

**Table of Contents**

[**I.**](#_heading=h.gjdgxs) **Executive Summary 2**

[**II.**](#_heading=h.30j0zll) **Competitive analysis 3**

[**III.**](#_heading=h.1fob9te) **Data definition 3**

[**IV.**](#_heading=h.3znysh7) **Overview, scenarios, and use cases 3**

[**V.**](#_heading=h.2et92p0) **Initial list of high-level functional requirements 3**

[**VI.**](#_heading=h.tyjcwt) **List of non-functional requirements 3**

[**VII.**](#_heading=h.3dy6vkm) **High-level system architecture 3**

[**VIII.**](#_heading=h.1t3h5sf) **Team and checklist 3**

[**IX.**](#_heading=h.4d34og8) **References 3**

# Executive Summary

As of now, the COVID-19 pandemic has impacted every area of our lives. Therefore, we cannot have the same routine as we were used to when we must distance ourselves and limit the exposure to others as much as possible. The number of COVID-19 patients is increasing dramatically to the point that some hospitals are struggling to keep up with the rising demand for beds and shortage of staff. Illness severity can range from mild to critical. All deaths usually occurred among patients with critical illness while patients with mild symptoms can be treated at home with the doctor’s instructions. Speaking of it, we would like to introduce our application to ensure everyone with COVID-19 symptoms can get safe home care and prevent the spread of coronavirus through close contact by going to a hospital while they can receive certified doctor’s instructions via a website application called CyberHealth.

CyberHealth is an online health monitoring platform where patients with COVID-19 symptoms can upload their medical history as well as vital signs in order to receive treatment from certified doctors. Licensed doctors will determine if you should be tested or not and provide timely care based on the illness severity. Our virtual care application commits to deliver a seamless user experience, responsive web design, and a user-friendly interface to all users regardless of patients or doctors.

# 

# II. Competitive analysis: (Huy - in process)

|  | CyberHealth | Teladoc | Mdlive | Mymdnow | Private Offices |
| --- | --- | --- | --- | --- | --- |
| Online visit | 5 | 5 | 5 | 5 | 0 |
| Fast & simple process | 5 | 4 | 4 | 3 | 2 |
| User interface | 5 | 5 | 3 | 5 | 0 |
| Fast access to covid’s doctors | 5 | 4 | 4 | 2 | 2 |
| Fast communication | 5 | 4 | 4 | 2 | 2 |
| Process tracking | 5 | 4 | 4 | 1 | 1 |
| **Mean** | 5 | 4.3 | 4 | 3 | 1.2 |

Private Offices(1.2):

* Private offices are the local and offline doctors offices. Patient usually follow one doctor for a long time, so the doctor is able to know the patients as well as their medical record. However, patient sometime need go through a long process from phone call, making appointment, be in the office, filling paperworks before they can see the doctor. In addition, since patients have to physically be in the office, the covid explosure is much higher. Furthermore, there are not many specific covid doctor office available locally.

Teladoc(4.3):

* Teladoc is a platform to connect doctor and patient by phone. They provide a good user interface and process. Since they are online service, they also eliminate the covid exposure. On the other hand, Teladoc focus on general healthcare, so there is a longer and more complicated process for patients to access to covid care.

Mdlive(4):

* Mdlive is a virtual doctors platform, which allow patient to see doctor by phone call, coumputer, or app. It provides many healthcare services. However, since they provide many services, not focus on covid, the process is longer and more complicated to access to covid care. Furthermore, their user interface is not eye-catching.

Mymdnow(3):

* Mymdnow is the information and booking website for MDnow urgentcare. The website is clean and informative. It provide good infomation for patients and allow them to make appointments. However, patient still need to physiclly be in the office, this can increase the covid exposure. In addition, MDnow is also a general healthcare provider, so it is a longer and complicated process to access to covid care. Futhermore, it’s a physically office, so patients will need to go through a long process from driving, filling paperworks, waiting for their turn.

CyberHeath(5):

* CyberHealth is a platform connecting patients and doctors, focusing on Covid care services. Even though this platform doesn’t provide a wide range of healthcare services, it is laser focus in Covid care to create a great experience for users. CyberHealth has a great clear, clean, and eye-catching user interface. It also provide a fast, simple, and effective process for both Covid patients and doctors. In addition, it allow patients to quickly access Covid treatment and medication. Futhermore, CyberHealth let patients and doctors effectively communicate and get updated on the treatment process.

Planned advantages:

At CyberHealth, since we are laser focus on Covid care, it gives us any advantages compare to our competitor. First, we are online service, so we can eliminate the Covid exposure to others and help patient access to doctors anywhere and anytime, even if they are sick. Second, we are focus on Covid care only, so we can make the process simpler, faster, and more effective for both patients and doctors. It also allow us to reduce the complexity of the system and focus on user experience and interface. Third, we provide many great and unique features such as process tracking, communication box, and saved treatments and prescription depend on condition. Those allow users to communicate effectively, get clear updated on their process, save time, and have a much better experience.

# Data definition

(Nha Tran: I will create a table for this part, but for now I will write it like this. Feel free to add any ideas on this.)

1. Home – User interface – The introductory page of a website

2. Service – User interface – List what we offer for users

3. Sign in – User service – Existing patients can upload their health information and see their treatment while doctors can see their patients and provide treatment

4. Register – User service – Allow new patients to create an account

5. User - Actors - Refer to both patients and doctors

6. Member - Actors - An user who has an account with the system

7. Non-member - Actors - An user who does not have an account with the system

8. Admin - Actors - An user can see doctor’s information as well as patient’s information

9. Profile - Use case scenarios - Store user’s information

10. Status - Use case scenarios - Display the illness severity of patients

11. Comments - User case scenarios - Leave a note

# Overview, scenarios, and use cases

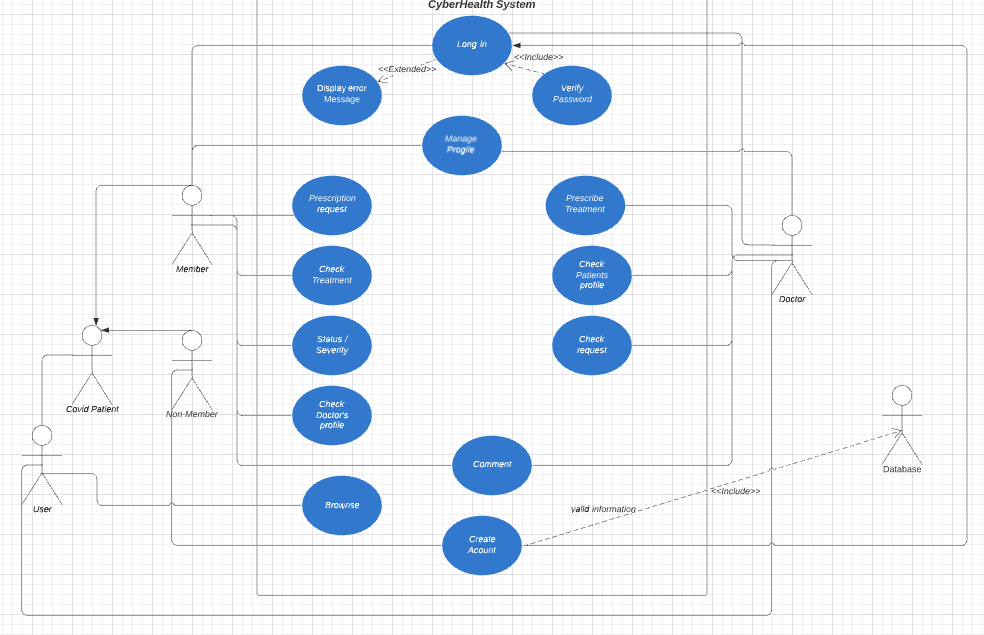
(Nelly: still in progress, feel free to add)

User Stories:

1. As a user, I want to browse the homepage and services to learn what cyberhealth provides.
2. As a member, I want to create or update my information so that my profile can be up to date.
3. As a member, I want to ask for prescription requests.
4. As a member, I want to check the treatment my doctor has prescribed.
5. As a member, I want to view my doctor's information.
6. As a member, I want to comment on my treatment, doctor, and progress.
7. As a member, I want to state my health status.
8. As a doctor, I want to create or update my information so that my profile can be up to date.
9. As a doctor, I want to check my patient's information to know what to prescribe them.
10. As a doctor, I want to give the patient treatment.
11. As a doctor, I want to check treatment requests so that I may prescribe treatment to these patients.
12. As a doctor, I want to check my patient's status so that I can follow their progress.
13. As a doctor, I want to comment on my patient's condition and treatment to help the patient.
14. As a non-member, I want to create an account so that I may receive cyberhealth services.

Use case diagram:

* please comment or add anything that might be missing from this diagram



# Initial list of high-level functional requirements

(Nha Tran: The formatting for this part I will do later. Right now I will focus on the contents. Feel free to edit anything.)

**Non-member**:

1. Browse the main homepage

* User can use the navigation bar and interact with the Home page and Service page buttons
* The system shall let this user see the contents of the main Home page and Service page

1. Create an account

* Users can create an account by entering their information such as username, email address, password, first name, last name, date of birth, phone number, and location
* The system shall validate if the username is registered or not
* The system shall validate all the information that users entered in the correct format
* The system shall make sure there is no empty field when users create their account
* The system shall display a message if an account is successfully created and redirect the user to the Sign in page.
* The system shall store user’s information provided above
* The system shall provide unique ID for each user

**Member:**

1. Sign in

* Users can sign in if their username and password are registered with the system
* Doctors can log in with their registered account
* Patients can log in with their registered account
* Admin can log in with account provided by developers
* The system shall let users with registered account to log in
* The system shall not let users with unregistered account to log in
* The system shall display the error message when users log in with incorrect credentials
* The system shall offer a register button in the sign in page

1. Edit Profile

* Users can see their interface depending on which role they are logging in
* Doctors can see their own interface and edit their information such as first name, last name, date of birth, password, phone number, location, and email address
* Patients can see their own interface and edit their information such as first name, last name, date of birth, password, phone number, location, and email address
* The system shall let users see and edit their information depending on their roles
* The system shall let doctors see their information and edit it
* The system shall let patients see their information and edit it
* The system shall validate all the information that users entered in the correct format
* The system shall make sure there is no empty field when users edit their profile
* The system shall let users save information that they edited and store it

1. View main page

* Doctors can see their information and patient’s information
* Patients can see their information and doctor’s information
* Admin can see both doctor and patient’s information
* The system shall let users browse their interface depending on their roles
* The system shall restrict what each role should see

1. Manage as a doctor

* Doctor can prescribe treatment to patients
* Doctor can check patient’s information
* Doctor can update status of patients
* Doctor can see a list of patients
* Doctor can check request from patients
* The system shall have a navigation bar
* The system shall have buttons for doctors to click on to prescribe treatment to patients
* The system shall have buttons for doctors to click on to check patient’s information
* The system shall have drop-down menu for doctors to click on to update status of patients
* The system shall have a list for doctors to see patients
* The system shall have a list for doctors to see the requests from patients
* The system shall let doctors browse and interact with buttons, list, etc

# List of non-functional requirements: (HN - Done - Feel free to add)

1. Performance: loading time should not exceed 1 second for users
2. Reliability: users can access the website 98% of the time without failure.
3. Recoverability: if problems happen to the website, it should be recovered no more than three days for major ones and 8 hours for minor ones.
4. Security \*\*\*: Only admin can view doctors and patients information. Only doctors can view a patient's medical record.
5. Compatibility: the website must works on multiple browsers(chrome, safari, etc) and devices(tablet, phone, laptop, ect)
6. Usability\*\*\*: the website must be user-friendly and prioritize user experience.
7. Data integrity\*\*\*: the system must keep all doctors and patient data secure and fully back-up for every record.
8. Easy to use: the website should be easy to use to any users, even in their first visiting.

\*\*\*: prioritize requirement.

# High-level system architecture: (Huy - Done so far. Add anything.)

1. Lamp server: we will hosting our Fall 2021 project on <https://lamp.cse.fau.edu/~cen4010_fa21_g22/>
2. WhatsApp: is the main communication tool for us.
3. Jira: we will use Jira to track our project process and tasks.
4. Github: will be used to building and editing our project together.
5. MySQL database: will be the main database for the project.
6. Languages:
   1. HTML: will be used to display all the documents on the website.
   2. CSS: will be used to decorate the website and pages.
   3. PHP: will be used to manage data and contents.
   4. Javascript: will be used for most back-end development of the project.
   5. Bootstrap: will be used for the initial construction for the website.

# Team and checklist

Group name: 22

Scrum master:

Product owner:

Front End Developer:

Back End Developer:

​​a) Team decided on basic means of communications

b) Team found a time slot to meet outside of the class

c) Front and back end team leads chosen

d) Github master chosen

e) Team ready and able to use the chosen back and front-end frameworks

f) Skills of each team member defined and known to all

g) Team lead ensured that all team members read the final M1 and agree/understand it before submission

# References

<https://winatalent.com/blog/2020/05/what-are-non-functional-requirements-types-and-examples/>