

# METU CENG491 2015 FALL

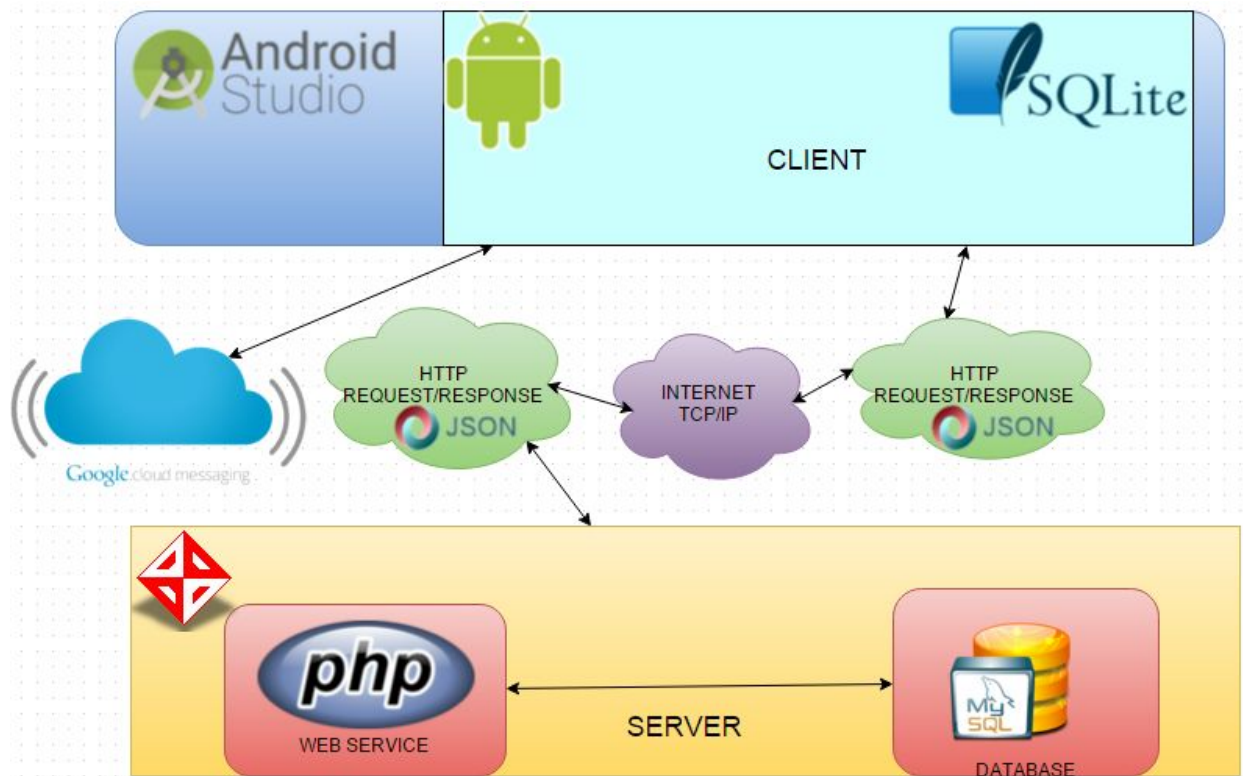
## *START-UP DOCUMENT*

<G23P45>

**Group Name:** <Code Whisperers>

**Project Name:** <NERS>

## 1. System Architecture



- Overall system architecture diagram is above. The diagram also shows the interactions and components and their dependencies. On the other hand, later on there may be new components to be added if the new component make easier our development process. As it is seen in the diagram our model is client-server based model. Clients are android operating systems in our system. Server is mysql database server.
- **Android Studio**

Android Studio is an integrated development environment (IDE) for developing for the Android platform. Our project will be developed in Android Studio IDE.

### Sqlite

SQLite is a software library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine. It is widely used in android development process. We will use sqlite to keep the received messages, photos in client side. This thought based on to keep the server side database minimized. When every user gets data which is needed, the server side data will be deleted or revised.

### PHP Web Service

In our system PHP Web service interacts with clients and take requests from clients. Http protocol and JSON serialization is used in this transfer process. Clients send HTTP request to PHP web service and PHP web service sends data to clients with JSON serialization. PHP web service is used in SQL injection to database.

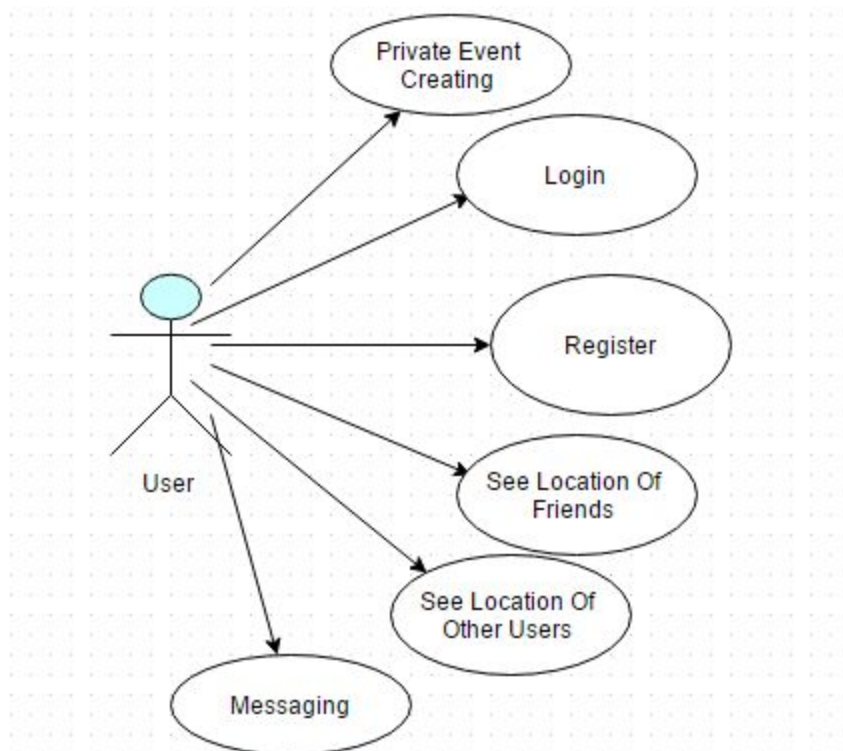
## Google Cloud Messaging

In our messaging system google cloud messaging will be used. Google Cloud Messaging provides a simple, lightweight mechanism that servers can use to tell mobile applications to contact the server directly, to fetch updated application or user data. The service handles all aspects of enqueuing of messages and delivery to the target application running on the target device.

Instead of developing our messaging system in our server, it will be more efficient to use google cloud messaging.

## MySQL Database

MySQL database will be used in our application. Users, events etc. will be kept in our mysql database.



*User Interaction Model can be seen as above.*

Users will access our system by using Android Application. Our system will run on a MySQL Server. Users interact with our system through Android Application. Creating an event, getting info about nearby events, logging to the system, registering into system is done with the connection between Android Application and MySQL Server.

When unregistered users wants to enter our system, our Android Java Classes generates a unique User Id for that new user and sends request via AsyncTask Classes to the Mysql Server and register that user to the system via running some PHP files under the server.

When registered users enters the system,for different operations that user want to make, we will have different classes to handle these operations such creating an event.As mentioned above,all we will make on this project is exactly that we should establish the connection between Php Server,Mysql database and Android Application correctly and interactive as much as possible.I will try to explain how the user interactions in the system will be implemented.

### Location - Based Messaging System Interaction

Location -Based Messaging System is the system that interacts with Php Server and Mysql Database.This system enables two users to see each others' locations and message simultaneously.Also, users' locations change with using GPS Service provided by Google Services also on the map.We will be using Google Cloud Messaging System for our messaging system and messages between the users will be stored on the users' smart phones .For that reason , we will be using the Android Cache System.For messaging part,we will not store the messages between users on our databases thank to the Android Cache Systems.Locations of the users updating within a time period (to be determined later).The locations will be updated via Java Classes extended Thread and sended to the PHP server and PHP files run some SQL Queries and updates the database with those locations.

## 2. Tentative Time Plan

TaskID	Short Name	Description
S_I	System Initialize	System environment will be formed in this period.Android Studio, MySQL server and SQLite will be initialized.System database will be created with relations.
M_R	Market Research	Market research will be done in this task.Similar projects will be analyzed.System of the project will be considered again.
U	User	User class will be implemented with required attributes such as name, username, password etc. in this task.
U_R_L	User Registration and Login	UserRegistration and UserLogin classes will be implemented in this task.In UserRegistration class, user will be recorded into the database In UserLogin class, location of user will be taken and this location will be updated into the database.

S_L_O_U	Show Location of Friends	ShowLocationFriends class will be implemented in this task. In this class, locations of user's friends will be taken from database periodically.
S_L_O	Show Location of Others	ShowLocationOthers class will be implemented in this task. In this class, locations of others will be taken from database periodically.
U_M	User Messaging	UserMessaging class will be able to communicate with the Google Cloud Messaging for this task.
P_N_R	Phone Number Registration	A subsystem which takes data from telephone operator and identify users' phone number and their smart phones will be integrated to NERS.
U_S	User Storage	User class and User table will be revised according to users' phone number. Unique identifier for users will be their personal phone numbers.
D_R	Database Revision	In this task, Database will be revised by adding Event table and necessary connection will be handled with User table in database as preparation for second part.
E	Event	Event class will be implemented with necessary attributes such as Unique Identifier, Organizer, Type etc. in this task.
P_E_C_U	Private Event Creation by User	PrivateEventCreationUser class will be implemented in this period. With this class, users may create private events by giving required informations such as place, time, description etc. These events may only be seen to the friends of users who create event in nearby environment.

- Construct your time plan as a simplified Gantt chart, as shown in the following table.

	Iteration1	Iteration2	Iteration3
S_I	■	■	■
M_R	■	■	■
U	■	■	■
U_R_L	■	■	■
S_L_O_U		■	■
S_L_O		■	■

U_M		■	■
P_N_R		■	■
U_S			■
D_R			■
E			■
P_E_C_U			■

### 3. Deliverables

Deliverable	Description	When? (Sprint#)
D1	Market research and system initialization will be done in this deliverable. We can inform people about marketing of this project and system components such as Android Studio, MySQL server and SQLite.	Sprint 1
D2	In this deliverable, users will see their locations and the other users on map. Locations of users and the other users will be updated periodically into the database.	Sprint 2
D3	In this deliverable, users may chat with the other users on map during seeing each others' locations.	Sprint 2
D4	In this deliverable, users may create an event which can be seen by friends of user who create event.	Sprint 3

### 4. Workload Distribution

	Sprint - I	Sprint - II	Sprint - III
Mehmet Gençol	S_I	S_L_O_U	U_S

Oğuz Artıran	M_R	S_L_O	D_R
Ahmet Melih Gedikli	U	U_M	E
Mustafa Murat Coşkun	U_R_L	P_N_R	P_E_C_U