**Project Proposal On**

**Online Vehicle Tax Pay System**

****

Submitted By: Submitted To:

Bishesh Man Singh Sudeep Lal Bajimaya

NCC ID: 00175019 Module leader

Softwarica College of IT and E-commerce

Contents

[**Chapter1. Introduction** 4](#_Toc5528855)

[1.1 Project Introduction 4](#_Toc5528856)

[1.2 Background of the project 4](#_Toc5528857)

[1.3 Problem Statement 4](#_Toc5528858)

[1.4 Description of the Project 5](#_Toc5528859)

[1.4.1 Feature of the Project 5](#_Toc5528860)

[1.5 Overview of the Project 6](#_Toc5528861)

[**Chapter2. Scope of the Project** 7](#_Toc5528862)

[2.1 Scope 7](#_Toc5528863)

[2.2 Limitations 7](#_Toc5528864)

[2.3 Aims 7](#_Toc5528865)

[2.4 Objectives 8](#_Toc5528866)

[2.5 Overview of the Scope 8](#_Toc5528867)

[**Chapter3. Development Methodology** 9](#_Toc5528868)

[3.1 Description of the Methodology 9](#_Toc5528869)

[3.2 Design Pattern 10](#_Toc5528870)

[3.3 System Architecture 11](#_Toc5528871)

[**Chapter4. Project Planning** 13](#_Toc5528872)

[4.1 Work Breakdown Structure (WBS) 13](#_Toc5528873)

[4.2 Milestones 14](#_Toc5528874)

[4.3 Gantt Chart 15](#_Toc5528875)

[**Chapter5. Risk Management** 17](#_Toc5528876)

[**Chapter6. Configuration Management** 19](#_Toc5528877)

[**Chapter7. Conclusion** 21](#_Toc5528878)

[**Chapter8. References** 22](#_Toc5528879)

List of Screenshots

[Screenshot 1: Waterfall Model 9](#_Toc5528008)

[Screenshot 2: MVC design pattern 10](file:///C:\Users\PRABIN\Documents\GitHub\EMS\Proposal\00174621_CP_Proposal.docx#_Toc5528009)

[Screenshot 3: Three-tier Architecture 12](#_Toc5528010)

[Screenshot 4: Work Breakdown Structure for Expenses Management System 13](file:///C:\Users\PRABIN\Documents\GitHub\EMS\Proposal\00174621_CP_Proposal.docx#_Toc5528011)

[Screenshot 5: Schedule of Expenses Management System 15](#_Toc5528012)

[Screenshot 6: Gantt chart of Expenses Management System 16](#_Toc5528013)

[Screenshot 7: Location for Git Push 19](#_Toc5528014)

[Screenshot 8: Tree structure for Expenses Management System 20](#_Toc5528015)

# **Chapter1. Introduction**

## 1.1 Project Introduction

Online Vehicle Tax Pay System is the system for paying vehicle tax online. This System will let owner pay the tax of their vehicle without getting in queue for hours. Vehicle Insurance is also processed through this system. This project will save valuable time of Vehicle Owner.

## 1.2 Background of the project

People are very busy nowadays. They want their life to be easier and simpler. In this busy life schedule, most of their time passes while standing in queue for many specific purpose. One of them is for paying their vehicle tax. So I intend to decrease this waste time. Here in this system, Owner need to enter their vehicle information and owner’s information like phone number, email id, address. The information will be saved in our system and will get their tax paid.

## 1.3 Problem Statement

Many of the Government System are now online but the process of paying vehicle tax is still in old way i.e standing in queue and wait for the turn. This consumes time of every individual. This is the problem in current system. So this project will focus only on possible way of paying vehicle tax efficient. With this project, Owner can be sure that their vehicle tax will be paid in time without any effort.

## 1.4 Description of the Project

Description of the project include following requirements:

**Front End Language: HTML, CSS, Javascript**

**Programming Language: PHP (Laravel Framework)**

**Tool: Dreamweaver**

**Platform: Web Based**

**Pattern: MVC (Model, View and Controller)**

**Database: MySQL**

For this Project I will use HTML, CSS, and Bootstrap for completing my frontend task while I will be using PHP for backend and MySQL for database. The System will be based on web. Tools like Dreamweaver will be used during this whole project.

## 

## 1.4.1 Feature of the Project

Features of the Expenses Management System include following things:

* **Register**
* For register user using their Email Address, Username and Password.
* New information regarding clients will sent to Email.
* **Login**
* For login purpose, user will using their Username and Password.
* **User will edit their profile**
* Profile of user include Username, Password, Email and many other things that related to user profile.
* User can edit their salary and expenses according to their needs.
* **User will add their income**
* For calculation of salary and expenses user will add their income.
* **User will edit their income**
* If there is increase or decrease in salary user will edit their income according to their needs.
* **User will add their expenses**
* **User will view expenses**
* This feature help user to view the list of expenses.
* Provide information about where the money is going.
* **Report of the expenses**
* User will get report of the expenses by calculating all expenses.
* **Logout**

## 

## 1.5 Overview of the Project

Expenses management system will used for business purposes or a person to record expenses of user. It helps to record how user spend money and their spending habits. Monitor costs all year to augment reasoning’s, lower assessable pay and spare your organization cash.

Here I will used different things to complete this project. Some of them are Java (programming language), Android Studio (tool), Android (platform), MVC (pattern), and SQL Lite (database).

The main theme of this project is to user make better money spend decisions with concur expense.

# **Chapter2. Scope of the Project**

## 2.1 Scope

Expenses Management System intended to support individual or business spending plan, track and potentially control your costs. It helps in tracing both user income and expenses. The Expenses Management System gives a coordinated arrangement to cash flow and expenses of user. It gives the capacity to gather your income into classifications and gives you a chance to set a financial plan and track costs in the classification.

## 2.2 Limitations

Individual probably will not know or recollect where they spent their pay consequently the following of the cost will be problematic. Pay of individual probably will not be customary thus, their computation may not be right. The user spending may be more than that of their income. This application will not run on IOS because this application is only for android platform.

## 2.3 Aims

* The main aim of this project is to develop application where a user can add, update and track income and expenses and to control the excessive money spending of user.
* Helps to calculate spending amount of users and to produce expenses reports of users.

## 2.4 Objectives

* To perform user based design.
* To develop user-friendly application.
* To manage time according to the user needs.
* To get better software with less bugs, the developed application should tested properly.
* To design different types of diagrams like use-case diagram, activity diagram, sequence diagram etc.
* To illustrate the flow of application with the help of diagrams.
* To perform this project as both individual and business purposes.
* To analyze the views of users regarding this project.
* For further support all the system development, development methodology, diagrams should documented properly.

## 2.5 Overview of the Scope

Overview of the scope include limitations, aims and objectives of the Expenses Management System. Limitation of the project describes that the user expenses might not managed properly because individual could not know where they spend their income and for what purposes and the spending of user could more than that of income. The main limitation of this project that it is only for android user not for IOS user. In this way, aim of this project is to calculate expenses report by calculating spending money. Finally, objectives describes how to get better software and the documentation of design pattern, development methodology used, performance of the Expenses Management System for further support.

# **Chapter3. Development Methodology**

## 3.1 Description of the Methodology

**Waterfall Model**

The waterfall model was first process model and it is very simple to understand and use. In a Waterfall model, each phase must completed before the next phase can begin and there is no overlapping on the phase. It is the earliest SDLC approach that used for software development. (Tussen de Vaarten, 2013-2018)

Phases of Waterfall model includes following points:

* Requirements analysis
* Design
* Implementation
* Verification
* Maintenance



Screenshot 1: Waterfall Model

I have chosen Waterfall Methodology due to following reason:

* This methodology is easy to understand and use.
* This methodology is mainly for small project.

Advantage of using Waterfall Model:

* Phases are proceed and completed one at a time. Therefore, that phase do not overlap.
* Clear estimation of cost according to the requirements.
* In this model, process of testing is more efficient and simpler.

Disadvantage of using Waterfall Model:

* Does not suitable for randomly change requirements.
* Going back to a phase is more difficult and costly efficient.
* Testing time comes very later. This is against the rule of software development.

## 3.2 Design Pattern

I will used **Model View Controller (MVC)** design pattern.

**Model:** Within the pattern, the model is the main components and the main purpose of the model is to manage the data, logics and rules of the application.

**View:** Output representation of the data in the form of a screen or user interface (UX).

**Controller:** Both model and view updates by controller. It control the data flow into model and updates the view if data changes. (Gootooru, 2019)



Screenshot 2: MVC design pattern

**Advantages of using MVC design pattern:**

* Rapid and parallel development process i.e. faster development process.
* Ability to create multiple views for a model.
* Supports TDD (Test Driven Development).
* The entire model does not affected by modification because both model part and view part are different.

**Disadvantages of using MVC design pattern:**

* Complexity will increase.
* Multiple technologies knowledge is required.
* Multiple number of programmers needed.
* In view, there is inefficiency of data access.

## 3.3 System Architecture

**Three Tier Architecture**

The three-tier architecture consists of three tier. They are:

* Presentation Tier
* Application Tier
* Data Tier

Presentation Tier:

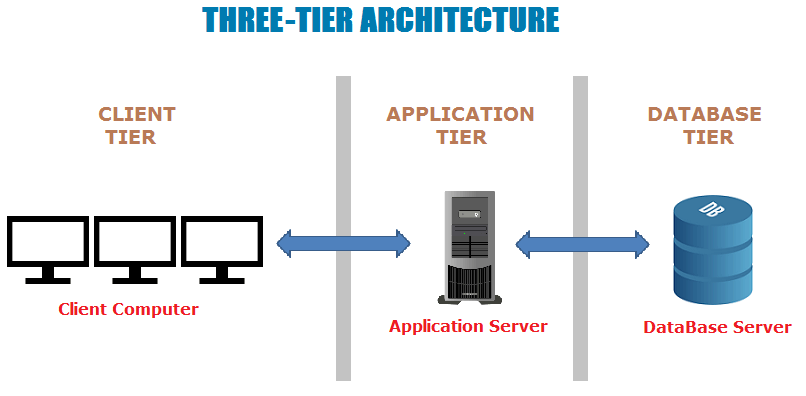
In the three-tier system, presentation tier represents the front-end layer and consists of the user interface. This tier build on web based technologies or other popular web development frameworks and communicates with others layers through API calls.

Application Tier:

The application tier consists functional business logic and it has often written in java and other programming language.

Data Tier:

The data tier consists database, data storage system and data access layer. Data accessed with the help of application layer via API calls. Examples of data storage system includes MYSQL, Oracle, and SQL Lite etc.

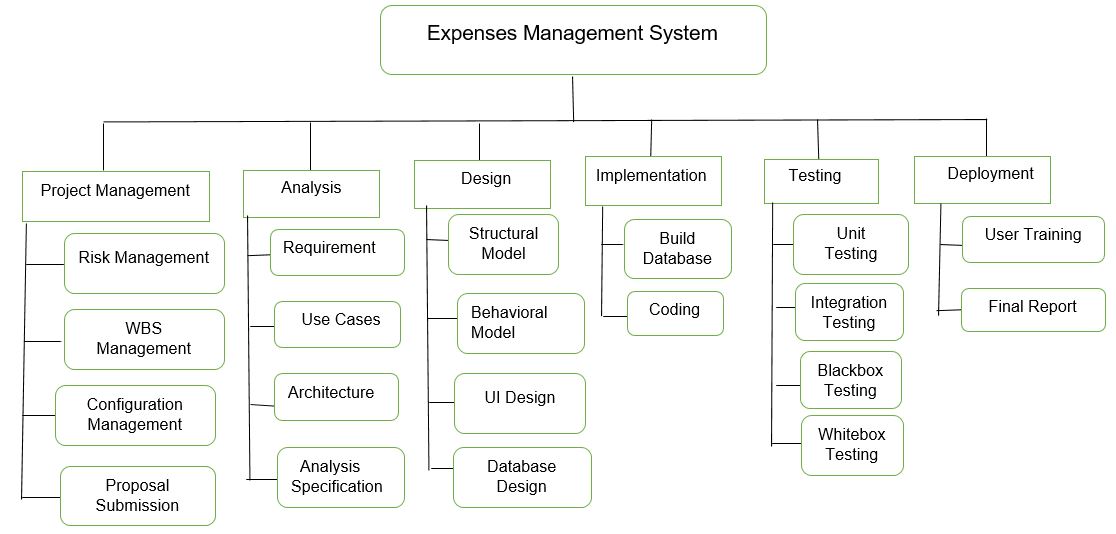
****

Screenshot 3: Three-tier Architecture

# **Chapter4. Project Planning**

## 4.1 Work Breakdown Structure (WBS)

To remove complexity and for manageable purpose a project is breakdown into smaller components called Work Breakdown Structure (WBS). It provides a hierarchical and incremental decomposition of a project into phases, deliverables and work packages. It breakdown the entire project into meaningful components. The Work Breakdown Structure for Expenses Management System are as given below:

* Project Management
* Analysis
* Design
* Implementation
* Deployment

Screenshot 4: Work Breakdown Structure for Expenses Management System

## 4.2 Milestones

**Milestones Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.N** | **Milestones** | **Start Date** | **End Date** | **Total Days (115)** |
| 1. | **Project Management**  Risk Management  WBS  Configuration Management  Proposal Submission | **03/25/2019**  03/25/2019  03/29/2019  04/02/2019  04/05/2019 | **04/07/2019**  03/28/2019  04/01/2019  04/04/2019  04/07/2019 | **14**  4  4  3  3 |
| 2 | **Analysis**  Requirement analysis  Use Case  Architecture  Analysis specification | **04/08/2019**  04/08/2019  04/16/2019  04/23/2019  05/02/2019 | **05/06/2019**  04/15/2019  04/22/2019  05/01/2019  05/06/2019 | **29**  8  7  9  5 |
| 3 | **Design**  Structural Diagram  Behavioral Diagram  UI Design  Database Design (ER, Data Dictionary) | **05/07/2019**  05/07/2019  05/12/2019  05/19/2019  05/26/2019 | **06/01/2019**  05/11/2019  05/18/2019  05/25/2019  06/01/2019 | **26**  5  7  7  7 |
| 4 | **Implementation**  Building Database  Coding | **06/02/2019**  06/02/2019  06/07/2019 | **06/21/2019**  06/06/2019  06/21/2019 | **20**  5  15 |
| 5 | **Testing**  Unit Testing  Integration Testing  Blackbox Testing  Whitebox Testing | **06/22/2019**  06/22/2019  06/26/2019  06/30/2019  07/04/2019 | **07/06/2019**  06/25/2019  07/29/2019  07/03/2019  07/06/2019 | **15**  4  4  4  3 |
| 6 | **Deployment**  User Training  Final Report | **07/07/2019**  07/07/2019  07/10/2019 | **07/12/2019**  07/09/2019  07/12/2019 | **6**  3  3 |

To complete my project I have evaluated total 110 days. I have a short period to complete this project .So to build a project there is a combination of different phases or milestones. Each phases or milestones have their own duration.

* To complete project management phase I have evaluated total 14 days. Project management phase involves sub-phases i.e. risk management, work breakdown structure (WBS), configuration management and proposal submission. I have evaluated 4 days for risk management, 4 days for WBS, 3 days for configuration management and 3 days for proposal submission.
* To complete analysis phase I have evaluated total 29 days. Analysis phase involves sub-phases i.e. requirement analysis, use case, architecture and analysis specification. I have evaluated 8 days for requirement analysis, 7 days for use cases, 9 days for architecture and 5 days for analysis specification.
* To complete design phase I have evaluated total 26 days. Design phase involves sub-phases i.e. structural diagram, behavioral diagram, UI design and database design. I have evaluated 5 days for structural diagram, 7days for behavioral diagram, 7 days for UI design and 7 days for database design.
* To complete implementation phase I have evaluated total 20 days. Implementation phase involves sub-phases i.e. building database and design. I have evaluated 5 days for building database and 15 days for coding.
* To complete testing phase I have evaluated total 15 days. Testing phase involves other sub-phases i.e. Unit testing, Integration testing, Blackbox testing and Whitebox testing.

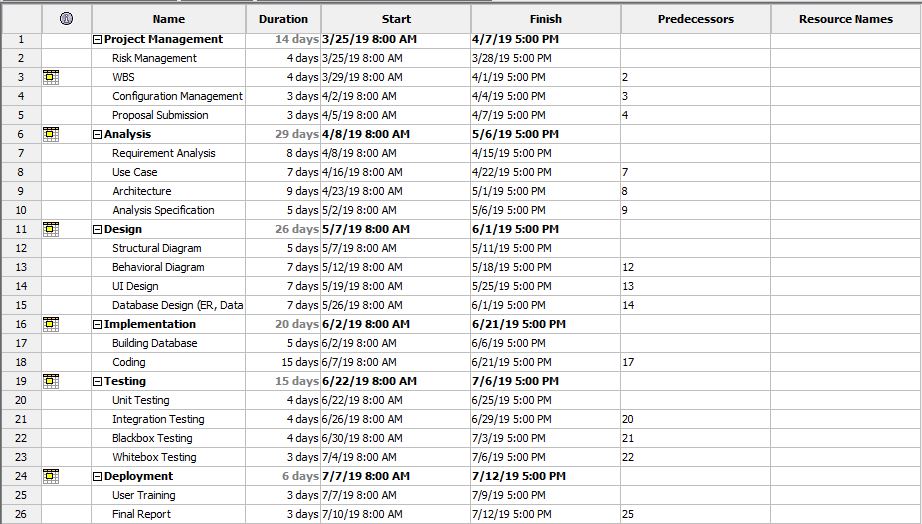
I have evaluated 4 days for unit testing, 4 days for integration testing, 4 days for Blackbox testing and 3 days for Whitebox testing.

* To complete deployment phase I have evaluated total 6 days. Deployment phase also involves sub-phases i.e. User training and Final report. I have evaluated 3 days for user training and 3 days for final report.

## 4.3 Gantt Chart

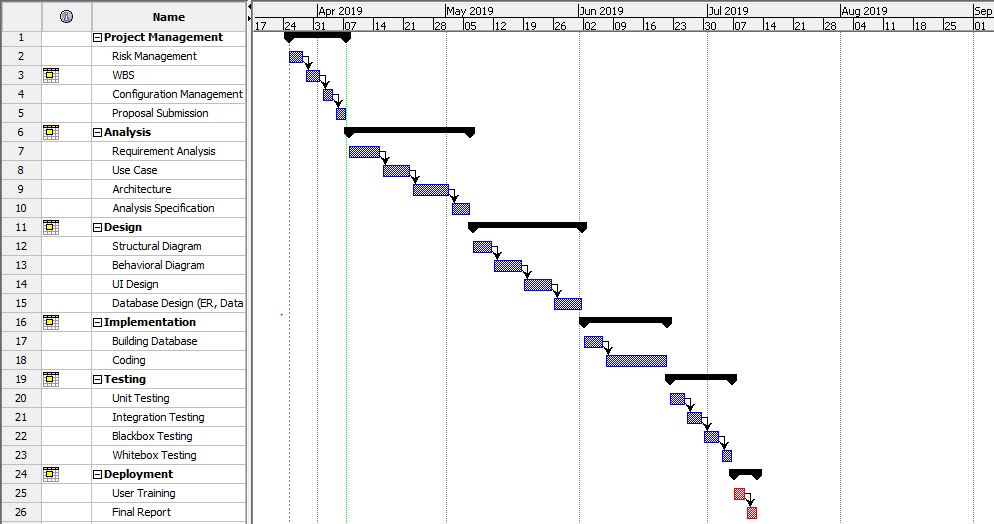
The schedule and the gantt chart of project are as given below:

Schedule:



Screenshot 5: Schedule of Expenses Management System

Gantt chart:



Screenshot 6: Gantt chart of Expenses Management System

# **Chapter5. Risk Management**

Risk management is the technique of identifying, assessing and controlling threats to a business using insurance, safety measures etc. The main purpose of risk management is to control or minimize the flow of unexpected risk, extra time and money. To get impact of risk on project we have mathematical calculation i.e.

Impact = Likelihood \* Consequences

Likelihood values of risk are as given below:

|  |  |
| --- | --- |
| **Likelihood** | **Value** |
| Low | 1 |
| Medium | 2 |
| High | 3 |

Consequences values of risk are as given below:

|  |  |
| --- | --- |
| **Consequences** | **Value** |
| Very Low | 1 |
| Low | 2 |
| Medium | 3 |
| High | 4 |
| Very High | 5 |

## 

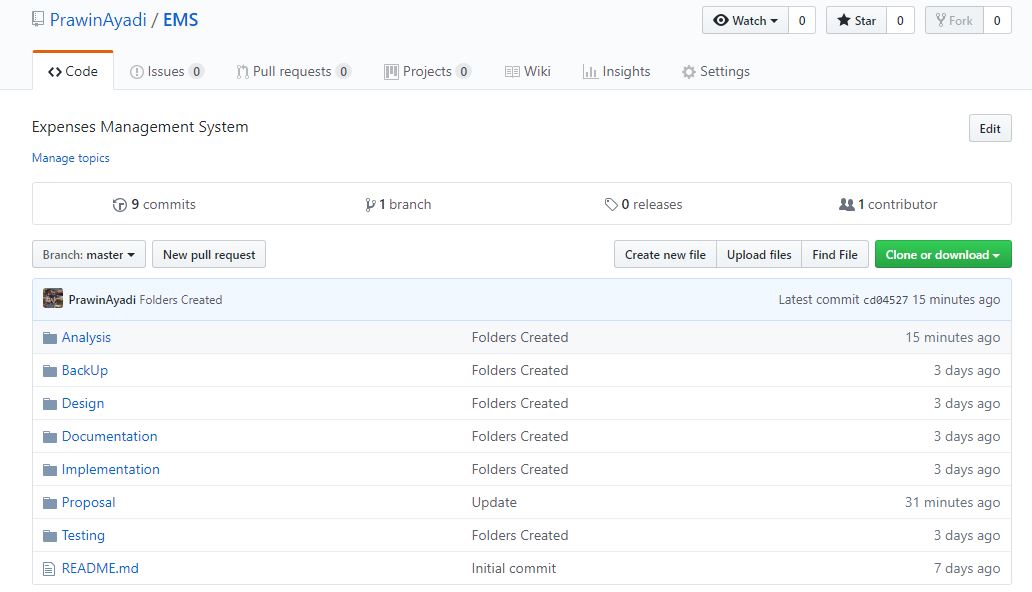
**Risk Table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.N** | **Risks** | **Likelihood** | **Consequences** | **Impact** | **Solution** |
| 1 | Lack of resources | 2 | 3 | 6 | Before to develop application all the resources needed for project should provide. |
| 2 | Operating System failure | 2 | 4 | 8 | Do not overuse computer and provide file backup system. |
| 3 | Problem in flow of electricity | 1 | 5 | 5 | Uninterrupted power supply (UPS) must provide. |
| 4 | Lack of knowledge in users | 3 | 3 | 9 | Provide training to the users before implementation. |
| 5 | Virus, Threats | 2 | 4 | 8 | Regular scheduled scans with the use of installed anti-virus software. |
| 6 | Natural Disaster like Earthquake, flood etc | 1 | 5 | 5 | Implementation of cloud back up system. |
| 7 | Hard drive failure | 1 | 4 | 4 | To prevent from failure reduce data load and provide back-up system. |

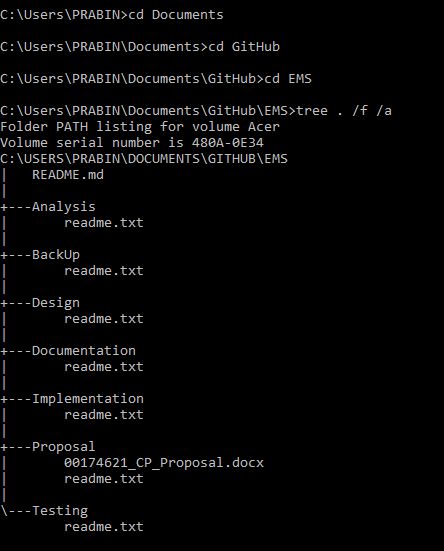
# **Chapter6. Configuration Management**

Configuration Management is the process for administrative activities concerned with the creation, maintenance, control change and quality control of the project. It helps to build the effectiveness, execution and dependability of the project. To effectively accessible from everywhere, files managed in sorted out structure.

Below information shows the directory structure of project:



Screenshot 7: Location for Git Push



Screenshot 8: Tree structure for Expenses Management System

# **Chapter7. Conclusion**

The main purpose of my project is to inspire the clients to lessen their spending. User can easily manage their expenses with the use of this application. To control or maintain the flow of expenses user can add their salary and daily expenses and it helps to manage the cash flow. This application is a valuable resource for cost saving, auditing and financial analysis. Error caused by a paper process and delay in reports reduced with the use of this application.

# **Chapter8. References**

Bennett, C., 2019. *The Economics Times.* [Online]   
Available at: https://economictimes.indiatimes.com/

Gootooru, N., 2019. *java2novice.* [Online]   
Available at: http://www.java2novice.com/java\_interview\_questions/mvc-pattern/

Tussen de Vaarten, A., 2013-2018. *TOOLSQA.COM.* [Online]   
Available at: https://www.toolsqa.com/software-testing/waterfall-model/