A

Project Report

On

"CA HOUSE"

Submitted in Partial Fulfilment for Degree of

DIPLOMA IN COMPUTER ENGINEERING

Submitted By

Ms. Pinal V. Chikhaliya (156470307003)

Mr. Jahan A. Gagan (156470307009)

Ms. Jinal D. Patel (156470307036)

Guided By

Mr. Nilesh M. Sanghani (Internal Guide)

Mr. Sanjay V. Ramani (External Guide)

2017 -2018

Department of Computer Engineering

SHREE TAPI BRAMCHARYASHRAM SABHA

COLLEGE OF DIPLOMA ENGINEERING, SURAT

CERTIFICATE

This is to certify that **Ms. Pinal V. Chikhaliya, Mr. Jahan A. Gagan, Ms. Jinal D. Patel** from **S.T.B.S College of Diploma Engineering** having Enrolment No: **156470307003, 156470307009, 156470307036** has completed **Report on the Semester V Project Report** having title **CA House** in a group consisting of **three** persons under the guidance of Faculty Guide **Mr. Nilesh M. Sanghani.**

The mentor from the industry for the project:

Name: **Mr. Sanjay V. Ramani**

Industry: **Raksh InfoTech**

Contact Details: **88660-24228**

Institute Guide – UDP / Industry Guide – IDP **Head of Department**

Acknowledgement

The success of any task relies on the efforts made by person but it cannot be achieved without cooperation of other persons which are being helpful. So, we would like to thank **S.T.B.S College of Diploma Engineering** and **Raksh InfoTech** for giving us the opportunity of doing this project.

The entire session of our phase I completion was a great experience providing us with the insight & invocation into learning various software engineering concepts & benefits of team work. We would like to take this opportunity to express our sincere thanks to all those people without whose support and co-operation, it would have been difficult to complete this project.

Primarily, we are very much thankful to our internal project guide **Mr. Nilesh M. Sanghani** and external project guide **Mr. Sanjay V. Ramani** for their leading guidance and sincere efforts throughout project work. They took deep interest in simplifying the difficulties. Also they have been consistent source of inspiration for us.

We also express thanks to **Mr. Jaydip D. Ujainiya,** our registration faculty for his personal involvement, constructive suggestion and thoughtful idea for betterment of the project.

We are grateful to our **H.O.D. Mr. V. V. Patel** and our beloved **Principal Prof. Y. S. Choupare** for providing us deep knowledge and all necessary resources.

We are also thankful to our **Friends** and all **staff members** of Computer Engineering Department for their valuable time and help for completion project.

Once again we are grateful to all those without whom this work would not have been successful.

Abstract

CA HOUSE will be the best app for the communication between Chartered Accountant(CA) and its client. One of the advantage of using this app will be the work of CA becomes paperless and CA will be able to work with documents at any time as they will be stored inside the app. Hence, document sharing will also be implemented. Also, common instruction for all clients can also be announced inside the app rather than announcing it to the individual client. The client will always also be updated with the rules and news of different kind of TAXES. Even if the client will have any query then the app has the facility that allow client to enter his query and CA as well as other clients will be able to answer his/her query.

Index

Chapter – 1
Introduction of Project

1.1 Introduction of Problem

1.1.1 Problem Statement

Project Name :	CA House.
Project Definition/Objective :	The basic aim of project is to provide communication between CA(Chartered Accountant) and it's client through app and web based application.
Project Type :	Android Application
Develop For :	Chartered Accountant (CA)
Project Guide :	Internal Guide – Mr. Nilesh Sanghani. External Guide – Mr. Sanjay Ramani.
Front End :	Android
Back End :	PHP with MySQL
Other Technology :	Firebase
Platform :	Windows 7
Documentation Tools :	MS Office
Submitted To :	S.T.B.S College Of Diploma Engineering
Developers :	1. Jahan Gagan 2. Pinal Chikhaliya 3. Jinal Patel

1.1.2 Description

- Due to regulation of GST (Goods and Services Tax) in market, the work of CA (Chartered Accountant) also get increases. Due to increase in work, it is not possible for CA to remain in touch with each and every client. So, to overcome these problem we are going to design an Android application, due to which CA can be in regular touch with its clients.
- For example, CA wants all client to submit their GST number to him. Now instead of informing clients by calling or mailing, directly the CA will put a message in the application and all the clients will receive that message and be informed. Moreover, the client will also applicable to send the GST number pic or file to CA using these application also rather than going individually and submit it.
- After coming of GST in the market, several rules of paying tax has been changed. Now, in such case client may not be updated with latest and current rule. These problem will also resolve by these application. The application will provide all the latest and current rule or information regarding payment of tax.
- At the end the application will be beneficial in terms of client as well as in terms of CA. The stuffs of both client and CA will be reduced due to release of these application.

Chapter – 2
Industry Introductions

2.1 Company Profile

❖ About Us

- Raksh Group is a best web solutions provider company at Surat, India. The state-of-art development centre helps to serve our worldwide clients. We are specialized in customized website, online marketing & email solutions. Raksh Group provides creative design solutions that enable organizations to increase sales, build brand awareness and increase customer loyalty. We do this by employing a range of high-end tools and motivated individual to create compelling work based upon client's needs.

❖ Our Services

- If you require any work related to Web Development, E-Commerce Solution, Mobile Application Development, SEO, Digital Marketing, Domain & Hosting, Software Development and any kind of Branding then don't hesitate Please Feel Free to Contact Us.

❖ The Office

- Address: 4040, 4th Floor, Palladium Mall, Yogi Chowk, Varachha, Surat, Gujarat 395010
- Phone: (+91) 96246-28228
- Phone: (+91) 88660-24228
- Email: info@rakshinfotech.com

Chapter – 3
Project Profile

3.1 Environment Description

3.1.1 Hardware and Software Requirement

❖ Hardware Requirement

For developing the application, the following are the Hardware Requirements:

- Processor: i3 or higher
- RAM: 8 GB
- Space on disk: minimum 5 GB
- Device: Android version 4.0.3 (ICS) and higher
- Minimum space to execute: 10 MB

❖ Software Requirement

For developing the application, the following are the Software Requirements:

- Operating System: Windows 7 or higher
- Language: Android, PHP
- Database: MySQL
- Tools: Android Studio, Sublime Text
- Technologies used: Android, MySQL, PHP, Firebase, MS Office
- Debugger: Android Emulator (4.0.3) and higher
- Server: WAMP Server

3.1.2 Technology Used

- **Android OS**

- **Android** is a mobile operating system developed by Google, based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. Android's user interface is mainly based on direct manipulation, using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with a virtual keyboard for text input. In addition to touchscreen devices, Google has further developed Android TV for televisions, Android Auto for cars, and Android Wear for wrist watches, each with a specialized user interface. Variants of Android are also used on game consoles, digital cameras, PCs and other electronics.

- Initially developed by Android Inc., which Google bought in 2005, Android was unveiled in 2007, along with the founding of the Open Handset Alliance – a consortium of hardware, software, and telecommunication companies devoted to advancing open standards for mobile devices. Beginning with the first commercial Android device in September 2008, the operating system has gone through multiple major releases, with the current version being 8.0 "Oreo", released in August 2017. Android applications ("apps") can be downloaded from the Google Play store, which features over 2.7 million apps as of February 2017. Android has been the best-selling OS on tablets since 2013, and runs on the vast majority of smartphones. As of May 2017, Android has two billion monthly active users, and it has the largest installed base of any operating system.

- Android's source code is released by Google under an open source license, although most Android devices ultimately ship with a combination of free and open source and proprietary software, including proprietary software required for accessing Google services. Android is popular with technology companies that require a ready-made, low-cost and customizable operating system for high-tech devices. Its open nature has encouraged a large community of developers and enthusiasts to use the open-source code as a foundation for community-driven projects, which deliver updates to older devices, add new features for advanced users or bring Android to devices originally shipped with other operating systems. The extensive variation of hardware in Android devices causes significant delays for software upgrades, with new versions of the operating system and security patches typically taking months before reaching consumers, or sometimes not at all. The success of Android has made it a target for patent and copyright litigation between technology companies.

❖ PHP

- **PHP** is a server-side scripting language created in 1995 and designed for web development but also used as a general-purpose programming language. As of January 2013, PHP was installed on more than 240 million websites (39% of those sampled) and 2.1 million webservers. Originally created by Rasmus Lerdorf in 1994, the reference implementation of PHP (powered by the Zend Engine) is now produced by The PHP Group, While PHP originally stood for Personal Home Page, it now stands for PHP Hypertext Pre-processor, which is a recursive backronym.

- PHP code can be simply mixed with HTML code, or it can be used in combination with various templating engines and web frameworks. PHP code is usually processed by PHP interpreter, which is usually implemented as a web server's native module or a Common Gateway Interface (CGI) executable.

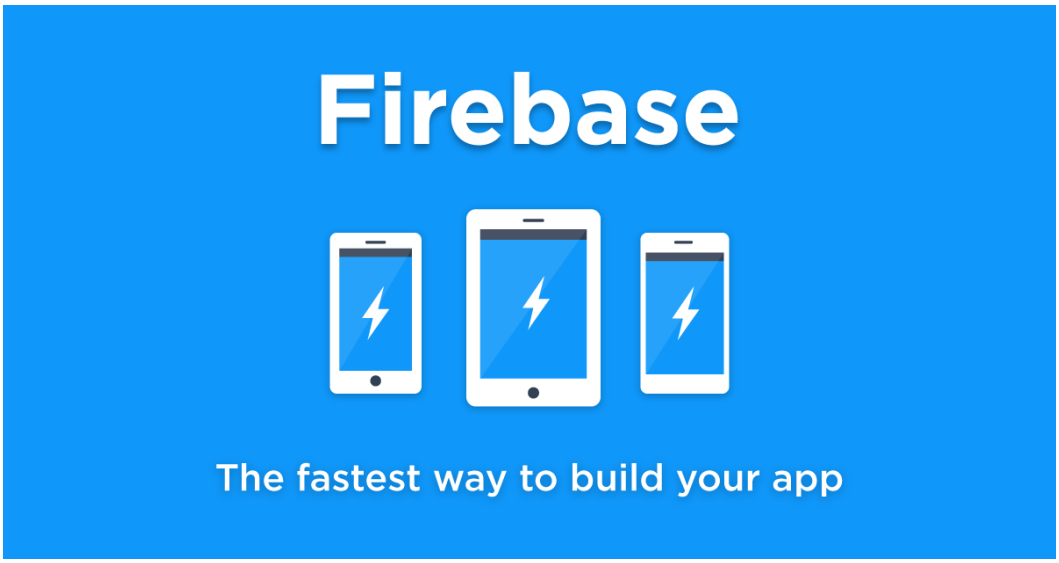
- After the PHP code is interpreted and executed, the web server sends resulting output to its client, usually in form of a part of the generated web page, for example, PHP code can generate a web page's HTML code, an image, or some other date. PHP has also evolved to include a command-line interface (CLI) capability and can be used in standalone graphical applications. The canonical PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP is a compatible with the three leading web servers:

1. Apache HTTP Server for UNIX and Windows
2. Microsoft Internet Information Server
3. Netscape Enterprise Server

❖ MySQL – My Structured Query Language

- **MySQL** is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. For proprietary use, several paid editions are available, and offer additional functionality.

❖ Firebase



- **Firebase** is a mobile and web application development platform developed by Firebase, Inc. in 2011, then acquired by Google in 2014.
- Firebase evolved from Envolve, a prior start-up founded by James Tamplin and Andrew Lee in 2011. Envolve provided developers an API that enables the integration of online chat functionality into their websites. After releasing the chat service, Tamplin and Lee found that it was being used to pass application data that wasn't chat messages. Developers were using Envolve to sync application data such as game state in real time across their users. Tamplin and Lee decided to separate the chat system and the real-time architecture that powered it. They founded Firebase as a separate company in April 2012.
- Firebase Inc. raised seed funding in May 2012. The company further raised Series A funding in June 2013. In October 2014, Firebase was acquired by Google. In October 2015, Google acquired Divshot to merge it with the Firebase team. Since the acquisition, Firebase has grown inside Google and expanded their services to become a unified platform for mobile developers. Firebase now integrates with various other Google services to offer broader products and scale for developers. In January

2017, Google acquired Fabric and Catalytic from Twitter to join those services to the Firebase team.

3.2 Existing System

3.2.1 Existing System Component

- In an existing system, the CA is carrying a lot of stuff with him such as maintaining client files or mustards to keep their document properly. Also, to announce instructions common to all clients, CA needs to announce it individually. Further, if the CA wants the client document, client had to go to CA and give it for its further process. Moreover, with the new rules and regulation of taxes, the client may not fully update with the news, rules and regulations.
- And due to documents kept at office, the CA needs to work from its office or to work somewhere else, the CA needs to carry the documents and should have to take care of it because he is carrying the sensitive documents of its client.

3.2.2 Drawbacks of Existing System

- A CA needs to maintain file or mustard for client documents.
- A CA needs to carry documents with him when needed.
- To announce common news to all clients, CA needs to announce individual client manually.
- For each new rule regarding tax, client needs to meet CA i.e., the clients are not updated with proper or complete news.
- If CA wants some client document then, client had to go to CA and provide it for its further process.
- Lot of paper work and stuffs.

3.3 System Planning

3.3.1 Feasibility Study

- The feasibility analysis is process of checking that this idea of project is able to achieve practically or not.
- It provides detailed description of the system to determine if and how it can possible to develop.
- It plays very important role in development of any system, its importance is increased when it is used as case.
- Feasibility of our project is based on Economic Feasibility, Technical Feasibility, Resources and Behavioural Feasibility.

1. Economic Feasibility

- The project is economically feasible as it only requires a mobile phone with Android operating system. The application will be free to download once released in Android market (Play Store).
- The users should be able to connect to internet through mobile phone and this would only be cost incurred on the project.

2. Technical Feasibility

- To develop this application, high speed internet connection, a database server, web server (using XAMPP), Android Studio, Sublime Text are required.
- We will use Android in-built emulator(Nougat) to compile & run this system.
- We will use android programming, PHP Scripting Language with MySQL and Firebase to make this system friendly and more reliable.

3. Resources Feasibility

- We need a proper knowledge of android language and php.
- We need some development tools like Android Studio, Sublime text & MS Office to achieve our aim.

4. Behavioural Feasibility

- The application will be behaviourally feasible since it will not require any technical guidance and all the modules will be implemented user friendly and will execute in a manner as they will design to.

5. Time

- We have one-year time to complete this project, but it is not sufficient to develop such a big and complex system.
- We prepared a schedule to accomplish this complex task in specified time period.

3.3.2 Requirement Analysis & Data Gathering

❖ Requirement Analysis

- Is this system being feasible for real application?
- Is this system requiring an active internet connection?
- This system requires any additional hardware?
- Is this system an open source?
- Is this system would be used by non-technical persons?

❖ Data Gathering

- Data gathering is the process of gathering and measuring information on variables of interest, in an established systematic fashion that enables one to answer stated research questions, test hypotheses, and evaluate outcomes.
- Data collection is a component of research in all fields of study including physical and social sciences, humanities, and business. While methods vary by discipline, the emphasis on ensuring accurate and honest collection remains the same. The goal for all data collection is to capture quality evidence that allows analysis to lead to the formulation of convincing and credible answers to the questions that have posed.

3.4 Proposed System

3.4.1 Scope

- After noticing all the expectations to this application and study this application will provide services like:
  - CA need not require individual client to inform his general instructions common to all.
  - It will make client fully updated with TAX and GST rules.
  - It will store documents inside it and hence CA will not be required to carry it through files or mustards.
  - It will provide inter-client communication.

3.4.2 Constraints

❖ Hardware Constraints

- The application will run correctly on all mobile phones manufacturing companies like Samsung, Sony, Nexus, LG, MI, Vivo, Micromax and others.
- This application demands high speed internet in the mobile phone for better usability.

❖ Software Constraints

- The application will run correctly in android 4.0.3(Ice Cream Sandwich) and higher versions. As per selected version all the clients will be able to use these application, as these version provides 100% strength of users.
- This application will be tested in all versions of android till exist now and above specification based on those testing.

3.4.3 Expected Modules

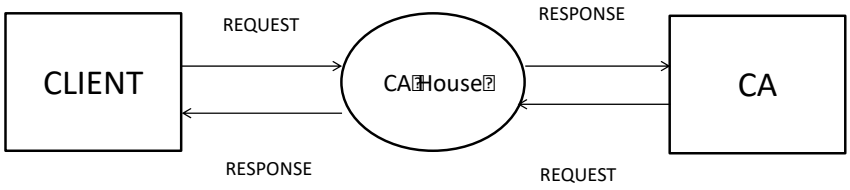
- **NEWS :** The CA will be able to announce the news and its clients will be able to read the announced news.
- **FORUM :** The client will be able to communicate with other clients and CA itself and vice-versa.
- **LOGIN :** The CA will provide login for each and every its client and based on that login client will be able to log in into the application.
- **DATA STORE :** The client will be able to store it document and access it anytime and anywhere when required.
- **DOCUMENT SHARING :** The client will be able to share documents with CA and CA will be able to share documents with client.
- **REPORT :** The client can will be able to report if there is any problem within the application to CA.

Chapter – 4
The whole Industrial process and problem study

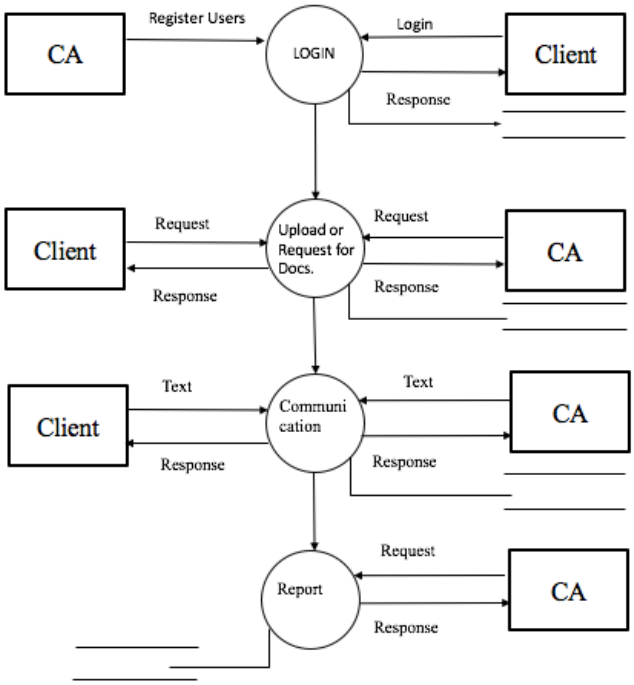
4.1 Data Flow Diagram

- A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).
- A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).

❖ Level-0 Diagram

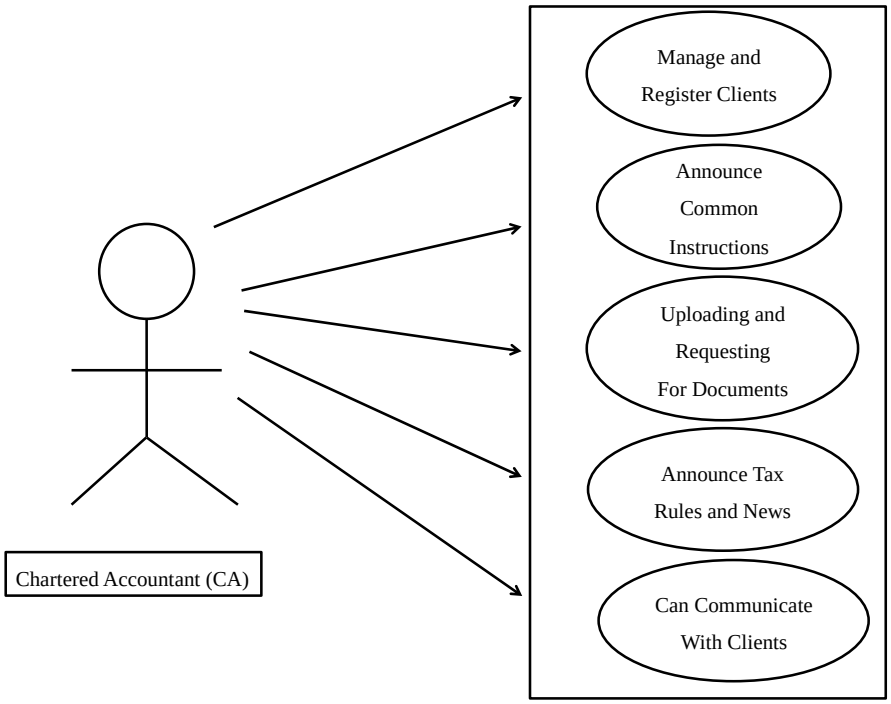


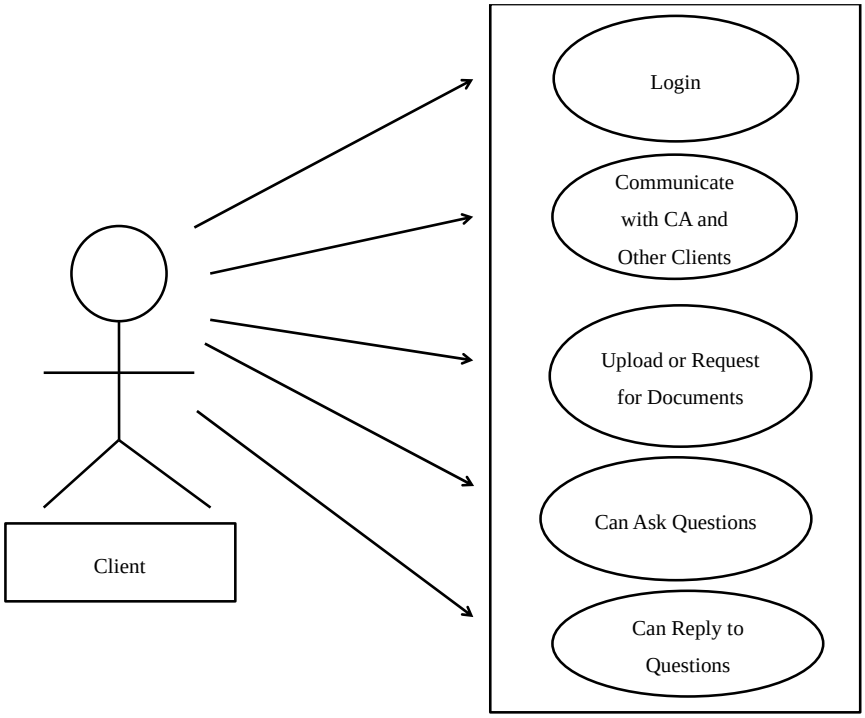
❖ Level-1 Diagram



4.2 Use – Case Diagram

- In software and systems engineering, a use case is a list of actions or event steps typically defining the interactions between a role (known in the Unified Modelling Language as an *actor*) and a system to achieve a goal. The actor can be a human or other external system. In systems engineering use cases are used at a higher level than within software engineering often representing missions or stakeholder goals. The detailed requirements may then be captured in the Systems Modelling Language (SysML) or as contractual statements.
- Use case analysis is an important and valuable requirement analysis technique that has been widely used in modern software engineering since its formal introduction by Ivar Jacobson in 1992.



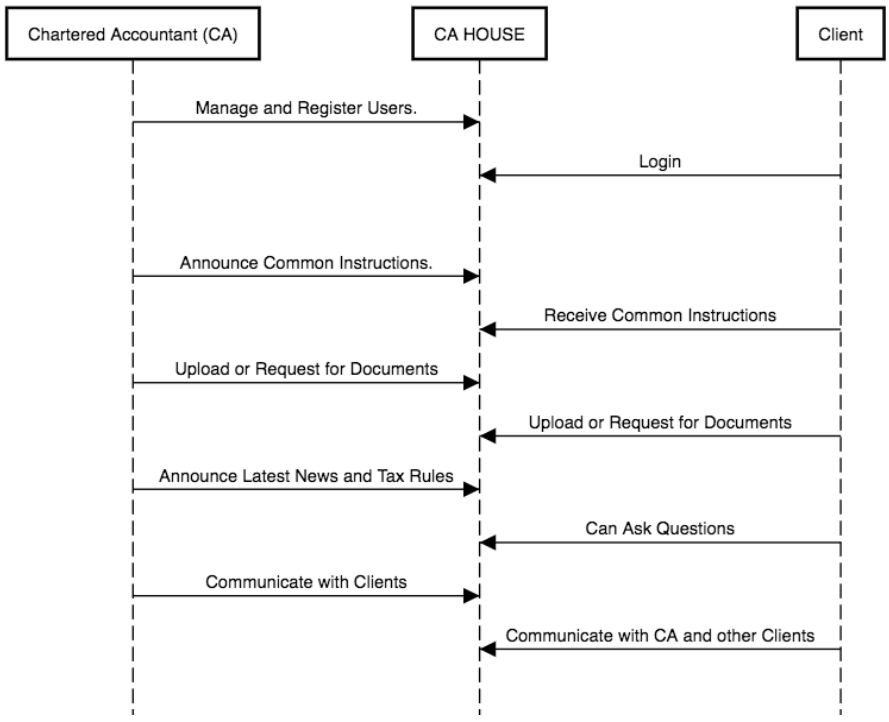

Login
Communicate
with CA and
Other Clients
Upload or Request
for Documents
Can Ask Questions
Can Reply to
Questions
Client

4.3 Sequence Diagram

- A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart.

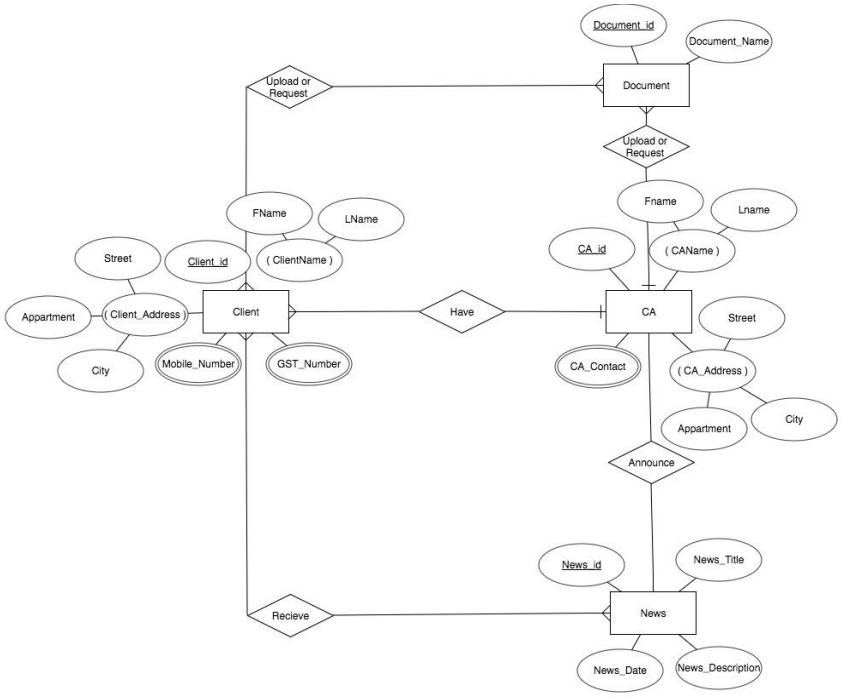
- A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

- A sequence diagram shows, as parallel vertical lines (*lifelines*), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.



4.4 Entity-Relationship Diagram

- An entity–relationship diagram (ER Diagram) describes inter-related things of interest in a specific domain of knowledge. An ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entity types.
- In software engineering an ER model is commonly formed to represent things that a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract data model that defines a data or information structure that can be implemented in a database, typically a relational database.
- Entity–relationship modelling was developed for database design by Peter Chen and published in a 1976 paper. However, variants of the idea existed previously. Some ER modellers show super and subtype entities connected by generalization-specialization relationships, and an ER model can be used also in the specification of domain-specific ontologies.



4.5 Data Dictionary

- A data dictionary, or metadata repository, as defined in the IBM Dictionary of Computing, is a "centralized repository of information about data such as meaning, relationships to other data, origin, usage, and format**."** The term can have one of several closely related meanings pertaining to database and database management system (DBMS):
  1. A document describing a database or collection of databases.
  2. An integral component of a DBMS that is required to determine its structure.
  3. A piece of middleware that extends or supplants the native data dictionary of a DBMS.

Table-Name: client_master

Column Name	Data Type	Size	Constraints
id	INTEGER	10	PRIMARY KEY, Auto Increment
fname	TEXT	50	NOT NULL
lname	TEXT	50	NOT NULL
email	TEXT	50	NOT NULL
mobile	TEXT	10	NOT NULL
password	TEXT	20	NOT NULL
gcmtokenid	VARCHAR	300	

Table-Name: news_master

Column Name	Data Type	Size	Constraints
news_id	INTEGER	10	PRIMARY KEY,

			Auto Increment
news_title	TEXT	50	NOT NULL
news_description	TEXT	100	NOT NULL
news_date	DATE		NOT NULL
news_time	TIME		NOT NULL
news_status	TEXT	50	NOT NULL

Table-Name: document_master

Column Name	Data Type	Size	Constraints
document_id	INTEGER	10	PRIMARY KEY, Auto Increment
document_name	TEXT	50	NOT NULL

Table-Name: request_admin_master

Column Name	Data Type	Size	Constraints
ca_id	INTEGER	10	PRIMARY KEY, Auto Increment
cid	INTEGER	10	NOT NULL
request_id	INTEGER	10	NOT NULL
document_id	INTEGER	10	NOT NULL
request_date	DATE		NOT NULL
requestby	TEXT	50	NOT NULL

status	TEXT	50	NOT NULL

Table-Name: request_client_master

Column Name	Data Type	Size	Constraints
cid	INTEGER	10	NOT NULL
request_id	INTEGER	10	NOT NULL
document_id	INTEGER	10	NOT NULL
request_date	DATE		NOT NULL
requestby	TEXT	50	NOT NULL
status	TEXT	50	NOT NULL

Table-Name: ca_master

Column Name	Data Type	Size	Constraints
id	INTEGER	10	PRIMARY KEY, Auto Increment
fname	TEXT	50	NOT NULL
lname	TEXT	50	NOT NULL
email	TEXT	50	NOT NULL
mobile	TEXT	10	NOT NULL
password	TEXT	50	NOT NULL
gcmtokenid	VARCHAR	300	

Table-Name: feedback_master

Column Name	Data Type	Size	Constraints
feedback_id	INTEGER	10	PRIMARY KEY
name	TEXT	50	NOT NULL
email	TEXT	50	NOT NULL
feedbackmsg	TEXT	50	NOT NULL
feedbackdate	DATE		NOT NULL
feedbacktime	TIME		NOT NULL

Table-Name: forum_question_master

Column Name	Data Type	Size	Constraints
id	INTEGER	10	PRIMARY KEY, Auto Increment
cid	INTEGER	10	NOT NULL
question	TEXT	50	NOT NULL
questionby	TEXT	50	NOT NULL
date	DATE		NOT NULL

Table-Name: forum_answer_master

Column Name	Data Type	Size	Constraints
id	INTEGER	10	PRIMARY KEY, Auto Increment
questionid	INTEGER	10	NOT NULL
cid	INTEGER	10	NOT NULL
answer	TEXT	50	NOT NULL
answerby	TEXT	50	NOT NULL
date	DATE		NOT NULL

Table-Name: upload_admin_master

Column Name	Data Type	Size	Constraints
id	INTEGER	10	PRIMARY KEY, Auto Increment
ca_id	INTEGER	10	NOT NULL
cid	INTEGER	50	NOT NULL
filename	TEXT	50	NOT NULL
uploadedby	TEXT	50	NOT NULL
date	DATE		

Table-Name: upload_client_master

Column Name	Data Type	Size	Constraints
id	INTEGER	10	PRIMARY KEY, Auto Increment
cid	INTEGER	10	NOT NULL
filename	TEXT	50	NOT NULL
uploadedby	TEXT	50	NOT NULL
date	DATE		

Chapter – 5
The problem solution outline

5.1 Input Design/Output Design

- Our project consists of two applications
    1. CA House Admin
    2. CA House Client
- Input Design/Output Design for CA House Admin

fig: Login

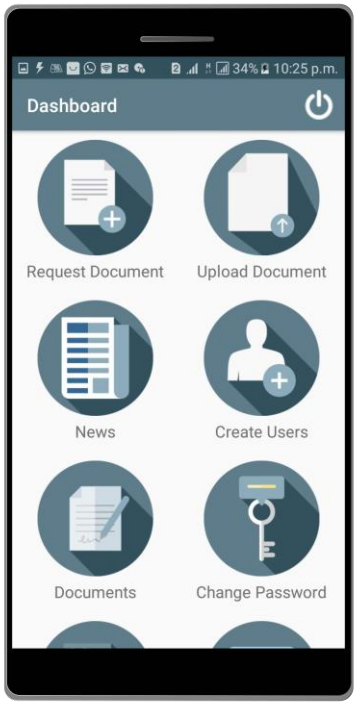


fig: CA Dashboard

- This is our login activity for CA. Login is based on mobile number and password. If the CA will enter correct mobile number and password the login will be successful and CA Dashboard will open.

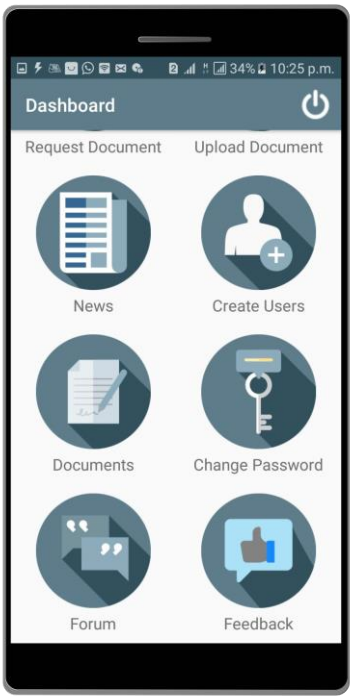


fig: CA Dashboard

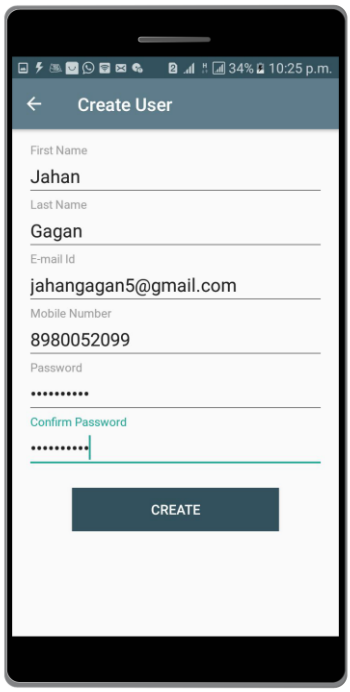


fig: Create User

- The CA will register account for its client. The Create User Activity will allow CA to enter some fields. Proper validations are assigned to each field. For Example, First Name Last Name cannot accept numbers, mobile number can accept only numbers etc.

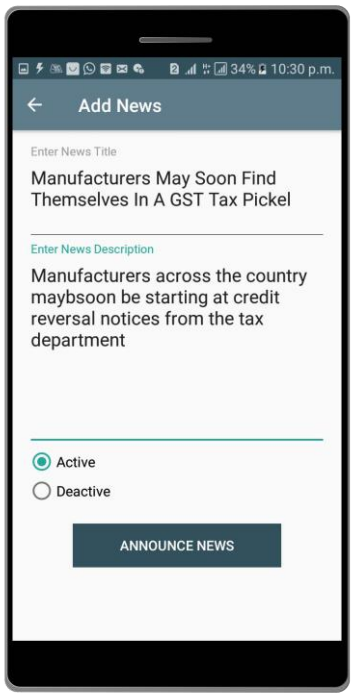


fig: Add News

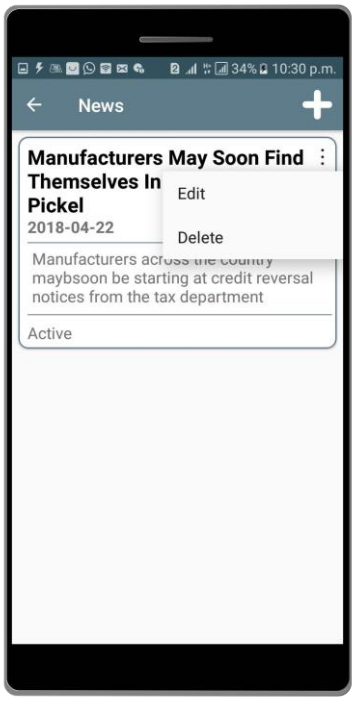


fig: News

- The CA will be allowed to announce news for its client. The News Activity will require CA enter news title, news description and select state Active or Deactive. Active means the news is created and will be displayed to client. Deactive means the news is ready but it will not have displayed to the clients.

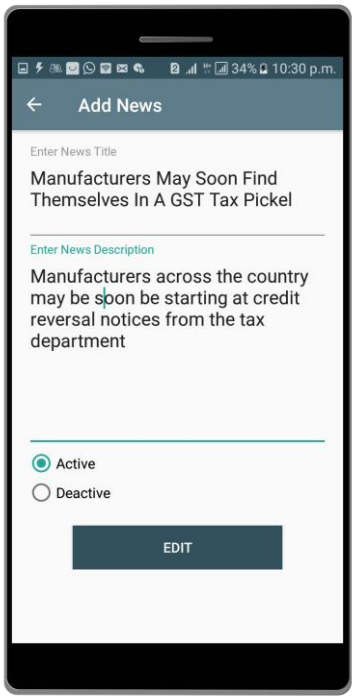


fig: Update News

fig: Delete News

- If the news is entered incorrect by CA, then he/she can update the news and can make it correct for the user. This activity will display previous news in the fields the CA just need to correct where there is a mistake and announce it. If the news is not worthy, then CA can remove the news permanently. The deleted news cannot be again retrieve.

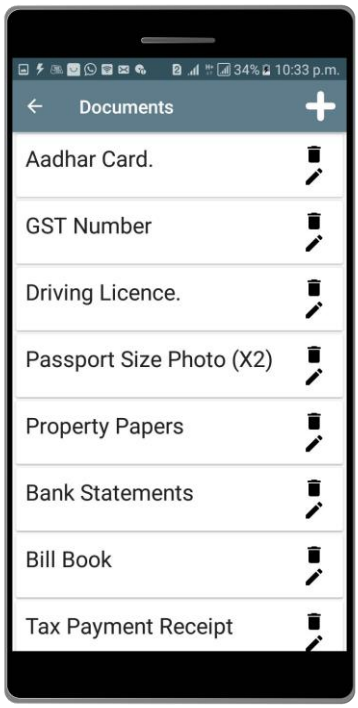


fig: Documents

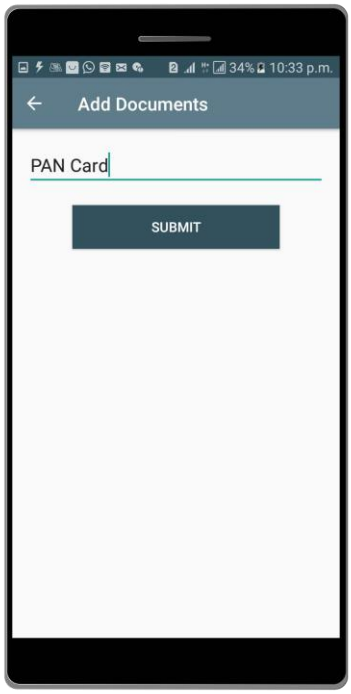


fig: Add Documents

- Documents Activity allows CA to announce documents that is going to require for calculating TAX and other stuffs. If the document is no longer needed, then that document can be removed. If the document name is entered incorrect then it could be update.

fig: Change Password

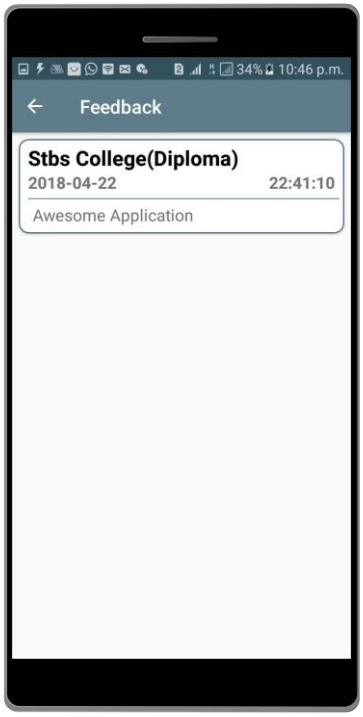


fig: Feedback

- Change Password Activity allows CA to change his/her password when required. Changing the password will require the CA to enter its old password and then new password can be set for him. After changing the password, the CA needs to login once again with the new password. Feedback Activity will display all the feedbacks of its client to CA.

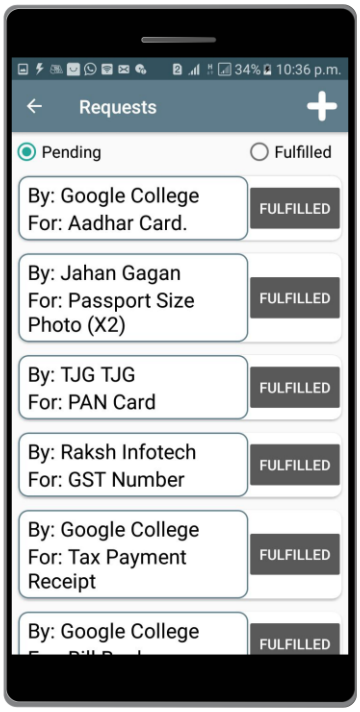
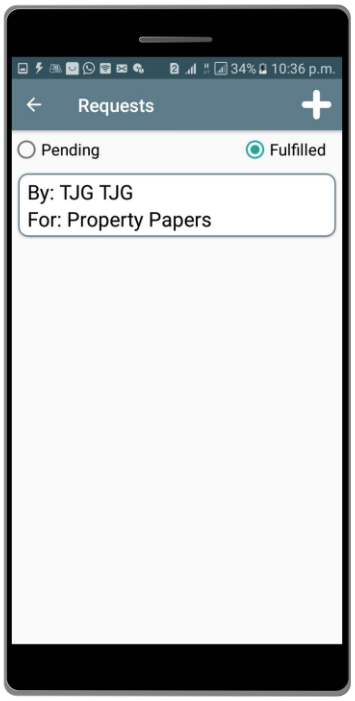


fig: Total Requests for Documents fig: Pending Requests for Documents

- Total Requests for Documents Activity will display total number of request for documents by clients to CA. Now if the CA wants to know the number of pending requests then it will select pending option and pending requests for document will be displayed. If the client requests for the document's, then the CA will get notification for the document's request.



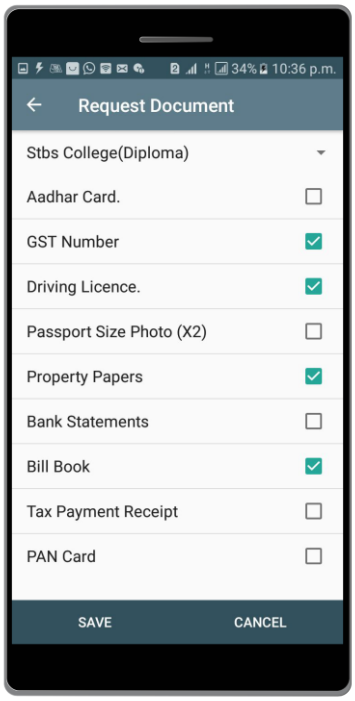


fig: Fulfilled Requested Document's

fig: Request Documents

- Fulfilled Requested Document's Activity will display the requests that is being fulfilled by the CA. In case, if CA requires documents from its client then it can also request for it. Similarly, the client will receive the notification that CA needs some documents.

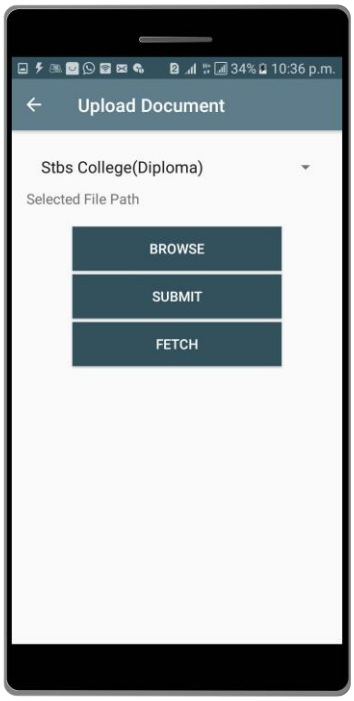


fig: Upload Document

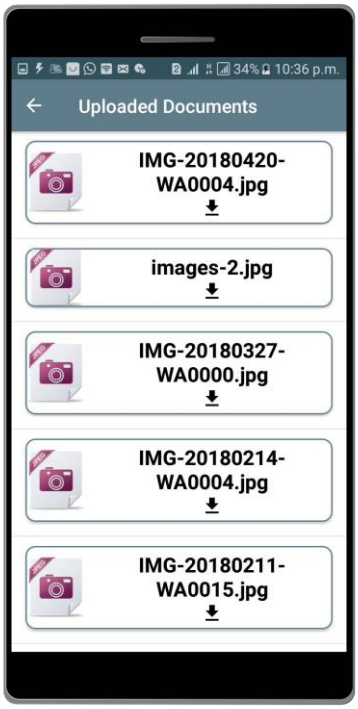


fig: Uploaded Documents

- Upload Document Activity allows CA to select client and upload requested document for his/her client. Uploaded documents will be stored on server. When the documents will be uploaded the client will receive a notification for it. The client can download uploaded document from the server and can use it whenever and wherever required using Uploaded Documents Activity.

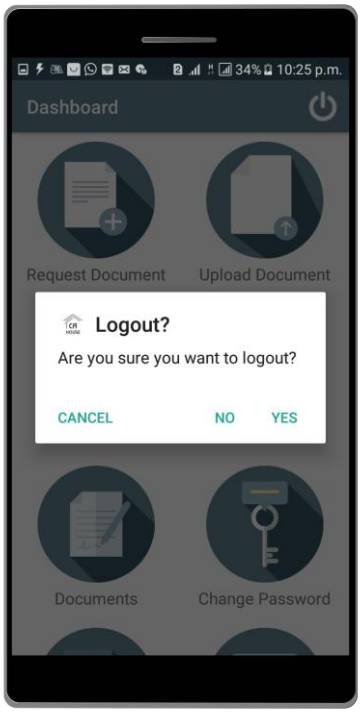


fig: Logout

- Logout Activity allows CA to logout from the application. If the CA doesn't logout from the application and closes the app and then open it again the session will be maintained and CA doesn't need to login again. But once CA logs out from the app he/she must to be login again.

❖ Input Design/Output Design for CA House Client

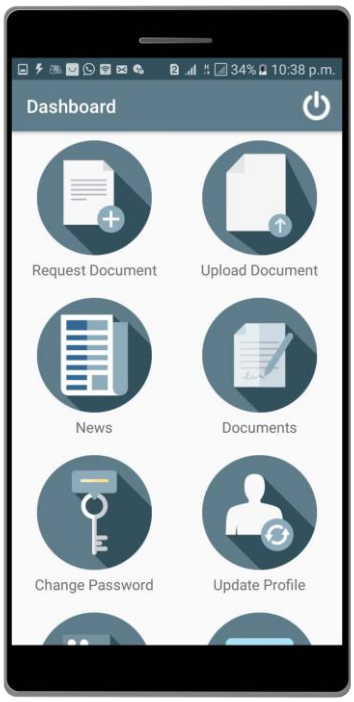


fig: Client Login fig: Client Dashboard

➢ This is our login activity for CA clients. Login is based on mobile number and password. If the client will enter correct mobile number and password the login will be successful and Client Dashboard will open.

fig: News

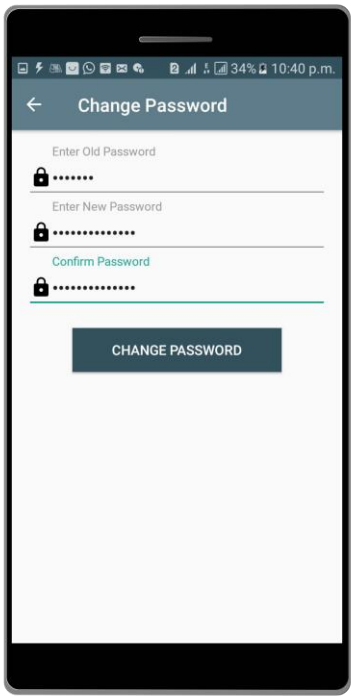
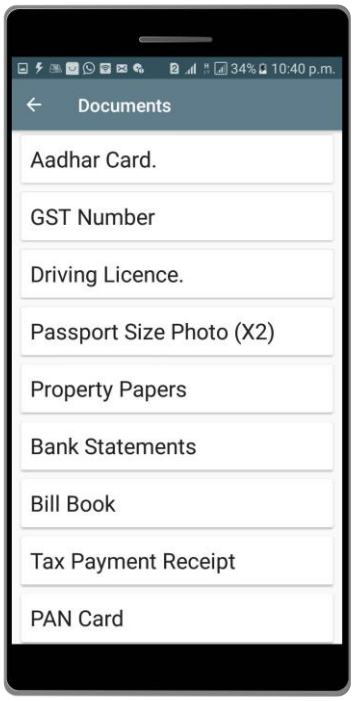


fig: Change password

- News Activity allows client to read the news announced by the CA. Change Password Activity allows client to change the password for login. Changing the password will require the client to enter its old password and then new password can be set for him. After changing the password, the client needs to login once again with the new password.



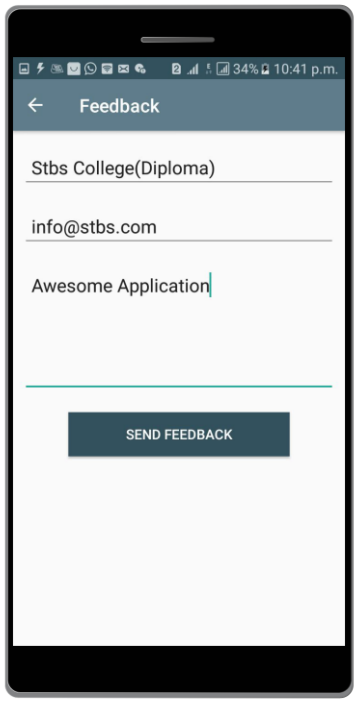


fig: Documents fig: Feedback

- Documents Activity allows client to display the list of documents to the client. Feedback Activity allows client to send the feedback of services and management through the application. The feedback can include change in service, change in management, any bugs etc.

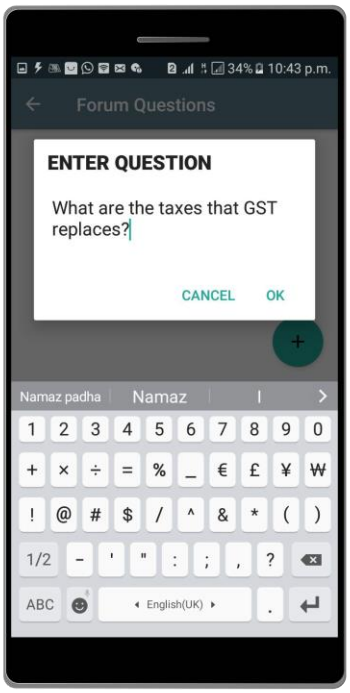
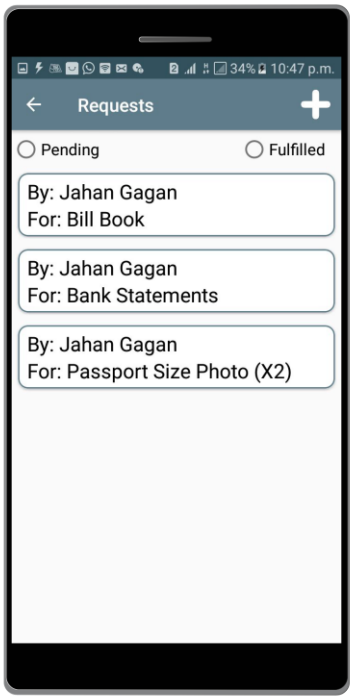


fig: Forum Question

fig: Forum Answer

- Forum Question Activity allows client to ask the question regarding any query in his mind. The client can ask any polite question. The CA as well as other clients can reply to the answer. Forum Answer Activity allows client to enter the answer regarding the query that is being asked.



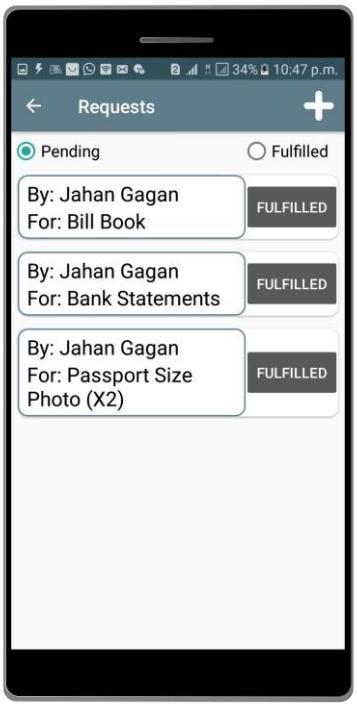
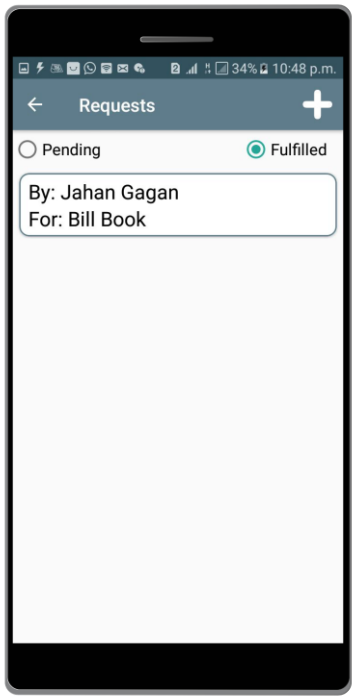


fig: Total Requests for Documents fig: Pending Requests for Documents

- Total Requests for Documents Activity will display total number of request for documents by CA to clients. Now if the client wants to know the number of pending requests then it will select pending option and pending requests for document will be displayed. If the CA requests for the document's, then the client will get notification for the document's request.



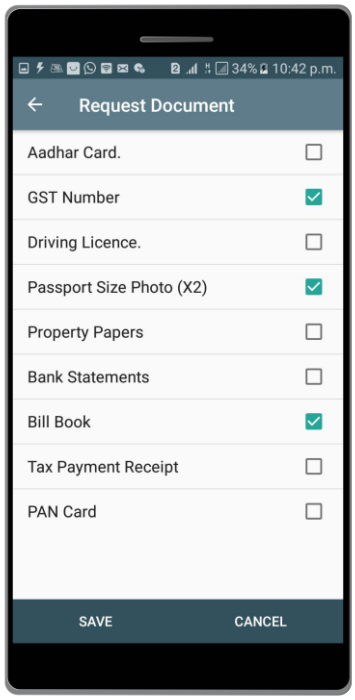


fig: Fulfilled Requested Document's

fig: Request Documents

- Fulfilled Requested Document's Activity will display the requests that is being fulfilled by the client. In case, if client requires documents from the CA then it can also request for it using Request Documents Activity. Similarly, the CA will receive the notification that client needs some documents.

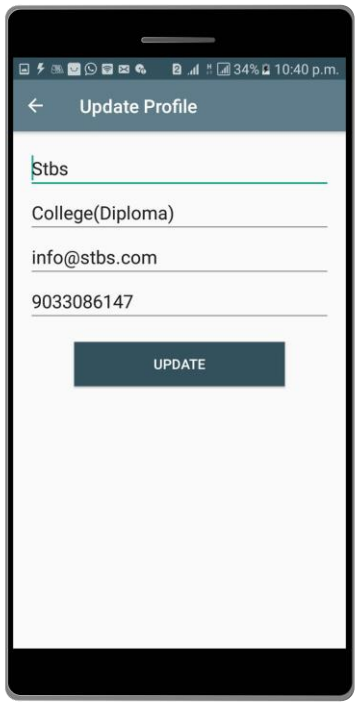


fig: Update Profile

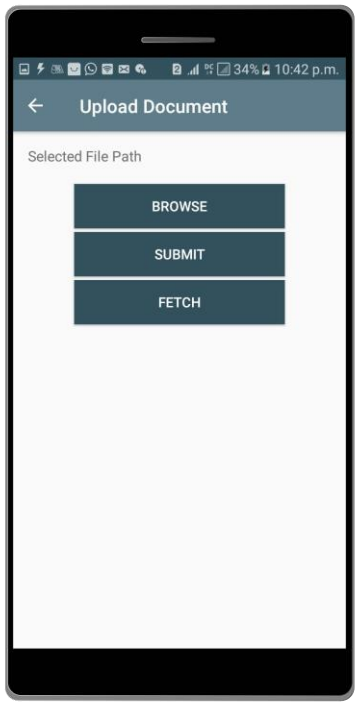


fig: Upload Documents

- Update Profile Activity allows client to update their profile. It ca be used when the email id or mobile number of client has been changed. Upload Document Activity allows user to upload the documents required to the CA.

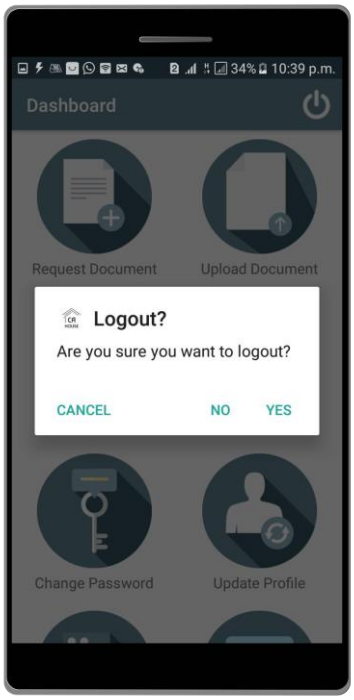


fig: Logout

- Logout Activity allows client to logout from the application. If the client doesn't logout from the application and closes the app and then open it again the session will be maintained and client doesn't need to login again. But once client logs out from the app he/she must to be login again.

5.2 Software Testing/Validation

- Software testing is a critical element of software quality assurances and represents the ultimate review of specification, design and coding. Testing is an exposure of a system to trial input to see whether it produc**es current output. Testing cannot be determined whether software meets user's** needs, only whether it appears to confirm to requirements. Testing can show that a system is free of errors, only that it contains error. Testing finds errors, it does not correct errors. Software success is a quality product, on time and within cost. Though testing can reveal critical mistakes. Testing should therefore,
  1. Validate performance.
  2. Detects error.
  3. Identify inconsistencies.

1. Test Objective

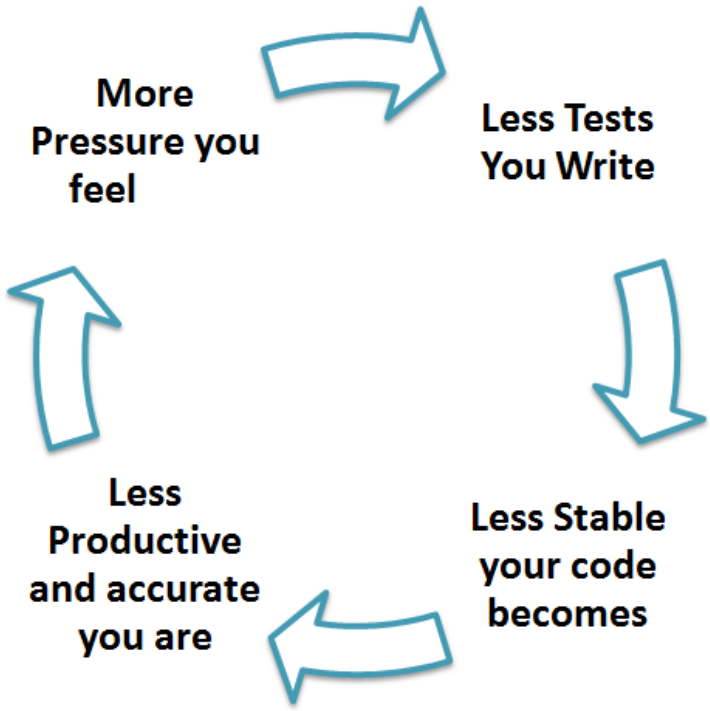
- There is strong evidence that effective requirement management leads to overall project cost savings. The three primary reasons for this are.
  1. Error in requirement typically cost over 10 times more to repair than other errors.
  2. Requirement errors typically comprise over 40% of all errors in a software project.
  3. In order to attain a flawless, error-free and efficient functioning system; too, software testing is an important phase of any software development life cycle.

- The system presented here is a blood bank management system based, various reports and data used for the same are the core of the system. The testing therefore becomes important in order to maintain the cost as well as improve performance and consistency. The testing procedure for the system has been divided in to various parts ranging for single unit testing to entire system testing.

2. Testing Principles

- All tests should be traceable to customer requirements.
- Tests should be planned long before testing begins.
- The Pareto principle applies to software testing.
- Testing should begin “in the small” and progress toward testing “in the large”. Exhaustive testing is not possible.

1. Unit Testing



- Unit testing focuses verification efforts on the smallest unit of software design- the software component or module. Using the component-level design description as a guide, important control paths are tested to uncover error within the boundary of the module. The unit test focuses on the internal processing logic and data structures within the boundaries of a component. This type of testing can be conducted in parallel for multiple components.

❖ Integration Testing

- Integration testing is a systematic technique for constructing the software architecture while at the same time conducting tests to uncover errors associated with interface. The objective is to take unit testing components and build a program structure that has been dictated by design.

- **Top- Down Integration**

➢ It is an increment approach to construction of the software architecture. Modules are integrated by moving downward through the control hierarchy, beginning with the main control module.

2. Validation Testing

➢ In validation testing, requirement established as part of software Requirements analysis are validate against the software that has been constructed. All validation criteria are tested. Validation testing provides the final assurance that software meets all functional, behavioural and performance requirements.

➢ The alpha test is conducted at the developer's site by end-used in a natural setting with the developer "looking over the shoulder" of typical users and recording errors and usage problems. It conducts in control environment.

➢ The beta test is conducted at end-user sites. Unlike alpha testing, the developer is generally not present. Therefore, the beta test is a "live" application of the software in an environment that cannot be controlled by the developer. The end-user records all problems that are encountered during beta testing and reports these to the developer at regular intervals. As a result of problems reported during beta tests, software engineers make modifications and then prepare for release of the software product to the entire customer base.

3. System Testing

➢ System testing is actually a series of different tests whose primary purpose is to fully exercise the computer- based system.

- **Recovery Testing**

➢ It is a system test that forces the software to fail in a variety of ways and verifies that recovery is properly performed. If recovery is automatic, re-initialization, crack pointing mechanisms, data recovery, and restart are evaluated for correctness. If recovery requires human intervention, the mean-time-to-repair is evaluated to determine whether it is within acceptable limit.

- **Security Testing**

➢ Security testing verifies that provides mechanisms built into a system will, in Fact, protect it from improper penetration. During security testing, the tester plays the roles of the individual who desire to penetrate the system. The role of the system designer is to make penetration cost more than the value of the information that will be obtained.

- **Stress Testing**

➢ Stress testing executes a system in manner that demands resources in abnormal quantity, frequency or volume.

- **Performance Testing**

➢ Performance testing is designed to test the run-time performance of software within the context of an integrated system. It occurs throughout all steps in the testing process. Even at unit level, the performance of an individual module may be assessed as tests are conducted. Performance tests are often coupled with stress testing and usually require both hardware and software instrumentation.

❖ **White Box Testing Principle**

➢ White-box testing sometimes called glass-box testing is a test design method that uses the control structure of the procedural design to drive test cases. Using white-box testing methods the software engineer can derive test cases that:

1. Guarantee that all independent paths within a module have been exercised at least once.
2. Exercise all logical decision on their true and false sides.
3. Execute all loops at their boundaries and within their operational bounds.
4. Exercise internal data structures to ensure their validity.
5. Testing is software quality assurance activity which is a very important to work the system successfully and achieve high quality of software. The

main objective of testing is to find yet an undiscovered error and at the same time checking the quality and reliability of system.

Chapter – 6
The outline of work to be carried future

6.1 Limitation

- Our application will require high speed internet.
- Entry in application is based on login, and if unauthorized person knows it then it can perform any malicious action with client documents.
- Proper use of mobile is necessary. If client is not eligible to use mobile phone, then it might have problem in accessing the application.

6.2 System Enhancement

- In future, we will make application eligible to support many to many communications i.e., multiple CA and multiple clients will be eligible to communicate by using these application.
- We will try to make the web application for our app so that client comfortness would be increased.
- Currently our application doesn't provide any security for client's sensitive information i.e., clients documents. So we will make these app eligible to provide a perfect security for the client documents,

Chapter – 7
Conclusion

7.1 Conclusion

The app will reduce all the worthless work of CA like maintenance of client documents, client's common announcement and make it paperless work. Also, the app itself, will store the data inside it and hence data-storage as well as document-sharing will be implemented. Moreover, the app will support the communication between client-client, client-CA and CA to Client. Further, the client will always be updated with new rules and latest news of different kind of tax.

❖ References/Bibliography

- https://developer.android.com/
- https://www.w3schools.com/php/default.asp
- https://www.tutorialspoint.com//android_online_training/android_activity_lifecycle.asp
- https://stackoverflow.com/questions/tagged/android
- https://www.quora.com/topic/Android-operating-system
- https://www.cleartax.com/