
# Location-based Panoramas of HKUST

Kong Yuet Yee
Prof. Gibson Lam and Prof. David Rossiter

# Introduction

The system contains a map view of HKUST with its panoramas. This allows users to have an experience locating inside the campus even if the place has not been visited before.

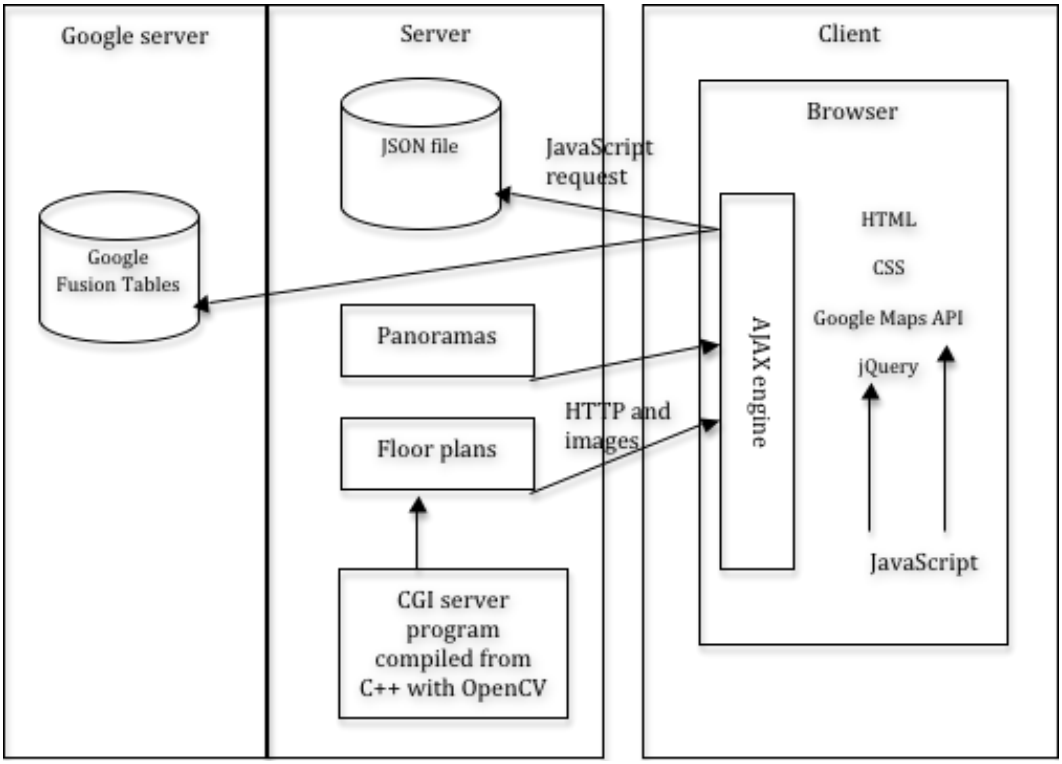
Regarding to the large area of the whole campus, a prototype is built to test the design and concept work. Therefore, this project is focuses on one of the buildings in the campus, Lee Shau Kee Business Building, for two reasons:

- It has a similar structure with the academic building of having several storeys but a smaller area.
- It is a new academic building that opened to use for two terms in Fall 2013. It is more possible that people do not get familiar with it and need the assistance of this system.

Objectives:

- Collect panoramic views of all floors of the Business Building.
- Build a data storage to store the collected panorama data.
- Provide a user-friendly graphical user interface to display requested panoramas.

# System Architecture



# Design and Implementation of User Interface

**Search bar**

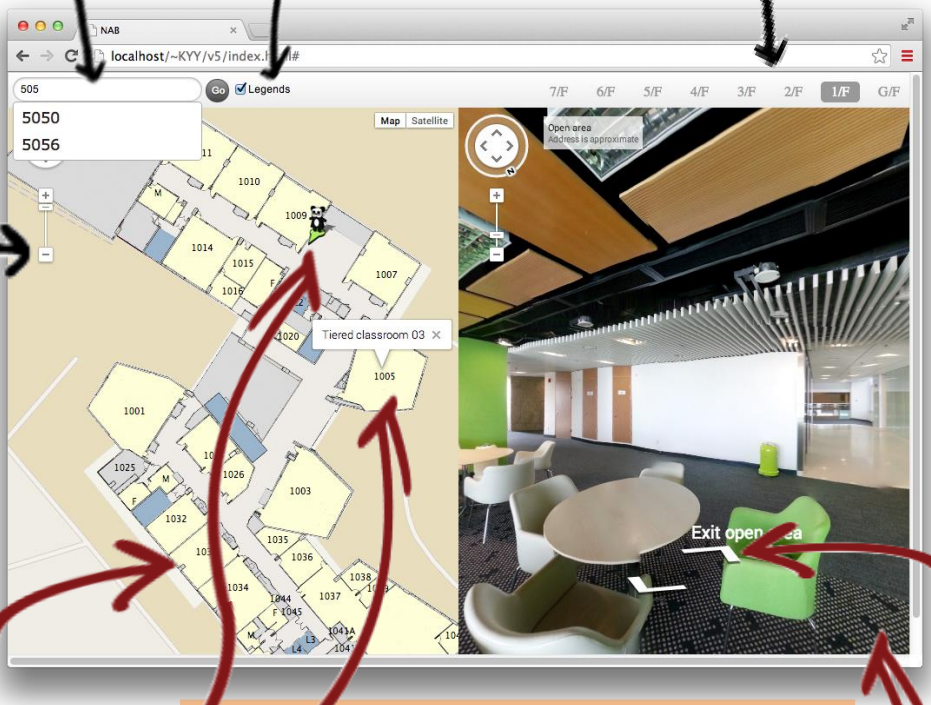
When user starts to input keywords of location, a drop-down menu with a list of locations matching with input is displayed through AJAX.
Then user can select the location with available panorama to view.

**Legend selector**

User can check or unchecked to choose legends are displayed on the floor plan or not.

**Floor selector**

When user selects a floor, the corresponding floor plan and legends are requested from the server and data source.
The pegman is updated to a location with available panorama that closest to the original latitude and longitude.
Finally the corresponding panorama is returned from the server and displayed on the right panel.




**Map controls**

They are provided by Google Maps API. When the zoom level is changed by user, another floor plan is returned and displayed.

**Links**

Given a particular panorama, its links to other panoramas are fetched from the JSON file using AJAX, and pushed to a stack, and then displayed with Google Maps API. When user clicks the arrow, it links to another panorama and the pegman updates its position.

**Floor plan**

In order to overlay floor plan on top of Google Maps through API, special floor plans of each floor of each zoom level are prepared. This is achieved by running CGI program on the server that programmed in C++ with OpenCV library.

**Legends**

Data about legends are stored in Google Fusion Tables, and they are fetched once the webpage is ready.
User can click on the legend for detailed information about the location.

**Pegman**

Location

When user clicks the links on panorama, the pegman is displayed on an updated location.
When user drags the pegman, it is forced to be placed to a position with available panorama that nearest to the position that user releases the pegman.

Heading direction

The heading direction of the pegman is calculated from the heading direction of the panorama. Then the server returns the correct orientation pegman image.

**Panorama**

Panoramas are stored on the server grouped by its located floor.
Panorama data is stored in a JSON file.
When a panorama is requested from either dragging a pegman, or clicking the link on a panorama, or selecting another floor, a panorama is requested by a panorama ID from the server. The server returns the panorama by HTTP.

# Panorama collection

With a fixed position, 37 photos are taken and they are stitched together to form a panorama.

The panoramas have the following requirements in order to use Google Maps API:

- Equirectangular projection
- Panorama dimension is 2:1 that covering 360° in horizontal direction and 180° in vertical direction

# Floor plan manipulation

First, floor plans of each floor and each zoom level are rotated and aligned with the same orientation of the Business Building. Then, the background of all the floor plans are removed by C++ programming using OpenCV library. Here is the algorithm:

1. The floor plan image is split into three matrices according to three channels, red channel, green channel and blue channel.
2. The original floor plan image is converted into gray image.
3. A mask is designed by examining each pixel of the gray image such that if the pixel value is greater than 250, the pixel value of the mask is set to 255. So a binary mask is created with white background and black foreground.
4. The mask is combined with the split images to form a 4-channel image, where the fourth channel is related to transparency.
5. An output floor plan is formed by assigned each pixel's fourth channels value. If the pixel value of the mask is 255, the fourth channel value is set to 0, i.e. transparent.

After that the floor plan image is split into tiles of dimension 256×256 pixels.

Finally, all the images are saved into different directories according to their floors and zoom levels.

# Results

- Besides the 7th floor is inaccessible, 100 panoramas are taken inside the Business Building covering all other floors.
- The floor plans and panoramas are displayed in a user friendly way by a webpage. It is responsive to browsers of different sizes.