



# **Tri Lingual Cabinet Paper Processing System (CPPS)**

**A dissertation submitted for the Degree of Master of  
Information Technology**

**J.G.Bandula Chandranath Kularatne**

**University of Colombo School of Computing**

**2020**



## **Abstract**

The main objective of this project is to develop an information system to automate the workflow of Memorandum Branch of the Office of the Cabinet of Ministers in Sri Lanka. Pre-Processing, queuing and preparation of Cabinet Memorandums for upcoming cabinet meetings have become a challenge to the staff. Similarly, there is a formal process that must be met before a Cabinet Paper is placed on agenda of the Cabinet meeting and that process must assure the accuracy, efficiency and punctuality. In case of the memorandums are related to Establishment or Tender board matters, those should be discussed at the sub-committee meetings prior to the main meeting and the decisions and recommendations of the sub-committee meetings are must be forwarded to the main meeting. This process takes more time and situation becomes more exacerbated when a large number of Cabinet memorandums are received in a day. The bulk of this work is done in the Memorandum Branch. Currently, some of the tasks are performed by using computer software, Microsoft Word and Excel and the rest is carried out as a manual process. Those applications have very limited and primary features. Because of the poor and time consuming partially-automated existing system, the officers who involve in this job, face many difficulties in performing those interconnected tasks on time.

By introducing the new Cabinet Paper Processing System (CPPS), it is intended to improve the overall accuracy, speed, productivity and quality of work carried out by the Office of the Cabinet of Ministers. The New System will address the said limitations and drawbacks of the existing semi-automated system. Further, the new system will offer new features such as generating supportive letters, one touch information access facility, progress monitoring and rich reporting facility in accurate and timely manner.

The Cabinet Paper Processing System (CPPS) is a LAN based automation system that can be accessed through the existing Local Area Network of the office. The system has adapted Layard (n-tier) architecture and OOP techniques. UML is used for the analysis and design. Visual Basic.Net language with .NET framework 4.5 and visual studio 2019 is used to develop the system and Microsoft Reports Viewer is used for reporting purposes. MySQL Server 5.00 is used to handle database, queries, stored procedures etc. System operates on existing LAN which already has servers, switches, routers and other network devices. Server is running on Windows Server 2013 R2 and clients operate on Windows 7, 8 or Windows 10.

User evaluation and feedback shows that the new CPPS information system is effectively developed and implemented. The CPPS Information System meets the requirements of the Cabinet Office.

## Declaration

This thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Student Name : J.G.Bandula Chandranath Kularatne,

Registration Number : 2016/MIT/010

Index Number : 16550108

.....

Signature

.....

Date:

This is to certify that this thesis is based on the work of Mr. J.G. Bandula Chandranath Kularatne under my supervision, and the thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by:

Supervisor Name:

.....

Signature

.....

Date:

## **Acknowledgement**

Success of a project is not only an effort of the author himself, but also the support, encouragement and help of many others. It was my great pleasure to conduct and implement this project work and complete the dissertation documentation on “3201 Individual Project” successfully.

I'm highly grateful to the Office of the Cabinet of Ministers, especially Mr. Donald Fernando Senior Additional Secretary and the Former Secretary to the Cabinet Mr. Sumith Abeysinghe who gave enormous support and guidance on behalf of the client of this project. I would like to thank Mrs. Sriya Jayawardane, the Progress Assistant (Head of the Memorandum Branch), who helped me to gather information on requirements properly.

I would like to extend my sincere thanks to my supervisor Professor G.K.A.Dias, UCSC, for his valuable time and support extended towards me to fulfill this academic assignment. Also I honor his immense contribution throughout the process of study. I highly appreciate his contribution throughout the documentation process including the project proposal, methodology, literature and on writing this report. I am grateful to him very much for his extensive assistance.

Finally I would like to thank the University of Colombo School of Computing (UCSC) and coordinators of the Post Graduate Division (MIT) degree program for giving me this opportunity to apply the knowledge which I have gained through the Master of Information Technology Study program.

J.G.Bandula Chandranath Kularatne

MIT 2017

Index No: 16550108

UCSC

# Table of Contents

Abstract.....	ii
Declaration.....	iii
Acknowledgement.....	iv
Table of Contents.....	v
List of Figures.....	viii
List of Tables.....	x
Definitions, Acronyms and Abbreviations.....	xi
Chapter 1 Introduction.....	1
1.1 Introduction.....	1
1.2 Problem.....	1
1.3 Motivation.....	4
1.4 Project Objective(s).....	4
1.5 Scope of the Project.....	6
1.6 Significance of the study.....	6
1.7 Dissertation Overview.....	7
Chapter 2 Background.....	8
2.1 Chapter Introduction.....	8
2.2 Project Background.....	8
2.3 Analysis of Similar Systems.....	9
2.3.1 Comparison Table (Similar Systems).....	10
2.4 System Development Methodology.....	10
2.5 Comparative Study of Development Methodologies.....	11
2.5.1 Comparison of Alternative Design Methodologies.....	12
2.6 Selected Development Methodology.....	13
2.7 Software Architecture Patterns.....	13
2.7.1 Layard (n-tier) Architecture.....	13
2.7.2 MVC Architecture.....	13

2.7.3 Event-Driven Architecture.....	14
Chapter 3 Analysis and Design.....	15
3.1 Chapter Introduction .....	15
3.2 Problem Identification .....	15
3.3 Understanding the Existing System .....	15
3.4 Existing System .....	16
3.5 DFD – Existing System .....	17
3.6 Information Gathering .....	17
3.7 Drawbacks of Existing System .....	20
3.8 Proposed System.....	21
3.9 Functional Requirements .....	21
3.10 Non-Functional Requirements .....	21
3.11 Business Modelling.....	22
3.12 Use case Diagrams and Descriptions .....	22
3.13 Use Case Diagram – Proposed System.....	23
3.14 Use Case Descriptions .....	24
3.15 System Design .....	25
3.16 System Structure Design.....	25
3.17 System Architecture design .....	25
3.18 Data Model Design (ER Diagram) .....	26
3.19 Detailed Design.....	28
3.20 Activity Diagrams .....	28
3.21 Login activity diagram.....	29
3.21.1 Wireframe Designs .....	29
Chapter 4 Implementation.....	30
4.1 Introduction.....	30
4.2 System and Software Design .....	30
4.3 Design Quality .....	31

4.3.1 Design Decisions .....	31
4.4 Development Environment .....	33
4.4.1 Software & Hardware Used .....	33
4.4.2 Re-usable Components .....	34
4.5 Project Folder Structure .....	35
4.6 Implementation Methodology.....	36
4.7 User Training .....	37
Chapter 5 Testing & Evaluation .....	38
5.1 Introduction.....	38
5.2 Testing method.....	38
5.2.1 Login Test Case .....	40
5.3 Feedback and Evaluation .....	42
5.4 Analysis of Feedback.....	42
Chapter 6 Conclusion & Further Work.....	44
6.1 Introduction.....	44
6.2 Problems Encountered .....	45
6.3 Learning Outcomes .....	45
6.4 Further Enhancements .....	46
<b>Appendix A</b> System Documentation.....	47
<b>Appendix B</b> Use Case Diagrams and Descriptions.....	48
<b>Appendix C</b> Design Documentation .....	58
<b>Appendix D</b> Test Cases.....	65
<b>Appendix E</b> User Documentation .....	73
<b>Appendix F</b> Sample Reports .....	84
<b>Appendix G</b> Code Listing .....	86
References.....	90

## List of Figures

Figure 3.1 DFD Existing System.....	17
Figure 3.2 Use Case Diagram for Proposed System.....	23
Figure 3.4 Structural Design .....	25
Figure 3.5 System Architecture Design .....	<b>Error! Bookmark not defined.</b>
Figure 3.6 ER Diagram .....	27
Figure 3.7 Activity Diagram Notations .....	<b>Error! Bookmark not defined.</b>
Figure 3.8 Activity Diagram - Login .....	<b>Error! Bookmark not defined.</b>
Figure 3.9 Login Wireframe .....	29
Figure 4.2 Phase wise Implementation Plan .....	36
Figure 5.1 Feedback Form .....	<b>Error! Bookmark not defined.</b>
Figure B.2 CP Registration Use Case Diagram .....	48
Figure B.3 Observation Preparation Use Case .....	49
Figure B.4 Sent Reminder Use Case.....	49
Figure B.5 Agenda Preparation.....	50
Figure B.6 Create Minutes .....	50
Figure B.7 Feed Decisions Use Case .....	51
Figure B.8 Manage Users .....	51
Figure B.9 CP Monitoring Use Case .....	52
Figure B.10 Reports Generation .....	52
Figure C.1 Activity Diagram - CP Registration.....	58
Figure C.2 Activity Diagram - Observations .....	58
Figure C.3 Activity Diagram - Generate Reminder.....	59
Figure C.4 Activity Diagram - Agenda Preparation .....	59
Figure C.5 Activity Diagram - Draft Minute Generate.....	60
Figure C.6 Wireframe - Main Form.....	61
Figure C.7 Wireframe - Cabinet Papers.....	61
Figure C.8 Wireframe - Add Cabinet Papers .....	62
Figure C.9 Wireframe - Observation .....	62
Figure C.10 Wireframe – Agenda/Sub Committee Agenda .....	63
Figure C.11 Wireframe - Supplementary Agenda .....	63
Figure C.12 Wireframe - Manage User .....	64
Figure C.13 Wireframe - CP Search .....	64
Figure E.1 CPPS .....	73



Figure E.2 Login Screen .....	73
Figure E.3 Main Dashboard.....	75
Figure E.4 Toolbar .....	75
Figure E.5 Language Selection .....	76
Figure E.6 Add Ministry .....	76
Figure E.7 Updating a Ministry .....	77
Figure E.9 Add new Cabinet Paper.....	79
Figure E.10 Update a Cabinet Paper.....	79
Figure E.11 Observation Main Module .....	79
Figure E.12 New Observation.....	80
Figure E.13 Update an Observation.....	80
Figure E.14 Creating a new Agenda.....	81
Figure E.15 Updating an Agenda.....	81
Figure E.16 Create Reminder .....	82
Figure E.17 Report Parameters .....	82

## List of Tables

Table 2.1 Similar System Comparison .....	10
Table 2.2 SDLC Methodologies Comparison.....	12
Table 3.1 Summary of Interviews preparation .....	19
Table 5.1 Test Case Considerations.....	39
Table 5.2 Login Test Case .....	41
Table B.1 Use Case Descriptions.....	53
Table B.2 Login Use Case Description.....	54
Table B.3 Logout Use Case Description.....	54
Table B.4 CP Registration Use Case Description.....	55
Table B.5 Use Case Description Observation Preparation .....	55
Table B.6 Send Reminders Use Case Description.....	56
Table B.7 Agenda Preparation Use Case Description .....	56
Table B.8 CP Registration – Use Case Description.....	57
Table D.1 UI Test Scenario - Main Menu .....	65
Table D.2 Functional Test Scenario - Main Menu .....	65
Table D.3 Functional Test Scenario - Add CP .....	67
Table D.4 Functional Test Scenarios - Observation .....	68
Table D.5 Functional Test Scenarios - Update Observation.....	69
Table D.6 Functional Test Scenario - Agenda.....	71
Table D.7 Functional Test Scenario - CP Module.....	72
Table E.1 System Requirements .....	73

## **Definitions, Acronyms and Abbreviations**

CASE : Computer Aided Software Engineering  
COM : Component Object Model  
CP : Cabinet Paper  
CPPS : Cabinet Paper Processing System  
CSS : Cascaded Style Sheet  
DFD : Data Flow Diagram  
ERD : Entity Relationship Diagram  
GUI : Graphical User Interface  
HTML: Hyper Text Markup Language  
LAN : Local Area Network  
MVC : Model View Controller  
OOP : Object Oriented Programming  
OS : Operating System  
RAD : Rapid Application Development  
RDBMS: Relational Database Management System  
SDLC : System Development Life Cycle  
SRS : Software Requirement Specification.  
UAT : User Acceptance Testing  
UI : User Interface  
UML : Unified Modeling Language  
UTF : Unicode Transformation Format  
WYSIWYG: What You See Is What You Get

# **Chapter 1 Introduction**

## **1.1 Introduction**

This chapter is the introductory chapter, gives an overview of this thesis and this study. The chapter begins by summarizing the problems with which this study was conducted. Then, the chapter proceeds to explain motivation and objectives of the study. The scope of the project of this study is then briefly outlined. Finally, an overview of how the study was conducted is provided.

## **1.2 Problem**

The Cabinet Office is the main decision making body of the Democratic Socialist Republic of Sri Lanka. This is the meeting place where ministers discuss government policies, public interests and controversial issues. It is established by the Constitution. Most of the important decisions or actions taken by Parliament are first discussed and collectively agreed upon by the Cabinet. The Cabinet decides its procedures. The Cabinet consists of Ministers. Head of the Cabinet is His Excellency the President. Secretary to the Cabinet is the head of the Establishment. Ministers outside the Cabinet (non-Cabinet Ministers) may participate to the Cabinet meetings only upon the invitation of the Cabinet of Ministers.

### **Sub Committees**

The Cabinet has sub-committees headed by designated Cabinet Ministers. Other members of the Cabinet become members of Committees. The Cabinet Sub-Committees will discuss relevant specific issues and make recommendations before the Cabinet meeting. Cabinet Sub-Committees are generally set up on specific subjects, such as Establishment matters, tendering matters. All Cabinet Sub-Committee decisions must be reported to the Cabinet of Ministers.

The Cabinet Office is the institution which provides the infrastructure for the Cabinet Meetings and organizes the Cabinet Meetings and distributes the Cabinet Decisions to the relevant Ministries and Institutions. There are 4 branches as Administration, Accounts, Memorandum and Transport. The Memorandum Branch is the main body responsible for matters such as receiving Cabinet Memorandums, maintaining a Registry, classifying them according to content, preparing Observation Letters, sending out reminders, preparation of Agendas for Cabinet / Subcommittee Meetings, drafting Cabinet Minutes.

Therefore, they have a big, responsible role in word processing. All the works done by the Memorandum Branch to perform the above functions are repetitive word processing tasks and must do on time and accurately.

**Key words:** Minute File, Cabinet Office, Memorandum Branch, Agenda, Observation

1. All the confidential documents relating to Cabinet Memorandums are sent to honorable Ministers and relevant top government officials. Therefore those documents must be prepared with utmost care, on time and without a single grammatical or spelling error. This is a very responsible and critical task for the officers of the Memorandum Branch.

For example, an officer spends approximately twenty minutes to prepare one observation letter. Accordingly, this process takes much time. Therefore, three officers have been assigned for preparation of observation letters. In certain instances the Cabinet Office has been unable to send the observation letters as expected, within the stipulated time period.

2. Currently, only the English version of the agenda for the next Cabinet / Subcommittee meeting is prepared by the Memorandum Branch. Sinhala and Tamil versions are prepared by the word processing staff in coordination with the Memorandum Branch Officers. Therefore, the proposed automated information system will also facilitate the function of the word processing staff.

3. The Memorandum Branch prepares draft cabinet minutes (hereafter named as Minute file) based on the agenda of the meeting to be held. This is very important work being done by the memorandum branch to save time.

For example, suppose there are 100 memorandums to be discussed in a meeting. In that case, the minute file approximately becomes 100 pages long; if we imagine the length of a single decision is one page (this is variable). This is purely a word processing task done with the assistance of translators. Hence, some of the word processing features such as copy / paste, macro, automatic replace, spell checker and tables are being used to create this minute file. Since this process includes lots of data entry work, the likelihood of inserting a wrong entry is very high. Therefore the final minute file preparation process requires a lot of proof reading for error correction.

Cabinet meetings are held on Wednesdays. The final minute file should be completed by Thursday evening in order to distribute decisions to Hon Ministers on Friday morning along with the agenda of the meeting to be held in the next week. A typical minute file contains 150-250 pages roughly. It is a very responsible task to prepare such a crucial and lengthy document accurately within two days. Officers must work on both Wednesday and Thursday even in nights to finalize it. Currently, 100% of the word processing task is done manually.

\* When consider a decision, 60% of a decision can be dynamically generated (Eg: serial number, signed date, title of the memorandum and the name of the submitted Minister and the Ministry, action taken and copied to columns etc). In other words, 60% of the work load of preparation of minute file can be automated. The balance 40% contains specific things discussed about the matter at the meeting and that should be entered manually.

4. In the existing system there is no facility to know a particular Cabinet Memorandum is included in an agenda or not. The only method of searching for such information is checking in the Cabinet Papers (CP) registry manually. And also, there is no a quick way of checking whether the observations of honorable Ministers have been received to the office or not, its date and the content. Searching for this kind of information and making administrative decisions on such information is time consuming and difficult process.
5. Officer has to check the observation letter registry to check whether the observations have been received on time or not. If there are no observations received within the specified time frame/ date, reminders are to be sent after 14 days. Searching for such information is practically a very difficult task since it is required to check the large registry of Observation Letters every day and send reminders for those who are not submitted the observations on time.
6. Information related to a cabinet memorandum is not readily available. To check the status of a memorandum, currently there is no proper way other than searching them in manually maintained registries for such details.

### 1.3 Motivation

Information systems are major components of modern business environments, regardless it is a Government or Private sector establishment. A Quality Information System will greatly reduce operational overhead and helps to make quality decisions and rich job satisfaction etc.. As an employee of the Office of the Cabinet of Ministers, I greatly felt that having a document automation system would effectively help the institution, to reach its goals. Therefore, author was greatly encouraged to do this project for the Office of the Cabinet of Ministers with the knowledge gained through the MIT program.

This project work provides a LAN based software solution, mainly for the Memorandum Branch of the Cabinet Office. This system will reduce the paper work and automate the document processing system with more management facilities. This study was helpful to gather more domain knowledge and much more skills about the project management and software development process.

### 1.4 Project Objective(s)

The objective(s)/purpose of this study is to develop a computerized document management system called "CPPS" to access information in three languages, prepare agendas and reminders, draft decisions and many other documents, and with maximum controlled access to information. Accordingly, it is expected to use the proposed Trilingual Cabinet Paper Processing System (CPPS) to automate the functions of the Memorandum Branch of the office of the Cabinet of Ministers. The main objectives of the project are summarized below.

- **Secure Data Store:** to securely store Information about cabinet memorandums including the title of the memorandum, received date, signed date, Ministry, type of memorandum, agenda details, observation details and the minutes of meetings in all 3 Languages etc. in a centralized location.
- **Maximum Efficiency:** to generate observation letters/agendas very quickly and effectively through a user friendly interface. This System will generate observation letters, agendas and draft minutes and template of the decision with a single mouse click.
- **Increase Productivity** – to dynamically generate all relevant documents by using the new system to increase productivity.

Ex: 1. User can generate an Observation calling letter very quickly in this new system instead of wasting time on preparing those letters manually. Not only that, the new system automate the entire document creation process, but also capable of generating them in all 3 Languages.

2. Creation of Agenda in no time. (Generate dynamically). User only needs to feed the serial numbers to the system and system will dynamically generate the Agenda as per the format (Template).

- **Reduce work load:** it is expected that the new system will generate 60% of the minute file dynamically. The remaining 40% will be completed manually by the officers. So, the new system would reduce the document preparation time, which indirectly minimize the chances of errors in documents as well.
- **Search Engine:** Finding any cabinet memorandum instantly; System supports for searching full-text (users can search not only memorandum serial number but also a part of the text in the title). This function gives a very high level view of the cabinet memorandum as well (the current status of the memorandum).
- **Paperless office:** Since the information is readily available in the system, process of coping and keeping printed documents can be minimized.
- **Instance Access:** New system is at centralized location (server) where authorized users can access information from any workstation at any time.
- **High Level Summary:** At the end of the day; the system produces a summery report consisting of total number of Memorandums received, total number of Observation calling letters / reminder letters generated and the total number of observations received. This function helps the officer to cross check the day's work as well.
- **Decision Support:** Management / Executive reporting facility (the system will generate periodic and ad-hoc reports)
- **Trilingual Support** – the new system supports dynamic automation of documents in all 3 languages (Unicode integration).



## 1.5 Scope of the Project

This project aims to automate the entire work flow of the Memorandum Branch of the Office of the Cabinet of Ministers and ease the critical document preparation process with minimum effort. The new system will speed up the paperwork process, make quick decisions about the status of a cabinet paper and provide quick search facility to access important information required for making management decisions. The new system will speed up the process of creating reports and centralize the storage of all information. In addition, the new system is capable of performing most of the activities not only in English language but also in all three languages (Eg: storing cabinet memorandums, agenda and minute details in all three languages). The new system will eliminate delays and reduce typing errors and provide quality and certified documentation.

After the Cabinet Meeting on Wednesday the decisions taken should be sent to the relevant Ministries by Friday. This is also a very responsible and meticulous process. The proposed computer system significantly automates this sub-process as well. (The system will create a draft decisions file which contains 60% of the minute file)

**Out of scope** – the proposed system does not have the facility to send the observation letters and decisions to the relevant ministries and institutions by email or other secured way because there is no consensus on e-mail communication between the Cabinet Office and other Ministries yet. This can be considered as a future improvement.

## 1.6 Significance of the study

This study is significantly important and benefited for the following persons;

- **Progress Assistant:** responsible for administering Memorandum Branch under the supervision of the Secretary to the Cabinet.
- **Memorandum branch employees:** responsible for entering cabinet memorandum data in all 3 languages, scan them to the system, preparation of observations, reminders, agendas and minutes etc.
- **Secretary to the Cabinet :** Head of the Department
- **Assistant secretaries :** responsible for administering policy, establishment, tender and legal matters pertaining to the Cabinet Memorandums
- **Translators:** responsible for translating documents.
- **Word processing staff:** responsible for word processing (Agenda, minutes, decisions etc.)

## **1.7 Dissertation Overview**

The First chapter is an Introduction that describes the scope, motivation, and the content of this dissertation. Then, the Background Chapter which describes the project background, similar systems studied and comparison of features and finally describes about methodologies. Third chapter is about Analysis and Design. This Chapter describes the main business of the office of the Cabinet of Ministers along with the problems exist in the manual system, identify the requirements and how the information is gathered and the details of the proposed system. At the end of the third chapter, design techniques that are used for this project are described. Here, mainly focuses on UML diagrams use to design the application. The software and hardware environment use to develop the system will be discussed in Implementation chapter. It also describes the other software tools, third-party components and frameworks use. The testing approaches followed and the test results will be mentioned in detail in the Evaluation Chapter. Also it contains how the client feedback was useful to fulfil their expectations. The final chapter is the Conclusion Chapter which includes achievements and the knowledge gained through this project, problems encountered and how the problems are solved, lessons learnt and the information about future improvements of the system.

At the end of the main 5 chapters, Appendixes, System documentation, design documentation, user documentation, test results, sample reports and code listing are shown. Finally, the reference section states all the materials that are used to write this thesis.

## **Chapter 2 Background**

### **2.1 Chapter Introduction**

This chapter is about the general project idea and its importance. The chapter also introduces the background, purpose and significance of the study. It proceeds by details about Similar Systems studied and outlining the proposed system, methodology, comparisons with similar systems.

At the end of this literature review, it was identified that combination of RAD and iterative models is the most appropriate development methodology for this development. It also shows that .NET is the ideal framework (tool) and Layard (n-tier) architecture for developing this client-server information system.

### **2.2 Project Background**

We are at the center of technology. Of course, it is worth noting that there is no field that is not affected by technology. As the Office of the Cabinet of Ministers, there are lot of information that need to be handled fast, accurately, decisively and timely manner. How rapidly and frequently does information grow in the Cabinet Office? How is the information handled and stored with its rapid growth to affect the functions of the Cabinet Office? How do we ensure that those who have access to information can access it at any time? To fulfil the above requirements, information is required to collect, store, classify, analyse, search, sort and disseminate. How an information system could answer this fast growing demand?

Just like any other institution, the Office of the Cabinet of Ministers is also affected by computing, rely on information systems in order to collect, store, manipulate, and access information. As the Cabinet Office is handling a bulk information and generate documents in three languages (English, Sinhala and Tamil), there is a requirement of integrating the entire work processes into modern technologies. The uniqueness of the procedures and the confidentiality by nature of the business of the Cabinet Office, leads this challenge a bit more complicated. The Office of the Cabinet of Ministers is of course not one of the real world entities where off the shelf / tailor made software systems are heavily used. Due to the large number of memorandums received to the office daily, as well as important administrative documents such as cabinet decisions, the Cabinet Office can be an ideal place to introduce a modern information system. This study is carried out to identify the most important part of the entire business process, investigate the work flows, determine the intra connectivity of processes and identify key functions and drawbacks. In this study had identified that Memorandum branch is handling the most important part of the entire business process and the work flow of the Memorandum branch has to be automated using modern information

technology. In addition, the study also took into account the requirement of storing basic information of the system users (staff) as well for security purposes.

The project title is “**Tri-Lingual Cabinet Paper Processing System [CPPS]**”. Main objective of this software is automating the entire work flow of the Memorandum branch of the Office of the Cabinet of Ministers.

## **2.3 Analysis of Similar Systems**

This sub chapter emphasizes the analysis of other systems / applications that are similar to the proposed system. It shows a comparison and discussion about the main features of those similar systems in order to get a general view. And also what these applications do, and how they are similar and dissimilar to the proposed system.

### **1. FileHold**

FileHold is a Full-featured document management system. This is a feature rich document Management and processing system. Even though FileHold is not exactly similar to the proposed system, some of features are available in it (see the comparison table 2.7.3 for detailed features comparison). Main Features provide by the FileHold are document scanning, capturing, indexing, storing, searching, versioning, and reviewing document workflow and approval. FileHold is integrated with Microsoft office as well. FileHold works in desktop, browser, and mobile device. FileHold document management software makes it easy for organizations to improve security, increase compliance, streamline operations, and reduce the use of paper while saving time and money.

Available at : <https://www.filehold.com>

### **2. Docmosis**

This is a template based document management system where users can make/customize document templates using either Microsoft Word / LibreOffice. Docmosis can quickly and easily add document generating and reporting facilities to any software application.

Available at: <https://www.docmosis.com/how-it-works/introducing-docmosis.html>

**Note :** It is extremely difficult to find exactly similar solution since the unique business process of the client. Therefore, proposed system will be the customized solution for the unique requirements of the Office of the Cabinet of Ministers.

### 2.3.1 Comparison Table (Similar Systems Table 2.1 Similar System Comparison)

Feature		FileHold	docmosis	Existing System
Dynamic Document Preparation (Observation Letters, Agendas, Reminders)		Partially	Partially	✘
Centralized Data Storage		✓	✓	✘
Localization		✘	✘	✓
Timely Notifications		✘	✘	✘
Microsoft Word Integration		✓	✓	✓
Content Searching		✓	✘	✘
Work Flow Management		✘	✘	✘
Reports (Management, Ad-hoc, periodic)		✘	✘	✘ Manually prepare
Secured Login		✓	✓	✘
Status Monitoring and generate Notifications		✘	✘	✘
Mode	Desktop	✓	✓	✘
	Web	✓	✘	✘
	Mobile	✓	✘	✘

*Table 2.1 Similar System Comparison*

## 2.4 System Development Methodology

Software Development Life Cycle (SDLC) is a highly disciplined process in developing a system and allows the analyst / designer / developer to learn how to use the process to analyse, design, implement, develop, test, evaluate and deploy an information system. SDLC tides key stages together in the process. There are many widely accepted SDLC models. Some of them are Waterfall, Prototype, Agile, Rapid Application Development (RAD), Iterative and Spiral. The most widely used said models were considered by the author of the “CPPS”. A comparative study on those models was done by the author before selecting the most appropriate model for the development of “CPPS”.

## **2.5 Comparative Study of Development Methodologies**

### **Waterfall Model**

This model is the most traditional and old software development method. This is a linear approach consisting of key phases. Those are requirement definition, analysis, design, implementation, testing/evaluation and maintenance. In this model, before a phase begins, previous phase must be 100% completed. Hence, usually there is no going back process to change the project or direction.

### **Prototyping Approach**

In this software development model, a prototype (working model) is built, tested, and modified when needed, until required prototype (model) is developed. It also creates the basis for producing the final system. The prototype model works best when project's requirements are unknown. It can be described as a recurring, trial, evaluation, and error correction process between the developer and the client.

### **Rapid Application Development (RAD)**

RAD is a short development process that produces a high quality system with low investment/cost. This Model consists of four phases as planning, design, construction and cutover. The design and construction phases are repeated until the user confirms that the product meets all the requirements. For projects with well-defined objectives and clear requirements and roles, RAD is more effective. Hence, RAD is particularly useful for small and medium scale projects.

### **Agile Development**

Agile methodology practices continuous development and testing of the project's software development lifecycle. Unlike the waterfall model, both development and testing phases are concurrent.

### **Iterative Development**

The process of iteration begins with a simple subset of software requirements and iteratively enhances until the entire system is developed. In each iteration, Design changes have been made and new functions could be added if need. This technique involves building a system through repeated cycles. This is somewhat similar to the RAD model.

### 2.5.1 Comparison of Alternative Design Methodologies

This section focuses on creating and evaluating alternative design strategies. The goal is to select the best of those strategies. Creating alternative design strategies involves a number of questions about system software and hardware, data conversion, development time, and personnel training needs. Keeping all these in mind, it is the skill of the system analyst to select the best strategy. See Table 2.2 for Different methodologies comparison.

<b>Feature</b>	<b>Waterfall</b>	<b>Prototype</b>	<b>Spiral</b>	<b>Agile</b>	<b>RAD/- Iterative</b>
Requirement Spec.	Beginning	Changing	Beginning	Changing	Beginning
Requirements	Well understood	Not-Well understood	Well understood	Well understood	Not-Well understood
Cost	high	high	intermediate	Very high	low
Guarantee of success	low	good	high	high	high
Simplicity	simple	simple	intermediate	complex	simple
Risk involvement	high	high	low	moderate	Intermediate
Modifications Incorporated	difficult	easy	easy	easy	Easy

*Table 2.2 SDLC Methodologies Comparison*

Out of the above alternative design methodologies, the best suited for the given scenario has to be selected that can be built. After considering the above comparison table that illustrated possible plus and minus facts, finally one strategy was selected to design the new system.

## **2.6 Selected Development Methodology**

After considering all those factors and discussing with the Project Supervisor, combination of RAD and Iterative methodologies (hybrid version) was chosen to develop the new system. RAD breaks down the project into smaller more manageable tasks. This task-based structure allows the developer to achieve maximum efficiency. The other major reason behind this choice is the developer works at the stakeholder's organization. It is helped the developer to have regular communication and constant feedback between him and the stakeholder. This is greatly helped to improve the efficiency of the design. Progress could be easily reviewed. Furthermore, selecting RAD and Iterative combination could make easy testing and debugging since it has shorter iterations.

## **2.7 Software Architecture Patterns**

“The success of any software application or system depends on the architecture patterns you use. These patterns not only guide designers and developers on how to design components, but also determine how those components should interact”[1]. Here, three most widely accepted and used architectural patterns were considered and briefly outlined.

### **2.7.1 Layard (n-tier) Architecture**

This approach is most common because it is usually built around the database, and many applications in today's business environment is used it to store information in relational databases. A modern Software application is most commonly broken up to 3 different tiers named layers. They are presentation (what user sees), application (business logic) and data layer (application data). The data enters the top layer (Presentation) and runs way down until reaches the bottom layer (application). Each layer has specific functions for data consistency, re-formatting values to keep them consistent, better security, scalability and for the ease of maintenance.

### **2.7.2 MVC Architecture**

“The Model-View-Controller (MVC) structure, which is the standard software architecture pattern offered by most of the popular web frameworks”[6]. It is also clearly a layered architecture. Just above the database at the bottom, is the model layer, which contains business logic and information about the types of data in the database. Top layer is named as view layer, which is CSS or JavaScript with dynamically embedded code or simple HTML. Middle layer, is the controller, which has various rules and methods for transforming the data between other two layers. In other words, this layer selects the right view that should be displayed to the users according to the user request.



### **2.7.3 Event-Driven Architecture**

Event Driven Architecture (EDA) is the Software Architecture pattern mostly used in more complex and dynamic systems such as transportation systems, social network systems or financial markets. Complex systems are constantly changing. “Change” is the norm. Event-driven architecture helps to manage this change by building a central unit that accepts all data and then delegates it to the separate modules that will handle it. Eg: Search Engine.

## **Chapter 3 Analysis and Design**

### **3.1 Chapter Introduction**

This chapter is about analysis and design of the proposed system. It starts with the analysis part that describes the existing system in detail and its processes. The purpose of system analysis in the system development process is to convert project objectives and defined tasks into new system operations. Most analysts use UML to document System requirements. The method of analysis is the process of sorting the problem becomes one of the elements in the analysis process in order to facilitate solving problems of an information system.

From this analysis method will obtain a decision point that will determine the changeover method whether the old system or the system running will be expanded, replaced or parallel run if necessary.

Systems design is the process of defining a solution. It includes defining how the new system will meet each of the requirements identified during the analysis.

### **3.2 Problem Identification**

Identifying the problem is one of the key functions in system analysis phase. Problems can be identified as questions that are to be solved. Therefore at this stage an effective and efficient analysis is required. Most SDLC models start with the analysis phase, which discuss the requirements of the process with stakeholders. The goal is to detailed problem identification and definition of the system requirements. In addition, this process will ensure that all participants understand the tasks clearly and understand how to implement each requirement.

### **3.3 Understanding the Existing System**

To understand the existing system, an analyst should conduct a preliminary investigation before doing thorough investigation on analysis phase. In preliminary investigation the author used interviews and sampling document reading (existing documents, manuals, files etc.) methods in order to have the clear idea of the existing system.

### 3.4 Existing System

Cabinet Papers are submitted by Honourable Ministers. It is expected that the Hon. Ministers should consult the relevant Ministers before submitting a memorandum affecting the scope of other Ministries portfolio, If not minister can submit his/her Proposals (Memorandums) to be discussed at the Cabinet meeting. A dedicate officer is reserved for receiving Cabinet Memorandums. Once received a Cabinet Memorandum to the Cabinet Office a unique serial number is being allocated to identify it.

Ex: 20/0045/214/xxx

This is recorded in a manual registry along with its title, received date, signed date and forwarded to Memorandum Branch for processing. Memorandum Branch is responsible for processing memorandums (categorizing them, preparing observation calling letters, preparing Agendas for Cabinet / Sub Committee meetings, preparing Draft Cabinet Minute, preparing reminders and preparing decision letters after the decisions are confirmed). Hence, Memorandum branch can be identified as the key section in the whole process. Work flow of the Memorandum branch could be briefly divided in to following main processes.

**Observation Creation Process:** Observations are called from Honourable Ministers. Normally, a Cabinet memorandum is not included into an agenda until receiving the observations from relevant important ministries. Typically Observations must be called from Minister of Finance, Minister of Public Administration and so on. This depends on the subject of the Memorandum. Sometimes Observations are called from many Ministries.

**Agenda Preparation Process (Tri-lingual):** received cabinet memorandums are queued in line for discussing in the up-coming cabinet and subcommittee meetings. Once observations are received, cabinet memorandum is being included in the agenda. Preparation of agenda is truly a word processing task. It is created using some word processing features such as tables, macros, numbering, bullets and paragraph formatting. Typically one agenda may contain around 60-100 cabinet memorandums.

**Cabinet Minutes Creation Process (Tri-lingual):** Decisions taken at meetings must be documented for further processing and send decisions to relevant ministries to take actions. This is another typical word processing task that involves number of officers. This high quality word processing task need attention, care, time and accuracy and must be done within the time frame.

**Decision Sending Process:** This process is the final job done by the memorandum branch. Decisions taken by the Cabinet are highly sensitive and confidential. They are arranged on the number sequence of the ministries and dispatched by hand on Friday along with the next week agenda and relevant cabinet memorandums.

### 3.5 DFD – Existing System

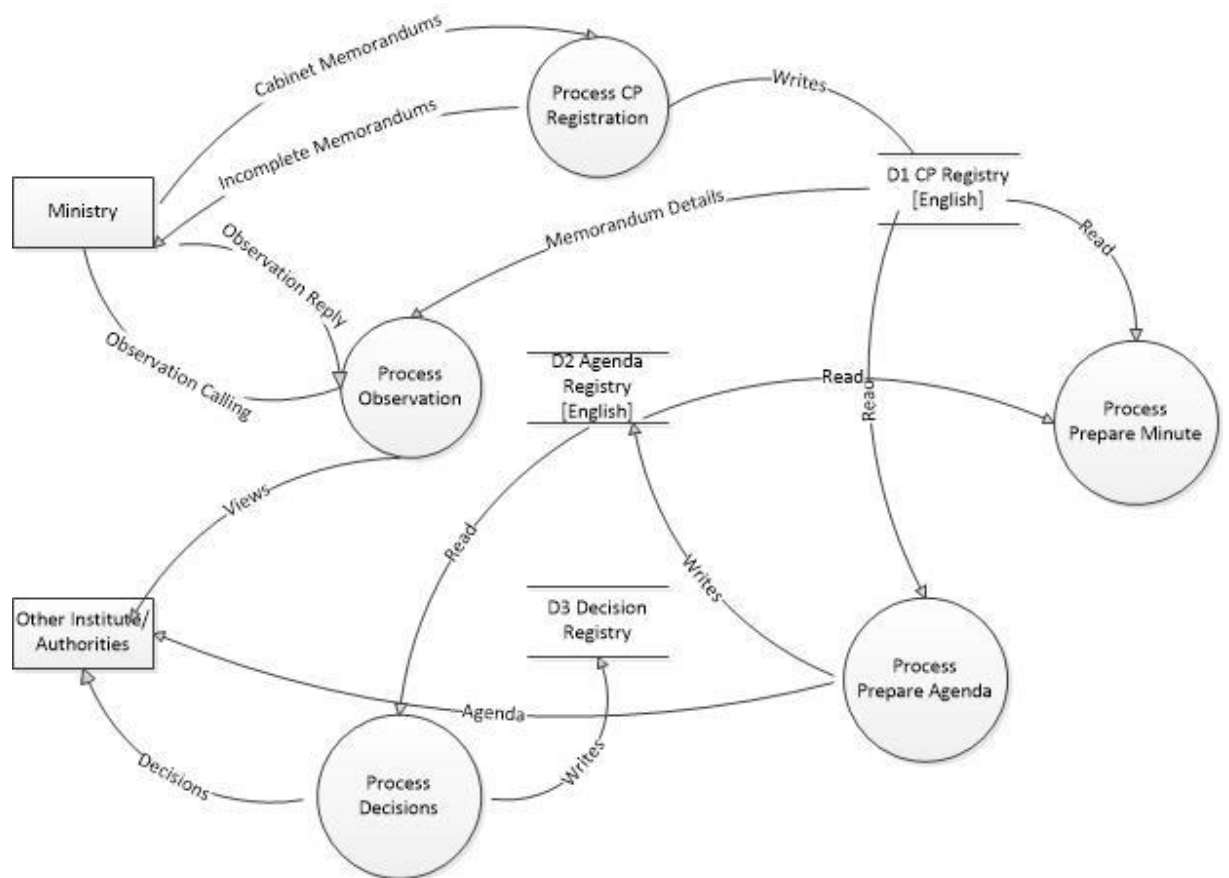


Figure 3.1 DFD Existing System

### 3.6 Information Gathering

#### Who should be interviewed?

The following are the key stakeholders included in this information gathering process.

- Secretary to the Cabinet
- Senior Additional Secretary
- Progress Assistant,
- Administrative staff of the Memorandum Branch.
- Word processing staff.
- Translators.
- Staff Officers of the other departments that represent different levels of hierarchy.

**Methods used to gather information:** information gathering is an utmost important step in the system analysis stage. Mistake in discovering requirements, might develop the wrong solution at the end. Hence, effective facts finding is more important and following are some of the techniques used to gather information.

### Interviews

As a primary information gathering tool, the interview method was chosen to gather information and understand the business process. This is the most common facts gathering technique that happens in face-to-face manner. The purpose was to find, verify, clarify, motivate end-users involved, identify requirements and gather their ideas and opinions. Table 3.1 illustrates the summary of criteria for the interview done by the author.

Position	Purpose	Questions about
Secretary to the Cabinet	<ul style="list-style-type: none"> <li>• Discuss the vision of the new system</li> <li>• Discussing on the benefits of the new system.</li> <li>• concurrence for the improvement of existing system</li> </ul>	<ul style="list-style-type: none"> <li>• How should be the format of observation letters?</li> <li>• The opinion about the draft template of cabinet minute?</li> <li>• Role base access control?</li> <li>• What areas need tri-lingual support?</li> </ul>
Senior Additional Secretary	<ul style="list-style-type: none"> <li>• Overview of the Current system</li> <li>• Current problems with observation generation time issues.</li> <li>• Mistakes in agendas, minutes and observations letters.</li> <li>• Current issues in communicating with other ministries.</li> </ul>	<ul style="list-style-type: none"> <li>• Challenges of the current system?</li> <li>• How can we reduce the number of employees working on each process?</li> <li>• How can we speed up the document preparation?</li> <li>• Information security and the type of information that are to be displayed in the system?</li> <li>• Role base access control?</li> <li>• What areas need tri-lingual support?</li> </ul>
Progress Assistant	<ul style="list-style-type: none"> <li>• Step-by-step discussion about the document preparation process.</li> <li>• To get an idea of content of documents and their format.</li> <li>• This will enable to get a low level and detailed information on the current system.</li> </ul>	<ul style="list-style-type: none"> <li>• How many employees work on a specific process?</li> <li>• How long does a specific process take?</li> </ul>
Management Officer / Officers	<ul style="list-style-type: none"> <li>• To know difficulties in the Current work flow.</li> <li>• To know time to do different tasks</li> <li>• To understand the work flow better.</li> </ul>	<ul style="list-style-type: none"> <li>• How long does a specific process take?</li> <li>• Current templates used?</li> <li>• The time spent on Tri-lingual document preparation</li> <li>• How do currently do many document tasks.</li> </ul>

		<ul style="list-style-type: none"> <li>• What is the biggest pain point related to your work?</li> <li>• What are currently doing to make tasks easier?</li> </ul>
--	--	--

*Table 3.1 Summary of Interviews preparation*

\*Keep in mind this is a relatively flat organization with few employees.

### **Observations**

Observations were done to gain an in-depth idea about how actually business procedures taken place at the actual environment as well as check to see if there are any issues in the business that have not been addressed.

The following processes were observed:

- Volume and Frequency of Cabinet Memorandums receiving every day.
- Manual Registry maintaining process.
- Observation and reminder letter preparation process and their workflow
- Background preparation for minute file creation.
- Volume and frequency of receiving Observation letters.

Being a staff member of the Office of the Cabinet of Ministers, the Observation method was quite easy and time saving process. Author could often visit the actual work environment. During the observation, author recorded necessary steps of the process and potential ways to improve it. The observation was carried out informally so as not to cause any intimidation or unrest to employees.

### **Existing Record Sampling**

Without looking and referring existing documents, it is very difficult to understand the user requirements fully. To get thorough knowledge on how the information needs to be produced, the layouts of letters and reports, the volume and frequency of information Existing Rerecord Sampling was much helpful.

There were plenty of facts to extract from existing documents. Those are:

- Manually Prepared Observation calling letters, reminders, agendas and draft minutes.
- Reports prepared weekly, monthly, on-demand.
- Layouts and formats of documents used.
- Guidelines and format of the Cabinet Memorandum.

- Draft minutes and decisions.
- Format of Decisions.
- Format of the Observation receiving letters sent by hon. Ministers.
- History of the Office of the Cabinet of Ministers.
- Manual Cabinet Paper Registry (English)
- Pending Cabinet Papers list that are to be included in agenda

## Questioners

Questionnaires were used to gather a broad amount of information from clients to see how the new system can best meet their requirements. Author handed over a pre-prepared questioner form with important short answer questions to be filled by elected end-users.

### 3.7 Drawbacks of Existing System

- **More man power:** from manual cabinet paper registration to send observation letters, reminders, agendas and drafting minutes are all done manually. Hence, needs more man power.
- **Time consuming:** preparation of single observation letter, an officer needs approximately 20 minutes.
- **Needs extra attention for creating documents:** most of documents are highly sensitive and confidential letters. Needs great attention when preparing.
- **Report preparation is difficult:** Officers need summaries and reports daily for work. They need to refer to many documents to prepare them. They use Microsoft Excel as a tool to prepare some of them.
- **Errors in Documents (high risk):** since manual type writing can always have errors in documents. Therefore proofreading is an important part before documents (eg. Decisions) are finalized.
- **Slow work flow management:** this led unnecessary time wastage between work flows. For this reason, it is difficult to immediately determine the current status of any Cabinet Memorandum. Ex: Observation of Honorable ministers has been received or not?, have all the other supportive documents ready before being referred to the Cabinet Meeting?
- **No centralized data access:** information is not available when needed
- Searching information about Memorandums, Observation, Agendas, decisions are limited.
- Manual maintenance of information can sometimes lead to loss of information.

### **3.8 Proposed System**

The main objective of this project is to design and develop computerized document management system with improved facilities, which would overcome all the limitations and drawbacks (listed chapter 3.7) of the existing system. The proposed system tries not only to eliminate/reduce those said difficulties, but also increase efficiency, generate accurate information on time. There will be no chance of errors while retrieving information using the proposed system.

### **3.9 Functional Requirements**

- The software should automatically validate users: Only authorized officers/users can use the System. Since new system introduces work flow management, different roles are assigned to. System should tracks individual transaction in the system.
- The system should allow users to enter memorandum, observation, agenda and minute details. User friendly data entry screen should supply with WYSIWYG features.
- Only Managerial and privileged level employees have rights to edit decision data. Software should support Role based access control where each user is divided in to groups and assigned what he/she can do in the system.
- The software system must be integrated with Microsoft word application. Documents generated by the proposed system must be able to open/edit in latest version of Microsoft Word application.
- The software system should instantly do context searching: Rich context based searching facility is needed in the new system even to search paragraph like text.
- System must have administrative functions (user registration, maintenance, change passwords, sever configuration etc)
- Authentication: New system must verify the identity of the users and only allow access onto the system based on a successful authentication.
- Reporting requirements: System must produce periodic, ad-hoc reports.

### **3.10 Non-Functional Requirements**

- Every unsuccessful attempt by a legitimate user to access/edit/alter an item of data shall be recorded on an audit trail.
- Portability. So moving from one OS version to other OS version does not create any problem and software must be available on any architecture (32bit/64bit)
- Availability: System is available and could be accessed from any workstation during office hours.
- Storage capacity: System must support to store large amount of data. System must have facility to store cabinet paper information for at least 20 year.



- Recoverability: should have backup and restore function. Any catastrophic loss should not cause data loss or corruption.
- High Performance: high speed data storage, retrieve and report generation.

### 3.11 Business Modelling

Graphical representation of business processes or workflows is referred to as Business Modelling. This is usually done through a graphical tool, such as flowchart, data flow diagrams. There are many CASE tools available for system modelling such as ‘Smart Draw’, ‘Rational Rose’, Here, the author used Microsoft Visio as a modelling tool.

“UML is a general purpose graphical modelling language”[2]. The main aim of UML is to visualize the system has been designed. It is very much similar to blueprints used in engineering. Business modelling is very easy with UML. It is surely an object-oriented language which is not only allows designer to specify the requirements of a system and capture the design decisions, but also to ease of communication among persons involved in the development effort. DFD, Use-case diagrams, Entity-relationship (ER) diagrams, Activity Diagrams, System Architecture Diagrams in this project were drawn using the these tools. There are a number of different notations in UML.

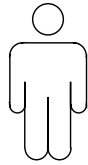
### 3.12 Use case Diagrams and Descriptions

Use case diagrams are used to represent the functionality of a system using actors and use cases. This includes set of actions, services, and functions that a system needs to perform. Since the object oriented analysis and design methodology is used in this project, functional requirements are presented in terms of use cases. The detailed descriptions of these use cases represent the requirements of the system.

- **Use case:** This simply shows how people interact/communicate with the system. It further describes how a person who actually uses that process or system can achieve a goal. Notation:



- **Actor:** It represents any external entity that interacts with the system. It can request services from the system. Notation:



- **Relationship:** a straight line represents the communication between use case and actor.

### 3.13 Use Case Diagram – Proposed System

Use case for proposed CPPS is presented in the following (Figure 3.3).

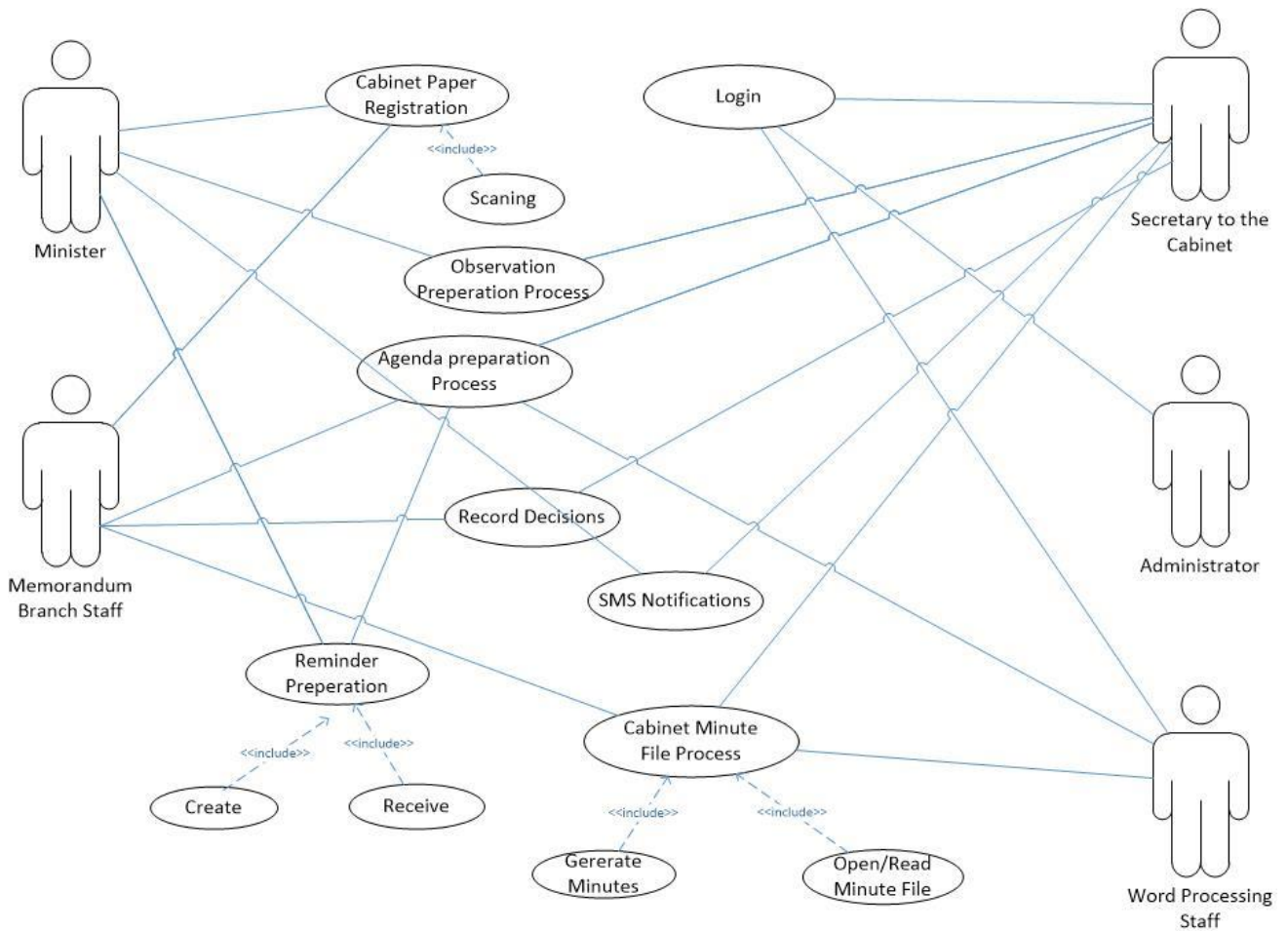


Figure 3.2 Use Case Diagram for proposed system

Detailed analysis using use case diagrams and use case descriptions are shown in Appendix B (Use Case Diagrams and Descriptions)

### 3.14 Use Case Descriptions (General Use Case Description Table 3.2)

Use Case	Use Case Description	Participating Actors and Roles
Login/Logout	This use case describes the event of login to the system and logout of the system	<ul style="list-style-type: none"> <li>• Memorandum branch staff(Primary Actor)</li> <li>• Administrator, Secretary, Word Processing branch (Secondary Actors)</li> </ul>
Cabinet Paper Registration	This use case describes the process of registering received cabinet papers to the system in all three languages and scans them.	<ul style="list-style-type: none"> <li>• Receiving officer, Memorandum branch (Primary Actor)</li> </ul>
Observation Preparation	This use case describes the process of preparing observation letters to be sent to get the honorable ministers views on the matter.	<ul style="list-style-type: none"> <li>• Memorandum branch (Primary Actor)</li> <li>• Authorized officer(Secondary Actor)</li> </ul>
Remainder Preparation	This use case describes the process of preparing observation reminder letters for those not receive observations within stipulated period.	<ul style="list-style-type: none"> <li>• Memorandum branch (Primary Actor)</li> <li>• Authorized officer(Secondary Actor)</li> </ul>
Agenda Preparation (All three Languages)	This use case describes the process of preparation of agendas for up-coming meetings (Main, sub-committee).	<ul style="list-style-type: none"> <li>• Memorandum branch (Primary Actor)</li> <li>• Secretary (Secondary Actor)</li> </ul>
Draft minute creation (All three languages)	This use case describes the process of creating the draft minute based on the agenda generated by the system according to the minute template.	<ul style="list-style-type: none"> <li>• Memorandum branch (Primary Actor)</li> <li>• Word Processing Branch, Translators, Staff Officers, Secretary (Secondary Actor)</li> </ul>
Decisions Feeding	This use case describes the process of entering decisions to the system which were taken at the meeting once they are confirmed (following week).	<ul style="list-style-type: none"> <li>• Staff Officers (Primary Actor)</li> <li>• Translators, Memorandum branch, Secretary (Secondary Actors)</li> </ul>
Manage Users	This use case describes the process of create and manage users.	<ul style="list-style-type: none"> <li>• Administrator (Primary Actor)</li> <li>• All system users(Secondary Actor)</li> </ul>

Table 3.2 Use Case Descriptions

### 3.15 System Design

System Design process can be described as converting of analysis models into design models. Basically it involves selecting strategies for building the new system e.g. Selecting software /hardware/interface/platform on which the system will be run, further, the system design enables the user to see if the system meets the requirements or presents a clear picture of the system. At the end of system design process, produces a report that contains the technical specifications and form of inputs and outputs. (Please see APPENDIX C for Design Documentation)

### 3.16 System Structure Design

A system can also be viewed as a collection of modules. Figure 3.4 shows main modular architecture of the CPPS. These modules will be main components at the implementation stage.

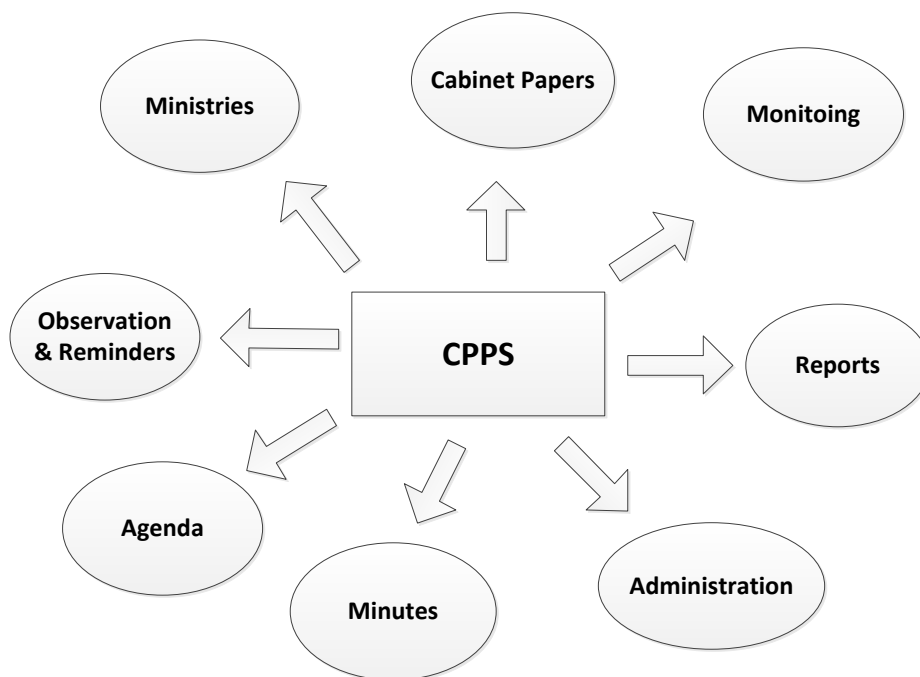


Figure 3.2 Structural Design

### 3.17 System Architecture design

Here the system is viewed as a collection of hardware and software components. The proposed system will be built on existing LAN and allow users to access the system simultaneously during working hours. The identified main users for the proposed system are Memorandum Staff, Secretary, Management Staff, Word Processing Staff and system administrator. The users have different roles and they will be connected to the database

through the interface. GUI interface will make the process of system user's activities easier and practical. Figure 3.5 shows the overall system architecture.

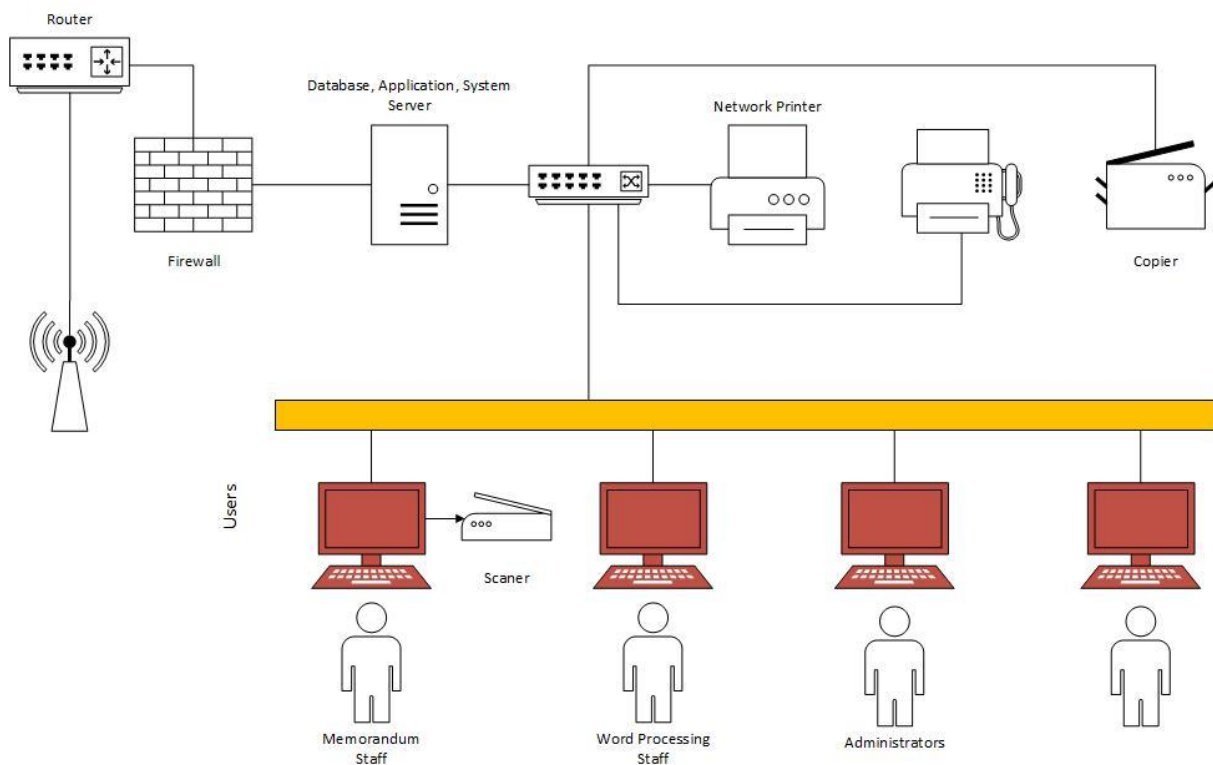


Figure 3.3 System Architecture Design

### 3.18 Data Model Design (ER Diagram)

Data can be explained as known facts which can be reported and have specific meaning (Eg. Cabinet Paper Number). It can be claimed that database is a partly ordered set of data. In addition to the definition, database represents logically linked set of real aspects of the world. To be able to operate with a database, a DBMS must be used. One of the recommended ways of how to develop Relational Model is to create ERD. It is basically a model, based on analysis of requirements. For that it uses entities, relationships, identifiers, attributes and instances. Entities represents an object in real world such as (user, ministry etc.) and is also a set of instances. Attributes are properties used for closer entity specification. Several types of relation cardinalities exists between entities - 1:1, 1:M, M:M. When entity-relational model is finished, the relational model can be created. Figure 3.6 shows the ER diagram for the proposed system. Here, UML Cows Foot Template was used to draw the ER.

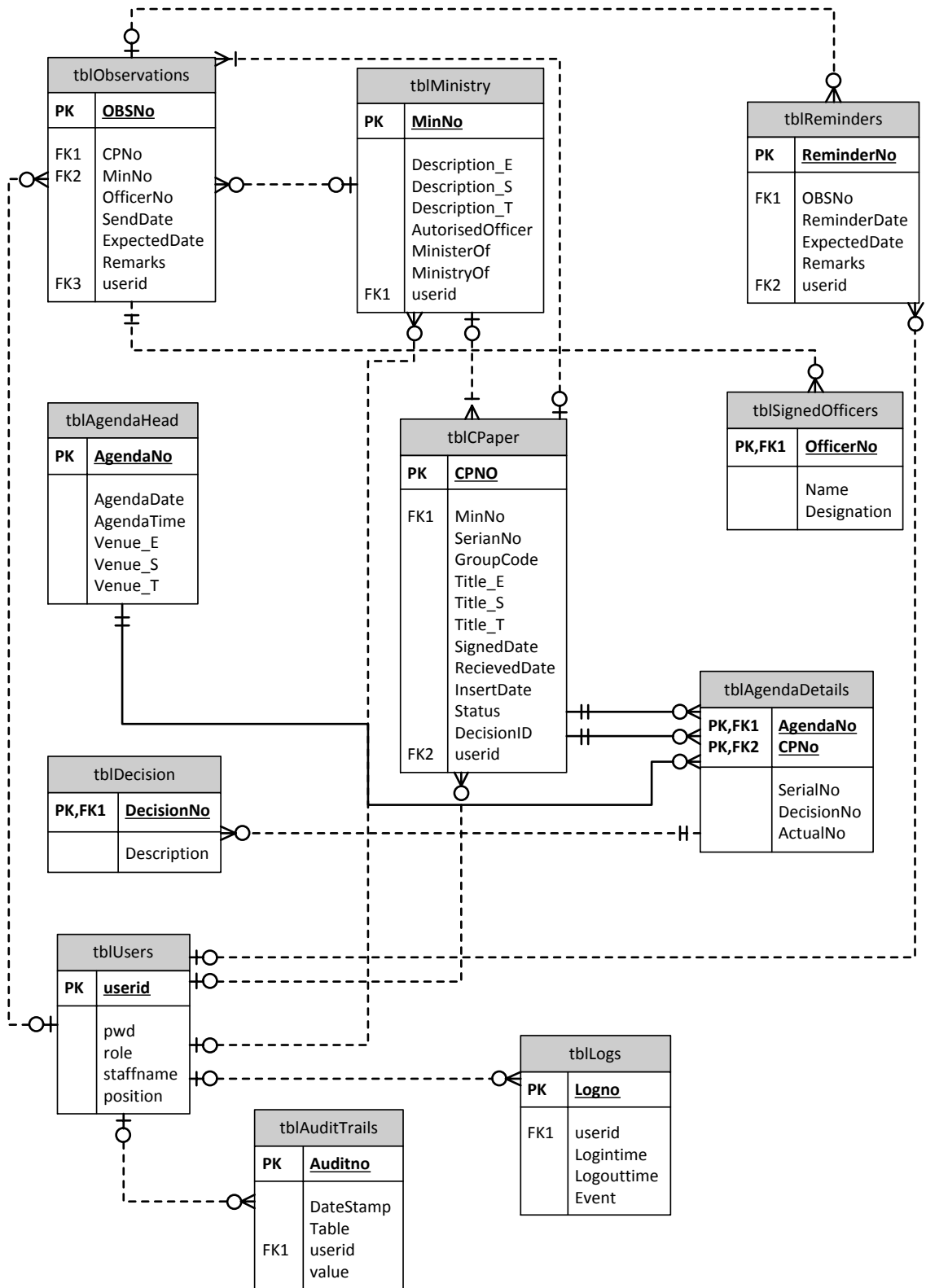


Figure 3.4 ER Diagram








### 3.19 Detailed Design

This is a detailed plan that directly deals with the implementation part and what the system and its subsystems looks like. It is more scaled and detailed towards modules and their implementations. Here detailed design was done with activity diagrams and wireframes (interfaces).

### 3.20 Activity Diagrams

The author used activity diagrams to describe the dynamic aspects of the system. This is similar to a flowchart / data flow diagram that represents the flow from one activity to another or within a single activity. Therefore, an activity is simply an operation or a function of the system. The control is shown by an arrow line that flows from one operation to another. Activity diagrams are often used in business process modelling. Notations used in activity diagrams are shown in Figure 3.7.

Notations:

Name	Symbol
Start Node	
Action State	
Control Flow	
Decision Node	
Fork	
Join	
End State	

*Figure 3.5 Activity Diagram Notations*

Figure 3.8 shows login activity diagram and rest are included in APPENDIX C

### 3.21 Login activity diagram

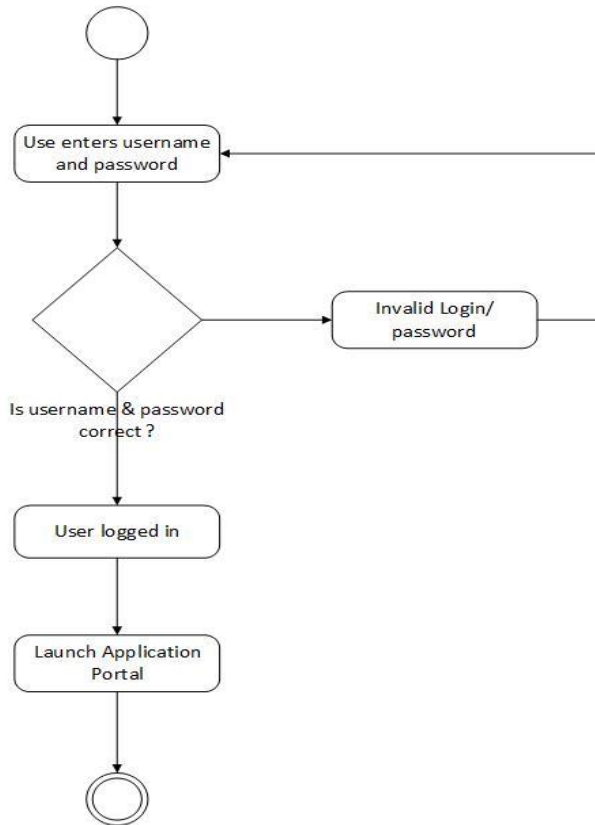


Figure 3.6 Activity Diagram - Login

#### 3.21.1 Wireframe Designs

Wireframe is also known as a screen blueprint which represents the skeletal framework of a system. “Wireframes are created for the purpose of arranging elements to best accomplish a particular purpose”[7]. Figure 3.9 show Login wireframe. Please refer Appendix C for detailed wireframe designs.

#### Login

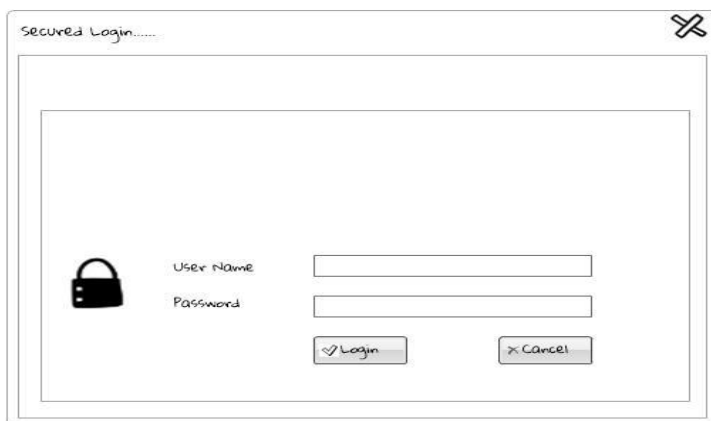


Figure 3.7 Login Wireframe



## **Chapter 4 Implementation**

### **4.1 Introduction**

“Implementation may involve developing programs in high-level or low-level programming languages or tailoring and adapting generic, off-the-shelf systems to meet the specific requirements of an organization”[3].

This chapter discusses the implementation of the system and tools used. It provides technical information about the system including system and software design decisions. It also outlined the system's structure, implementation methodology and procedures. An important part of this stage was that the author created an executable version of the software (working model).

### **4.2 System and Software Design**

The system was developed using a hybrid version of RAD and Iterative models as described in the early chapters 2.4, 2.5, 2.6. This meant that the benefits of the RAD could be combined with the Iterative approach. Thus, at an overall system level Iterative development approach was followed and at a more detailed level, RAD prototype approach was applied. Since it is RAD development, for a chosen module an initial prototype was developed, and shown to the user and re-designed with user feedback. This small subsystem of initial prototype was created for user evaluation (This could be considered to be an initial prototype). After that another subsystem (chunk) is developed. Each is a separate prototype. Lastly, all subsets are integrated and the final version was developed and shown to the user. Each prototype could be considered as a functional component of the final system and testing is done at the end of the each prototype completion. The overall system and software were designed based on user requirements outlined in the early sub chapters 3.9 and 3.10. Then, Integration and system testing were carried out which ultimately can be referred to as the integration of the prototypes as a system and the testing of the whole system. Performance and maintenance must be performed when the system is installed and deployed to live performance. Maintenance is the correctness and improvement of the system. This includes fixing identified bugs. Finally, the User Acceptance Test (UAT) was performed to verify that the system was completed. Upon completion the UAT, the system will then be handed over to the users fully and the necessary maintenance undertaken. Software design activities include the architectural design, module design, interface design (eg: the interface between the sub-systems, component design, data structure design and algorithm design (mainly for automation of document). The overall architecture is shown in the following figure 4.1.

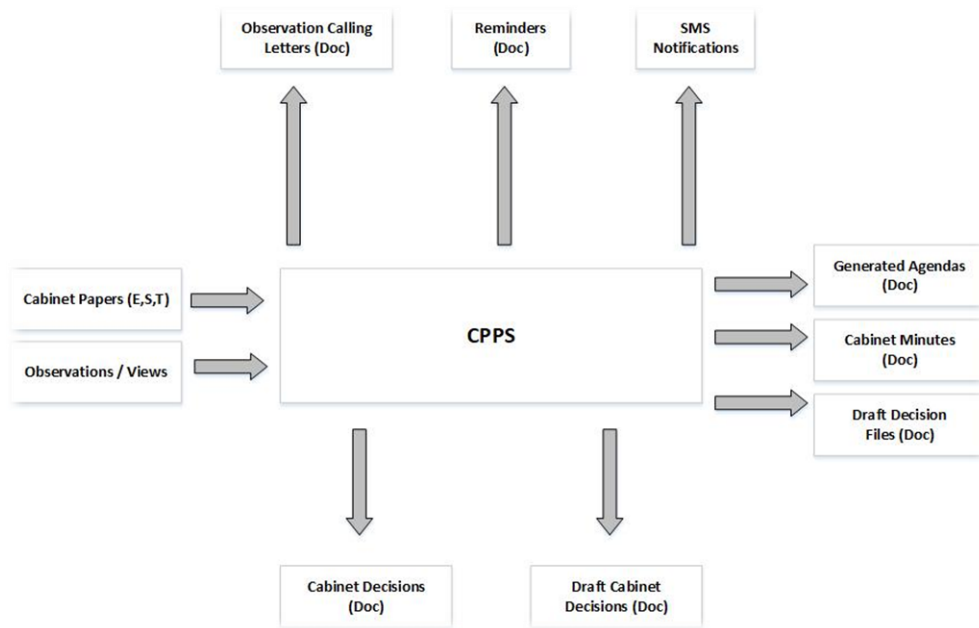


Figure 4.1 Overall system architecture

### 4.3 Design Quality

One important problem in any modern project is design quality. Many factors contribute to the overall design quality. Some of them are module cohesion, module coupling, maintainability and adaptability. “Module Cohesion is the internal relationship between the parts of a component”[4, P 286]. Coupling is a measure of how strength the component interconnections. It is the relation among components. It means that components are independent of each other. It is acceptable to have high cohesion and low coupling. While maintainability is also an important factor since it helps to change when it needs. Factors that affect Maintainability include cohesion and coupling, naming conventions, meaningful identifiers (variables), internal documentation (comments and indentation) and complexity. Adaptability means how easy it is to change the design. This project tried to maintain high cohesion and low coupling, by having all the related elements together in their functions and any changes required could be effected only on component alone, causing no impact on the other components.

#### 4.3.1 Design Decisions

“The architecture highlights early design decisions that will have a profound impact on all software engineering work that follows and, as important, on the ultimate success of the system as an operational entity”[4, P 245] .

**Why Desktop was chosen?** Desktop applications have user-friendly, more responsive, more customizable, better user interfaces, and better keyboard support. Furthermore, desktop applications are easier to implement and available anytime offline and provide user faster response time and rich user experience. Since the unique nature of the business process of the Office of the Cabinet of Ministers, desktop approach was chosen to build high secure information system that ensures high confidentiality and security.

Furthermore, Microsoft Office Application suite is heavily used in current work environment for different kinds of word processing tasks. Current Environment is operated on Local Area Network and use of file sharing and active directory capabilities. Therefore, desktop system is best suited for the existing environment.

**Why .NET Technology and Framework?** This is highly regarded and accepted framework works on object-oriented programming methodology which eliminates unnecessary codes for the developers. [11] .NET has plenty of re-useable code components that help to minimize cost and speed up development (best suited for RAD). Proposed system integrates with Microsoft Office programs, so .NET is also from same family.

**Why MySQL?** Since developer uses software architecture patters Layard architecture where bottom most layer is the Relational database that is used store data in the application. MySQL is best suited for the Layard architecture and more importantly it is free and open source MySQL RDBMS has huge community base. CPPS application is Unicode support application and MySQL does provide full four byte UTF-8 support and also release frequent updates for enhance performance and security [12].

**Why InnoDB?** MySQL offer two main DB engines MYISAM and InnoDB. Preference is given for the InnoDB engine since it supports Referential Integrity, Transactions and Atomicity, Row Locking over Table locking, and greater data integrity [12].

**Why Unicode?** Proposed system is tri lingual support information system and will process tri lingual data. Since the target environment is a Windows based, the Unicode Encoding system is chosen to enable that facility.

## 4.4 Development Environment

Visual Studio/.Net framework combination is used as main development environment and VB.Net is the main coding language for the ‘CPPS’ software solution. The system uses the Layard (n-tier) architecture as mentioned in the early in the Background chapter. .Net Forms, Elements, buttons, built in-components are used in the Presentation Layer of the system. N-tier framework is used for request processing and validations as the Presentation Logic Layer. .Net framework is used as Data Access Layer. MySQL is used as backend database. All technologies mentioned above are open source. Combination of these technologies and frameworks gave many benefits such as validation, transaction rollback, database independency with the help of the robustness of the .Net Development Environment.

### 4.4.1 Software & Hardware Used

Software development involves both Software and hardware tools. This section briefly list of the main pieces of software and hardware used in this project: table 4.1 shows summary of software and hardware used.

Windows Server 2013 R2: Available: <http://www.microsoft.com/>

Microsoft Visual Studio 2019: IDE/Compiler Available:

<https://visualstudio.microsoft.com/vs/community/>

Microsoft Windows 10: Available: <http://www.microsoft.com/>

MySQL 5.0 Available: <https://dev.mysql.com/downloads/>

MySQL Workbench: <https://dev.mysql.com/downloads/workbench/>

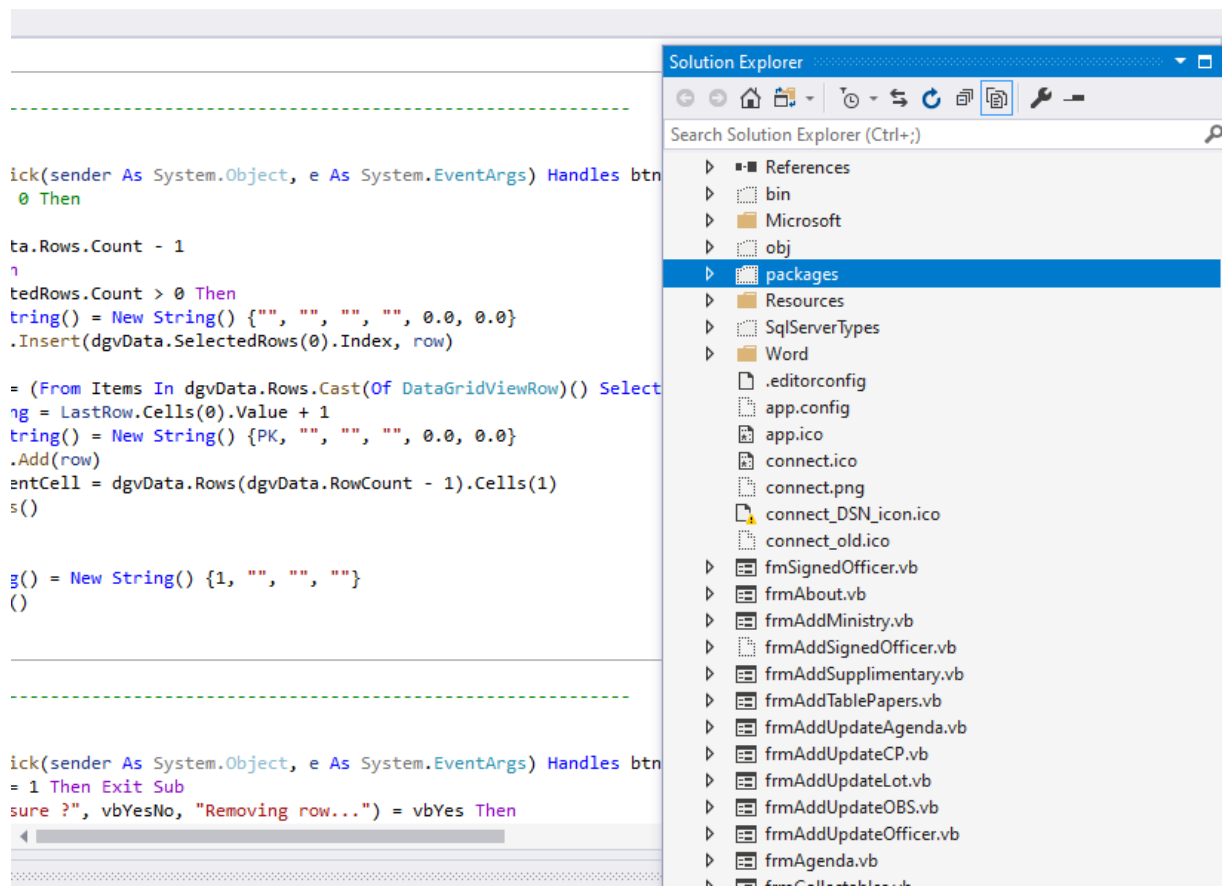
Feature	Server Machine	Development Machine
Processor	Intel Xenon Processor	Desktop computer with 2.9 GHz Core i5 processor
Memory	8GB RAM	4GB RAM
Storage	1TB Hard Disk, RAID	256 GB SSD
OS	Windows Server 2013 R2 64 bit	Windows 10 64 bit
Input	Keyboard and Mouse	Keyboard and Mouse
Database	MySQL	Workbench client tool
Other	Internet Connection	Internet Connection

*Table 4.1 Software & Hardware Summary*

#### 4.4.2 Re-usable Components

- **DateTimePicker Control** : This control allows users to select / edit dates easily with the help of built in combo box components. If user clicks on the drop arrow in the DateTimePicker control, it displays a month calendar where user can select the required date.
- **ListView Control** : The ListView control is used to display a list of items. This project used this control heavily as it has very rich properties. This control displays a list of items along with icons and allows easily adding and removing items from it. This control allows displaying data in tabular form. Furthermore, this control allows adding other components like Check Boxes; Radio Buttons in combination and filtering, pagination are also possible.
- **Microsoft Report Viewer**: To view reports that has been deployed. This control allows render reports easily with its navigation, pagination, printing functions.
- **ToolStripe Control**: ToolStrip control provides functionality of a Windows toolbar controls. This class represents and serves as a base class of MenuStrip, StatusStrip, and ContextMenuStrip classes and helps to build toolbars, menus etc.
- **Windows Image Acquisition (WIA)**: CPPS supports document scanning. For this developer used WIA platform. The WIA platform enables any application to interact with imaging hardware and standardizes the interaction with the application and scanners. This re-usable component greatly reduces coding effort since this tool allows application to talk to and interact with different scanners without requiring driver coding.
- **Microsoft Interop API** : to communicate with Microsoft COM modules this standard Microsoft Interface was used.

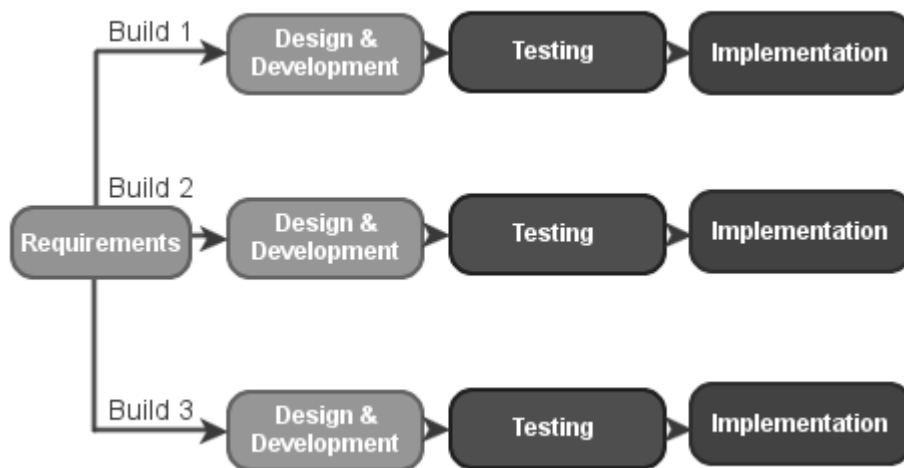
## 4.5 Project Folder Structure



## 4.6 Implementation Methodology

There are different types of implementation approaches such as direct, parallel and phased implementation [2]. To implement this CPPS, combination of phased and parallel methods were used since the CPPS has modules as per in the high level architectural design showed in the (Figure 3.4). In this method, before completely changing over to new system, modules developed first are installed and run simultaneously with the current system until users are well trained and ready to roll out the entire system.

Phase wise implementation process enables to execute by incrementally aligning the product with the end-user. Figure 4.2 shows phase wise implementation process.



*Figure 4.2 Phase wise Implementation Plan*

First, implementation was done after completing 1<sup>st</sup> build. Then the build (module) was demonstrated to the end-users. They were given chances to work and test run it. System was tested by the end-users individually and then they were asked to test the system in client-server environment (LAN) for the entire cycle. Then based on the user feedback, adjustments/modifications were made. First module is mainly for the data entry work (CP Registration module) and there are lot of pending Cabinet Papers to be entered to the system. This data entry task takes much time. Parallel run was done for a week after end-users verified that all functionality is existed. Second build was implemented after identifying that there were no any operational issues and all the system outputs of the implemented module were as expected. That was Documents Generation module where agendas, observations, reminders and some reports generation is automated. This process continues and moves to implement next build and so on, and then live implementation was started. All users were given user accounts and passwords in a live implementation. Because user access rights and roles are linked to user accounts, access is activated after logging into the system, and the user is permitted only for his or her role. Password change facility at any time is provided to users

through the system. For some areas, after a live run, the system has the necessary procedures to back up the database every night for use in case of data loss or hardware failure.

## 4.7 User Training

The key users of the system were trained to truly understand the system before it started live. Since the system is iteratively build; necessary user training was given after each module installation to the relevant users/groups prior to whole system is operated in live. If users don't like it, they are less likely to work in a system. If they believe that the system does not help them, or they believe the system is not too easy to use, it cannot be considered a successful system. So onsite training is crucial. Author followed two training approaches here.

- **Individual Training:** Even though this is more costly method, but most effective. Each user was individually given training on how to perform common tasks.
- **Classroom style training:** Group training was organized for group of users to train/show users how the software works and how to perform common tasks. Usually group is about 15 to 20 persons.



# Chapter 5 Testing & Evaluation

## 5.1 Introduction

“The goal of testing is deliberate intention of finding errors. A good test is one that has a high probability of detecting errors. Therefore, it is very much crucial to design/develop/implement a computer based information system with high “testability””.

Testing and evaluation procedures are discussed in this chapter. In SDLC, testing and evaluation play a critical role. This is because it verifies and validates the software and checks that it works as expected. It is essential to check whether the software meets the needs of the stakeholder. A preliminary examination of each functionality and behaviour is carried out in the sub-component implemented by the developer. Therefore, the following testing approaches were used to make this phase a success.

## 5.2 Testing method

During the system design phase test cases were prepared to test the system with sample data that includes field names, expected outputs, interface standards and this was later compared with the actual test results.

**Unit Test:** Author followed Manual Unit Test approach. Here, the unit testing was done with white-box testing according to test case. Further manual unit test was done in one/two or three steps. Author executed UI Test Scenario (for some input forms) to verify all input forms and their elements were according to the requirements and UI Fundamentals. Then Functional test scenario was done to verify each input element’s functionality and Security Scenario (for some input forms) was done to verify whether the system conforms against security standards.

**Integration Test:** after finishing implementing modules, integration test was performed to check whether inter-communications between modules are properly functioning.

**System Test:** System test was done after all modules were integrated properly to test all non-functional requirements of the system.

**UAT:** Then the system was implemented in the client site and the client performed user acceptance testing (UAT). The main purpose of the UAT is to check the validity of the business flow from beginning to end. It is a kind of black box testing where two or more end-users are involved.

CPPS Application is forms based solution. Most input screens, which are presented to the end-users, are forms. Therefore author followed specific test case considerations (Table 5.1) when test all data entry forms.

<b>Required Fields</b>	User cannot save/proceed without filling required fields and gives user a friendly warning message accordingly if the data fields are left blank.
<b>Data Type Mismatches</b>	If the input form has fields for dates, numeric, currency or other specific data types, ensure that only valid data can be entered.
<b>Field Widths</b>	For text box fields that allow data entry, make sure that the width of data entered does not exceed the width of the table field (a friendly validation message will be displayed if exceeds).
<b>Save Confirmations</b>	System will give a friendly confirmation message if users add/edit all required data fields and also allowing changing of data without saving, user will be prompted save if he/she leaves the screen.
<b>Delete Confirmations</b>	If the user deletes an item, user is prompted a friendly confirmation message.
<b>Type ahead</b>	When is using combo boxes or some specific search occasions, type ahead facility is provided.
<b>Table Scrolling</b>	System is equipped with tabular (grid) display of information data to the user. If the information stretches past one page, the scrolling facility is given. This scrolling facility is given both horizontally and vertically (table headers keep intact).
<b>Error Logging</b>	In the event of fatal errors occur; ensure that the application has facility to writes those errors to a database table for later review.
<b>Error Messages</b>	Make sure all error messages are grammatically correct, informative, and not condescending.
<b>Shortcuts</b>	Ensure that if short cut keys allows in the application (like CTRL+S to save), test each shortcut to ensure it works in all situations and no collision with built in shortcuts.

*Table 5.1 Test Case Considerations*

### 5.2.1 Login Test Case

This section describes test cases and scenarios used in the Login screen testing process. Rest test cases / scenarios and results are illustrated in Appendix E

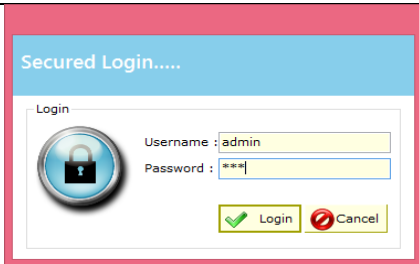
**Test ID: 1**

**Function: Functional Test Scenario for User Login**

**UI Test Scenario**

- Verify that all the text-boxes, labels, buttons are available in the Login form.
- Verify that the all form element properties such as font and font size of the labels and the text written on are clearly visible.
- Verify that the all UI elements are as per the specifications.

Detailed functional test scenario is given in the Table 5.2

Test Case#	Function	Expected Output	Status
1	Verify that immediately after the login screen opens, cursor should remain on the username.		Pass
2	Verify that the user can access different controllers using the "TAB" key on the keyboard.		Pass
3	Verify that the password is in masked.		Pass
4	Verify whether or not the password can be copied.		Pass
5	Verify that the user has access to the system by entering a valid username and password and clicking the "Login" button.	System Launches	Pass
6	Verify that the user can login by entering valid username and password and by pressing the "ENTER" key.	System Launches	Pass

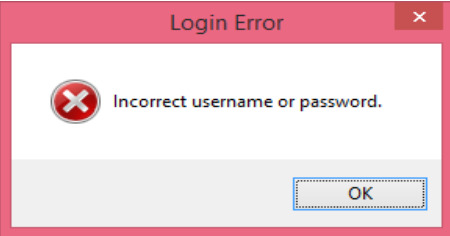
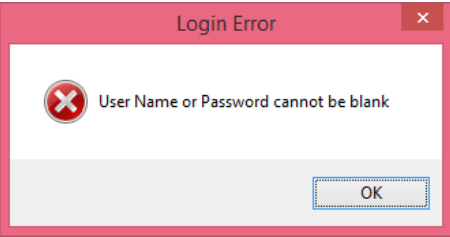
7	Verify that the system is not accessible with an incorrect username and password.	Validation Message 	Pass
8	Verify that the validation message should be displayed if there is an empty username or password	Validation Message 	Pass

Table 5.2 Login Test Case

### Security Test Scenario for Login

- Verify that there should be a limit for number of unsuccessful login attempts by the user. - **Pass**
- Verify that in case of wrong credentials, system must display a message like “incorrect username or password” instead of an exact message pointing to the incorrect field. This allows the hacker not to give a clue as to whether one filing is correct. Otherwise he just wants to try different combinations on other fields. - **Pass**
- System must show the name of the current logged user. - **Pass**

### 5.3 Feedback and Evaluation

At the UAT, a user feedback (Figure 5.1) form was distributed among the different types of users to cover all functional areas of the system to evaluate the system and to get their satisfactory level by allowing the defined questionnaire to be answered freely. Feedback summary prepared after getting user feedback is shown in the Figure 5.1

<b>User Feedback Form</b>	
Name.....	Date.....
Title/Role.....	
Give your feedback for the following items	
Description	Rating [1-5]
User friendliness	
Features	
Speed	
Performance	
Easy to use	
Accuracy	
Comments.....	
.....	
.....	

*Figure 5.1 Feedback Form*

### 5.4 Analysis of Feedback

After getting feedbacks from various users, a summary was made for evaluation. Table 5.3 shows the summary of feedback. It shows positive user reactions and therefore, it can be considered that the system was successfully developed and they accepted the system. Considering the results of the UAT, all the operational requirements provided by the Client are met and the new system ensures that users can perform their tasks effectively and efficiently. Few minor changes were made to the system based on user feedback.

Module /Function		Completed Function		Ease of use for function				
		Y	N	v.easy	easy	neutral	difficult	v.difficult
Login		✓		✓				
Navigate through main menu (responsiveness)				✓				
Icons responsiveness				✓				
Adding CP					✓			
Adding an Agenda					✓			
Observations				✓				
Edit/Update Function					✓			
Create Minute				✓				
Change Password					✓			
Localization	Sinhala				✓			
	Tamil		x					
CP Progress Monitoring				✓				
Reports					✓			

Table 5.3 Feedback Summary

## Chapter 6 Conclusion & Further Work

### 6.1 Introduction

This chapter briefly discusses how successful the design, problems faced and how did overcome those, what have been achieved, limits of the work carried out and possible future enhancements. Office of the Cabinet of Ministers is planning to go e-Cabinet (Fully Automated Information System) solution in the future as it is in Estonia [Available: <https://e-estonia.com/solutions/e-governance/e-cabinet/>].

Scope of this system is to automate Memorandum Branch work flow and its related functions. That includes automatic creation of documents (observation calling, reminders, agendas, draft decisions, draft minutes etc) and one touch information search. The system was built using technologies Layard Architecture, OOP, VB.Net with .NET framework and MySQL.

In the first two chapters a brief overview of the technologies used, such as .NET Framework, VB.NET, MySQL Server, was introduced. The Analysis and Design chapter sheds light on the basic features of the CPPS and the sub-sections mentioned. In addition, this chapter outlines the requirements for an application, from the perspective of the stakeholder. In Design Sub-Chapter, these requirements are broken down into composite components consisting of the data model, application architecture, and sub-module design. Finally, the project objectives are met through the Implementation chapter, which is basically the same structure as the previous chapter. Chapter 5 summarizes how CPPS inspection, deployment, and evaluation are performed.

By the time the report is written, the first version of full CPPS was successfully deployed. According to customer feedback as stated in Chapter 5, this solution met with success in all aspects. Thus, it can be presumed that implemented solution suits all defined requirements. However, despite of the solution success, the report introduces the sub chapter called Further Enhancements for further extensibility, which according to Author could be implemented in the next CPPS version as it make as fully automated CPPS. On the basis these information it could be claimed that all objectives were considered were successfully met.

## **6.2 Problems Encountered**

From the very beginning it was difficult to grasp the right requirements due to the narrowing IT knowledge of the stakeholder. Client was finding difficult to exactly express the requirements and how information should be produced by the proposed system. Due to this, author had to conduct interviews few times to gather exact requirements and alternative methods and procedures. And also stakeholders were not always present for requirement gathering since the target environment is so critical Government body and officers are always busy with their work. The system analysis and design phases were therefore somewhat delayed. Hence, the additional time needed for the analysis phase was wider than planned. At the later part of the development process there was pandemic situation due to COVID-19 in the country that also caused delays at testing and evaluation. While developing the system, some key requirements had to be changed and accordingly modifications were made in database structure to meet those requirements. So the author needed an extra time to rewrite the code parts of specific affected modules.

Another issue with the development process was that the author has to study the localization integration and the Unicode system and some language functions relevant to. To grasp the essential level of knowledge in these sections, author used online tutorials and books

## **6.3 Learning Outcomes**

The knowledge gained throughout the project is truly valuable. In the process of learning from feasibility to the end of development, the author gained a wealth of valuable experiences. It is worth remembering that the project has provided a great opportunity to test and implement the most important theories, disciplinary practices and new technologies learned throughout the MIT degree program. It facilitated to learn new and updated technologies like of (.Net Framework, COM Modules, Unicode, Entity frameworks). Specially should mention here efforts were taken to learn the Layard Architectural Design Pattern was truly a valuable one.

The ability to work with many people in the collective field, improve technical and intellectual skills are some of the things that I gained from working on this project. Proper project planning and time management are another important aspects learned in this project.



## **6.4 Further Enhancements**

“CPPS” is developed by analysing the requirement of Office of the Cabinet of Ministers in Democratic Socialist Republic of Sri Lanka. The advantages of this software are that this can be enhanced, modified or changed to the growing requirements of the client, and it is developed using open-source software tools. Some of the improvements that can be implemented to this Document Automation System are adding the entire workflow of other departments in the Office of the Cabinet of Ministers including and automate complete decision making and decision sending process as future enhancement (Fully automated e-Cabinet).

## **Appendix A System Documentation**

### **Installation**

This section provides guidelines to the setup of the CPPS program. This is to assist the users to install the system in their PC's, Servers or Workstations. Installation is simple, as the new system will be installed in the existing environment. Hardware acquisition is not part of the installation process since system was developed for existing LAN environment which is being already installed and configured. Infrastructure is being facilitated (Servers, Work Stations, Switches, Routers, Cabling all being existed). Therefore, there are no extra steps to be taken in to consider for the hardware acquisition. The developer considered that users were too busy using the current system and didn't have time to involve into this. These non-technical issues delayed the deployment process and slowed down the adoption and use of CPPS. The installation procedure is given bellow;

- Small installation program is written to install at client workstation which is being readily available on shared installation folder in the file server.
- Database configuration is being included in the installation program and as a system option is to set database configuration if user is willing to do so later on.
- When first deployment is available, an executable installation program is given to the user with the access path.
- There is no special release of the system for each customer. But first release one is being installed into the CP Registration officer.

### **Feature Upgrade**

As releases are goes on application functionality should spread across many computers in the network. An important factor is that; when the application software has to be changed, reinstallation should be done on every client computer. This issue is simply overcome with the inclusion of Auto-update feature. A system notification is given to the user that an upgrade is available so that user can upgrade it via the system.

## Appendix B Use Case Diagrams and Descriptions

**Login/Logout Use-Case:** Figure B.1 is Login Use Case Diagram

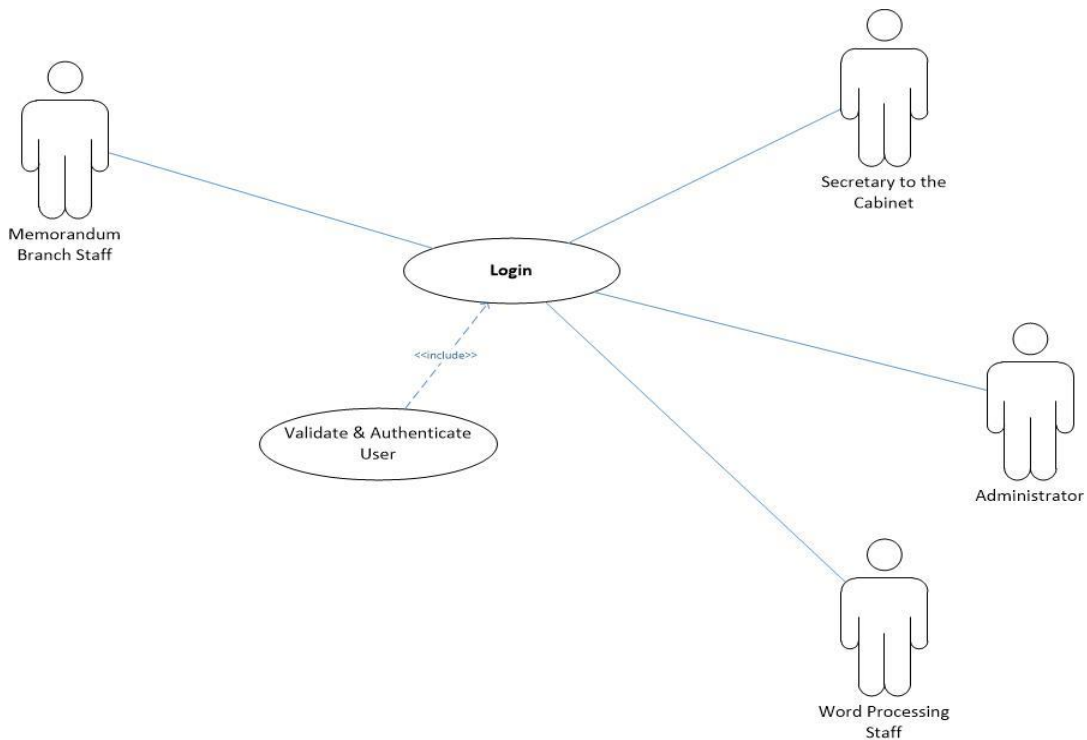


Figure 1 Login Use Case Diagram

**CP Registration Use-Case:** Figure B.2 CP Registration Use Case

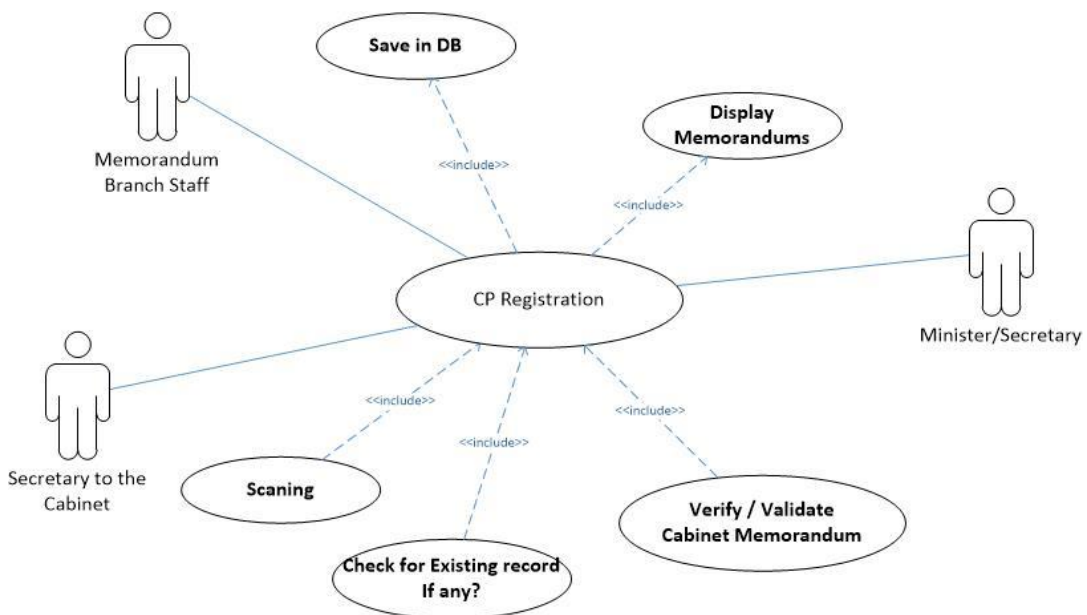
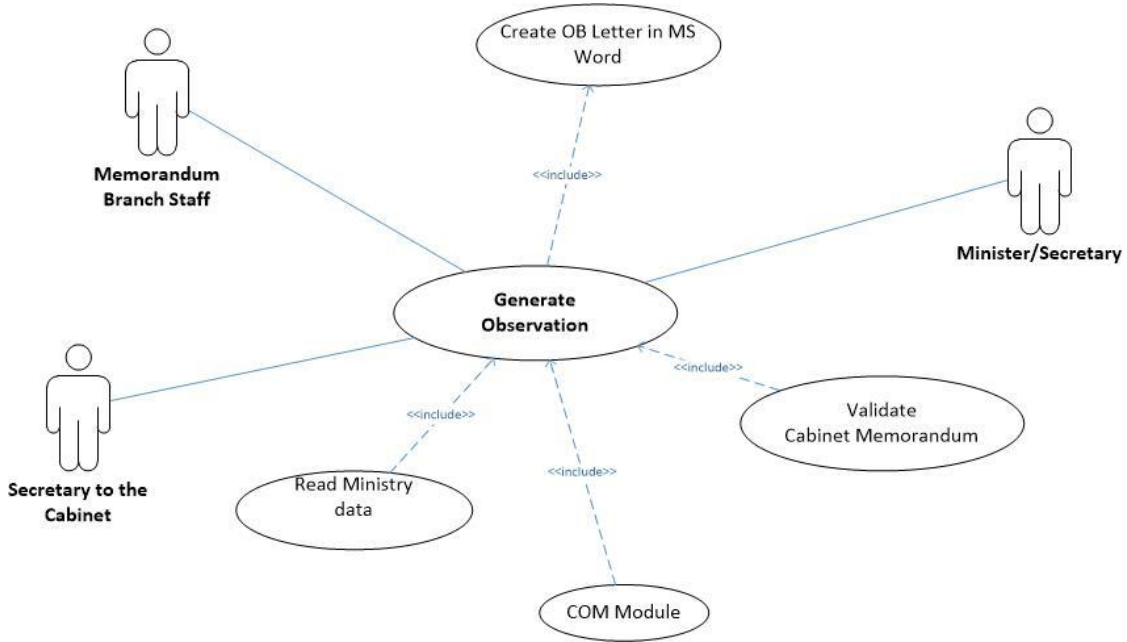


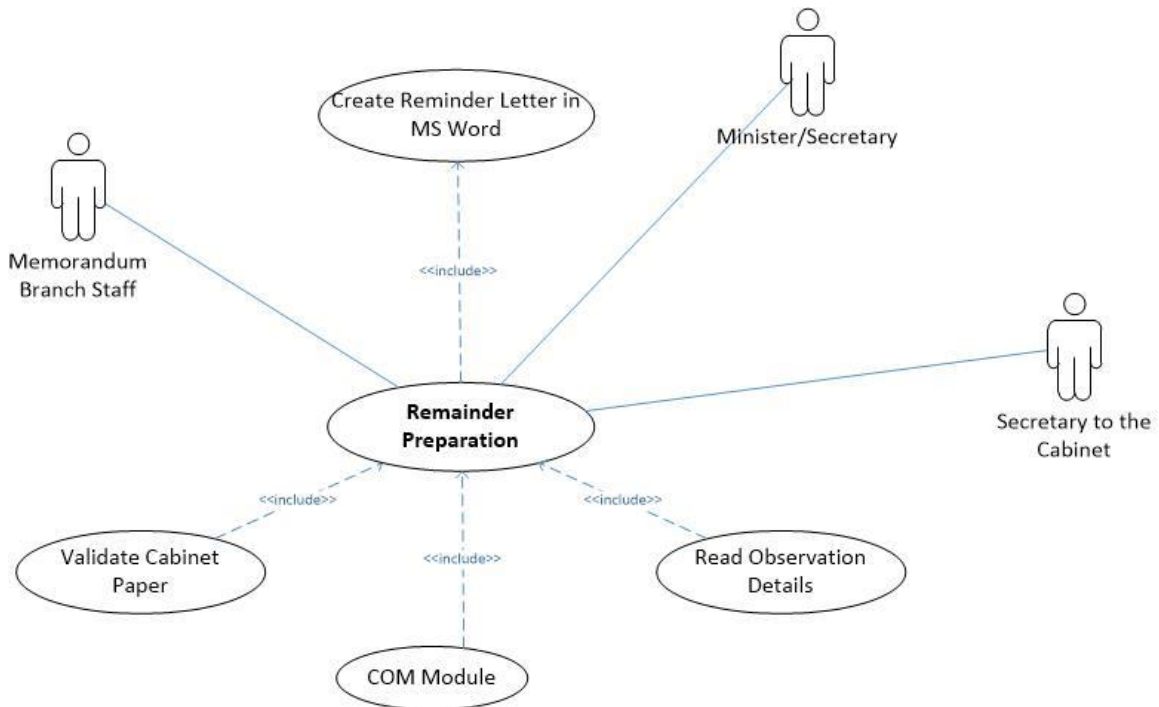
Figure 2 CP Registration Use Case Diagram

**Observation Letter Preparation Use Case :** Figure B.3 is Observation Preparation Use Case



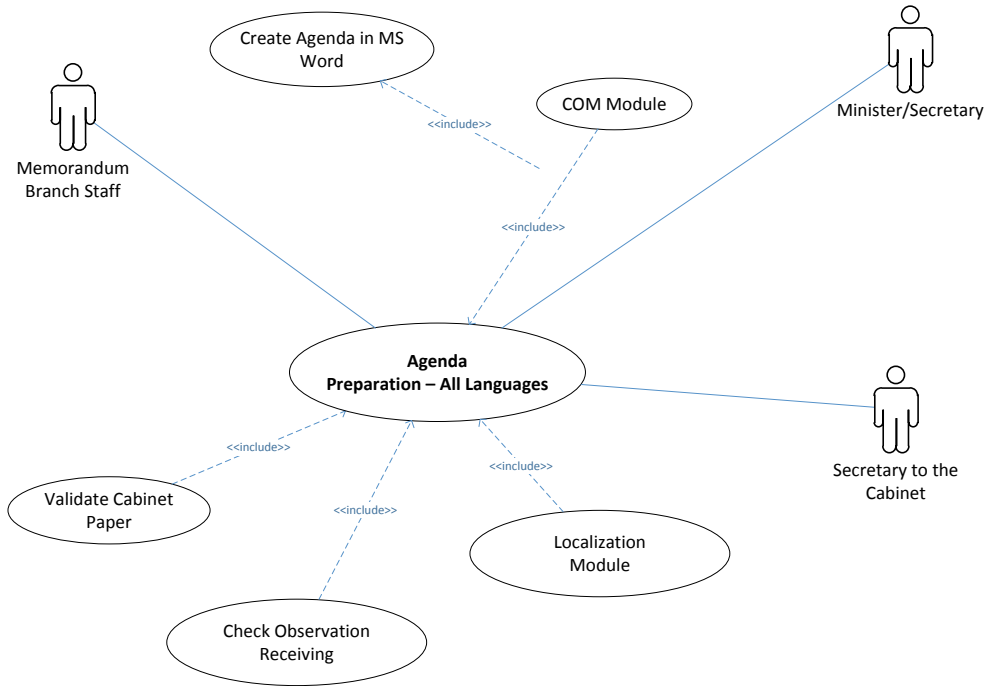
*Figure 3 Observation Preparation Use Case*

**Reminder Call-up Use-Case :** Figure B.4 Send Reminders Use Case



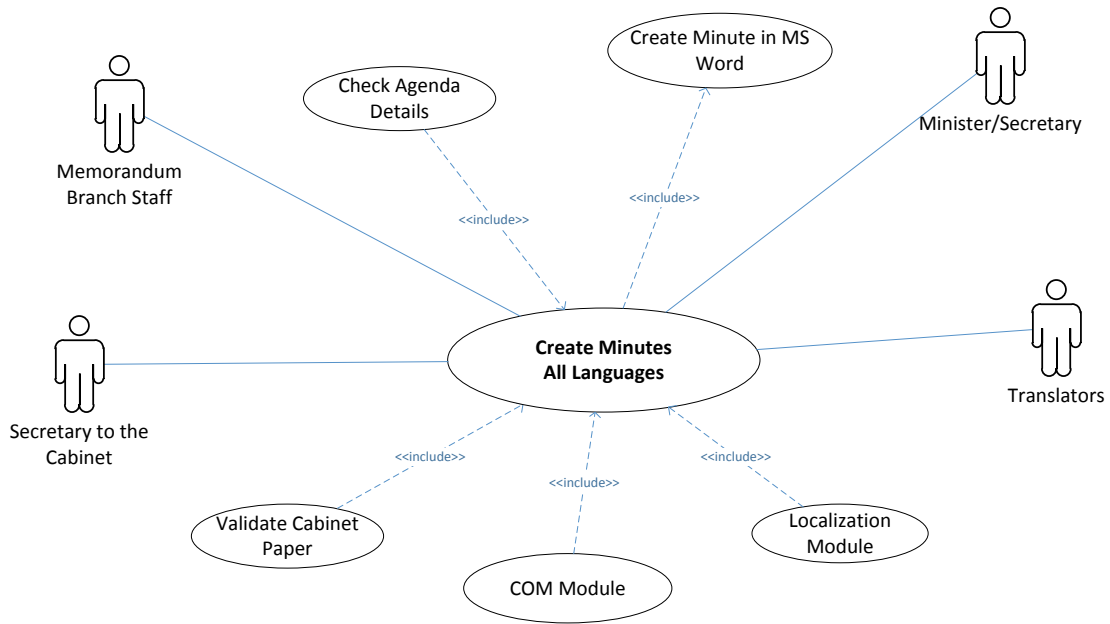
*Figure 4 Sent Reminder Use Case*

**Preparation of Agenda Use-Case: Figure B.5 Agenda/Draft Minute Preparation Use Case**



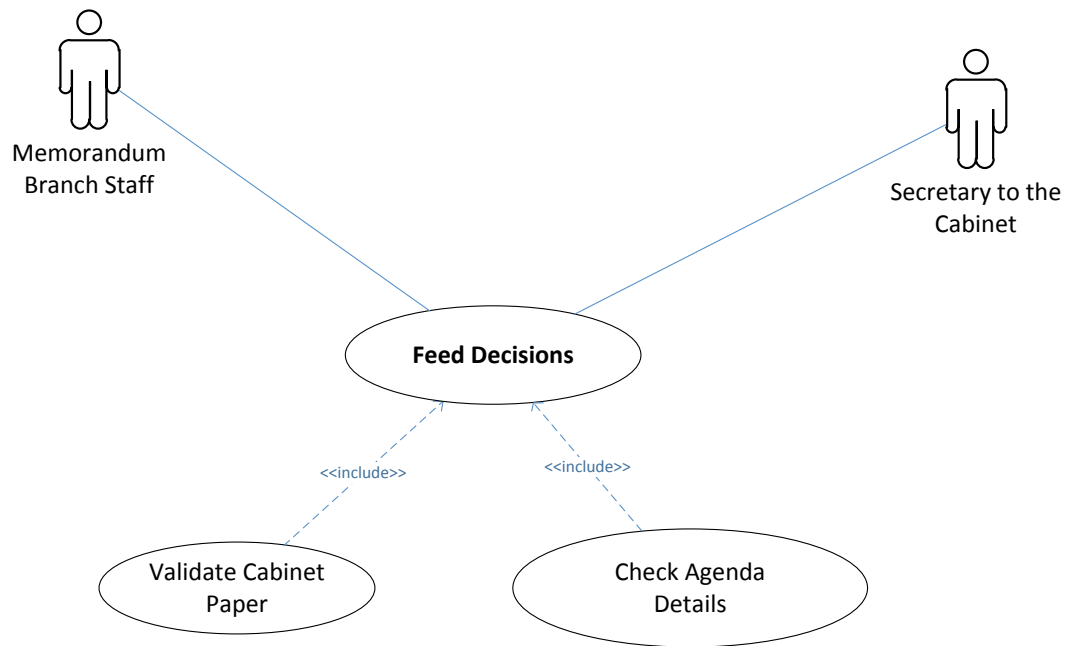
*Figure 5 Agenda Preparation*

**Create Minutes Use-Case : Figure B.6 Create Minutes Use Case**



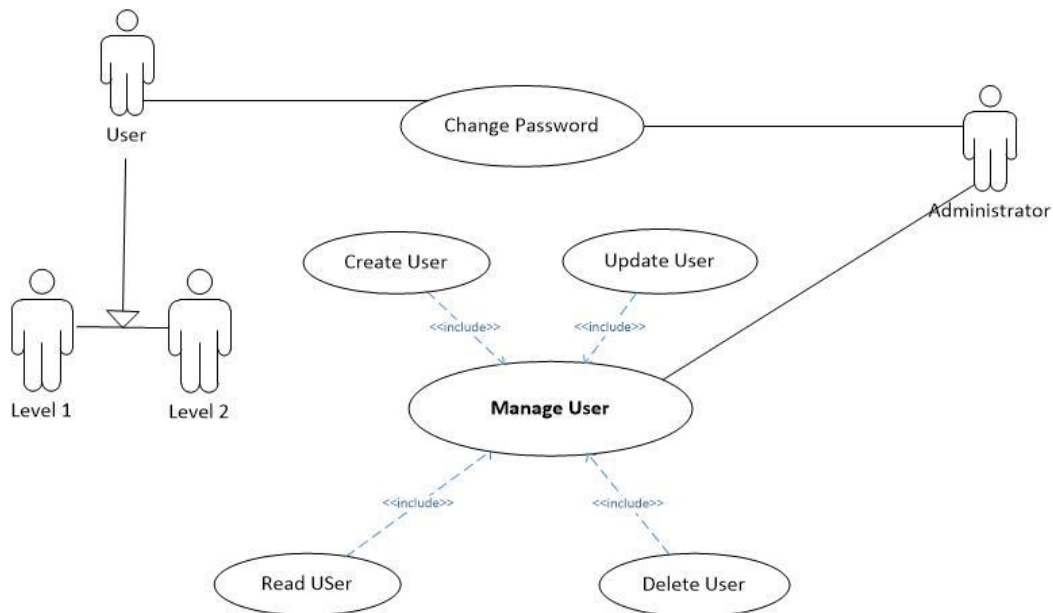
*Figure 6 Create Minutes*

**Feed Cabinet Decisions Use Case: Figure B.7 Feed Decision back into the system Use Case**



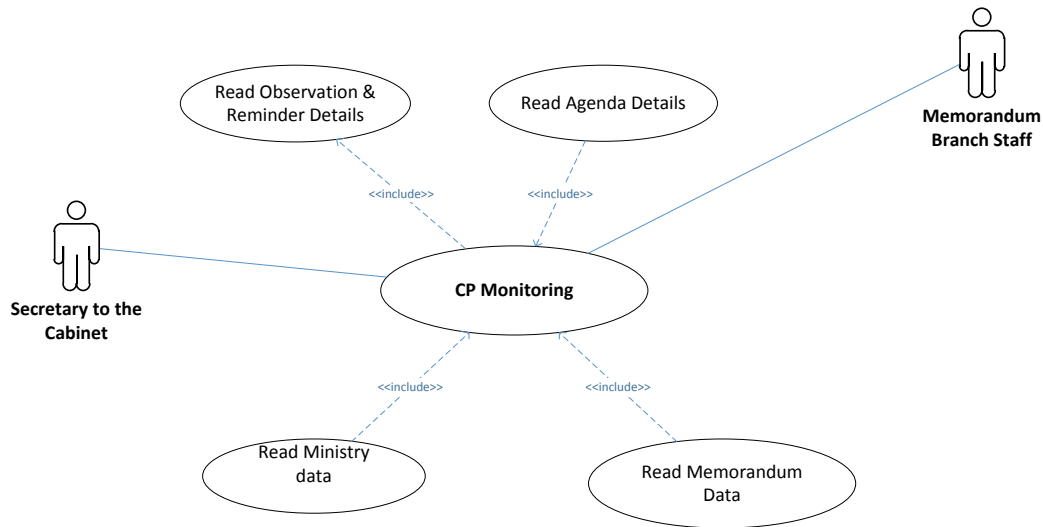
*Figure 7 Feed Decisions Use Case*

**Manage Users Use-Case : Figure B.8 Manage Users Use Case**



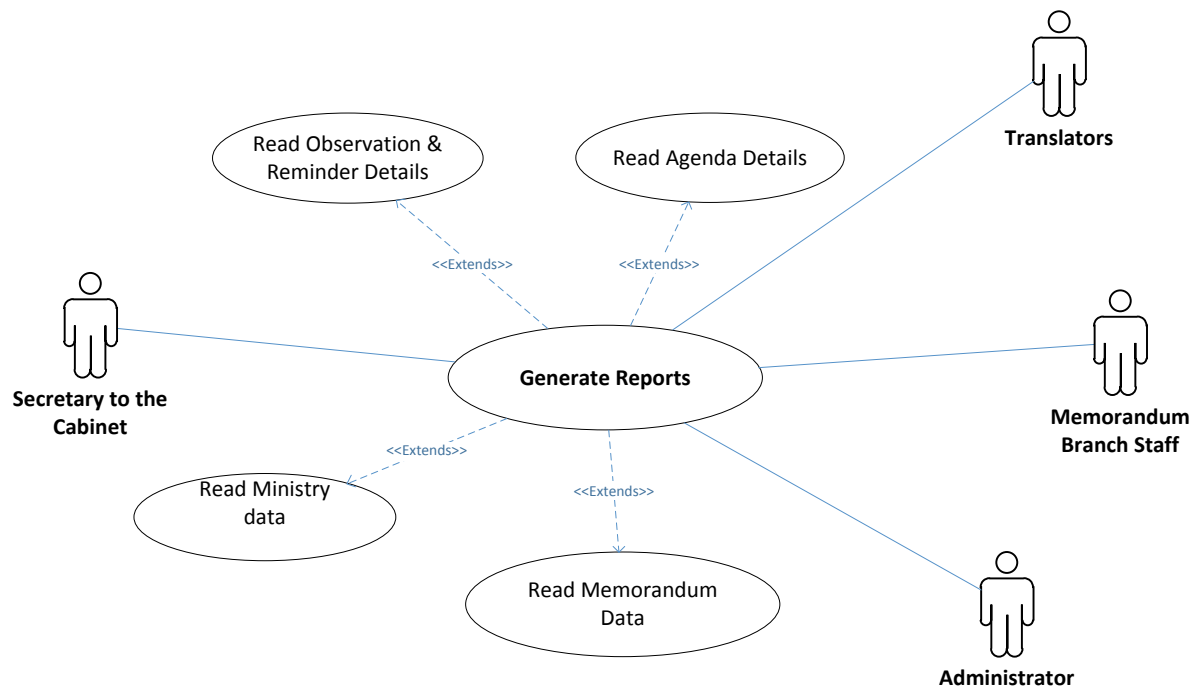
*Figure 8 Manage Users*

**CP Monitoring Module Use-Case : Figure B.9 CP Monitoring**



*Figure 9 CP Monitoring Use Case*

**Reports Generation Module Use-Case : Figure B.10 Reports Generation**



*Figure 10 Reports Generation*

**Use Case Descriptions:** Summary of Use Case Description are given in the Table B.1

<b>Use Case Name</b>	<b>Use Case Description</b>	<b>Participating Actors and Roles</b>
<b>Login/Logout</b>	This use case describes the event of login to the system and logout of the system	<ul style="list-style-type: none"> <li>• Memorandum branch staff(Primary Actor)</li> <li>• Administrator, Secretary, Word Processing branch (Secondary Actors)</li> </ul>
<b>Cabinet Paper Registration</b>	This use case describes the process of registering received cabinet papers to the system in all three languages and scans them.	<ul style="list-style-type: none"> <li>• Receiving officer, Memorandum branch (Primary Actor)</li> </ul>
<b>Observation Preparation</b>	This use case describes the process of observation preparation letters to be sent to get the honorable ministers views on the matter.	<ul style="list-style-type: none"> <li>• Memorandum branch (Primary Actor)</li> <li>• Authorized officer(Secondary Actor)</li> </ul>
<b>Remainder Preparation</b>	This use case describes the process of preparation observation reminders for those that did not receive observations within given period.	<ul style="list-style-type: none"> <li>• Memorandum branch (Primary Actor)</li> <li>• Authorized officer(Secondary Actor)</li> </ul>
<b>Agenda Preparation (All three Languages)</b>	This use case describes the process of preparation of agendas for upcoming meetings (Main, sub-committee).	<ul style="list-style-type: none"> <li>• Memorandum branch (Primary Actor)</li> <li>• Secretary (Secondary Actor)</li> </ul>
<b>Draft minute creation (All three languages)</b>	This use case describes the process of creation draft minutes based on agenda generated from the system according to the minute template	<ul style="list-style-type: none"> <li>• Memorandum branch (Primary Actor)</li> <li>• Word Processing Branch, Translators, Staff Officers, Secretary (Secondary Actor)</li> </ul>
<b>Decisions Feeding</b>	This use case describes the process of feeding back of decisions taken at the ministries once they are confirmed (following week).	<ul style="list-style-type: none"> <li>• Staff Officers (Primary Actor)</li> <li>• Translators, Memorandum branch, Secretary (Secondary Actors)</li> </ul>
<b>Manage Users</b>	This use case describes the process of create and manage all end-users of the system	<ul style="list-style-type: none"> <li>• Administrator (Primary Actor)</li> <li>• All system users(Secondary Actor)</li> </ul>

*Table B.1 Use Case Descriptions*



<b>Use Case Description Name : Login</b>	
<b>ID :</b>	1
<b>Priority :</b>	High
<b>Actors</b>	Primary: Memorandum Branch Staff; Secondary: Word Processing Staff, Translators, Administrator.
<b>Preconditions:</b>	User account must be created.
<b>Flow of event</b>	1. The users insert their username and password. 2. If their username and password has been recorded in database, the user can access the system; else the user is given an error message and re-prompt login screen.
<b>Post conditions:</b>	1.The users login in the system

*Table B.2 Login Use Case Description*

<b>Use Case Description Name: Logout</b>	
<b>ID :</b>	2
<b>Priority :</b>	High
<b>Actors</b>	Primary: Memorandum Branch Staff; Secondary: Word Processing Staff, Translators, Administrator.
<b>Preconditions:</b>	User must be logged in.
<b>Flow of event</b>	1. After finishing access the system, the users will logout the system. It is more for the security. 2. After Log out, the users will be re-prompted login screen..
<b>Post conditions:</b>	1.The users login screen is prompted.

*Table B.3 Logout Use Case Description*

<b>Use Case Description Name : Registration of CP</b>	
<b>ID :</b>	3
<b>Priority :</b>	High
<b>Actors</b>	Primary : Memorandum Branch Staff, Senior Assistant Secretary; Secondary : Secretary/Minister
<b>Preconditions:</b>	1. User must be logged in. 2. Cabinet paper must be verified.
<b>Flow of event</b>	1. User opens CP Registration form form. 2. Change Language. 3. Fills ups all compulsory fields if preferred. 4. Clicks on Save button
<b>Post conditions:</b>	1. User saves CP in the system.

*Table B.4 CP Registration Use Case Description*

<b>Use Case Description Name : Observation Preparation</b>	
<b>ID :</b>	4
<b>Priority :</b>	High
<b>Actors</b>	Primary : Memorandum Branch Staff, Senior Assistant Secretary; Secondary : Secretary/Minister
<b>Preconditions:</b>	3. User must be logged in. 4. Cabinet paper must be registered and validated in the system
<b>Flow of event</b>	5. User opens observation form. 6. Fills ups observation calling information and assign authorized officer signature. 7. Clicks on generate button
<b>Post conditions:</b>	2. User saves observation in the system. 3. Observation calling letter is sent to the honorable minister by post/fax.

*Table B.5 Use Case Description Observation Preparation*

<b>Use Case Description Name : Send Reminders</b>	
<b>ID :</b>	5
<b>Priority :</b>	Moderate
<b>Actors</b>	Primary : Memorandum Branch Staff, Senior Assistant Secretary; Secondary : Secretary/Minister
<b>Preconditions:</b>	<ol style="list-style-type: none"> <li>1. User must be logged in.</li> <li>2. Cabinet paper must be registered and validated in the system.</li> <li>3. Observation letter has been sent.</li> </ol>
<b>Flow of event</b>	<ol style="list-style-type: none"> <li>1. User opens remainder form.</li> <li>2. Read Observation calling information and from the system and assign authorized officer signature.</li> <li>3. Clicks on generate button</li> </ol>
<b>Post conditions:</b>	<ol style="list-style-type: none"> <li>1. User saves observation in the system.</li> <li>2. Remainder letter is sent to the honorable minister by post/fax.</li> </ol>

*Table B.6 Send Reminders Use Case Description*

<b>Use Case Description Name : Agenda Preparation</b>	
<b>ID :</b>	6
<b>Priority :</b>	High
<b>Actors</b>	Primary : Memorandum Branch Staff, Senior Assistant Secretary; Secondary : Secretary/Minister
<b>Preconditions:</b>	<ol style="list-style-type: none"> <li>1. User must be logged in.</li> <li>2. Cabinet papers must be registered and validated in the system.</li> <li>3. Observations of honorable ministers must have been received.</li> </ol>
<b>Flow of event</b>	<ol style="list-style-type: none"> <li>1. User opens create agenda form.</li> <li>2. Enter Cabinet papers serial numbers which are scheduled to be included in the up-coming meeting.</li> <li>3. Read Observation letters.</li> <li>4. Clicks on generate button.</li> </ol>
<b>Post conditions:</b>	<ol style="list-style-type: none"> <li>1. User saves agenda information in the system.</li> <li>2. Agenda is ready for further actions.</li> </ol>

*Table B.7 Agenda Preparation Use Case Description*

<b>Use Case Description Name : CP Registration</b>	
<b>ID :</b>	7
<b>Priority :</b>	High
<b>Actors</b>	Primary : Memorandum Branch Staff, Senior Assistant Secretary; Secondary : Secretary to the Cabinet
<b>Preconditions:</b>	<ol style="list-style-type: none"> <li>1. User must be logged in.</li> <li>2. Cabinet papers must be received according to the stipulated guidelines by the stakeholder.</li> <li>3. 3 version of CP must be received.</li> </ol>
<b>Flow of event</b>	<ol style="list-style-type: none"> <li>1. User opens CP Registration form.</li> <li>2. Enter compulsory data fields including CP Number.</li> <li>3. System runs the validates routine.</li> <li>4. User scans CP to9 the system</li> </ol>
<b>Post conditions:</b>	<ol style="list-style-type: none"> <li>1. User saves CP information in the system.</li> <li>2. CP is ready for process further.</li> </ol>

*Table B.8 CP Registration – Use Case Description*

## Appendix C Design Documentation

### Cabinet Paper Registration activity diagram

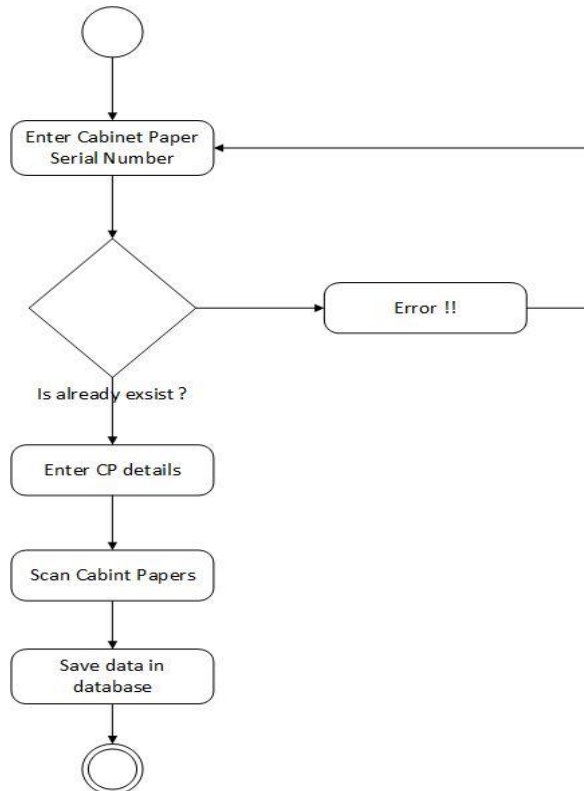


Figure 11 Activity Diagram - CP Registration

### Process Observation Preparation Activity Diagram

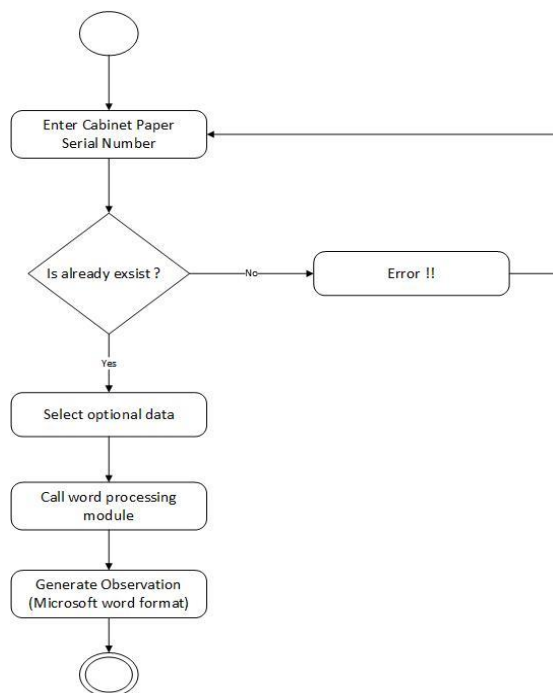


Figure 12 Activity Diagram - Observations

## Reminder alert activity diagram

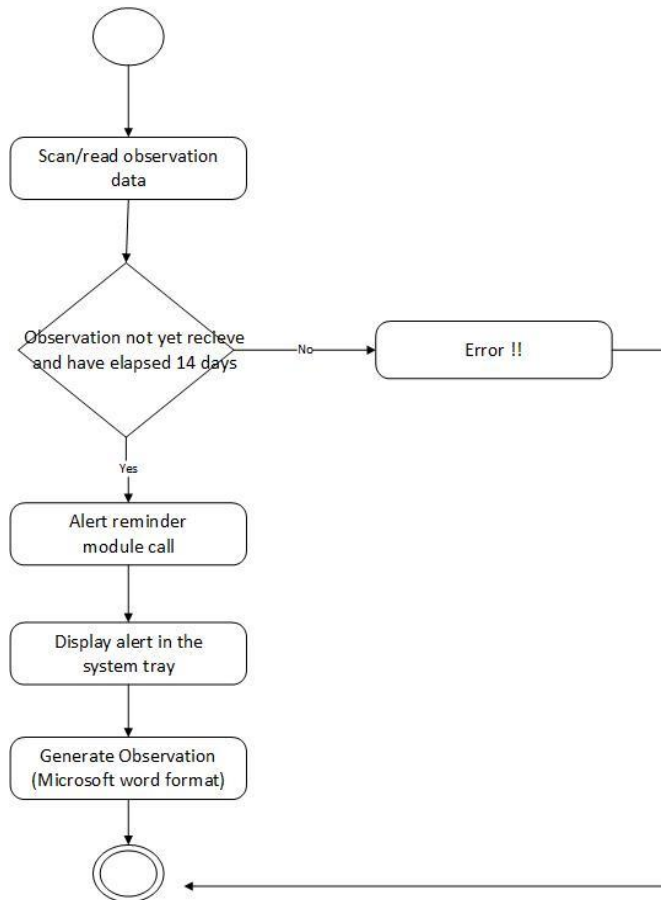


Figure 13 Activity Diagram - Generate Reminder

## Agenda Generates activity diagram

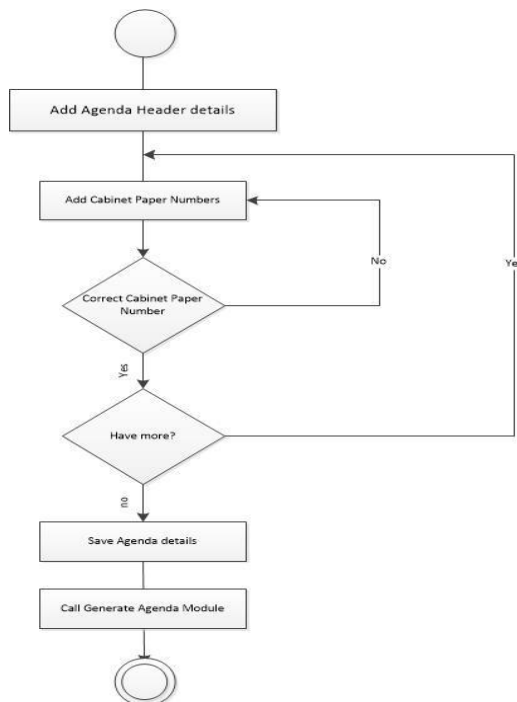


Figure 14 Activity Diagram - Agenda Preparation

## Draft Minute Preparation Activity diagram

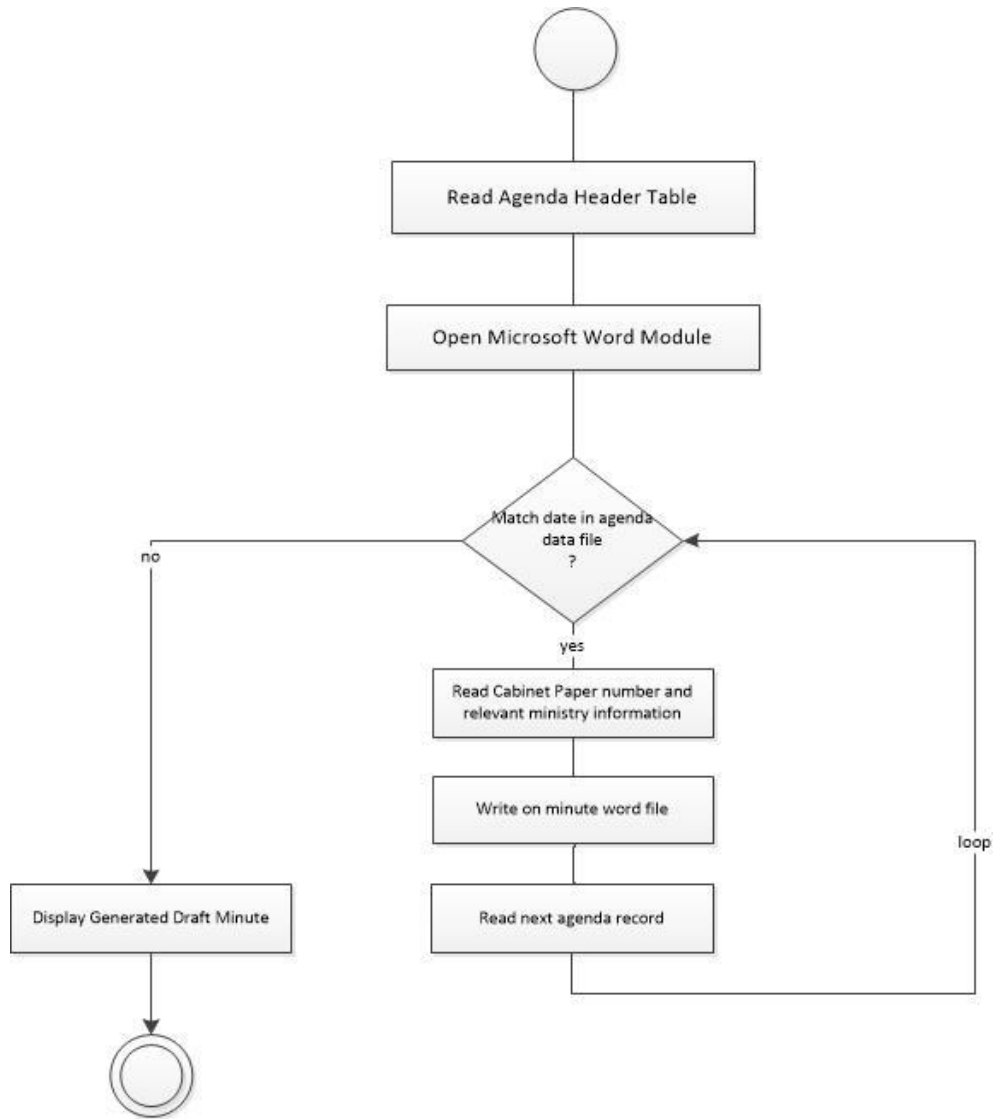


Figure 15 Activity Diagram - Draft Minute Generate

# UI Designs with Wireframes

## Dashboard

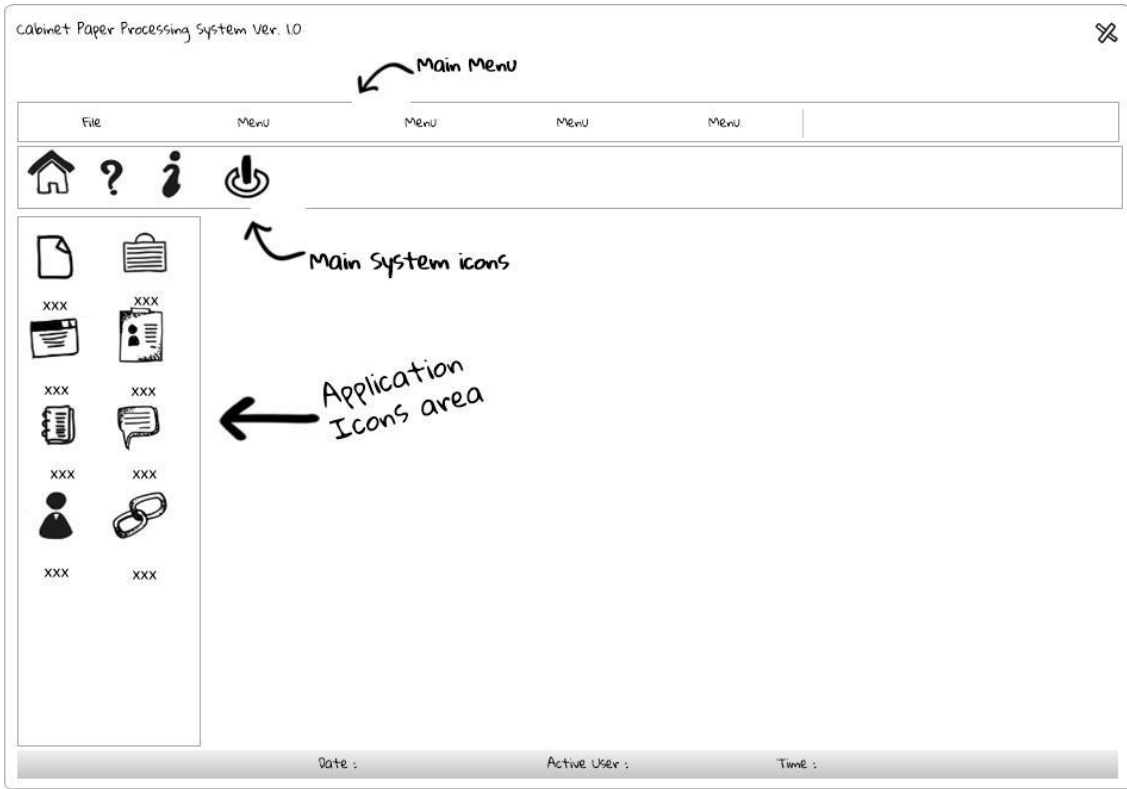


Figure 16 Wireframe - Main Form

## Cabinet Papers – Main Interface

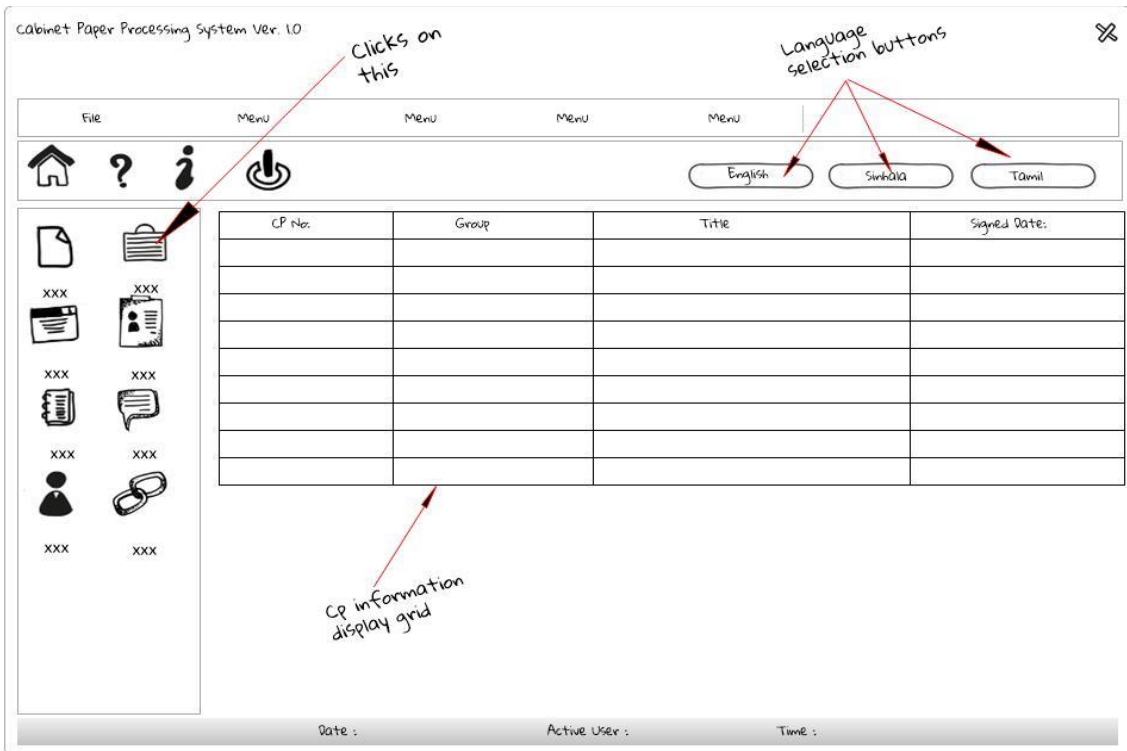


Figure 17 Wireframe - Cabinet Papers



## Cabinet Papers Adding/Update

A wireframe for a form titled "Adding Cabinet Papers". The form contains the following fields and controls:

- CP No.:** A text input field with a handwritten mask "xx/xxxxx".
- Ministry:** A dropdown menu.
- Signed by:** A dropdown menu.
- OBS No.:** A text input field.
- Signed Date:** A date picker.
- Expected Date:** A date picker.
- Remarks:** A large text area.
- Buttons:** "Save" and "Cancel" buttons at the bottom right.

Figure 18 Wireframe - Add Cabinet Papers

## Observations

A wireframe for a form titled "Adding Cabinet Papers", identical to Figure 18. It includes the same fields: CP No. (with mask "xx/xxxxx"), Ministry, Signed by, OBS No., Signed Date, Expected Date, Remarks, and Save/Cancel buttons.

Figure 19 Wireframe - Observation

## Agenda / Sub Committee Agenda

Creating an agenda

English    Sinhala    Tamil

Agenda type:

Agenda No.:  xxxxx/xx

Venue:

Agenda Date:

Time:

Sr. No.	CP No.	Paper No :	Title: --
1.			
2.			
3.			
4.			

Add row    Del row

Print    Save    Cancel

Figure 20 Wireframe – Agenda/Sub Committee Agenda

## Agenda – Supplementary

Creating an Supplementary agenda

English    Sinhala    Tamil

Agenda No.:  xxxxx/xx

Venue:

Agenda Date:

Time:

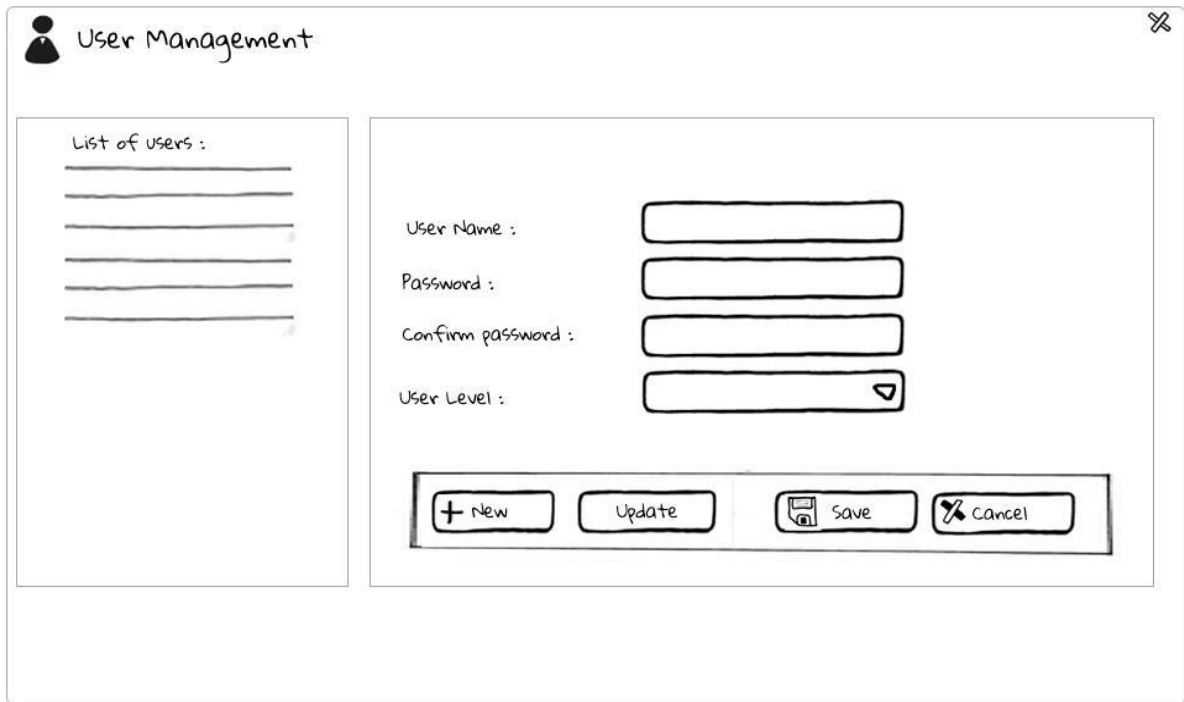
Sr. No.	CP No.	Paper No :	Title: --
1.			
2.			
3.			
4.			

Add row    Del row

Print    Save    Cancel

Figure 21 Wireframe - Supplementary Agenda

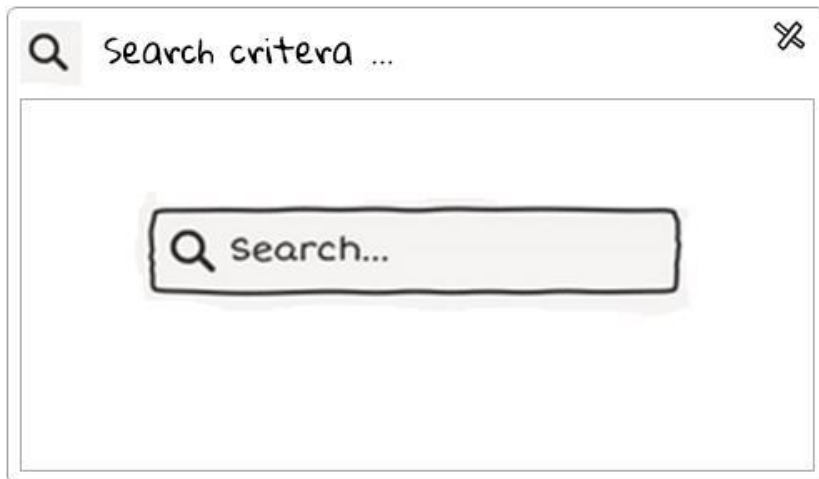
## Manage Users



A wireframe for a 'User Management' window. The window title is 'User Management' with a user icon on the left and a close icon on the right. The main area is divided into two panels. The left panel is titled 'List of users :' and contains five horizontal lines representing a list. The right panel contains a form with four fields: 'User Name :', 'Password :', 'Confirm password :', and 'User Level :'. Each field has a corresponding input box. The 'User Level' field has a dropdown arrow. Below the form is a toolbar with four buttons: '+ New', 'Update', 'Save' (with a floppy disk icon), and 'Cancel' (with a close icon).

Figure 22 Wireframe - Manage User

## Search ...



A wireframe for a 'Search criteria' window. The window title is 'Search criteria ...' with a search icon on the left and a close icon on the right. The main area contains a single search input field with a search icon and the text 'search...'.

Figure 23 Wireframe - CP Search

## Appendix D Test Cases

### Test Case ID: 2 Main Menu

#### UI Test Scenario

Sr#	Function	Expected Output	Status
1	Verify that all the labels and controls including Toolstrips, icons, and menu items Main form.		Pass
2	Verify that the font type and size of the labels and the text written on the different elements (Application and system icons) should be clearly visible and relevant colours.		Pass
3	Verify that the screen, menu, buttons, icons size, color, and other UI elements are as per the user requirements and specifications.		Pass

Table D.1 UI Test Scenario - Main Menu

#### Functional Test Scenarios for Main Menu

Sr#	Function	Expected Output	Status
4	Main form opens.	by default the application background should display state logo in the centre slightly low contrast.	Pass
5	Verify that the user is able to navigate and click/ access the different controls by pointing mouse.	Opens Relevant Form /Report	Pass
6	Verify that the ALT+ Opens Relevant Menu	Relevant Sub menu	Pass
7	Verify that CTRL+F4 close the main application.	Warning message weather to from the system	Pass
8	Verify that all the Menu Names, Icon Images, Names are according to the User the user requirements.	System Launches	Pass

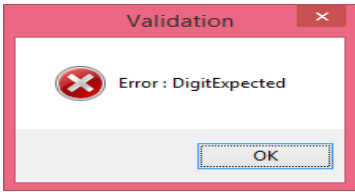
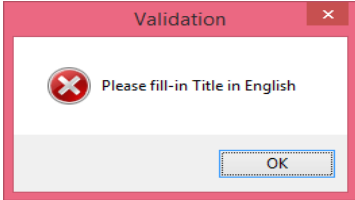
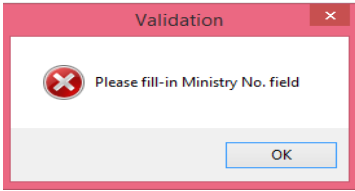
Table D.2 Functional Test Scenario - Main Menu

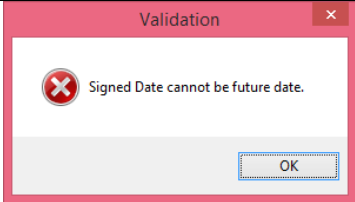
### Test Case ID:3 Adding CP

#### Pre-conditions

1. CP Number is the format of XXXX length is for 4 digits starting from 0001 to 9999.  
Input filed is masked for numeric data.
2. Ministry number is format of XXX and field is masked for numbers only length is 3.
3. Serial number is any length but first 3 should be digits. Max length is 10 characters.

#### Functional Test Scenarios for Adding CP

Sr#	Function	Expected Output	Status
1	Verify that as soon as the Add CP form opens, by default the cursor should remain on the CP No textbox.		Pass
2	Verify that the user is able to navigate or access the different controls by pressing the 'Tab' key on the keyboard.		Pass
3	Verify that CP Number field only accepts Numeric data	Warning Message 	
4	Check the behavior of form by not filling up any data into the form.	Warning message 	Failed*
5	Check the by not filling up the required fields.	Warning message 	Failed*
6	Verify that the other text input field's lengths check against the spec (titles)	Warning message	Pass
7	For Received Date, Signed Date fields Check if the form accepts the valid	Warning message	Pass

	Check Signed Date and Received Date should not be a future date		
8	Verify that the user has access to the system by entering a valid username and password and clicking the "Login" button.	System Launches	Pass
9	Verify that the user has access to the system by entering a valid username and password and pressing Enter key.	System Launches	Pass
10	Verify that the system is not accessible with an incorrect username and password.	Validation Message	Pass
<p>* Test case 4, 5 are failed since system must produce a warning message to the user about non filling of required fields. Error was occurred due to validation module's logic structure. Author rectified the source code and re-compile and test is succeeded Pass</p>			

*Table D.3 Functional Test Scenario - Add CP*

## Test Case ID:4 Observations

### Pre-conditions

1. CP Number, Ministry and Signed Officer are given as separate lists to the user by which user selects required CP Number, Ministry Name and Officer Name.
2. Sending Date and Expected Date are displayed as current date, User must change them as required
3. Observation number is automatically appended once user selects CP Number.
4. Microsoft word application is opened and minimize in the tray.

### Functional Test Scenarios for Observation

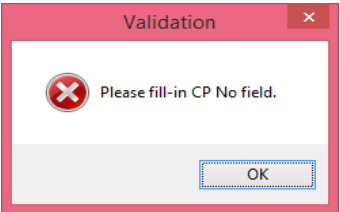
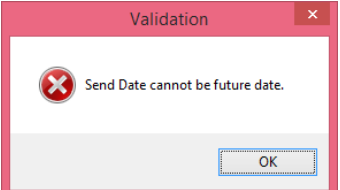
Sr#	Function	Expected Output	Status
1	Verify that as soon as the Observation form opens, by default the cursor should remain on the CP No textbox.		Pass
2	Verify that the user is able to navigate or access the different controls by pressing the 'Tab' key on the keyboard.		Pass
3	Check the by not filling up the required fields.	Warning message 	Pass
4	For Send Date and Expected Date fields; Check if the form accepts the valid dates	Warning message 	Pass

Table D.4 Functional Test Scenarios - Observation

## Test Case ID:5 Update Observations

### Pre-conditions

1. Observation Number is already in the list view window.
2. User selects/highlights the required record.
3. Clicks on Update button.

### Functional Test Scenarios for Update Observation

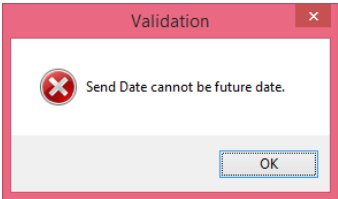
Sr#	Function	Expected Output	Status
1	Verify that as soon as the Update Observation form opens, by default the cursor should remain on the CP No textbox.		Pass
2	Verify that the user is able to navigate or access the different controls by pressing the 'Tab' key on the keyboard.		Pass
3	For Send Date and Expected Date fields; Check if the form accepts the valid dates	Warning message 	Pass

Table D.5 Functional Test Scenarios - Update Observation

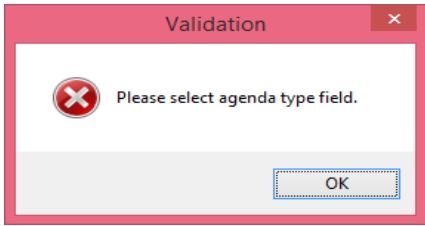
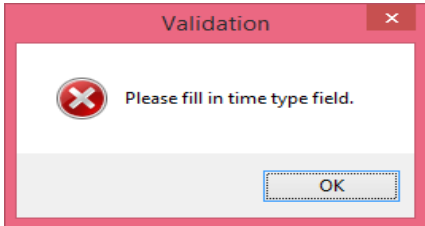
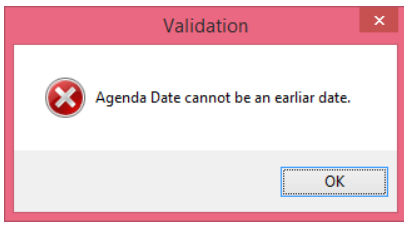


## Test Case ID:6 Agenda Preparation

### Pre-conditions

1. Upon selection from agendatype combo box Agenda number is automatically filled by the system accordingly.

### Functional Test Scenarios for Agenda Preparation

Sr#	Function	Expected Output	Status
1	Verify that as soon as the Agenda form opens, by default the cursor should remain on the Agenda Type Combo Box.		Pass
2	Verify that the user is able to navigate or access the different controls by pressing the 'Tab' key on the keyboard.		Pass
3	Verify that Agenda No: automatically assigned upon selection agenda date		Pass
4	Check the behavior of form by not filling up any data into the form.	Warning message 	Failed*
5	Check the by not filling up the required fields.	Warning message 	Failed*
6	Verify that the other text input field's lengths check against the spec (titles)	Warning message	Pass
7	For Agenda Date; Check if the form accepts the valid date  Check Agenda Date should not be an earlier date	Warning message 	Pass

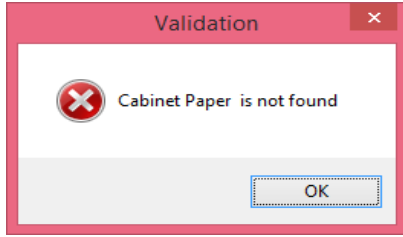
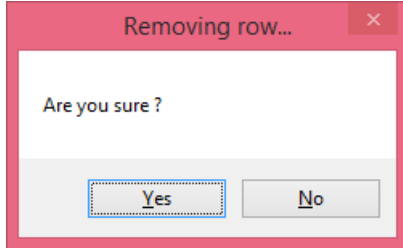
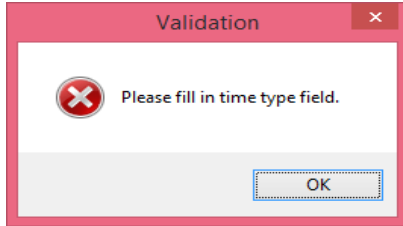
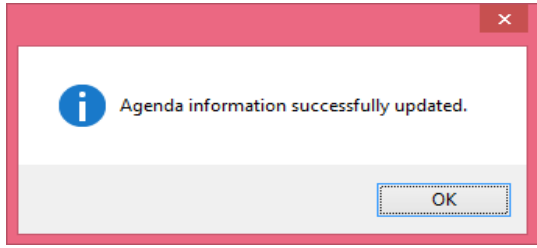
8	Verify that serial number is auto generated in the CP list grid. First serial number is started with 04.		Pass
9	Verify that user enters CP Number in second column.	Warning Message 	Pass
10	Verify that Remove Row should give delete confirmation	Warning Message 	*Failed
11	Save Button without filling all required data fields	Warning Message 	
12	Save Button without filling all required data fields	Successful Message 	
<p>* Test case 10 is failed since system must produce a warning message to the user about confirmation for removing a row, but failed to pops up the warning message instead remove at once. Error was occurred due to a logical error in the code. Author rectified the source code and re-compile and test is succeeded. Pass</p>			

Table D.6 Functional Test Scenario - Agenda

## Test Case ID: 7 Cabinet Paper Modules

### Pre-conditions

- Upon clicks Cabinet Papers Application Icon system opens Cabinet Paper as a grid.

### Functional Test Scenarios for Cabinet Paper Module

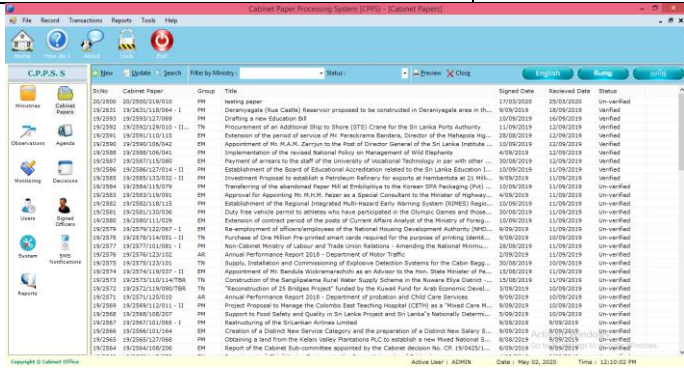
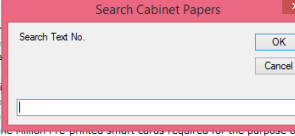
Sr#	Function	Expected Output	Status																																																																																																																																												
1	Verify that as soon as the Agenda form opens, by default the grid displays list of Cabinet papers in English Version most recent received at the top row.		Pass																																																																																																																																												
2	Verify that the grid has outline border and Column Heading are bold face and font size is 1px larger than text.		Pass																																																																																																																																												
3	Verify that pagination works with Page up, Page down keys and up, down, left, right keys.		Pass																																																																																																																																												
4	Check the combo box has typeahead functionality		Pass																																																																																																																																												
8	Horizontal/vertical scroll bars		Pass																																																																																																																																												
9	Language buttons access relevant language list of data		Pass																																																																																																																																												
10	Search Button pops up search dialog box	 <table border="1"> <thead> <tr> <th>SrNo</th> <th>Cabinet Paper</th> <th>Group</th> <th>Title</th> <th>Signed Date</th> <th>Received Date</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td>2592/129/010 - I</td> <td>...</td> <td>IN</td> <td>Procurement of an Additional Ship to shore (STS) Crane for the Sri Lanka Ports Authority</td> <td>11/09/2019</td> <td>11/09/2019</td> <td>1</td> </tr> <tr> <td>2591/110/115</td> <td>...</td> <td>EM</td> <td>Extension of the period of service of Mr. Parakkrama Bandara, Director of the Mahapala Hig...</td> <td>28/08/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2590/106/042</td> <td>...</td> <td>EM</td> <td>Appointment of Mr. M.A.M. Zarrjun to the Post of Director General of the Sri Lanka Institute ...</td> <td>10/09/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2588/106/041</td> <td>...</td> <td>PM</td> <td>Implementation of the revised National Policy on Management of Wild Elephants</td> <td>4/09/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2586/127/014 - II</td> <td>...</td> <td>PM</td> <td>Establishment of the Board of Educational Accreditation related to the Sri Lanka Education ...</td> <td>11/09/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2585/133/032 - II</td> <td>...</td> <td>PM</td> <td>Investment Pr...</td> <td>9/09/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2584/115/079</td> <td>...</td> <td>PM</td> <td>Transferring o...</td> <td>10/09/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2583/119/091</td> <td>...</td> <td>EM</td> <td>Approval for A...</td> <td>4/09/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2582/118/115</td> <td>...</td> <td>PM</td> <td>Establishment</td> <td>10/09/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2581/130/036</td> <td>...</td> <td>PM</td> <td>Duty free vehi...</td> <td>30/08/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2580/111/029</td> <td>...</td> <td>EM</td> <td>Extension of c...</td> <td>10/09/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2579/122/067 - I</td> <td>...</td> <td>EM</td> <td>Re-employe...</td> <td>9/09/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2578/114/051 - II</td> <td>...</td> <td>TN</td> <td>Purchase of O...</td> <td>9/09/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2577/101/081 - I</td> <td>...</td> <td>PM</td> <td>Non-Cabinet Ministry of Labour and Trade Union Relations - Amending the National Minimu...</td> <td>2/08/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2576/123/102</td> <td>...</td> <td>AR</td> <td>Annual Performance Report 2018 - Department of Motor Traffic</td> <td>2/09/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2575/123/101</td> <td>...</td> <td>TN</td> <td>Supply, Installation and Commissioning of Explosive Detection Systems for the Cabin Bagg...</td> <td>30/08/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2574/119/037 - II</td> <td>...</td> <td>EM</td> <td>Appointment of Mr. Bandula Wickramarachchi as an Advisor to the Hon. State Minister of Fe...</td> <td>15/08/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2573/110/114/TBR</td> <td>...</td> <td>TN</td> <td>Construction of the Sanglipalama Rural Water Supply Scheme in the Nuwara Elya District ...</td> <td>15/08/2019</td> <td>10/09/2019</td> <td>1</td> </tr> <tr> <td>2572/119/090/TBR</td> <td>...</td> <td>TN</td> <td>"Reconstruction of 25 Bridges Project" funded by the Kuwait Fund for Arab Economic Devel...</td> <td>3/09/2019</td> <td>10/09/2019</td> <td>1</td> </tr> </tbody> </table>	SrNo	Cabinet Paper	Group	Title	Signed Date	Received Date	Status	2592/129/010 - I	...	IN	Procurement of an Additional Ship to shore (STS) Crane for the Sri Lanka Ports Authority	11/09/2019	11/09/2019	1	2591/110/115	...	EM	Extension of the period of service of Mr. Parakkrama Bandara, Director of the Mahapala Hig...	28/08/2019	10/09/2019	1	2590/106/042	...	EM	Appointment of Mr. M.A.M. Zarrjun to the Post of Director General of the Sri Lanka Institute ...	10/09/2019	10/09/2019	1	2588/106/041	...	PM	Implementation of the revised National Policy on Management of Wild Elephants	4/09/2019	10/09/2019	1	2586/127/014 - II	...	PM	Establishment of the Board of Educational Accreditation related to the Sri Lanka Education ...	11/09/2019	10/09/2019	1	2585/133/032 - II	...	PM	Investment Pr...	9/09/2019	10/09/2019	1	2584/115/079	...	PM	Transferring o...	10/09/2019	10/09/2019	1	2583/119/091	...	EM	Approval for A...	4/09/2019	10/09/2019	1	2582/118/115	...	PM	Establishment	10/09/2019	10/09/2019	1	2581/130/036	...	PM	Duty free vehi...	30/08/2019	10/09/2019	1	2580/111/029	...	EM	Extension of c...	10/09/2019	10/09/2019	1	2579/122/067 - I	...	EM	Re-employe...	9/09/2019	10/09/2019	1	2578/114/051 - II	...	TN	Purchase of O...	9/09/2019	10/09/2019	1	2577/101/081 - I	...	PM	Non-Cabinet Ministry of Labour and Trade Union Relations - Amending the National Minimu...	2/08/2019	10/09/2019	1	2576/123/102	...	AR	Annual Performance Report 2018 - Department of Motor Traffic	2/09/2019	10/09/2019	1	2575/123/101	...	TN	Supply, Installation and Commissioning of Explosive Detection Systems for the Cabin Bagg...	30/08/2019	10/09/2019	1	2574/119/037 - II	...	EM	Appointment of Mr. Bandula Wickramarachchi as an Advisor to the Hon. State Minister of Fe...	15/08/2019	10/09/2019	1	2573/110/114/TBR	...	TN	Construction of the Sanglipalama Rural Water Supply Scheme in the Nuwara Elya District ...	15/08/2019	10/09/2019	1	2572/119/090/TBR	...	TN	"Reconstruction of 25 Bridges Project" funded by the Kuwait Fund for Arab Economic Devel...	3/09/2019	10/09/2019	1	Pass
SrNo	Cabinet Paper	Group	Title	Signed Date	Received Date	Status																																																																																																																																									
2592/129/010 - I	...	IN	Procurement of an Additional Ship to shore (STS) Crane for the Sri Lanka Ports Authority	11/09/2019	11/09/2019	1																																																																																																																																									
2591/110/115	...	EM	Extension of the period of service of Mr. Parakkrama Bandara, Director of the Mahapala Hig...	28/08/2019	10/09/2019	1																																																																																																																																									
2590/106/042	...	EM	Appointment of Mr. M.A.M. Zarrjun to the Post of Director General of the Sri Lanka Institute ...	10/09/2019	10/09/2019	1																																																																																																																																									
2588/106/041	...	PM	Implementation of the revised National Policy on Management of Wild Elephants	4/09/2019	10/09/2019	1																																																																																																																																									
2586/127/014 - II	...	PM	Establishment of the Board of Educational Accreditation related to the Sri Lanka Education ...	11/09/2019	10/09/2019	1																																																																																																																																									
2585/133/032 - II	...	PM	Investment Pr...	9/09/2019	10/09/2019	1																																																																																																																																									
2584/115/079	...	PM	Transferring o...	10/09/2019	10/09/2019	1																																																																																																																																									
2583/119/091	...	EM	Approval for A...	4/09/2019	10/09/2019	1																																																																																																																																									
2582/118/115	...	PM	Establishment	10/09/2019	10/09/2019	1																																																																																																																																									
2581/130/036	...	PM	Duty free vehi...	30/08/2019	10/09/2019	1																																																																																																																																									
2580/111/029	...	EM	Extension of c...	10/09/2019	10/09/2019	1																																																																																																																																									
2579/122/067 - I	...	EM	Re-employe...	9/09/2019	10/09/2019	1																																																																																																																																									
2578/114/051 - II	...	TN	Purchase of O...	9/09/2019	10/09/2019	1																																																																																																																																									
2577/101/081 - I	...	PM	Non-Cabinet Ministry of Labour and Trade Union Relations - Amending the National Minimu...	2/08/2019	10/09/2019	1																																																																																																																																									
2576/123/102	...	AR	Annual Performance Report 2018 - Department of Motor Traffic	2/09/2019	10/09/2019	1																																																																																																																																									
2575/123/101	...	TN	Supply, Installation and Commissioning of Explosive Detection Systems for the Cabin Bagg...	30/08/2019	10/09/2019	1																																																																																																																																									
2574/119/037 - II	...	EM	Appointment of Mr. Bandula Wickramarachchi as an Advisor to the Hon. State Minister of Fe...	15/08/2019	10/09/2019	1																																																																																																																																									
2573/110/114/TBR	...	TN	Construction of the Sanglipalama Rural Water Supply Scheme in the Nuwara Elya District ...	15/08/2019	10/09/2019	1																																																																																																																																									
2572/119/090/TBR	...	TN	"Reconstruction of 25 Bridges Project" funded by the Kuwait Fund for Arab Economic Devel...	3/09/2019	10/09/2019	1																																																																																																																																									

Table D.7 Functional Test Scenario - CP Module

# Appendix E User Documentation

## About the CPPS

The CPPS is a software tool for storing, analysing, processing, monitoring and reporting tri-lingual data related to Cabinet Memorandums in the Office of the Cabinet of Ministers.

Learning to use this software is easy. Users are advised to review this document thoroughly before make use of the CPPS software. This user guide presents an overview of the application's features and gives step-by-step guide for completing a variety of tasks.

## System requirements

Please ensure that your system needs or exceeds the following requirements before installing the CPPS application.

<b>CPU</b>	Inter family 2.0 GHz or Higher
<b>Memory</b>	1 GB (2 GB Recommended)
<b>Hard Disk</b>	100 GB
<b>OS</b>	Windows 7,8,10
<b>Connection</b>	Local Area Connection

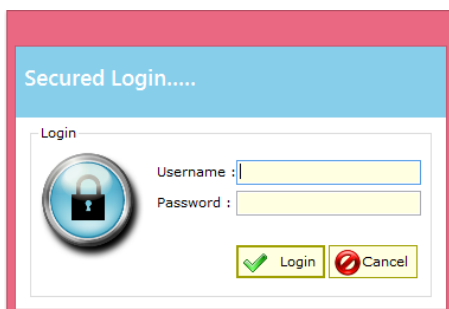
*Table E.1 System Requirements*

- To starts the CPPS application, double-click its desktop icon, or choose Start > CPPS from the Start menu. CPPS launches.



*Figure 24 CPPS*

## Login Screen



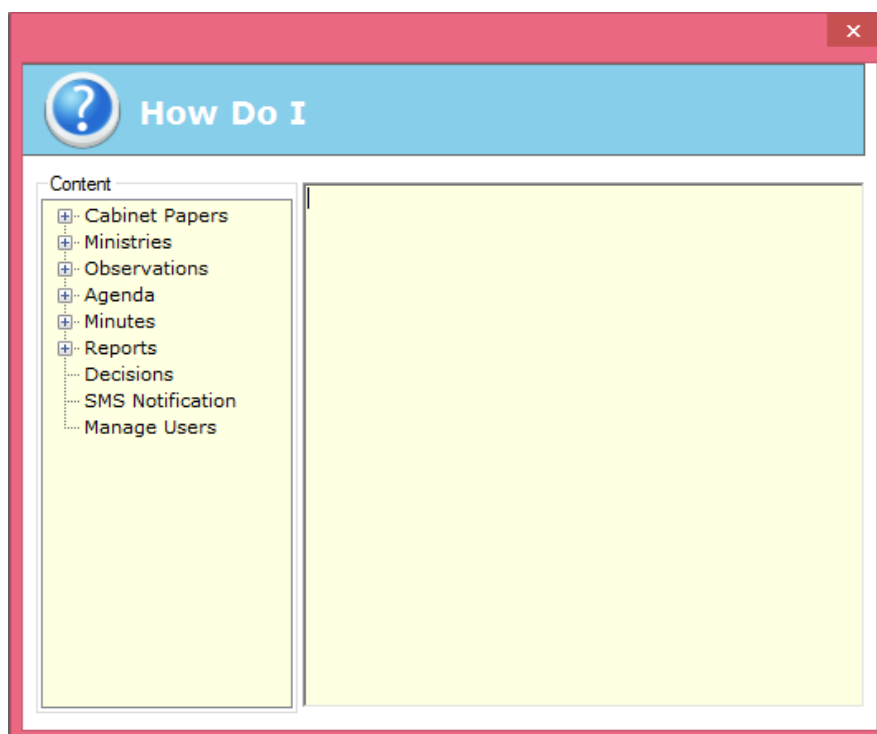
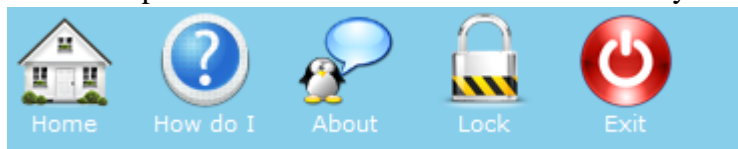
*Figure 25 Login Screen*

Users can login to the system by entering the username and password. User can change his/her password through the system at any time by accessing Fil > Change Password option. After they login to the system, there is a system tree in the left side, which displays the links to main modules user can access. The user access will be defined on user role or level, which they are mapped in the system.

### Getting additional help

CPPS has built in help facility that gives users a general idea how to do common tasks in the application.

- For technical support, please contact [infor@cpps.gov.lk](mailto:infor@cpps.gov.lk)
- Online help. Please access. How do I button in the system



## Main Dashboard

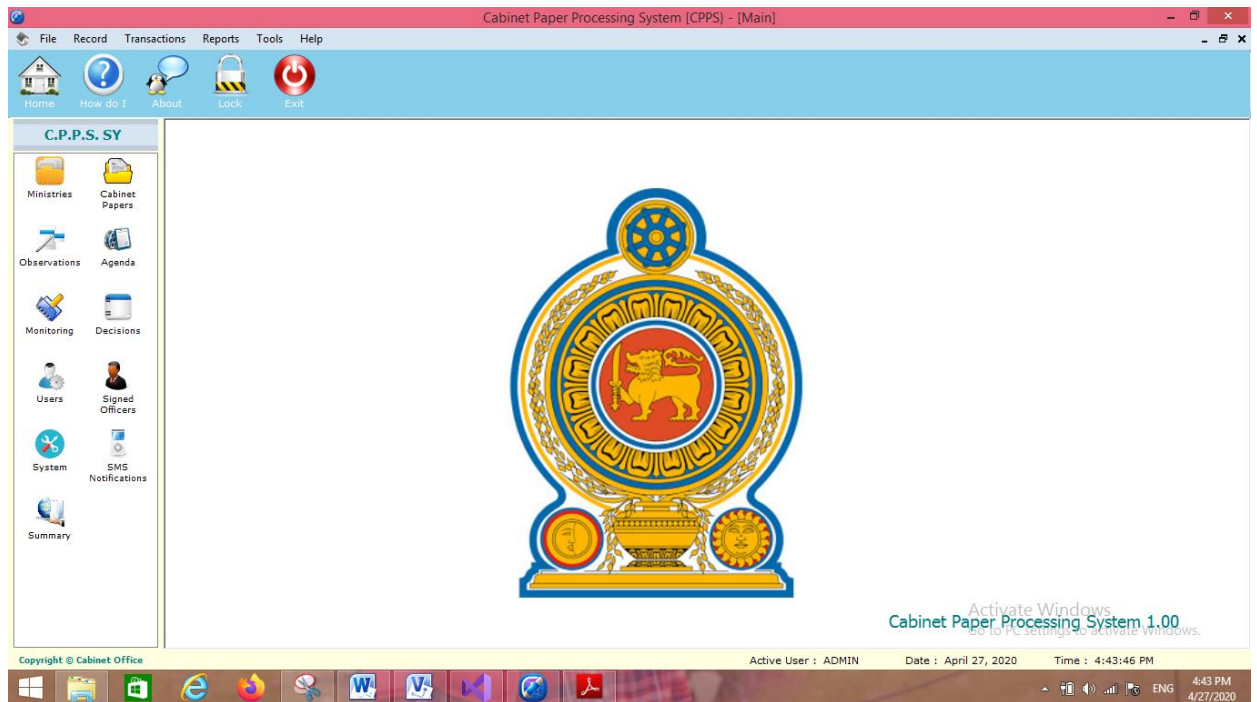


Figure 26 Main Dashboard

The system is divided into eight modules. Such as,

- Ministries : Maintain Ministry Details in all 3 languages.
- Cabinet Papers: Maintain CP Details in all 3 languages.
- Observations : Prepare, Manage Observation and Reminder Details.
- Agendas : Create/Edit Agenda and draft minutes in all 3 languages
- Decisions : Feeding back decision details to the system
- Monitoring : single click monitoring tool
- SMS Notifications : Generate SMS notification
- System : System administration module where admin can add/change users, database configuration
- Reports : users can access/print reports.

System operates on linear menu structuring pattern where each module opens relevant from with its existing data loaded in to a grid to ease of access and speed. Top of the page is toolbar consists of New, Update, Search, Refresh and Close Buttons Respectively (Figure E.4)

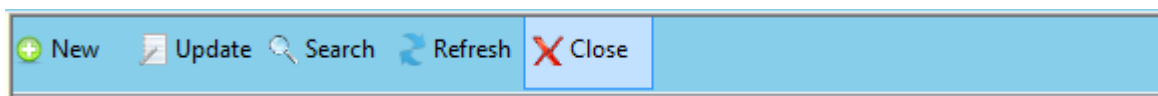


Figure 27 Toolbar

## Switching language



Figure 28 Language Selection

System makes use of a language button. User can switch the preferred language just by clicking the button.

## To Add New Ministry (Ministry Module)

Upon selecting the new Cabinet system needs to set up new Ministry Details to the system. Ministry Module maintains Ministry details in all 3 languages.

**Add Ministry** – Click on New 

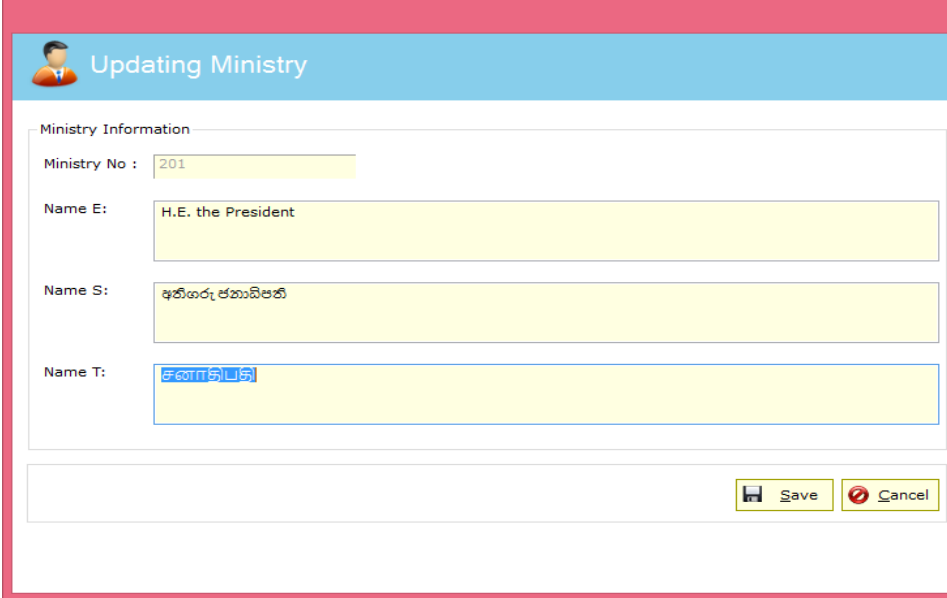
This opens up new form where user can add a new ministry

A screenshot of a web form titled 'Adding Ministry'. The form has a blue header with a user icon and the title. Below the header is a section titled 'Ministry Information' with four input fields: 'Ministry No', 'Name E', 'Name S', and 'Name T'. The 'Name T' field contains the text 'y8'. At the bottom right of the form are two buttons: 'Save' and 'Cancel'.

Figure 29 Add Ministry

## Update a ministry details

Upon clicking Ministry icon system launches list of ministries as grid. Select the ministry you want to change to and Click on Update Button. System loads Update Ministry form. Made your changes and click on save button.



The screenshot shows a web form titled "Updating Ministry". The form is enclosed in a light blue border. At the top left, there is a small icon of a person and the title "Updating Ministry". Below this, the form is divided into a section labeled "Ministry Information". This section contains three text input fields: "Ministry No : 201", "Name E: H.E. the President", and "Name S: அகிலர் அமைச்சர்". Below these fields is a "Name T:" field containing the Tamil text "சுனாதிபதி". At the bottom right of the form, there are two buttons: "Save" and "Cancel".

Figure 30 Updating a Ministry



## Add CP (Cabinet Memorandums)

This module is all about dealing with Cabinet Papers. Add, Update, Delete, Cancel, Verify, Withdraw of a Cabinet Memorandum is done here. Clicks on Cabinet Paper module icon. Clicks on required language button.

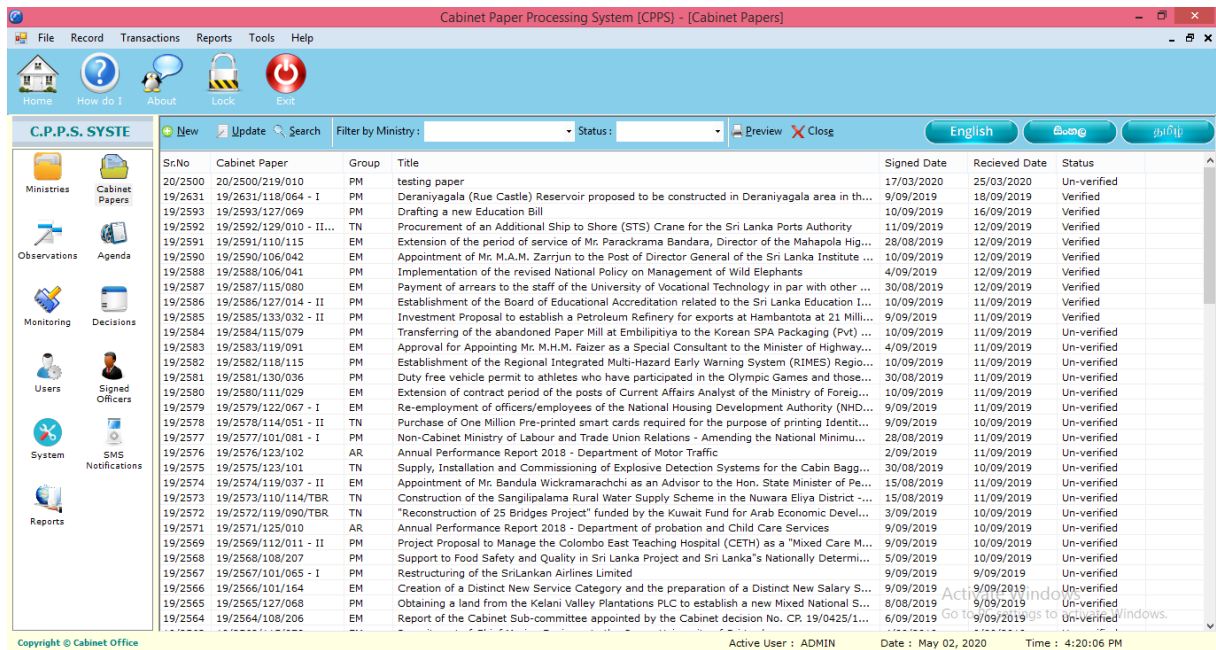


Figure 31 Cabinet Memorandum Portal

Add new Cabinet Paper – Click on 

When user wants to add the Sinhala/Tamil title user must switch language Windows + Space.

The screenshot shows the 'Adding Cabinet Papers' form. It contains the following fields and controls:

- Information** section:
- CP No:** A text input field with a placeholder 'XX/XXXX'.
- Group:** A dropdown menu.
- Ministry No:** A text input field with a folder icon.
- Secondary Min:** A text input field.
- Serial No:** A text input field with a placeholder 'XXX'.
- Title English:** A large text area.
- Title Sinhala:** A large text area.
- Title Tamil:** A large text area.
- Signed Date:** A date picker set to 3/21/2020.
- Received Date:** A date picker set to 5/ 2/2020.
- Buttons:** 'Save' and 'Cancel' buttons at the bottom right.

## Update Cabinet Paper details

Select the cabinet paper number to be edited from the list and click on Update button. Update does not allow user to edit CP Number.

**Updating Cabinet Paper**

**Information**

CP No: 19/2631    XX/XXXX

Group : PM

Ministry No: 118    Secondary Min:

**Description**

Serial No: 064 - I    XXX

Title English: Deraniyagala (Rue Castle) Reservoir proposed to be constructed in Deraniyagala area in the Kegalle District

Title Sinhala:

Title Tamil:

Signed Date : 9/ 9/2019

Received Date: 9/18/2019

Save    Cancel

Figure 32 Add new Cabinet Paper

Figure 33 Update a Cabinet Paper

## Observation Module –

This module allows user to create, update, Observation details.

OBS No	CP Number	Ministry	Send Date	Expected Date	Received Date	Remarks
B19/2494A	19/2494	204	06/04/2020	14/04/2020	Not Received	Urgent Reply
B20/0152A	20/0152	210	30/08/2019	30/08/2019	Not Received	Test

Copyright © Cabinet Office    Active User : ADMIN    Date : May 02, 2020    Time : 6:09:09 PM

Figure 34 Observation Main Module

**New observation** – clicks on



Figure 35 New Observation

User can print observation letters through this module. Just clicks on print button and system will initiate Microsoft word and generates Observation letter. System only saves raw data and does not save generated observation letter in the system. This saves data storage since system can generate observation letter at any time.

**Update observation** – click on

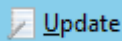


Figure 36 Update an Observation

To create an Agenda – Clicks on



icon and then

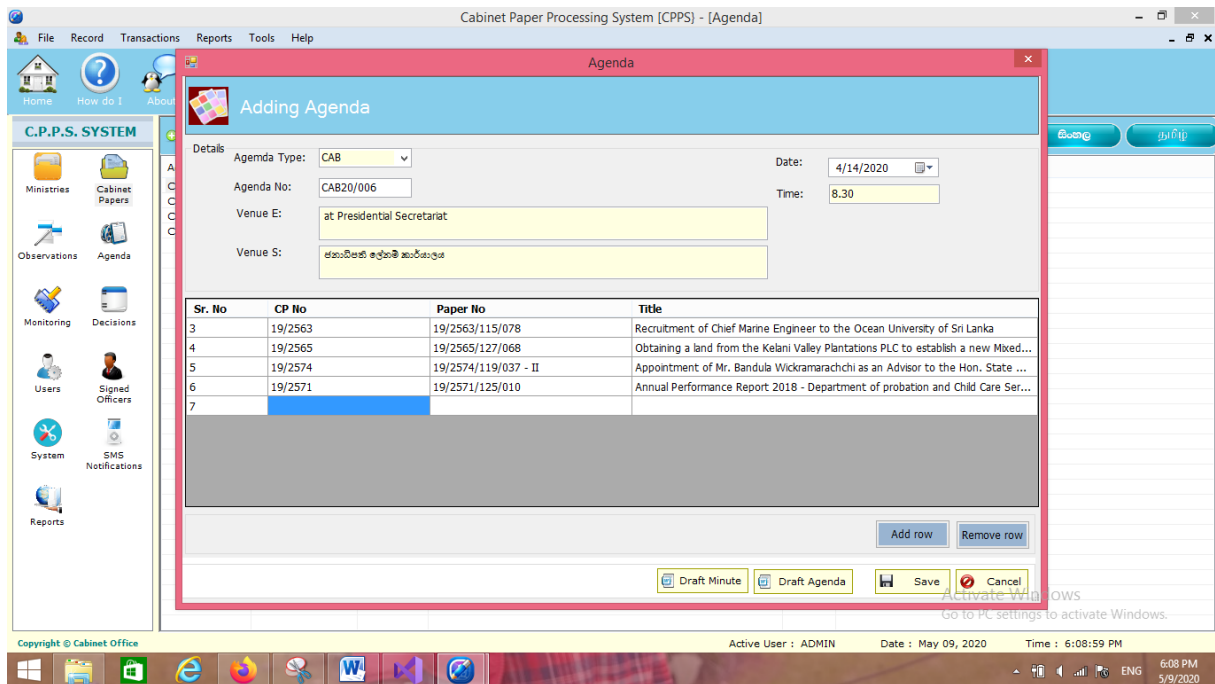


Figure 37 Creating a new Agenda

This interface allows user to enter details for creating agendas. User has to fill up compulsory fields as indicated in the form (date, time, venue, CP Numbers, language etc.) and clicks on draft agenda button to create an agenda in Microsoft Word format. Please refer Figure G.2 for sample Agenda Format. to save agenda in the system clicks on Save Button.

To update an existing agenda – clicks on

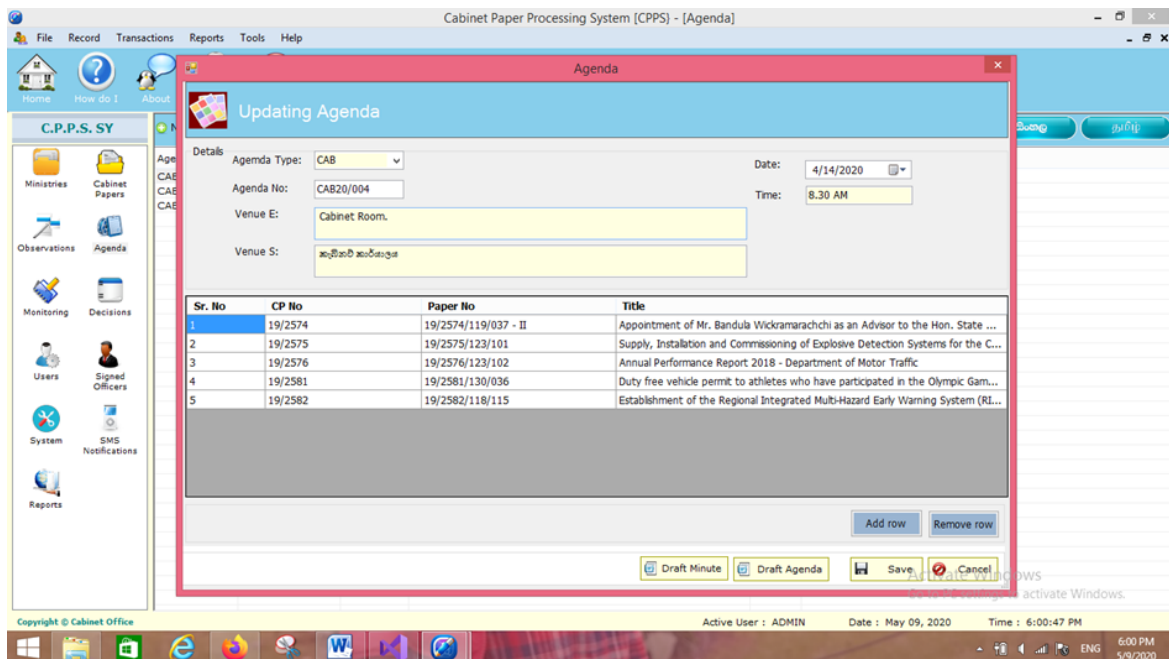
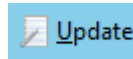


Figure 38 Updating an Agenda

Point (select) agenda number from the list and clicks on update button. System opens saved agenda for modification.

### To create a reminder

This function is a part of observation module. Reminder letters will be sent to those who have not received Observation Letters on the due date. Interface is same observation. First point(select) observation record and clicks on Update button. To print it clicks on Reminder button.

Updating Observation

Observation Information

Cabinet Paper : 20/0797

Ministry: 204

Signed by : 16

OBS No : b20/0797a

Send Date: 5/13/2020

Expected Date: 5/15/2020

Received Date: 6/18/2020

Remarks:

Make sure to open an instance of Microsoft Word. before execute Print Command.

Reminder Print Save Cancel

Figure 39 Create Reminder

### Working with Reports

Main reports can be accessed in one location. Clicks on Reports icon bring user to report parameter interface where user can select customise report options.

Reports Menu

Report: Not included in any agenda

Ministry: All

Select Date Range

From: 6/18/2020

To: 6/18/2020

Load Close

Figure 40 Report Parameters

User may select report option, ministry wise and date range

i.e List of cabinet Memorandums received from Ministry of Finance and Planning from 01/05/2020 to 31/06/2020.

Clicks on Load button brings next intermediate level where user can see the print preview of the report

# Appendix F Sample Reports

## 1. Sample System generated Agenda (Sinhala Version)

රහස්‍යකමයි

2020 ජූනි මස 3 වනදා පෙරවරු 10.30 ට  
ජනාධිපති කාර්යාලයේදී පැවැත්වීමට නියමිත  
අමාත්‍ය මණ්ඩල රැස්වීමේ න්‍යාය පත්‍රය

**(අ) රැස්වීම ආරම්භය**

01. ආගම සිහි කිරීම.
02. අතිගරු ජනාධිපතිතුමා විසින් අමාත්‍ය මණ්ඩලයේ සාමාජිකයින් පිළිගැනීම සහ අදහස් දැක්වීම.
03. පැමිණීමට නොහැකි වීමට හේතු දැන්වීම.
04. (අ) 2020 අප්‍රේල් මස 01 වන දින පැවැත්වූණු අමාත්‍ය මණ්ඩල රැස්වීමේ වාර්තාව සම්මත කිරීම.

**(ආ) පෙර පැවැති රැස්වීම්වලට අදාළ කරුණු කිසිවක් වේ නම්:**

අනු: අංකය	අමාත්‍ය මණ්ඩල පත්‍රිකා අංකය	අමාත්‍යාංශය	මාතෘකාව හා දිනය
1	20/0838/204/082	පුද්ගල, ආර්ථික සහ ප්‍රතිපත්ති සංවර්ධන	අයිසලා - සංවර්ධන - බැංකුව - විසින් - ශ්‍රී ලංකා රුපියල - මිල - අදාළ - වූ - අත්වරිල - බැඳුම්කර නිකුත් කිරීමේ යෝජනාව - 2020-4-6
2	20/0829/206/034-1	නාගරික සංවර්ධන, ජල සම්පාදන සහ නිවාස සහ පුනම්	නාගරික සංවර්ධන, ජල සම්පාදන හා නිවාස සහ පුනම් - අමාත්‍යාංශය - යටතේ - ඇති ඉංජිනේරුමය ඉදිකිරීම් ආයතන සඳහා - සාප්පු ලෙස තොරතුරු ප්‍රදානය කිරීම - 2020-4-6

## 2. Sample System generated Agenda (English Version)

**Confidential**

**AGENDA FOR THE MEETING OF THE CABINET OF MINISTERS TO BE HELD ON WEDNESDAY 3 JUNE, 2020 AT THE AT THE PRESIDENTIAL SECRETARIAT**

**(A) Commencement of the Meeting :**

01. Religious Observance.
02. Welcome of Members of the Cabinet and briefing by H.E. the President.
03. Excuses for absence.
04. (a) Confirmation of the minutes of the Cabinet Meeting held on 08th April 2020.

**(B) Cabinet Papers:**

S.No	Cabinet Paper No.	Ministry	Title & Date
1	20/0838/204/082	Finance, Economy and Policy Development	Proposal to issue Offshore Sri Lankan Rupee (LKR) Linked Bonds by the Asian Development Bank - 6/04/2020
2	20/0829/206/034-1	Urban Development, Water Supply and Housing Facilities	Direct awarding of contracts to Engineering & Construction Organizations under the Ministry of Urban Development, Water Supply and Housing Facilities - 6/04/2020

### 3. Sample System generated Agenda (Tamil Version)

(இரகசியமானது)

2020 யூன் மாதம் 03 ஆம் திகதி புதன்கிழமை மு.ப.10.30 மணியளவில்  
சனாதிபதி செயலகத்தில் நடாத்துவதற்குள்ள அமைச்சரவைக்  
கூட்டத்துக்கான நிகழ்ச்சி நிரல்

(அ) கூட்ட ஆரம்பம் :

01. மத அனுட்டானம்.
02. அதிமேதகைய சனாதிபதி அவர்களினால் அமைச்சரவை உறுப்பினர்களை வரவேற்றலும் கருத்து தெரிவித்தலும்.
03. சமூகமளிக்க முடியாமைக்கான அறிவித்தல்.
04. (அ) 2020 யூன் மாதம் 03 ஆம் திகதி புதன்கிழமை நடாத்தப்பட்ட அமைச்சரவைக் கூட்டத்தின் நிகழ்ச்சிக் குறிப்பினை உறுதிப்படுத்துதல்.

(ஆ) அமைச்சரவைப் பத்திரங்கள் :

தொடர் இல.	அமைச்சரவைப் பத்திர இலக்கம்	அமைச்சு	தலையங்கமும் திகதியும்
1	200828/204/082	நிதி, பொருளாதாரம் மற்றும் கொள்கை அபிவிருத்தி	ஆசிய அபிவிருத்தி வங்கியினால் இலங்கை ரூபாவினை அடிப்படையாகக் கொண்டு கரை கடந்த முறிகளை வழங்கும் பிரேரிப்பு - 04/2020
2	200829/206/034-1	நகர அபிவிருத்தி, நிர்வாகம் மற்றும் நிர்வாகம்	நகர அபிவிருத்தி, நிர்வாகம் மற்றும் நிர்வாகம் விடமைப்பு வசதிகள் அமைச்சின்

### 4. Sample System generated Observation letter

CP 20/0920/206/038

2020-06-10

Mr. S.R. Attiyalla  
Secretary,  
Ministry of Finance, Economy and Policy Development

Dear Sir,

Payment of salaries to the Staff of Public Enterprises, which come under the purview of the Ministry of Urban Development, Water Supply and Housing

Please find enclosed herewith the above captioned Cabinet Memorandum dated 2020-06-09 submitted by the Hon. Minister of Urban Development, Water Supply and Housing Facilities, which is scheduled to be taken up for consideration at the next Meeting of the Cabinet of Ministers.

I would be most grateful if you could please bring the contents of the said Memorandum to the notice of your Hon. Minister and forward his observations to this office at your earliest.

Your cooperation in this regard is very much appreciated.

Yours sincerely,

M.G. ~~Tharissa Bandula~~ Sekil S. Amarasena  
Senior Assistant Secretary Secretary to the Cabinet of Ministers

Ce: 01. Mr. A.K. ~~Seneviratne~~ for necessary action please  
Deputy Secretary to the Treasury  
(A copy of the Cabinet Memorandum is annexed)

Obs#-220/0920a



# Appendix G Code Listing

## 1. Login Module

```
Imports System.IO
Imports System.Security.Cryptography
Imports System.Text
Imports MySql.Data.MySqlClient

' /
' / Developer : J.G.Badula Kularatne for Office of the Cabinet of Ministers.
' / eMail : bandula_ck@yahoo.com
' / Purpose: Login module. securely login to the system
' / Microsoft Visual Basic .NET (2019)
' /
2 references
Public Class frmLogin
    Dim cipherText As String
    Private Sub Login()
        Dim wrapper As New Simple3Des(txtPassword.Text)
        If txtUsername.Text = "" Or txtPassword.Text = "" Then
            MsgBox("User Name or Password cannot be blank", MsgBoxStyle.Critical, "Login Error")
            Exit Sub
        End If
        Try
            sql = "SELECT * FROM tblUSERS WHERE Username = '" & txtUsername.Text & "'"
            ConnDB()
            cmd = New MySqlCommand(sql, conn)
            dr = cmd.ExecuteReader()
            If dr.Read = True Then
                cipherText = dr("pwd")
                ' call wrapper class for decryption function
                Dim plainText As String = wrapper.Decrypt(cipherText)
                If txtPassword.Text = plainText Then
                    MsgBox("Welcome to Cabinet Paper Information System [CPIS].", MsgBoxStyle.Information, "Login")
                    frmMain.tsslUser.Text = UCase(txtUsername.Text)
                End If
            End If
        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
    End Sub
End Class
```

## 2. Load User Function

```
Imports MySql.Data.MySqlClient
Imports System.Security.Cryptography

' /
' / Developer : J.G.Badula Kularatne for Office of the Cabinet of Ministers.
' / eMail : bandula_ck@yahoo.com
' / Purpose: Module for user management
' / Microsoft Visual Basic .NET (2019)
' /
2 references
Public Class frmUser
    Dim cipherText As String
    Dim adding As Boolean
    Dim updating As Boolean
    Private Sub LoadUsers()
        ' Load users to Listview
        Try
            sql = "SELECT Username FROM tblUSERS ORDER BY Username"
            ConnDB()
            cmd = New MySqlCommand(sql, conn)
            dr = cmd.ExecuteReader()
            Dim x As New ListViewItem
            ListView1.Items.Clear()
            ' loop through all users
            Do While dr.Read = True
                x = New ListViewItem(dr("Username").ToString)
                ListView1.Items.Add(x)
            Loop
        Catch ex As Exception
            MsgBox(ex.Message)
        Finally
            cmd.Dispose()
            conn.Close()
        End Try
    End Sub
End Class
```

### 3. Validate Old Password Function

```
1 reference
Private Function ValidateOldPassword() As Boolean
    'Validate for user existence
    Dim wrapper As New Simple3Des(txtOldPassword.Text)
    Try
        sql = "SELECT * FROM tblUsers WHERE Username =" & txtUsername.Text & " "
        ConnDB()
        cmd = New MySqlCommand(sql, conn)
        dr = cmd.ExecuteReader()

        If dr.Read = True Then
            Dim plainText As String = wrapper.Decrypt(dr("pwd"))
            If txtOldPassword.Text = plainText Then
                Return True
            End If
        End If
    Catch ex As Exception
        MsgBox(ex.Message)
    Finally
        cmd.Dispose()
        conn.Close()
    End Try
End Function
```

### 4. Add new User Function

```
Private Sub AddUsers()
    ' Add new users
    Try
        Dim wrapper As New Simple3Des(txtPassword.Text)
        Dim cipherText As String = wrapper.Encrypt(txtPassword.Text)
        sql = "INSERT INTO tblUSERS(Username, Pwd, Role, staffName, Position) VALUES(' & txtUsername.Text & ', ' & c
        ConnDB()
        cmd = New MySqlCommand(sql, conn)
        Dim i As Integer
        i = cmd.ExecuteNonQuery()
        If i > 0 Then
            MsgBox("New user successfully saved.", MsgBoxStyle.Information, "User")
            SaveEventLogs(frmMain.tsslUser.Text, "tblusers", "Add", " Add new user " & txtUsername.Text & " by " & frmM
        End If
    Catch ex As Exception
        MsgBox(ex.Message)
    Finally
        cmd.Dispose()
        conn.Close()
    End Try
End Sub
```

## 5. MySQL Connection Function

```
Imports MySql.Data.MySqlClient
0 references
Module SQLConn
    Public ServerMySQL As String
    Public PortMySQL As String
    Public UserNameMySQL As String
    Public PwdMySQL As String
    Public DBNameMySQL As String
    Public CharSetMySQL As String
    Public sql, sql1 As String
    Public ds As New DataSet
    Public cmd As MySqlCommand
    Public dr As MySqlDataReader
    Public da As MySqlDataAdapter
    Public conn As New MySqlConnection
99+ references
    Public Sub ConnDB()
        conn.Close()
        Try
            conn.ConnectionString = "Server = " & ServerMySQL & "; " & _
                & "Port = " & PortMySQL & "; " & _
                & "Database = " & DBNameMySQL & "; " & _
                & "user id = " & UserNameMySQL & "; " & _
                & "password = " & PwdMySQL & ""
            conn.Open()
        Catch ex As Exception
            Try
                conn = New MySqlConnection("datasource = SERVER14; port=3306; username=root; password=12345; database=cpps; Character Se
                conn.Open()
            Catch
                MsgBox("Please configure database.", MsgBoxStyle.Information, "Database")
            End Try
            MsgBox("The system failed to establish a connection", MsgBoxStyle.Information, "Database Settings")
        End Try
    End Sub
End Sub
```

## 6. Save user logs function

```
Public Sub SaveLogs(Log_Type As String, u_name As String)
    ' this public function saves user logins/logouts
    Select Case Log_Type
        Case "Login"
            Try
                sql = "INSERT INTO tbllogs(username, datein, dateout) VALUES ('" & u_name & "', '" & Date.Now.ToString(
                ConnDB()
                cmd = New MySqlCommand(sql, conn)
                Dim i As Integer
                i = cmd.ExecuteNonQuery()

                Catch ex As Exception
                    MsgBox(ex.Message)
                Finally
                    cmd.Dispose()
                    conn.Close()
                End Try
            Case "Logout"
                Try
                    sql = "UPDATE tbllogs SET dateout = '" & Date.Now.ToString("s") & "' WHERE username = '" & u_name & "'
                    ConnDB()
                    cmd = New MySqlCommand(sql, conn)
                    Dim i As Integer
                    i = cmd.ExecuteNonQuery()

                    Catch ex As Exception
                        MsgBox(ex.Message)
                    Finally
                        cmd.Dispose()
                        conn.Close()
                    End Try
            End Select
    End Sub
```

## 7. Search CP Function

```
Public Function SearchCP(searchStr As String) As Boolean
    ' this function is for searching a CP for existance
    Try
        sql = "SELECT COUNT(*) FROM tblCPaper WHERE CPNo = '" & searchStr & "'"
        ConnDB()
        cmd = New MySqlCommand(sql, conn)
        Dim i As Integer
        i = cmd.ExecuteScalar()
        If i > 0 Then
            SearchCP = True
        Else
            SearchCP = False
        End If
    Catch ex As Exception
        MsgBox(ex.Message)
    Finally
        cmd.Dispose()
        conn.Close()
    End Try
End Function
```

## 8. Class for Add/Update Cabinet Papers

```
Imports MySql.Data.MySqlClient
' / -----
' / Developer   : J.G.Badula Kularatne for Office of the Cabinet of Ministers.
' / Purpose     : This Class is for adding and updating Cabinet Papers
' / Microsoft Visual Basic .NET (2019)
' / -----
2 references
Public Class frmAddUpdateCP
    Public AddLotSearch As Boolean
    Public isCP As Boolean = False
    Dim date1 As String

    1 reference
    Private Sub AddCP()
        Try
            date1 = CStr(Date.Today)
            sql = "INSERT INTO tblCPaper(CPNo, GroupCode, PrimaryMinNo, MinSerialNo, Title_E, Title_S, SignedDate, RecievedDate,
            ConnDB()
            cmd = New MySqlCommand(sql, conn)
            Dim i As Integer
            i = cmd.ExecuteNonQuery()
            If i > 0 Then
                MsgBox("Cabinet Paper information successfully added.", MsgBoxStyle.Information, "")
                SaveEventLogs(frmMain.tsslUser.Text, "tblcpaper", txtCPNo.Text, " added by " & frmMain.tsslUser.Text)
            Else
                MsgBox("Failed in adding Cabinet Paper Registration.", MsgBoxStyle.Critical, "Error")
                SaveEventLogs(frmMain.tsslUser.Text, "tblcpaper", txtCPNo.Text, " failed adding by " & frmMain.tsslUser.Text)
            End If
        Catch ex As Exception
            MsgBox(ex.Message)
        Finally
            cmd.Dispose()
            conn.Close()
        End Try
    End Sub
```

## References

- [1] Mark Richards, "Software Architecture Patterns" 2015, O'Reilly
- [2] G. Booch, *The Unified Modeling Language User Guide*, Pearson Education India, 2005.
- [3] Ian Sommerville, *Software Engineering*, 9<sup>th</sup> edition, Addison-Wesley, 2015.
- [4] Roger S. Pressman, *Software Engineering*, 7th edition A Practitioner's Approach, McGraw Hill, 2010.
- [5] G. J. Myers, C. Sandler, T. Badgett, *The Art of Software Testing*, John Wiley & Sons, Sep 23, 2011 - Business & Economics
- [6] H. Mcheick and Y. Qi, "Dependency of components in MVC distributed architecture," in Electrical and Computer Engineering (CCECE), 2011 24<sup>th</sup> Canadian Conference, 8-11 May 2011, pp. 6-10.
- [7] visual-paradigm [online], available: <https://www.visual-paradigm.com/learning/handbooks/agile-handbook/desktop-application-wireframe.jsp>
- [8] <http://www.filehold.com/home>
- [9] <http://www.docmosis.com/>
- [10] <https://e-estonia.com/solutions/e-governance/e-cabinet/>
- [11] Overview of .Net Framework. [Online]  
Available at: <https://docs.microsoft.com/en-us/dotnet/framework/get-started/overview>  
[Accessed 10/04/2020]
- [12] MySQL Administration. [Online]  
Available at : <https://www.mysqltutorial.org/mysql-administration.aspx>  
[Accessed 22/04/2020]