**CHAPTER 1**

**INTRODUCTION**

The project “Agent Insurance Renewal” is about how the agent views all the policies of the clients that are registered with him. The agent can use the feature of notifying the client through an E-mail or a SMS alert about the policy expiry. The alerts are prioritized with respect to the date of Renewal. The payment will be done by the client with the help of the agent to the Insurance company.

**1.1 Problem Definition**

The existing system contains client-based insurance renewal where the client can view the policy details and proceed to payment. The proposed system is designed for Agents who has a list of policy data from which the Renewal Notice can be generated and sent to the respective client. The client can receive the notice via Email or SMS. The renewal notice contains all the details of the policy along with the No Claim Bonus, Agent Commission, and other taxes with respect to the Government.

**1.2 Objective of the Project**

The main objective of the project is to reduce the work of the client and to keep them updated about the date of expiry and also the status of their policy. The agent can search any client with their respective Policy number or the Vehicle number registered with the policy.

* Reduces the work of client
* Reminders by the Agent about Policy expiry
* Calculated renewal notice sent to E-Mail

**1.3 Significance of the Project**

The Agent Insurance Renewal has major significance as follows:

* Agent based insurance renewal reminder
* Renewal notice generation
* An E-Mail and SMS based reminder
* Inclusion of payment with GST

**1.4 Outline of the Report**

Agent Insurance Renewal Application has a major outline report which is as follows:

* Accuracy
* Result to be received very quickly
* User friendly
* To speed up the operation
* Managing and maintaining data becomes easier.

**CHAPTER 2**

**LITERATURE REVIEW**

Kotlin is a statistically typed, general-purpose programming language with type inference. Kotlin mainly targets the JVM, but also compiles to JavaScript or native code (via LLVM). Kotlin is officially supported by Google for mobile development on Android. Kotlin is included as an alternative to the standard Java compiler. The Android Kotlin compiler lets the user choose between targeting Java 6 or Java 8 compatible bytecode. Firebase is a mobile and web application development platform developed by Google. Firebase Auth is a service that can authenticate users using only client-side code. It supports social login providers Facebook, GitHub, Twitter and Google (and Google Play Games). Additionally, it includes a user management system whereby developers can enable user authentication with email and password login stored with Firebase. Firebase also provides a Realtime database and backend as a service. The service provides application developers an API that allows application data to be synchronized across clients and stored on Firebase's cloud. The database is also accessible through a REST API and bindings for several JavaScript frameworks. Developers using the Realtime database can secure their data by using the company's server-side-enforced security rules.

**Advantages of the process:**

* Save Time
* Calculate bill amount based on current GST
* claim bonus
* Secured database by using firebase

**‘**

**CHAPTER 3**

**SYSTEM ANALYSIS**

System study is the process of investing a system, identifying problems, and using the information to recommend to the system. The following chapter provides a description of the existing chapter and its drawbacks. It also provides an overview of the proposed system, its advantages implementing ANDROID in “Agent Insurance Renewal App”.

**3.1 Existing System**

The existing system of Insurance renewal is mostly on the webpages and applications which are of third-party company’s application. Those application acts as a broker or an intermediate for the purpose of Insurance renewal. There is no support for E-mail or SMS notification. The existing application does not include insurance renewal amount calculation based on the current GST prices.

**3.1.1 Drawbacks**

The main drawbacks for the existing system are as follows:

* Takes more processing time.
* Not designed for specific agency.
* Does not prioritize alerts based on Expiry date.

**3.2 Proposed System**

The system is to be built with KOTLIN (a light weight scripting language) and Firebase for database. It calculates the renewal amount with respect to Government updates. It also provides SMS and Email notifications based on the expiry of the renewal date. If the client forgets to renew their policies, the notification message will be sent for the second time to ensure that the client pays their renewal amount on time. It generates a renewal notice with Included taxes, no claim bonus and other commissions. The payment will be done by the client to the Insurance company with the help of the agent. The agent also selects the type of vehicle for pursuing the renewal process.

**3.2.1 Advantages**

The major advantages for proposed system are as follows:

* + - Manage the policies on-the-go.
    - Auto-filling up of the forms with the client details.
    - Message alerts will be sent on regular basis.
    - No need to wait in long queues for renewal of policies.

**3.3 Feasibility study**

A feasibility analysis involves a detailed assessment of the need, value and practicality of a proposed enterprise, such as systems development. The process of designing and implementing record keeping systems has significant accountability and resource implications for an organization. Feasibility analysis will help you make informed and transparent decisions at crucial points during the developmental process to determine whether it is operationally, economically and technically realistic to proceed with a particular course of action.

Most feasibility studies are distinguished for both users and analysts. First, the study often presupposes that when the feasibility document is being prepared, the analyst is in a position to evaluate solutions. Second, most studies tend to overlook the confusion inherent in system development – the constraints and the assumed attitudes. For feasibility analysis, some understanding of the major requirements for the system is essential.

**3.3.1 Tests of feasibility**

Feasibility study can be done in three ways, they are:

* Economic Feasibility
* Technical Feasibility
* Operational Feasibility

**3.3.1.1 Economic Feasibility**

This study is carried out to check the economic impact will have on the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products have to be purchased.

**3.3.2 Technical Feasibility**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes for the implementing this system.

**3.3.3 Operational Feasibility**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

**CHAPTER 4**

**SYSTEM SPECIFICATION**

The system specification provides detailed information about different phases in the project. The phases in the system describes four modules of the system. They are In Client Details, login authentication, Email/SMS notification.

**4.1 Hardware specification**

The hardware requirements that are required to process the system are as follows:

|  |  |  |
| --- | --- | --- |
| Processor | : | Intel(R) Core(TM) i3 CPU |
| RAM | : | 4.00GB |
| System Type | : | 32-bit Operating System, x86-based processor |
| Monitor | : | LED screen |
| Keyboard | : | ASCII Keyboard with 108 keys |
| Mouse | : | Touch pad |
| **4.2 System specification** | | |
| Front end | : | XML and Kotlin |
| Back end | : | Firebase |
| Domain | : | Insurance |
| Frame work | : | Android studio |
| Operating system | : | Windows 7/8/10 |

**CHAPTER 5**

**SOFTWARE DESCRIPTION**

**5.1 Front End**

**5.1.1 Android Studio**

Android Studio is the [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) for [Google](https://en.wikipedia.org/wiki/Google)'s [Android](https://en.wikipedia.org/wiki/Android_(operating_system)) [operating system](https://en.wikipedia.org/wiki/Operating_system), built on [JetBrains](https://en.wikipedia.org/wiki/JetBrains)' [IntelliJ IDEA](https://en.wikipedia.org/wiki/IntelliJ_IDEA) software and designed specifically for [Android development](https://en.wikipedia.org/wiki/Android_software_development). It is available for download on [Windows](https://en.wikipedia.org/wiki/Windows), [macOS](https://en.wikipedia.org/wiki/MacOS) and [Linux](https://en.wikipedia.org/wiki/Linux) based operating systems. It is a replacement for the [Eclipse Android Development Tools](https://en.wikipedia.org/wiki/Eclipse_(software)#Android_Development_Tools) (ADT) as the primary IDE for native Android application development.

**5.1.2 Features**

* [Gradle](https://en.wikipedia.org/wiki/Gradle)-based build support
* Android-specific [refactoring](https://en.wikipedia.org/wiki/Code_refactoring) and quick fixes
* [Lint](https://en.wikipedia.org/wiki/Lint_(software)) tools to catch performance, usability, version compatibility and other problems
* [ProGuard](https://en.wikipedia.org/wiki/ProGuard_(software)) integration and app-signing capabilities
* Template-based wizards to create common Android designs and components

**5.1.3 Java**

Java is a general purpose computer programming language that is [concurrent](https://en.wikipedia.org/wiki/Concurrent_computing), [class-based](https://en.wikipedia.org/wiki/Class-based_programming), [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming), and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "[write once, run anywhere](https://en.wikipedia.org/wiki/Write_once,_run_anywhere)" (WORA),--meaning that [compiled](https://en.wikipedia.org/wiki/Compiler) Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to [bytecode](https://en.wikipedia.org/wiki/Java_bytecode) that can run on any [Java virtual machine](https://en.wikipedia.org/wiki/Java_virtual_machine) (JVM) regardless of [computer architecture](https://en.wikipedia.org/wiki/Computer_architecture).

**5.1.4 Editions**

* [Java Card](https://en.wikipedia.org/wiki/Java_Card) for smart-cards.
* [Java Platform, Micro Edition](https://en.wikipedia.org/wiki/Java_Platform,_Micro_Edition) (Java ME) – targeting environments with limited resources.
* [Java Platform, Standard Edition](https://en.wikipedia.org/wiki/Java_Platform,_Standard_Edition) (Java SE) – targeting workstation environments.
* [Java Platform, Enterprise Edition](https://en.wikipedia.org/wiki/Java_Platform,_Enterprise_Edition) (Java EE) – targeting large distributed enterprise or Internet environments.

**5.1.5 Features**

* Simple
* Familiar
* Object oriented
* High performance

**5.1.6 Kotlin**

Kotlin is a [cross-platform](https://en.wikipedia.org/wiki/Cross-platform_software), [statically typed](https://en.wikipedia.org/wiki/Static_typing), [general-purpose](https://en.wikipedia.org/wiki/General-purpose_programming_language) [programming language](https://en.wikipedia.org/wiki/Programming_language) with [type inference](https://en.wikipedia.org/wiki/Type_inference). Kotlin is designed to interoperate fully with [Java](https://en.wikipedia.org/wiki/Java_(programming_language)), and the [JVM](https://en.wikipedia.org/wiki/Java_virtual_machine) version of its [standard library](https://en.wikipedia.org/wiki/Standard_library) depends on the [Java Class Library](https://en.wikipedia.org/wiki/Java_Class_Library), but type inference allows its [syntax](https://en.wikipedia.org/wiki/Syntax_(programming_languages)) to be more concise. Kotlin mainly targets the JVM, but also compiles to [JavaScript](https://en.wikipedia.org/wiki/JavaScript) or [native code](https://en.wikipedia.org/wiki/Machine_code) (via [LLVM](https://en.wikipedia.org/wiki/LLVM)). Kotlin is sponsored by [JetBrains](https://en.wikipedia.org/wiki/JetBrains) and [Google](https://en.wikipedia.org/wiki/Google) through the Kotlin Foundation.

Kotlin is officially supported by [Google](https://en.wikipedia.org/wiki/Google) for [mobile development](https://en.wikipedia.org/wiki/Mobile_app_development) on [Android](https://en.wikipedia.org/wiki/Android_(operating_system)). Kotlin is included as an alternative to the standard Java compiler. The Android Kotlin compiler lets the user choose between targeting Java 6 or Java 8 compatible bytecode.

Kotlin relaxes Java's restriction of allowing [static](https://en.wikipedia.org/wiki/Static_(keyword)) methods and variables to exist only within a class body. Static objects and functions can be defined at the top level of the package without needing a redundant class level.

**5.1.7 Features**

* Kotlin is Open-Source
* Kotlin Supports Full Java Interoperability
* Kotlin Comes With Lazy-Loading Feature
* Data Classes in Kotlin
* Collection Filtering
* Extension Functions

**5.2 Back End**

**5.2.1 Firebase**

Firebase evolved from Envolve, a prior startup founded by James Tamplin and Andrew Lee in 2011.Firebase's first product was the Firebase Realtime Database, an API that synchronizes application data across iOS, Android, and Web devices, and stores it on Firebase's cloud. The product assists software developers in building real-time, collaborative applications. Firebase is a [mobile](https://en.wikipedia.org/wiki/Mobile_application) and [web application](https://en.wikipedia.org/wiki/Web_application) development platform developed by Firebase.

**5.2.2 Services**

* Firebase Analytics
* Firebase Cloud Messaging
* Firebase Authentication
* Realtime Database
* Firebase Storage
* Firebase Hosting

**5.2.3 Features**

* Email & password, Google, Facebook, and Github authentication.
* Realtime data.
* Ready-made api.
* Built in security at the data node level.
* File storage backed by Google Cloud Storage.
* Static file hosting.
* Treat data as streams to build highly scalable applications.

**CHAPTER 6**

**PROJECT DESCRIPTION**

**6.1 Problem Definition**

The existing system contains client-based insurance renewal where the client can view the policy details and proceed to payment. The proposed system is designed for Agents who has a list of policy data from which the Renewal Notice can be generated and sent to the respective client. The client can receive the notice via Email or SMS. The renewal notice contains all the details of the policy along with the No Claim Bonus, Agent Commission, and other taxes with respect to the Government.

**6.2 Overview of the Project**

Our Insurance renewal system validating Expiry date, policy number, estimating the GST calculation. Insurance Renewal System also maintains the database of the policy details of clients. The agent can use the feature of notifying the client through an E-mail or a SMS alert about the policy expiry. The alerts are prioritized with respect to the date of Renewal. The payment will be done by the client with the help of the agent to the Insurance company.

**6.3 Modular description**

The AGENT INSURANCE RENEWAL APP has the following modules

* Firebase authentication and fetch Realtime database
* Provide an Email /SMS alert feature for agent
* Generate a renewal notice based on Taxes, agent commission
* Inclusion of payments

**6.3.1 Firebase authentication and fetch Realtime database**

Authenticate users with their email addresses and passwords. The Firebase Authentication SDK provides methods to create and manage users that use their email addresses and passwords to sign in. Firebase Authentication also handles sending password reset emails.

The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in realtime to every connected client. When you build cross-platform apps with our iOS, Android, and JavaScript SDKs, all of your clients share one Realtime Database instance and automatically receive updates with the newest data. The Realtime database which includes the details such as

* Policy number
* Name of insurer
* Sum insured
* Mail
* Vehicle make, Model, Year
* Vehicle number
* Issued date, Expiry date
* Ex coverage amount, premium
* Tax id, amount, percent

**6.3.1.1 Policy number**

A policy number is assigned to a policy by an insurance company once you have purchased insurance from them. This number is a reference point for the insurance company.

**6.3.1.2 Name of insurer**

The policyholder name with whom an insurance contract is made, and whose interests are protected under the policy.

**6.3.1.3 Sum insured**

The sum assured is the amount of money an insurance policy guarantees to pay up before any bonuses are added. In other words, sum assured is the guaranteed amount the policyholder will receive.

**6.3.1.4 Mail**

Mail id given by the insured to send the notification regarding policy.

**6.3.1.5 Mobile Number**

Mobile number is given by the insured to send the notification regarding policy.

**6.3.1.6 Vehicle make, Model, Year**

The year in which the vehicle is created is vehicle year. The company who manufactures the vehicle is make (Maker). A company may have several model.

**6.3.1.7 Vehicle number**

A vehicle identification number (VIN) is a unique code. Identification number given for a particular vehicle for registration purpose.

**6.3.1.8 Issued date, Expiry date**

Issue date is the date on which the insurance is being taken by the client. Expiry date is the date on which the tender of the insurance ends and the renewal has to be done.

**6.3.1.9 Ex coverage amount, premium**

Ex-coverage amount is the amount of the previous insurance policies. The premium amount is the one for the renewal of the policies.

**6.3.1.10 Tax id, amount, percent**

Tax id is the one which indicates the id of the amount of tax paid or to be paid by the customer/client. The amount is the one which has to be paid by the customer/client. Percent is the one the percentage of tax paid or to be paid by the customer/client.

* + 1. **Provide an Email /SMS alert feature for agent**

The application is built in such a way that it provides an email or SMS alert to the agent when the expiry date of the insurance policy nears. The existing system doesn’t have the facility of sending the notifications via Email or SMS. The system is built in such a way that the alert notification will be sent before a month of the expiry date. The notification will also be sent when the client fails to renew the insurance policy.

* + 1. **Generate a renewal notice based on Taxes, agent commission**

The system is built in such a way that it sends the notification based on the taxes revised by the Government and it calculates and sends the premium amount based on the agent commission. For example, if the percentage of the GST for the insurance renewal is 5% the amount for the renewal will be calculated along with that percentage.

* + 1. **Inclusion of payments**

The inclusion of the payments is that all the miscellaneous payments regarding the renewal of the policies which are undertaken by the client or customer for their motor vehicle. The payment can be done with the revised GST amount and the other taxes.

**6.4 Data Flow Diagram**

Database

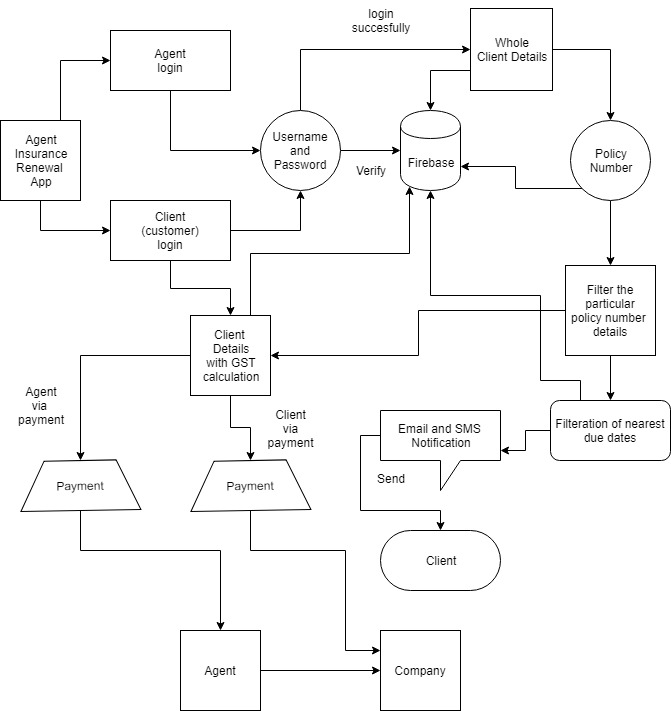
Client Details

Payment

SMS / Mail

Notification

**6.4.1 DFD Level 0**



**6.4.2 DFD LEVEL 1**

**6.5 Database design**

Database design is the process of producing a detailed data model of database. This data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity.

The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data. In the relational model these are the tables and views. In an object database the entities and relationships map directly to object classes and named relationships. However, the term database design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall database application within the Firebase.

**Table details**

**6.5.1 Client Database**

|  |  |
| --- | --- |
| **FIELD NAME** | **TYPE** |
| VCH\_POLICY\_NO | String |
| DTT\_ISSUE\_DATE | String |
| DTT\_EXPIRY\_DATE | String |
| DTT\_INCEPTION\_DATE | String |
| VCH\_VEHICLE\_NO | String |
| NUM\_MANUFACTURE\_YEAR | String |
| VCH\_ENGINE\_NO | String |
| VCH\_CHASSIS\_NO | String |
| NUM\_SERVICE\_TAX\_ID | String |
| NUM\_SERVICE\_TAX\_PERCENT | String |
| NUM\_SERVICE\_TAX\_AMOUNT | String |
| NUM\_MODEL\_ID | String |
| EMAIL\_ID | String |
| MOB\_NUM | String |
| NUM\_EX\_COVERAGE\_AMT | String |
| NUM\_EX\_COVERAGE\_PREMIUM | String |
| NUM\_EX\_COVERAGE\_TOT | String |
| VCH\_INSURED\_NAME | String |
| VCH\_INSURED AMOUNT | String |

**6.5.2 Login Authentication**

|  |  |  |
| --- | --- | --- |
| **IDENTIFIER** | **PROVIDERS** | **USER** |
| **arun@gmail.com** | **Email** | **Agent** |
|  |  |  |

**6.6 Input design**

Input design is the one of the most important phases of the system design. Input design is the process of connecting the agent to the client by getting the information from the client. The aim of the input design is to ensure the maximum possible level of accuracy and also ensures that input is accessible that understood by the agent and clients. The input design is the part of overall system design, which requires careful attention. Input design features can ensure the reliability of the system and produce result from accurate data or they can result in the production of erroneous information.

**6.7 Output design**

The output design presents the manipulated data to the end user. The output design act as medium of communication to the agent. A quality output is one, which meets the requirement of the end user and presents the information clearly. In any system results of processing are communicated to the agents and to other systems through outputs. The output design deals with determining how the information is to be displayed for immediate need and also for the hard copy output.

**CHAPTER 7**

**SYSTEM TESTING**

This chapter describes the system testing used in the project and types of testing analysed to reduce the inefficiency of the project. The system testing includes the testing method and testing strategies used in project.

**7.1 Testing methods**

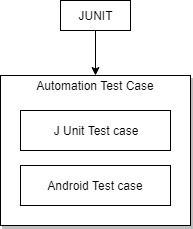
Testing the behaviour of whole software/system as defined in software requirement specification (SRS) is known as system testing, its focus is to verify the customer requirements are fulfilled. System testing is done after integration testing as complete. System testing should test functional and non-functional requirements of the software.

Testing the software system or software application as a whole is referred as a System Testing of the software. Software testing of the application is done on complete application software to evaluate software’s overall compliance with the business, functional and end-user requirements. The system testing comes under black box software testing. Therefore, the knowledge of internal design or structure or code is not required for this type of testing.

**7.2 Types of testing**

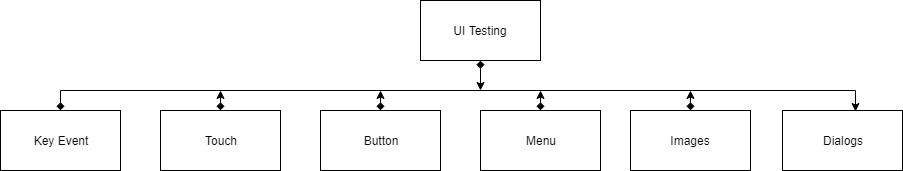
**7.2.1 Unit testing**

Unit Tests include sets of one or more programs which are designed to verify an atomic unit of source code, such as a method or a class. Unit testing focuses on the smallest unit of the software design. It’s open source framework for automating unit testing. Android Testing Framework is powerful tool for developer to write the effective unit test program.



**Fig 7.1 Unit Testing**

An addition to Unit testing is User Interface (UI) tests. These relate to UI components of your target application. UI tests ensure that your application return the correct UI output in response to sequence of user actions on device.

\

**Fig 7.2 Unit Testing**

The common way to performance UI tests on device is Android instrumentation. But this has performance issues. One of the best tools to conduct UI testing on Android is Robotium.

**Test Result:** The above-mentioned test is carried out and no defects are found.

**7.2.2 Integration testing**

Integration Testing is a level of software testing are individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. Test drivers and test stubs are used to assist in Integration Testing.

Types of Integration Testing:

* component integration testing
* system integration testing
* **Component integration testing:** Testing performed to expose defectsin the interfaces and interaction between integrated components.
* **System integration testing:** Testing the integration of systems andpackages; testing interfaces to external organizations (e.g. Electronic Data Interchange, Internet).

**7.2.3 Functional testing**

Functional testing is a type of testing which verifies that each function of the software application operates in conformance with the requirement specification. This testing mainly involves black box testing and it is not concerned about the source code of the application. It involves exercising the code with nominal input values for which the expected results are known, as well as boundary values and special values, such as logically related inputs, files of identical elements and empty files. This testing involves checking of User Interface, APIs, Database, security, client/ server applications and functionality of the Application under test. The testing can be done either manually or using automation.

**7.2.4 Stress testing**

Stress testing (sometimes called torture testing) is a form of deliberately intense or thorough testing used to determine the stability of a given system or entity. It involves testing beyond normal operational capacity, often to a breaking point, in order to observe the results.

The test results found while undergoing stress testing in our app are

* the response of the application when the device orientation is changed continuously in different angles.
* the app screen is loading continuously without any network and data failures.
* the application is not crashing while scrolling the list up and down rigorously.
* the application does not get failed when the user performs multiple operations for different data cells.

**7.2.5 White box testing**

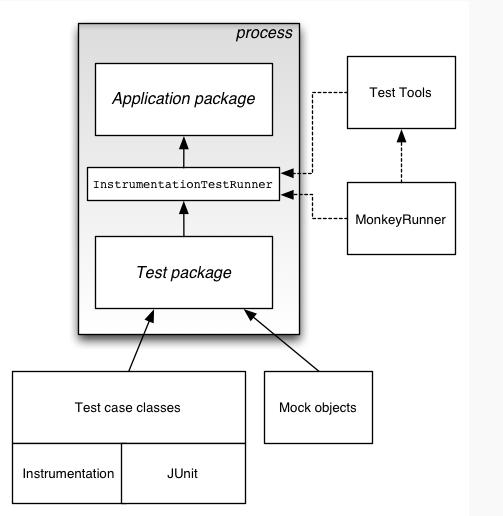
White Box Testing is the testing of a software solution's internal coding and infrastructure. It focuses primarily on strengthening security, the flow of inputs and outputs through the application, and improving design and usability. In our Insurance Renewal App, white box testing involves the testing of the software code. In testing internal security holes are managed and poorly structured paths in the coding process is rectified. The flow of specific input through the code is tested and expected outcome is also predicted. The proposed system involves testing a series of predefined inputs against expected or desired outputs so that when a specific input does not result in the expected output. Initially the source code is understood and test cases are created and executed. In the proposed system white box testing is done by code coverage analysis which checks every possible path (if-else and other conditional loops) of a software application.

**7.3 Testing strategy**

The testing strategy includes the different types of testing which is used in the project to improve the efficiency.

**7.3.1 Android testing framework**

One of the standard testing frameworks for Android application is Android testing framework. It is a powerful and easy-to-use testing framework that is well integrated with the Android SDK tools.



**Fig 7.3 Android testing framework architecture**

1. **Mobile Application for agents** is the target application which needs to betested.
2. **Instrumentation Test Runner** is the test case runner that executes test caseon target application.

It includes:

* 1. **Test Tools:** ASDK tools for building test. They are integrated inandroid studio.

1. **Monkey Runner:** a tool that provides API’s for writing program

which control an android device or emulator outside of android code.

1. **Test package** are organized into test projects. This package follows naming

convention. The application under test has a package name of “com.example.” than Test package should be “” Test package includes 2 objects as below:

1. **Test case classes:** Include test methods to execute target application.
   1. **Mock objects:** Includes mock data that will be used as sample inputfor test case.

**CHAPTER 8**

**SYSTEM IMPLEMENTATION**

System implementation is the construction of the new system by considering the existing system and implementing it. The most important task in a project is a delivery of the given system within the allotted deadline. Implementation in the phase in which one has to be cautions because all the efforts undertaken during the project will be useful only if the software is properly implemented according to the plan made. The construction phase does two things: build and test a functional system that gives business and organizational design requirements and implement the interface between the new system and existing production system.

System testing of the software or hardware is a testing being conducted on a complete, integrated system to evaluate the system’s compliance with its specified requirements. In system testing the behaviour of the whole system is tested as defined by the scope of the development project. It may include tests based the risk and requirements specifications, business process, use cases, or other high-level description of the system behaviour, interaction with the operating system and the system resources. System testing is most often the final test to verify that the system to be delivered meets the specification and its purpose. System testing is carried out by specialist’s tester or independent tester. It should investigate both functional and non-functional requirement of the testing.

**8.1 Quality assurance**

The aim of quality assurance is to maintain or to ensure the quality of the system being developed. It checks whether it satisfies the requirement and meet organizational goals. The quality assurance goal in the system life cycle involves the following components:

**Quality factors specifications** was done to determine the factors thatlead to high quality of a system.

**Correctness** is the extent to which a program meets system specification.

**Reliability** is the degree to which a program meets system specification.

**Efficiency** is the amount of computer resource required by the entireprogram to perform a function.

**Usability** is the effort required learning and operating the system.

**Maintainability** is the ease with which the program errors are corrected.

**Testability** is the effort required to test a program to ensure its correctperformance.

**Portability** is the case of transporting a program from one hardwareconfiguration to another.

**Accuracy** is required precision in input editing, computation, and output.

**Error Tolerance** is the error correction and deduction versus erroravoidance.

**Expandability** is the ease adding or expanding existing database.

**Access control and adult** is the control of access to the system and theextent to which that access can be audited.

**8.2 System maintenance**

Software maintenance in software engineering is the modification of a software product after a delivery to correct faults, to improve performance or other attributes. The results obtained from the evaluation process help the organization to determine whether its information systems are effective and efficient. The process of monitoring, evaluating and modifying of existing information system to make required or desirable improvement may be termed as System maintenance.

A common perception of maintenance is that it merely involves fixing defects. However, one study indicated that over 80% of maintenance effort is used for non-corrective action. This perception is perpetuated by users submitting problem reports that in reality are functionality enhancement to the system.

System maintenance is an ongoing activity which cover a wide variety of activities including removing program and design errors, updating documentation and test data and updating user support. For the purpose of convenience, maintenance may be categorized into three classes namely,

* **Adaptive** –Modifying system to cope with changes in the softwareenvironment.
* **Perfective** –Implementing new or changed user requirements whichconcern functional enhancement to the software.
* **Corrective** –Diagnosing and fixing errors, possibly once found by auser.
* **Preventive** –Increasing software maintainability or reliability to preventproblem in the future.

An integral part of software is the maintenance one, which requires an accurate maintenance plan to be prepared during the software development. It should specify how user will request modifications or report problem. The budget should include resource and cost estimate. A new decision should be addressed for the developing of every new system feature and its quality objective. The software maintenance which can last for 5 to 6 years or even decades, after the development process the call for an effective plan which can address the scope of software maintenance. The tailoring of the post-delivery/deployment process. The designation of who will provide maintenance and an estimate of the lifecycle cost. The selection of proper enforcement of standards is the challenging task right from the early stage of software engineering which has not got definite importance by the concerned stakeholder.

**CHAPTER 9**

**CONCLUSION AND FUTURE ENHANCEMENTS**

**9.1 Conclusion**

The Agent Insurance Renewal helps the agent as well as the client to renew their policies on the go. They need not spend separate time in waiting in long queues for the renewal process. As the input data here is only the policy number, it will be easy for the agent to know about the policies that have been taken by the particular client. Here, If the client forgets to renew their policies, the renewal alert will be sent to the client for the second time to remind them.

**CHAPTER 9**

**APPENDIX**

**9.1 Source code**