

INFORMATION MANAGEMENT SYSTEM



By
Muhammad Shoaib
Usman Rajput

Submitted to the Faculty of Computer Software Engineering, National University of Sciences and Technology, Islamabad in partial fulfillment for the requirements of a
B.E. Degree in Computer Software Engineering

MAY 2016

CERTIFICATE OF CORRECTNESS AND APPROVAL

Certified that work contained in this thesis “TMS” carried out by Muhammad Shoaib and Usman Rajput, under the supervision of Dr. Hamaad Afzal for partial fulfillment of Degree of Bachelor of Software Engineering is correct and approved.

Approved By

(Dr. Hamaad Afzal)

Department of Computer Software Engineering

Military College of Signals, Rawalpindi

Dated: _____

ABSTRACT

"IMS" project has been developed keeping in view the importance of accessing data and how it is presented to users working on it. The system is designed to represent Public related information on maps in geographical form . The system provide opportunity to access basic information at gross root level i.e villages, UCs and Districts. Information is gathered by the field users, checked at unit level and then passed to district and Administration headquarter. At each level data is checked and corrected by the authorized user if necessary. Data is displayed in graphical, tabular and geographical form.

Many government and non government organization are working at gross root level. They need location wise data for their activities. Manually handled data is costly and causes delay. The organizations need to represent data in order to cover more areas. There should be a platform that provides the required data in easy and compact form. So there is a need of a web application which will minimize the cost and time of data gathering.

"IMS" makes the job more prominent and easy by making the geographical representation of data on map with the help of Google API in tabular form. Moreover information will also be centrally available in graphs and tables.

COPYRIGHT NOTICE

We very solemnly declare that the work presented herewith is the result of sole effort of our group, comprising of Muhammad Shoaib, Usman Rajput and is free of any kind of plagiarism in part or whole. We also declare that the dissertation has never been submitted previously in part or whole in support of another award or qualification either at this institution or elsewhere.

DEDICATION

To our respected teachers whose kind guidance and unfailing support made this mammoth task easy for us and to our very dear parents whose unceasing prayers gave us strength and courage to complete the work of this magnitude.

ACKNOWLEDGEMENTS

We are very humbly grateful to Almighty Allah for bestowing us with the strength and resolve to undertake and complete the project.

We owe a special debt of gratitude to our supervisor, Dr. Hamaad Afzal for the continuous supervision, motivation and support provided to us and for their continuous and valuable suggestions, guidance, and instructions from time to time right through the project.

We would also like to acknowledge and thank the benign faculty members of the department of Computer Software Engineering for their continuous support. It was their help and guidance which helped us complete the project in due time.

Table of Contents

CHAPTER 1	1
1. INTRODUCTION	1
1.1. MOTIVATION	1
1.2. PROJECT SCOPE	2
1.2.1. Project Vision	2
1.2.2. Project Objective	2
1.3. DELIVERABLES	2
1.4. DOCUMENT ORGANIZATION	3
1.5. SUMMARY	3
CHAPTER 2	4
2. LITERATURE REVIEW	4
2.1. INTRODUCTION	4
2.2. PROBLEM DOMAIN	4
2.3. RELATED WORK	4
2.3.1. OAS	4
2.3.2. Digital Mapping System	4
2.4. LIMITATIONS	5
2.5. TECHNOLOGICAL & SOFTWARE REQUIREMENTS	5
2.6. HARDWARE REQUIREMENTS	5
2.7. SUMMARY	6
CHAPTER 3	7
3. SOFTWARE REQUIREMENT SPECIFICATIONS	7
3.1. SOFTWARE INTERFACES	7
3.2. PRODUCT FEATURES	7
3.2.1. Resource repository of Users	8
3.2.2. Activity Handling	8
3.2.3. Login/Access Rights	9
3.2.4. Report generation	9
3.2.5. Maps	10

3.3. OTHER NONFUNCTIONAL REQUIREMENTS -----	10
3.3.1. Performance -----	10
3.3.2. Safety -----	10
3.3.3. Security -----	10
3.3.4. Software Quality Attributes -----	11
3.4. USER CLASSES AND CHARACTERISTICS-----	12
3.4.1. Admin User-----	13
3.4.2. End User -----	13
3.5. ASSUMPTIONS AND DEPENDENCIES -----	13
3.6. SUMMARY -----	13
CHAPTER 4 -----	15
4. SYSTEM DESIGN SPECIFICATIONS -----	15
4.1. OVERVIEW OF THE MODULES-----	15
4.1.1. Description of the modules-----	15
4.2. STRUCTURE AND RELATIONSHIPS-----	16
4.2.1. Overall Structure of the system-----	16
4.3. UML DIAGRAMS -----	18
4.3.1. Use Case Diagram-----	19
4.3.2. Class Diagram with description -----	21
4.3.3. Sequence diagrams -----	23
4.3.4. Activity diagrams-----	23
4.4. SUMMARY -----	23
CHAPTER 5 -----	24
5. IMPLEMENTATION DETAILS -----	24
5.1. INTRODUCTION-----	24
5.2. MVC IMPLEMENTATION -----	24
5.3. MAP IMPLEMENTATION -----	24
5.4. USER INTERFACE -----	24
5.5. USER MANAGEMENT MODULE-----	25
5.6. SUMMARY -----	25

CHAPTER 6	-----26
6. TESTING	-----26
6.1. INTRODUCTION	-----26
6.2. TESTING LEVELS	-----26
6.2.1. Unit Testing	-----26
6.2.2. Integration Testing	-----26
6.2.3. System Testing	-----27
6.3. SUMMARY	-----27
CHAPTER 7	-----28
7. CONCLUSION AND FUTURE WORK	-----28
7.1. CONCLUSION	-----28
7.2. FUTURE WORK	-----28
APPENDIX A	-----29
USER MANUAL	-----29
APPENDIX B	-----35
USE CASES DESCRIPTION	-----35
APPENDIX C	-----40
SEQUENCE DIAGRAMS	-----40
APPENDIX D	-----46
ACTIVITY DIAGRAMS	-----46
APPENDIX E	-----52
UNIT TESTING – TEST CASES	-----52
APPENDIX F	-----54
INTEGRATION TESTING – TEST CASES	-----54
APPENDIX G	-----56
SYSTEM TESTING – TEST CASES	-----56
GLOSSARY	-----58

List of Figures

Figure 1: Overview of the modules	15
Figure 2: description of modules	16
Figure 3: Overall Structure of the System	17
Figure 4: Use Case Diagram	19
Figure 5: Class Diagram	21
Figure 6: User Manual -Login	29
Figure 7: User Manual - Dashboard.....	30
Figure 8: User Manual- User Profiling	30
Figure 9: User Manual- Overall Report	31
Figure 10: User Manual -DM Tools	32
Figure 11: User Manual - Admin.....	33
Figure 12: User Manual - Maps	34
Figure 13: User Manual - User	34
Figure 14: Manage User Sequence	40
Figure 15: Login Sequence	41
Figure 16: Create Form Sequence.....	41
Figure 17: Quality assurance Sequence	42
Figure 18: Search Sequence	43
Figure 19: Generate report Sequence.....	43
Figure 20: Data Management Sequence.....	45
Figure 21: dashboard Activity Sequence	45
Figure 22: Manage User Activity	46
Figure 23: Create Form Activity.....	47
Figure 24: dashboard Activity	48
Figure 25: Login Activity	49
Figure 26: Report Activity	50
Figure 27: Search Activity	52

List of Tables

Table 1: Project Vision	2
Table 2: Hardware Requirements client	5
Table 3: Hardware Requirements Server	5
Table 4: Recommended Hardware Requirements client	6
Table 5: Recommended Hardware Requirements Server	6
Table 6: Software Interface Requirements client.....	7
Table 7: Software Interface Requirements Server	7
Table 8: Software Interface Requirements database server	7
Table 9: Software Development Apps Used.....	7
Table 10: Use Case Description - Manage User	35
Table 11: Use case Description - Login Feature.....	35
Table 12: Use case Description- Edit/view profile	36
Table 13: Use case Description - Publish Activity	37
Table 14: Use case Discription - Generate Report.....	37
Table 15: Use case Discription - Map Service.....	38
Table 16: Use case Discription-Serach Activity	39
Table 17: Unit Testing- Login Feature.....	52
Table 18: Unit Testing- Plan Activity Module.....	53
Table 19: Integration Testing- Administrator Module.....	54
Table 20: Integration Testing- User Module.....	54
Table 21: Integration Testing- Activity Form Module.....	55
Table 22: System Testing- Login.....	56
Table 23: System Testing- Create User.....	56
Table 24: System Testing- Submit input Activity Form.....	57
Table 2: Glossary.....	58

1. Introduction

The key purpose of technology is to serve mankind. Information plays key role to fulfill the purpose. Rapid Access and presentation of information gets importance in this regard. IMS project is basically developed keeping in view the importance of accessing data and how it is presented to users working on it. The system is designed to represent Public related information on maps in geographical form . The system provide opportunity to access basic information at gross root level i.e villages, UCs and Districts on maps. Information is gathered, checked at unit level and then passed to district and Administration headquarter. At each level data is checked and corrected if necessary.

Many government and non government organization are working at gross root level. They need location wise data for their activities. Manually handled data is costly and causes delay. The organizations need to represent data in order to cover more and more areas. There should be a platform that provides the required data in easy and compact form. So there is a need of a web application which will minimize the cost and time of data gathering. The geographical representation of data on map with the help of Google API in tabular form make the job more prominent and easy, Moreover information will also be centrally available.

1.1. Motivation

IMS is intended to facilitate masses to streamline public benefitting activities with a common platform. The project has multidimensional effects on the way of addressing issues in the society. Industry can refer IMS for carrying out Corporate Social Responsibility Projects. Society in general will also be able to highlight core public issues and help the needy in their respective domains.

The app will take care of the overall data handling mechanism. Field users identify Public related issues. The same data is dealt by authorized users at respective levels and corresponding activities are carried out.

1.2. Project Scope

The basic goal is to insure authorized user access in order to help and maintain the smooth running of the business by providing information on the firm's data (such as accounting figures). Employees from different levels will then evaluates this information. To Identify critical issues (violence, education, health, family issues) and represent in a geographical form. Survey form is created and Data Entry Operator enters data by IMS app. Reports will be generated in tabular, graphical and geographical form.

1.2.1. Project Vision

For	Government use and the Users, who wish to offer public services and the seekers who need them.
What	The IMS App is a web-based app that provides the users with interface to publish the community related activities and services .while allowing the users to search for specific service based activities on the requisite filters.
The	Public service web app IMS.
Is	mainly categorized as a well structured platform for public services planned.
Those	seekers do not have to keep track of the available public services manually.
Unlike	the existing practice of conventional information handling used.
Our Product	can not only save time but also provide a central platform to carry out public service activities.

Table 1: Project Vision

1.2.2. Project Objective

The objective of this project is to develop an web app that chains the communication process among various users. The app will populate the data onto the database server with respect to personal credentials of each user.

1.3. Deliverables

1. Web Application
2. Documentation

1.4. Document Organization

This document provides basic knowledge about IMS. Initially the description of project has been given, the next chapter discusses related software products and their functionalities and how IMS is different from them. Requirement specifications have been covered in chapter 3 and 4, while the Design specifications are discussed in chapter 5. Implementation of the system is described in chapter 6 and further chapters ponders upon the analysis of the software, testing techniques employed and suggestion about the future work for enhancing the functionalities and capabilities of the system.

1.5. Summary

IMS is a community service mobile application which is intended to carry out activity based service of public. This chapter is an introduction to IMS vis-à-vis its scope, objectives, and motivation for developing this software.

Chapter 2

2. Literature Review

2.1. Introduction

IMS is an activity based public service project. The main purpose of the project is to meet the needs of multiuser environment in such manner that data flow from field to Headquarter and vice versa can be carried out in an efficient way and represented on maps in required form.

2.2. Problem Domain

IMS is a web app which deals with information management that includes a large scale public issues to be documented. The problem of constantly looking for the desired public issues and services, till it is found, is catered for all by the IMS, enabling the user to carry out required activities.

2.3. Related Work

The related work helps in understanding the existing solution of the problem domain with a better insight. In connection with reviewing the related work, following website have been reviewed during the course of studying the related work:

2.3.1. OAS Pak Army

Office Automation System: A structured portal based information communication system currently working in army.

2.3.2. Digital Mapping System Pak Army

The unit which deals with managing and representing data on maps.

2.3.3. Online FIRE System KPK

This is an online web based complaint system prepared by KPK police for public ease, to launch FIR.

2.3.4. Trip Discovery.Com

A website offering online hotel booking and trip facilities to tourist in Islamabad.

2.4. Limitations

1. data accuracy is dependent upon the field user's work.
2. The contents of the application will be in English language only.
3. The server will be unavailable in case of maintenance and testing issues.
4. backup server configuration is provided.
5. The searching will be Service Title – based.
6. Map Location accuracy dependent on Google API's.

2.5. Technological& Software Requirements

The front end interface of the IMS will be developed based on MVC structure and the backend part involving database management will be developed on MYSQL server. The network connection will involve, HTTP/HTTPS Protocol.

2.6. Hardware Requirements

2.6.1. Minimum Requirements:

Client Side			
Internet Connection	Processor	RAM	Disk space
1 MB	Intel Pentium (iv) 1.5 MHz	1 GB	80 GB

Table 2: Hardware requirements client

Server Side			
Internet Connection	Processor	RAM	Disk space
4 MB	Intel Core 2 Duo 2.0 MHz	4 GB	1 TB

Table 3: Hardware requirements server

2.6.2. Recommended Requirements:

Client Side			
Internet Connection	Processor	RAM	Disk space
4 MB	Intel Core i3 2.5 MHz	4 GB	200 GB

Table 4: Recommended Hard ware requirements client

Server Side			
Internet Connection	Processor	RAM	Disk space
16 MB	Dell PowerEdge R715	16 GB DDR3	5 TB

Table 5: Recommended Hard ware requirements server

2.7. Summary

This chapter introduces us to the features and limitations of IMS. The section further highlights some of the shortcomings and mentioning about the technological, hardware requirements of the system.

3. Software Requirement Specifications

3.1. Software Interfaces:-

Client Side	
Operating System	Browser
Windows XP, Vista, 7, 8, 10, Linux	Chrome, Firefox, Opera, Internet Explorer

Table 6: Soft ware interface requirements client

Web Server	
Operating System	Server
Linux (Ubuntu)	Apache v2.2+ or WAMP

Table 7: soft ware interface requirements web server

Data Base Server	
Operating System	DBMS
Linux (Ubuntu)	MySQL

Table 8: soft ware interface requirements data base server

Development End	
Operating System	Tools
Linux (Ubuntu), Windows 7, 8	CodeIgniter, JS library, Google API V 3, Dreamweaver, QGIS

Table 9: software development Apps used

3.2. Product Features

IMS must provide functions to create community service and track its status. It should also provide functionalities for allowing the seeker and donor to publish their

requirement, while getting updates about the status of other ongoing community services. Following are the functions that IMS should provide:

3.2.1. Resource repository of users

3.2.1.1. Description and Priority

IMS will allow the maintenance of a complete database of resources of different users. For this purpose IMS will allow the user registration process. Moreover, the system will support the features that allow the automatic updating of the central repositories whenever a particular user updates the status of its activities in their own system.

3.2.1.2. Stimulus/Response Sequences

Input: User will input Resource data.

Output: The system will update the database repository.

3.2.1.3. Functional Requirements

REQ – 1: The system should have feature to insert data of resources.

3.2.2. Activity handling

3.2.2.1. User Description and Priority

IMS will allow the searching of activities by the authorized user. The request will be sent to the IMS App server; and resource repository will be searched to find a match for the requested activity based on the area of interest. Upon finding a match, list of available activities will be displayed.

3.2.2.2 Stimulus/Response Sequences

Input: User will input activity data.

Output: The system will display the matched list of available activities.

3.2.2.3. Functional Requirements

REQ – 1: The system should have feature to search for a particular activity.

REQ – 2: The system should display the current status of the selected activity.

3.2.3. Login/Access Rights

3.2.3.1. Description and Priority

This feature would allow logins to users based on their credentials.

3.2.3.2. Stimulus/Response Sequence

Input: User will login using name and password.

Output: System will present features according to the access rights available to the user.

3.2.3.3. Functional Requirements

REQ – 1: The system shall allow the users to login after Authentication.

3.2.4. Report Generation

3.2.4.1. Description and Priority

This feature would allow admin to maintain records of all the categories of users. Different users would have different access level to the records. The report of the records will be generated based on the area of interest.

3.2.4.2. Stimulus/Response Sequence

Input: Admin will enter a query.

Output: System will generate a report in a pre-defined format.

3.2.4.3 Functional requirements

REQ – 1: A central database management system will be implemented to store records.

REQ – 2: The system shall generate the reports of the record based on the given criteria.

3.2.5. Maps

3.2.5.1. Description and Priority

Google Map is to be incorporated in the app to assist the user in locating the area where the activity is to be performed.

3.2.5.2 Stimulus/Response Sequence

Input: user will select a particular activity.

Output: System would show the map and relevant details of the area of interest.

3.2.5.3. Functional Requirements

REQ – 1: Integration of maps to our system.

3.3. Other Nonfunctional Requirements

3.3.1. Performance

IMS has to be efficient in its response and operation. The product domain requires that the software is optimized in terms of performance. The data flow should happen in the most efficient way.

3.3.2. Safety

SR – 1: In case of data loss, system will back up the data and will restore it as per demand.

SR – 2: System will be deployed on online Google platform so that they've their inherent fault-tolerance capabilities.

Output: System would show the map and relevant details of the area of interest.

3.3.3. Security

SE – 1: Users shall be required to log in to the IMS for their own credential information.

SE – 2: The system shall permit only authorized users to do administrator's task.

- SE – 3: The system shall permit users to view only their own profile and data that are intended for them.
- SE – 4: The system must perform an encoding technique such as hashing to save all passwords securely.
- SE – 5: The System will provide confidentiality and integrity.

3.3.4. Software Quality Attributes

Quality attributes of IMS are described below. By following these attributes, the quality of IMS will be improved.

3.3.4.1. Runtime System Qualities

At runtime IMS has to provide its users with functionalities so that they can publish and search for the desired activities. Some of the runtime qualities that should be considered in the development of IMS are described here.

3.3.4.2. Functionality

IMS must provide functions to publish and search the service activity. IMS must provide the functions of authentication of user.

3.3.4.3. Availability

IMS should be available 24x7.

3.3.4.4. Usability

The system should present all functionalities in such a way that nothing is missed by the user.

3.3.4.5. Non-Runtime System Qualities

These are qualities of IMS which are required to make this app useful for further enhancements. It will also be helpful in future development as well as extending system to different environments.

3.3.4.6. Modifiability

IMS must support modifiability so any further improvements or features are easy to incorporate.

3.3.4.7. Portability

The IMS App should be able to work on multiple Android hardware platforms. The IMS servers should be a platform-independent and should support interoperability.

3.3.4.8. Interoperability

Web app will be interoperable with Google App Engine.

3.3.4.9. Testability

Different quality tests should be performed so that IMS is free from faults and perform according to requirements.

3.3.4.10. Business Rules

No specific organization is targeted but overall rules and procedures for linking the users will be governed by the legal rules and procedures.

3.4. User Classes and Characteristics

The software has multi user types. All types of users have different access level to the system and its data and can perform functions assigned to their respective roles.

3.4.1. Admin User

The admin user can create new user and manage the credentials of existing users. He will review and approve the published services and manage the database while addressing the reports and complaints of the users about the services. He will act as a contact person for the IMS App. He will also generate reports that will be helpful for analyzing the trends in service activities.

3.4.2. End Users

The user can register himself with his unique email ID and password.

3.4.2.1. DIA User

The DIA user can publish and analyzed information about a particular activity service.

3.4.2.2. District Level Users

These users perform further activities on receiving data from field users.

3.4.2.2. Field User

The Field users are the activity data entry operators.

3.5. Assumptions and Dependencies

AS – 1: Basic assumption for development of IMS is that system should be available 24x7 since an update can happen at any time.

AS – 2: Users of IMS should be assumed to have an computer system with internet access.

DEP – 1: There will be a permanent dependency on Google Maps for location services.

DEP – 2: System is dependent on database server to store user data.

3.6. Summary

This chapter describes the requirements of the system as described by Project Supervisor. It includes interface, functional and nonfunctional requirements along

with the main features, system would provide to the end user. These requirements have been set after checking the feasibility of the system. These requirements have been considered as the fundamental principles for testing and standardization of the product.

4. System Design Specifications

4.1. Overview of the modules

The system will be architected mainly in 3 fundamental parts “Front End Interface”, “Core System”, and “Database” as shown in the following abstract diagram:

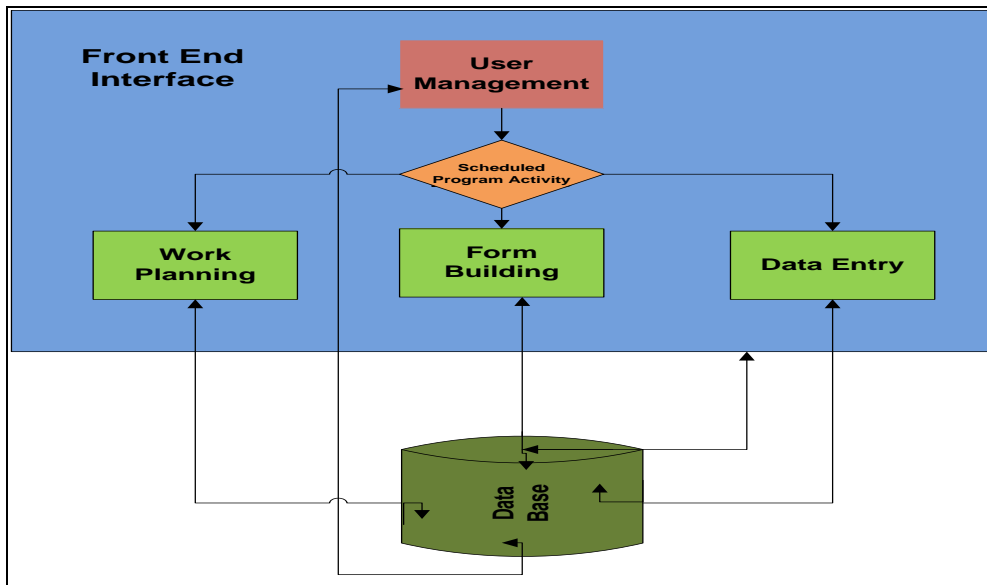


Figure 1 - Overview of the Modules

4.1.1. Description of the modules

The “Front End Interface” is maintained for sub components of core system . Each user is given role and authority by user management module and then allowed to access work planning , form building or Data entry Module accordingly. The work planning includes user specific project activities. Form building Module presents dynamic scenarios of forms, containing meaningful information. Authorize user can access data entry module, where useful information is loaded. The information in all the above steps will also be connected to the Database for querying the related information.

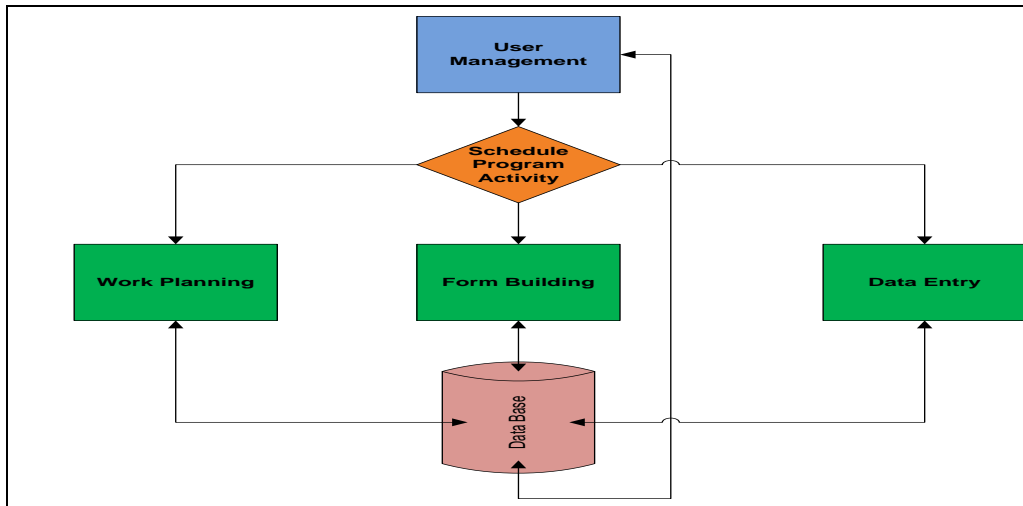


Figure 2 - Description of Modules

4.2. Structure and relationships

Focusing upon the internal structure of the system, this section ponders upon the interrelationships and dependencies among various components.

4.2.1. Overall Structure of the system

The diagram shows the main components of the system along with their interactions with each other. It mainly describes the system structure which is further augmented by the explanatory text as follows:

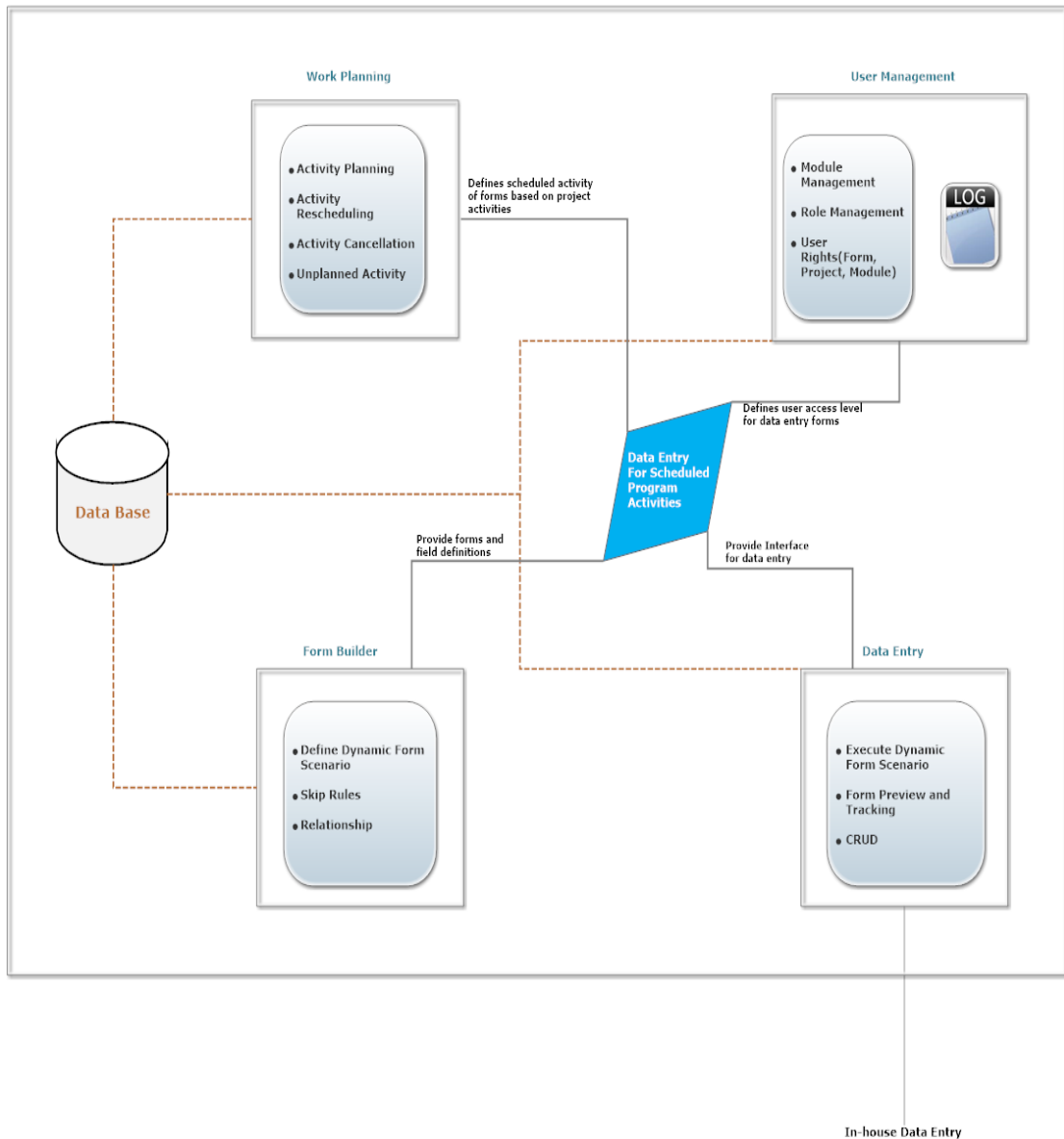


Figure 3: Overall Structure of the System

4.2.1.1. User Management

Administrator controls three types of users, DIA, DO and a FO. All kind of users are given authority by sub module USER ROLE. User can access through **User Log in** and modify contents according to his authority .

4.2.1.2. Module Access

This module controls user's access to rest of modules. After login user is provided with access to either work activity, Forms or Data entry. User module can communicate with other modules through Module access.

4.2.1.3. Dashboard activities

This module controls user activities. Activities include main dashboard, activity Logs and Reports. User can generate any kind of activity ,modify and delete it. Main Dashboard provide interface for activities. User can generate reports here.

4.2.1.4. Form Building

This module deals with form's creation, review, use and delete (CRUD). It is responsible to analyze the mapped activity information according to the proximity-based sorting algorithm. This module returns the form of closest activity matched, based on the subject location.

4.2.1.5. Data Entry

This module deals with data CRUD. It generates and organizes the data of the Activity with a view to return those to the Front End Interface for onward Broadcast of the same as Notifications.

4.3. UML Diagrams

4.3.1. Use Case Diagram

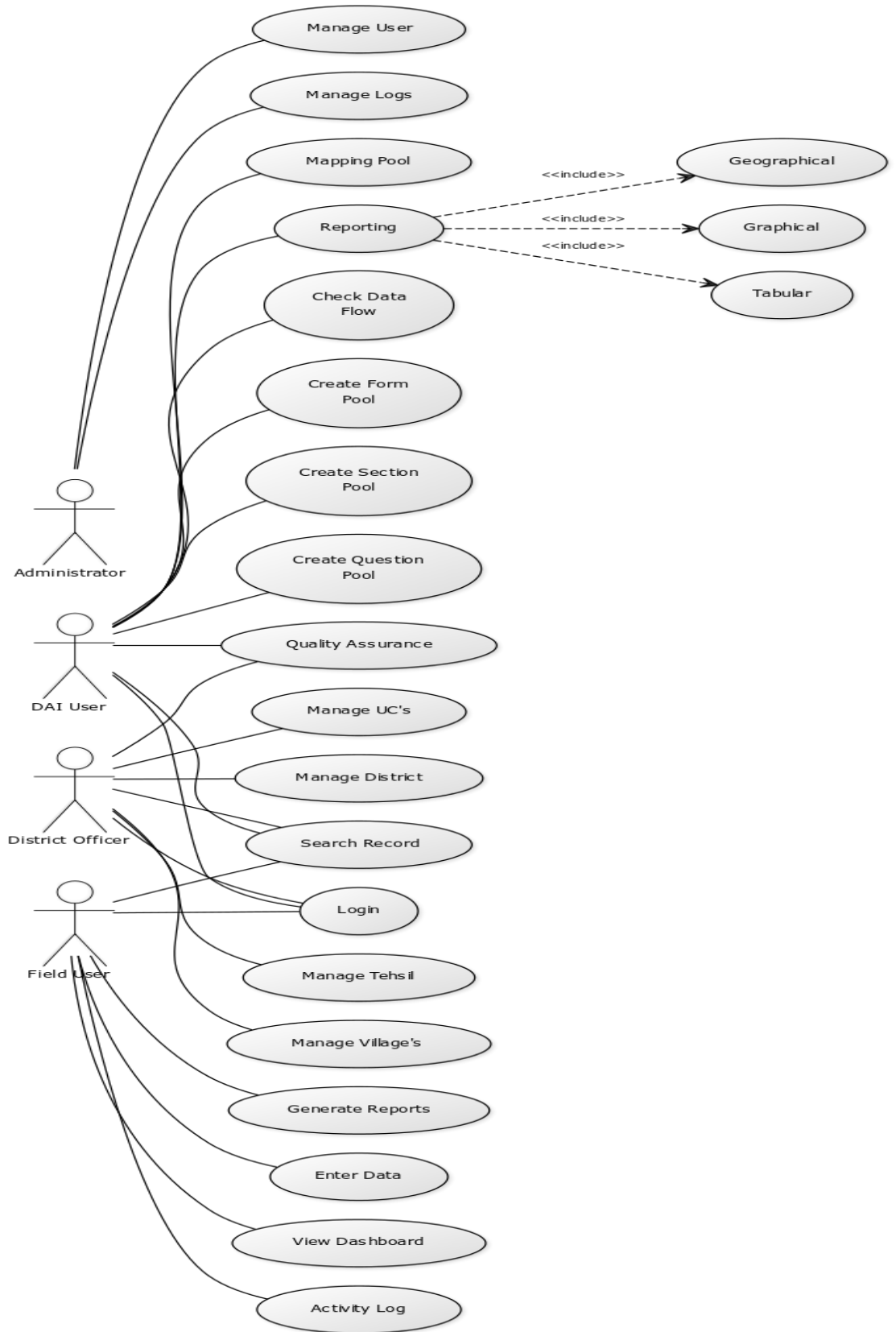


Figure 4: Use Case Diagram

4.3.1.1. Use Cases Description

The key use cases depict the main functionality of IMS. Figure 4 mentions the interaction of the users with the system. IMS has mainly two types of users: admin and the rest of the other users. Both the types of users have different access levels to the database, and functionality of the system.

The admin user can manage user accounts. He can view/ edit the profile. The other users can either post a activity or request for it. The detail scope of all the Use Cases is mentioned in **Appendix B**.

4.3.2. Class Diagram with description

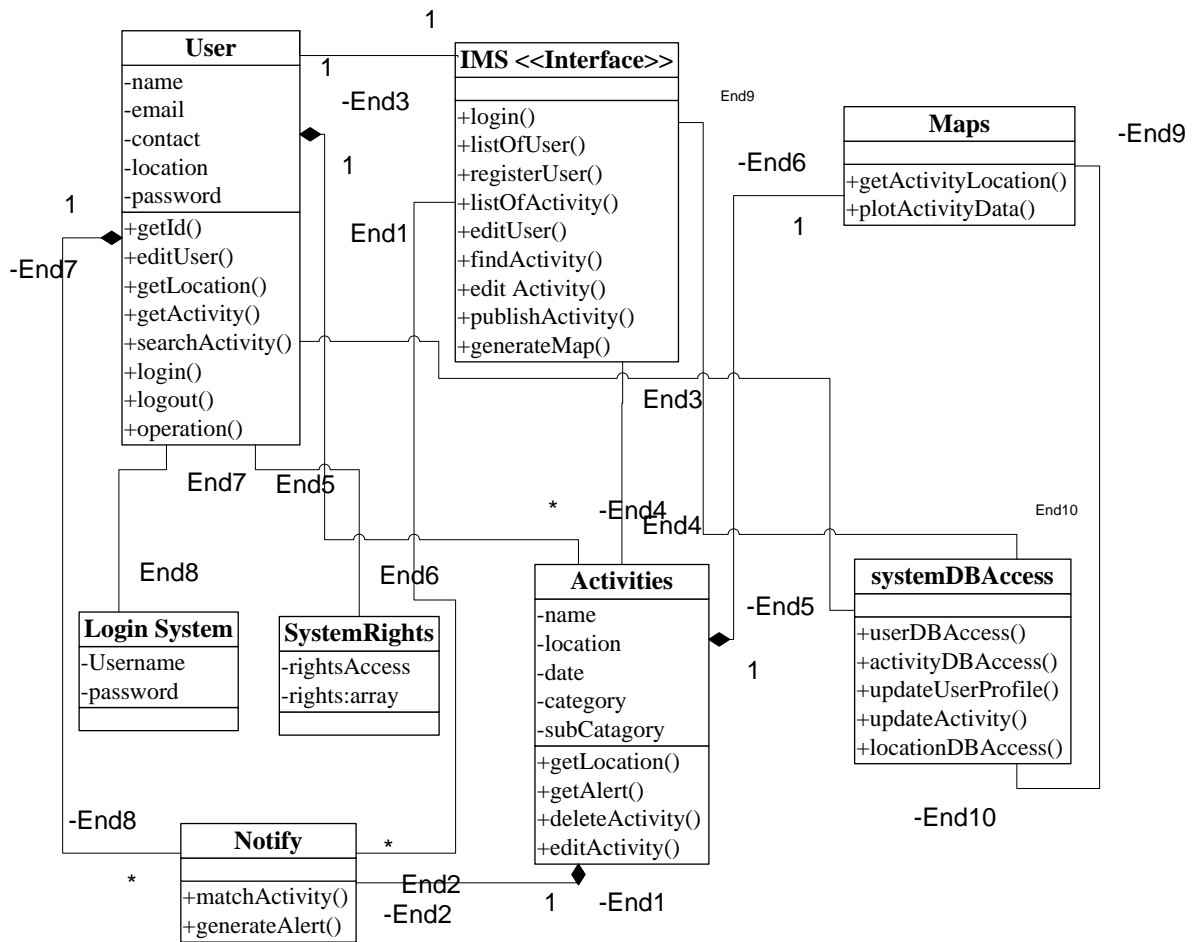


Figure 5: Class Diagram

4.3.2.1. Description of the diagram

The diagram reveals class-structure of the application.^[3] All the classes shown in the class diagram are further described in the sub sections.

4.3.2.1.1. IMS <<Interface>>

Contains all the functions which our application has to offer. It calls all the functions from respective classes as and when invoked by the user.

4.3.2.1.2. Activities

This class contains the information about all the activities.

System DB Access

This is the main class to access database objects. All the data operations are performed through this class. All other classes send or retrieve data from database through this class. It acts as an interface between all software classes and database system.

4.3.2.1.3. User

It contains all the functions required for user management features.

4.3.2.1.4. System Rights

This class has information roles and responsibilities for each user type. So all the roles and their details and functionalities offered for different roles are in this class.

4.3.2.1.5. Login System

This class contains data and functions for users and organizations to login. This class simply authenticates user information and based on which user is given access to the software.

4.3.2.1.6. Maps

This class maps the information of the activities done by user and then plot its location on the map.

4.3.2.1.7. Login System

This class contains data and functions for users and organizations to login. This class simply authenticates user information and based on which user is given access to the software.

4.3.2.1.8. Notify

This class matches the activity occurrence and accordingly notifies the user.

4.3.3. Sequence diagrams

A Sequence diagram is an interaction diagram that shows how processes operate with one another and what is their order. It is a construct of a Message Sequence Chart. The sequence diagrams show the object interactions arranged in time sequence. These are typically associated with use case realizations in the Logical View of the system under development. These have been mentioned in **Appendix C**.

4.3.4. Activity diagrams

This section shows the activities that a user needs to perform to accomplish a task. Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e. workflows). Activity diagrams show the overall flow of control. The activity diagrams of IMS will ponder upon the course of actions, adoptable by the users of the system. These have been mentioned in **Appendix D**.

4.4. Summary

This chapter described the design of the software taking into consideration different assumptions and constraints that applied on the system because of its goals and requirements. Component Diagram, Class Diagram and Use case diagram have been added to explain system functionalities. A few low level design diagrams have also been shown to ascertain the internal behavior of the system.

5. Implementation Details

5.1. Introduction

The detailed design is transformed into an application by using various technologies. The implementation details are discussed in the following sections giving details of the system's internal working.

5.2. MVC implementation

The project is implemented in MVC design pattern. XAMPP is used as unified platform for PHP, MYSQL and Database.

5.3. Map implementation

QGIS has been used to prepare maps at different levels. Open layer JS library and Google API3 has been used for GIS purpose.

5.4. User Interface

Since IMS is a mobile application so it needs to be more interactive, user friendly and responsive. Navigation flow of the application should be self-driving so as to guide the user in completing his desired task/activity. Interface is an important part of any mobile application and it is treated as a separate module. The issues addressed in the user interface design are that no functionality is hidden from user and data should be presented in a clear way to end user so nothing is missed by him.

The interface is developed using Android native Layouts, list views, buttons, action bars and drawer layout. The interface is kept separate from business layer by making fragments classes. Each fragment in project hierarchy represents a different screen.

5.5. User Management Module

This module covers all functionalities associated with a user. It provides functionalities to create a new user, edit user, authorize user to login into the system and display system features to a user.

5.6. Summary

Implementation details of IMS are discussed in this chapter. A brief introduction to different tools and technologies employed is also given.

6. Testing

6.1. Introduction

To ensure quality of the product, testing is conducted. Accuracy of functions performed by IMS has to be tested and maintained to improve quality of software. Software testing techniques and results obtained are discussed in the coming sections.

6.2. Testing Levels

Separate modules are developed to provide different functionalities of IMS. All of these modules are tested at different levels in their development and post-integration process. Different levels at which IMS has been tested, and results obtained are described in this section.

6.2.1. Unit Testing

Unit testing involves the testing of each module at the completion and at times, during the very course of development of the module. It is a testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use. IMS web application has also been passed through unit testing process, wherein the various units have been tested in accordance with the anticipated output of each unit. The detail account of test cases has been explicitly mentioned in **Appendix A**.

6.2.2. Integration Testing

IMS's different modules which were developed and tested independently were also tested during integration to ensure system stability. Integration testing helped in ensuring that different modules when combined give complete functionality and nothing is missed or some functionality doesn't give error when integrated with other modules. Integration testing gave us more than 90% results ensuring that most modules were integrated with others as well as compatible. This shows that errors

were minimized during integration testing. The detail account of all the integrations testing process vis-à-vis its results has been mentioned in **Appendix B**.

6.2.3. System Testing

System testing was performed at the end of development and integration of IMS. Complete system was tested using sample data. All sub modules including user registration, publishing/ seeking the services, managing the credentials information, viewing/ editing the profile etc. were tested as a whole using the sample data. Almost 90% of test cases were successful ensuring that most of errors and bugs in the system were removed and system was stable enough to perform optimally. The detail account of the test cases has been mentioned in **Appendix C**.

6.3. Summary

Testing not only maintains the software quality but also improves overall usability of the project. At different stages of development suitable testing techniques were used to ensure product works accurately and efficiently. All errors detected during testing were removed and the test cases were prepared and made part of this document for the future compliance.

7. Conclusion and Future Work

7.1. Conclusion

The public service web-app IMS provides platform for managing information geographically on maps. Users can search their desired activities in different domains. IMS facilitates the users in constructing a framework between the service providers and service seekers. Hence remove the requirement for ad-hoc ways and means of public services.

7.2. Future Work

Future work on this project can include developing android version, adding more options categories of services, support for iOS, Windows platform and incorporation of enhanced search mechanism by resorting to context based search.

User Manual

Overview of User Manual for IMS

This is the user manual for public service web app “IMS”. IMS has the following different interfaces for Users. Moreover Admin have web interface to manage the database.

Navigation path for Users

Any user desirous to post any activity request needs to sign in with a valid user account. This is the Login Screen of the IMS application.



Figure 6: User Manual - Login Screen

Main Dashboard

Once successfully logged in, user goes to Main Screen. On this screen user have the option to view details. User is also displayed all the main categories of the activities.

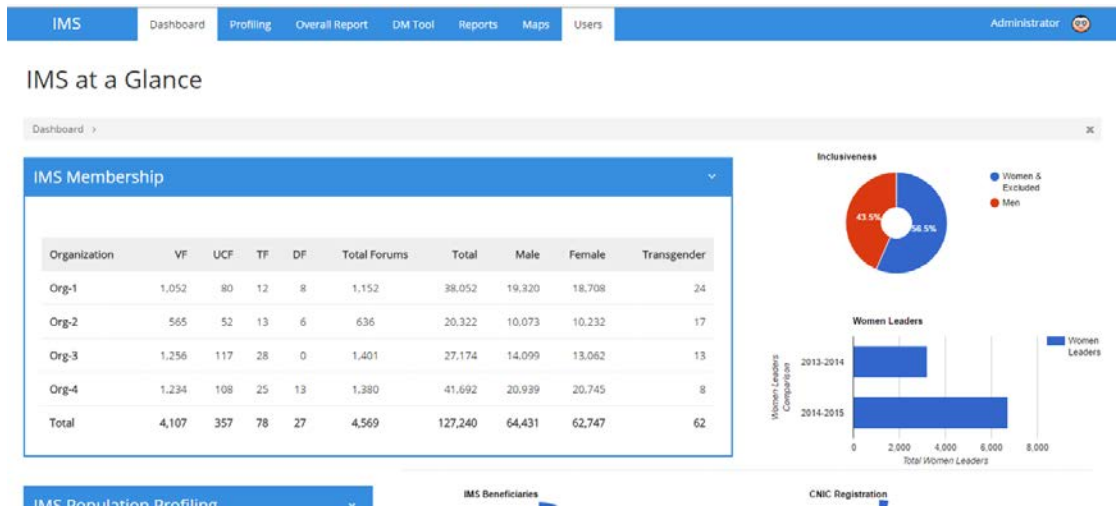


Figure 7: User Manual - Main Dashboard

User Profiling

Details of user Profiles can be viewed here. locationwise profiles are maintained. Users number on different forums are also given.

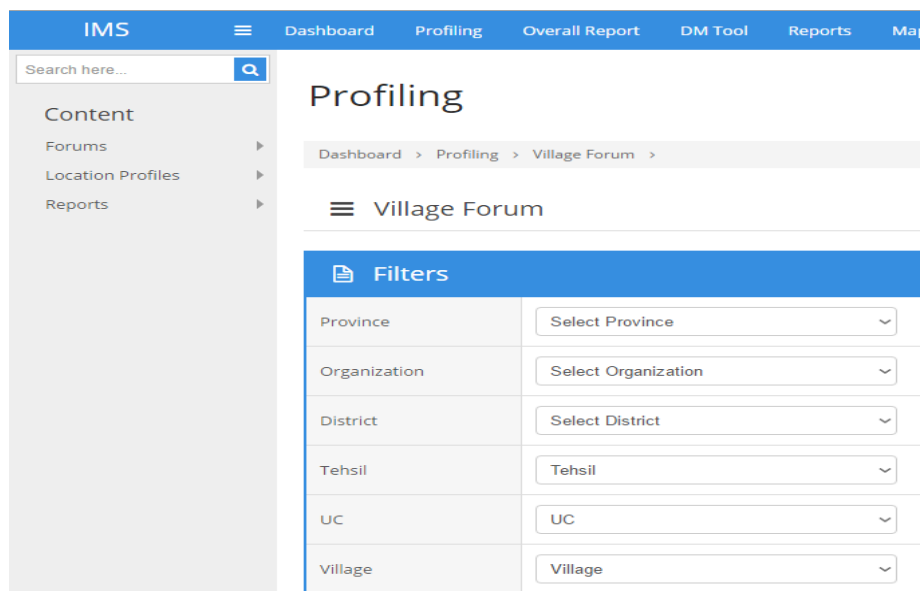


Figure 8: User Manual - user Profiling

Overall Report

Any user can view over all report of all activities of the project. User can modify the report according to its authorization.

IMS																
Dashboard	Profiling	Overall Report	DM Tool	Reports	Maps	Users	Administrator									
Overall Report																
Checklists	Bahawalnagar	Bahawalpur	Bannu	Bhakkar	CHARSADDA	Chitral	Faisalabad	Lahore	LAYYAH	Lower Dir	Nankana Sahib	Pakpattan	T.T Singh	Abbottabad	Attock	Haripur
AC Data Tool (Focal Person Profile)	10	13	0	10	11	0	11	0	10	0	0	10	10	12	1	12
AC (Monthly Sheet)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Awareness Raising Training	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CNIC and Voter Registration	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

Figure 9: Overall report

DM Tools

Data management tool has sub menus to display data of activities in different areas. It has 7 options as displayed in snap shot.

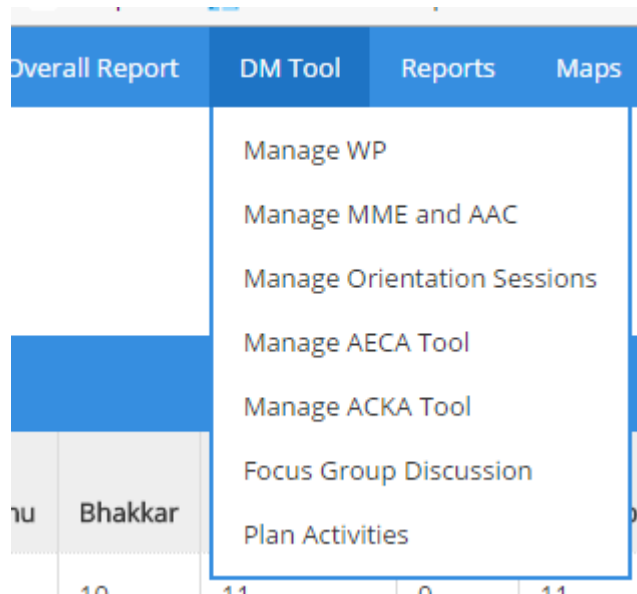


Figure 10: DM Tools

Admin

Administrator profile and account setting can easily be navigate from main menu as given.

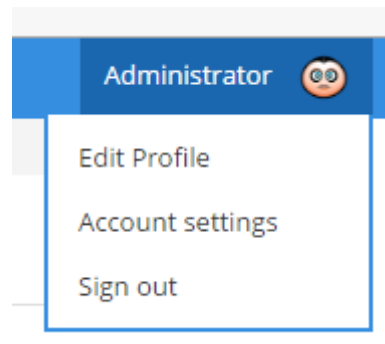


Figure 2: User Manual - Admin

Maps

Maps is placed on main dashboard. Users can easily navigate through all the location wise data on maps in 4x categories as shown in the snap shot.

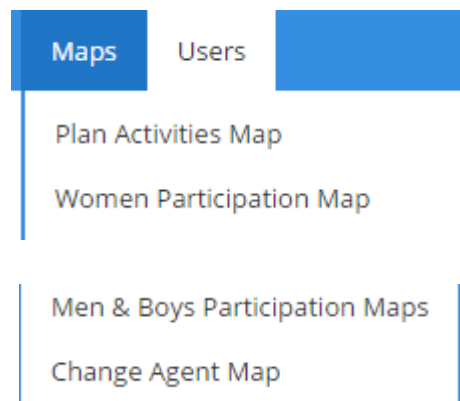
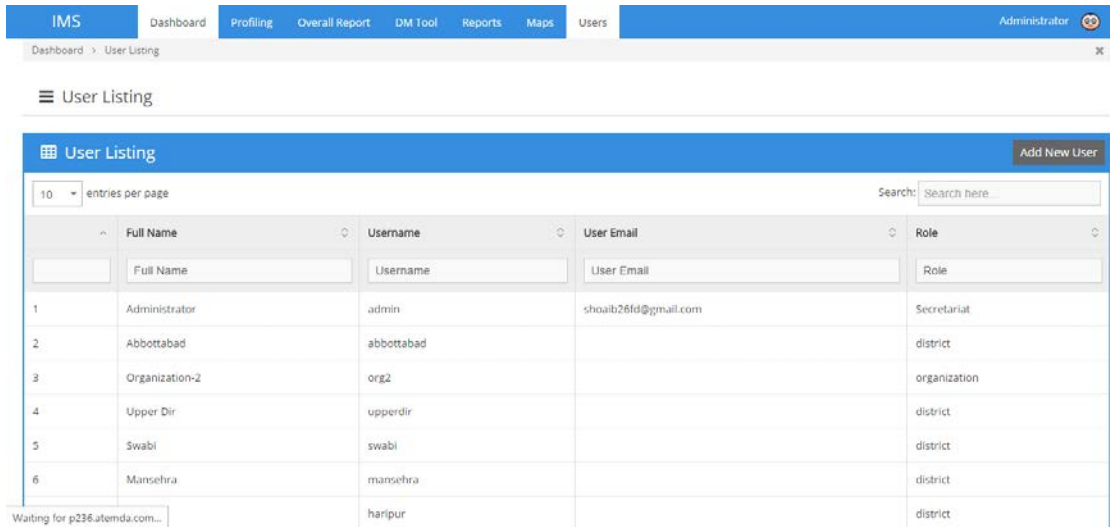


Figure 12: User Manual - maps

User

All the user details are given in this menu. A user can view number of other users with their roles and names as shown from main menu.



The screenshot displays the 'User Listing' page in the IMS system. The page features a navigation bar with 'IMS' and 'Users' tabs, and a breadcrumb trail 'Dashboard > User Listing'. A search bar and 'Add New User' button are located at the top right of the table area. The table lists six users with the following details:

	Full Name	Username	User Email	Role
1	Administrator	admin	shoaib26fd@gmail.com	Secretariat
2	Abbottabad	abbottabad		district
3	Organization-2	org2		organization
4	Upper Dir	upperdir		district
5	Swabi	swabi		district
6	Mansehra	mansehra		district

Figure 13: User Manual - User

Use Cases Description

Manage Users

Use Case ID:	1		
Use Case Name:	Manage Users		
Actors:	Administrator		
Created By:	Shoaib	Last Updated By:	Shoaib
Date Created:	29/9/2015	Date Last Updated:	29/9/2015
Description:	1. Admin has to login to the system to manage users i.e. create, read, update and delete user profiles.		
Preconditions:	1. Admin has to login.		
Post conditions:	1. The System must record the change.		
Normal Flow (primary scenario):	1. The actor creates, reads, updates and deletes the user details. 2. Click on create or read or update or delete button as required.		
Alternative Flows:	1. The actor will contact the system maintenance team to check if there is some error with database systems and has to resolve the error.		

Table 10: Use Case Description - Manage User

Login

Use Case ID:	2		
Use Case Name:	Login		
Actors:	Administrator, User		
Created By:	Shoaib	Last Updated By:	Shoaib
Date Created:	29/9/2015	Date Last Updated:	29/9/2015
Description:	A user tries to login to the system.		
Preconditions:	1. User has to open the login page first.		

Post conditions:	1. If the use case was successful, the actor is now logged into the system. If not the system state remains unchanged.
Normal Flow (primary scenario):	This use case starts when an actor wishes to log into the System. 1. The system requests that the actor enter his/her ID and password. 2. The actor enters his/her ID and password. 3. The system validates the entered ID and password and logs the actor into the system.
Alternative Flows:	1. Invalid Name / Password If in the Basic Flow the actor enters an invalid ID and/or password, the system displays an error message. The actor can choose to either return to the beginning of the Basic Flow or cancel the login, at which point the use case ends.

Table 11: Use Case Description - Login

Edit/View Profile

Use Case ID:	3		
Use Case Name:	Edit/View Profile		
Actors:	Administrator, User		
Created By:	Shoaib	Last Updated By:	Shoaib
Date Created:	29/9/2015	Date Last Updated:	29/9/2015
Description:	A user tries to edit/view his profile.		
Preconditions:	1. User has to open the sign in page first.		
Post conditions:	1. The System must show the profile and record the changes made therein.		
Normal Flow (primary scenario):	1. The user clicks profile options. 2. The system enables the user to view or modify his profile information. 3. The system records the changes made in the profile.		

Alternative Flows:	<ol style="list-style-type: none"> 1. The user tries to modify his profile ID. 2. The system throws error message, mentioning that user cannot change his profile ID.
--------------------	---

Table 12: Use Case Description - Edit/View Profile

Publish Service

Use Case ID:	4		
Use Case Name:	Publish Activity		
Actors:	User		
Created By:	Usman	Last Updated By:	Usman
Date Created:	28/09/2015	Date Last Updated:	28/09/2015
Description:	User will publish the activity performed.		
Preconditions:	<ol style="list-style-type: none"> 1. User will sign in first. 2. User must be on the Dashboard-activities-page. 		
Post conditions:	1. User has successfully published the activity with necessary details.		
Normal Flow (primary scenario):	<p>This use case starts when the user wishes to publish a activity in his area.</p> <ol style="list-style-type: none"> 1. User publishes an activity. 2. System updates the resources database. 		
Alternative Flows:	<p>Limited Connectivity</p> <p>In case of connectivity issues with the database, the system will allow the user to retry publishing the activity.</p>		

Table 13: Use Case Description - Publish Activity

Generate Report

Use Case ID:	5		
Use Case Name:	Generate Report		
Actors:	Administrator		
Created By:	Usman	Last Updated By:	Usman
Date Created:	28/9/2015	Date Last Updated:	28/9/2015
Description:	The system will enable the Administrator to generate the		

	report of the activities offered and requested.
Preconditions:	1. User has to login to the system and open report generation form to enter criteria for report generation.
Post conditions:	1. The system will generate the detail report of the services.
Normal Flow (primary scenario):	1. Administrator selects the report generation-criteria followed by clicking “generate” button. 2. The system generates the report against the matched criteria.
Alternative Flows:	1. The criteria selected by the Administrator don’t hold any service. 2. The system returns error message, mentioning that no record exists.

Table 14: Use Case Description - Generate Report

Map Service

Use Case ID:	6		
Use Case Name:	Map Service		
Actors:	User		
Created By:	Shoaib	Last Updated By:	Shoaib
Date Created:	29/9/2015	Date Last Updated:	29/9/2015
Description:	The user location will be marked using Location Service API.		
Preconditions:	1. User is logged in. 2. Either the service or the request is published.		
Post conditions:	1. User location will be marked by the system.		
Normal Flow (primary scenario):	1. User current location is marked by the system.		
Alternative Flows:	1. User manually enters his location.		

Table 15: Use Case Description - Map Service

Search Activity

Use Case ID:	7		
Use Case Name:	Search Activity		
Actors:	User		
Created By:	Usman	Last Updated By:	Usman
Date Created:	28/4/2015	Date Last Updated:	28/9/2015
Description:	The user will search for a particular activity.		
Preconditions:	<ol style="list-style-type: none"> 1. User is logged in. 2. User is in dashboard activities page. 		
Post conditions:	<ol style="list-style-type: none"> 1. User has successfully found the desired activity. 		
Normal Flow (primary scenario):	<ol style="list-style-type: none"> 1. User enters the details of desired activity. 2. System returns the searched result. 		
Alternative Flows:	<p>Limited Connectivity</p> <p>In case of connectivity issues with the database, the system will allow the user to retry publishing the activity request.</p>		

Table 16: Use Case Description - Search Activity

Sequence Diagrams

Different Scenarios and their corresponding events are discussed in this section with the help of sequence diagrams.

Manage User

Description: This scenario describes the sequence of events that take place when either the users is created or edited.

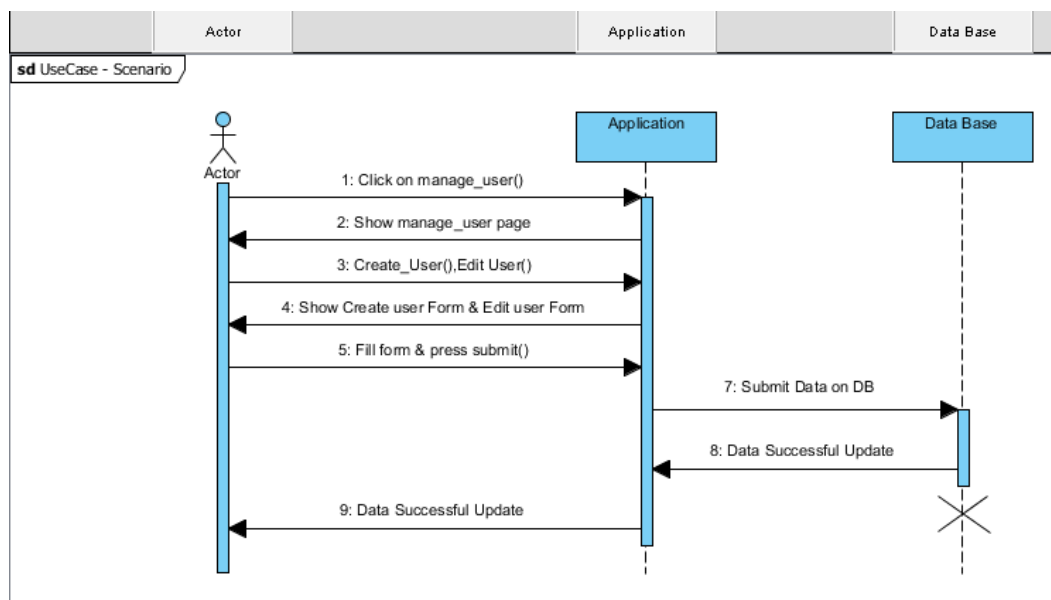


Figure 14 - Manage User Sequence

7.2.1.1. Login

Description: This scenario describes the sequence of events that take place when the users log in to the system. The alternative prospects of the events have also been catered for in case the user tries to log in without valid credentials.

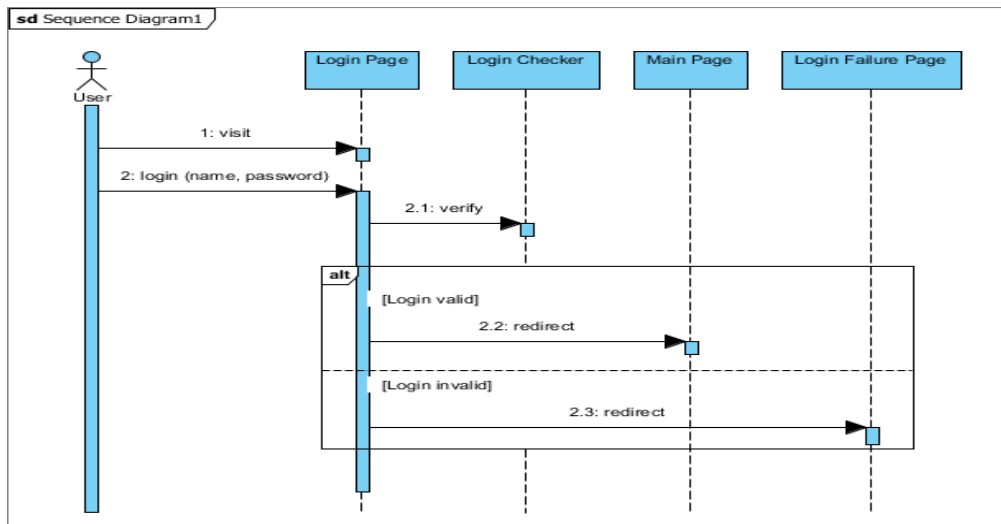


Figure 15 - Login Sequence

7.2.1.2. Create Form

Description: This scenario describes the sequence of events in which Data input and analysis officer create, fill and submit form.

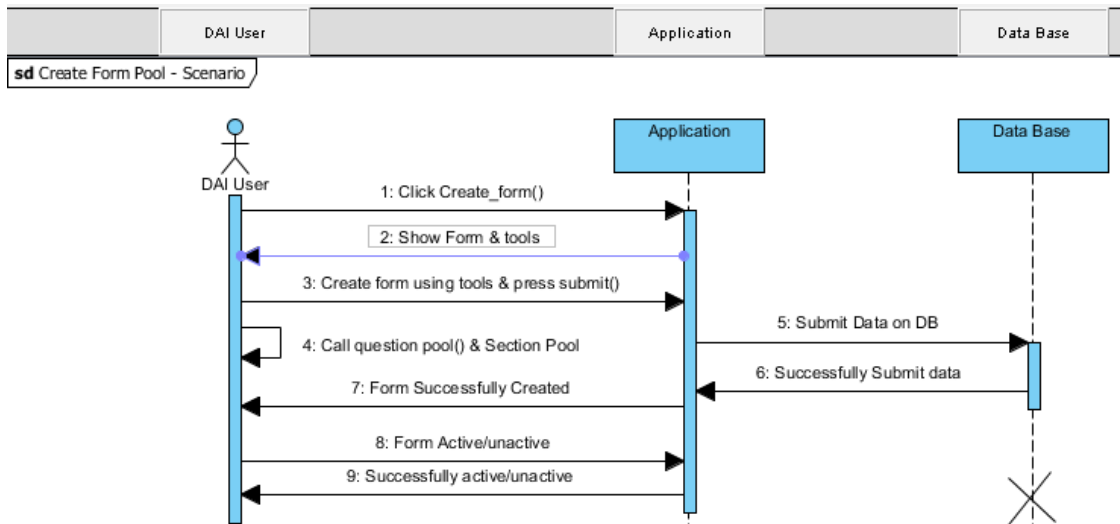


Figure 16 - Create Form Sequence

7.2.1.3. Quality Assurance

Description: This scenario describes the sequence of events in which DIA user and DO insure quality of data.

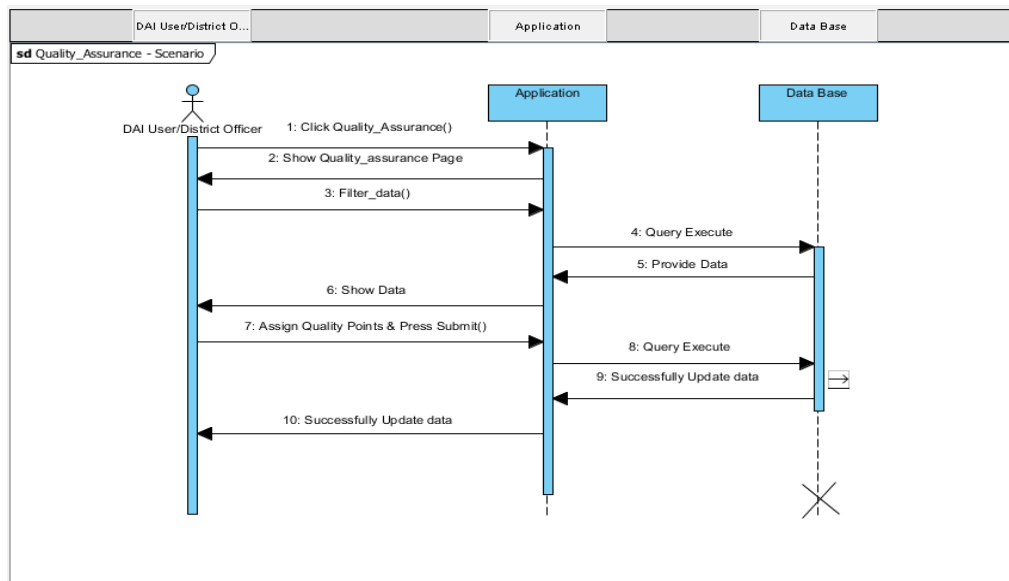


Figure 17 - Quality Assurance Sequence

7.2.1.4. Search

Description: This gives the sequence of events in which DIA and DO user search data and generate report.

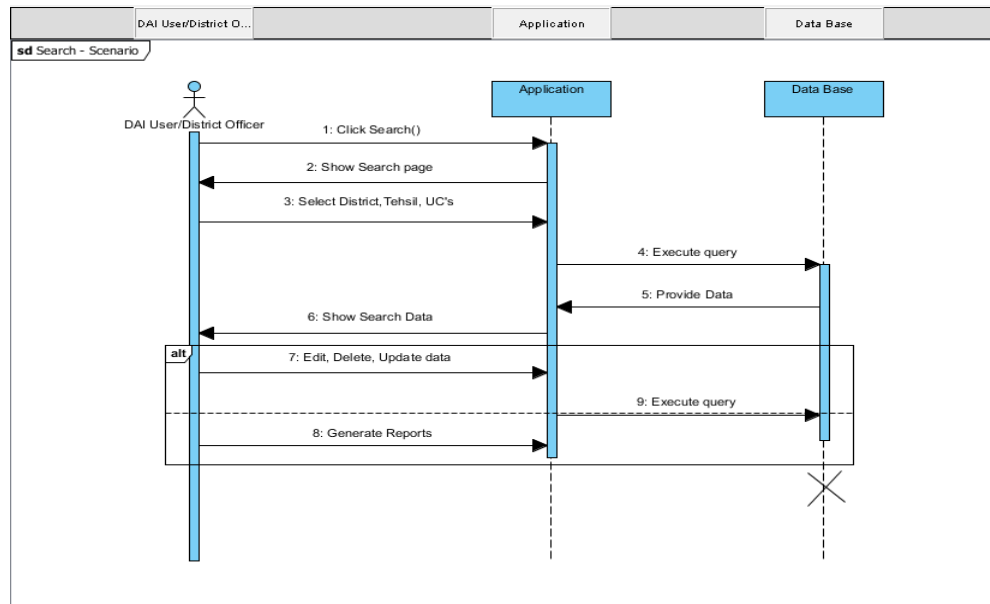


Figure 18 - Search Sequence

7.2.1.5. Generate Reports

Description: This shows sequence of events in which DIA user generate report location wise in different forms.

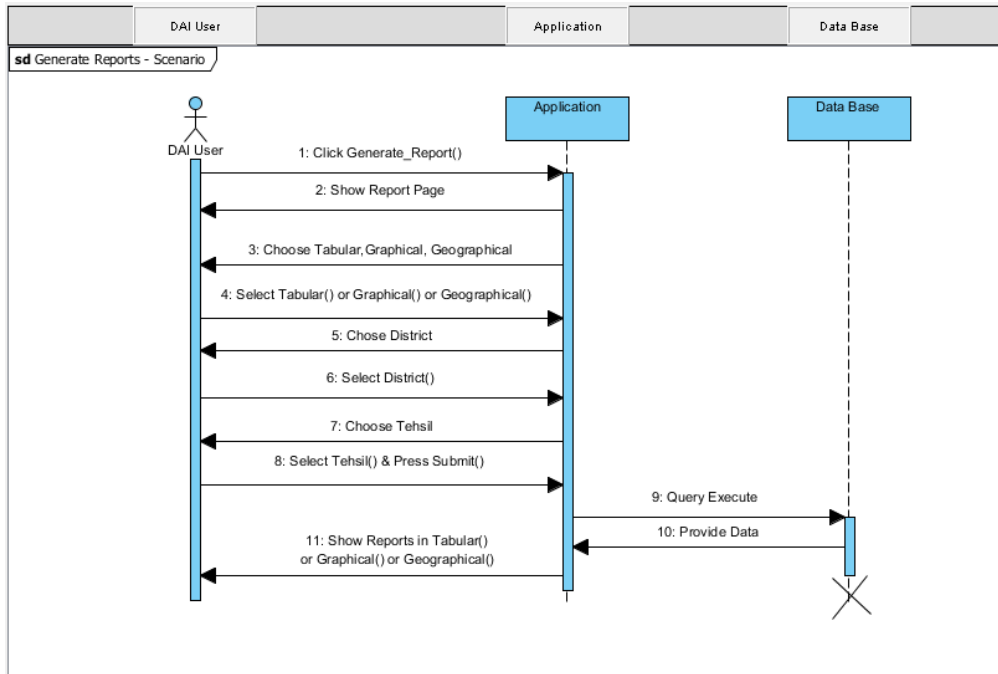


Figure 19 - Generate report Sequence

7.2.1.6. Manage District, Tehsil, UC's and Village

Description: This shows sequence of events in which DIA user manage data location wise.

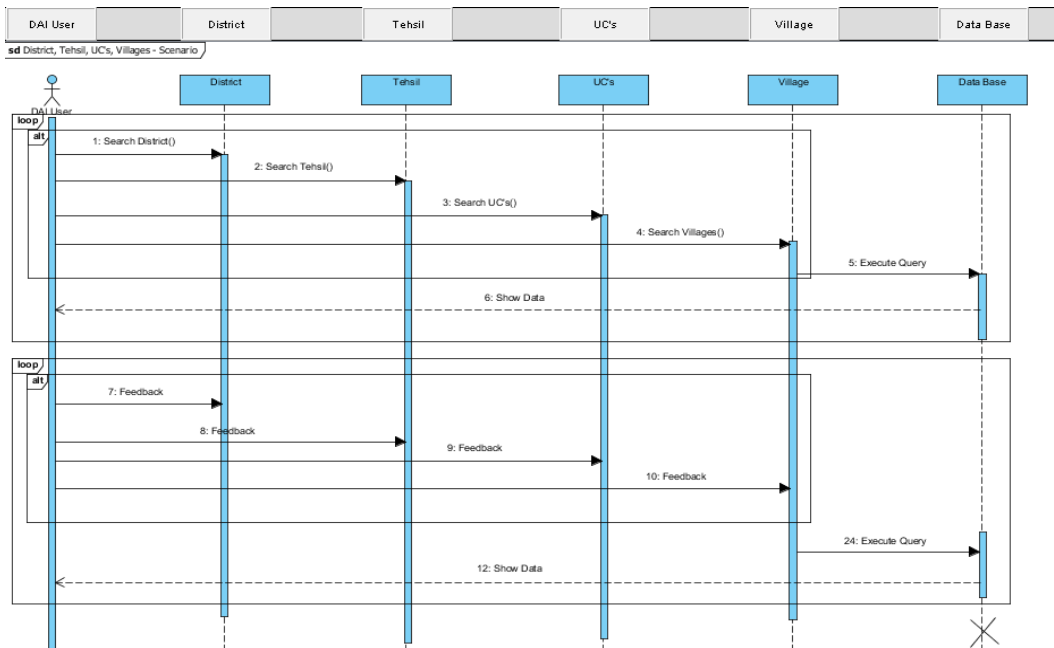


Figure 20 - Location wise data management Sequence

7.2.1.7. Dashboard, Activity Log, Form

Description: This scenario describes the sequence of events in which FO user manipulate Dashboard and Forms.

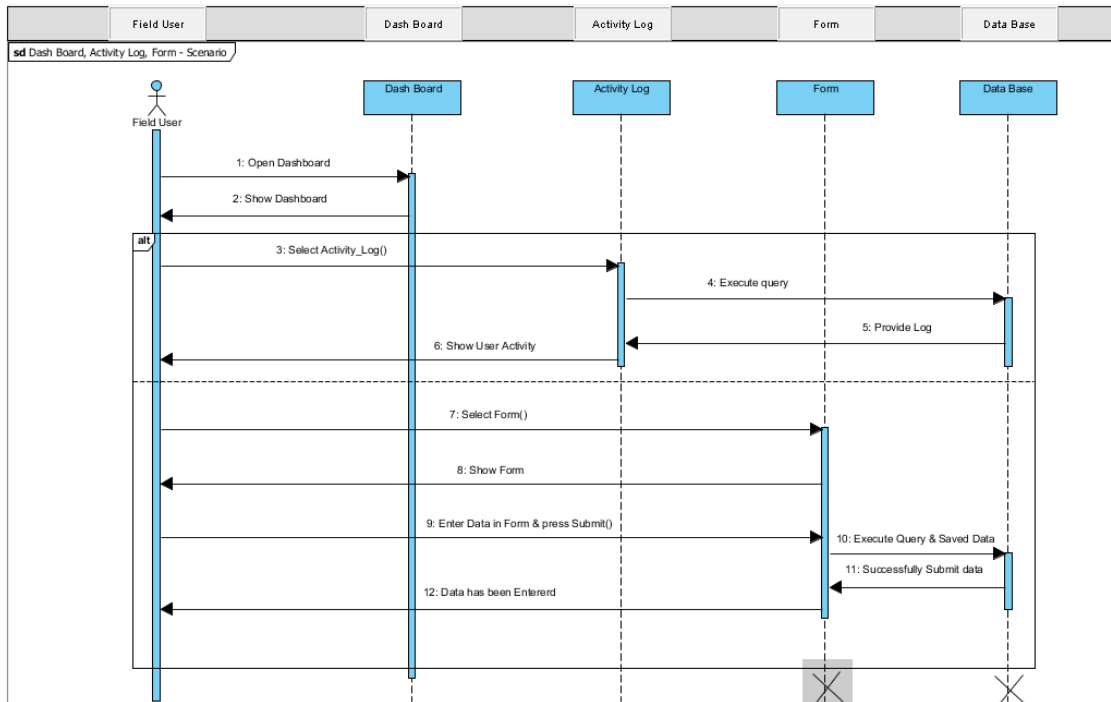


Figure 21 - dashboard Activities Sequence

Activity Diagrams

This section shows the activities that a user need to perform to accomplish a task.

Manage User

Description: This scenario describes the flow of activities in necessary for the Administrator to manage rest of the users.

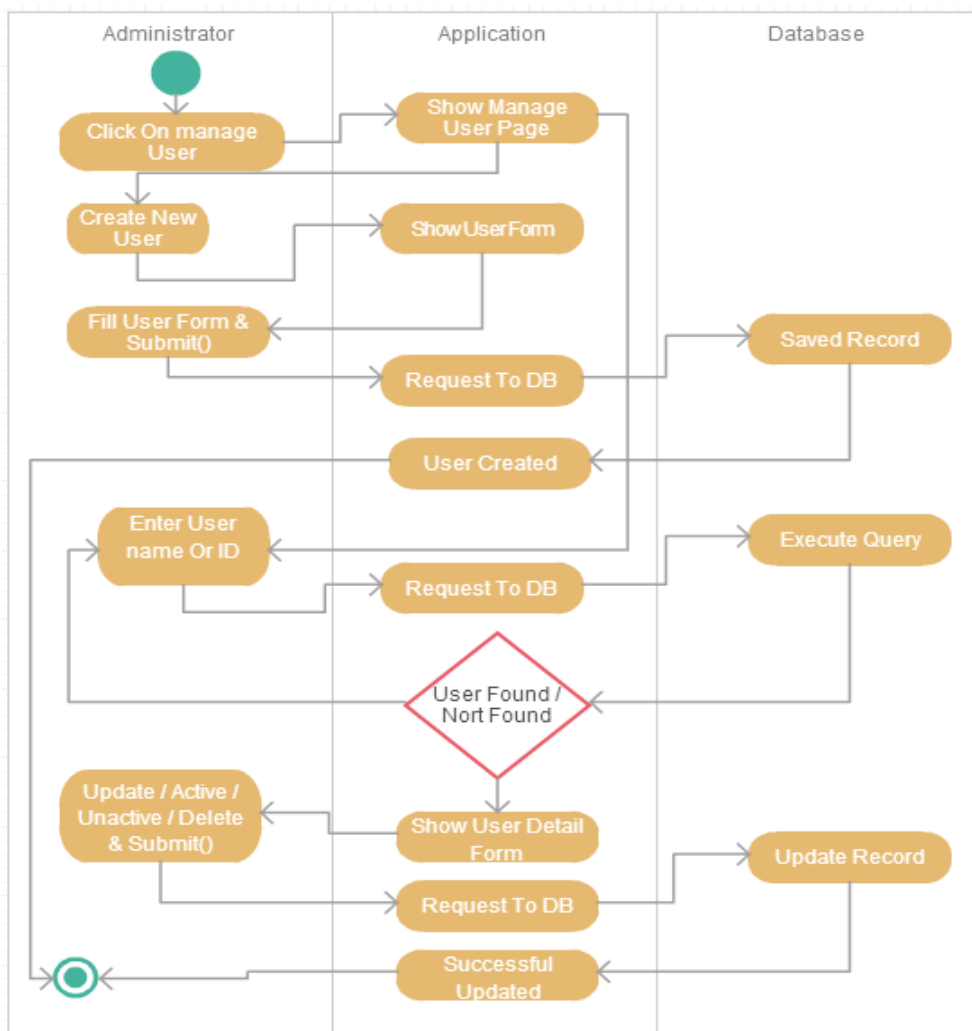


Figure 22 - Manage User Activity

Create Form

Description: This scenario describes the flow of activities necessary for the user to create and save form.

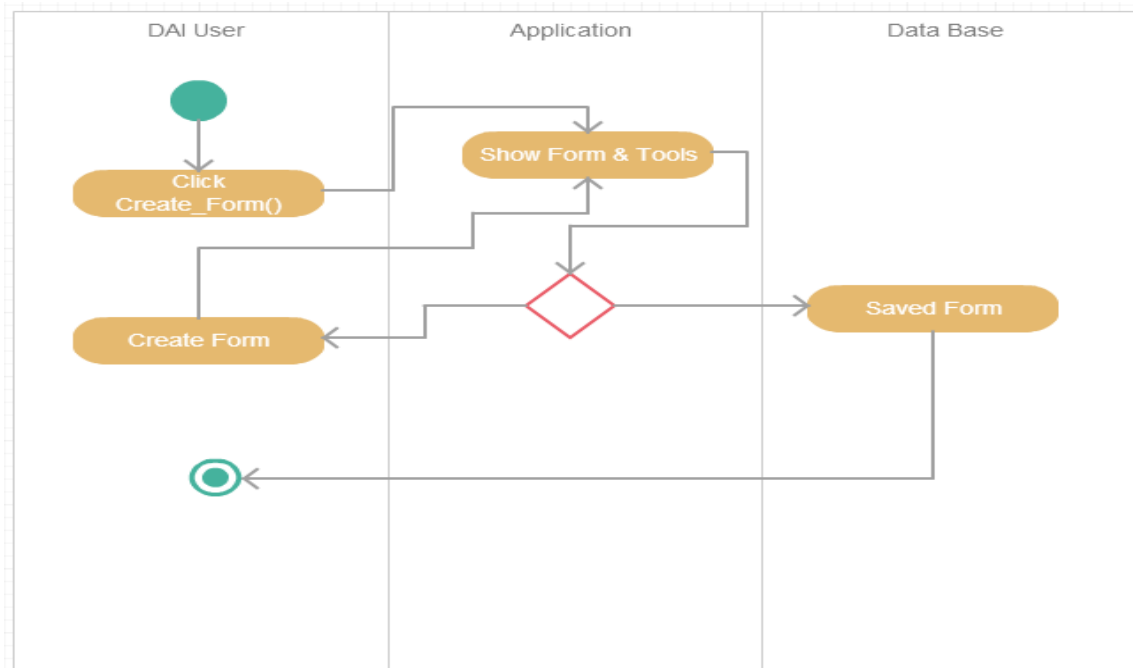


Figure 23 - Create Form Activity

Dashboard Activities

Description: This scenario describes the flow of activities required to interact with the system in the event of either publishing or seeking for a particular activity.

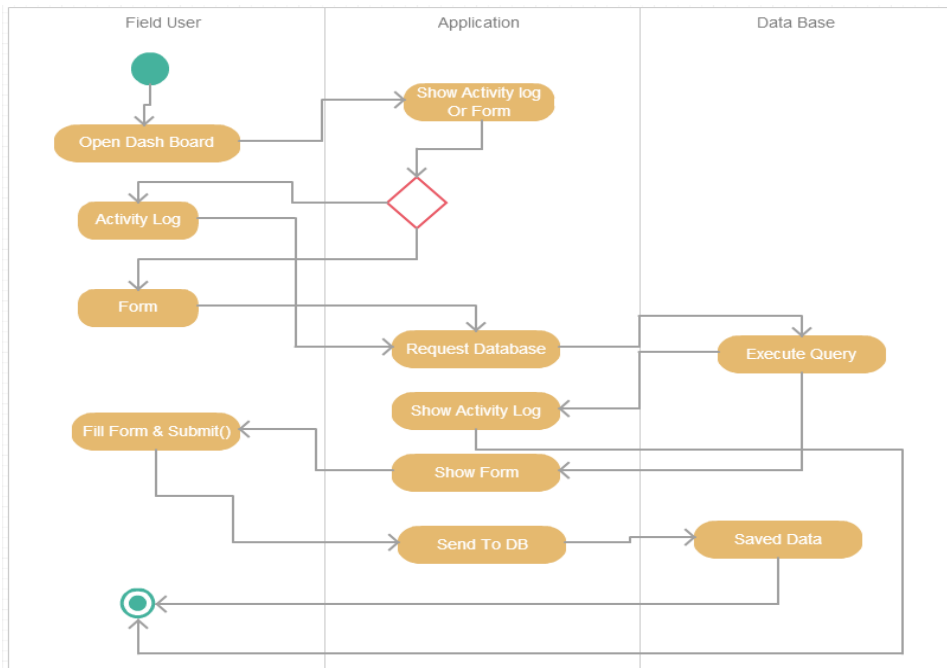


Figure 24 - Dashboard Activity

Login Activity

Description: This scenario describes the flow of activities in which the user is authenticated.

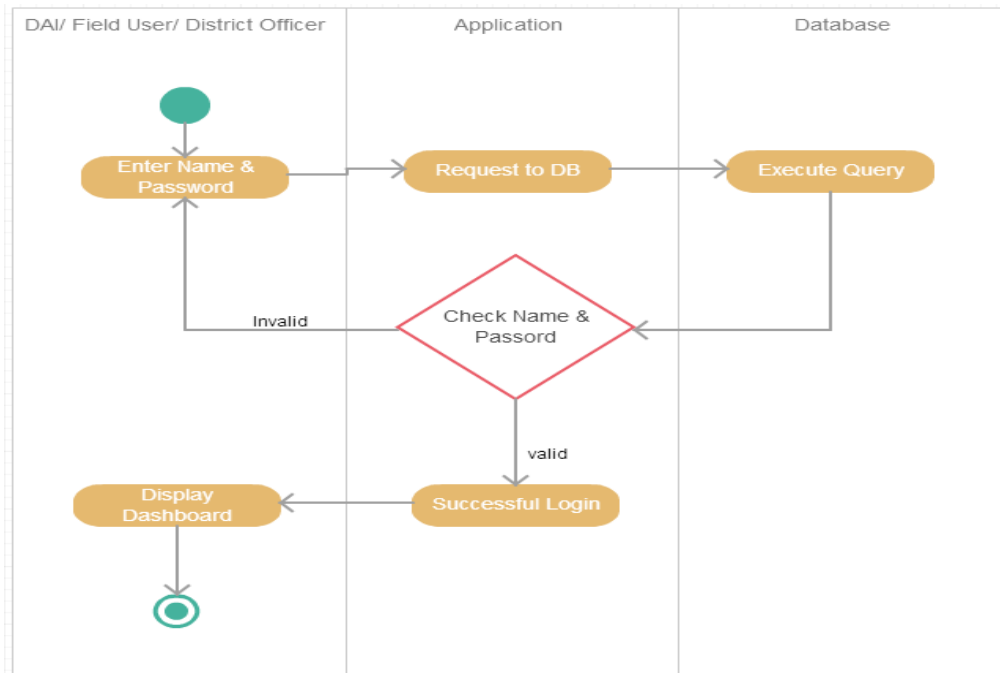


Figure 25 - Login Activity

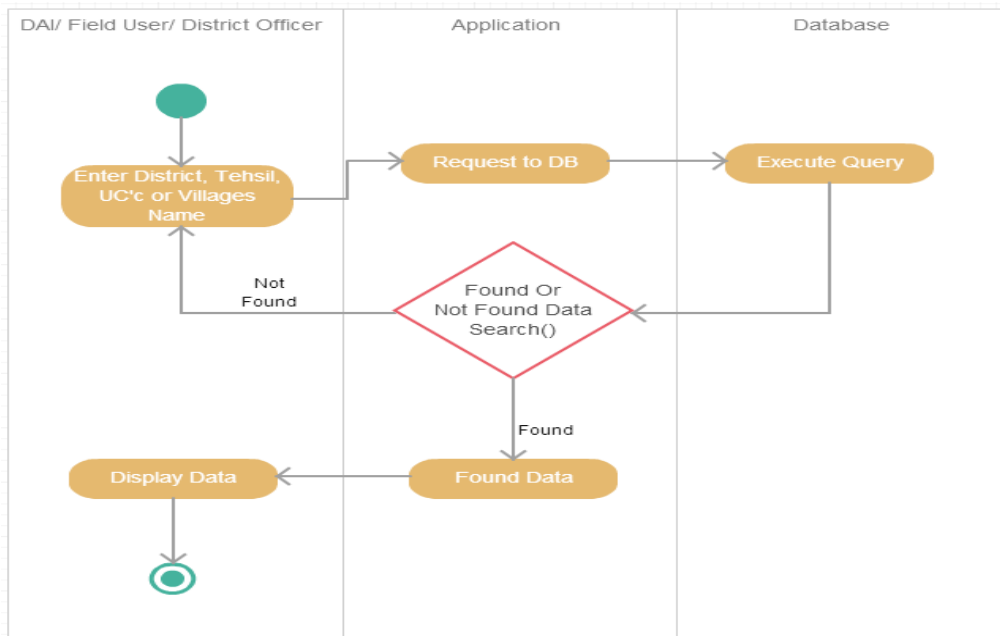


Figure 27 - Search Activity

Appendix-E

Unit Testing – Test Cases

Login Feature Testing

Test Case Name	Login Feature Testing
Test Case ID	1
Description	This feature asks the user to enter his/her credentials for login. The user can register to the Server database if not already registered. This test case is aimed to check that this feature works according to the user requirements.
Testing Technique Used	Black Box Testing
Preconditions	The system is running and connected to the Server Database
Input Values	Username string and alphanumeric characters Password string and alphanumeric characters
Valid Inputs	Valid/Authorized Username string and alphanumeric characters Valid/Authorized Password string and alphanumeric characters
Steps	Enter username Enter Password Press Sign me in button
Expected Output	The user credentials will be passed to the server for verification. The valid users will be directed to Main dashboard of the application
Actual Output	Connection String has a value of User=admin and Password = 12345
Status	PASS

Table:17- Unit Testing - Login Feature

Data Insertion Testing

Test Case Name	Plan Activity Testing
Test Case ID	2
Description	This test case is used to check the database insertion issues.
Testing Technique Used	White Box Testing
Preconditions	System is running and database is connected.
Input Values	Valid insertion string and alphanumeric
Valid Inputs	Valid insertion string and alphanumeric
Steps	<ul style="list-style-type: none"> -Post the data on server using “post” method of form -Call a specific function related to the form -Check the connectivity of server with database for particular insertion.
Expected Output	<ul style="list-style-type: none"> -Generate success message after successful insertion -Generate error message in case of failure
Actual Output	Insertion String has a value of
	Organization: Partner1 District : Lower Dir Subcategory: Blood Month: March
Status	PASS

Table:18-Unit Testing - Plan Activity

Appendix-F

Integration Testing – Test Cases

Increment 1: Administrator Module

Test Case ID	Input(s)	Initial Condition	Output(s)	Sequence of Actions	Result
4	-Username -Password	System is running and database is connected	-Administrator status logged in and redirected to main dashboard	-Webpage for login appears	PASS

Table:19-Integration Testing - Administrator Module

Increment 2: User Module

Test Case ID	Input(s)	Initial Condition	Output(s)	Sequence of Actions	Result
5	-User name -User email -Full name -Password -Role	Application is running, and on create new user screen	User is registered successfully	User registration option is selected Fill in the credentials Click on registered	PASS
6	username password	Application is running	User is signed in	Enter already registered username and password credentials	PASS

Table:: 20-Integration Testing - User Module

Increment 3: Activity Form Management

Test Case ID	Input(s)	Initial Condition	Output(s)	Sequence of Actions	Result
8	-Code in code editor and inputs in textboxes from user	System is running, database is connected - -Code is inserted in code editor -Inputs are inserted in textboxes and submitted to server	Code is compiled and output displayed in output textbox	-Data is filled -Save button is pressed for processing -Cancel is for cancelation of form	PASS

Table:21-Integration Testing - Activity Form Module

System Testing – Test Cases

Test Case 9

Test Case ID	9
Test Case name	Login
Input(s)	On Application load login screen appears then enter username and Password. Then click on Sign me in button
Output	User is presented with application dashboard
Result	Success

Table:22-System Testing - Login

Test Case 10

Test Case ID	10
Test Case name	Create User
Input(s)	Select Create user and then enter User Name, User email, Password, Full name, Role. Then click on Registered button
Output	User is presented with user listing page
Result	Success

Table:23-System Testing - Create User

Test Case 11

Test Case ID	11
Test Case name	Submit Input of Activity Form
Input(s)	Enter the inputs in the textboxes and select boxes and click on Save Activity button
Output	Activity data listed in tabular form
Result	Success

Table:24-System Testing - Submit Input Activity Form

Glossary

API	Application Programming Interface
App	Application
AS	Assumption
Black box Testing	Testing emphasizes on the external behavior of the software entity
CO	Constraints
CRUD	Create, Read, Update, Delete
DBMS	Database Management System
DEP	Dependency
FRs	Functional Requirements
GUI	Graphical User Interface
HTML	Hyper Text Markup Language
HTTP	Hypertext Transfer Protocol (HTTP) is a widely used communications protocol for communication over a computer network, with especially wide deployment on the Internet
IDE	Integrated Development Environment
iOS	Mobile Operating System created and developed by Apple
JavaScript	Client side scripting language used to create dynamic web content and user interface
MCS	Military College of Signals
NFRs	Non Functional Requirements
NUST	National University of Science and Technology
OE	Operating Environment
OS	Operating System
IMS	Information management system
REQ	Requirement
SQL	Structured Query Language
OAS	Office Automation System
QGIS	cross-platform free and open source desktop GIS application that provides data viewing, editing, and analysis capabilities.
SRS	Software Requirements Specification

UD	User Documentation
UML	Unified Modeling Language
White Box Testing	Testing emphasizes on the internal behavior of the software entity

Table 25: Glossary

Bibliography

https://play.google.com/store/apps/details?id=appinventor.ai_joriogra.HelpMe

[Accessed: 04-09-2014]

[Accessed: 08-09-2014]

Office automation system Pakistan army

Digital Mapping System Pakistan army

www.Aurat Foundation .com

www.SAPPK.com

SPO website

[Accessed: 09-09-2014]

^[1] Wikipedia “database management”

[Accessed: 04-03-2015]

[Accessed: 12-03-2015]

<http://w3school.com/php>

www.tutorialspoint.com/php/

www.html5andcss3.org/

<https://www.youtube.com/watch?v=9gTw2EDkaDQ>

www.htmldog.com/guides/css/beginner/