PavanKumar Bollaram(class id:07)

 Preetham Kumar Danaboina(class id:11)

Lakshmi Priyanka(class id:22)

Varaprasad jaggu (class id:49)

Project Increment Plan1

GOEASY

**Existing services/API:**

1. http://api.eventful.com/rest/events/search?[...](http://api.eventful.com/docs/auth)&keywords=books&location=San+Diego&date=Future

This API is used to get the live events happening across a specific location where the location can be retrieved from location and geo sensor of android mobile device

1. <https://data.cityofchicago.org/resource/alternative-fuel-locations.json?>

This API also used to get the details of the location or city

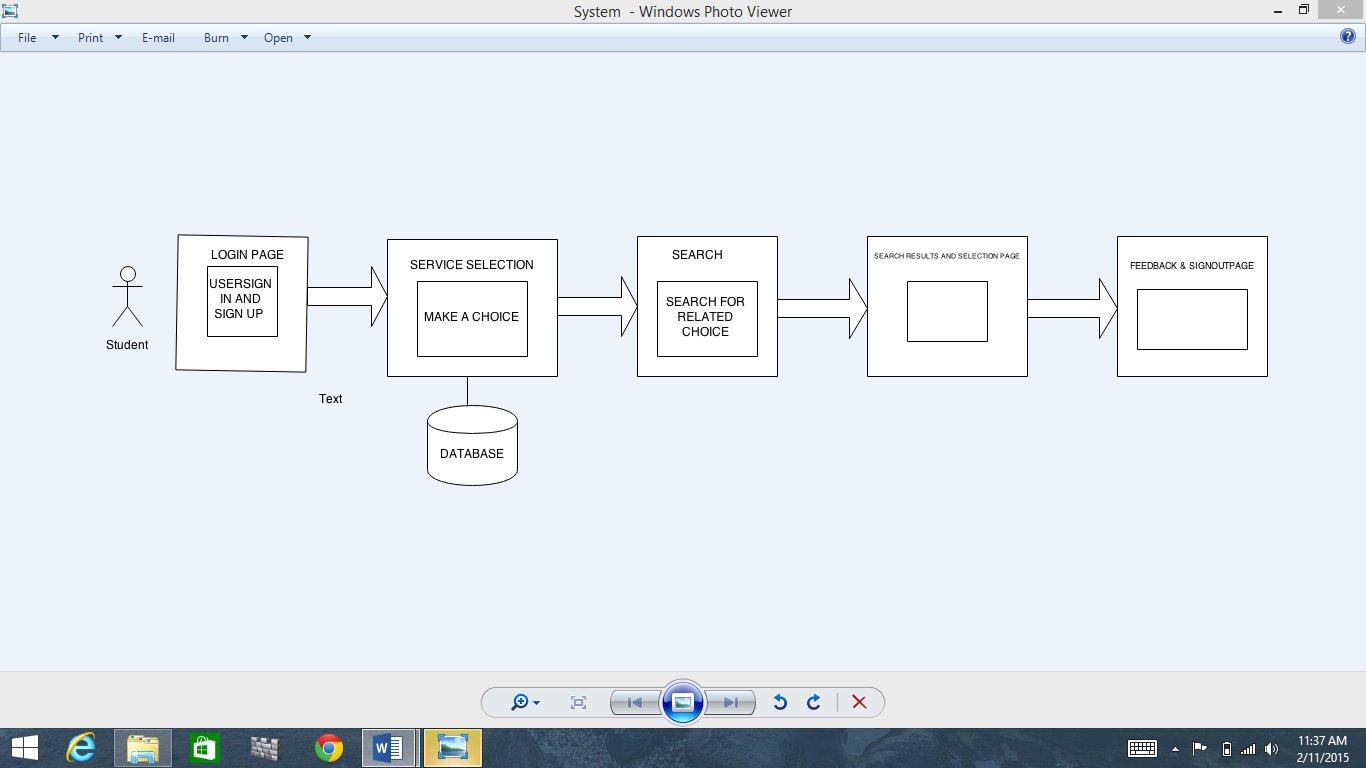
1. https://maps.googleapis.com/maps/api/js?v=3.exp&sensor=true

This API is used for location tagging and navigation purpose which can be used by end user in our application.

**Detail Design of Services:**

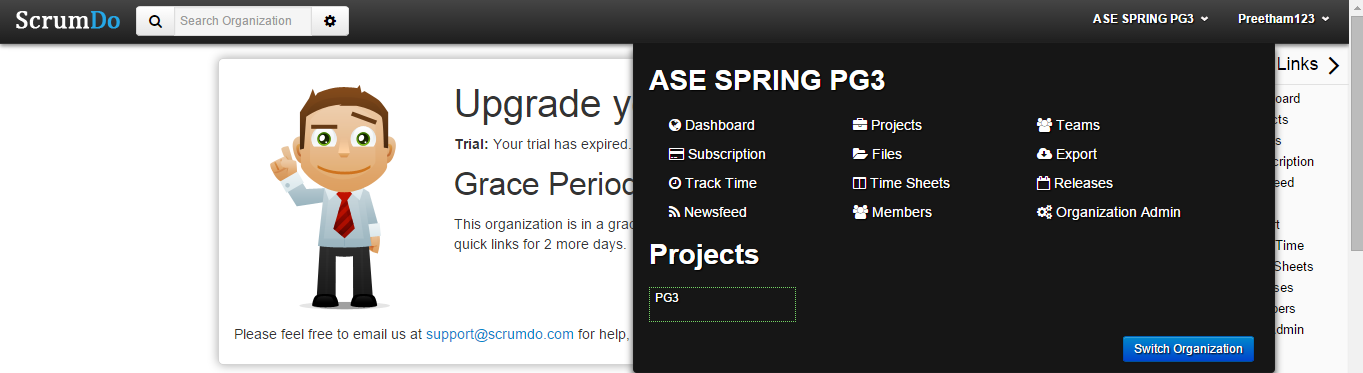
Go Easy Web services:

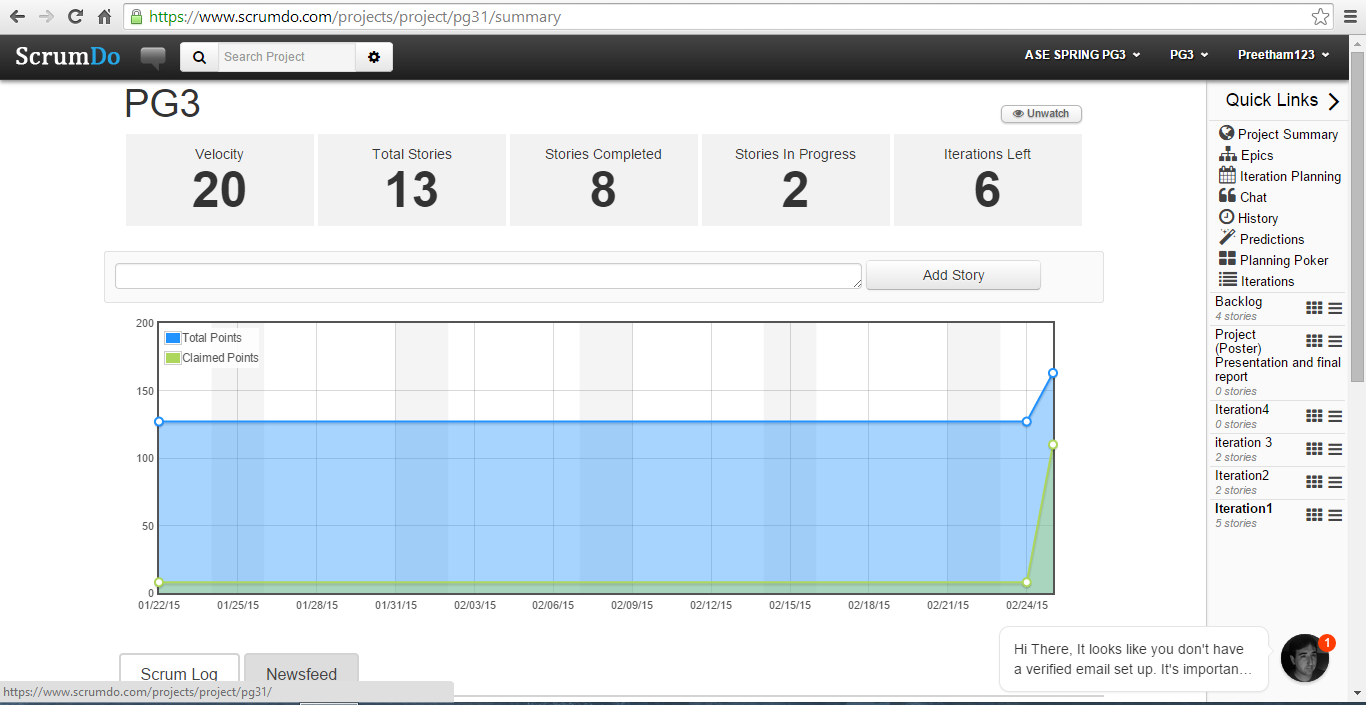
We are developing a customized web service which includes all the required data at one place which going to ease the way of searching for the events by end user.

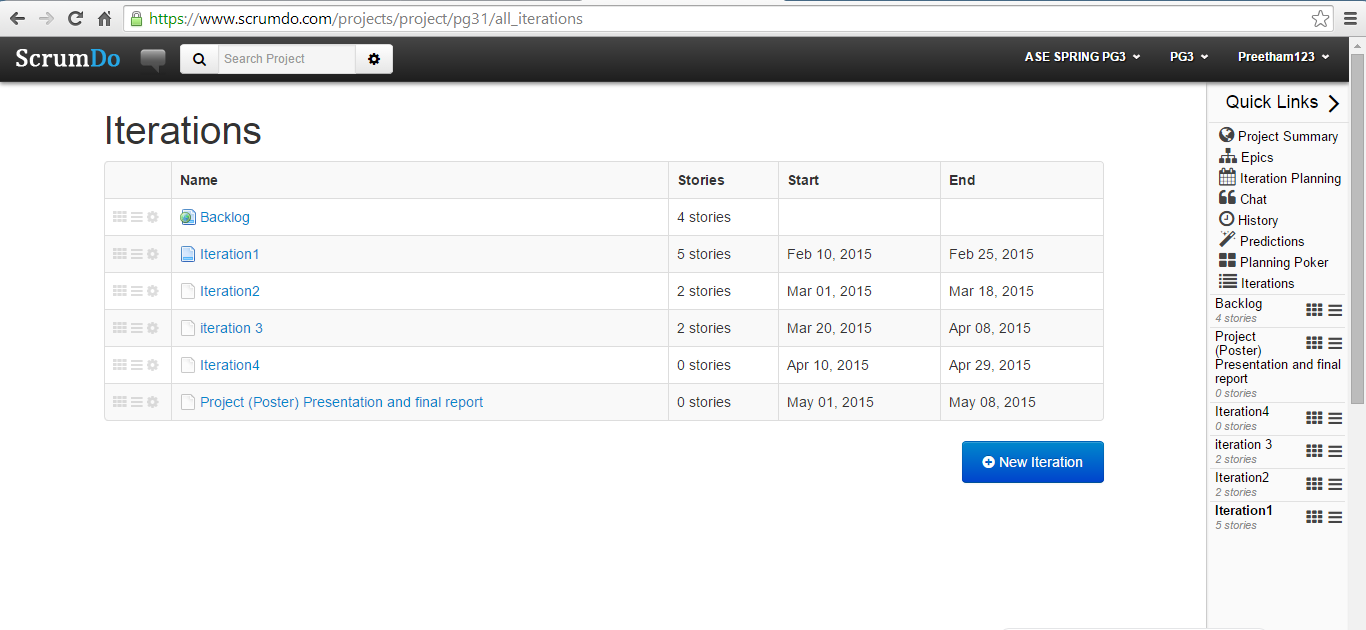


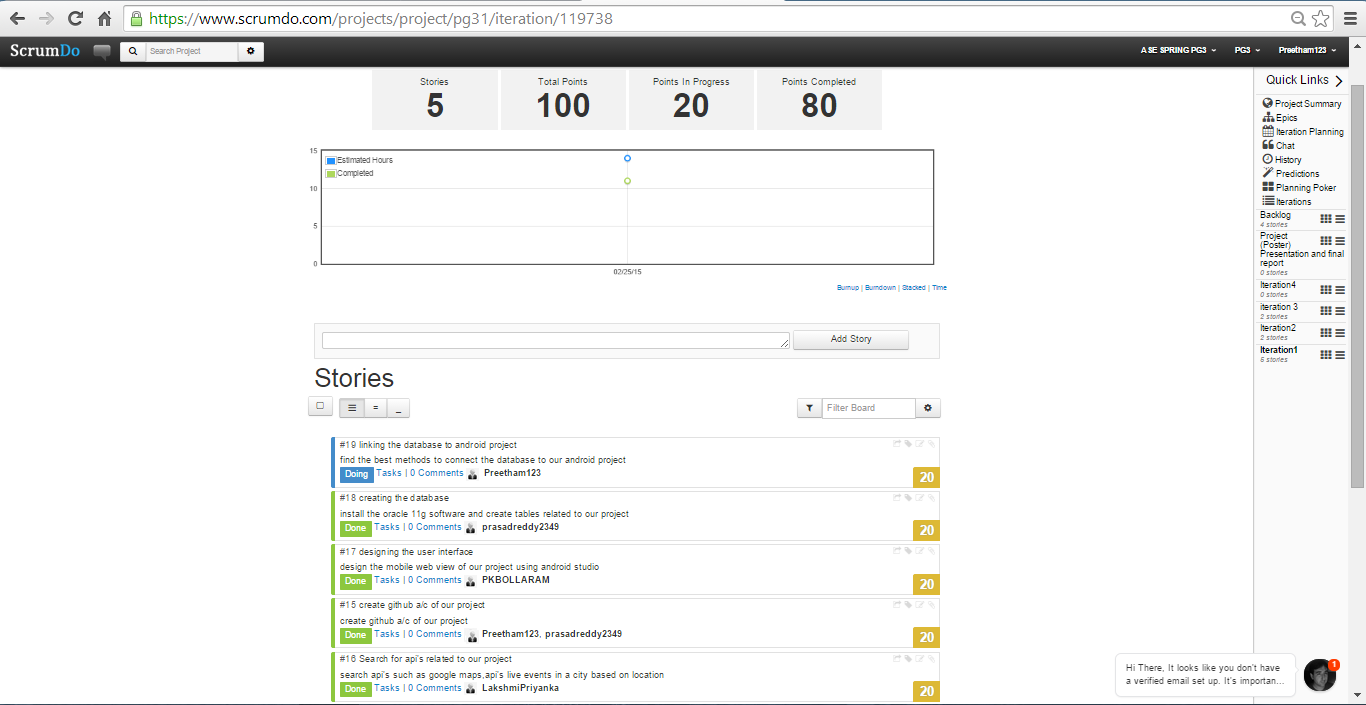
This architecture explains flow and behavioral description of each layer starting with Login page of our application to the data layer. The application mainly consists of three section which includes entertainment related events, university related events and transportation related events. We are using publically available API to connect to their end point and get the response (usually JSON).

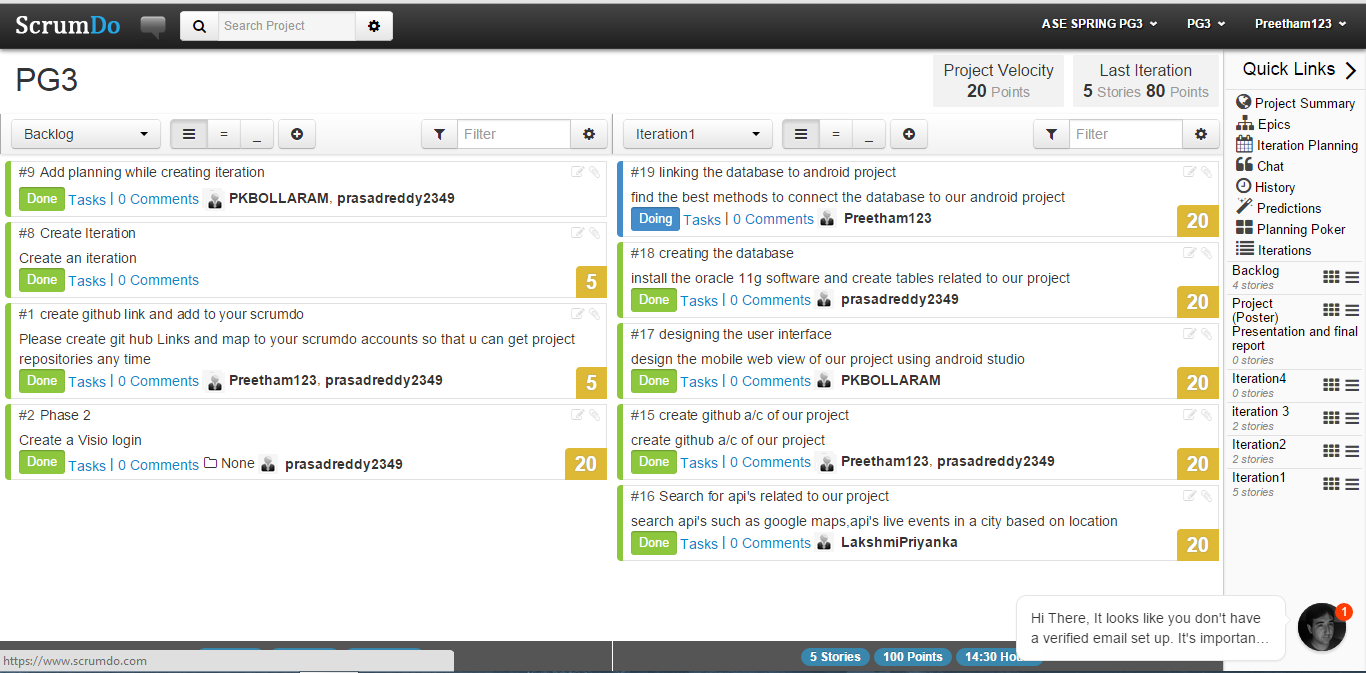
**ScrumDO and stories**:

****

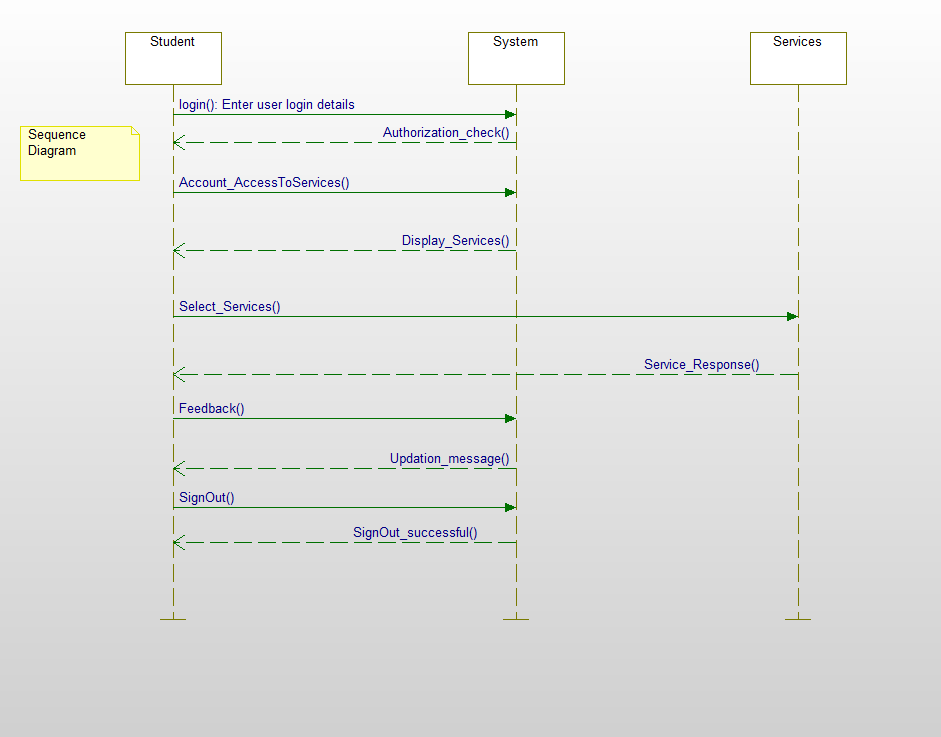
****

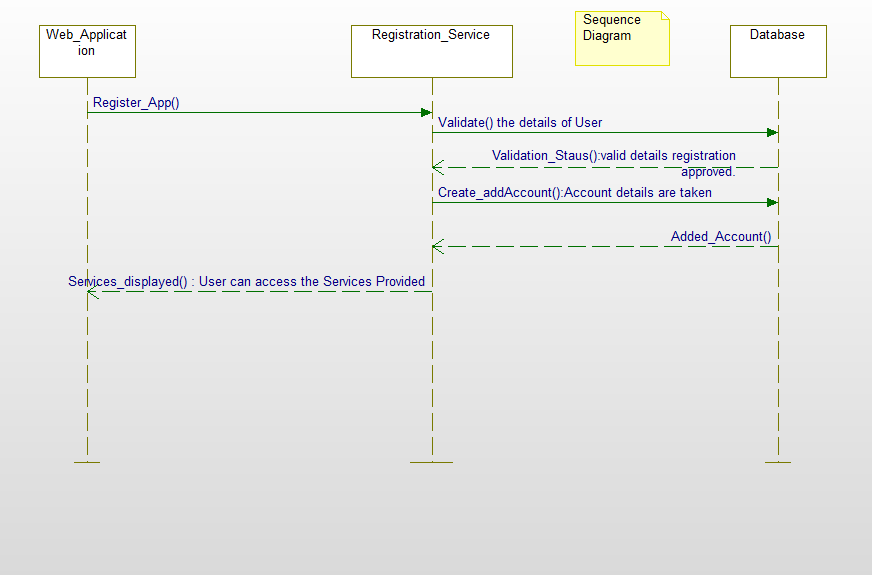
****

****

****

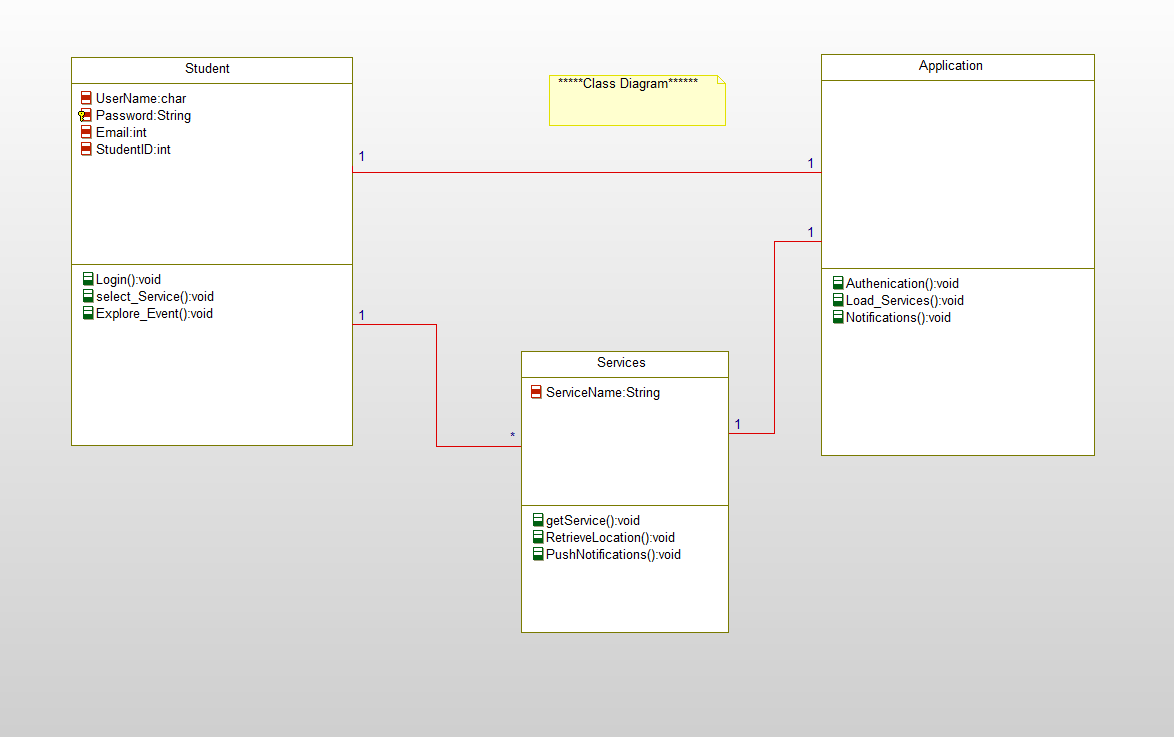
**Sequence Diagram:**

****

****

This sequence diagram explains an abstraction level of system implementation with required services and flow of each service. User can register to events using Registration services offered by application which gets the users data and persists to database. This very same data can be fetched from the data layer using display services call and dashboard is updated with these results. Along with the userful live events data a small level navigation part is also going to be implemented in this application.

**Class Diagram:**



**Design of Mobile Client Interface**

Web services

DashBoard

Valid user?

Login screen

Data layer(DB)

**Design of Unit test cases:**

To be implemented in following increment.

**Implementation**

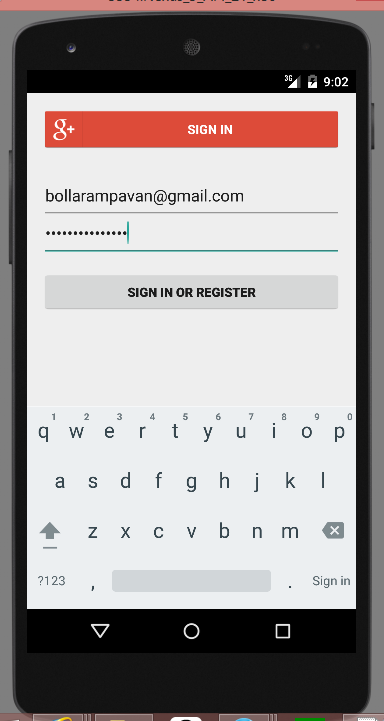
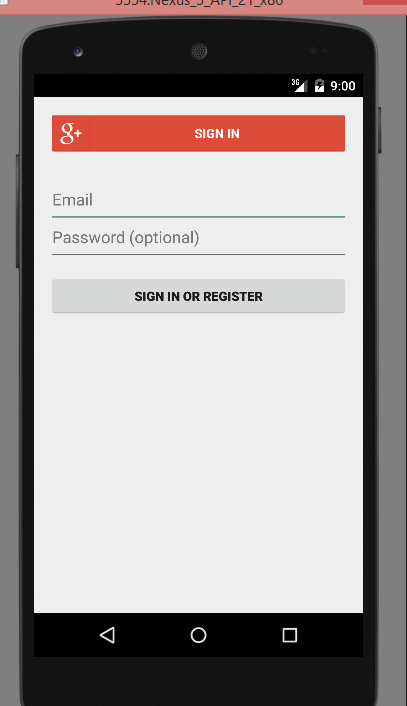
**Implementation of REST services**

As of now we are using three of available web service API’s to get the data

* http://api.eventful.com/rest/events/search?[...](http://api.eventful.com/docs/auth)&keywords=books&location=San+Diego&date=Future
* <https://data.cityofchicago.org/resource/alternative-fuel-locations.json?>
* <https://maps.googleapis.com/maps/api/js?v=3.exp&sensor=true>

Customized REST service needs to implemented for enhancement of our application which will be further detailed next increment.

**Implementation of User interface(Mobile Interface)**

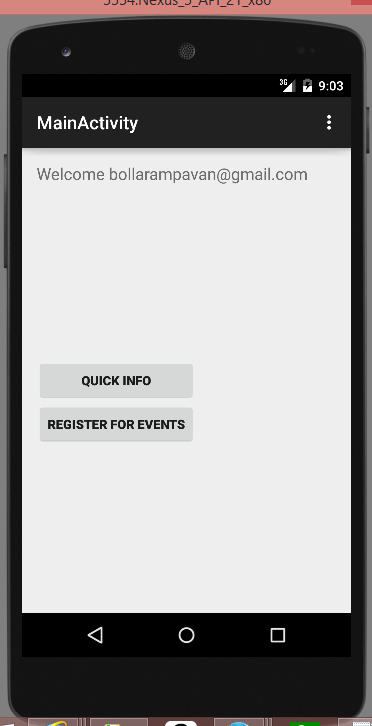


Login screen

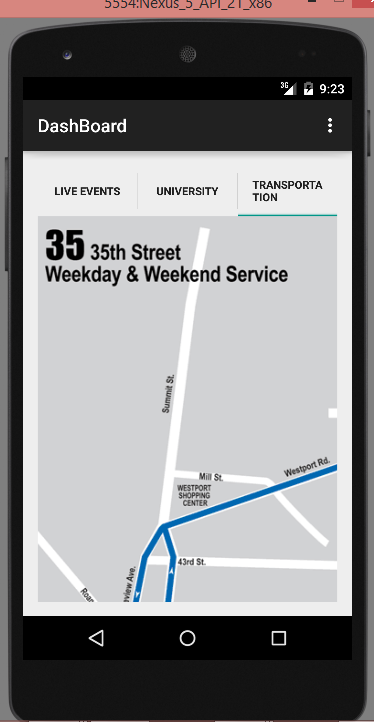
Login to system uses existing google+ sign in services so that end user can easily login with his/her gmail id’s which reduces the use of separate database for our application .This login screen contains two separate parts which are

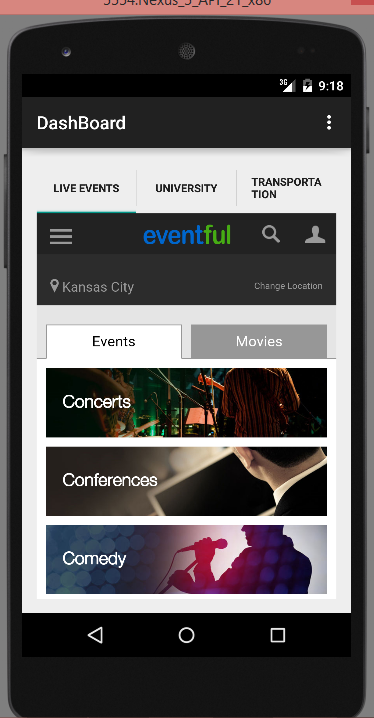
SIGN IN : This button used incase user is not registered with our application

EMAIL and PASSWORD fields which takes input from user and validates the user details. Once validated user can login to application and can check his dashboard



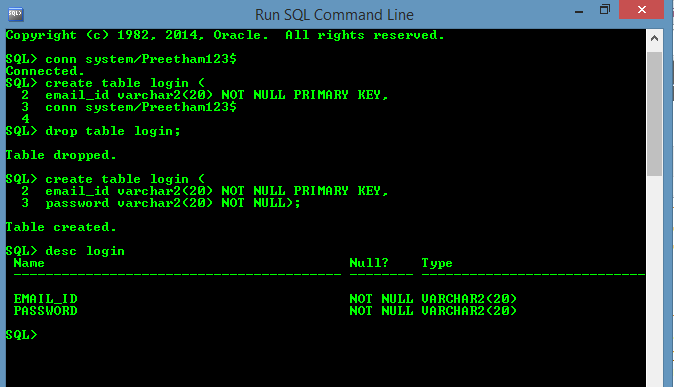
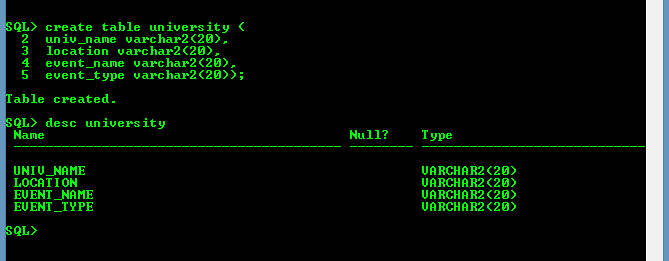
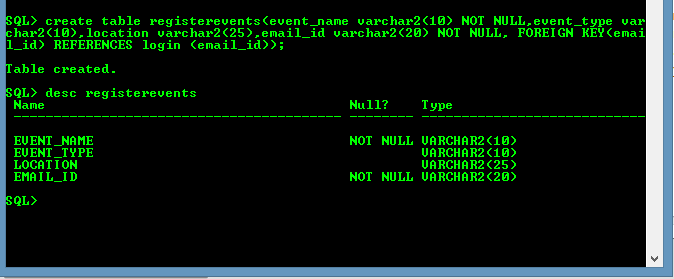
Welcome Page – This page displays the user with which email id he/she login to application and provides **QuickInfo** and **Register** for Events buttons. If we click on quick info user is navigated to dashboard screen with useful info about live events and all.

**Dashboard screen** This is dashboard screen which will be displayed when user login to application. This is complete mashup kind of application where end user can easily check the required tab.



We simplified GUI with simple tab host concept so that end user can easily navigate between Live-events, University and transportation tabs. The content is displayed according to the selection of tab.

**Database Design implementation**



**Implementation of test cases**:

Yet to be implemented.

**Testing**

Will be detailed in next increment plan.

**Deployment**

**ScrumDo**

<https://www.scrumdo.com/projects/project/pg31/summary>

**GitHub**

**Source code-**[**https://github.com/pavankumar-b/ASEspringSem/Increment1**](https://github.com/pavankumar-b/ASEspringSem/Increment1)

**Documentation-**[**https://github.com/pavankumar-b/ASEspringSem/IncrementDoc**](https://github.com/pavankumar-b/ASEspringSem/IncrementDoc)

**Project Management**

**Implementation status Report**

**Work completed:**

* **Description**

Android Login screen is implemented

Successfully implemented Dashboard screen with tabhost interface

Login screen with google sign in option

Separate tab screen for each functionality

Database table creation

* **Responsibility**

Task1: Android Login screen and implementation of tab interface / Pavankumar Bollaram

Task2: Web API’s(Google SignIn) implementation/Lakshmi Priyanka

Task3: Database creation/ Varaprasad

Task4:Dashboard screen design and navigation/ Preetham kumar

* **Time Taken**

200 Man hours

* **Contribution**

PavanKumar Bollaram (25%)

Preetham kumar(25%)

Lakshmi Priyanka(25%)

Varaprasad(25%)

**Work to be completed**

Task1: Custom Web API design and Implementation (pavankumar Bollaram)

Task2: Database integration with android interface (varaprasad/Preetham kumar)

Task3: University API implementation and integration (Lakshmi Priyanka)

Time to be taken (400 Man hours)

**Issues/Concerns**

Facing issue while establishing connection with database

Incomplete information about web API’s and their documentation

Android studio performance is very poor and time taking