Website for Hotel Booking

General Description

The proposed web application allows the user to book hotels. The user can:

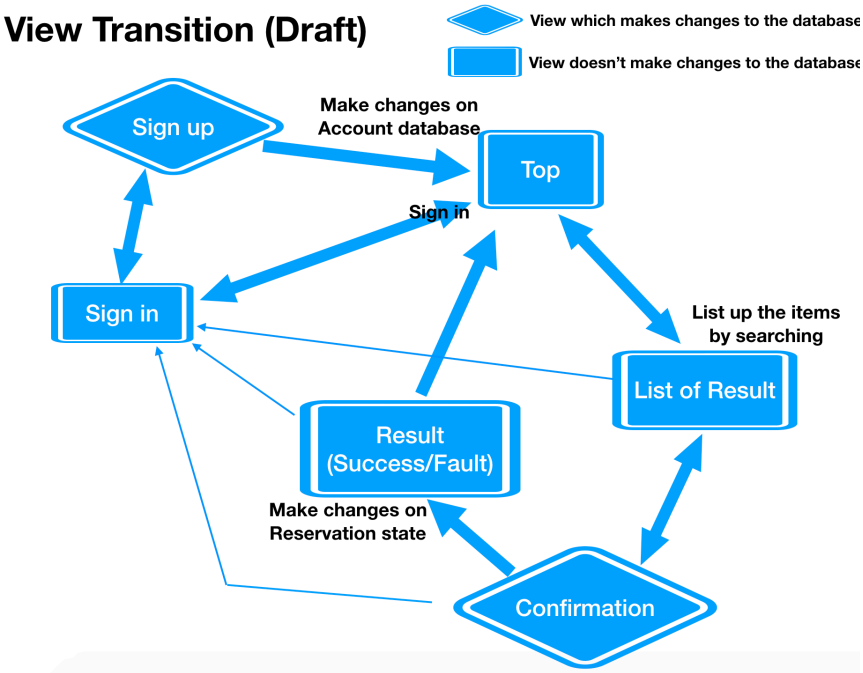
1. Select the city from the predefined list.
2. Select the type of a room needed (standard / double / deluxe).
3. Choose the desired date in the calendar.
   (We assume that the length of stay is always one night).
4. Choose the hotel from the list of found matches.
5. Type own name and confirm own choice with OK button.
   (The system should notify that the reservation is completed successfully.)
6. Additionally, the user can register in the system, so that the app will not have to ask the user name during the next booking.
7. (Optional) Allow to book a hotel for any number of days.

Notes:

1. There are N (≈4) cities in the list.
2. Each city has at least M (≈5) hotels.
3. There are at least R (≈10) rooms in each hotel.
   (Each room belongs to one of possible types.)
4. Reservation can be done for the next 14 days.
5. Users need to log in to an existing account or create a new account to make a reservation.

Suggestions for the user interface:

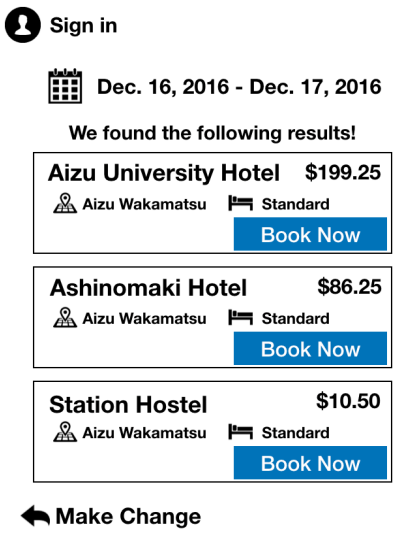
1. After the desired city, date, and the room type are chosen, the app displays the list of matching hotels, having available rooms for the specified criteria.If no hotels is found, the system suggests to the user to change search parameters.
2. After confirming the booking, the system displays the details (date, city, hotel name, room type).
3. Registration requires entering a unique combination of login/password.
4. If a certain user is logged in, there is no need to ask for a user name anymore.



User Interface (Sketch)

1. Top

2. List of search result

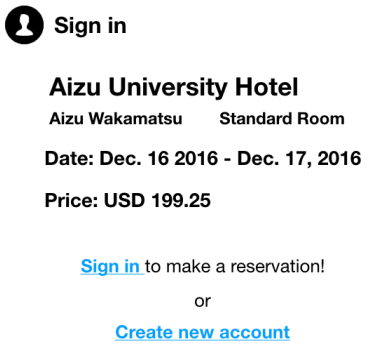


Make Change

3. List of search result (No items found)

4. Confirmation (Signed in)

5. Confirmation (Not signed in)

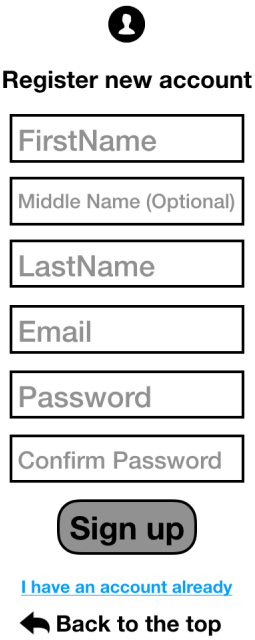


6. Sign in

7. Result (Success)

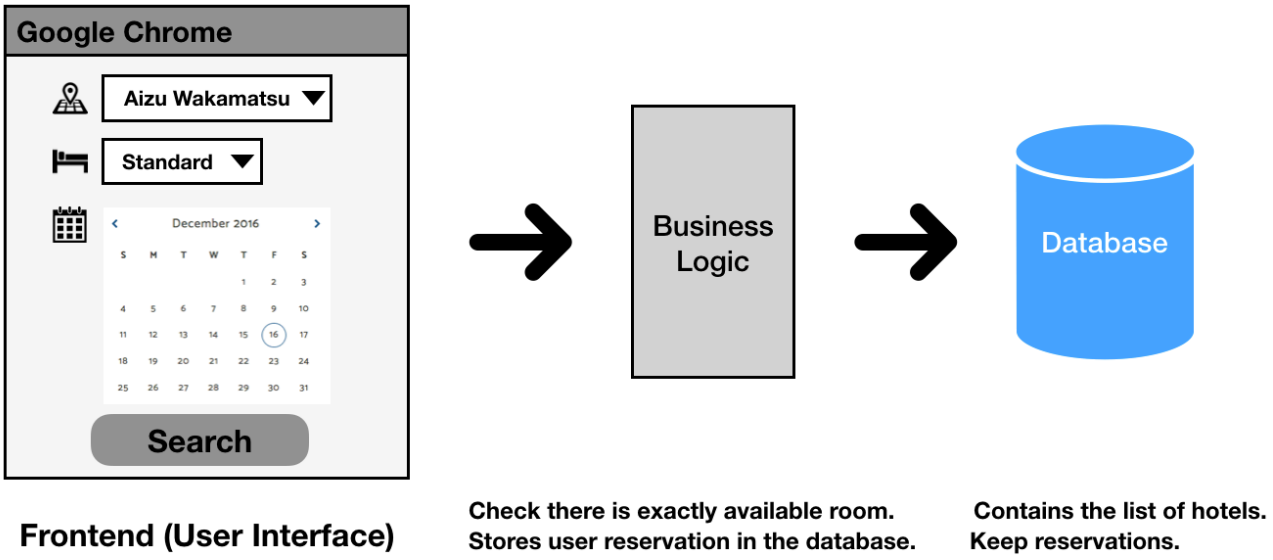
8. Result (Fault)

9. Sign up



Proposed Architecture

The system will be implemented using client-server architecture. It will include the elements shown in the picture below. We will use Django framework for building the main app logic, and Vue.js for the rich user interface (Optional).



Database Structure

The database will contain 5 tables, cities, hotels, rooms, users and reservations, having the following structure.

Table *cities*.

Column	Type	Comment
id	int	Unique city ID
name	char(64)	City name (such as “Aizu Wakamatsu”)

Table *hotels*.

Column	Type	Comment
id	int	Unique hotel ID
name	char(64)	Hotel name (such as “Aizu Grand Hotel”)
city	int	ID of the city (foreign key).

Table *rooms*.

Column	Type	Comment
id	int	Unique room ID
hotel	int	ID of the hotel (foreign key).
type	char(3)	Room type in defined number (e.g. Standard room is STD)
price	decimal	Price shown in the list and confirmation. Max 6 digits and 2 decimal places.

Table *users*.

Column	Type	Comment
id	int	Unique user ID
f_name	char(16)	User’s first name (such as “Donald”)
l_name	char(16)	User's last name (such as “Trump”)
m_name	char(16)	User's middle name (such as “John”) (It can be empty)
email	email(320)	User email address (such as “m5000000@u-aizu.ac.jp")
password	char(128)	Hash value of the password

Table *reservations*.

Column	Type	Comment
id	int	Unique reservation ID
date	date	Date of the reservation (such as “2016-12-16”)
room	int	ID of the room (foreign key).
user	int	ID of the user (foreign key).

Project Plan

1. Setup the system (all required software tools).
2. Create the database and design its structure.
3. Create the business logic. Make sure it works without user interface.
4. Create the prototype of the user interface in plain HTML.
5. (Optional) Create a rich interface in Vue.js