The final report of the project entitled

**CRONY**

Submitted in partial fulfillment of the requirements for the degree of

**Bachelors of Technology**

In

**Electronics and Communication**

Batch (2013-2017)



**Submitted To: Submitted By:**

**Gazalpreet kaur**

**ECE-8th Sem**

**1308574**

**TABLE OF CONTENTS**

**Chapter 1 INTRODUCTION**

* 1. Introduction……………………………………………………………………………………….11
  2. Objective…………………………………………………………………………………………….12
  3. Problem Definition……………………………………………………………………………..12
  4. Limitation in Existing System………………………………………………………………13
  5. Proposed System………………………………………………………………………………..14

**Chapter 2 REQUIREMENT ANALYSIS**

2.1 Processing Environment: Hardware/Software…………………………………….15

2.2 Hardware Requirements…………………………………………………………………….15

2.3 Software Requirements………………………………………………………………………15

2.4 Intents Used……………………………………………………………………………………….19

2.5 Technology Used………………………………………………………………………………. 20

**Chapter 3**

3.1 Architecture……………………………………………………………………………………….24

3.1.1 Advantages of 3-Tier Architecture……………………………..............24

3.2 Prototype-Rapid Application Development………………………………………..26

3.2.1 Advantages of RAD………………………………………………………………..27

3.3 Data Flow Diagrams (DFDs)…………………………………………………………………28

3.4 Tables…………………………………………………………………………………………………35

**Chapter 4**

4.1 Modules……………………………………………………………………………………………..36

4.1.1 Modules Specification……………………………………………………………36

4.2 Business Functions……………………………………………………………………………..36

4.3 Users………………………………………………………………………………………………….37

4.4 Roles………………………………………………………………………………………………….38

4.4.1 User………………………………………………………………………………………38

4.4.2 Database Administrator………………………………………………………..38

**Chapter 5 WEB SERVICES**

5.1 Web services…………………………………………………………………………………….39

5.2 Web services Specifications………………………………………………………………42

5.3 Feasibility Study………………………………………………………………………………..46

**Chapter 6 TESTING**

6.1 Testing………………………………………………………………………………………………..49

6.2 Levels of Testing…………………………………………………………………………………51

6.3 Tools of Testing…………………………………………………………………………………..51

**Chapter 7**

7.1 Future Scope………………………………………………………………………………………52

7.2 References and Bibliography………………………………………………………………52

7.3 Conclusion………………………………………………………………………………………….53

**SCREENSHOTS……………………………………………………...54**

**DECLARATION**

I hereby declare that the Training Report entitled "**CRONY**" is an authentic record of my own work as requirements of six months training during the period for the award of degree of BTech in Electronics and Communication of PTU, jalandhar is our original work and has not been submitted for the award of any other degree or other similar title.

**GAZALPREET KAUR**

**ACKNOWLEDGEMENT**

I would like to take this opportunity to express my sincere indebtedness and sense of gratitude to all those who have contributed greatly towards the successful completion of my project **“CRONY”.**

A project is a major milestone during the study period of a student. As such, this project was a challenge to me which came up like an opportunity to prove my caliber. So it is great pleasure to have the opportunity to extend my heartfelt thanks to everybody who helped me through the successful completion of this project.

It would not have been possible to see through the undertaken project without the guidance of **Mr. Rohit Sharma and Mr. Mohit Sharma.**

I would also like to thank DAVIET for giving me such an opportunity to use my theoretical knowledge to develop practical and real-life industry ready applications. I was exposed to industrial environment through this project. While developing project, I also got a chance to learn skills like deadline management, resource optimization and team-work which will surely help me throughout my life.

At the end, I thank everyone who has helped and supported me during entire development process of “**CRONY**”.

**ABSTRACT**

A **personal assistant** (CRONY) is a software agent that can perform tasks or services for an individual. These tasks or services are based on user input,, meeting reminders, and the ability to access information from a variety of online sources ( news, user schedules, Note making, Birthday reminder setting ,Alarm, Medicine Reminder etc.). Examples of such an agent are Microsoft's office assistant and later Crotona, Apple 's Siri, Google 's Google now(earlier Android versions had voice search via Google Search).

Personal assistant technology (CRONY) is enabled by the combination of mobile devices, application programming interfaces (APIs), and the proliferation of mobile devices, application programming interfaces. However, intelligent automated assistants are designed to perform specific, one-time tasks specified by user voice instructions, while smart personal agents perform ongoing tasks (*e.g.*, schedule management) autonomously.

**LIST OF FIGURES**

List of data flow diagrams:

1. Login

2. Manage user profile

3. User account management

4. Sign Up

5. Note-Making

6. Reminders

**LIST OF TABLES**

1. Usermaster

**Company Profile**

Situated at the center of one of the most beautiful and developed cities of India, Develop Tech IT Solutions Pvt. Ltd. was established by a group of young, energetic, determined and talented pros in 2013. Develop Tech IT Solutions Pvt. Ltd., right from its origin in 2013 has always focused on what is called stability in the market and customer satisfaction to the full extent. It is not always about the numbers, but the quality of service they provide to customers that gives the inspiration and spirited driving force. Got nurtured under the guidance of think-tanks Mr. Rohit Sharma (Director) and Mr.Vishal Kedia (Managing Director), who have established as one of the best companies in this sector. This is supported by the fact that they have been awarded as “Emerging Young Entrepreneur of the year 2013” by STPI (Software Technology parks of India) and CII (Confederation of Indian Industry).

Develop Tech IT Solutions Pvt. Ltd., an emerging IT company is exploiting its each and every bit to get on par with the definition of IT given by intellectuals. Develop Tech IT Solutions Pvt. Ltd. is run by a group of young, innovative and energetic professionals capable of accepting any challenge and fulfilling it to the fullest.

Develop Tech IT Solutions Pvt. Ltd. is a strategic solutions and professional services provider to advise, transform and optimize business and technology processes while minimizing risk. It is offering a large number of services across various sectors with a large number of solutions to each and every problem. Its core services include Console Applications, System Applications, and Web Applications & Mobile Applications.

It has been providing services to both the public as well as private sectors achieving great success and credibility. Proficiency in each and every sector concerned with IT is motto so that it can give its best each and every time to clients. DevelopTech IT Solutions Pvt. Ltd. is providing its services all across the globe including its major clients in the US and Australia.

**CHAPTER-1**

**1.1 INTRODUCTION**

A mobile assistant application(CRONY) have many features. The basic command types  should be about news, reminders, set alarm, call a contact, Gmail.

* Set alarm
* Set reminders for your important tasks
* Open music player
* Open Facebook, Whatsapp, Snapchat, Instagram
* Open Dialer
* Voice Commands to open apps
* Note-Making by speech to text
* Make your shopping list
* Calendar
* Seeing Horoscope
* NEWS

Aim of this project is to create an android application that will help user to manage his meetings efficiently on the android mobile phones. while most of the inbuilt notes and to-do applications provide user to manage their meetings, our application handles meeting topics, their venues. following are the main features of our application.

1. Management of meetings: user can save the meetings along with the topic, venue, date and time, the persons that are to be invited to the meeting.

2. We can delete our notes and shopping list just by long pressing on the list we made.

3. Opening a dialer to call someone and messaging through whatsapp.

4. We can make our shopping list by voice to text.

5. Your sign can help heal your soul means you can access your horoscope daily.

6. Never miss your important tasks by setting alarm through this application.

7. It shows us NEWS.

8. We can set birthday reminders.

Though there are some android applications that handle this meetings related task, they are not as efficient as our application.

# 1.2 OBJECTIVE

An application that prevents human dependency for performing tasks like wake up call, reminders for appointments, events and meetings , note making, fetching news and information, social connectivity and entertainment purposes with just a few taps like opening music player. Notifications may be enabled for as many times as per requirements and an add-on DO NOT DISTURB mode that prevents notifications when not required. Unlike normal reminders this application keeps one reminding with the pending tasks until they aren't performed. Person now a days is so busy that he is unable to remember his important meetings, medicines and even his important dates like Birthday dates of his friends so to overcome that problem we have developed the application for that which works as a cheap personal assistant for every person. Person can place this application in his phone and can be free from his most of reminders.

###### 1.3 PROBLEM DEFINITION

The traditional speech recognition application uses heavy data and do not perform well in offline mode. This tool provides a shared screen where a user can select its most desired apps and display them on the main screen. We also provide a VOICE COMMAND to open mostly used application. If applications are not available in the phone then it provides the download link for them to further access that tool if needed. A personal assistant helps with time and daily management, scheduling of meetings, and note taking. The role of a personal assistant can be varied, such as answering phone calls, taking notes, scheduling meetings, emailing, texts etc.

But a person can do mistakes in making notes , forget to remind you the important meetings, can lost some important notes. So our app basically works like a personal assistant but without doing same mistakes. Note-Making in crony is more efficient as the user now works on mobile rather than on paper.

###### LIMITATION IN EXISTING SYSTEM

A Personal Assistant is a person who manages the formal papers, set reminders, make notes and other essential things while the person does not need to bother anything about that. Personal assistant provides facilities like:-

• Making notes by voice command.

• Provides the latest news feeds.

• Set meetings and alarm to them.

• Opens mostly used social apps by voice commands.

• Make and manages the shopping list.

Various operational works that are done in a Personal Assistant are:-

• Voice search.

• Voice commands.

These are the various jobs that need to be done in a phone by the person mostly note making and shopping list. All these works are done on papers. The work is done as follows:-

• Information of notes is done by just writing on paper with pen and every time the information is stored freshly.

•Information is generally recorded on the document, which contains information. It is destroyed after some time period to decrease the paper load.

•Records of notes are maintained in pre-formatted sheets, which are kept in a file.

•Information about various list is not kept as any document. Doctors themselves do this job by remembering various medicines.

###### PROPOSED SYSTEM

1. Management of meetings: user can save the meetings along with the topic, venue, date and time, the persons that are to be invited to the meeting.

2. Handling of notes and shopping lists: User can make notes, shopping list and delete them whenever wanted.

3. Opening of mostly used application from the main tool itself.

4. Sending message: user can send text message to any of his contacts, using the same application.

# CHAPTER-2

**REQUIREMENT ANALYSIS**

**2.1 PROCESSING ENVIRONMENT: HARDWARE/SOFTWARE**

**2.1.1 HARDWARE REQUIREMENT**

This application was processed on a system with configuration

* 8.00GB RAM, Windows
* Intel(R) core (TM) i3-4005U CPU @1.70GHz
* 64-bit Operating System

**2.1.2 SOFTWARE REQUIREMENTS:**

**NetBeans 8.1**

* NetBeans 8.1

NetBeans IDE 8.1 provides out-of-the-box code analyzers and editors for working with the latest Java 8 technologies--Java SE 8, Java SE Embedded 8, and Java ME Embedded 8. The IDE also has a range of new tools for HTML5/JavaScript, in particular for Node.js, KnockoutJS, and AngularJS; enhancements that further improve its support for Maven and Java EE with PrimeFaces; and improvements to PHP and C/C++ support.

.

**Database**

* MySQL Database Server 5.5

MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open-source web application software stack (and other "[AMP](https://en.wikipedia.org/wiki/List_of_AMP_packages)" stacks). LAMP is an acronym for "[Linux](https://en.wikipedia.org/wiki/Linux), [Apache](https://en.wikipedia.org/wiki/Apache_HTTP_Server), MySQL, [Perl](https://en.wikipedia.org/wiki/Perl)/[PHP](https://en.wikipedia.org/wiki/PHP)/[Python](https://en.wikipedia.org/wiki/Python_(programming_language))". [Free-software](https://en.wikipedia.org/wiki/Free_software) open-source projects that require a full-featured database management system often use MySQL.

**Database detail**:

* A database is a collection of information that is organized so that it can easily be accessed, managed, and updated. In one view, databases can be classified according to types of content: bibliographic, full-text, numeric, and images. In computing, databases are sometimes classified HTML & CSS

**Hypertext Markup Language**, commonly referred to as **HTML**, is the standard markup language used to create web pages.

* HTML stands for **H**yper **T**ext **M**arkup **L**anguage
* A markup language is a set of **markup tags**
* HTML documents are described by **HTML tags**
* Each HTML tag **describes** different document content

The most prevalent approach is the relational database, a tabular database in which data is defined so that it can be reorganized and accessed in a number of different ways. A distributed database is one that can be dispersed or replicated among different points in a network. An object-oriented programming database is one that is congruent with the data defined in object classes and subclasses.

Computer databases typically contain aggregations of data records or files, such as sales transactions, product catalogs and inventories, and customer profiles. Typically, a database manager provides users the capabilities of controlling read/write access, specifying report generation, and analyzing usage. Databases and database managers are prevalent in large mainframe systems, but are also present in smaller distributed workstation and mid-range systems such as the AS/400 and on personal computers. SQL (Structured Query Language) is a standard language for making interactive queries from and updating a database such as IBM's DB2, Microsoft's SQL Server, and database products from Oracle, Sybase, and Computer Associates.

Database will have the tables of user account having users information stored in it, package master containing details of the package and user package master which will contain the details required for login and the type of package user will be using.

**Web Server**

* Tomcat 8.0.15 :Apache Tomcat 8 is an open source software implementation of the Java Servlet, JavaServer Pages, Java Unified Expression Language and Java WebSocket technologies.

**Application Server**

* Glassfish v4.1

World’s first Java EE 7 Application server. **Java Platform, Enterprise Edition** or **Java EE** is a widely used enterprise computing platform developed under the Java Community Process.

**Operating System**

* Microsoft Windows XP and above / Linux /mac

**Linux**  is a Unix-like and mostly POSIX-compliant computer operating system (OS) assembled under the model of free and open-source software development and distribution.

* **Windows XP**  is a personal computer operating system that was produced by Microsoft as part of the Windows NT family of operating systems.

**Android Studio**

Android Studio is the official IDE for Android app development, based on IntelliJ IDEA. On top of the IntelliJ’s powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building android apps such as: A flexible Gradle-based build System.

**JDK**

The **Java Development Kit** (**JDK**) is an implementation of either one of the Java SE, Java EE or Java ME platformsreleased by Oracle Corporation in the form of a binary product aimed at Java developers on Solaris, Linux, Mac OS X or Windows. The JDK includes a private JVM and a few other resources to finish the development of a Java Application. Since the introduction of the Java platform, it has been by far the most widely used Software Development Kit (SDK).On 17 November 2006, Sun announced that they would release it under the GNU General Public License (GPL), thus making it free software.

**SDK**

A software development kit (**SDK** or "devkit") is typically a set of software development tools that allows the creation of applications for a certain software package, software framework, hardware platform, computer system, video game console, operating system, or similar development platform.It may be something as simple as the implementation of one or more application programming interfaces (APIs) in the form of some libraries to interface to a particular programming language or to include sophisticated hardware that can communicate with a particular embedded system. Common tools include debugging facilities and other utilities, often presented in an integrated development environment (IDE). SDKs also frequently include sample code and supporting technical notes or other supporting documentation to help clarify points made by the primary reference material.

The **Android SDK** (software development kit) is a set of development tools used to develop applications for **Android** platform. The **Android SDK** includes the following: Required libraries:

* 1. Debugger.
  2. An emulator.

**Mobile**

An android mobile with minimum ice cream sandwich 4.0 installed in it.

**SQLite**

SQLite is Database used in android. **SQLite** is an in-process library that implements a self-contained, server less, zero-configuration, transactional SQL database engine.

Unlike client–server database management systems, the SQLite engine has no standalone processes with which the application program communicates. Instead, the SQLite library is linked in and thus becomes an integral part of the application program. The library can also be called dynamically. The application program uses SQLite's functionality through simple function calls, which reduce latency in database access: function calls within a single process are more efficient than inter-process communication. SQLite stores the entire database (definitions, tables, indices, and the data itself) as a single cross-platform file on a host machine. It implements this simple design by locking the entire database file during writing. SQLite read operations can be multitasked, though writes can only be performed sequentially.

**2.2 INTENTS USED :**

**Facebook Intent :**

The Facebook API is a platform for building applications that are available to the members of the social network of Facebook. The API allows applications to use the social connections and profile information to make applications more involving, and to publish activities to the news feed and profile pages of Facebook, subject to individual users privacy settings. With the API, users can add social context to their applications by utilizing profile, friend, Page, group, photo, and event data. The API uses RESTful protocol and responses are in JSON format.

**Google+ Intent:**

Google Plus is a service to share links, photos and other content. The Google Plus API allows developers to access publicly-available Google Plus content, including user information and publicly shared items.

For example: Google play service library

.**Instagram Intent :**

Instagram is a photo sharing iPhone app and service. Users take photos and can share them with Instagram contacts, as well as friends on other social networks like Twitter and Facebook. The Instagram API provides access to user authentication,

friend connections, photos and all the other elements of the iPhone app--including uploading new media.

**2.3 TECHNOLOGY USED:**

**Java:** Java is Platform Independent, Secure, Object Oriented, Scalable, and Robust Programming Language.

It consists of two parts

* + - JVM stands for Java Virtual Machine, which is run time environment to execute the java programs.
    - Java API (Application Programming Interface) that consists of inbuilt classes used in java programs.
    - Simple: Java is an extension of C and C++ with added feature of garbage collection and improved memory management.
    - Object-oriented: Object oriented programming deals with objects and there behaviors and hence an analogy of real world can be found in programs.
    - Robust: Java is intended for writing programs that must be reliable in a variety of ways.
    - Secure: Java is intended for use in networked/distributed environments. Toward that end, a lot of emphasis has been placed on security.
    - Portable: There are no "implementation dependent" (machine/ processor dependent) aspects of the specification.
    - Interpreted: Java byte codes are translated on the fly to native machine instructions (interpreted) and not stored anywhere.
    - High performance: Java Byte code is more efficient and its interpreted nature provides high performance.
    - Multithreaded: Multithreading is a way of building applications with multiple threads (more than one processes running at the same time).
    - Dynamic language: Java programs carry substantial amount of run time information.

**JDBC:**

JDBC (Java Database Connectivity) is an API, which is used for the communication of java programs with different databases. JDBC allows multiple implementations to exist and be used by the same application. The API provides a mechanism for dynamically loading the correct Java packages and registering them with the JDBC Driver Manager. The Driver Manager is used as a connection factory for creating JDBC connections.

**Social Media API:** MONITORING AND ANALYTICS:

Use Social API and patent-pending technology to deploy social media content and text analytics within your applications to provide customers with leading-edge and intelligent services. Social media APIs can be used to access our large repository of social media content instantly. Widgets can be used to effortlessly integrate social media metrics with beautiful charts and graphs in your internal portals, application dashboards, and sharepoint.

**MySQL:**

MySQL is used as database, used to store data. It is RDBMS. The MySQL development project has made its [source code](https://en.wikipedia.org/wiki/Source_code) available under the terms of the [GNU General Public License](https://en.wikipedia.org/wiki/GNU_General_Public_License), as well as under a variety of [proprietary](https://en.wikipedia.org/wiki/Proprietary_software) agreements. MySQL was owned and sponsored by a single [for-profit](https://en.wikipedia.org/wiki/Business) firm, the [Swedish](https://en.wikipedia.org/wiki/Sweden) company [MySQL AB](https://en.wikipedia.org/wiki/MySQL_AB), now owned by [Oracle Corporation](https://en.wikipedia.org/wiki/Oracle_Corporation). For proprietary use, several paid editions are available, and offer additional functionality.

On all platforms except Windows, MySQL ships with no [GUI](https://en.wikipedia.org/wiki/Graphical_user_interface) tools to administer MySQL databases or manage data contained within the databases. Users may use the included [command line](https://en.wikipedia.org/wiki/Command_line) tools, or install [MySQL Workbench](http://dev.mysql.com/downloads/workbench/) via a separate download. Many third party GUI tools are also available.

**Scripting Languages:**

* **XML, XHTML Extensible Markup Language** (**XML**) is a markup language that defines a set of rules for encoding documents in a format which is both human-readable and machine-readable.

**Extensible Hypertext Markup Language** (**XHTML**) is part of the family of XML markup languages. It mirrors or extends versions of the widely used Hypertext Markup Language (HTML), the language in which Web pages are formulated.

* **Others**

UML

The **Unified Modeling Language** (**UML**) is a general-purpose, developmental, modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system.

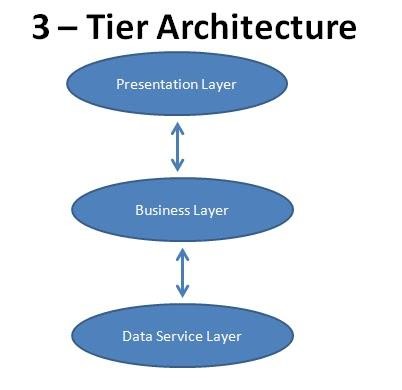
JUnit – Unit Testing

**JUnit** is a unit testing framework for the Java programming language. JUnit has been important in the development of test-driven development, and is one of a family of unit testing frameworks which are collectively known as xUnit that originated with SUnit.

**CHAPTER-3**

**3.1 ARCHITECTURE**

A three tier architecture is a client server architecture in which the functional process logic, data access, computer data storage and user interface are developed and maintained as independent modules on separate platforms. Three tier architecture is a software design pattern and as well established software architecture.



**The three tiers in a three-tier architecture are:**

1. **PRESENTATION LAYER:** Occupies the top level and displays information related to services available on a website. This tier communicates with other tiers by sending results to the browser and other tiers in the network.

2. **BUSINESS LOGIC LAYER:** Also called the middle tier or logic tier, this tier is pulled from the presentation tier. It controls application functionality by performing detailed processing.

3. **DATA TIER:** Houses database servers where information is stored and retrieved. Data in this tier is kept independent of application servers or business logic.

**3.1.1 ADVANTAGES OF 3TIER ARCHITECTURE**

The main advantages of the 3 Tier Architecture are:

**Flexibility**: By separating the business logic of an application from its presentation logic, 3 Tier architecture makes the application much more flexible to changes.

**Maintainability:** Changes to the components in one layer should have no effect on any others layers. Also, if different layers require different skills (such as HTML/CSS is the presentation layer, PHP/Java in the business layer, SQL in the data access layer) then these can be managed by independent teams with skills in those specific areas.

**Reusability**: Separating the application into multiple layers makes it easier to implement reusable components. A single component in the business layer, for example, may be accessed by multiple components in the presentation layer, or even by several different presentation layers (such as desktop and the web) at the same time.

**Scalability**: A 3Tier architecture allows distribution of application components across multiple servers thus making the system much more scalable.

**Reliability**: A 3Tier architecture, if deployed on multiple servers, makes it easier to increase reliability of a system by implementing multiple levels of redundancy.

**3.2Prototype-RAPIDAPPLICATION DEVELOPMENT**

Rapid application development (RAD) is both a general term used to refer to alternatives to the conventional waterfall model of software development .In general, **RAD approaches to software development put less emphasis on planning tasks and more emphasis on development**. In contrast to the waterfall model, which emphasizes rigorous specification and planning, RAD approaches emphasize the necessity of adjusting requirements in reaction to knowledge gained as the project progresses. RAD approaches also emphasize **a flexible process** that can adapt as the project evolves rather than rigorously defining specifications and plans correctly from the start.

In addition to James Martin's RAD method, other approaches to rapid development include Agile methods and the spiral model. RAD is especially well suited (although not limited to) developing software that is driven by user interface requirements. Graphical user interface builders are often called rapid application development tools.

The waterfall solution to this was to try and rigidly define the requirements and the plan to implement them and have a process that discouraged changes to either. The new RAD approaches on the other hand recognized that software development was a knowledge intensive process and sought to develop flexible processes that could take advantage of knowledge gained over the life of the project and use that knowledge to reinvent the solution .

**3.2.1ADVANTAGES OF RAPID APPLICATION DEVELOPMENT MODEL**

**Risk reduction**: A prototype could test some of the most difficult potential parts of the system early on in the lifecycle. This can provide valuable information as to the feasibility of a design and can prevent the team from pursuing solutions that turn out to be too complex or time consuming to implement. This benefit of finding problems earlier in the lifecycle rather than later was a key benefit of the RAD approach. The earlier a problem can be found the cheaper it is to address.

Users are better at using and reacting than at creating specifications. In the waterfall model it was common for a user to sign off on a set of requirements but then when presented with an implemented system to suddenly realize that a given design lacked some critical features or was too complex. In general most users give much more useful feedback when they can experience a prototype of the running system rather than abstractly define what that system should be.

**Prototypes** can be usable and can evolve into the completed product. One approach used in some RAD methodologies was to build the system as a series of prototypes that evolve from minimal functionality to moderately useful to the final completed system. The advantage of this besides the two advantages above was that the users could get useful business functionality much earlier in the process.

**Better quality**: By having users interact with evolving prototypes the business functionality from a RAD project can often be much higher than that achieved via a waterfall model. The software can be more usable and has a better chance to focus on business problems that are critical to end users rather than technical problems of interest to developers. Possibility of lesser defects due to prototyping in nature

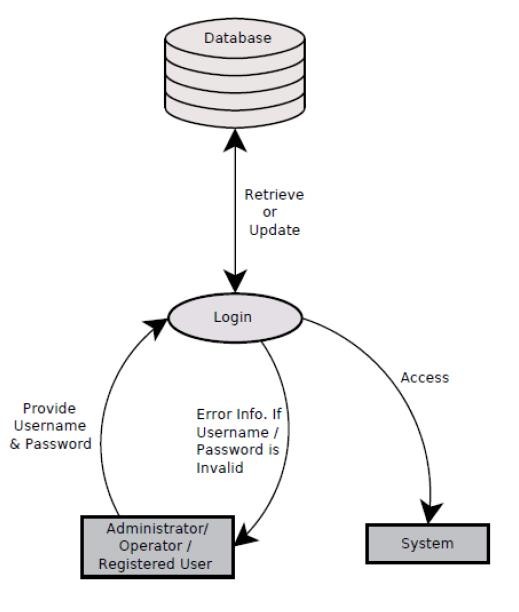
**3.3 DATA FLOW DIAGRAMS (DFDs)**

DFDs only involve four symbols. They are:

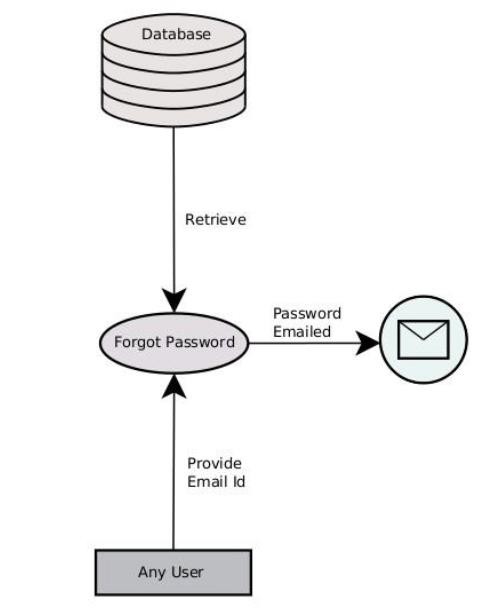
* Process
* Data Flow
* Data Store
* External entity

|  |  |
| --- | --- |
| http://members.tripod.com/~myyee/cs457/process.gif | **Process** Transform of incoming data flow(s) to outgoing flow(s). |
| http://members.tripod.com/~myyee/cs457/data.gif | **Dataflow** Movement of data in the system. |  |

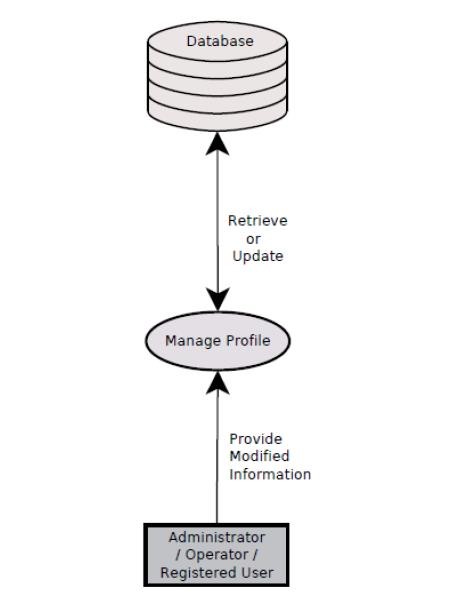
**1. LOGIN**



**2. FORGOT PASSWORD**

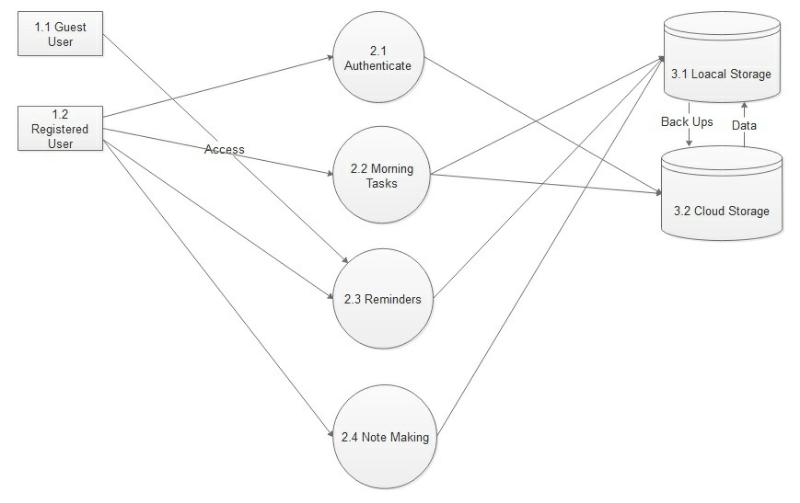


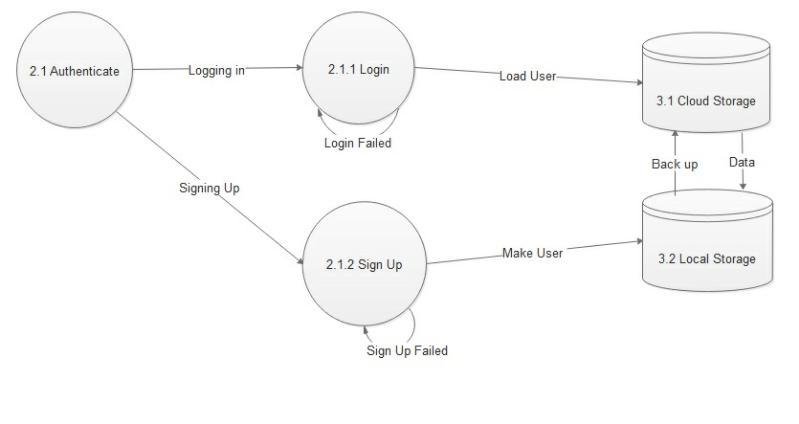
**3. MANAGE PROFILE**



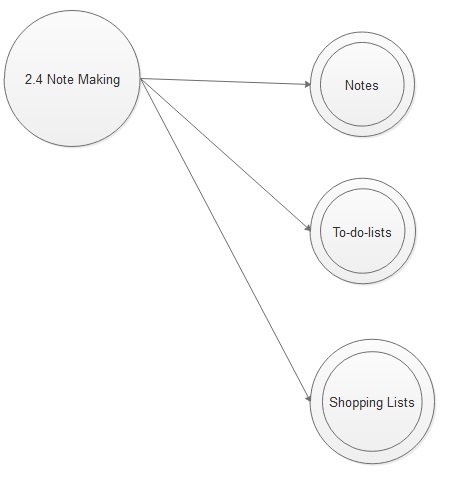
**4. APPLICATION VIEW**

# 

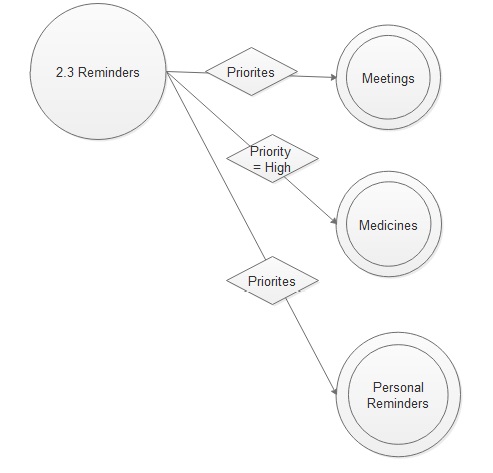
****



**5. NOTE-MAKING**

****

**6. REMINDERS**

****

**3.4 TABLES:**

**USERMASTER**

|  |
| --- |
| **Usermaster** |
| userid INT(11) |
| name VARCHAR(20) |
| username VARCHAR(20) |
| email VARCHAR(30) |
| password VARCHAR(10) |
| mobilenumber VARCHAR(13) |
| dateofbirth VARCHAR(10) |
| city VARCHAR(20) |
| Profileimage TEXT |

###### Chapter-4

###### 4.1 MODULES:

**4.1.1 MODULES SPECIFICATION**

Various modules of the CRONY are:

1. **User Authentication:** All the functionality required for users such as login will be generated in this. It will authenticate whether a user is valid user or unauthenticated user by matching username provided.
2. **User Profile Management:** All the functionality required for registered user such as Users profile & notifications will be achieved in this module.
3. **Database Management:** In this database will be managed. In this all the information regarding user like user account in which user is logged in will be achieved in this module.

###### 4.2 BUSINESS FUNCTIONS

###### Some of the most important business functions that are used in Crony application are:

* **Note Making**: User can create a note with Speech To Text facility of android or with in built keyboard. The note created is concatenated with date, day and accurate time for distinction.
* **Managing Notes**: In management, user can check the notes. It also facilitates the user to delete unwanted or old notes**.**
* **Shopping list creation:** User can create shopping list by using speech to text facility.
* **Managing Shopping List Creation:** User can manage (edit and delete) shopping list just by long pressing on any list.
* **Alarm:** User can set alarm for morning or any important tasks. Never miss your important tasks by setting alarm through this application.
* **Reminders:** Person can set reminders for medicines, birthday reminders, important meetings.
* **Show news:** For daily news feed , person can access crony app.
* **Horoscope:** Your sign can help heal your soul means you can access your horoscope daily.
* **Social Networking:** we can access facebook, google, gmail, snapchat, whatsapp from Crony.

###### 4.3 USERS

* User
* Database admin

###### 4.4 ROLES

**4.4.1 User**

* Can create an account for full access or use the app as a guest after logging from google plus.
* Manage notes and shopping list of all the users.
* Can order CRONY to open other apps.

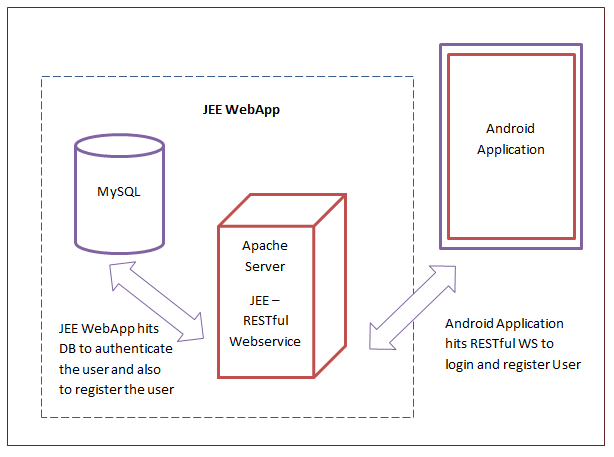
**4.4.2 Database Administrator**

* Add User
* Manage Database
* Manage Details of Users
* Deletes User after user deletes account
* Add support users to manage its settings

**CHAPTER-5**

**5.1 WEB SERVICES**

A **web service** is a service offered by an electronic device to another electronic device, communicating with each other via the World Wide Web. In a Web service, Web technology such as HTTP, originally designed for human-to-machine communication, is utilized for machine-to-machine communication, more specifically for transferring machine readable file formats such as XML and JSON. In practice, the web service typically provides an object-oriented web-based interface to a database server, utilized for example by another web server, or by a mobile application, that provides a user interface to the end user. Another common application offered to the end user may be a mashup, where a web server consumes several web services at different machines, and compiles the content into one user interface



**Components of a web service:**

From an implementation point of view, a web service typically has two distinct

components:

• Some 'logic', which actually carries out the real work required to provide the

service functions, whatever they might be. For example, a simple service to

calculate VAT might include at its heart a piece of code that receives a monetary

value in sterling, such as £10, multiplies that value by 17.5% to calculate the VAT

(£1.75) and returns the new total value including VAT (£11.75).

• An interface, which is implemented in XML. This describes how to put a request to the service and what the expected response or result may consist of. In addition, in order to actually make use of any web service we implement, we need

also to implement a client that constructs our request (based on the XML interface)

and displays the result returned by the web service.

This view of a web service should help to explain the different approaches to

producing a web service.

**Explanation:**

The term "web service" describes a standardized way of integrating web-based applications using the XML, SOAP, WSDL and UDDI open standards over an Internet protocol backbone. XML is the data format used to contain the data and provide metadata around it, SOAP is used to transfer the data, WSDL is used for describing the services available and UDDI lists what services are available.

A web service is a method of communication between two electronic devices over a network. It is a software function provided at a network address over the web with the service *always on* as in the concept of utility computing.

.Many organizations use multiple software systems for management. Different software systems often need to exchange data with each other, and a web service is a method of communication that allows two software systems to exchange this data over the internet. The software system that requests data is called a *service requester*, whereas the software system that would process the request and provide the data is called a *service provider*.

Different software may use different programming languages, and hence there is a need for a method of data exchange that doesn't depend upon a particular programming language. Most types of software can, however, interpret XML tags. Thus, web services can use XML files for data exchange.

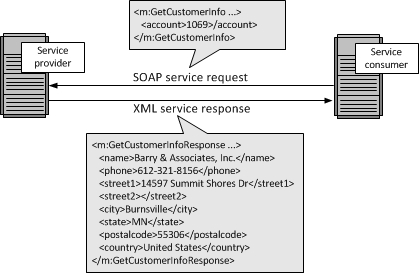
Rules for communication between different systems need to be defined, such as:

* How one system can request data from another system.
* Which specific parameters are needed in the data request.
* What would be the structure of the data produced. (Normally, data is exchanged in XML files, and the structure of the XML file is validated against an .xsd file.)
* What error messages to display when a certain rule for communication is not observed, to make troubleshooting easier.

All of these rules for communication are defined in a file called WSDL(Web Services Description Language), which has a.wsdl extension.A directory called UDDI (Universal Description, Discovery and Integration) defines which software system should be contacted for which type of data. So when one software system needs one particular report/data, it would go to the UDDI and find out which other system it can contact for receiving that data. Once the software system finds out which other system it should contact, it would then contact that system using a special protocol called SOAP (Simple Object Access Protocol). The service provider system would first validate the data request by referring to the WSDL file, and then process the request and send the data under the SOAP protocol.

**5.2 WEB SERVICES SPECIFICATIONS:**

Three specifications for Web Services are illustrated in this section: SOAP, REST, and JSON.



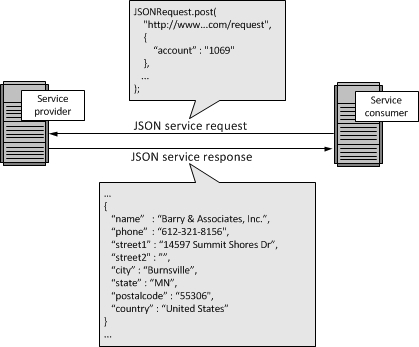
**REPRESENTATION STATE TRANSFER:**

Representation State Transfer (REST) appeals to developers because it has a simpler style that makes it easier to use than SOAP.



**JAVA SCRIPT OBJECT NOTATION:**

While both SOAP and REST use XML for interchange, JavaScript Object Notation (JSON) uses a subset of JavaScript. This is illustrated in the figure below. More on [JSON](http://www.service-architecture.com/articles/web-services/javascript_object_notation_json.html).



**WHEN TO USE REST,JSON or OTHER OPTIONS:**

There really is no "best" option for Web Services. Generally, you will use whatever your service provider supports. If you use multiple service providers, it is easily possible that you will be using all three Web Services specifications: SOAP, REST, and JSON.

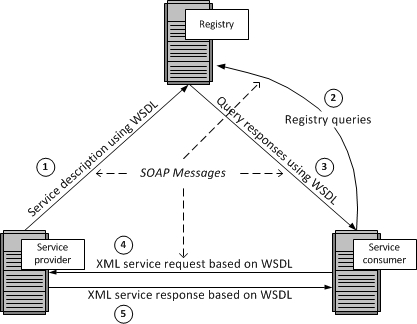
**HISTORY OF WEB SERVICES SPECIFICATION:**

Web Services Description Language (WSDL); Universal Description and Discovery (UDDI); and SOAP formed the original Web Services specification. This section provides a history.

**WEB SERVICES DESCRIPTION LANGUAGES:**

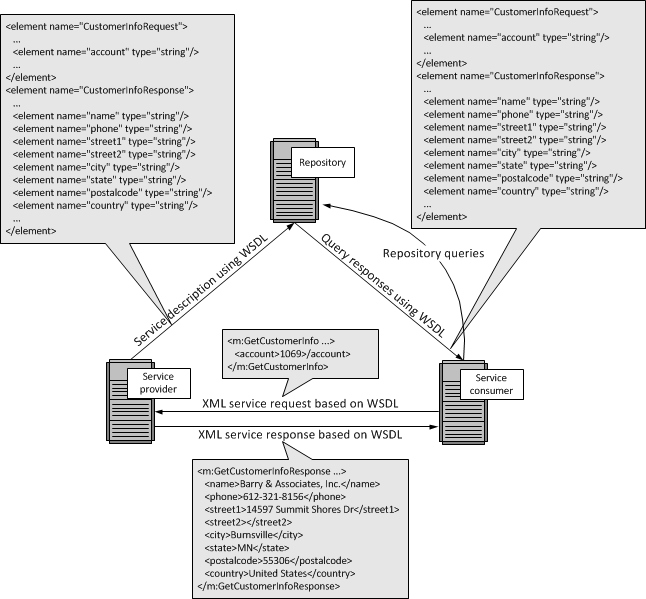
The Web Services Description Language (WSDL) forms the basis for the original Web Services specification. The following figure illustrates the use of WSDL. At the left is a service provider. At the right is a service consumer. The steps involved in providing and consuming a service are:

1. A service provider describes its service using WSDL. This definition is published to a repository of services. The repository could use Universal Description, Discovery, and Integration (UDDI). Other forms of directories could also be used.
2. A service consumer issues one or more queries to the repository to locate a service and determine how to communicate with that service.
3. Part of the WSDL provided by the service provider is passed to the service consumer. This tells the service consumer what the requests and responses are for the service provider.
4. The service consumer uses the WSDL to send a request to the service provider.
5. The service provider provides the expected response to the service consumer.



**UNIVERSAL DESCRIPTION ,DISCOVERY AND INTEGRATION(UDDI):**

The repository shown in the above figure could be a UDDI registry. The UDDI registry was intended to eventually serve as a means of "discovering" Web Services described using WSDL. The idea is that the UDDI registry can be searched in various ways to obtain contact information and the Web Services available for various organizations. How much "discovery" was ever used is open to discussion. Nevertheless, even without the discovery portion, the UDDI registry is a way to keep up-to-date on the Web Services your organization currently uses. It can be used at design time and with governance. An alternative to UDDI is the ebXML Registry.



**5.3 FEASIBILITY STUDY**

Prior to stating whether the system we have to develop is feasible or not we believe that we should emphasize on what is implied by the word “Feasibility”. Feasibility is the measure of how beneficial or practical the development of the system will be to the organization. It is a preliminary survey for the systems investigation. It aims to provide information to facilitate a later in-depth investigation.

**Types**

There are various measures of feasibility that helps to decide whether a particular project is feasible or not. These measures include –

* **Economic Feasibility**: An evaluation of development cost weighted against the ultimate income or benefit derived from the developed system.
* **Technical Feasibility**: A study of function, performance and constraints that may affect the ability to achieve an acceptable system.
* **Operational Feasibility**: A proposed system is beneficial only if it can be turned into an information system that will meet the operational requirements of an organization. A system often fails if it does not fit within existing operations and if users resist the change. Important issues a systems developer must look into are:
* Will the new system be used if implemented in an organization?
* Are there any major barriers to implementation or is proposed system accepted without destructive resistance?

Another important fact to be regarded is the security control, which is handled by the system. Since data regarding each Customer and the Organization is confidential, security is a key issue. Information falling into the wrong hands could jeopardize the entire organization. Unlike in semi-computerized systems

**Economic feasibility**

In making recommendations a study of the economics of the proposed system should be made. Even though finding out the costs of the proposed project is difficult we assume and estimate the costs and benefits as follows. According to the computerized system we propose, the costs can be broken down in two categories.

* Costs associated with the development of the system.
* Costs associated with operating the system.

**Technical feasibility**

Based on the outline design of the system requirements in terms of inputs, output, Procedures, the technical issues raised during technical feasibility include:

* Does the necessary technology exist to do what is proposed?
* Does the proposed equipment have the technical capacity to hold the data required to use in the new system?
* Adequate responses provided by the proposed system?
* Is the system flexible enough to facilitate expansion?
* Is there any technical guarantee of accuracy, reliability, ease of access and data security?

The system developer’s task is to view needed capabilities in light of currently available technology. Our site works hand in hand with high technology. A database has to be maintained in order to update and backup data whenever required. To create databases we use SQL server. After taking the above facts into consideration we can state that the new proposed system is technically feasible.

**Operational feasibility**

Operational feasibility refers to the measure of solving problems with the help of a new proposed system. It helps in taking advantage of the opportunities and fulfills the requirements as identified during the development of the project. It takes care that the management and the users support the project.

**CHAPTER -6**

**6.1 TESTING:**

Software testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results. Although crucial to software quality and widely deployed by programmers and testers, software testing still remains an art, due to limited understanding of the principles of software. The difficulty in software testing systems from the complexity of software: we can not completely test a program with moderate complexity. Testing is more than just debugging. The purpose of testing can be quality assurance, verification and validation or reliability estimation. Testing can be used as a generic metric as well. Correctness testing and reliability testing are two major areas of testing.Software testing is a trade-off between budget, time and quality.

**6.2 LEVELS OF TESTING**

**Black Box Testing**

Testing without knowledge of the internal workings of the item being tested. Tests are usually functional.

**White Box Testing**

Testing based on an analysis of internal workings and structure of a piece of software. Includes techniques such as Branch Testing and Path Testing.Also known as Structural Testing.

**Unit Testing**

Functional and reliability testing in an engineering environment.Producing tests for the behavior of components of a product to ensure their correct behavior prior to system integration.

**Integration Testing**

Testing in which modules are combined and tested as a group. Modules are typically code modules, individual’s applications, client and server applications on a network etc. Integration Testing follows unit testing and precedes system testing.

**Functional Testing**

Validating an application or website conforms to its specifications and correctly performed all its required functions. This entails a series of tests which perform a feature by feature validation of behavior, using a wide range of normal and erroneous input data. This can involve testing of the product’s user interface, APIs, database management, security, installation, networking etc. Testing can be performed on an automated or manual basis using black box or white box methodologies.

**System Testing**

Testing conducted on a complete, integrated system to evaluate the system’s compliance with its specified requirements. System Testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic.

**Acceptance Testing**

Testing to verify a product meets customer specified requirements. A customer usually does this type of testing on a product that is developed externally.

**Performance Testing**

Performance Testing can be applied to understand your application or WWW site’s scalability, or to benchmark the performance in an environment of third party products such as servers and middleware for potential purchase. This sort of testing is particularly useful to identify performance bottlenecks in high use applications. Performance Testing generally involves an automated test suite as this allows easy simulation of a variety of normal, peak and exceptional load condition.

**Stress Testing**

Testing conducted to evaluate a system or component at or beyond the limits of its specified requirements to determine the load under which it fails and how. A graceful degradation under load leading to non-catastrophic failure is the desired result. Often Stress Testing is performed using the same process as Performance Testing but employing a very high level of simulated load.

**6.3 TOOLS OF TESTING**

**Android Emulator:** Android Emulator is used to run, debug and test the android application. If you don't have the real device, it can be the best way to run, debug and test the application. The emulator available in the android SDK is not just a tool that allows you to easily test applications without having to install it to a real device, or even having one.

**Junit:** It is an open-source testing framework for java programmers. The java programmer can create test cases and test his/her own code. It is one of the unit testing framework. Current version is junit 4. To perform unit testing, we need to create test cases. The **unit test case** is a code which ensures that the program logic works as expected. The **org.junit** package contains many interfaces and classes for junit testing such as Assert, Test, Before, After etc.

**Android Mobiles:** Android mobile is used to run, debug and test the android applications. you can easily run any android application on the android mobile. But your mobile should be compatible with the minimum android version and the maximum android version which is selected by the developer.

**CHAPTER-7**

**7.1 FUTURE SCOPE:**

The vision is to build an architecture for Personal assistant for personal’s help and this will surely helps human in everyway and we can improve this project by adding more features like twitter, Booking of Cabs ,Tickets, Getting vacations spot in holiday, getting Location and many more.

### The personal assistant (CRONY) becomes better.

### Mobile-friendly responsive design websites will become the norm.

### Social media will find you as you browse.

### Search will become more social.

**5**. Person’s Tension Reduces as it helps for setting Reminders, alarm and many more.

**7.2 REFERENCES AND BIBLIOGRAPHY**

[www.github.com](http://www.github.com)

[www.stackoverflow.com](http://www.stackoverflow.com)

[www.androidstudio.com](http://www.androidstudio.com)

[www.androidprojects.com](http://www.androidprojects.com)

<http://www.webopedia.com/>

<https://netbeans.org/kb/docs/web/ajax-quickstart.html>

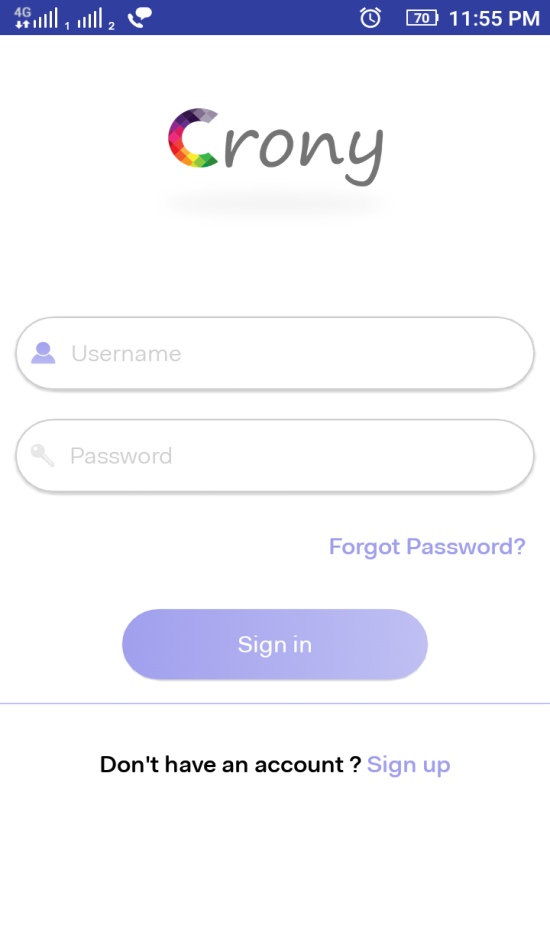
<http://www.tutorialspoint.com/servlets/>

**7.3 CONCLUSION:**

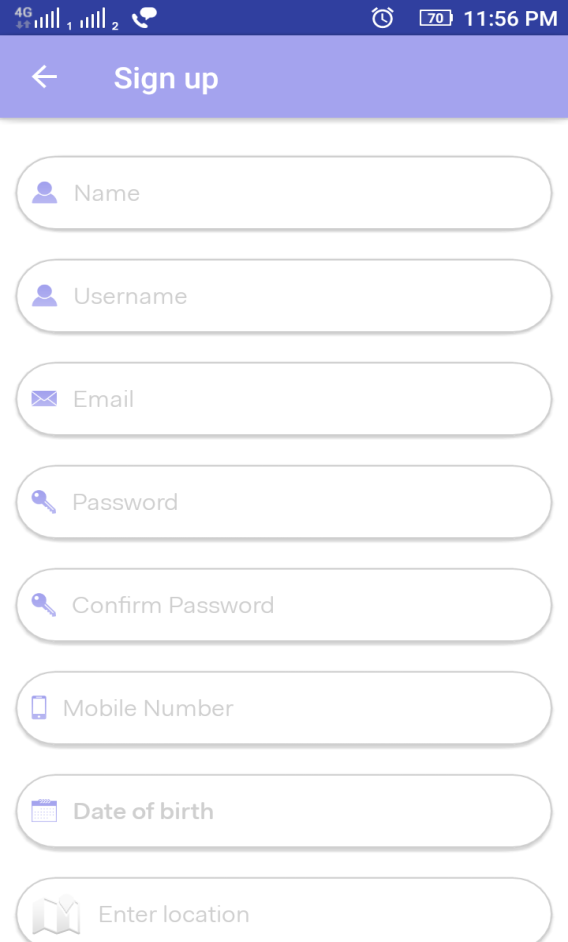
This CRONY will Definitely help Every person and will become a better assistant for person and this android application will be helpful to remove stress working as personal assistant. According to the overall description in the context, the purpose of the project is to develop an Android application that provides an personal assistant with the functionalities as calling services, message sending, mail exchange, alarm, reminders for important meetings, open music playere, NEWS, searching through google, horoscope. It will be much easier for users to communicate with their phone, and the mobile phone will be much more “Smart” as a human assistant. This project is focusing on the Android development over the Note-making by using speech to text ,shopping list with voice commands, Opening apps by using Speech-To-Text, . As all those functionalities and services for the project have been explained, the main structure and construction of the project has been basically illustrated with its goals.

**SCREENSHORTS**

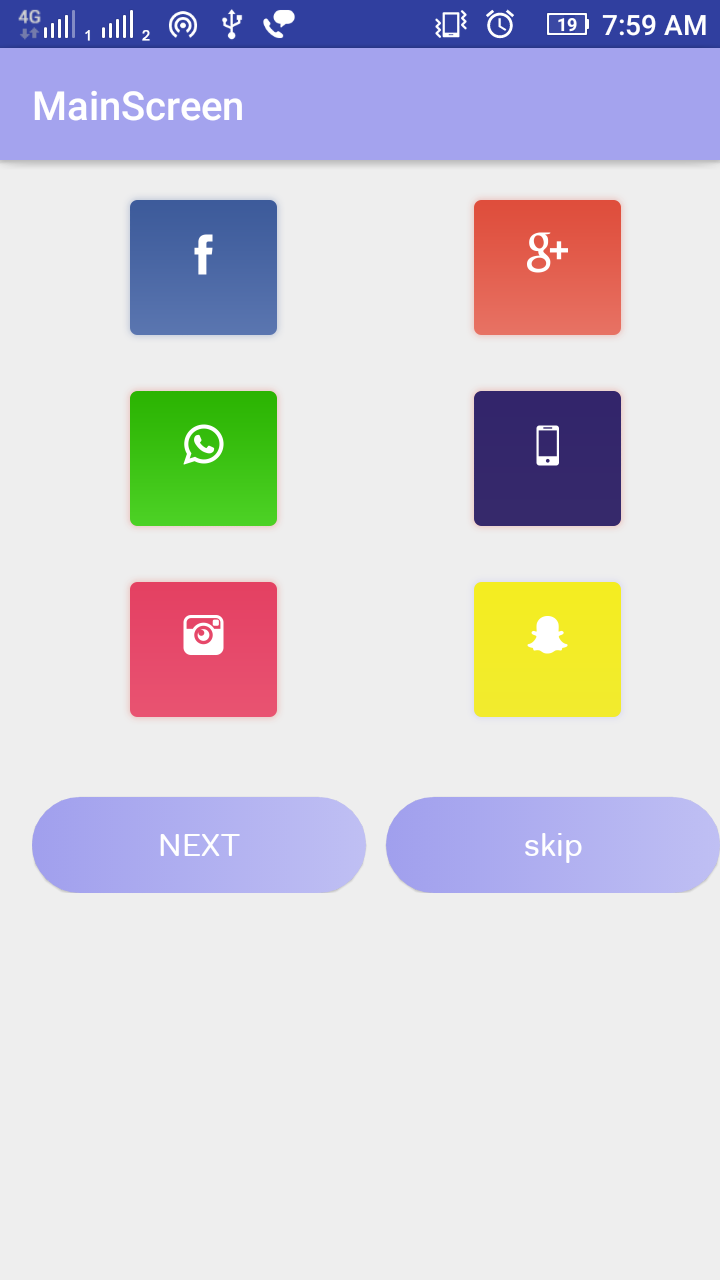
**SIGN IN SCREEN**

****

**SIGN UP SCREEN**

****

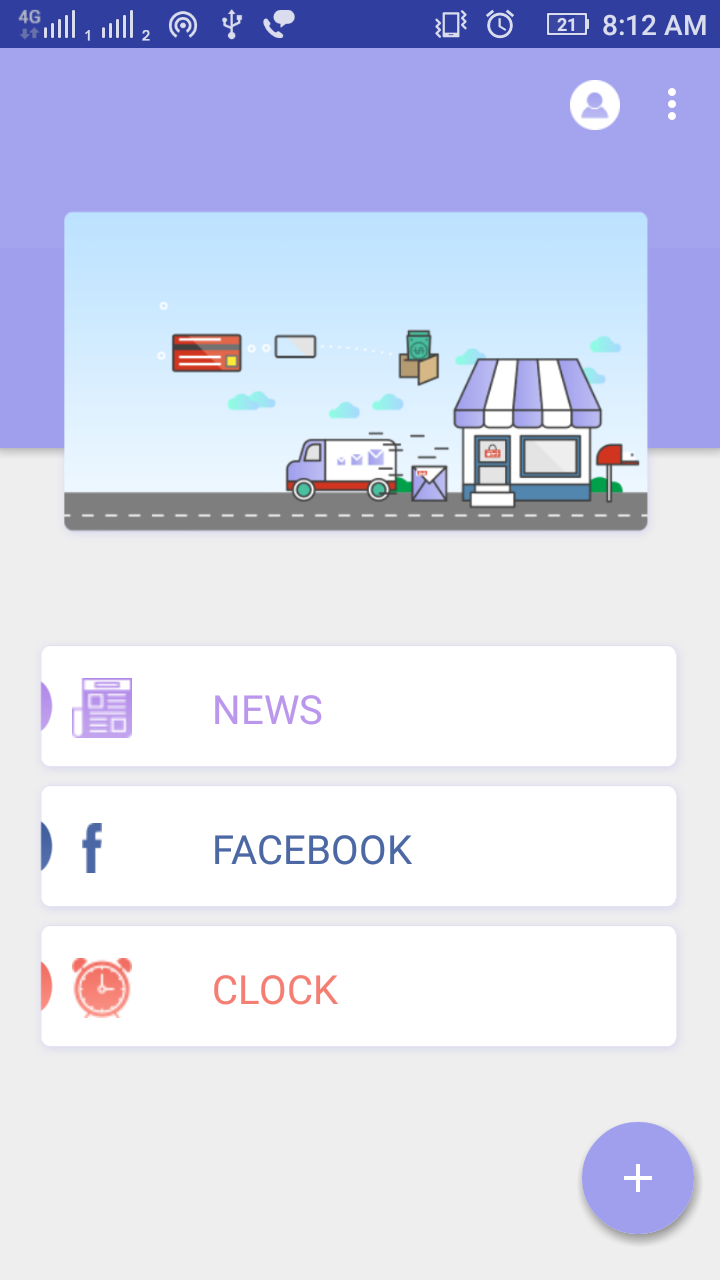
**SELECT PRIORITIES**

****

**EDIT PROFILE**

****

**HOME SCREEN**

****