

End User Application

Platform: iPhone

Coding Language: C# via iPhone SDK

Functionality:

This program is the interface which the end user sees. It will use the iPhone's google maps technology as a basis. The application's main function is to send the user's GPS co-ordinates to our server along with a query for the closest tagged locations surrounding the users current position. It will receive from the server a list of all tags within a set radius of the provided GPS co-ordinates.

The information returned for each tag will consist of a company name, the GPS co-ordinates for placement, an item name, item code and a jpeg image, and a jpeg or gif thumbnail.

This application will display the thumbnail at the GPS location given. If the phone operator enters the vicinity (a set radius) of the displayed thumbnail, the program will display the full jpeg image and provide options as to how the user can interact with the location. The user can either choose to ignore the image, or interact with it. In the case of interaction, the application must send information back to the server including GPS co-ordinates, and the item code.

Key Technologies:

- Google Maps
- PHP or similar language for server interaction
- iPhone SDK

Essential Capabilities:

- The device IMEI number must be sent along with EVERY server request. As our end users will not be setting up accounts, the only way to distinguish between users is via their device IMEI so this is essential.
- The server interaction must be SECURE. As we are sending personal GPS co-ordinates, we cannot run the risk of having this information exploited.

Data provided by us:

- All graphics and design
- Required data structures

Client Front End

Platform: Web

Coding Language: XHTML with CSS, using JOOMLA for web app. dev.

Functionality:

This is the web based front end for our clients. This will consist of approx. 25 pages, but many of these will double up on information (and programming functions). The whole interface will be styled with CSS to keep consistent.

At the forefront will be a client login (or sign up sheet for new clients). Once logged in the client will be provided with the following options:

- Upload content and change existing content.
- Edit personal information
- Make payments
- View a tagged google map for all clients' tags

All the above information will be retrieved from a server database using PHP (or similar).

Key Technologies:

- Google Maps API
- PHP or similar language for server interaction
- XHTML and CSS (and JOOMLA or similar)

Essential Capabilities:

- When uploading content clients must be able to set GPS display coordinates by clicking on a google map.
- The interface must be capable of referring information to a 3rd party credit card payment gateway via a secure connection.

Data provided by us:

- All graphics and design
- An image representation of every required page

Server Back End

Platform: Linux

Coding Language: MySQL, PHP

Functionality:

The server side programming is the most complex of this project. The first function of the server will be to store information in several databases:

- **Client Database:** This will store the general information about the client including personal information and passwords. Each client will be given a ten digit unique code, generated by the server in a sequence beginning with 1000000000.
- **Geo Tag Database:** This will store the content each client uploads along with information about specific tags including GPS data and tag hit counters. Each Geo Tag for a unique client will be given a four digit randomly generated code.
- **End user Database:** This will be the largest of the databases. Each end user device will query the server using the device IMEI. If the IMEI is not already in this database, a new entry will be created with the device IMEI as the unique user ID. As the user views and collects tags, additional information will be stored in their entry for later retrieval by the device.
- **Statistics database:** Every time any interaction occurs between a device and the server a log entry must be recorded on a stats database. The log will include the device IMEI, the client unique code for the viewed tag along with that tag's unique code, and whether or not the user chose to interact with the tag.

Along with the databases, the server must have many re-usable functions for data retrieval / storage. For example, we will need a 'currentSubscriber' function which checks the users IMEI number against the current user database and returns a Boolean value to the device. We will also need a 'Find Nearby Tags' function which takes GPS co-ordinates as input and returns an array of nearby tagged locations. Each of these functions could be stored in a separate PHP file on the server for easy retrieval by the end user application.

Key Technologies:

- PHP
- MySQL
- Data logging

Essential Capabilities:

- Security of information is our number one priority with the server programming. We cannot run the risk of having this information compromised.

Functionality Spec Sheet

- We cannot be limited to conventional database size restrictions. Specifically, our stats database will be quite large with millions of entries so it is essential that the database can handle this.
- SPEED. The coding MUST be efficient. The devices accessing the server's functions will have limited bandwidth connections so it is extremely important that they be served information as quick as possible.

Data provided by us:

- Database structure charts

Admin Control Panel

Platform: Web

Coding Language: XHTML, CSS and PHP

Functionality:

This part of our project is almost identical to the client interface. Upon entering the admin username and password, the admin will be given two search boxes – one which searches current client company names, and one which searches current client unique IDs. Upon clicking search the page will query the database with the parameters and return to the admin a list of clients matching (or close to matching) the parameters. Once a client is clicked from the list the admin is presented with a very similar interface to what the client would see, with the exception of every piece of information being editable.

Essential Capabilities:

- Query the server and allow information manipulation.
- Again, SECURE. This is extremely important here!