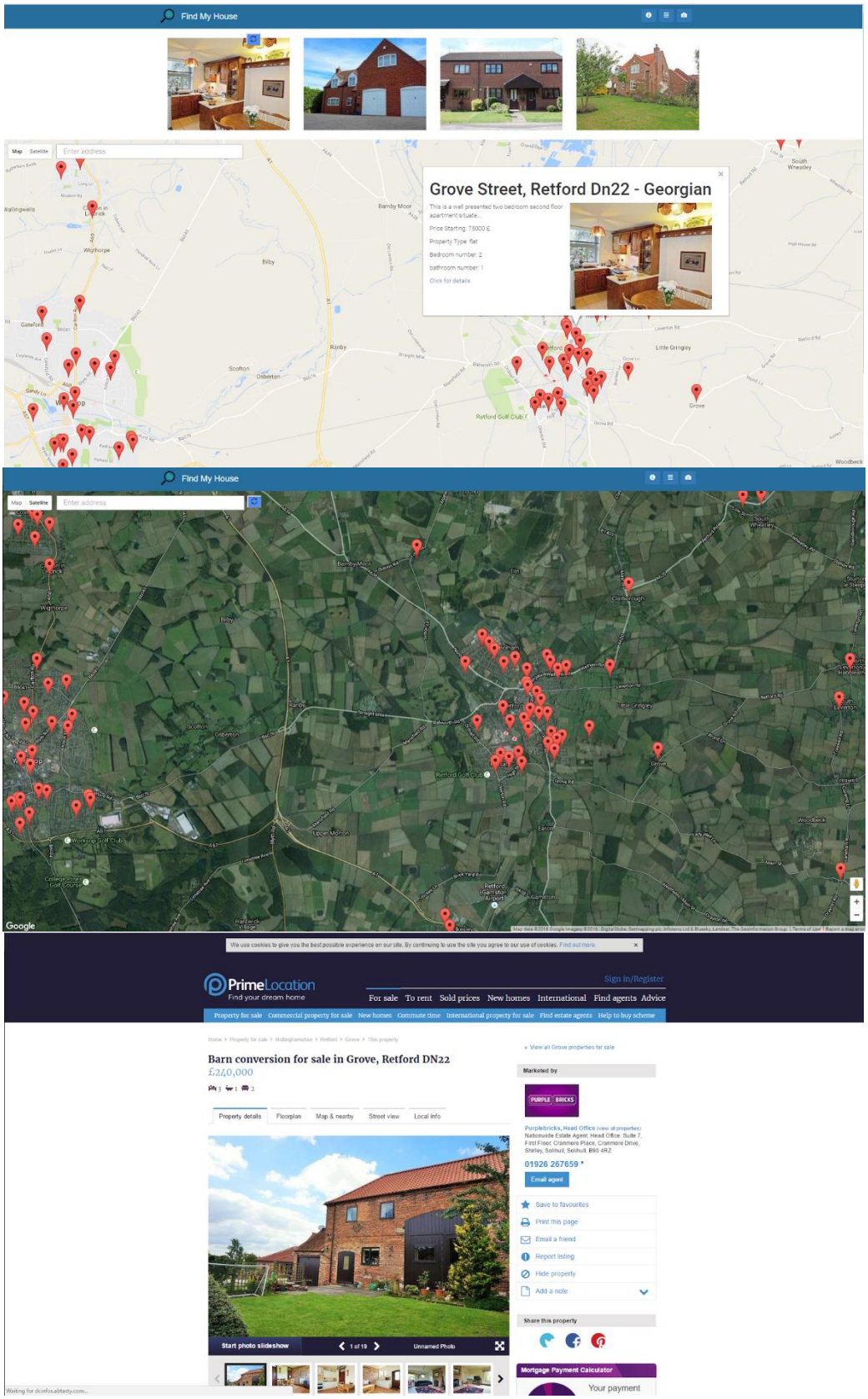







Find My House
London, United Kingdom
Google
Find My House
Enter address
Google
Find My House
A638
Google


Find My House
Map
Satellite
Enter address
Grove, Retford Dn22 - Shared Garden
This , characterful three bedroom barn conversion is ideal for a bu...
Price Starting: 240000 £
Property Type: house
Bedroom number: 3
bathroom number: 1
Click for details
Search Settings
Keywords
house
cottage
studio
apartment
balcony
fireplace
gym
lift
Price Interval
239999
max
Travel Time
Destination Name
minutes
Drive
Calculate
Stop
Bedrooms
Enter locations here to find property nearby
The menu icon gives further search options
Click markers to view property information
Drag the person onto the street to enter streetview.
Click these buttons to change the zoom level.


Find My House
Grove Street, Retford Dn22 - Georgian
Find My House
Barn conversion for sale in Grove, Retford DN22
£240,000


Find My House
A638
Retford, England
View on Google Maps
A638
Google
© 2016 Google
Terms of Use
Report a problem

Summary of Key Benefits and Achievements

This design implementation presents a new way for home buyers and investors to not only browse and search for properties but also gather information. With a much more interactive visual interface, this application can bring many benefits for users looking to invest in or purchase property.

The main function of our program is to allow:

- Users such as first home buyers or investors to quickly discover available housing within the UK. They're also presented with the option of personalising their searches through the use of filters.
- Observation of available properties and to observe the actually surrounding area of the available property for the audience to visualise how the livelihood of this area is and if they would actually be interested in living with such surrounding environment, as well as what is actually to be expected of the property's exterior/surrounding.

Filters and features available would include:

- The ability to compare distances between a chosen property and another property. In providing this function, users will be able to choose housing based on distances between other desired properties such as schools or plazas.
- New implementation which gives a better sense of direction and positioning geographically, the ability to see key areas surrounding the property instead of just a traditional list without any geographical or visual information of the surrounding area.
- Also including traditional selectable filters such as bedrooms amount, price range, keywords such as house/apartment, optimized for filtering out properties
- Mobile compatibility, so users can access the application wherever they are and be able to access geographical information of properties in desired area. As real estates traditionally does not do geographical representations much, our app can allow them to find the property straight away and head towards them if desired.

With these changes from standard real estate applications, we believe that this new design implementation can change the way people look at real estate and becomes easier to get into purchasing their own ideal properties in a good location.

Team Organization and Conclusion

Responsibilities and Organisation of the Team

Assigning of tasks was according to the strengths of each team member:

- Alva responsible primarily for the front-end development of the software. His responsibilities were to discuss and collate ideas and decisions made about the user interface within team meetings, and implement those ideas into the software. Working with Matthew, he was responsible for the development of the project. With the ideas of the team, he implemented the general user interface to our map screen, the main menu website, and the interface to all features our program has to offer. Furthermore, he also worked closely with Matthew to ensure that these visual elements hooked correctly onto the backend of our system for total functionality.
- Andrew worked on reports and some details on front end features.
- Charley largely worked on reports, both compiling and writing, attempted to help with coding, researched and organised the real estate API used. Back ups in case the suggested one was not feasible was also provided.
- Chris worked on reports, discussed and helped refine the idea. Advised on the technologies to used.
- Edwin worked mainly on the reports, also researching and providing ideas and options for the implementation of technologies during discussions, e.g. server research, Django server properties; as well as any other extra work which needs managing in a short period of time and making communications with Alva to keep track of his development side with Matthew to understand what is going on for the report.
- Matthew was responsible for server backend development of the software and client-side JavaScript coding. His responsibilities are to make sure the software is functional as discussed in team meeting. He made the use of Nestoria API to get the real estate's information from backend to frontend and processed the data to provide many searching features and display the data over front-end visual elements designed by Alva.

Team Conclusions

Alva:

How did the project go in your opinion

Although I'm glad that the project was a success and a functioning end product was developed, I am unhappy about the entire development phase of the project. I would have preferred if the development team consisted more than just 2 people, and that we would have had a more even work distribution. I felt that if the group worked more as a team and everyone put in their necessary input, then the project would have been developed faster, more stress free, and the final version would be more polished.

Issues/Problems:

I encountered many issues and problems that I would do differently in the future.
Firstly, the team workload was split in an inefficient manner. Two team members had to develop the entire project, while the rest wrote the report. While this was not what I had intended at first, it was the final outcome as certain members didn't properly do their preparation on using the Django language, and others were not responsive on time. This meant that the decision for this workload split was forced upon us. In the future, I will make sure that we clarify as a team what needs to be done and when it is due early on before the development begins.
There was also minimal testing done with the functionality of our software. Our software features were only tested for basic functionality, which meant that bugs most definitely would exist that we are not aware about. This was a result to only having two developers on the development team. In the future, I believe it would be beneficial to have a member focused on testing the program to ensure that full functionality is achieved.
Group meetings were also extremely inconsistent, and there were team members who would almost never show up to any group meeting. This resulted to team members who fell behind, didn't know what the project was up to, and hence had extremely minimal impact on the project development. In the future, it would be best to clarify at the beginning of strictly set project meetup times, and to enforce that group meetings were vital to proper development of the project.
The software itself was fairly unrefined and unpolished. Although it was functional, there are design choices that I would still change now. This was a result due to lack of team experience in front end development, lack of members developing the project and overall lack of time due to the slow development pace. In the future, not only should I remain more vigilant of design choices set out by the team, but I should also keep a closer communication profile with my tutor to validate all design implementation choices the group has made.
Overall, although our project has been a success due to initial functionality working and the implementation of additional features, there are many aspects I would change in the future to ensure a better development phase, and a more complete and polished end product as a result.

Andrew:

How did the project go in your opinion

The project had its own hiccups and there were some constraints with time for developing the web application. General inexperience with web development did not make this task any simpler. The final product was for the most part identical to the product that was envisioned in the planning phase. There are some usability issues that require sorting out, but overall the project accomplished what it was intended to do.

Issues/Problems

There was a lack of initiative and some uncertainty in the development process. There weren't enough people experienced with web development and testing of the web application. In general, there were some members who were left confused with what was happening.

The largest concern was with the distribution of work. If more people were experienced in web development or volunteered to do work, this would have been less of an issue.

How would you do it differently?

The final product came out well and I don't believe that I would like to change the design. If I were to do it again, I would prefer to have come back with more experience with languages and frameworks such as HTML, CSS, Javascript and Bootstrap. I would attempt to make the project more aesthetic and solve the usability issues. The work distribution is also an aspect that I would certainly revise and have more people working on backend and frontend development.

Charley:

How did the project go in your opinion

Despite have a functional product, the project itself was all over the place. Organisation was a struggle within the group as everyone had their own priorities to deal with. This meant that when it came to doing work, the majority of us were unable to contribute as much as we would've liked to.

Issues/Problems

Coming into the semester, a lot of us had little to no experience with web development and learning it from scratch within the given time frame was no easy task. It was made even harder simply because there was also a lack of experience with the languages we chose to use. This meant that when it came to developing the product, there was a lot of miscommunication, misunderstanding and overall dysfunction. The lack of team meetings was also a problem. Each member of the team had varying timetables and varying priorities which made it extremely difficult to set up a meeting in person or online. This led to unequal work distribution a lot of the time.

How would you do it differently?

If this project was done once again, I would try to be stricter in terms of meeting times and work distribution. I would also try to be more organised in terms of what i needed to do or achieve throughout the week so that the workload would be evenly distributed.

Chris:

How did the project go in your opinion

In the end, there was a functional product. The process of getting to the end product however was needlessly harder than it should have being. In my opinion, our project had a fine start, with everything on time. However, as time past, organisation and leadership within the team was lacking. Without an actual deadline, we would have taken much longer than anticipated. Despite those shortfalls, the project was a success with a functional product that can be expanded with extra planned functionalities that we were not able to work on, such as property search for Australia.

Issues/Problems

The main problem in this project was the lack of leadership. The team as a whole were eager to work on the project, with each contributing their own weight. However, the lack of direction and planning was holding us back on producing a product we had initially imagined. This meant the final product was lacking in the polish that we had initially planned, as well as extra features that would have made the product much more appealing to users.
In addition to the lack of leadership, the lack of experience in web development meant we had to often compromise on the methods used to achieve the project. These compromises led to time wasted on the already short development cycle. This included manually testing the Nestoria API to get the result we wanted. In addition, we had not anticipated the lack of free to use/publically available real estate APIs in Australia, which meant we were only able to implement the product for a UK market.

How would you do it differently?

Given another chance on the project, I would implement deadlines that would properly reflect our development goals. This would be in addition to taking charge of the team if no one raises to the occasion.

Edwin:

How did the project go in your opinion

Overall, the end product of the project is a success and it does what we specified originally, that itself was satisfactory. However, overall the development have to admit was a bit of a disaster in the mid process, if we divide the project into 3 phases of planning phase, development and report phases, we can categorise it into planning phase- the project went well overall, just a bit lack of communication between members and lack of motivation, the development phase was challenging, not that the allocated members were bad though we were all inexperienced, but that there was a lack of developers to work alongside with them, I would like to greatly credit Alva and Matthew pulling this off and the project without them would never been achieve this height, even though the rest did not participate in development, this is compensated by some of them being motivated to be the ones to complete the reporting phase. Otherwise, overall the project went reasonable, not exceptional, but many people pulled their own weight in different aspects, although it would be better if some more focus on development would be better.

Issues/Problems

The major problem we had would be the lack of development experience by most of the members as well as the lack of communication and time management of the members. Half of the team does not have as much experience with the web development or api, which lead to confusion and lack of direction to go, that was probably one of the main issues, as we head into the development phase, this became more serious as only part of the team can develop the actual product while the others are stuck with handling the report. Also, outside of mentor meeting, there was seldom communication or brainstorming session as a whole, though there was small 1-3 people talks at different times in discussing about ideas. I think

the major problem in the beginning was that we did not have a distinct leadership to coordinate members their workload, which only came from Alva afterwards in later stages, is when we actually did have some much more efficient workload being accomplished.

How would you do it differently?
Honestly, I think in the beginning we did not exactly grasp what was needed in the project for the later phases, we also had a lack of communication between members and lack of leadership, which if differently done, we should have elected a leader definitely earlier and assign members to different sections, given the right push we would have progressed much more further than what we have now. Also if done differently then we probably would have had alot more team meetings outside mentor meetings, but this project, as everyone was so busy with other courses, there wasn't much we could have done. Probably have to be more assertive as well in communicating, i find that the team wasn't that talkative or getting messages across. However, the most important bit after doing this project would be to set deadlines for each workload and be sure people has their allocated parts done, as well as grasp the entire requirements for the project and make sure i am capable of doing any parts of the project if i was allocated to the task.

Matthew:
How did the project go in your opinion
The project ends pretty well but there are some problems during the process. We did not plan it very well before coding phase and again, everyone is lack of experiences in web-development. However I am at least learning web-development from this project including a few programming languages, the way of using APIs, front end design strategies etc.

Issues/Problems and possible solutions.
We need more people on coding. As mentioned above, the work allocation was pretty bad thus everyone may not be able to see what needs to do. I would like to introduce an issue tracker in the future in order to put developing issues there so everyone can have their own work allocated.

The lack of web-development experience leads to an embarrassing situation. I code a new feature but it sometimes does not work directly and even make some other parts of code broken. It's time consuming to do basic tests on codes – I spent too much time on fixing the syntax errors.
I am learning it from zero web-development experience. I coded majority of backend and the understandability of my codes is very limited to new web-developer like me. This should be one of the reasons why our team contributions are low.

We need more frequent team meetings. Everyone is busy during the whole semester and we have different time schedules. We cannot focus on this project at the same time while there are always team member in other courses tutorial or doing other course's assignments. Again, issue tracker may be used in this case to have everyone's words there. We do not need everyone to be online at the same time, but everyone can say some ideas related to a particular issue.