Family Tree Maker Using PhoneGap

Department of Computer Science & Engineering,
B.V.B.C.E.T Hubli

May 6, 2013

ACKNOWLEDGMENT

Gratitude takes three forms, A Feeling from the heart an expression in words and giving in return. We take this opportunity to express our heart full feelings.

We are very much thankful to **Prof. K.R.Biradar**, Head of the Department of Computer Science and Engineering, B.V.B College of Engineering and Technology, Hubli for providing us an opportunity to undertake this project in the form of the final year project.

We take this immense opportunity to thank **Mr. Varun Kulkarni and Mr.Gururaj Pujar**, CLOGIK, Bangalore for providingus an opportunity to work on the industrial project and there timely mentoring will definitely have impact on our project.

We wish to place on the record our deep sense of gratitude to our beloved Project Coordinator **Prof.Vijayalaxmi M.**, Associate Professor, Computer Science and Engineering for extending us constant support and encouragement.

The progress of our project till design phase, right from very beginning would not have taken shape and proceeded in a uniform, disciplined and smooth manner without the guidance of our Project Guide **Prof. Manjula Pawar**, Associate Professor, Computer Science and Engineering who helped in analyzing our efforts in right direction.

We also take this opportunity to our parents and friends who constantly provided us the everlasting moral support, encouragement and for suggestive critics.

Akshata Shirish Nayak
Amit Chigadani
Anusha Shetty
Dev Ashish

ABSTRACT

A device / platform independent Family Tree Maker. Shall be able to add details of individuals and their relationship (Father, mother, Brother, Sister, Spouse, Son, Daughter). Software should show a graphical representation of the family linkages. Based on the Date of Birth or Date of Death, one shall be able to zoom in to or zoom out of the timeline and see the change in the Family tree according to the change in the time line (years).

Keyword :*FTM-Family Tree Maker*

Contents

List of Figures

List of Tables

Chapter 1

PREAMBLE

1.1 Introduction

The DIGITIZE FAMILY is a mobile Application that can display a complete graphical representation of ones family lineage. With this application, you can see both ancestor and descendants in form of a family tree where each node reperesents a member of the family.

Every user shall add images and a relevant details of members of the family which can be keyed in and viewed. In this Key parameters that shall be captured: Name, Surname, date of Birth, Sex, Date of Death (if applicable), Relation, Photo. It shall allow customizable fields to be added like Gotra, Family Lineage etc.This application will also provide other features such as Setting of reminder,zoom in and zoom out and user can view entire timeline of each member of the family.

1.2 Problem Definition

A mobile device/platform independent Family Tree application. That shall be able to add details of individuals and their relationship.Based on the Date of Birth or Date of death.In this application one shall be able to zoom in to or zoom out of the timeline and see the change in the Family tree accordingly. The user will be able to set reminder as well as view a relationship path.This application will be running on two different operating system namely Android and windows with the help of a common platfrorm PhoneGap.

1.3 Objectives of the project

- Multiple Family creation.
- Graphical representation.
- Import Tree as a Node (Import a family as a Node to an existing family).
- Adding family photos, images of records, videos, and more.
- When you locate a family member you shall quickly merge your findings directly into your tree.
- Each family tree shall be stored in different file.

Chapter 2

SYSTEM STUDY

2.1 Existing System

The existing systems are geni for windows operating system and apps like ancestory for android.

2.1.1 Advantages Of Existing System

- The existing sytem are web-based system which have links and access over the web.

2.1.2 Disadvantages Of Existing System

- Member exchange from a family feature not available.
- Searching a family by unique name feature not available.
- They are only available for the users of a particular operating system.

2.2 Proposed System

The new technique namely,Phone gap, which considers the requirements of usability and effectiveness simultaneously. Application shall show a graphical representation of the family linkages.Based on the Date of Birth or Date of Dead, one shall be able to zoom in to or zoom out of the timeline and see the change in the Family tree accordingly.

2.3 Advantages Of Proposed System

- Application can be ported on 7 operating systems.
- Photos and videos can be added in addition with documents or attachments.
- Import Tree as a Node (Import a family as a Node to an existing family).
- When you locate a family member you shall quickly merge your findings directly into your tree.

2.4 Constraints

- This will be a Standalone application which shall run on mobile devices.

Chapter 3

SOFTWARE REQUIREMENT SPECIFICATION

3.1 Introduction

3.1.1 Purpose

The purpose of this document is to provide a very high level technical overview for the mobile application to be known as Family Tree. Specifically this document gives the details of the functional and data requirements for the application. It helps user to keep their family linkage, their information, photos, videos etc digitally in their pocket

3.1.2 Scope Of Project

- The proposed system shall help the user to keep information about the family.
- Shall help to strore photos of members and events.
- Shall allow to view the events in accordance to the timeline.
- Shall allow to store and view messages.
- Shall allow to set and view reminders.

3.1.3 Intended Audience

- Project managers from C LOGIK- They can know what all functions will be implemented in the application which would be part of their existing software LIFE OPTIMISER.

- Developers- We will have clear understanding of the requirements while designing the project.

- Marketing Staff- It will help C LOGIK company to understand the project before marketing the product.

- End Users- Any family members, who can have a clear understanding of the product which they are going to use.

3.2 Overall Description

3.2.1 Product Perspective

This product would be a part of the existing software Life optimizer which ensures better health for you and your family, manages your investments and finance, ensures happiness and brings productivity in all aspects of your life. This product would add a feature to it by creating the family tree of the user.The Family Tree application is an Application shall provide a quick access to keep in touch with family.

- By entering the details of his family members.

- By adding family photos, images of records, videos, and more.

- When you locate a family member you shall quickly merge your findings directly into your tree which you shall print or share with others.

3.2.2 User Classes And Characteristics

- Family member- The user shall know how to use the Smartphones like android, windows phone, ios and symbian.

3.2.3 Assumptions

- The user shall be familiar with Smartphones like android, windows phone, ios and symbian.

3.2.4 Dependencies

- This system is dependent on the operating system that it works and the dimensions of the screen of the phone/device.

3.3 Requirement Specification

3.3.1 Functional Requirements

- This application will be used to have relationships with the family member spanning any distance.

- The User shall create and login into his account with Login ID and password.

- The User of this application shall create Multiple Family Trees.

- The User of this application shall Export the Tree (exchange with other family tree).

- The User of this application shall Import Tree as a Node (Import a family as a Node to an existing family).

- The User shall be able to search any of his family members with search option.

- The User shall be able to store and view messages.

- The User shall be able to set and view reminder.

- The User shall be able to add family photos, images of records, videos, and more.

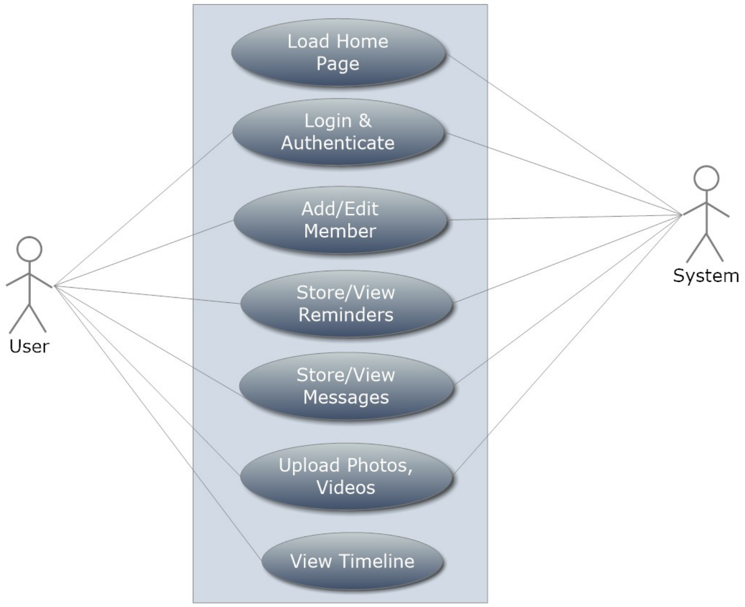


Figure 3.1: Use-Case Diagram

3.3.2 Nonfunctional Requirements

- Performance Requirements
  - The User shall be able to search any of his family members within a second.
  - The application shall load the family tree within fraction of seconds regardless of the size of family tree.
- Safety Requirements
  - The software should take periodic backups of the scans to prevent the loss of any data.
- Security Requirements
  - The User shall be provided with the Login ID and password to keep his profile secured.

3.3.3 Software Quality Attributes

- Functionality
  This software will deliver on the functional requirements mentioned in this document.
- Availability
  This system will be always available once downloaded .
- Flexibility
  It provides the users to switch from touch screen to keypad and vice versa.
- Learnability
  The software is very easy to use and comes with documentation which reduces the learning work.GUI is user friendly and thus, user can use this application very easily.
- Portability
  This application is supported by the mobile device which has the softwares mentioned above.

3.3.4 User Documentation

Provides users with a step by step process of how to install the application .The document assists the user how to use the product to meet the specifications. (Ex : help menu)

3.4 Software and Hardware requirement Specification

3.4.1 Software Requirements

- Operating System : iOS, Android, Blackberry, Windows Phone, Palm WebOS, Bada and Symbian.
- Software Tools : Phone gap.
- Data Base : SQL Lite.

3.4.2 Hardware Requirements

- Processor : 600MHz or Above
- Main Memory : 512MB or Above
- Disk Space : 10 MB or Above

3.5 User Interface

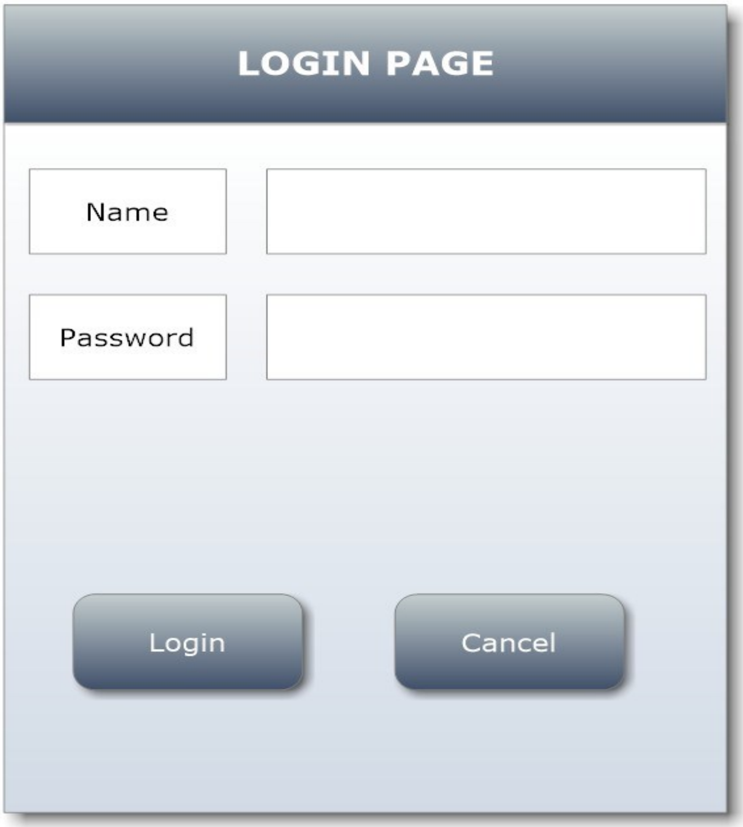


Figure 3.2: Login

The first page shall appears as illustrated in figure3.2, which is the user authentication page where the user has to enter the respective username and password, once validated the user can sign into the homepage.

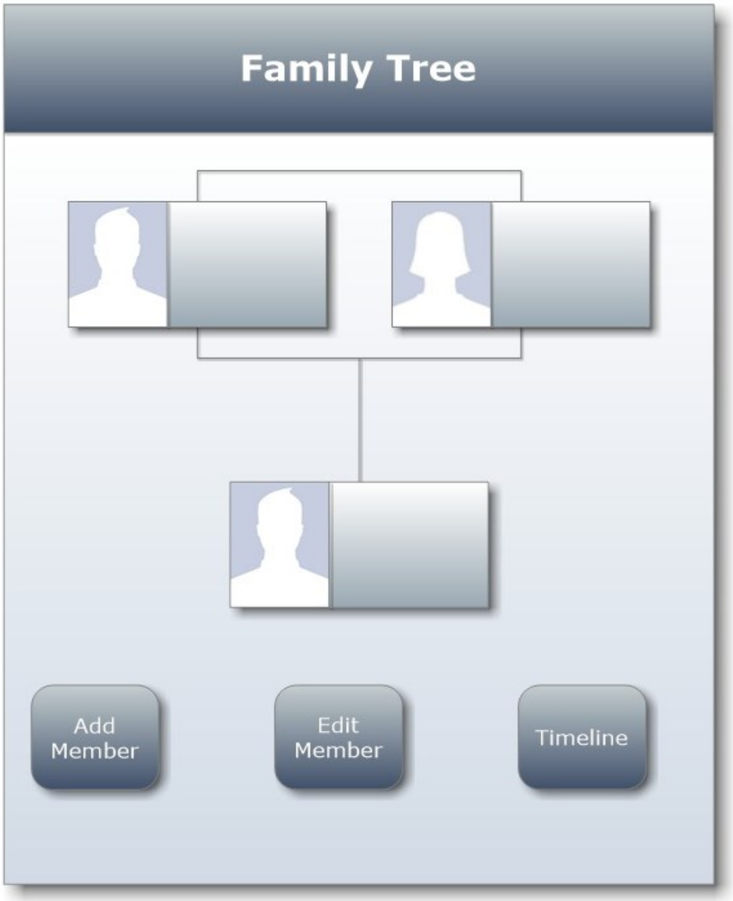


Figure 3.3: Family Tree Screen

This is the homepage where the family tree shall appears as in figure 3.3 and has various features to enhance the tree once scrolled through node of tree such as zoom in, zoom out. And various other options such as merge tree, delete a member, add a member and view timeline of events of family members.

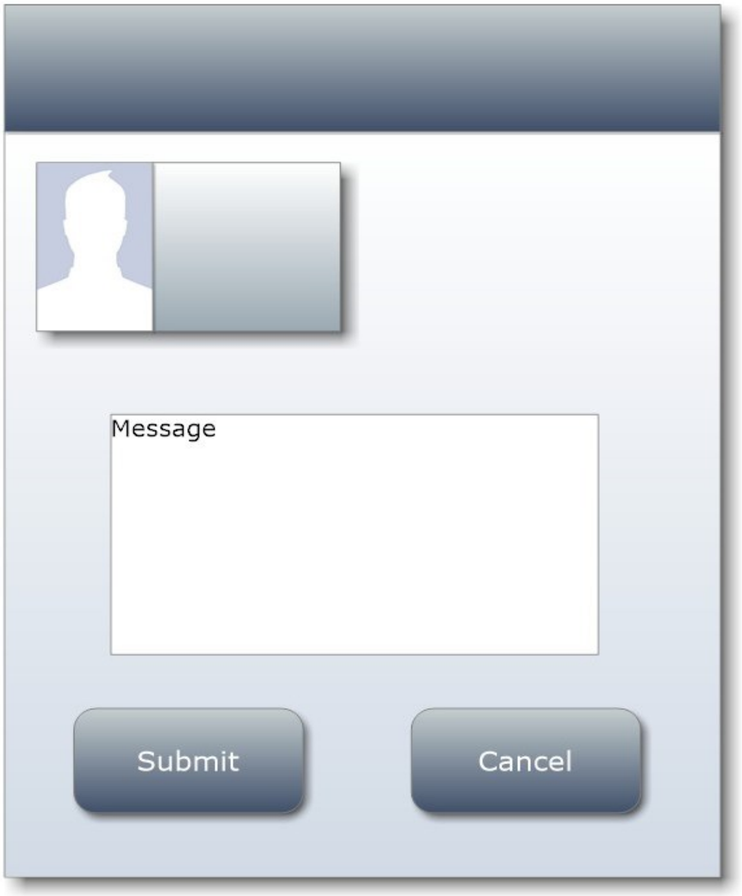
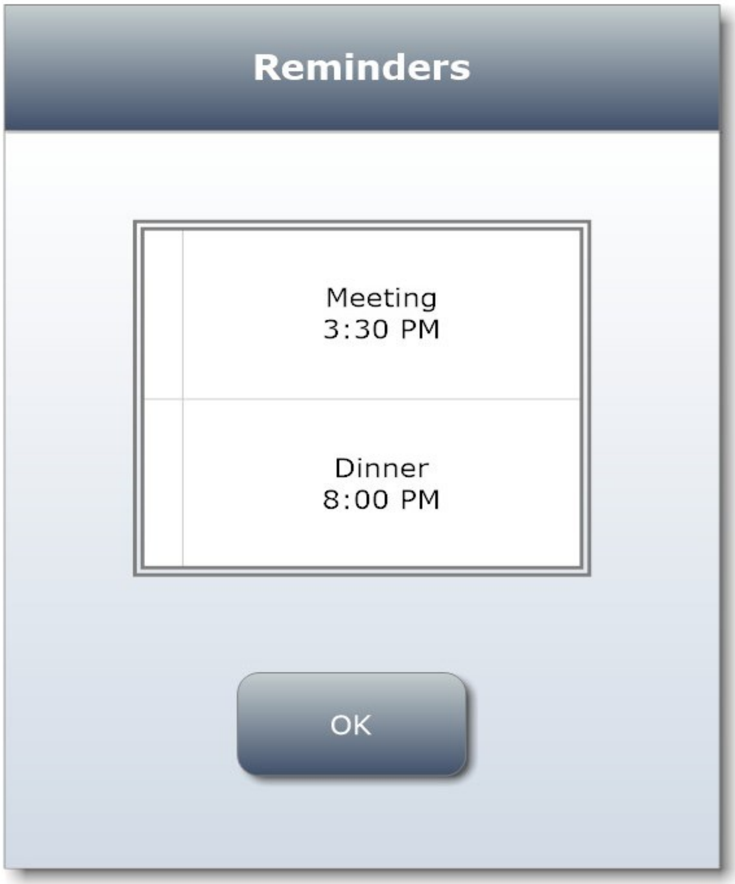


Figure 3.4: Store messages

This feature is for each family member where the user can write a message regarding a member of family and save to view it later shall appear as in figure 3.4.



Reminders

Meeting
3:30 PM

Dinner
8:00 PM

Figure 3.5: Store and view reminders

A calendar with all the events such as birthdays, anniversary etc. of family members saved to be viewed on the calendar shall appear as in figure 3.5.

3.6 Software Interfaces

- Phone Gap (Apache Cordova) It enables software programmers to build applications for mobile devices using JavaScript, HTML5 and CSS3.
- JDBC-ODBC is used as an interface for database connectivity.

Chapter 4

RESOURCE REQUIREMENTS

4.1 DEVELOPMENT ENVIRONMENT

4.1.1 HARDWARE REQUIREMENTS

Computing Requirements

- Processor : 832 MHz and Above
- Main Memory : 3 MB
- Disk Space : 10 MB or above

4.1.2 SOFTWARE REQUIREMENTS

Development OS : Windows XP, or Windows 7.

Software engineering tools required for requirement management.

- Design: SmartDraw 2010.
- Construction: PhoneGap 2.2.0, Eclipse 3.4+, Xcode 4.5+, Apache Ant.
- Documentation : MikTeX 2.9.

4.2 DEPLOYMENT ENVIRONMENT

Software Requirements

Software's such as Database Management System will depend upon the operating system of smart-phone.

4.3 ACCEPTANCE TEST PLAN

Test case ID	Requirement	Input	Expected output	Actual output
01	Login Page	User has to input user name and password.	The user will be prompted to the family tree page.	-
02	Family Tree Page	User has to create a new member or update	The tree shall be updated.	-
	tree.			
03	Store/View messages	User has to select particular family member to view the message.	The messages will be displayed.	-
04	Store and view reminders	User has to feed the events.	The reminders pop up for the set event.	-
05	Import tree as a node	User has to enter the family name that has to be imported.	Display of Updated family tree after importing.	-
06	Export a node (family member)	User has to select member which has to be exported.	Export successful	-
07	Upload photos, videos and documents	User has to select the files from the system	Timeline will be updated with the photos, videos or documents	-

Table 4.1: Test Plan Table

Chapter 5

SYSTEM DESIGN

5.1 Introduction

5.1.1 Summary

Purpose

The purpose of this document is to explore the logical view of architecture design, sequence diagram, data flow diagram, and user interface design and exception conditions of the software.

Scope

The scope of this design document is to extract the best possible design for the software.

Document's intended audience

- Project managers from C LOGIK- They can know the design of our application which would be part of their existing software LIFE OPTIMISER.
- Developers- We will have clear understanding of the design before implementing the project.

- Marketing Staff- It will help C LOGIK company to understand the design of the project before marketing the product.
- End Users- Any family members, who can have a clear understanding of the product which they are going to use.

5.1.2 Design Goals and Non Goals

- To optimize the design for the most frequent or important tasks.
  Adding photos, messages and reminders of a person would be frequent tasks rather than creating the tree or adding a member.

- To make the interface accessible and visible to users. Application shall provide proper choices for the users to easily access the information within the interface.

5.1.3 Common Scenarios

- Login
  Accepting username and password.
  Verification.
  Authentication.

- Create tree
  Creation of new nodes.
  Creation of new links.

- Search member
  Traverse along the tree
  Find for a member

- Insert node
  Accept the information
  Search for appropriate position in the tree
  Add the member to the tree

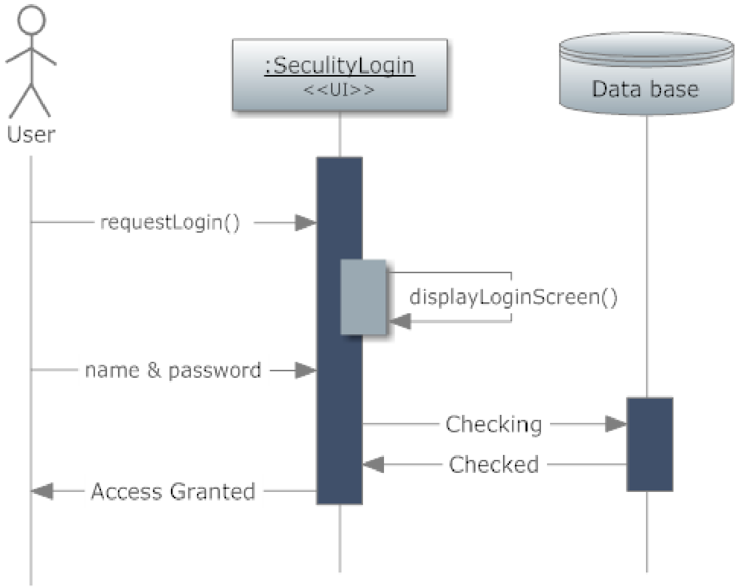


Figure 5.1: Log-In Scenario

- Delete node
  Member information to be deleted
  Search for appropriate member in the tree
  Delete the member to the tree

- Merge node
  Search for both the nodes
  Merge the nodes
  Display the tree

- Merge tree
  Search for the node where the tree must be linked
  Search for tree
  Merge the tree to the node
  Display the tree

- Zoom

Zoom in operation
Zoom out operation

- Timeline
  Display of the events in accordance to the years of events.

- Reminder
  Set the reminders
  View the reminders
  Trigger the reminder

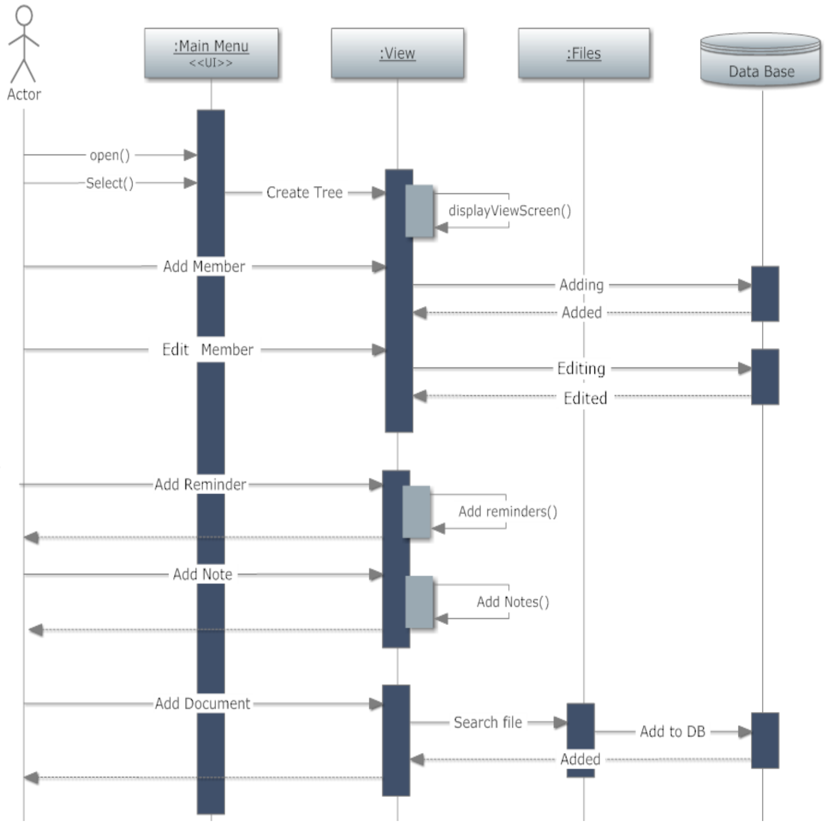


Figure 5.2: Create Tree Scenario

5.2 Architectural Design

5.2.1 Block Diagram

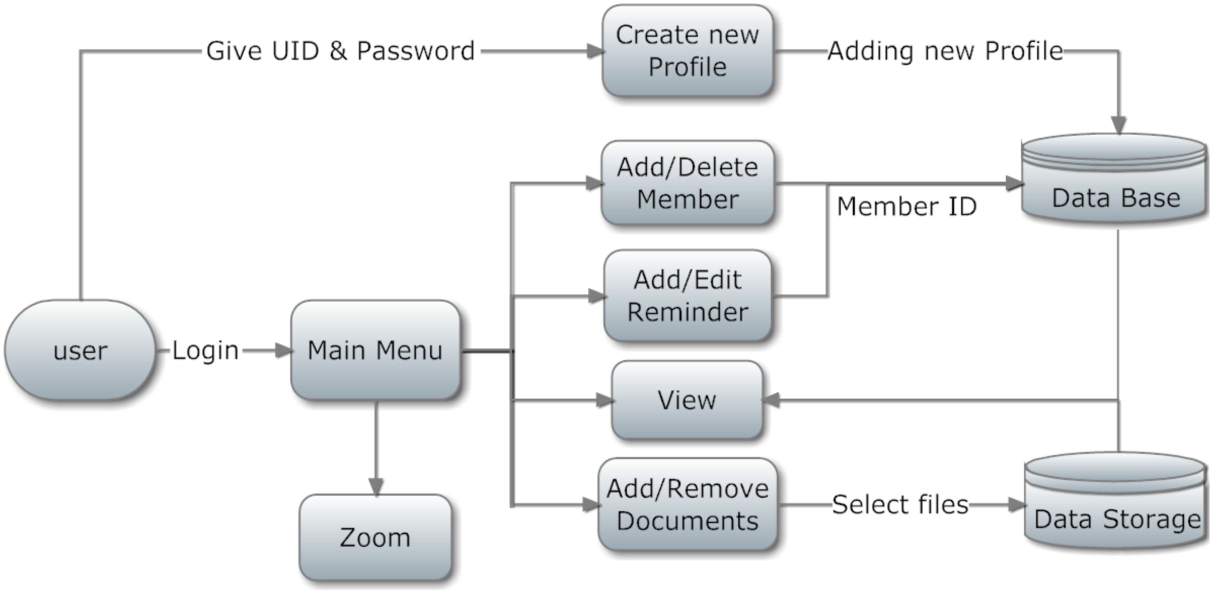


Figure 5.3: Block daigram

The above block diagram speaks about module wise functions.It shows interaction of main menu with different modules like add/delete member, merging, editing, zoomin, zoom-out and the database and data storage linking.

5.2.2 Logical View

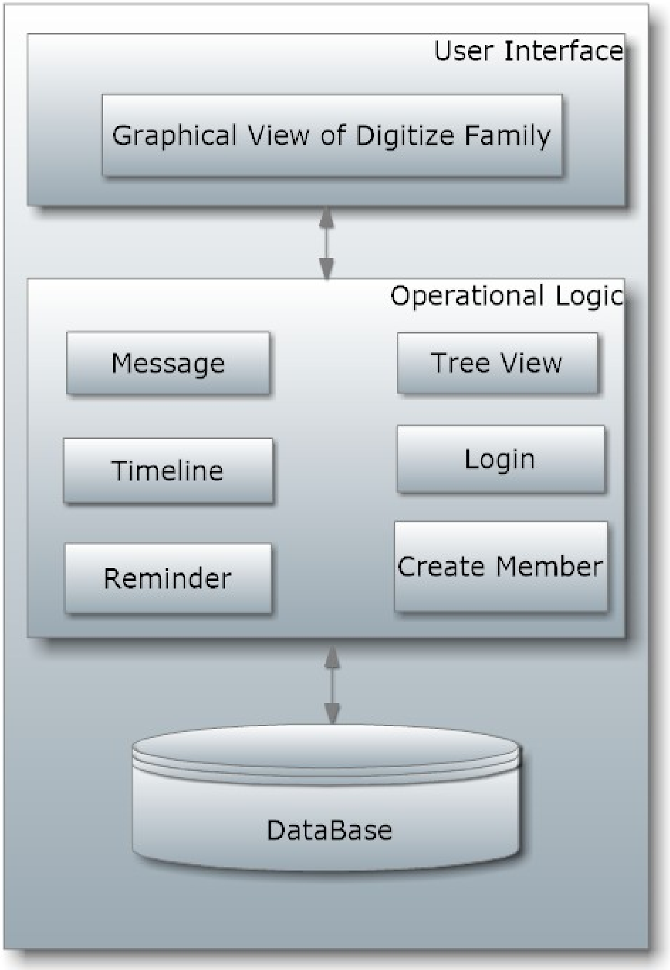


Figure 5.4: Logical view daigram

The Figure shows the interaction with the family tree home page and the different modules like updating, merging, editing, zoomin, zoomout and interns the interactions of the different databases with the modules mentioned above.

5.3 Detailed Design

5.3.1 Data flow Diagrams

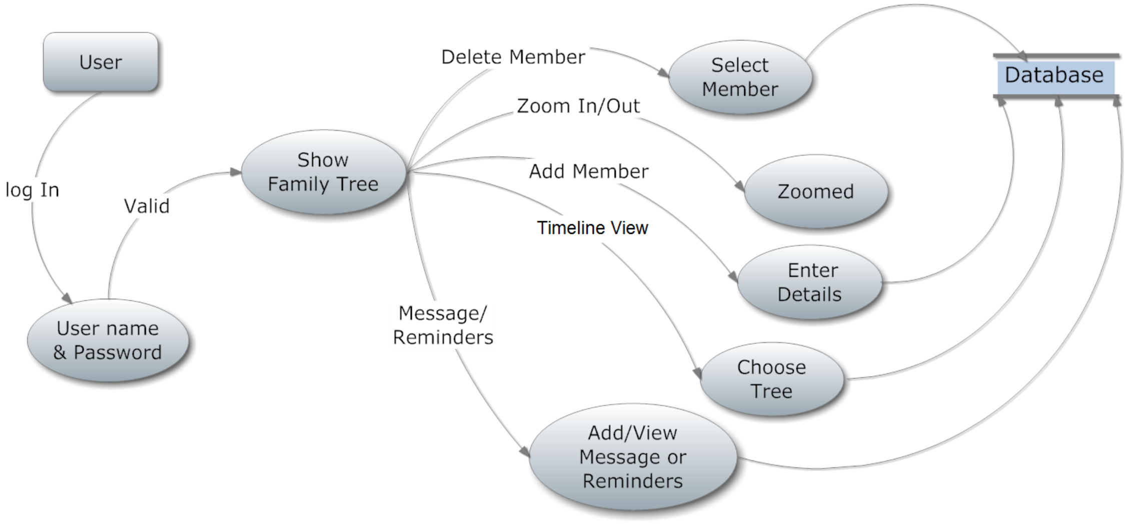


Figure 5.5: DFD

The Figure shows the detail design of the family tree in terms of the data flow diagram; it shows all the operations and the process interactions between them and the databases. The fig shows all the operations of the application like adding, viewing messages or reminders, adding deleting members of the tree etc.

5.3.2 Flow Chart

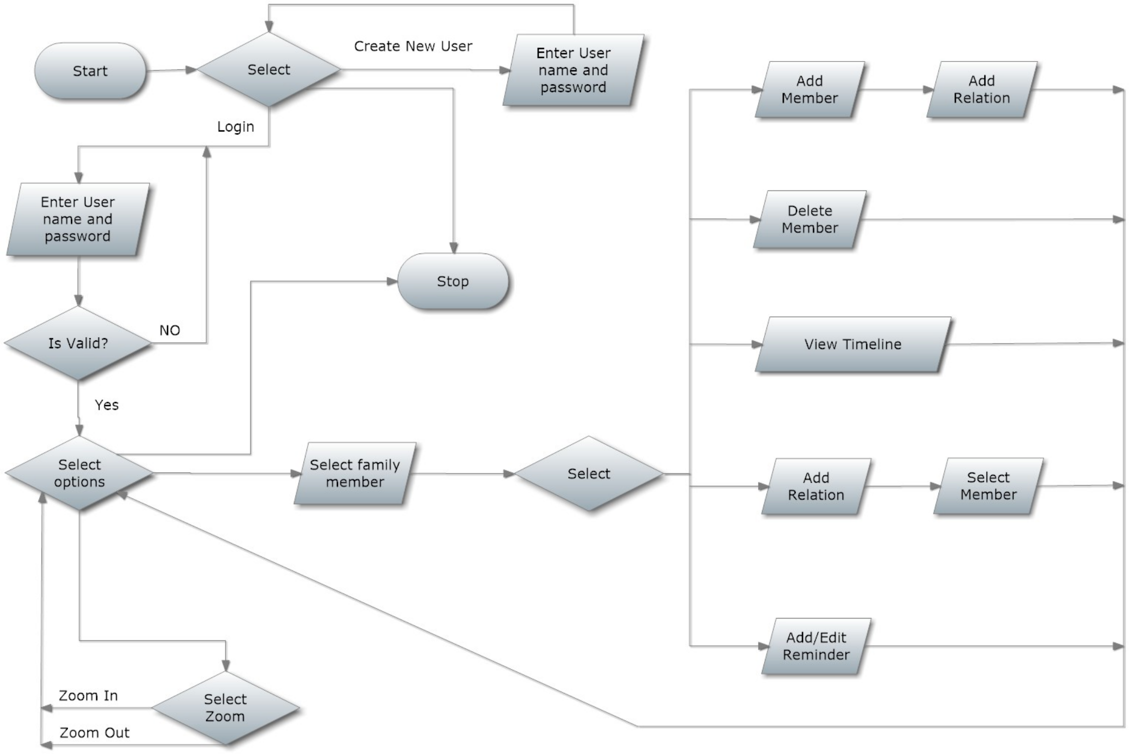


Figure 5.6: Flow Chart

The Figure shows flow of control for the Family tree. The fig shows all the operations and the linking between them.

5.3.3 User Interface Design

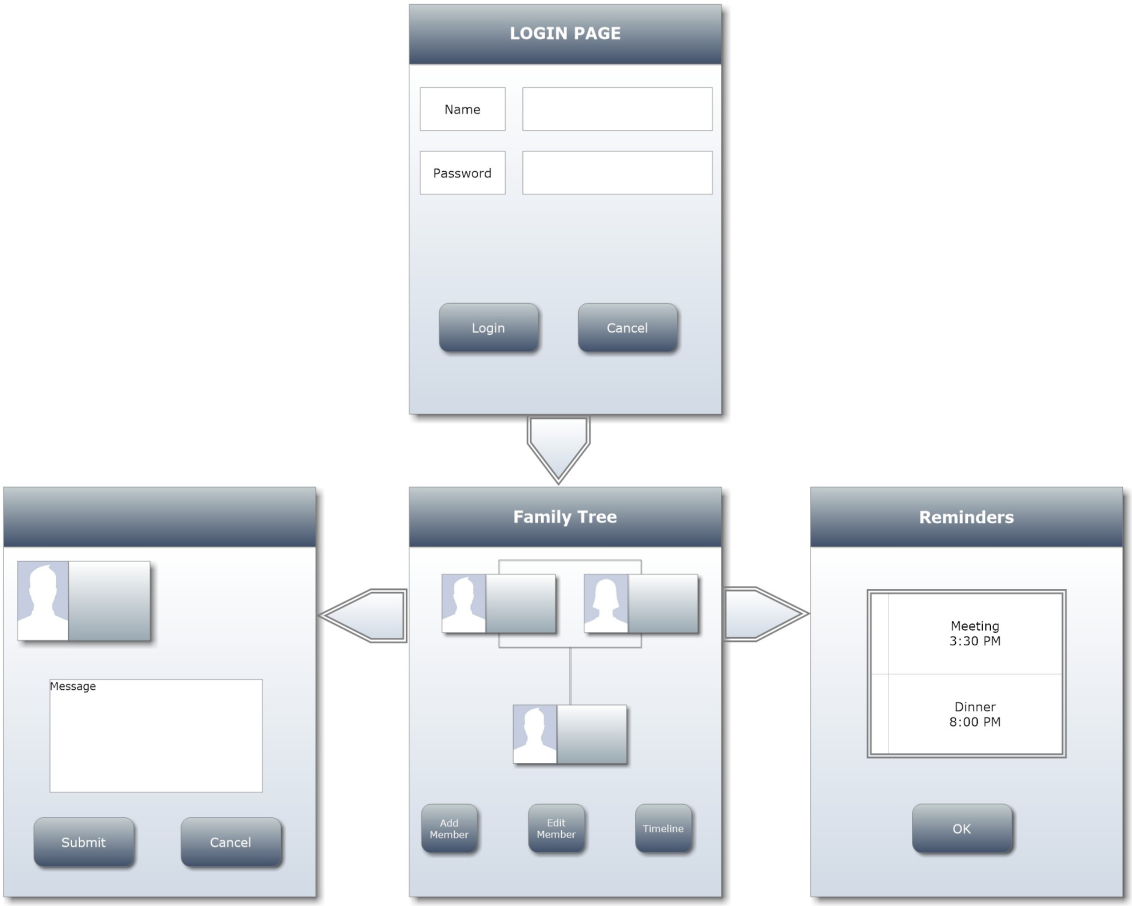


Figure 5.7: Interface Navigation

The figure shows all the user interfaces and screen layouts of the screens along with the hierarchy n flow of screen layouts on the press of few buttons and the actions performed on the screens.

5.3.4 Database Design

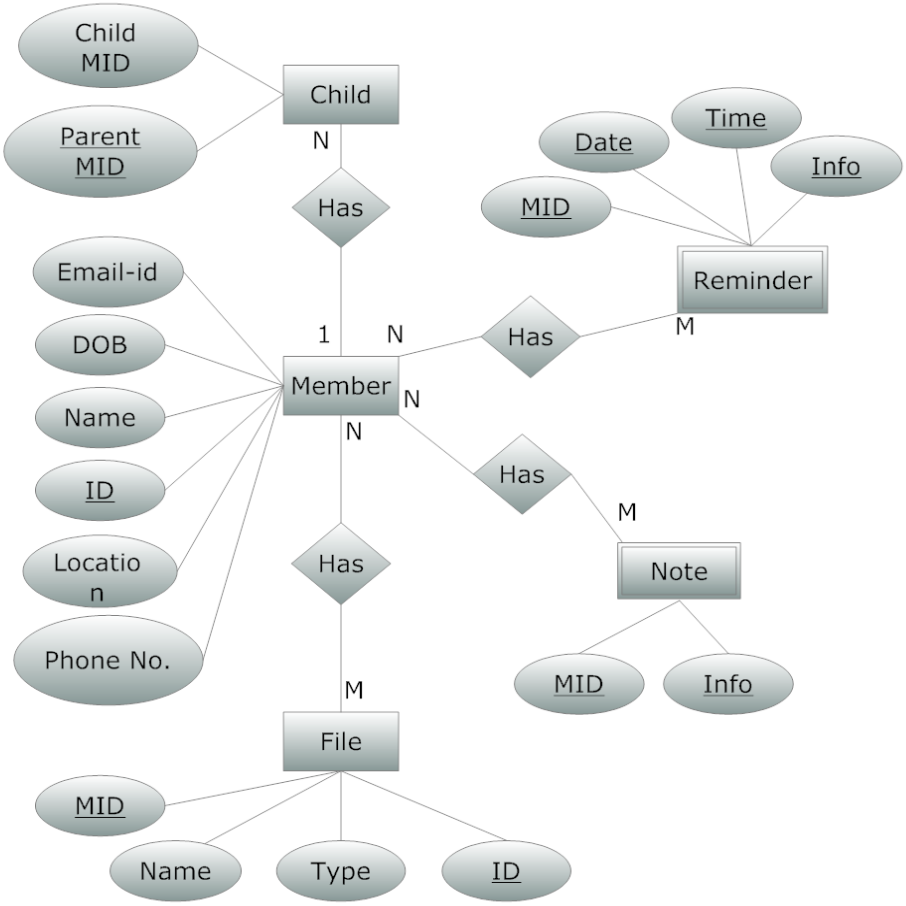


Figure 5.8: Entity Relationship Diagram

The figure shows the ER diagram of the different tables used in the tables of the different databases. Here member child and file are the relations and note and reminders are derived relations. The different relations are with a relation has.

The derived relation note has a partial key of mid and info and reminder with the entire attribute contributing to the partial key.

5.3.5 Logging

- The application shall store the database in its own device.
- Separate file is maintained for each individual family tree so that it can be easily portable to another device.

5.3.6 Exceptions

- Providing incorrect password for selected user-name.
- Deleting a person by name who does not exist in the family tree.
- If Atttached file is deleted.

5.3.7 Localization

Phone Gap (Apache Cordova)

It is an open-source mobile development framework produced by Nitobi, purchased by Adobe Systems. It enables software programmers to build applications for mobile devices using JavaScript, HTML5 and CSS3, instead of lower-level languages such as Objective-C. The resulting applications are hybrid, meaning that they are neither truly native nor purely web based.

- It reduces the cost of developing and maintaining applications on multiple platforms.
- Applications are portable from one OS to another.
- Phone Gap currently supports development for the operating systems Apple iOS, Google Android, HP webOS, Microsoft Windows Phone, Nokia Symbian OS and BlackBerry.
- JDBC-ODBC is used as an interface for database connectivity.

5.3.8 Deployment diagram

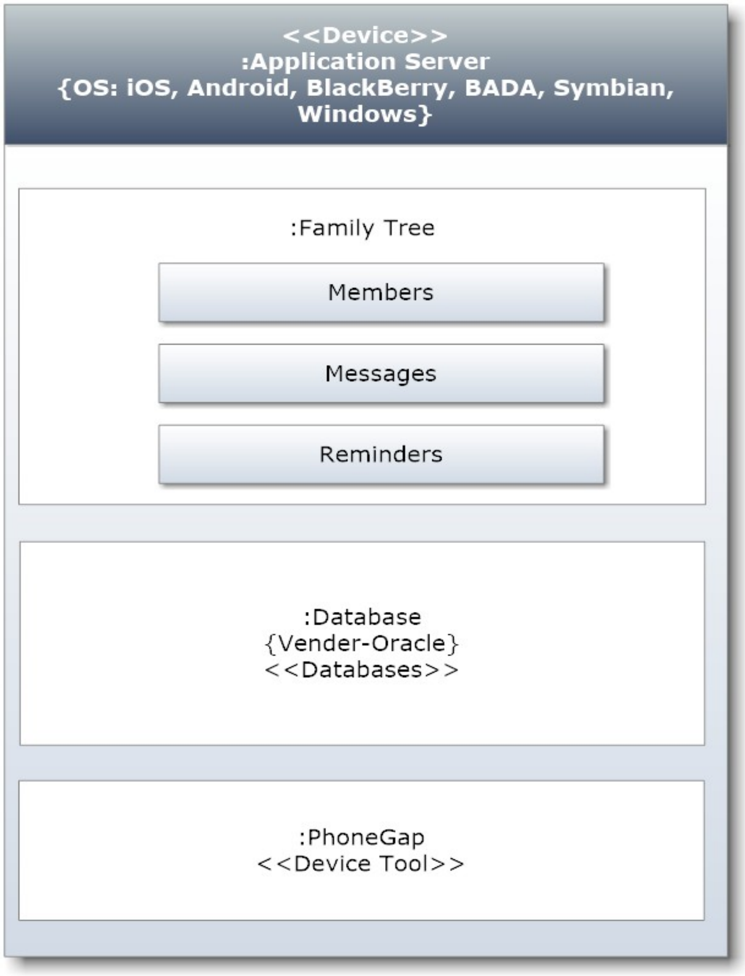


Figure 5.9: Deployment diagram

The above deployment diagram shows the different database and the application server and the different tools used in this application, database and phonegap tool.

5.3.9 Design Decisions

- Tool : Phonegap tool
- Operating system : Apple iOS, Google Android, HP webOS, Microsoft Windows Phone, Nokia Symbian OS and BlackBerry.
- Languages : JavaScript, HTML5 and CSS3
- Database : Sq lite
- Each tree is stored and maintained in a different file.
- User inputs either by touch screens or the key pad of the mobile devices.

5.3.10 Open issues

Each operating system uses different databases so making it general for all languages.

Chapter 6

IMPLEMENTATION

6.1 Module 1:Welcome

6.1.1 Homepage

- **Description:**It is the first page that appears when the application is launched.
- **Input:**User clicks the Go button.
- **Output:**The user is directed to the sign-in page.

6.1.2 Sign-in Page

- **Description:**In this page if the user is the new user ,this user can sign-up to this application by entering all the given fields .But if the user has already a account then the user can directly sign-in into the application.
- **Input:**
  - **New User:**The first time user will enter the fields such as the user name and password.
  - **Exixting User:**The user just clicks on the Already existing user.
- **Output:**
  - **New Use:r**The user will be directed to the Create Account page.
  - **Exixting User:**The user will sign in into the main page where he/she has his/her family tree which was Already created.
- **Pseudo Code:**

Algorithm 1 Sign-in Page

```
if existing user then
  Enter UID and PWD
  if correct UID and PWD then
    Goto Tree page
  else
    print text
  end if
else if new user then
  Click on the sign up
  Enter details
  Stored in the data base
  Print new member!
  Goto tree page
end if
```

6.1.3 Create Account

- **Description:**The first time user has to fill in the fields to create the family tree.
- **Input:**Some Information of user such as firstname, lastname, date of birth, gender etc.
- **Output:**
- **Pseudo Code:**

Algorithm 2 Create Account

```
Read all the data entered
if All required fields entered then
    Add to the database
else
    Alert the user
end if
```

6.2 Module 2:Tree

6.2.1 Blood Line

- **Description:**When the user chooses particular node via click and touch the relationship between the user and that particular node or the member is highlighted.
- **Input:** The user chooses the particular member of the tree (node).
- **Output:**The path between the user and the chosen node is highlighted by the colour yellow.
- **Pseudo Code:**

Algorithm 3 Blood Line

```
Click on the member in the tree
Retrieve his or her UID
For Ancestor
while FID and MID not equal to NULL do
    Goto FID and MID of member
    Highlight them
end while
For descendants
if UID is Male then
    Check all nodes with FID=UID
end if
if UID is Female then
    Check all nodes with MID = UID
end if
Highlight the path
Add highlighted nodes to CID []
for ( i, CID[i] not equal to NULL, i++) do
    if CID[i] is Male then
        Check all nodes with FID=CID[i]
    else if CID is Female then
        Check for all node with MID = Cid [i]
    end if
    Highlight them
    Add the highlighted node to CID
end for
```

6.2.2 Add a member

- **Description:**The user of this application can add members to the existing family tree to grow the tree further.
- **Input:**The user chooses the particular node in the tree and then the different buttons are displayed like add father, add mother, add brother, add sister etc. The user will choose a button as per which member he/she wishes to add .The user will then enter all the details in the given fields.
- **Output:**The node(family member) in a tree is created with all the details entered by the user.
- **Pseudo Code:**

Algorithm 4 Add a member

```
Enter the details of the members
Data stored in data base
A node is created.
```

6.2.3 Create a tree

- **Description:**In this application a tree for the user is created depicting all the relationship, here each member is the node of the tree.
- **Input:**Details of family members stored in Database.
- **Output:**Graphical representation of the family tree as pedigree chart.
- **Pseudo Code:**

Algorithm 5 Create a tree

```
while not end of data do
    Read from the database
    Send the read data to Json
    Json reads it
    Display tree from older node
    Create new node if spouse exists
    Add to the already created node
end while
```

6.3 Module 3:Features

6.3.1 Set Reminder

- **Description:**The user will be able to set reminder on any particular day and the user will be alerted for the same on the set date
- **Input:**The user will choose the day on which he/she wishes to set reminder and in the note field the user will enter the reminder note.
- **Output:**The user will be alerted on that day when logged in with a dialog box appearing which will display the date with the note set.
- **Pseudo Code:**

Algorithm 6 Set Reminder

```
Read date
Read the reminder to be set(message)
Add to the database
On Login
if current. Date=reminder. Date then
    Alert the user
end if
```

6.3.2 Timeline

- **Description:**The user can view the entire timeline of the family tree with the timeline from birth to death of any member of the family.
- **Input:**Details of family members stored in Database.
- **Output:**Graphical representation of the family tree as pedigree chart.
- **Pseudo Code:**

Algorithm 7 Timeline

```
while not end of data do
    Read from the database
    Send the read data to Json
    Json reads it
    Display tree from older node
    Create new node if spouse exists
    Add to the already created node
end while
```

Chapter 7

TESTING

Software testing is a technical process to recognize and ensure the accuracy, the security, totality and quality of newly developed computer software and is performed to verify that the completed software package functions according to the expectations defined by requirements/specifications.
The reasons to carry out the Testing are as follows

- To check whether the tree is created properly with all the relationships shown exactly as the details entered by user.
- To check whether the relationship path shows all relationship accurately.
- To check whether all the features work accurately without accepting and as well as showing the warnings for the invalid entry.
- To compare performance of different data entered into the database.

7.1 Test Cases

A test case normally consists of an unique identifier, requirement references from a design specification, preconditions, events, a series of steps (also known as actions) to follow, input, output, expected result, and actual result. Clinically defined a test case is an input and an expected result. This can be as pragmatic as for condition x your derived result is y, whereas other test cases described in more detail the input scenario and what results might be expected. It can occasionally be a series of steps but with one expected result or expected outcome.

7.1.1 Test Cases for Unit Testing

Test case ID	Requirement	Input	Expected output	Actual output	Test States	Test Priority
01	Family Tree Home Page	Click the GO button to get in to the login page.	User will be prompted to Authentication page.	User is directed to authentication page.	Pass	4
02	Authenticated login Page	Enter user name and password.	User will be prompted to the family creation page.	User is directed to the family tree.	Pass	2
03	Store and view reminders	Feed the important events as he wishes and set reminders.	Reminders pop up for the set event.	User can store reminders and its see them.	Pass	3
04	Graphical representation of tree	Click on the trees node.	Blood relatives of node will be highlighted.	Blood relatives of node are highlighting.	Pass	1

Table 7.1: Test Cases for Unit Testing

7.1.2 Test Cases for Acceptance Testing

Test Id	Test Condition	Expected Output	Actual Output
1.	Enter into the family tree login page by clicking on the GO	Sign up page for new user and login page for existing user is displayed	Sign up page for new user and login page for existing user is displayed
2.	Input username and password, if new user has to create account.	The user will be prompted to the family creation page.	The user will be prompted to the family creation page
3.	Set a reminder on any date.	The reminders pop up for the set event, when user logs in.	The reminders pop up for the set event, when user logs in.
4.	User has to click on the trees node.	Blood relatives of node will be highlighted.	Blood relatives of node are getting highlighted.

Table 7.2: Test Cases for Acceptance Testing

7.1.3 Deployment and Benchmark Testing

Digitize Family was tested on the following devices

- Samsung Galaxy Y
  Android 2.3.6 Fast 832MHz Processor
- Samsung Galaxy Grand
  Android 4.1.2 Dual Core 1.25GHz Processor
- Micromax Canvas 2
  Android 4.1 Dual Core 1GHz Processor
- Sony Ericsson Xperia Neo V
  Android 2.3 1GHz Scorpion Processor

Device	Loading Time	Delay	Comment
Samsung Galaxy Y	∼30s	∼10s	Response Slow
Samsung Galaxy Grand	∼15s	∼5s	Response is best better view of tree due large screen
Micromax Canvas 2	∼16s	∼6s	Response is good better view of tree due large screen
Sony Ericsson Xperia Neo V	∼25s	∼8s	Response is good

Table 7.3: Deployment and Benchmark Testing

Chapter 8

RESULTS

8.1 Login Page

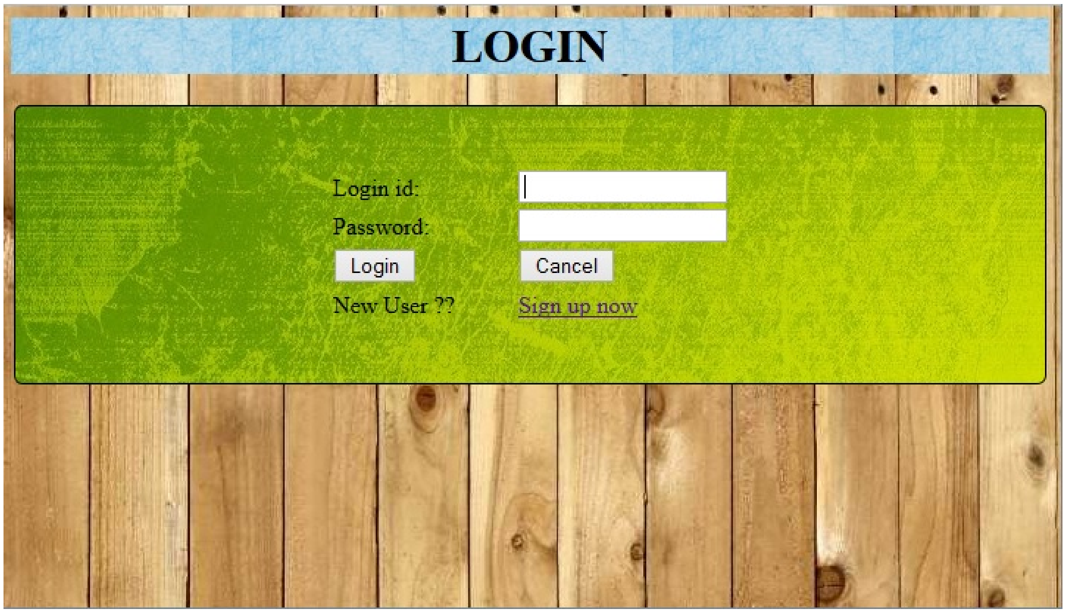
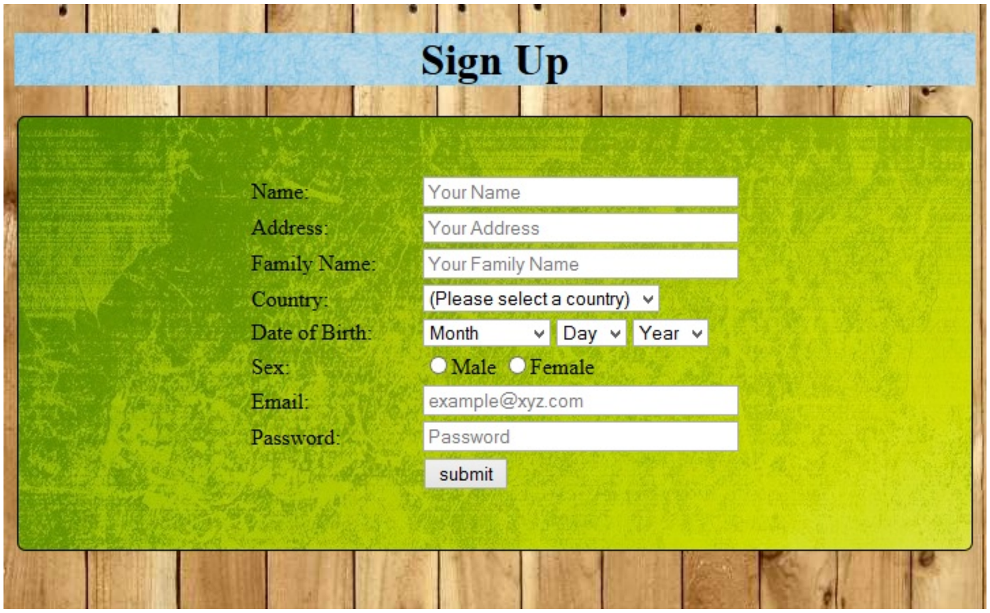


Figure 8.1: Login Page

8.2 Signup Page



Sign Up

Name: Your Name

Address: Your Address

Family Name: Your Family Name

Country: (Please select a country)

Date of Birth: Month Day Year

Sex: ○ Male ○ Female

Email: example@xyz.com

Password: Password

Figure 8.2: Signup page

8.3 Tree Page

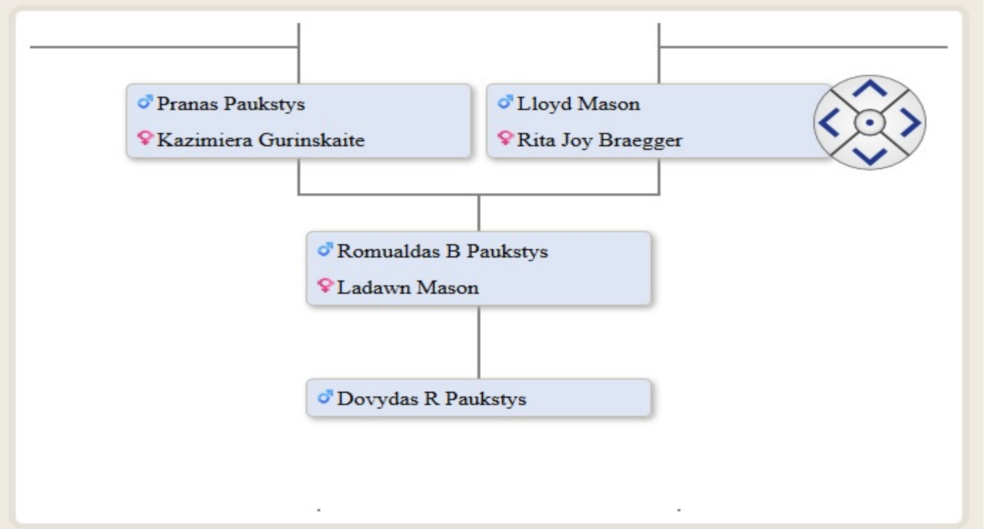


Figure 8.3: Tree page

8.4 Bloodline View

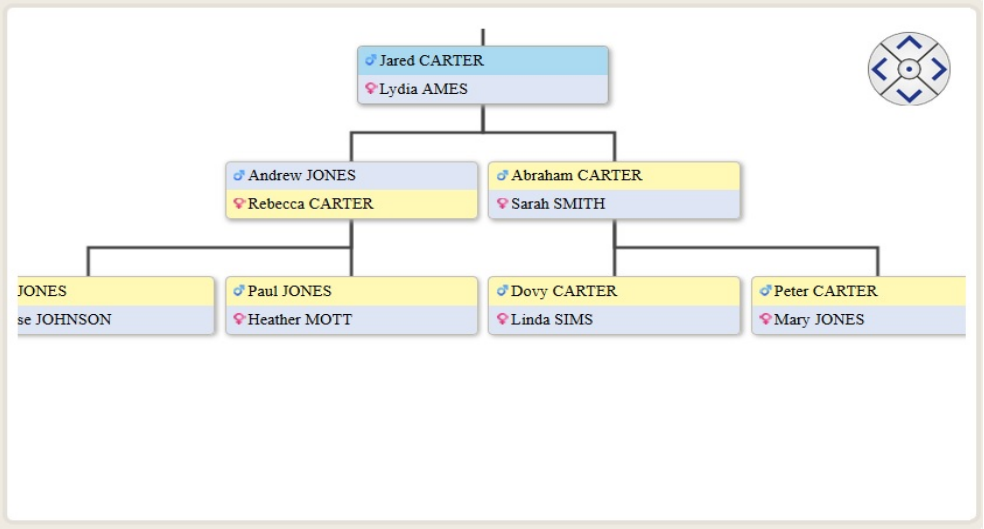


Figure 8.4: Bloodline View

8.5 Timeline View

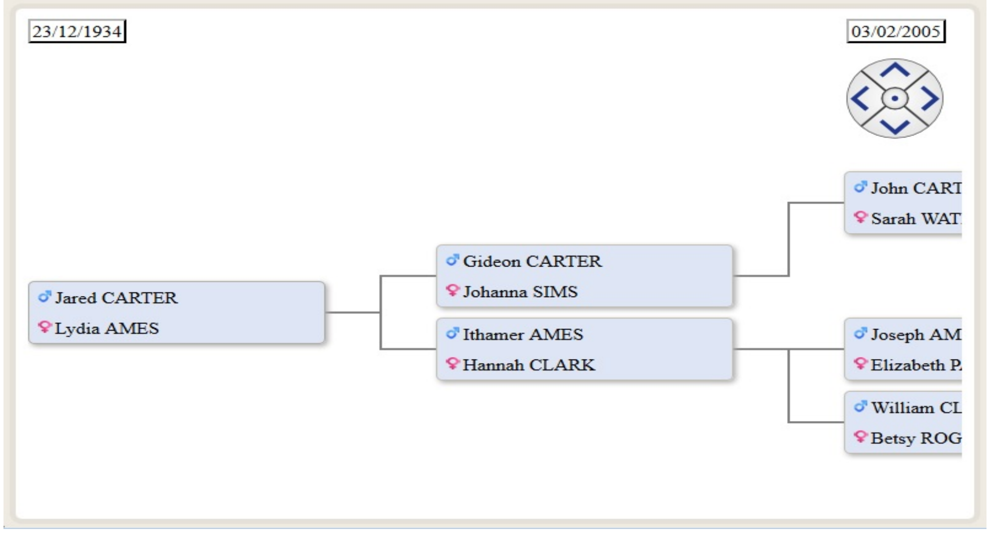


Figure 8.5: Timeline View

8.6 Set Reminders

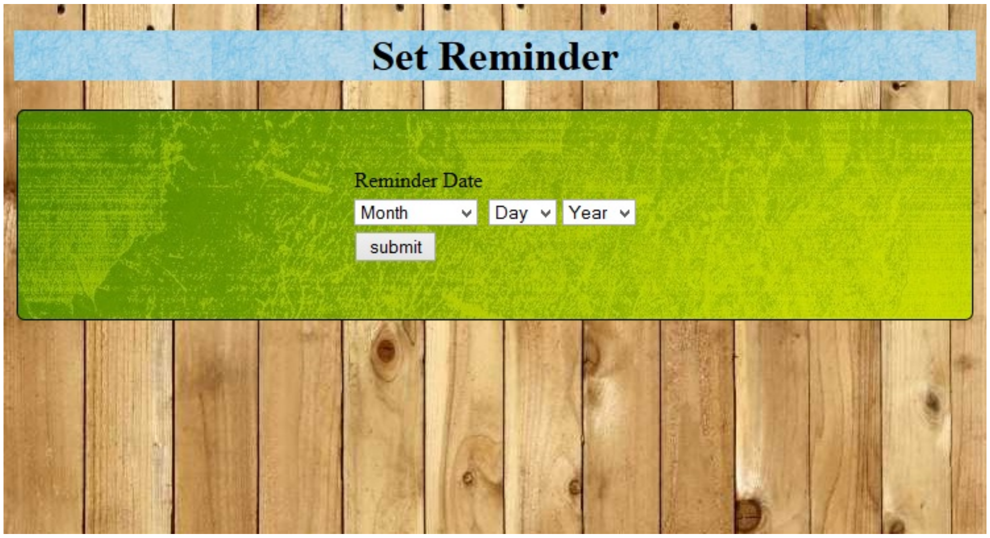


Figure 8.6: Set Reminder

Chapter 9

CONCLUTION AND FUTURE SCOPE

FAMILY TREE USING PhoneGap aims in keeping its user in touch with the family through this application which bring together generations of a family in a very simple way by representing it in a tree.And along with this with its features makes the whole process interactive and simple. Since this application is for six different OS this makes it accessible to all type of mobile phone users. Hence with this application technology brings one closer to their family.

Chapter 10

APPENDIX

10.1 Glossary

Abbreviation	Meaning
CSS	Cascading style sheet
GUI	Graphical User Interface
HTML	Hyper Text Markup Language
JDBC	Java Data Base Connectivity

10.1.1 Terminology

PhoneGap : Easily create apps with the only free open source framework that supports 7 mobile platforms.

Bibliography

[1] Pankaj Jalote. An Integrated approch to software Engineering. 3rd Edition,2005.

[2] Michael Blaha. Object oriented modelling and design with UML. 2nd Edition,2005.

[3] Adobe system Inc. docs.phonegap.com 2rd Edition,2004.

[4] Refsnes Data. www.w3schools.com Oreilly medias Headfirst web services

[5] Refsnes Data. www.w3schools.com Oreilly medias Headfirst java script