## 

|  |
| --- |
| Company logo  IMG-20170430-WA0018.jpg |
| Project Title |
| CS 2062 – Object Oriented Software Development |
|  |
| **Group members and roles** |
|  |

|  |
| --- |
|  |

**Project Report Template**

**This document provides you a template for the report and guidelines on the level of details required. You can change the template if necessary to match your needs and the structure of your project. However, all the required analysis and models should be included in the report.**

**Workload balance**

**Work done in project deliverables and contribution to the report should be equally distributed among team members. Each member will be evaluated based on his/her contribution.**

**Hint: You can (roughly) equally divide use cases. But make sure each member has at least one critical (essential) use case.**

**Font for paragraph text**

**Times New Roman 12**

**Single (1.0) spaced**

**1** **Introduction**

This chapter should be a group effort. Clearly state the contribution of individual team members.

## **1.1** **Client#####**

Introduce the client. State all the information of the client that is **related to the project**. Eg: Company structure, Locations, Staff Details.

## **1.2** **Problem Statement**

We have to develop tax calculation system for the sub divisional secretary office in Bemmulla.

The sub divisional secretary office has their income from assessment taxes and rental payments. All the taxes are collected according to the policies of the government. Administrative of the office can change their policies after the government confirm their rules changers.

The problem of that divisional secretary office is they keep all their records in books. So it takes more space to keep records and hard to find previous payment details about tax payers. And also they do all the calculation manually according to the rates, so it takes more time.

## **1.3** **Vision**

System Objectives

Business Benefits

System Capabilities

## **1.4** **System Scope**

**Clearly define the system scope. What requirements are included in the system and what are not.**

To obtain the business benefits listed, the system shall include the following capabilities.

1. Calculate the taxes from the system according to the tax rates.

2. Provide the weekly, monthly reports.

3. All the taxpayer's’ records are in the database, so can get taxpayer’s previous payments report.

4. Administrator can control taxpayer's’ accounts easily.

5. Taxpayer can get computerized bill after the payments.

6. Taxpayer will inform about due date of their payments.

7. Taxpayer will inform when they didn’t pay taxes in correct time.

# **2** **Inception Phase**

## **2.1** **Introduction######**

Clearly state individual contributions.

## **2.2** **Stakeholder Analysis**

**Identify stakeholders and their category, role, product focus, schedule focus and cost focus .**

## **2.3** **Risk Identification and Feasibility analysis**

**Identify any possible risks and analyze the feasibility of the project.**

## **2.4** **Requirement analysis**

### **2.4.1** **Requirement gathering methods**

Requirement gathering is essential for make system useful and user friendly. So that we put much effort on gathering requirements. We used following methods to gather accurate and very essential requirements.

1. Interviewing

2. Analyzing documents.

3. Observing present system.

We were able to interview all the people who are going to make use of our system and gather information regarding the problems they face with current system, the things that were needed make their work easy and accurate and also about the technical knowledge of the officers.

Now they have a manual system of keeping books and we were able to take note regarding the pros and cons of current system. Also we were able to take some copies of the policies of taxation and their accounting books.

### **2.4.2** **Essential Use Case list**

* Add a new taxpayer.
* Remove an existing tax payer account.
* Update basic policies.
* Request Annual/ Monthly report.
* Make tax/rental payment.
* View tax payer report.

### **2.4.3** **Non Functional requirements**

**Security Requirements**

Since all the accounting data and other useful information are stored in the local database as well remote database, it is essential that to only administrator of the system can make changes in the database data. Also user account access should be secure.

**Safety Requirements**

Local database can be damaged or crashed due to failure in the system, so it is necessary to keep backups of local database in a remote database.

**Performance Requirements**

The system should be accurate and fast. Exception handling must be done so that errors should be handled without making any loss of information. Also system should be able to handle large database with the growth of the system, without making any delays.

# **3** **Elaboration**

## **3.1** **Introduction**

Clearly state individual contributions.

## **3.2** **Domain Modeling**

### **3.2.1** **Event table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Event** | **Trigger** | **Source** | **Usecase** | **Response** | **Destination** |
| Admin wants to add a new tax/rental payer | Add new tax/rental payer | Admin/officer | Add a new tax/rental payer | New account | Admin |
| Admin wants to remove an existing tax/renal payer account | Remove tax/rental payer account | Admin | Remove existing tax/rental payer | State changes of the relevant taxpayer account |  |
| Admin wants to update basic policies | Update basic policies | Admin | Update basic policies | Update basic policies | Admin |
| Admin/Officer wants to view the basic policies | Request basic policies | Admin/officer | View basic policies | Display basic policies | Admin/Officer |
| Admin/officer wants to view the tax/rental payer details | Request  tax/rental  payer details | Admin/officer | View tax/rental payer details | Display the basic taxpayer details | Admin/Officer |
| Admin wants a whole report (Annual/Monthly) | Request payment details | Admin/officer | Get report (Annual/Monthly) | Display the report | Officer |
| False payments | Update  payments | Admin Admin | Cancel the payments | Cancel the payment | - |
| Tax/rental payer wants to pay the rental or tax | Update payments | Officer | Make tax/rental payments | Updated details | tax/rental payer |
| Tax/rental payer wants a report about previous payments | Request tax/rental payer details | Tax/rental payer | View tax payer report | Display report | tax/rental payer |
| System send notifications to the Admin/officer if tax/rental payer didn't pay the payment | Request tax/rental payer details | - | Send notifications to the Admin/Officer | Display notifications | - |
| System send notifications to tax/rental payer to inform about due date | Request tax payer details | - | Send notifications to the tax/rental payer | Display notifications | - |

**Event based approach**

### **3.2.2** **Use case diagram**

Full use case diagram for the system

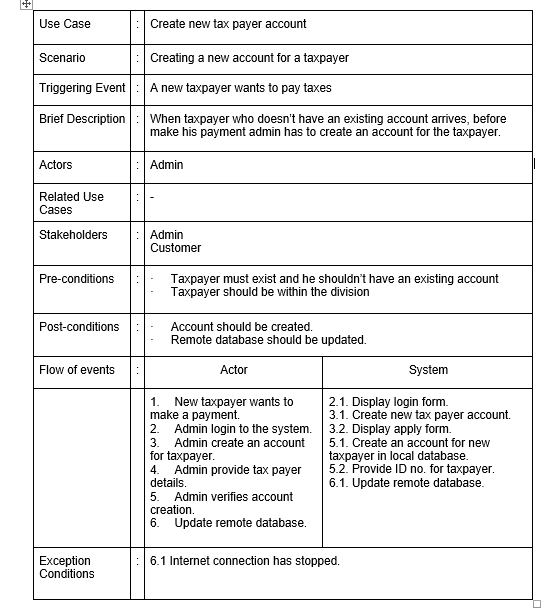
### 

### 

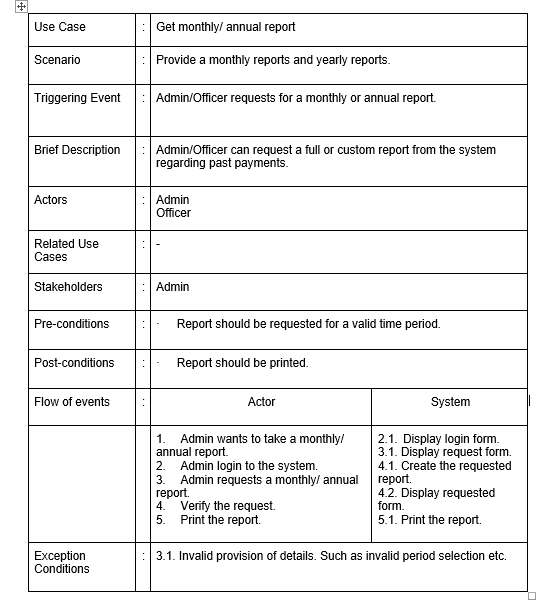
### 

### 

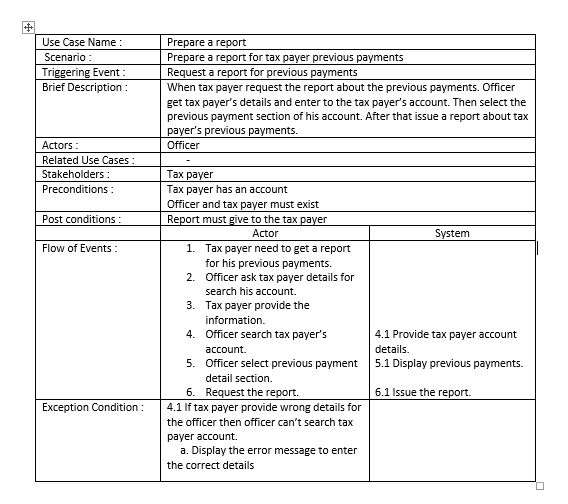
### **3.2.3** **Use case detailed descriptions**



Use Case Detailed Description for Add new taxpayer use case



Use Case Detailed Description for get monthly/annual report use case



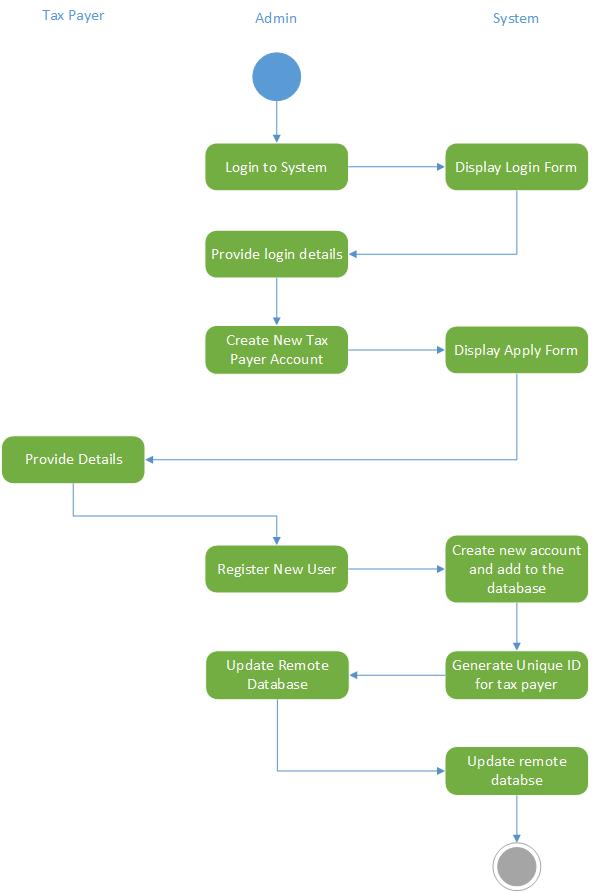
Use Case Detailed Description for get taxpayer's previous payment detail report use case

**Provide use case detailed descriptions *at least* for the essential use case list identified for 2.6.2.**

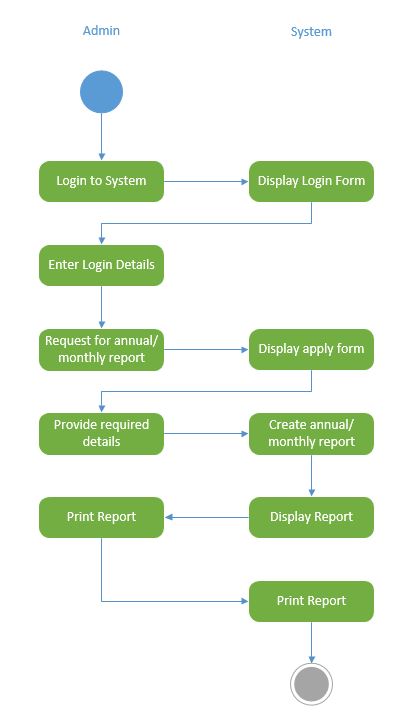
**Each member of the group should prepare at least one essential use case description.**

### 

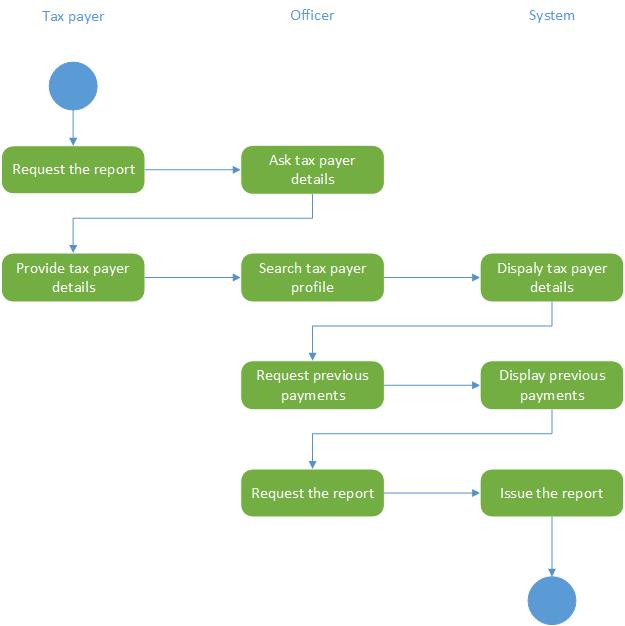
### **3.2.4** **Activity Diagrams**



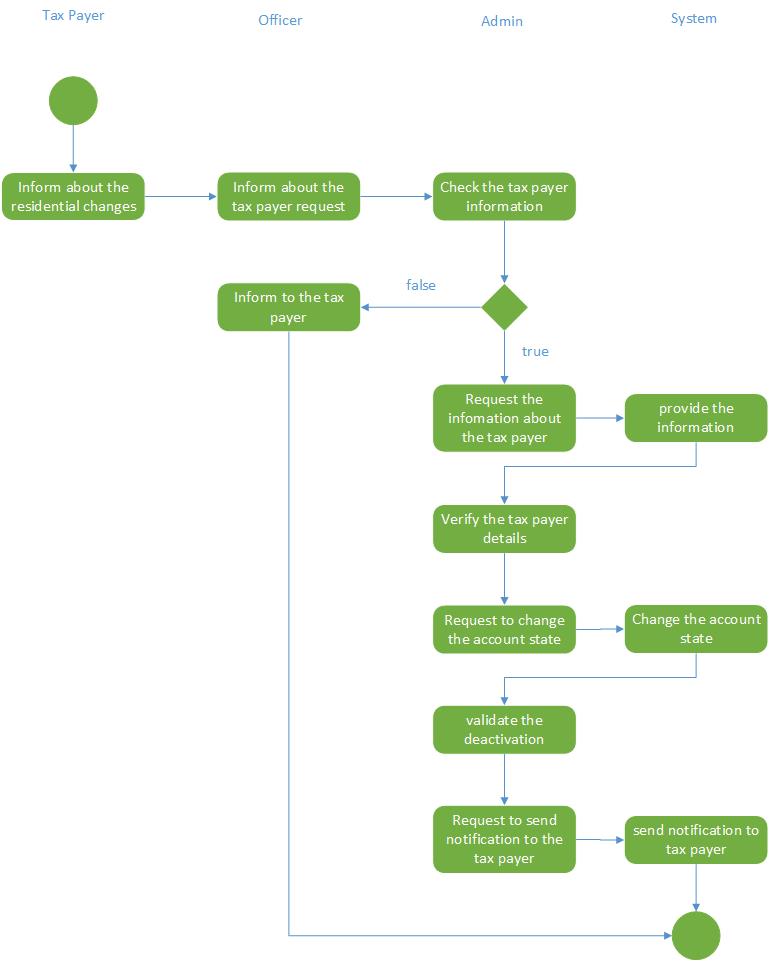
Activity Diagram for Add new taxpayer use case



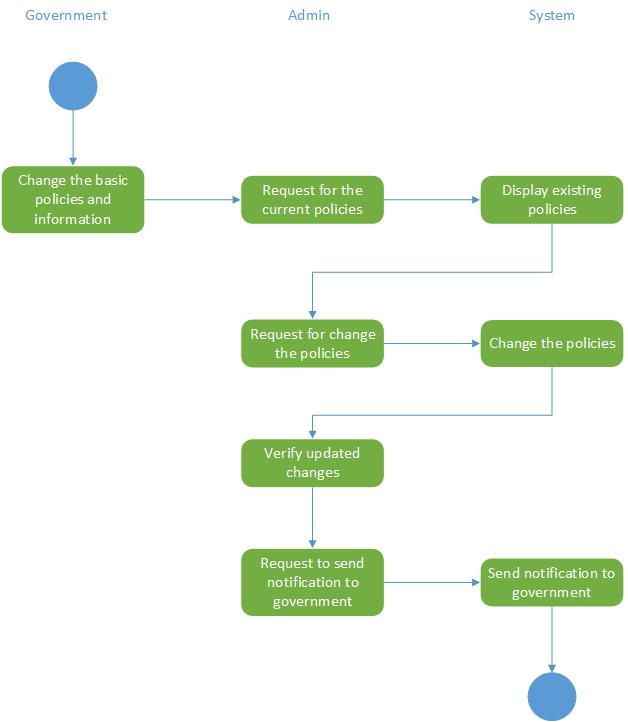
Activity Diagram for get annual/monthly report use case



Activity Diagram for get taxpayer's previous payment detail report use case



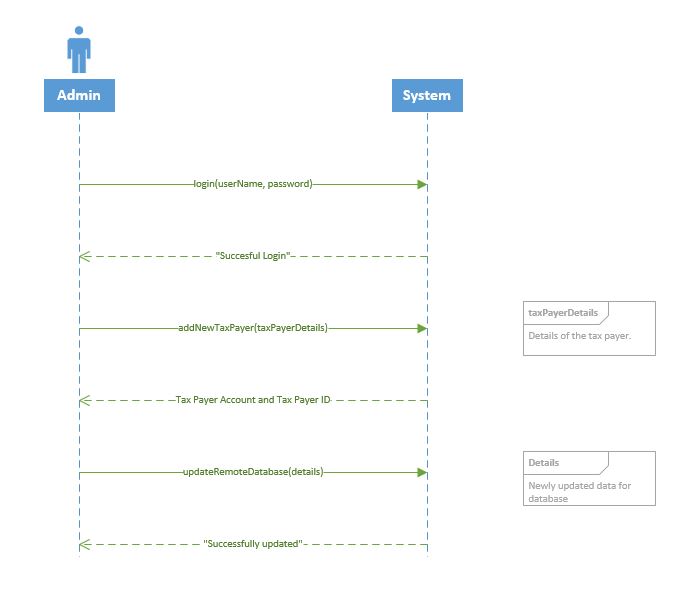
State Diagram for change the state of an existing taxpayer account use case



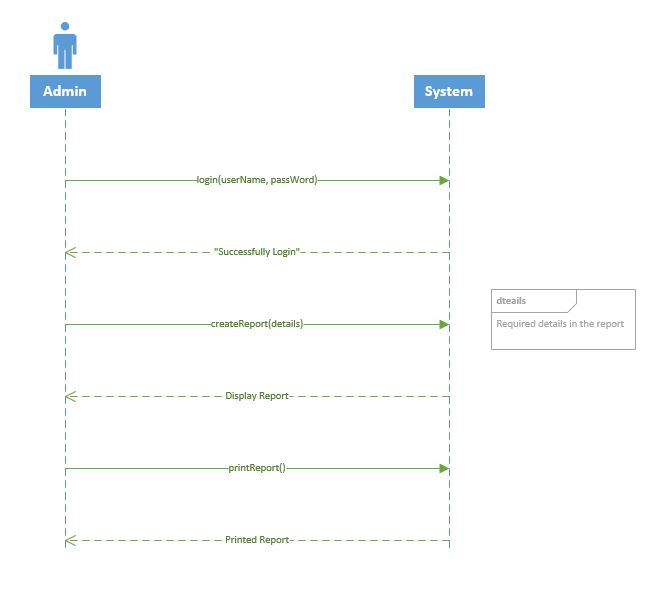
State Diagram for update the basic policies use case

**Activity diagrams *at least* for the essential use case list identified for 2.6.2.**

### **3.2.5** **System Sequence Diagrams**



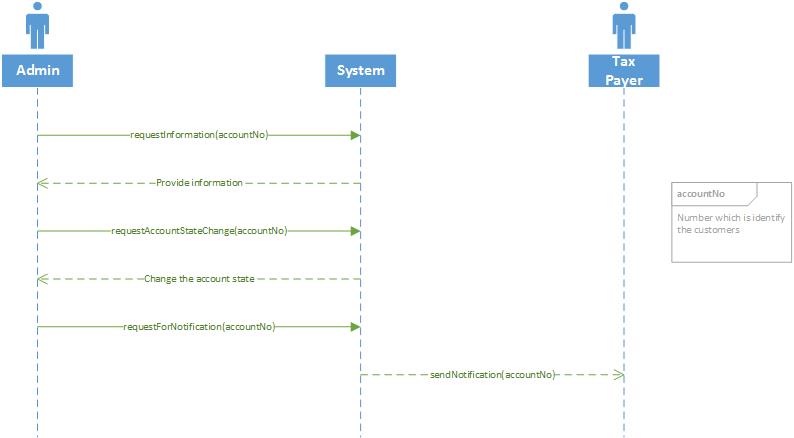
System Sequence Diagram for add new taxpayer use case



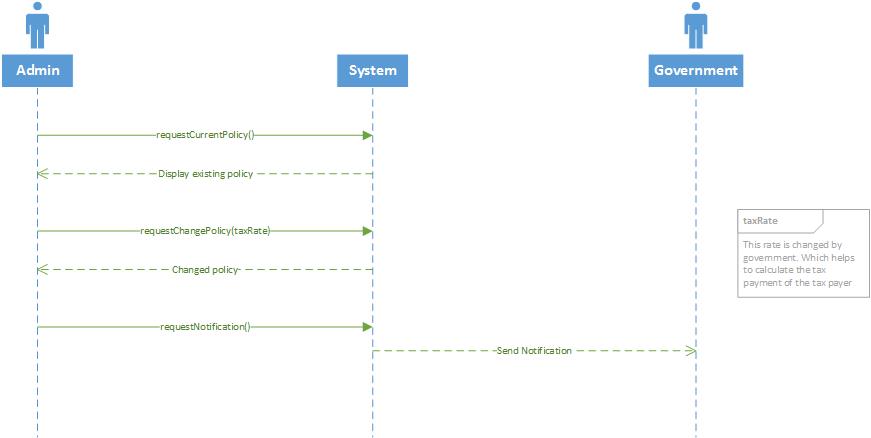
System Sequence Diagram for get annual/ monthly report use case



System Sequence Diagram for get taxpayer's previous payment detail report use case



System Sequence Diagram for change the state of an existing taxpayer account use case



System Sequence Diagram for update the basic policies use case

**SSD s *at least* for the essential use case list identified for 2.6.2.**

**Entity based approach**

### **3.2.6** **Domain class diagram**

Domain class diagram for the full system

### **3.2.7** **State chart diagrams**

Identify the objects that have multiple states during their lifecycle, and draw the state chart diagrams for those objects.

# **4** **Conclusion**

## **4.1** **Tools Used####**

State any tools you have used and their purpose.

## **4.2** **Project Status####**

Current state of the project and what is planned next.

## **4.3** **Problem encountered**

**LIST** all the problems you encountered during the project so far, and **briefly** describe how you overcame those.

# 