

**TRAVELER’S EYE**

**Project Title:**

Traveler’s Eye

**Introduction:**

Being international students we all understand the intricacies involved in travel. The stress of keeping track of all the information about the destination and the travel, especially when it comes to travelling alone, takes away from the fun one can have while travelling. Traveler’s eye is aimed to make a traveler’s life easy by providing relevant useful information whenever the user needs it, at a simple button click or the touch of a finger.

**Project Goals and Objectives:**

Traveler’s Eye will be able to provide information regarding convenient places of stay with respect to the user’s area or place of interest. Also the user would be able to select if he/she is on a business trip or a leisure trip. Based on the information provided by the user the app will be able to assist the user in a better way. The app will also provide information like the traffic conditions and the weather conditions.

Once the destination has been selected the app will be able to provide the traffic and weather related data so that the user can make an informed decision on the modes of transport or the time of travel. The app also provides a lot of details about the destination like the currency used, language spoken, history of the location, crime rate, which areas are a must visit and which places to avoid.

The overall objective is to provide the user a very safe and comfortable experience while he is travelling.

**Related work:**

There are a lot of different applications available which offer some of the similar features but they are mainly localized to only one specific location or city. Tourist Guide Paris, Yokohama Tourist Guide (Local), Tourist Guide for Greece, Manchester Tourist Guide etc. There are a few other applications which are not localized, but they are mostly based on booking trips.

None of the other applications offer the complete package like the Traveler’s Eye. Almost all the single city based tourist guide applications only concentrate on simply showing the user all the popular places in that particular city, a user is not allowed to choose any other city so if your trip covers multiple cities then you will have to install new app for each new city.

And all the tourist guide applications which are dynamic and let the user select different destinations are more concerned about booking tickets, hotels, luxury trips, cruises etc. Also they are only concerned about leisure trips and the most exotic locations and destinations. Such applications manly focus on selling tickets rather than providing the user with good usable relevant information in a convenient and hassle-free way. A normal day to day traveler or a low budget traveler has no relevant information in these applications.

In Traveler’s Eye we try to be as dynamic and flexible as possible. The user can travel to any number of cities using just one application. Also the Traveler’s eye provides more than just information about the tourist spots, it provides information about weather, traffic conditions, navigation etc. These services provide useful information to all the users no matter what kind of trip they are on. Also the security features in the Traveler’s Eye application are very unique. So far we did not come across any application which provides the automatic location update service.

**Proposed System**

**Requirement Specification**

**Functional requirements:**

1. Emergency services: (priority high)

* This function facilitates user to inform the local emergency services like local police, hospital by just giving a click on the corresponding button, in an emergency case.
* It also handles the task of updating his location to the emergency contacts.
* It also allows user to update/notify user’s location to the contacts defined by the user at regular intervals of time. Thus allowing him to stay in touch with group people as he intends, throughout his journey.
* It has to function in such a way that it helps user in the needful times.

1. Location details: (priority high)

* People on travel either on business trip or leisure trip will attain a situation at some point of time to know their current location. System must be able to retrieve the user’s location and display it for him using geo-location services.
* User will also have the facility of navigation. He can give some destiny and attain the route map to his desired location from his location.

1. Navigation: (priority high)

* Navigation is one of the important service provided to a user.
* It allows user to get directions from place to another place. It serves as a guiding tool that rests in the hands of user. As user enters his destination, system makes use of the google maps and displays direction from the source to the destination.
* If there are multiple paths from source to destination then those paths are to be clearly displayed so that the choice could be made by the user. Eventually making user feel comfortable to move between places.

1. Exploring the city: (priority high)

* As user enters his destiny system provides the facility of exploring more about their place of stay like placed to be visited, its usual temporal conditions, history of the place, interesting facts of the place, kind of currency that is been used, languages spoken, crime rate of the locality etc.
* Individual travelling to the place for the first time may not have the ease of travelling which thereby pulls him away from enjoying his journey. In traveler’s eye system allows him to get the info with a single click.
* System takes his mode of travelling and guide him accordingly throughout his journey.

1. Traffic and weather related information: (priority high)

* Traveler plans his plan and starts his journal accordingly. But traffic and weather of the place keeps varying from time to time.
* It is of good purpose if he knows the situations prevailing in the city as he enters the city. Thus Traffic and Weather are other two prominent issues that are to be considered from the traveler’s perspective.
* After reaching the place, say the person is here to attend some meeting then he will turn keen to know the traffic conditions of the place, so that he could reach the meeting on time.
* Weather allows user to decide the means of transportation to be used. System allows him to know the temperature of the place before he reaches the location.
* System retrieves the weather information using weather API basing on the geo co-ordinates of the user’s current location.

1. Finding food in the proximity: (priority medium)

* System allows user to avail a special feature that is providing the exact location of some source of food say restaurants. By just clicking on the “find food” button user can get the location of the restaurants in the proximity.
* Once user clicks the option system must find the nearest location of the restaurants and display it to the user upon google maps. For this purpose system makes use of the API that locates restaurants using geo co-ordinates of the user.
* User can also get the help of navigating to the selected restaurant.
* Besides providing the locality of the restaurant, system includes a little detail of the kind of food that is available in the restaurant.

1. Social Networking: (priority low)

* System also provides user to update himself user the social networking websites like Facebook, Twitter, Google+ etc.
* For enabling this feature API’s of the networking sites are been used.

**Non-Functional Requirements**:

1. As Emergency issue is been addressed in traveler’s eye, system has to put some threshold on the delay time in notifying the emergency contacts in the database.
2. While intimating the emergency contacts the system must notify with the exact location of the user making use of his current geo-location co-ordinates.
3. Weather of the place must have to be retrieved exactly. System must update the weather information repeatedly, so that he can get himself prepared to face the forthcoming weather conditions.
4. Traffic of the location must also be refreshed at regular intervals of time say every 10 minutes as traffic keeps changing with respect to time.
5. Considering the finding food service, the distance of finding must be set with some value as the bound value say 5-6miles.
6. The number of places has also to be set up with some value may be 10 restaurants in the closest proximity.

**Technological & architectural requirements:**

**Server Operating System:**

Windows operating system.

**Presentation layer:**

HTML5, Java Script, JQuery, CSS

**Web Framework services & languages:**

.net framework, Restful and SOAP web services, C#, IIS server

**Mobile phone requirements:**

Any mobile device that supports jQuery and HTML5 can be used for this project.

**Data base:**

ADO.net

**Business Requirements & Work Flow Analysis:**

In this project, we use Agile Process Management for project development.

**Agile Development process:**

Agile Development process is gaining a great acceptance in software mainstream development community. It can be used for the Development, testing and the management aspects of the project. Agile Process is favorably accepted because it improves the communication among the team members, and can help in achieving the desired goals. Agile Process follows some set of principles which are useful to the project development.

ScrumDo is one of the Agile Software Development model

1. ScrumDo is different from the traditional sequential Approach (Water fall model)
2. It mainly focus on the holistic and flexible development of the product
3. ScrumDo is useful to perform:

* Backlog management
* Iteration Planning
* Story
* Epic Story

**Work Flow Analysis:**

1. Initially a proper planning is made to the project approach.
2. Then the work flow is divided in to Iteration’s.
3. For each Iteration’s stories are created.
4. A brief meeting of approximately 15minutes will be performed daily in order to know about the status of the work done and what we have to do.
5. Continuous testing and Continuous Integration is done in order to rectify the negative outcomes.
6. Assessment of the project progress is done and new requirements will be added by updating the product backlog.
7. Finally, the desired project is delivered.

**Over all System Architecture:**



**Page Layout:**



**System Specification:**

**Primary Services:**

* The important service that the application provides is that it traces the current location of the user and shows the navigation using maps. Using this feature the user can search the nearby restaurants, hotels and make his trip comfortable.
* Once the user uploads his destination he can view the traffic and weather information at the destination. This information helps him to prepare well for his tour. He can choose the right mode of transport for his travel in the destination once he is familiar with these conditions.
* The application provides the emergency contact service. Depending on the destination the application loads the local police station number, hospital number and any emergency number that will be helpful to user at the time of emergency. This is the unique feature of this application.

**EXISTING SERVICES:**

Some of the services in this application uses some predefined API’s.

->Google Maps API:

These are used to give the view of the street or city or any place on the globe. This view can be a map or a satellite image. This api is the heart of the application.

<https://developers.google.com/maps/>

->Yahoo Weather API:

The weather service provided here mainly depends on yahoo weather services .The yahoo weather service provides some API which helps us to get the current weather information of the place we need.

<http://developer.yahoo.com/weather/>

->Google Traffic Layer API :

Google provides traffic layer API .This helps to show the traffic at the requested place dynamically. The application uses this to implement the traffic service.

<https://developers.google.com/maps/documentation/javascript/layers#TrafficLayer>

->Facebook API:

These helps us to use some of the social plugins and Facebook login functionalities into our application.

<https://developers.facebook.com/docs/reference/apis/>

->Twitter API:

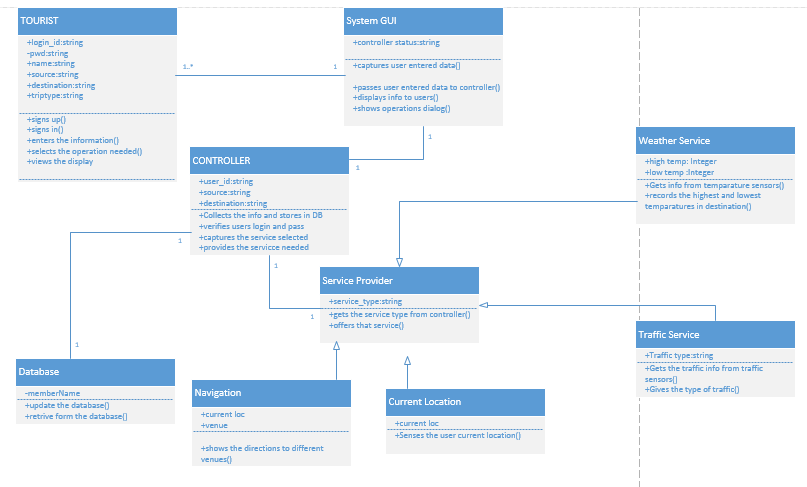
By using these API’s we can login into twitter and can tweet or follow and can provide some publicity to our application.

<https://dev.twitter.com/>

**New Services:**

* Suggests the nearby places of interest to the user according to his tastes and convenience.
* Automatic updating of emergency contacts.
* Automatic location updating to the user specific contacts.
* User can connect with social networking sites like Facebook and twitter.
* The app allows the user to take pictures and also to listen to the music.

**CLASS DIAGRAM FOR OVERALL APPLICATION:**



The class diagram describes the static structure of the application. The class diagram consists of classes, attributes and operations.

**Classes:**

Tourist: This class refers to the user of the application. The user should first sign up with application and them login into it and give his details. And then can use the services of the application.

System GUI: This is the interface between the user and the application. It captures the user’s data and passes it on to the controller .And displays the useful information to user as requested by the controller.

Controller: This is the heart of the whole application. It analyses everything. It updates and verifies database. Contacts the service provider and then brings the service needed into play.

Database: The database is managed here. The storage of information is done here.

Service Provider: This is used to provide services to the application. Each service is a sub class to this super class.

1. Weather service: This gives the weather conditions at the given destination. This gives the highest and lowest temperatures at the given position.
2. Traffic service: This gives the traffic updates at the place required by the user.
3. Current Location: The current location of the user is identified.
4. Navigation: This shows the navigation from to user’s current position to the place he wants to move.

**Relationship-Multiplicity:**

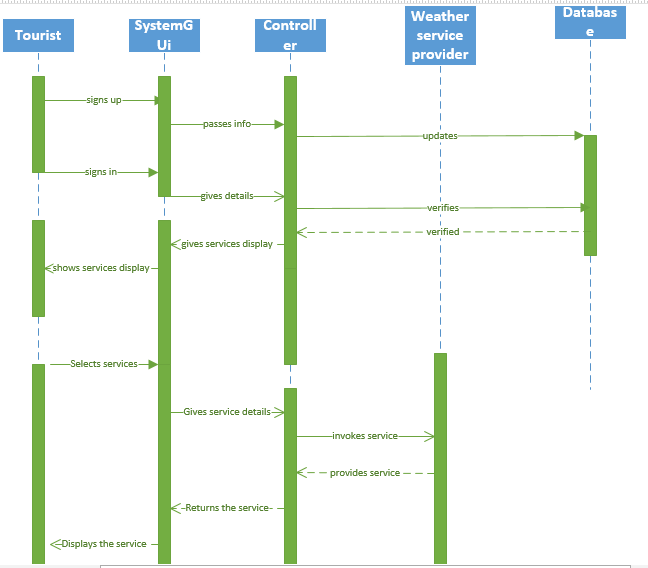
Tourist-System GUI: The System GUI is same and it can be used by multiple user’s .Hence the multiplicity is one or many-one. The relationship is association relationship.

System GUI-Controller: There is a purely one to one relationship between these two. As only single piece of them exists. They have an association relationship.

Controller-Service Provider: These both also have association relationship. And they have one to one multiplicity.

Controller-Database: They do have one to one multiplicity and association relation.

Service provider-weather service, Service provider-traffic service, Service provider-navigation, service provider-current location: These have parent child relationship. The Service provider is the parent class and the individual services are the child classes.

**SEQUENCE DIAGRAM FOR WEATHER SERVICE:**

This is an interaction diagram which shows the interaction of one class with another class with respect to time.

**Steps:**

1. The user first signs up using System GUI.
2. The GUI informs the controller.
3. The controller stores the information in the database.
4. The user now logs in again by taking the help of System GUI.
5. The System GUI informs the controller.
6. The controller verifies the database.
7. The database sends the authentication.
8. The controller asks the System GUI to display the services.
9. The System GUI displays the services to the user.
10. The user selects one of the services.
11. This is informed to the controller with the help of the System GUI.
12. That service is provided by the service provider by contacting its sub classes.
13. The service provider provides the service to the controller.
14. The controller passes it to System GUI.
15. Finally, the required service is displayed to the user.

**SERVICE SPECIFICATION:**

After adding the google maps to the application using google API, The application should detect the current location of the tourist. And then it should be able to show the direction to the place that the tourist requests to move to.

To display the weather services we use yahoo weather API and the application should take the input of the destination and show the current weather over there. .For each destination it should request the yahoo services and bring the required output.

Google provides traffic layer we use this and obtain the traffic information that is just required by the user. We should get the traffic updates just for the required area.

The app will also load the local police station number, hospital number and any such emergency numbers as soon as the user selects the destination. So even in emergency all the user has to do is press one button. Additionally the app also allows some emergency contacts to be defined by the user. The app also has an automatic location update feature which send’s the user’s location information to the user defined emergency contacts at regular intervals of time. The same feature can also automatically send the user’s location to the emergency services in case of an emergency.

The user can connect with social networking sites like Facebook and twitter by just giving his login id and password in our application.

The user can take pictures and post to his friends and he can listen to music of this interest by specifying it while signing up.

Priorities:

|  |  |  |
| --- | --- | --- |
| **SERVICE** | **IMPORTANCE** | **DIFFICULTY** |
| Tracking the current location of user | High | High |
| Navigation | High | High |
| Emergency Contacts | High | High |
| Traffic | High | High |
| Weather | High | High |
| Facebook | Moderate | Low |
| Twitter | Moderate | Low |
| Find Food | Moderate | Moderate |
| Taking pictures and posting | Low | High |
| Music | Low | Moderate |

**Bibliography:**

<https://play.google.com/store/apps/details?id=com.sekond.touristguide.paris>

<https://play.google.com/store/apps/details?id=yokohama_guide.a2ydesigns>

<https://play.google.com/store/apps/details?id=Quinto.app.GreekTouristGuide>

<https://play.google.com/store/apps/details?id=com.mymobilemanchester>

<https://play.google.com/store/apps/details?id=jp.softbank.mb.fa>

<http://www.slideshare.net/DennisDegryse/agile-development-8481635>

<http://research.microsoft.com/apps/pubs/default.aspx?id=56015>

**Team:** Phoenix

**Members:**

Ashok Rudraraju

Sindhu Koneru

Sujitha Onteru

Sudheer Kumar Raganaboina