Project Proposal

PROJECT NAME: Accommodation Web Portal

GROUP NAME: superNB

GROUP MEMBERS: Wenfei Guo(z5135080)
Tao Bai (z5111218)
Zhichao He (z5282955)
Shengchen Xue (z5111075)

TABLE OF CONTENT

Background/Problem Domain

Nowadays, more people prefer to look for accommodations online. This is because traditional way of finding place is too complicated. Online accommodation reservations is a popular method for customers to booking places. Customers can book rooms on a computer or mobile phone by using online safely to protect their privacy and by using several online travel agents to compare prices and facilities at different hotels.

Prior to the Internet, Customers can email, telephone the hotel or providers directly. Nowadays, Accommodation web portal also have pictures and information of hotel and rooms, to help customers get to know this place and choose the most suitable one. Many also allow reviews of the traveler to be recorded with the online travel agent.

Online accommodation reservations are also helpful for those people or hotel who want to rent out their available room. They can provide their places online to let more people to view and help to promote their accommodations.

Existing System and Drawbacks

Some existing accommodation booking websites like Airbnb mainly focuses on individual provider so customers could only search personal house property to rent. However, if a customer would like to live in resorts or any other accommodations owned by hotel groups, they need to go to another website and search for the accommodations separately. Therefore, we supply a platform for individual providers and hotel group providers so customers can find both types of accommodations on our websites.

Also, for some existing hotel booking websites, they lack of filters of furniture and domestic appliances but customers may have requirements on television or kettles so in our websites we add some relative selection on our website. Furthermore, some booking websites lack of filters of hotel types so when customers only want car hotels, they cannot do the searching efficiently. For that reason, we will add some specific filters on the hotel types.

Aim

Our team's goal is to build a convenient website to help customers to reserve or rent a suitable place and allow providers to publish their accommodations. In our project, we will concentrate on implement of functional design rather than user interface design. For that reason, UI design will have lower priority. Also, implementing concrete search filter has higher priority because it is an advanced feature which other websites do not have.

The deliverables of this project includes:

- Site text search with extra filters could give concrete search to customers and let them find the specific accommodations that they want.
- Order review will display all the order history of that user makes so that he could find where he rent before more conveniently.
- User information updating system is open to every user. They can update their personal details anytime in order to make sure trade information is up to date.

- Accommodation advertising Module will give provider access of uploading related information to help them promote their accommodations.
- Visitor request will provide customers chance to show their requirements and filter satisfied accommodations.
- Google Map will show the location of accommodations on Map picture and help user find position clearly.
- Auto Recommendation will provide users some protension accommodations for better customer experiences.

Project Choice

Our project selection is based on our team members' coding experience. Tao Bai and Wenfei Guo have more experience on html coding and website design. However, Shengchen Xue and Zhichao He is the only two people in this group who have learned MySQL and database system. Also, noone in this group has learned machine learning so this only project that is suitable to our group.

Epics

*One unit represents two hours working of two members

Accommodation Advertising Module

A good accommodation advertisement will attract consumer to choose. Therefore in this part we need to show information as detailed as possible. It can help consumers understand details of accommodation conveniently and find a suitable place efficiently, and it can also help providers to promote their accommodations.
We need to build a tidy and plentiful interface to let provider can show information that they wanted , and let customers understand this accommodation.
Features: （Google map，grade system）

- "Become a host" Button: If users want to rent out their house, they can click this button and it will jump to a new website name "house_inform.html" to fill information of their houses after login.
  "Become a host"Button need link with "house_inform.html" to make sure it can be jump into this pages after users click it.
- Selected boxes for choose type of provider:
  - personal(rent out their own places)
  - company(hotel)
- If provider click company before:
  - show select boxes for choose kind of hotel:
    - unit hotel
    - apaertment hotel
    - car hotel
    - Resort
  - show some select boxes for choose level of hotel:
    - five star

        - four star
        - three star
- If provider click personal before:
    - create some select boxes for choose kind of place which providers want to list in the first line:
        - 3 Choice:
            - Apartment,
            - Unit,
            - House

          just can choose one type
    - After choose place type, display “What will guest have?” and 3 new choices under kind of place:
        - Entire place(Guests have the whole place to themselves. This usually includes a bedroom, a bathroom and kitchen)
        - private room(Guests have their own private room for sleeping. Other areas could be shared)
        - shared room(Guests sleep in a bedroom or a common area that could be shared with others.)
    - create some short text boxes for provider to input based information of this accommodation:
        - location of this accommodation
        - max number of guest
        - numbers of bedrooms, bed, bathroom and kitchen
        - bed size
        - available dates
        - fee per week
    - In personal, create some select boxes for “What amenities do provider offer?”:
        - way of check-in(auto checkin or have to face to face)
        - Amenities(can be multi-selected, for example, whether have wifi, hangers, kitchen, hair dryer…if have somethings doesn’t appear in options, provider can add in “read more about this space”
    - In personal, create some select boxes for “What space can guests use?”: Kitchen, swimming pool, gym, parking, Laundry – washing machine, Laundry – dryer, Hot tub, Lift.
    - In personal, create some accessibilities text boxes for provider to fill in some accessibilities：

      It will fully shown under “Accessibility” button on the web. Providers can add any accessibilities here.
- Location text boxes and google map: Provider can enter place location or find it in google map which on the web. When providers enter location, it can auto display nearest location to help provider enter a correct location
    - unit number
    - street number
    - suburb
    - City
    - State
    - Country

    - PostCode
- A big text box named “read more about this space” for provider to add additional detail about this accommodation:
  need to have enough space for providers to fill information.
  when consumers click “read more about this space”, it will show these information which provider wanted to show.
- Able to upload pictures about this accommodation.It can let provider show their house more intuitively and can also help consumer know this house more directly.
- submit button at the end of the webpage: After providers upload their information, they can click this button and the system will transfer these information in backend and this place will enter into database.

Use react can help us to build these title, text boxes and selected boxes. Then build a form in jsp page and confirm necessary arguments and let provider uploaded, after providers finished upload, click “submit” button to submitted into backend using.

This epic has an estimated difficulty score of 8/10, with a time estimated of sixteen units.

Accommodation Booking Module

A list of accommodations that satisfied customer request will be displayed after user press the ‘search’ button. Customers can click any result accommodation for further detail.
Features:

- Provider’s personal detail: the contact email or mobile number of accommodation provider will be displayed for customers that are interested in this accommodation.
- Related accommodation information: Accommodation Advertisement information will be shown in tag way.
- check-in and check-out date: Users need to confirm the check-in and check-out date before they book the accommodation.
- booking button: After users make sure they want to book this accommodation, they can press this button to finish booking.
- Customer Detail Confirm: Customers need to fill in and confirm their personal detail before they finish booking.
- Statement approvement: This is the final step of Accommodation Booking module. Customers need to promise to obey the rule of booking before the action works.

Applying react technology to display personal detail and related Accommodation information.

This epic has an estimated difficulty score of 6/10, with a time estimated of twelve units.

Visitor Request Module

In this project, we focus on providing users the most accurate and specific information based on their requests. This module can be helpful in finding the acquired accommodation properties and filtering out inappropriate results. The “Request Screen” will contain a combination of CMS search bars, Date-Related Fields that can show calendar, a user-defined search bar and a search button.
Features:

- CMS Search Bars: We will operate CMS in WordPress way which can take advantage of websites’ own full-text search functions. In this way, users can receive the information they acquire with just a few clicks. All CMS bars are supposed to be in “any” mode before users’ modification. The four CMS bars that will appear on our web page are shown below.
    - Location Type Bar
        - all cities in NSW(e.g. Sydney, Newcastle, Albury)
        - any
    - Preferred Price Range Bar
        - lower than $100
        - between $100 and $300
        - over $300 per day
        - any
- User Defined Search Bar: If the user already has a specific preferred accommodation. User Defined Search Bar will allow him to manually insert script or code into the blank field. This will provide users the specific result without finding it in a list of accommodations that have similar properties.
- Search Button: After users fill in request information, they can press this button to search for result.
- Provider Type Bar:
    - Company
    - Personal
    - Any
- Date-Related Fields: Users can upload their preferred check-in and check-out date here. After they click the blank field, a calendar will pop up for them to specify check-in and check-out date.
    - Check-In Field
        - specified check-in date in dd-mm-yy format
        - any
    - Check-Out Field
        - specified check-out date in dd-mm-yy format
        - any
- Google Map: Users can specify their preferred location range by applying Location Type Bar and User Defined Search Bar. The location result can be shown in map. Google Map application can help to display all accommodations from that location in a map picture.
- Auto-Recommendation: Based on the request history of customers, the web page will display several accommodations as recommendation below the request section. These recommendations will automatically shown when customers log in the webpage again.

We will apply React technology to achieve request proposal of CMS bars, Date-Related Fields that can show calendar and the user-defined search bar. After customers uploaded their acquirement, they can simply click ‘search’ button which will deliver those data to backend for result. HTML and CSS technique can be helpful in User Interface design.

This epic has an estimated difficulty score of 7/10, with a time estimated of fourteen units.

Accommodation Search Module

This part contains one of the most key functions of the website. This module implements the searching system which helps the user to find target accommodation information according to the request. As there are thousands of different information in the database, using a search module is much better than looking for them aimlessly. Users’ experience will be largely improved. The implementation of this part is mostly backend’s work, so there is no much user interface of search module.
Features:

- MySQL database: all the operations such as adding, updating and searching for information are based on the database. The database must be designed according to the requirement of the website. We should create essential tables to store the information and control the amount of the tables. Otherwise, the relations between each table will be difficult to establish.
- Search button: this is the only visual part of the search module. After the button is clicked, the webpage will then process the input request and send a request to the backend to do the searching.
- Search request processing: the input request should be processed and analyzed for further use. First, we must check whether the submit button was clicked and whether there is any request is input or selected. Only after we make sure that the requests are valid can we continue to search for the information.
- Information searching: the backend does accordingly query operations in the database to find whether there is any appropriate item which matches the user’s request.
- Result display: After searching is done, the webpage should show the user the searching result. If there are no matching results, the webpage should tell the user about the situation. Otherwise, the result of searching should be displayed in a table by doing some query in the database. And also, the results should be sorted by released date or the rating. The result sorting is also implemented by MySQL query in the backend.

This epic has an estimated difficulty score of 4/10, with a time estimated of eight units.

Accomodation Review Module

After finishing the order, user can make a review to the current order. On the review page, user can rate this accomodation and write reviews to it.
Features:

- Rating bar: On the review page, user can select the number (0-5) in the rating bar to rate the accommodation that they lived in. The rating will be delivered to the backend database and the database will be updated.
- Review textbox: User can write review articles in the review textbox. After user write review articles, the review will be posted to the backend netserver (JVM) by HTML forms then the database updated and the review will be delivered to the frontend and shown on the accommodation website.

This epic has an estimated difficulty score of 2/10, with a time estimated of four units.

Google Map

In order to increase the location accuracy and improve the user experience, we introduce Google Maps to assist the user to find target accommodations and provide a better description of them. We mainly use the location display function of Google Maps in the website, user can see where exactly the accommodations are located in the real world. This feature will be embedded into the following modules:

- Accommodation advertising: When the hosts (property owners) upload their accommodation information, the address of the property must be filled in. After that, Google Maps will be automatically invoked to choose a location point which is closest to the address information. If the host is not satisfied with the location, the location can be modified by dragging the map and choosing new ones. After that, the address information will be stored in the database.
- Visitor request: The function is also useful for the visitors who are looking for accommodations on the website. When users choose the location of their target accommodations, besides entering the address or the suburb information, the website provides Google Maps for users to choose the location on the map. This is more visual and convenient for users to choose.
- Search module: After processing the users' request (including the location information), the backstage will search in the database for matching results. After that, the results will be displayed on the webpage. There will be a Google Map on the top of the page where most of the accommodations found will be displayed with brief descriptions. The center of the results on the Google map will be the location that the user chose if any.

This epic has an estimated difficulty score of 5/10, with a time estimated of ten units.

Auto-Recommendation

The implementation of this part is very important to the website, sometimes the results which strictly linked to the users' requests are not enough for the users to compare and choose. Thus we need to implement a auto-recommendation system to provide more opportunities to both visitors and hosts. To implement the auro-recommendation system, we basically use Collaborative Filtering strategy, including User Collaborative Filtering (UserCF) and Item Collaborative Filtering (ItemCF).

- UserCF: A method of making automatic predictions about the interests of a user by collecting preferences or taste information from many users. In our system, we use this method to predict users' target and interest based on many similar users' browsing record. Then a few recommendations will be generated by some analyses. Results produced by this method is more uptodate and is closely related to the current trend in choosing accommodations.

- ItemCF: A strategy which automatically predict the interests and targets of a user by collecting the history preferences and searching records of the same user. The difference between this strategy and UserCF is that the recommendations made by ItemCF will be more accord with the user's own styles, price and other preferences. Thus is most likely to find out what the user really like.

This epic has an estimated difficulty score of 5/10, with a time estimated of ten units.

User Information Management System (*Some features may partially implemented)

For this website application, there are two types of accounts : Website User and Administrator. The user could make online requests or provide new accommodations only after they signed in as user so users should provide some personal information to create their own user accounts. Also, we need to encrypt users' information because account information should be private and cannot be viewed by the administrator account.

Features:

- The 'Sign Up' and 'Sign In' buttons will be on the right upper corner of the main page and search page of the website. After clicked 'Sign Up' button, the page will skip to the sign up page. After clicked 'Sign In' button, the page will skip to sign in page. Also, if the user doesn't sign in but he clicked the 'Post a new accommodation' button on the main or 'Make a request' button on the accommodation details page, the current page will skip to the sign in page.
- On the 'Sign Up' page, user should fill in some text boxes to finish sign up.
    - Account id text box, user should provide their unique user id, user id should be only in 'A-Z', 'a-z' and '0-9'. The system will remind user to create another ID if the current id that this user creating is existed.
    - Account password text box, user should provide their user password, password should be in 'A-Z', 'a-z' and '0-9'.
    - Security question bar and security question answer. User could choose one of the security questions and type the answer to the security question.
    - The contact email text box. The contact email is used to verify the identity of the user.
    - The 'Finish Sign Up' button, if all the information that the user provided is valid then skip to the main page of the website. Otherwise, refresh the page and let user sign up again.
- On the 'Sign In' page, user should fill in some text boxes to finish sign in.
    - Account id text box, user should provide their user id.
    - Account password text box, user should provide correct user password.
    - If user forgets the password, he can click the 'Forget Password', then to the relevant page to answer the security question and reset the password.
- After user created a new account, the user id and the answer of security question will not be encrypted but the password will be encrypted by Rijindael Encryption (AES) and stored encrypted password in the database. When the users try to sign in, the system will encrypt the current password that user typed and do a query to the user id that user

typed to find corresponding encrypted password in the database. After that, the system will do a match for the found encrypted password which in the database and the current encrypted password. If these two encrypted passwords are in same, then the user log in.

- To store users' information, a net server is necessary because this is a public website. For that reason, our team will use Google Cloud as net server to store the information. Also, MySQL is used to add, remove and search the information in the backend database system.
- The username will be displayed on the right upper corner after user signed in. After clicking the username, the page will skip to user profile page. In the user profile page, there are three buttons (Accommodations, Orders, Profile) and relative child pages
    - On the 'Accommodations' page, there will be a list of pictures of accommodations that this users supplied and users can click each picture to visit the relative information page.
    - On the 'Orders' page, users can view all the orders they made. Each order will display brief information. Also, users can click each order to view whole details.
    - On the 'Profile' page, users are able to modify personal details (contacts, name, etc)
    - These three features above will be implemented by HTML form and updating of database.

This epic has an estimated difficulty score of 9/10, with a time estimated of eighteen units.

Final Epic Selection

From the above epics, we have chosen to deliver the following:

Fully Implemented:

Accommodation Advertising Module

Visitor Request Module

Accommodation booking Module

Accommodation Search Module

Accomodation Review Module

Partially implemented:

User information Management System

Google Maps

Auto recommendation module

Team Skill Set

Language: Java, Java Scripts, MySQL, React, CSS, HTML, Flask
Strengths: Design, communication, organisation, teamwork, project management
Experience: User Interface design, Website development, Neural Networks and Deep Learning
Frontend: Wenfei Guo(z5135080)，Tao Bai (z5111218)
Backend: Zhichao He (z5282955), Shengchen Xue (z5111075)

Project Methodoldgy

Agile Software Development Methodology will be applied as our project methodology. Scrum is an agile process that allows us to rapidly and repeatedly inspect actual working software. Our team contains four members, with product owner and scrum master roles.

Scrum: Github Board
Code Management: Github
Programming language: explained in team skill set section
Project Scrum
Scrum team:

Wenfei Guo(z5135080@ad.unsw.edu.au)

Tao Bai (z5111218@ad.unsw.edu.au)

Zhichao He (z5282955@ad.unsw.edu.au)

Shengchen Xue (z5111075@ad.unsw.edu.au)

Scrum master: Wenfei Guo
Developer: All scrum team members
Sprint Review Meeting: 10:00 - 11:00 Tuesday
Sprint Retrospective Meeting: 13:00 -15:00 Thursday

Project Time-Frame and Schedule contain fortnightly demos and final presentation. Detailed Stand-up/Sprint Schedule and weekly meeting times are also shown in sections below.

Project Time-Frame/ Schedule

Friday 4th October	Project Proposal	week3
Wednesday 9th October	Progress demo	week4
Wednesday 23th October	Progress demo	week6
Wednesday 6th November	Progress demo	week8
Wednesday 20th November	Project final demo	week10
Wednesday 20th November	Software Quailty	week10
Wednesday 20th November	Peer Assessment	week10

Stand-up/Sprint Schedule

Our team will use Agile Software Development Methodology to make customer involvement more active, with stand-ups three times a week and fortnightly sprints. We will also update individual working diary and code on Github for members to know where others are up to.

Stand-up

- Monday (In Person)
- Wednesday (In Person)
- Friday (In Person)
- Sunday (Online)

Sprints

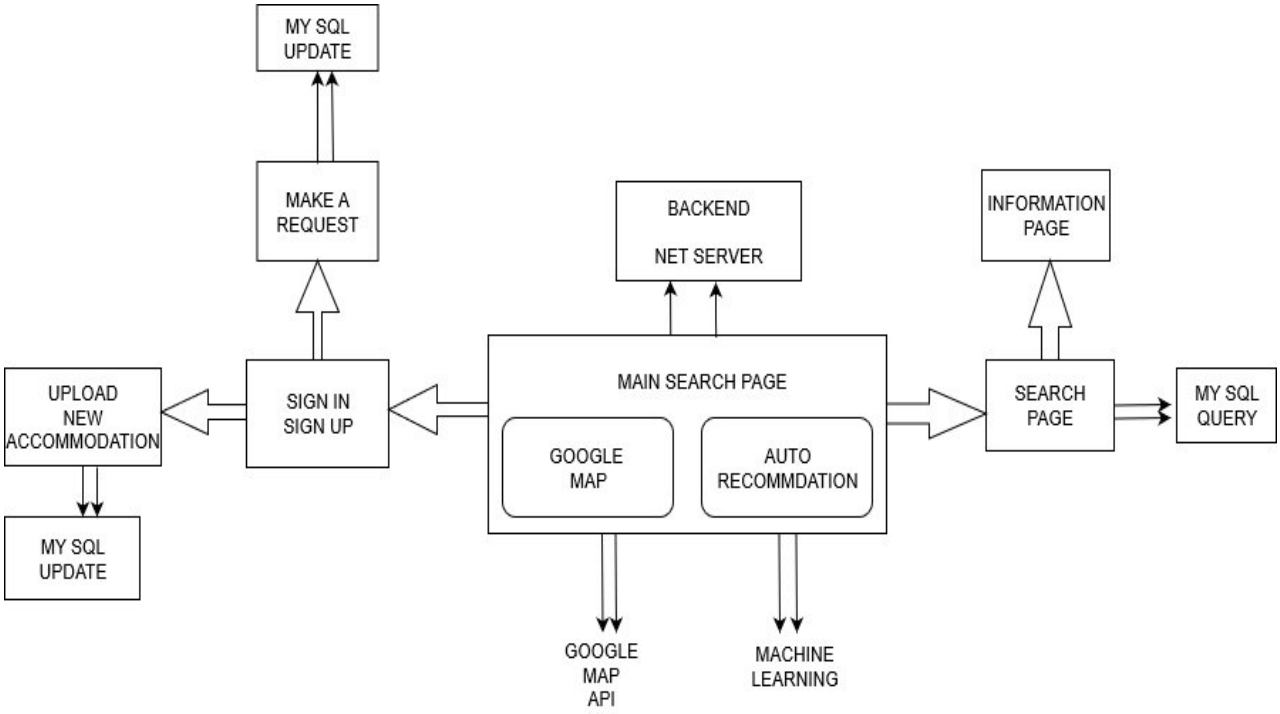
- Week 2/3
    - Visitor Request Module Epic User Interface design section
      (Wenfei Guo(z5135080)，Tao Bai (z5111218))
    - User Information Management System Epic login page/ register page User Interface design section
      (Wenfei Guo(z5135080)，Tao Bai (z5111218))
    - Backend environment set up and Partial Database construct
      (Zhichao He (z5282955), Shengchen Xue (z5111075))
    - Embed Google Maps in the website (as this part will occur in many modules)
      (Zhichao He (z5282955), Shengchen Xue (z5111075))
- Week 4/5
    - Accommodation Advertising Module Epic User Interface design section
      (Wenfei Guo(z5135080)，Tao Bai (z5111218))
    - Implement the backend of the User Information Management System
      (Zhichao He (z5282955), Shengchen Xue (z5111075))
    - Implement the backend of the Visitor Request Module
      (Zhichao He (z5282955), Shengchen Xue (z5111075))
- Week 6/7
    - Accomodation Review Module Epic User Interface design section
      (Wenfei Guo(z5135080)，Tao Bai (z5111218))
    - Implement the backend of Accomodation Review Module and Accommodation Advertising Module
      (Zhichao He (z5282955), Shengchen Xue (z5111075))
- Week 8/9
    - Finish User Interface design (frontend design) Section
      (Wenfei Guo(z5135080)，Tao Bai (z5111218))
    - Backend & Frontend connection
      Wenfei Guo(z5135080)，Tao Bai (z5111218), Zhichao He (z5282955), Shengchen Xue (z5111075)
    - Implement the Auto-Recommendation system
      Zhichao He (z5282955), Shengchen Xue (z5111075)
- Week 10
    - Presentation preparation

Wenfei Guo(z5135080)，Tao Bai (z5111218), Zhichao He (z5282955), Shengchen Xue (z5111075)

Weekly Meeting Times

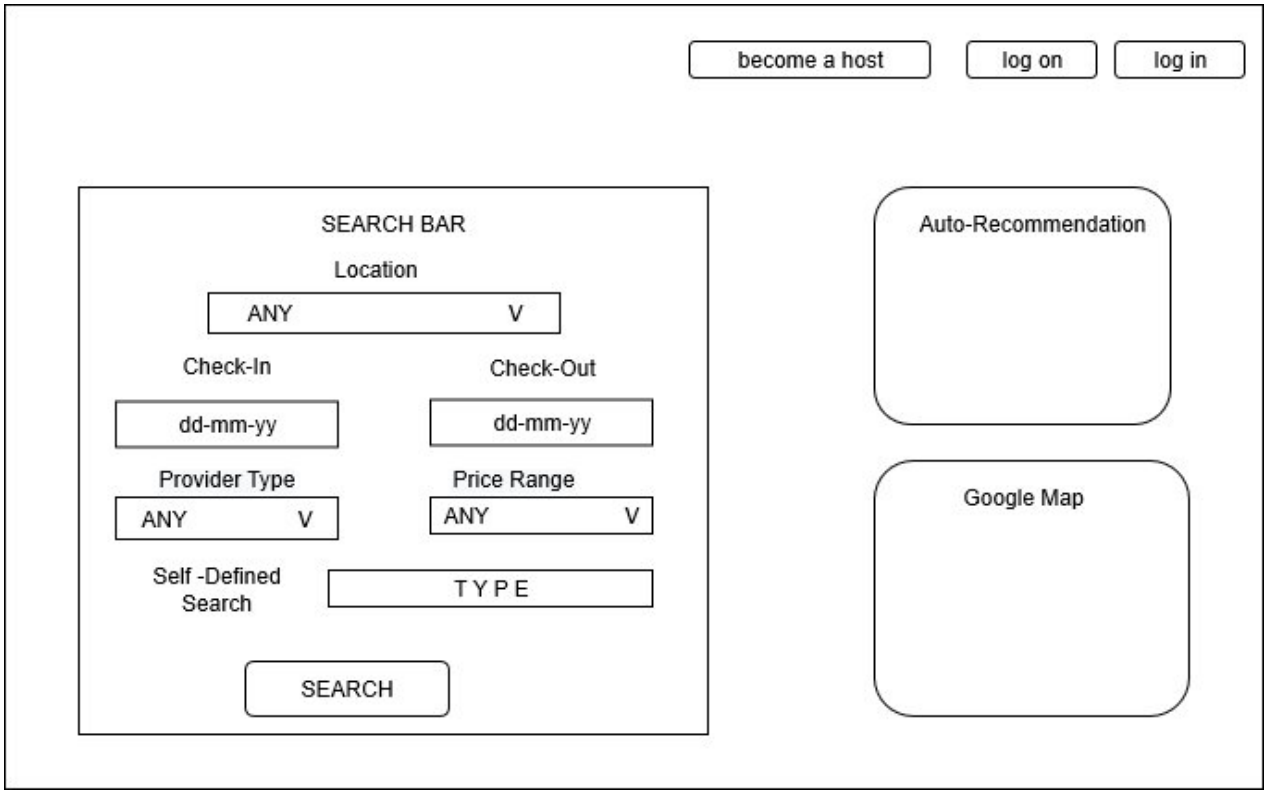
- Monday Red Center 14:00 - 16:00
- Wednesday BrassME Lab 10:00 - 12:00
- Friday Red Center 11:00 - 13:00
- Sunday Wechat Group 15:00 - 17:00

SYSTEM MODULES AND FLOW

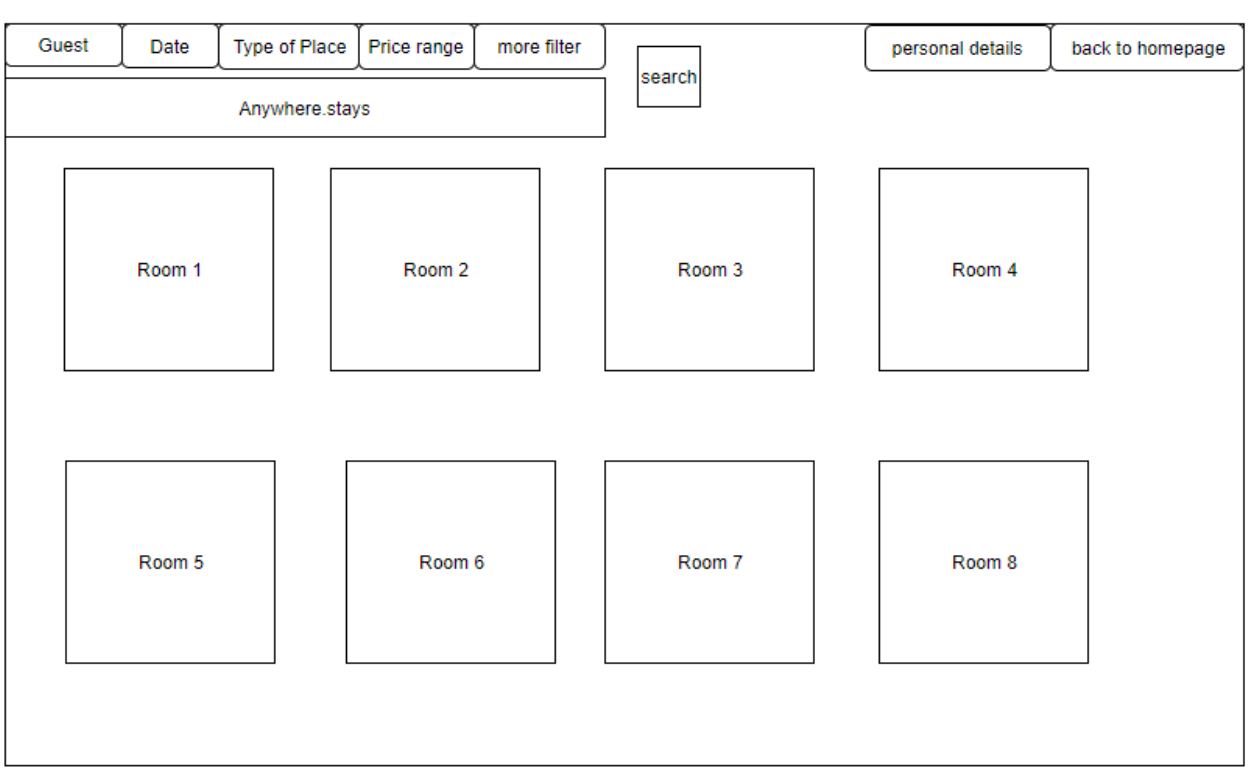


UI DESIGN DISPLAY

HOME PAGE:



RESULT PAGE:



ACCOMMODATION PAGE: