|  |  |
| --- | --- |
|  | **MINISTRY OF EDUCATION AND TRAINING** |

|  |
| --- |
| **FPT UNIVERSITY** |
| Capstone Project Document |
| Clinic Management Application |

|  |  |  |
| --- | --- | --- |
| **SWP490\_G23** | | |
| **Group members** | Nguyen Thi Trang | SE05803 |
| Nguyen Duc Thien | SE05883 |
| Le Thi Thu Trang | SE05909 |
| Do Ngoc Khanh | SE06047 |
| Do Trung Duc | SE05844 |
| **Supervisor** | Mr. Bui Dinh Chien | |
| **Capstone Project code** | CMA | |

- Hanoi, 12/2020 -

**Table of Contents**

[Acknowledgement 4](#_heading=h.1fob9te)

[Definition and Acronyms 4](#_heading=h.3znysh7)

[I. Project Introduction 5](#_heading=h.2et92p0)

[1. Overview 5](#_heading=h.tyjcwt)

[2. Product Background 5](#_heading=h.4d34og8)

[3. Existing Systems 5](#_heading=h.2s8eyo1)

[4. Business Opportunity 5](#_heading=h.17dp8vu)

[5. Software Product Vision 5](#_heading=h.3rdcrjn)

[6. Project Scope & Limitations 5](#_heading=h.26in1rg)

[II. Project Management Plan 6](#_heading=h.lnxbz9)

[1. Overview 6](#_heading=h.35nkun2)

[2. Management Approach 6](#_heading=h.z337ya)

[3. Master Schedule 6](#_heading=h.4i7ojhp)

[4. Project Organization 6](#_heading=h.2xcytpi)

[5. Project Communication 6](#_heading=h.1ci93xb)

[6. Configuration Management 6](#_heading=h.3whwml4)

[III. Software Requirement Specification 7](#_heading=h.1pxezwc)

[1. Overall Description 7](#_heading=h.49x2ik5)

[2. User Requirements 7](#_heading=h.3o7alnk)

[3. Functional Requirements 7](#_heading=h.23ckvvd)

[4. Non-Functional Requirements 7](#_heading=h.1hmsyys)

[5. Other Requirements 7](#_heading=h.vx1227)

[IV. Software Design Description 8](#_heading=h.3fwokq0)

[1. Overall Description 8](#_heading=h.1v1yuxt)

[2. System Architecture Design 8](#_heading=h.28h4qwu)

[3. System Detailed Design 8](#_heading=h.2lwamvv)

[4. Class Specification 8](#_heading=h.206ipza)

[5. Data & Database Design 8](#_heading=h.4k668n3)

[V. Software Testing Documentation 9](#_heading=h.3ygebqi)

[1. Overall Description 9](#_heading=h.2dlolyb)

[2. Test Plan 9](#_heading=h.4bvk7pj)

[3. Test Cases 9](#_heading=h.kgcv8k)

[4. Test Reports 9](#_heading=h.34g0dwd)

[VI. Release Package & User Guides 10](#_heading=h.1jlao46)

[1. Deliverable Package 10](#_heading=h.43ky6rz)

[2. Installation Guides 10](#_heading=h.2iq8gzs)

[3. User Manual 10](#_heading=h.xvir7l)

[VII. Appendix 10](#_heading=h.1baon6m)

[1. Glossary [Optional] 10](#_heading=h.3vac5uf)

[2. References [Optional] 10](#_heading=h.2afmg28)

[3. Others [Optional] 10](#_heading=h.pkwqa1)

# Acknowledgement

*[Fill team’s acknowledgement here…]*

# Definition and Acronyms

*[Fill all the definitions, acronyms,… used within the document] in the table format as below]*

|  |  |
| --- | --- |
| **Acronym** | **Definition** |
| PWM | Psychology website |
| AWS | Amazon Web Services |
| BA | Business Analysis |
| BR | Business Rule |
| ERD | Entity Relationship Diagram |
| GUI | Graphical User Interface |
| PM | Project Manager |
| SDD | Software Design Description |
| SPMP | Software Project Management Plan |
| SRS | Software Requirement Specification |
| UAT | User Acceptance Test |
| UC | Use Case |
| API | Application Program Interface |

# I. Project Introduction

## 1. Overview

### 1.1 Project Information

* Project name: Clinic Management Application
* Project code: CMA
* Group name: SWP490-G23
* Software type: Web-based application

### 1.2 Project Team

#### a. Supervisor

|  |  |  |  |
| --- | --- | --- | --- |
| **Full Name** | **Email** | **Phone Number** | **Title** |
| Bui Dinh Chien | ChienBD@fe.edu.vn |  | Lecturer |

#### b. Team Members

|  |  |  |  |
| --- | --- | --- | --- |
| **Full Name** | **Email** | **Mobile** | **Role** |
| Nguyen Thi Trang | trangntse05803@fpt.edu.vn | 0337631111 | Leader |
| Lê Thị Thu Trang | tranglttse05909@fpt.edu.vn | 0961818500 | Member |
| Do Ngoc Khanh | khanhdnse06047@fpt.edu.vn | 0971703376 | Member |
| Do Trung Duc | ducdtse05844@fpt.edu.vn | 0962481497 | Member |
| Nguyen Duc Thien | thienndse05883@fpt.edu.vn | 0566662225 | Member |

## 2. Product Background

Nowadays, private clinics are appearing more and more with the goal of reducing the load of large hospitals. In Vietnam, private clinics serve about 60% of outpatient visits. Most of the people who open the clinic are doctors of the big hospital and they understand that the problem in the big hospital is not enough to supply the large number of patients. Patients often spend money to use the service, they always want the best treatment. However, in major hospitals, patients have to wait in long lines. Most major hospitals will not support the booking number, but if you want to visit these hospitals, patients usually have to come very early in the day. On average, patients spend about 3 hours waiting for the results of their blood tests and it can take all day just for the results of the examination. In addition, many procedures are not instructed in detail to departments. In private clinics, they do not provide waiting services, make appointments in advance, check-in quickly, in addition the doctors are enthusiastic and attentive. Therefore, the tendency of people to seek counseling rooms is increasing.A easy-to-use clinic management software suitable for clinic size is always the top concern of clinics.

## 3. Existing Systems

Nowadays, there are many diversified systems being used for management clinics and hospitals. The following is one of many management systems for clinics that has been used by many people and with specific users.

### 3.1 *Ultrasound Clinic*

***Description:***

The work of the clinic is done entirely on Word and Excel.

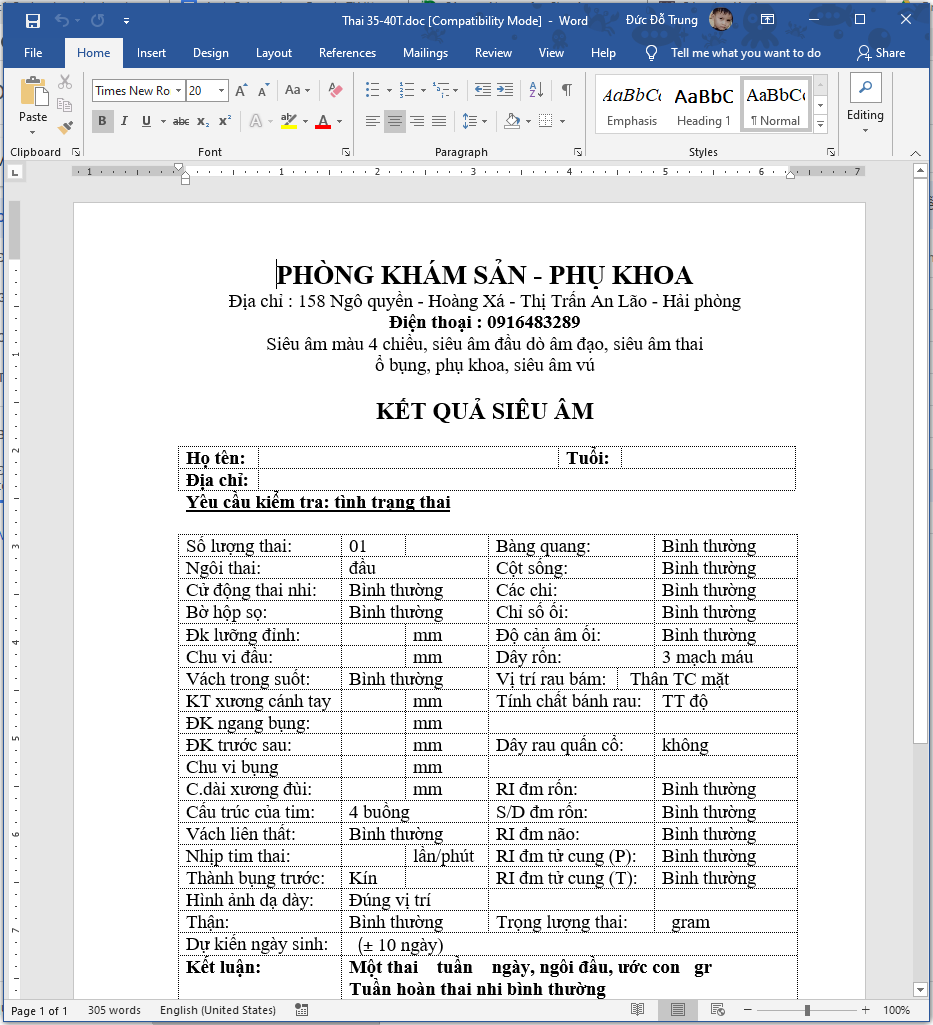
******

Figure 1: Ultrasound Clinic template

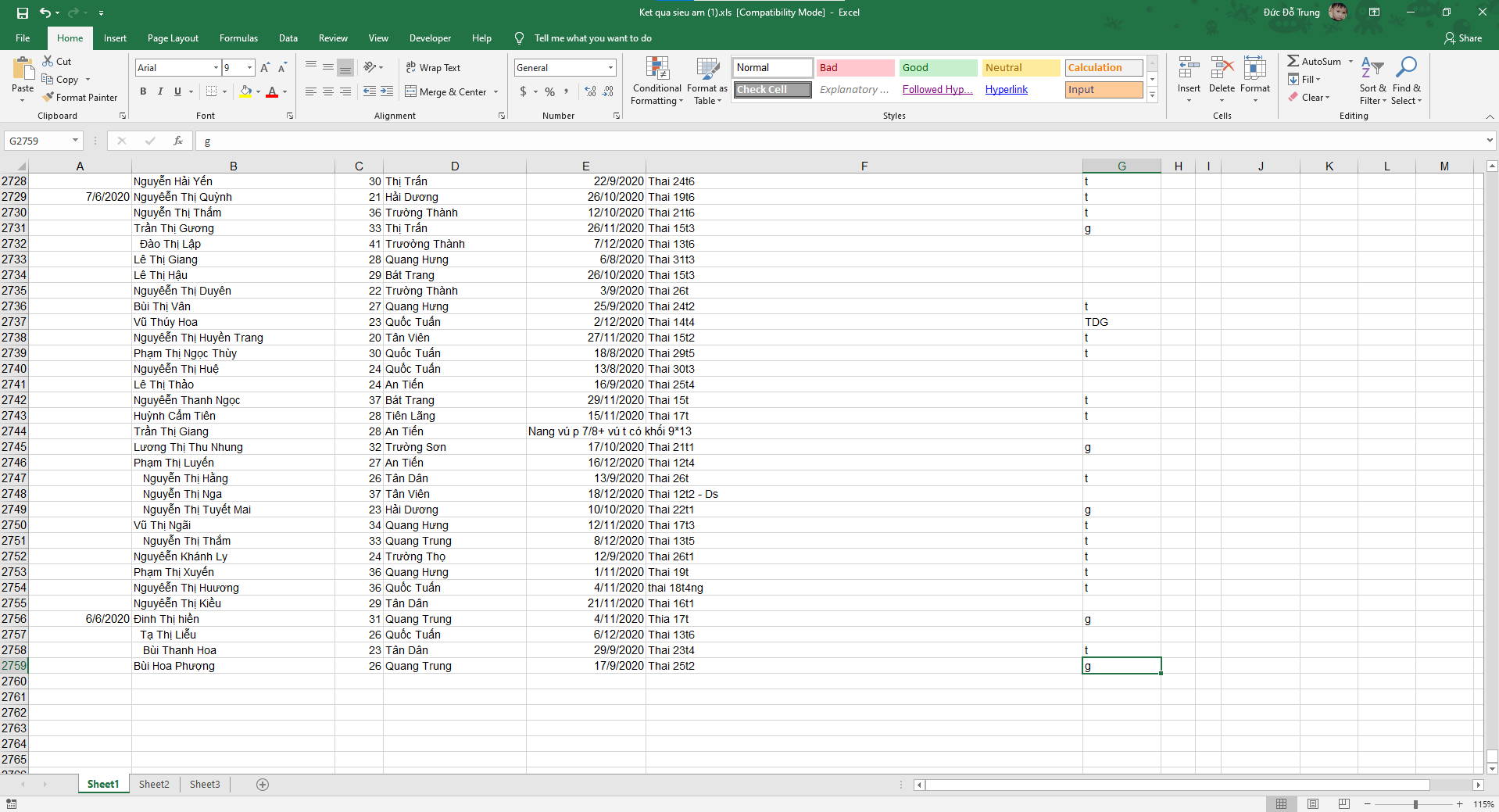
******

Figure 2: Ultrasound Clinic

***As-is:***

* **Step 1:** Patients in need of examination (can be scheduled by phone).
* **Step 2:** The nurse asks the patient what service ultrasound they want to use and then opens the “*template corresponding*” to the service on the computer.
* **Step 3:** The doctor examines the patient. During an ultrasound, the doctor will read the results and the nurse will fill in the results.
* **Step 4:** After the ultrasound is complete, the nurse prints the ultrasound results for the patient.
* **Step 5:** After the above step, the nurse must open the Excel file and re-enter the Word file information above to manage the medical records.
* **Step 6:** The doctor prescribes the drug to the patient.
* **Step 7:** The patient takes the medicine at the pharmacy and pays for it.
* **Step 8:** When the old patient comes to the examination, the employee must look up the patient's information in the Excel file.

***Pain-point:***

* Using cash for payment is the main method
* Difficult to find patients, the number of patients with the same names on the excel file is too many.
* Takes time in importing word files to excel files.
* Difficulty in filtering outdated data.
* Difficulty in managing word files, excel templates.
* Difficulty administering drugs according to various information (date of entry, quantity, expiry date, supplier, unit price).

***Pros & cons:***

|  |  |
| --- | --- |
| **Pros** | **Cons** |
| * Simple to use. | * File loss, data loss (crashing computer, virus, ...). * Ease of input data errors. * Medicine management in paper. * There is a high cost compared to many clinics. * Difficulty in data statistics.. |

### 

### 3.2 BigSoft

***Name*:** BigSoft

***Description*:** Eye clinic management system.

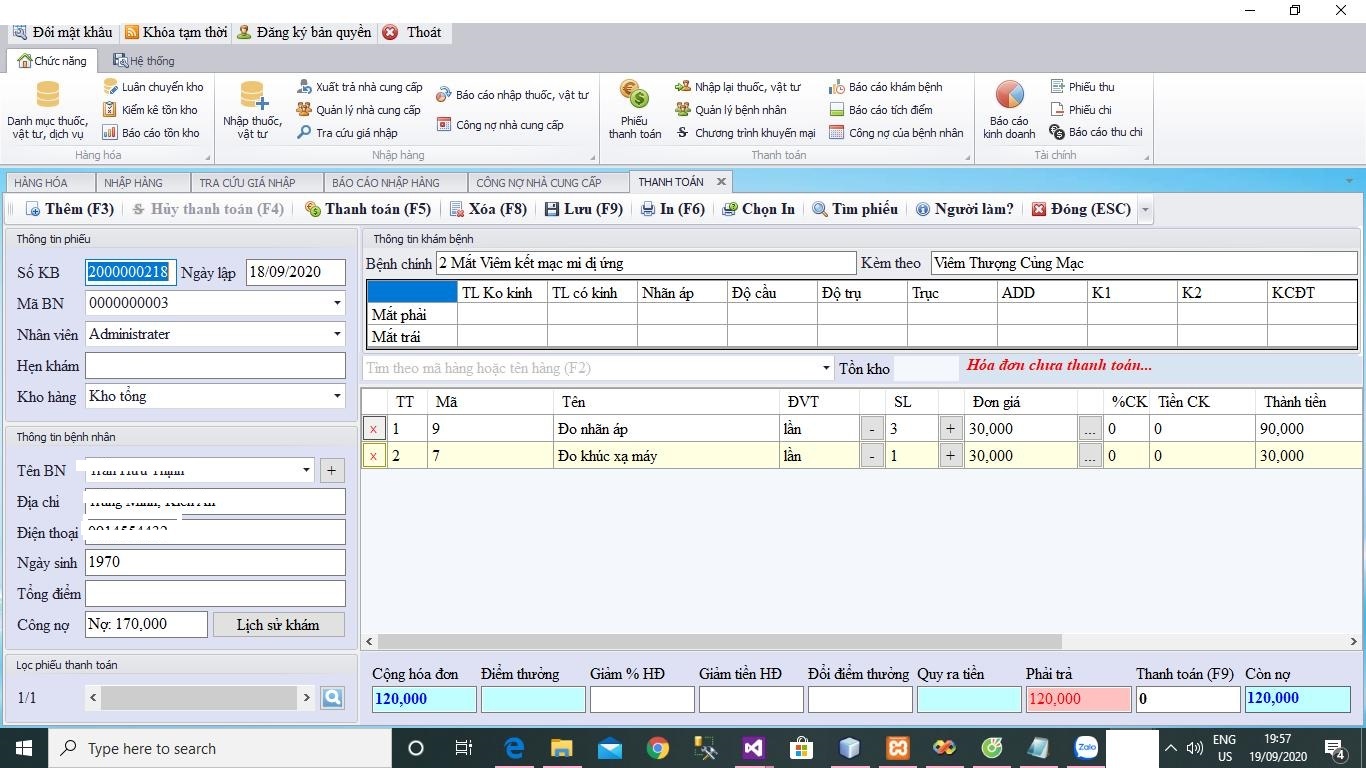


Figure 3: Existing system BigSoft

***As-is:***

* **Step 1:** The patient comes to the clinic.
* **Step 2:** Nurses enter client information in the information section of the software.
* **Step 3:** The doctor examines the eye, then the nurse enters the result information into the software.
* **Step 4:** The doctor concludes the main disease and then the nurse enters the information into the software.
* **Step 5:** The doctor prescribes the service or prescribes the medicine and then enters the information into the software.
* **Step 6:** Nurses use software to print bills.
* **Step 7:** Patients pay for drugs and services.
* **Step 8:** The nurse noted the patient paid on the software.

***Pain-point:***

* The software has no reports on the shelf life of the drug.
* There is no customer support channel.
* The system does not decentralized access.

***System actors:***Administrator

***Features:***

* Management of medical records.
* Financial management, revenue.
* Manage pharmacies, drug stores.
* Medical consumption management.
* Create reports, statistics.
* Subclinical management.
* Management re-examination.
* Suppliers management.

***Pros & cons:***

|  |  |
| --- | --- |
| **Pros** | **Cons** |
| * Simple software interface. * The software still works normally without the internet. * High Secure. * Simple user interface that is easy to use * Installed directly on an offline computer. Not affected by the internet. | * Only run Windows computer environments. * The software does not divide roles. * The manager cannot manage it remotely. * Do not immediately switch to another computer for use when the computer is currently using the problem (virus, win error, ...) |

### 

### 3.3 Nano clinic

***Name*:** Nano clinic

***Description*:** ABClinic management system.

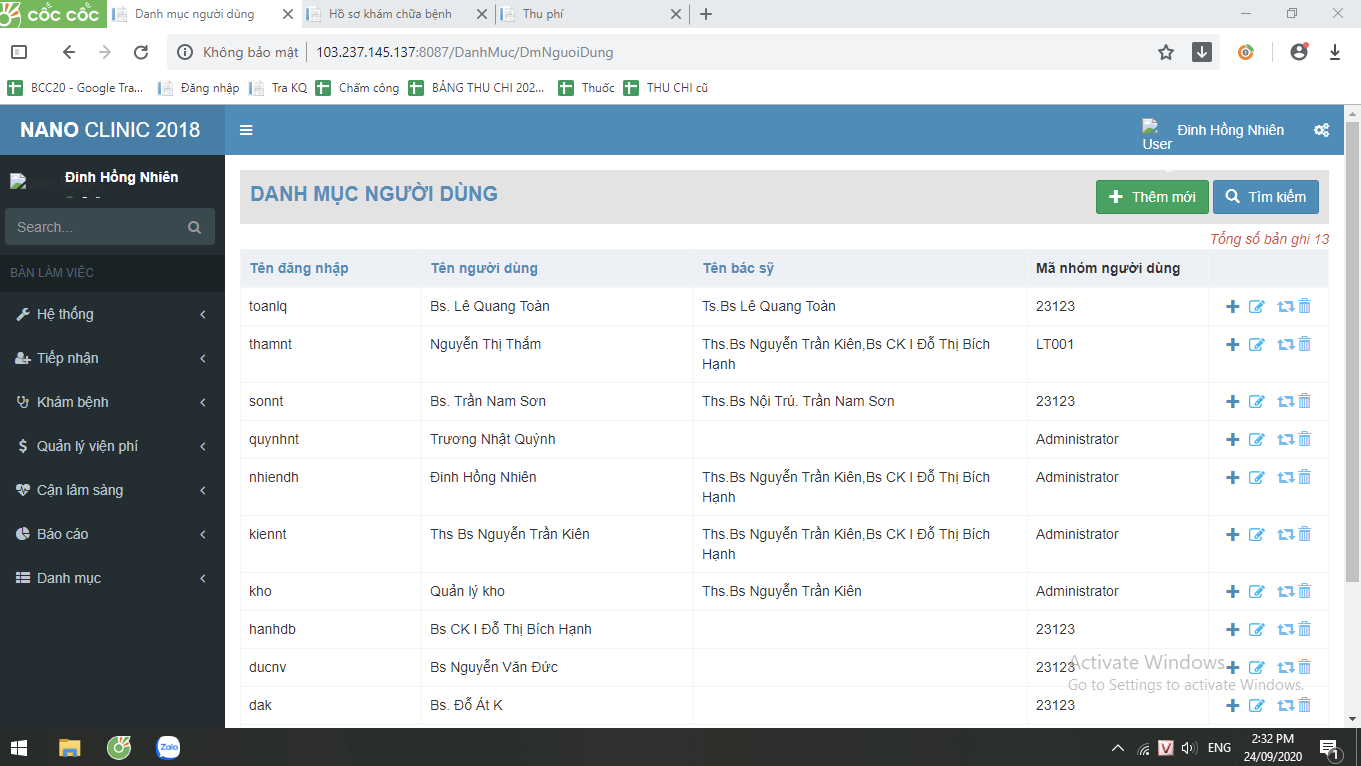


Figure 4: Existing system Nano clinic

***As-is:***

* **Step 1:** The patient makes an appointment with the doctor in advance about the examination schedule.
* **Step 2:** The patient comes to the clinic.
* **Step 3:** The receptionist nurse enters administrative information (name, age, address, ...) into the software.
* **Step 4:** The nurse asks the symptom and asks the patient what he wants to do, then generates the questionnaire in the system.
* **Step 5:** The nurse takes the patient to see the doctor.
* **Step 6:** The doctor does a preliminary check (heart, pulse, blood pressure), but the results are not saved to the system.
* **Step 7:** The doctor or nurse chooses which tests to perform on the "*medical tests prescription*" form on both system and paper .
* **Step 8:** The patient goes for an examination then brings the "*medical tests prescription report*" back to the doctor.
* **Step 9:** The nurse captures the "*medical tests prescription report*" and then saves them on Google Drive.
* **Step 10:** The doctor diagnoses and “*prescribes*” the drug.
* **Step 11:** The patient payment and nurses use the system to print bills.

***Pain-point:***

* Some features are not necessary and do not save user time compared to manual jobs available at work.
* Some features are slow to respond.
* The software does not have cost and debt management features.
* Test results when you want to save on the software, the implementation steps are still very complicated.
* It is not possible to compare many old patient results directly on the software.
* Cannot be used if the clinic has multiple branches.

***System actors:***Administrator

***Features:***

* Manage users on the system
* Manage categories.
* Receiving patients.
* Manage medical examination process.
* Financial management.
* Report management, systematic statistics.
* Management of drugs and supplies.
* Manage subclinical examination.

***Pros & cons:***

|  |  |
| --- | --- |
| **Pros** | **Cons** |
| * High Secure. * Help management wherever you are. * Unlimited device access number. | * The software still has many bugs. * The software cannot be used without a network. * Software has complex business. Difficult to use, making it difficult to manage and supervise clinic business activities. * There is a high cost compared to many clinics. * It does not respond to the detailed management requirements of the clinic. |

### 

## 4. Business Opportunity

Many private clinics require a management system tailored to the size and characteristics of their clinic. Graphic user interfaces are easy to use and full of management functions. However, the owner of the clinic can manage anytime, anywhere and makes it easy for patients to book appointments. Such a management system will help the clinic operate more smoothly, saving more time and money. It doesn't take employees too long to report and handle the archive. Receiving and processing patient records, optimal testing.

## 5. Software Product Vision

For private clinic owners who manage online clinic operations on the internet, laptops help to manage more accurately and transparently. Unlike the current management software, when the power is out or Windows is not installed, it will be difficult to handle medical requirements. Plus, when using a web-based application, you can easily manage it on your phone. Besides, when there are features suitable to the size and needs of the clinic, it will help to handle and manage the clinic more optimally over time.

## 6. Project Scope & Limitations

### 6.1 To be

* **Step 0:** Patients can make an appointment or come directly to the clinic (In the case of appointment will be given priority in order)
* **Step 1:** Patients who come to the clinic meet the nurse in the patient admission office.
* **Step 2:** The nurse asks the patient whether to come for the new examination or re-examination
  + Case 1: Patient comes for the new examination:
    - Nurses fill administrative information (name, age, address ...) into the patient reception window on the system.
  + Case 2: Patients come to re-examination or have been examined at the clinic:
    - Nurses find the patient's administrative information already available.
* **Step 3:** The nurse asks the patient's symptoms and wishes (what examination, which doctor?) And creates the “*medical examination forms”*of patients on the software.
* **Step 4**: The nurse printed the “*number card”* to the patient ( ordinal numbers, consulting room name, doctor) given to the patient.
* **Step 5:** Patient pays for medical services at cashier’s, the cashier prints the bill with the payment confirmation for the patient.
* **Step 6:** Patients who come to the waiting room see the consulting room name and ordinal number.
* **Step 7:** The doctor will do a physical examination (measure blood pressure, heart rate, temperature, ask medical history, ...) and record the results into *medical examination form* on the system.
* **Step 8:** Doctors diagnose the disease
  + Case 1: Appoint patients to perform subclinical services (ultrasound, blood, urine, endoscopy, X-ray ..) on the “*medical tests prescription”*and print it given to the patient.
  + Case 2: In case of not doing subclinical services, the patient will be prescribed the drug on the "*prescription*" form by the doctor.
* **Step 10:** Patient comes to pay for subclinical services at cashier’s ( the cashier prints the bill with the payment confirmation for the patient).
* **Step 11:** Sampling patient (In case of ultrasound or X-ray, endoscopy, .. patient will be taken to ultrasound room / scan and wait for ultrasound / scan).
* **Step 12:** Patients who receive results after completing a subclinical service such as a “*medical tests prescription report* “back to the previously designated physician.
* **Step 13:** Doctors diagnose the disease according to “*medical tests prescription report*” thereafter fill in “*medical examination form”* and the drug on the "*prescription*" form by the doctor (if any).
* **Step 14:** The patient brings "*prescription*" to the pharmacy and payment ( the cashier prints the bill with the payment confirmation for the patient) thereafter receives medicine.

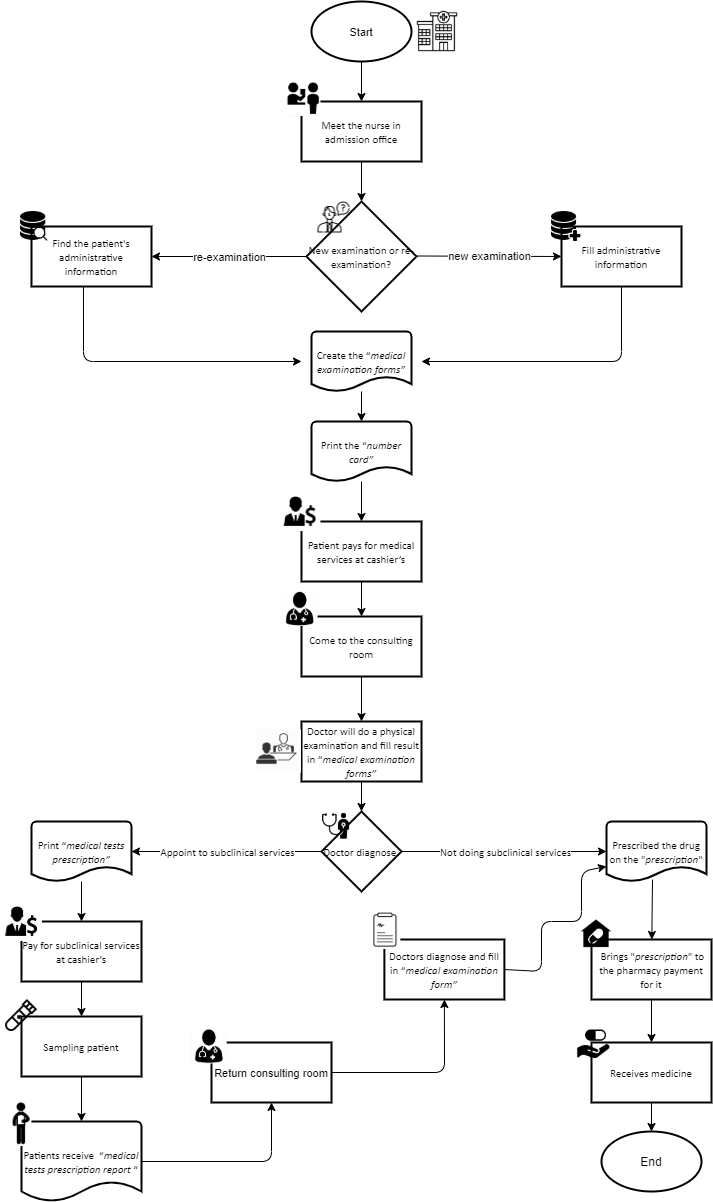


Figure 5: Flowchart tobe of CMA

### 6.2 Major Features

**6.2.1. Role staff**

* Make an appointment.
* Manage appointments.
* Receiving the patient (making the “*medical examination form”*) in the Admission Office.
* Print the ordinal number slip (“*ordinal number”*).
* See the number of examined or waiting patients in each waiting room.
* Manage the list of “*medical examination forms”.*
* View the medical history and physical exam of patients.
* Enter the clinical examination results in the “*medical examination form”.*
* Indication of subclinical services.
* Print list type of “*medical tests prescription”* form.
* Enter the results diagnosed in “*medical examination forms”*.
* ***“****Prescribing****”*** to patients.
* Save “*medical examination form”* into the system*.*
* End of examination
* Receive “*medical tests prescription”*.
* Manage the list of subclinical examinations.
* Enter subclinical results according to the available forms.
* Print “*medical tests prescription report”*.
* Get patient information.
* Collecting fees service, confirming collection.
* Collection of pharmaceutical and materials fees
* Create a receipt for cash.
* Create a payment for cash.
* Create liabilities form.
* Statistics receipt of revenue and expenditure.
* Reporting patient liabilities.
* Reporting liabilities with suppliers.
* Patient contact information management.
* Create customizable reports for management
* Administration of drug entry.
* Statistics pharmaceutical and materials.
* Drug cancellation.
* Statistics on drug destruction.
* Make a stock release note.
* Statistics of material warehouse.
* Make a note to re-import drugs and supplies.
* Statistics of drug and materials re-import slip.
* Managing the drug list.
* Suppliers contact information
* Statistics prescriptions.
* Management of drug release.
* Statistics on drug sales.
* Re-import the drug.
* Medicines re-enter statistical sheets.
* Print invoice for sale.

**6.2.2. Role manager**

* Account management
* Account decentralization
* View the history of the system
* Report management template
* Generate customizable “*medical tests prescription”*form.

### 6.3 Limitations & Exclusions

LI-1: The system is only for medium-sized private clinics.

LI-2: Not affiliated with insurance.

# II. Project Management Plan

## 1. Overview

### 1.1 WBS & Estimation

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **WBS Item** | **Complexity** | **Est. Effort**  **(man-days)** |
| ***1*** | ***Existed systems research*** |  | ***10*** |
| 1.1 | Identify the theme of the Capstone project | Complex | *5* |
| 1.2 | Contact with end users of existed systems | Complex | *4* |
| 1.3 | Defines project name | Simple | *1* |
| ***2*** | ***Initiating*** |  | ***9*** |
| 2.1 | Kick-off | Simple | 1 |
| 2.2 | Identify stakeholder | Complex | 4 |
| 2.3 | Develop project introduction document | Medium | 4 |
| ***3*** | ***Planning*** |  | ***17*** |
| 3.1 | Define scope | Simple | 2 |
| 3.2 | Choose working model process | Medium | 1 |
| 3.3 | Choose tools & techniques | Complex | 1 |
| 3.4 | Develop project schedule | Medium | 1 |
| 3.5 | Development project resource | Medium | 3 |
| 3.6 | Define risk project | Medium | 4 |
| 3.7 | Develop project plan document | Medium | 5 |
| ***4*** | ***Analysis*** |  | ***26*** |
| 4.1 | Gather Requirements | Complex | 5 |
| 4.2 | Training Strategy | Medium | 4 |
| 4.3 | Implementation Strategy | Medium | 4 |
| 4.4 | Create Requirements Management Plan | Simple | 1 |
| 4.5 | Capture Additional Client Information | Medium | 4.5 |
| 4.6 | Build Conceptual Systems Design | Complex | 2.5 |
| 4.7 | Develop Software Requirement document | Medium | 5 |
| ***5*** | ***Design*** |  | ***31*** |
| 5.1 | Ensure Architectural Foundation | Medium | 5 |
| 5.2 | Validate Standards and Guidelines | Complex | 5 |
| 5.3 | Design the Solution | Medium | 6 |
| 5.4 | Build the Technical Systems Design | Complex | 5 |
| 5.5 | Review the Design | Medium | 5 |
| 5.6 | Develop Software Design Document | Medium | 5 |
| ***6*** | ***Implementation*** |  | ***40*** |
| 6.1 | Prepare for Implementation | Simple | 5 |
| 6.2 | Perform Training | Medium | 5 |
| 6.3 | Implement the Solution | Complex | 20 |
| 6.4 | Pilot Test the Solution | Medium | 5 |
| 6.5 | Monitor the Solution | Simple | 5 |
| ***7*** | ***Testing*** |  | ***24*** |
| 7.1 | Validate Test Coverage | Simple | 5 |
| 7.2 | Integration Testing | Complex | 5 |
| 7.3 | System Testing | Complex | 5 |
| 7.4 | User Acceptance Testing | Complex | 4 |
| 7.5 | Develop Software User Guides Document | Medium | 5 |
| ***8*** | ***Deployment*** |  | ***3*** |
| 8.1 | Deploy on server |  | 3 |
| ***Total Estimated Effort (man-days)*** | | | ***160*** |

### 1.2 Project Objectives

Provide the overall project objective description and then the specific target metrics of your project in terms of time, cost, and quality. For example

* Timeliness (%):
* Allocated Effort (man-days): 160
* Defect Distribution:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Quality Stage** | **No. of Defects** | **% of Defect** | **Notes** |
| 1 | Reviewing | 100 | 15 |  |
| 2 | Unit Test | 250 | 20 |  |
| 3 | Integration Test | 280 | 35 |  |
| 4 | System Test | 320 | 25 |  |
| 5 | User Acceptance Test | 50 | 5 |  |
| ***Total*** | | ***1000*** | ***100%*** |  |

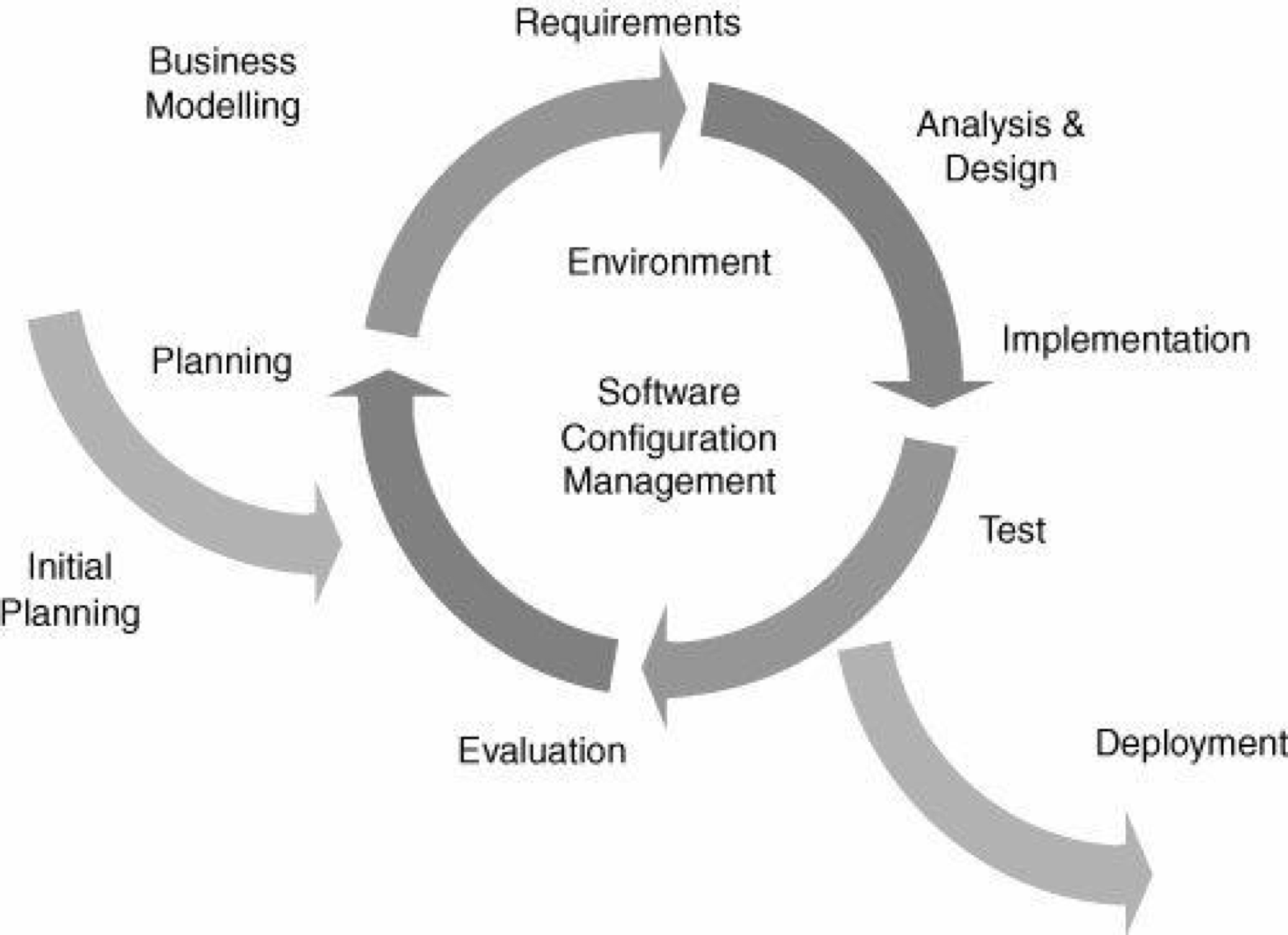
### 1.3 Project Risks

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Risk Description** | **Impact** | **Possibility** | **Response Plans** |
| 1 | Project team member do not meet deadlines | High | Medium | Other members support if the task is an important task |
| 2 | Data loss | High | Medium | Restore backed up data from GitHub.  If a requirement has a new update, all members have to join the meeting to be aware and make decisions. |
| 3 | Conflict between team members | High | High | Sit down and talk frankly |
| 4 | Requirement  changed | Medium | Medium | Discuss with the Supervisor to decide which requirements should be implemented.  Change requirement. Develop for new required func |
| 5 | Spirit goes down | Low | High | Communicate regularly to make people more close |
| 6 | Lack of skill and knowledge for a specified work | Medium | High | Learn more courses on coursera |
| 7 | Team members may distraction | High | High | Understand team members' schedule and assign suitable tasks and set appropriate deadlines.  Require team members to set high priority for the project. |
| 8 | Team members do not understand about requirements | Medium | High | Every members when join develop the project must join the develop for SRS |
| 9 | Lack of Supervisor support | Medium | Low | Define a meeting schedule with the Supervisor.  Ask for support from other sources. |
| 10 | The designed database may be a failure. | High | High | Team members researchers more about how to design a database.  Team members discuss and review the project database.  Ask the support from the supervisor or the database expert |
| 11 | Business problem | Medium | High | Make sure the business logic of any ideas is carefully analyzed. |
| 12 | Source code may be conflicted | Low | High | Pull source code before committing the source code.  Use the backup version, discuss with other members and continue to work. |

## 2. Management Approach

### 2.1 Project Process

Clinic management application is develop according to the following model:



*Figure 1: Software Process Model*

The Iterative and Incremental Software Process Model is mostly used when the scope of the project is big, the major requirements are defined clearly, some more details will be added later in software development. By using this software process model, we break down the developing system task into a series of smaller tasks which will be completed separately, allowing us to take advantage of what was learned during development of earlier parts of the system. In addition, the iterative model is easier than other models when the issues are discovered. They are fed back to the team, and solutions will be found while the project is still in development.

### 2.2 Quality Management

Some of the quality approach :

* **Defect Prevention:** Discuss with the team before implementing a feature for exceptional cases that could occur, use pair programming to handle defects in the coding phase.
* **Reviewing:** Apply reviewing at different level:
  + Self review: Member reviews his/her source code by himself/herself
  + Peer review: Each member reviews other members’ source code
  + Final review: Leader review source code of his/her team and merge if there are no problems left
* **Unit testing:** Create unit test for api provided.
* **Integration testing:** is the phase in software testing in which individual software modules are combined and tested as a group. Integration testing is conducted to evaluate the compliance of a system or component with specified functional requirements.
* **System testing :** is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to evaluate the end-to-end system specifications.

### 2.3 Training Plan

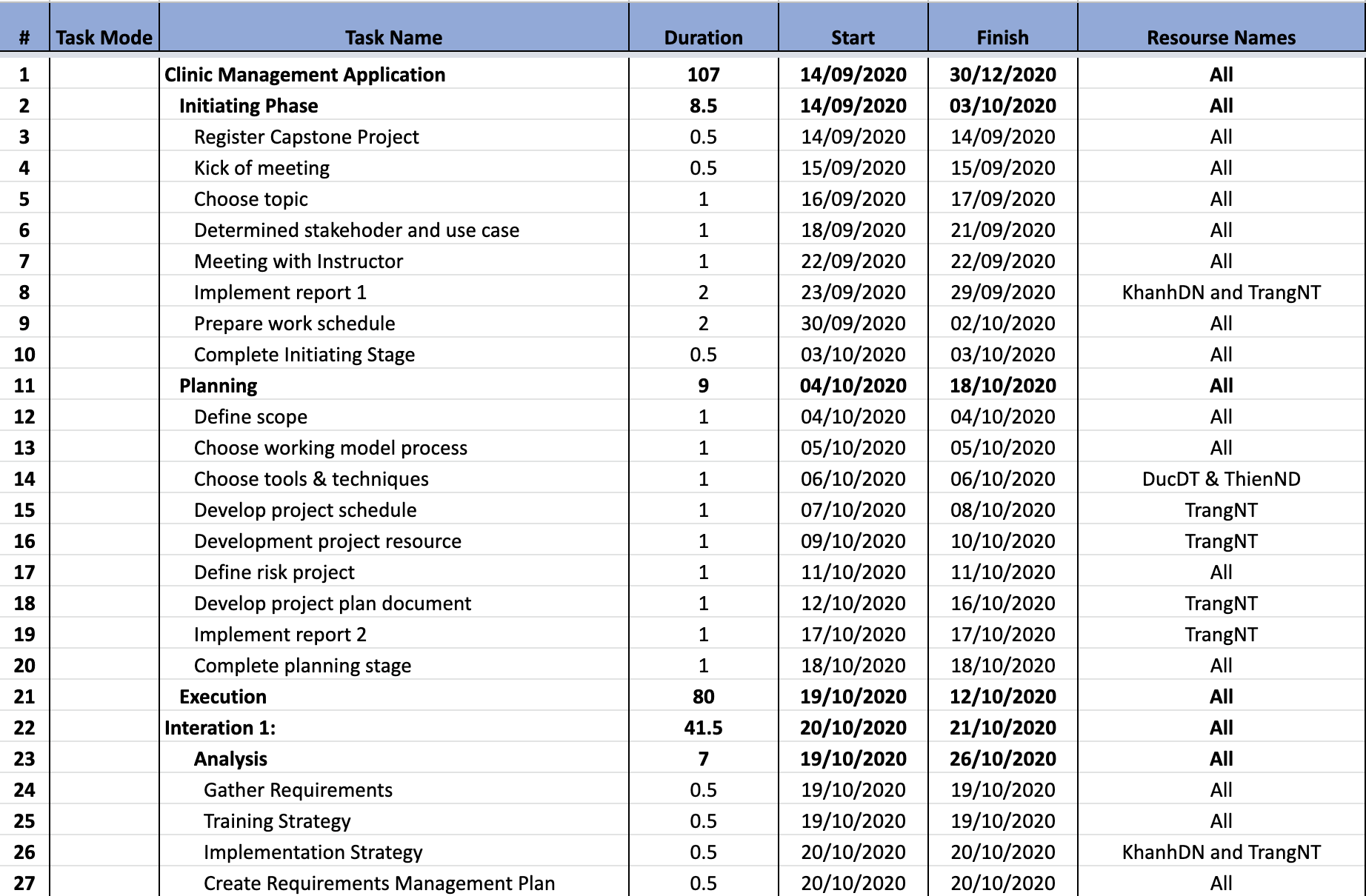
To be sure all members must have knowledge of code and members understand requirements of business. We have organized training for all members based on the plan below.

|  |  |  |  |
| --- | --- | --- | --- |
| **Training Area** | **Participants** | **When, Duration** | **Waiver Criteria** |
| Git, Github | All | 09/28, 1 hour | Mandatory |
| PostgreSQL | All | 09/28, 1 hour | Mandatory |
| Java Spring Boot | All | 09/29, 3 hour | Mandatory |
| Angular | All | 09/30, 4 hour | Mandatory |

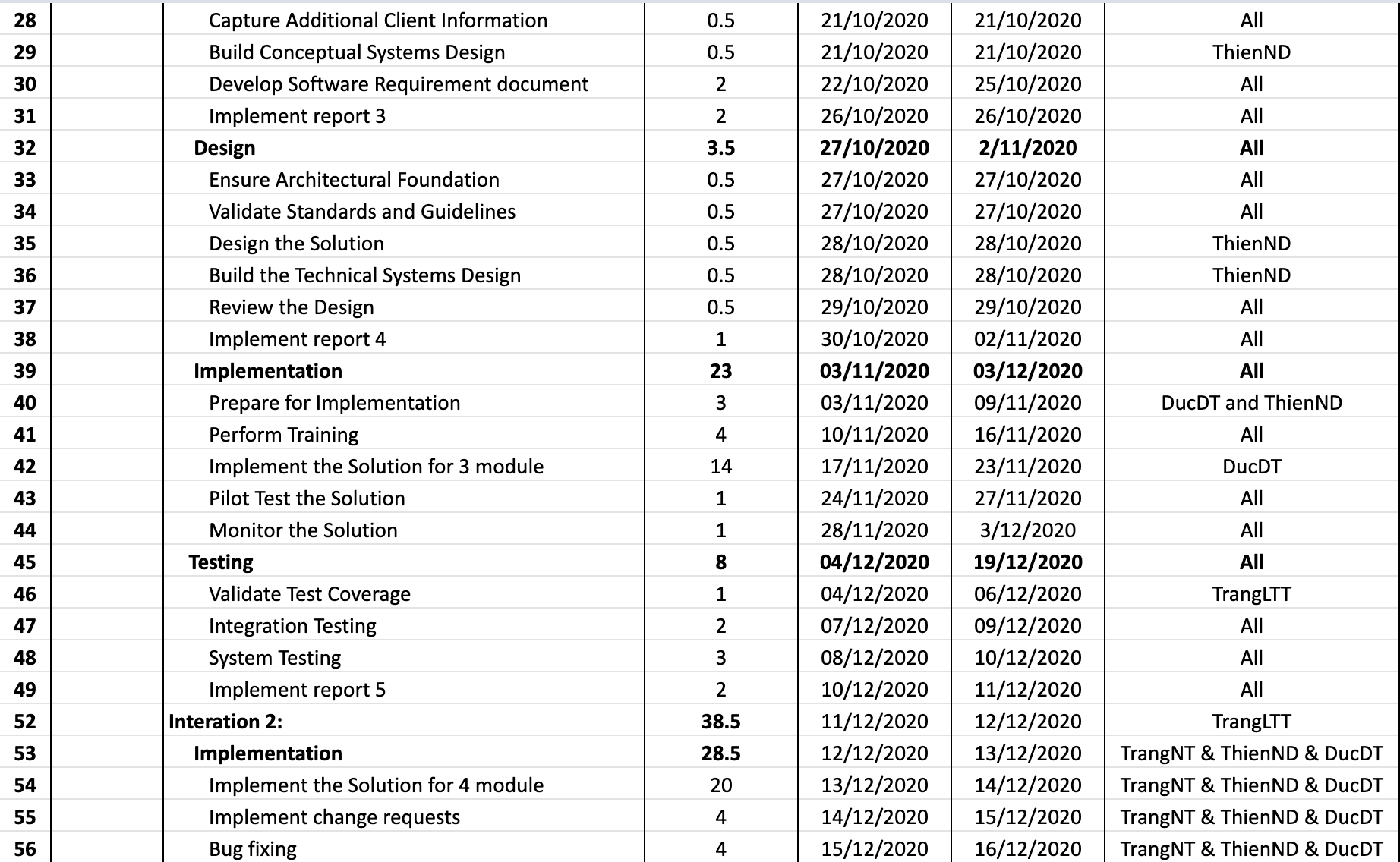
## 3. Master Schedule

Control objectives are checked according to the scheme's mode. We have posed a clear set of project’s schedule for completion deadlines.

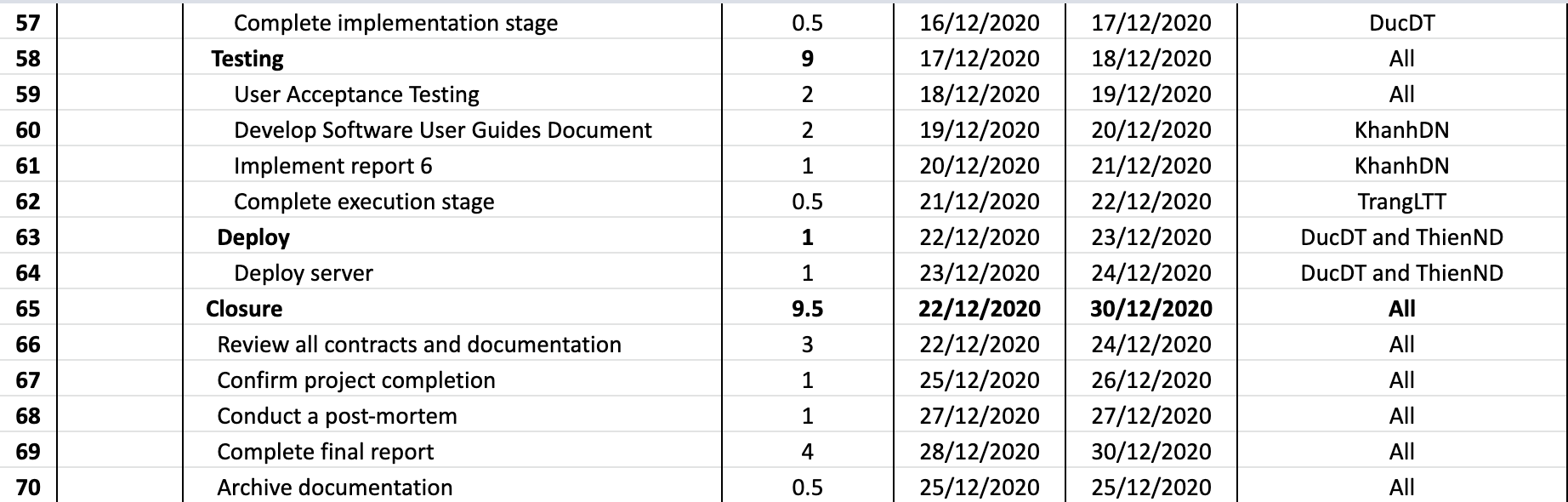
|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Deliverable** | **Due Date** | **Deliverable Scope** |
| 1 | Initiating Phase | 20 | The role of stakeholders.  Collect customer requests.  Define the objectives, scope and requirements of the project.  Project kick-off meeting. |
| 2 | Project Plan | 15 | Establish detailed WBS for the project.  Make a specific task list.  Scheduling. |
| 3 | SRS | 7 | Define Use cases, business rules, ERD . |
| 4 | Design | 7 | Architecture Design, Detailed Design, Database. |
| 5 | Implementation | 30 | Coding & Unit testing. |
| 6 | Testing | 18 | System testing, Integration Testing, Acceptance testing. |
| 7 | Deploy | 2 | Deploy on the server. |
| 8 | Closure | 10 | Solve project backlog.  Complete documentation project.  Handing over the project results to the end user. |



*Figure 2.1: CMA Schedule*



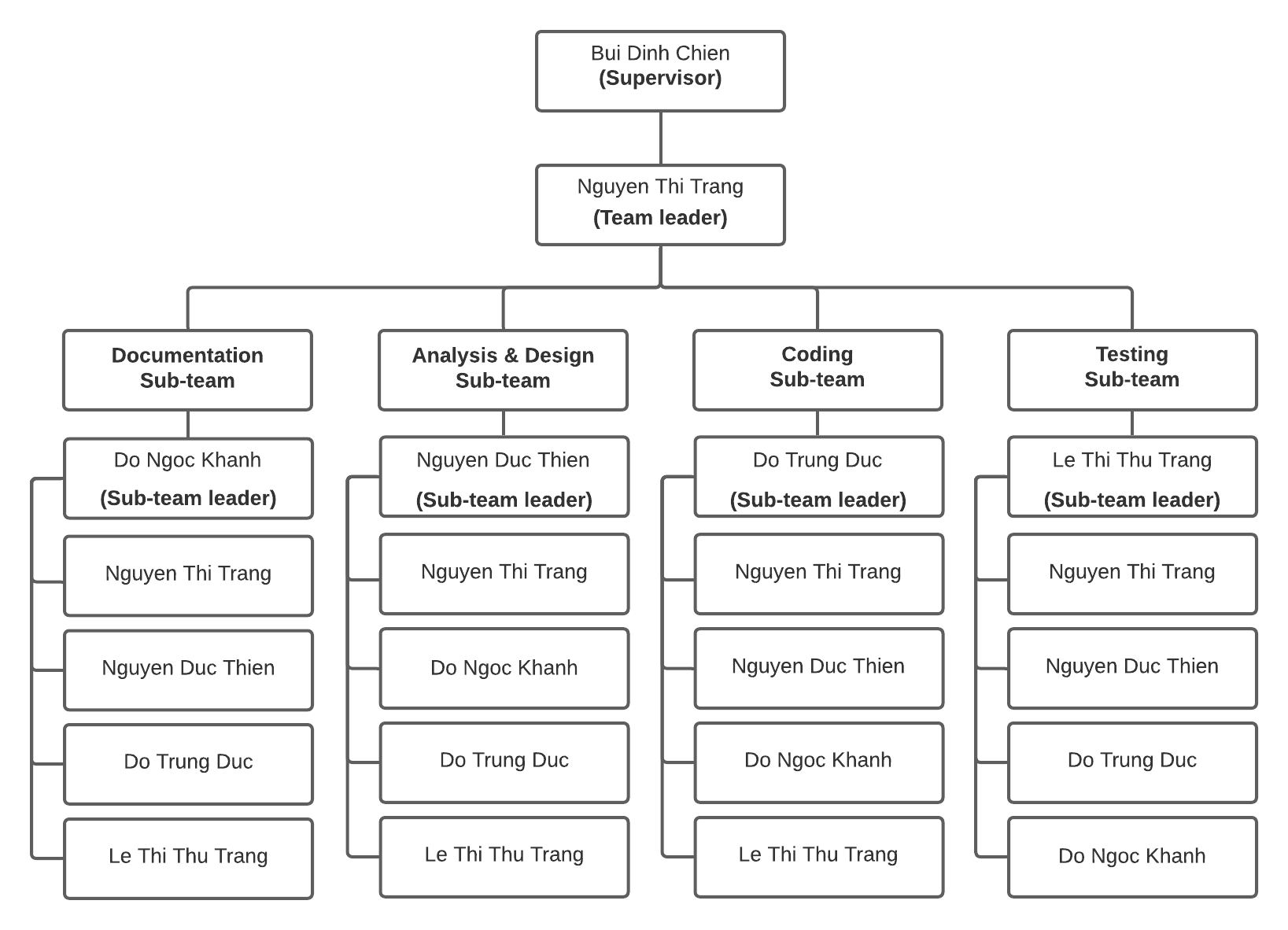
*Figure 2.2: CMA Schedule*



## 4. Project Organization

### 4.1 Team & Structures

CMATeam organization structure above includes 4 sub-teams: Documentation sub-team, Analysis & Design sub-team, Coding sub-team and Testing sub-team. Each sub-team has a sub-team leader and sub-team members:



*Figure 3: Team & Structures*

### 4.2 Roles & Responsibilities

This is the detail description about role and responsibility of each member in the organization structure:

|  |  |  |
| --- | --- | --- |
| **Role** | **Name** | **Responsibility** |
| Team leader | Nguyen Thi Trang | Have overall responsibility for the project and assign tasks to members.  Tracking member’s work.  Report working status to the instructor.  Managing reports and necessary documentation.  Controlling time management. |
| **Documentation Team** | | |
| Documentation Sub-team leader | Do Ngoc Khanh | Complete all of the documents.  Project Introduction.  Software Requirement Specification.  Software User Guides.  Presentation Slide. |
| Member #1 | Nguyen Thi Trang | Project Management Plan.  Presentation Slide. |
| Member #2 | Nguyen Duc Thien | Software Design Document. |
| Member #3 | Do Trung Duc | Help Software Requirement Specification. |
| Member #4 | Le Thi Thu Trang | Test plan document.  Installation guide. |
| **Analysis & Design Team** | | |
| Analysis & Design Sub-team leader | Nguyen Duc Thien | System Architecture design.  Screen design.  Sequence diagram.  Create use case specification. |
| Member #1 | Nguyen Thi Trang | Help create use case specifications.  Help Screen design.  Help database design.  Create a Class diagram. |
| Member #2 | Do Ngoc Khanh | Help ERD Diagram. |
| Member #3 | Do Trung Duc | ERD Diagram.  Database design.  Database Dictionary. |
| Member #4 | Le Thi Thu Trang | Use case specification.  Help Class diagram. |
| **Coding Team** | | |
| Coding Sub-team leader | Do Trung Duc | Decide technique and tools to use.  Train other members about web development.  Keeping track of development work done by other  coding team members.  Coding back-end & front-end.  Merge code.  Control source code. |
| Member #1 | Nguyen Thi Trang | Coding back-end. |
| Member #2 | Nguyen Duc Thien | Coding front-end.  Deploy web application to host server.  Help Decide technique and tools to be used. |
| Member #3 | Do Ngoc Khanh | Coding back-end. |
| Member #4 | Le Thi Thu Trang | Coding front-end. |
| **Testing team** | | |
| Testing Sub-team leader | Le Thi Thu Trang | Create a test plan.  Responsible for the test plan.  Controlling testing activities.  Create test reports.  Manage test resources and assign test tasks Create unit test and bug log.  Create test cases (Acceptance test).  Report test result.  Implement test cases. |
| Member #1 | Nguyen Thi Trang | Help create test case (Integration test, System test,  Acceptance test). |
| Member #2 | Nguyen Duc Thien | Help create test case (Integration test, System test,  Acceptance test).  Help implement test cases. |
| Member #3 | Do Trung Duc | Help create test cases (Acceptance test).  Help implement test cases. |
| Member #4 | Le Thi Thu Trang | Help create test cases (Integration test, System test, Acceptance test). |

## 5. Project Communication

### 5.1 Communication Plan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Communication Item** | **Who/ Target** | **Purpose** | **When, Frequency** | **Type, Tool, Method(s)** |
| Weekly meeting schedule | All member and Supervisor | Assign tasks.  Solve the problems.  Review report. | Slot 2 wednesday, once per week | Offline |
| Daily meeting schedule | All member | Check the progress of groups | 20h, everyday | Google meet |
| Unscheduled meeting | All member | Handle important work |  | Google meet, offline. |
| Training | All member | New training of knowledge for members | Slot 2 & 3, monday and wednesday and friday. | Offline, google meet. |
| Communication channel | All member | Assign tasks in the team  Means of communication, work management. | Everyday | Trello, facebook group, phone, google meet. |

### 5.2 External Interface

#### a. FU Contacts

|  |  |  |  |
| --- | --- | --- | --- |
| **Function** | **Contact Person**  **(name, position)** | **Contact address**  **(email, telephone)** | **Responsibility** |
| Supervisor | Bui Dinh Chien | ChienBD@fe.edu.vn  sdt….. | Provide document template  Give instruction to project team  Review deliverables  Supervise project status |
| Supporter | Ngo Thi Vinh Ha | HaNTV2@fe.edu.vn | Provide document template  Receive report project  Answer questions about the project |

#### b. Customer Contacts

|  |  |  |  |
| --- | --- | --- | --- |
| **Function** | **Contact Person**  **(name, position)** | **Contact address**  **(email, telephone)** | **Responsibility** |
| Supporter | Nguyễn Trần Kiên | 096.159.1593 | Support for the research system of business and analysis exist. |
| Customer | Nguyễn Thị Hạnh | 091.648.3289 | Acceptance of products and supply requirements. |

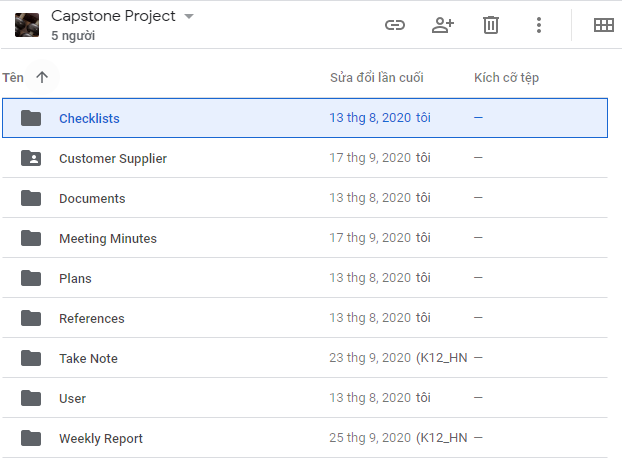
## 6. Configuration Management

### 6.1 Tools & Infrastructures

|  |  |
| --- | --- |
| **Programming languages** | Java, Typescript |
| **Framework** | Java Spring Boot, Angular |
| **DBMS** | PostgreSQL |
| **IDEs/Editors** | Visual Studio Code, Eclipse |
| **UML tools** | Lucidchart.com |
| **Version Control** | Github, Google drive |
| **Deployment server** | Google compute engine, Docker |
| **Project management tool** | Trello.com |
| **Mockup design tool** | Draw.io, figma.com |

### 6.2 Document Management

Document will be managed through Google Drive. We have created shared memory that has the following directory structure:



* **Checklists**: Store checklist files (coding review, test review, coding review, ...)
* **Customer Supplier**: Contains files, documents, images provided by customer
* **Documents**: Contains all document files of the project (reports, design documents, api documents, test documents, ...)
* **Meeting Minutes**: containing Meeting Minutes report files weekly
* **Plans:** Contains document files related to project planning
* **References**: Contains reference document files
* **Take Note**: Take note files after each team meeting
* **User**: Include subfolders for each member. Each folder will contain separate documents or work in progress
* **Weekly Report**: Containing Weekly Report document

### 6.3 Source Code Management

Source code will be managed through Github

* We have two main branches with an infinite lifetime:
  + Main: to be the main branch where the source code of HEAD always reflects a production-ready state.
  + Develop: to be the main branch where the source code of HEAD always reflects a state with the latest delivered development changes for the next release
* Each Feature will be deployed to work on a branch  
  Feature branch are used to develop new features for the upcoming or a distant future release  
  May branch off from: develop  
  Must merge back into: develop  
  Branch naming convention: anything except master, develop, \*-feature
* Hotfix branches are very much like release branches in that they are also meant to prepare for a new production release, albeit unplanned  
  May branch off from: master  
  Must merge back into: develop and master  
  Branch naming convention: \*-hotfix

# III. Software Requirement Specification

## 1. Overall Description

### 1.1 Product Overview

*[This section presents a high-level overview of the product and the environment in which it will be used, the anticipated users, and known constraints, assumptions, and dependencies]*

### 1.2 Business Rules

*[Provide common business rules that you must follow. The information can be provided in the table format as the sample below]*

## 2. User Requirements

*[Provide specification of the user requirement: actor, use cases, use case diagram, use case specification, etc.* ***These contents are optional****, it can be ignored if the functional requirements are clear enough]*

## 3. Functional Requirements

*[Provide specification of software system: screen, functions, system data, diagrams, functional specification, etc.]*

### 3.1 System Functional Overview

### 3.2 <<Feature Name 1>>

#### a. <<Function Name 1>>

*[A function can be a screen or a non-screen function (listed in the part 5.1 above). In this part, you need to provide the details on the related function, focus on mentioning below information*

* *Function trigger: how this function is triggered (navigation path, a timing frequency, etc.*
* *Function description: actors/roles, purpose, interface, data processing, etc.*

*Screen layout: mockup prototype of the screen*

## 4. Non-Functional Requirements

### 4.1 External Interfaces

*[This section provides information to ensure that the system will communicate properly with users and with external hardware or software elements.]*

### 4.2 Quality Attributes

*[List all the required system characteristics (quality attributes) specification]*

## 5. Other Requirements

*[List out other requirements, appendix information etc. in this part]*

# IV. Software Design Description

*[Provide final software design information follow the template as part II in the Report #4]*

## 1. Overall Description

### 1.1 Assumptions

*[Describe any items, based on which to design future system]*

### 1.2 Design Constraints

*[Describe any global limitations or rules that have a significant impact on the design of the system's software (and describe the associated impact]*

### 1.3 [Technology Suggestion]

*[This section is optional. If this section isn’t set, remove it out of the document]*

## 2. System Architecture Design

### 2.1 Overall Architecture

*[Select suitable architecture style and describe the architectural diagram in the relationship/connection to the external systems. The content of this section include the overall diagram and the explanation for each of the diagram components]*

### 2.2 System Architecture

*[Select suitable architecture patterns for sub-systems. The content of this section include the overall diagram and the explanation for each of the diagram components]*

### 2.3 Package Diagram

*[Provide the package diagram for each sub-system. The content of this section include the overall package diagram(s) and the explanation for each package]*

## 3. System Detailed Design

### 3.1 <Feature Name1>

*[Provide the detailed design for the feature <Feature Name1>. It include Class Diagram, Class Specifications, and Sequence Diagram(s)]*

#### a. Class Diagram

*[This part presents the class diagram for the relevant feature]*

#### b. Sequence Diagram(s)

*[Provide the sequence diagram(s) for the feature]*

## 4. Class Specification

*[Provide the description for each class, including Class Methods information]*

## 5. Data & Database Design

### 5.1 Database Design

*[Provide the tables relationship and table specifications]*

### 5.2 Data File Design

*[List out the data files and file formats which are used in the system]*

# V. Software Testing Documentation

*[Provide final software testing information follow the template as part II in the Report #5]*

## 1. Overall Description

### 1.1 Test Model

*[Provide the testing model for your project which fits to your selected software development process model, i.e V-Model, Iterative, etc.]*

### 1.2 Testing Levels

*[List out and describe all testing levels that would be performed in the project: unit testing, integration testing, system testing, acceptance testing (alpha, beta),..]*

### 1.3 Testing Types

*[List out and describe all testing levels that would be performed in the project: functional testing, non-functional testing, structural testing, changes testing, etc.]*

## 2. Test Plan

### 2.1 Test Stages

*[Clearly state the stage in which the test will be executed. Identified below are the stages in which common test are executed]*

### 2.2 Resources

#### a. Human Resources

*[List and provide the details on roles and responsibilities of the project members who would involve in testing works]*

#### b. Environment

*[List and provide the details about the tools (software, hardware, infrastructure) which the project would use for testing]*

### 2.3 Test Milestones

*[Separate test milestones, which should be identified to communicate project status accomplishments]*

### 2.4 Deliverables

*[Define deliveries of Testing. Refer to the Project Management Plan for more details]*

## 3. Test Cases

*[Prepare the details on the test cases following the provided template*

* *Unit Test Cases: Report5\_Unit Test Case.xls*
* *Other Test Cases: Report5\_ Test Case.xls]*

## 4. Test Reports

*[Provide the test result, statistics and the relevant test analysis for your testing in the project]*

# VI. Release Package & User Guides

*[Provide final software testing information follow the template as part II in the Report #6]*

## 1. Deliverable Package

### 1.1 Source codes & documents

*[The section will list all source programs/classes and documents with version number in this release. You can see the example following table for reference, can customize or delete if not using belong to each project characteristics]*

### 1.2 Known Issues, Limitations & Restrictions

*[List out all known (pending) issues, limitation and restrictions in the final deliverable package]*

## 2. Installation Guides

### 2.1 System Requirements

*[Define any system requirements necessary to support the application. These can include the supported host operating systems and network platforms, configurations, memory, peripherals, and companion software.]*

### 2.2 Setup Files

*[List of installation scripts and setup files]*

### 2.3 Installation Instruction

*[Includes installation instructions and configuration guidelines]*

## 3. User Manual

### 3.1 Terms and definitions

*[Detail about terms and definitions used in this user guide.]*

### 3.2 System requirements

*[Define any system requirements necessary to support the application. These can include the supported host operating systems and network platforms, configurations, memory, peripherals, and companion software(s).]*

### 3.3 Application Usage

*[Provide the guides to use each of the feature of the application or software system]*

### 3.4 Troubleshooting

*[Descript some common troubles while using application.]*

# VII. Appendix

## 1. Glossary [Optional]

*[Place all definitions or abbreviation used in this document]*

## 2. References [Optional]

*[Place all referenced materials used in this document]*

## 3. Others [Optional]

*[Provide any other information here…]*