COVID-19 Vaccination Record Software

Test Plan

By

Yanhang Yang 622115513

Jun Zhang 632115505

Haizhao Li 622115504

Zhuoyuan Li 622115515

Suchanun sirijanya 622115040

Department of Software Engineering

College of Arts, Media and Technology

Chiang Mai University

Project Advisor

Jayakrit Hirisajja

Document History

History	Status	Date	Viewable	Editable	Responsible
Add • Introduction • Test Plan and Test Procedure	Draft	March 2nd2023	YYH, JZ, HZL,ZYL,SS JH	YYH, JZ, HZL,ZYL,SS	YYH
Add • Unit test • System test	Draft	March 3th 2023	YYH, JZ, HZL,ZYL,SS JH	YYH, JZ, HZL,ZYL,SS	YYH, JZ, HZL,ZYL,SS
Update • Unit test • System test	Draft	March 4th 2023	YYH, JZ, HZL,ZYL,SS JH	YYH, JZ, HZL,ZYL,SS	YYH, JZ, HZL,ZYL,SS
Update • Unit test • System test	Draft	March 5th 2023	YYH, JZ, HZL,ZYL,SS JH	YYH, JZ, HZL,ZYL,SS	YYH, JZ, HZL,ZYL,SS
Add • Integration test	Draft	March 6th 2023	YYH, JZ, HZL,ZYL,SS JH	YYH, JZ, HZL,ZYL,SS	YYH, JZ, HZL,ZYL,SS
Update • Integration test	Draft	March 8th 2023	YYH, JZ, HZL,ZYL,SS JH	YYH, JZ, HZL,ZYL,SS	YYH, JZ, HZL,ZYL,SS

* YYH= Yanhang Yang
* JZ= Jun Zhang
* HZL= Haizhao Li
* ZYL= Zhuoyuan Li
* SS=Suchanun sirijanya
* JH= Jayakrit Hirisajja

Contents

Introduction

1. Objective

This document is the Test Plan. It helps us determine the effort needed to validate the quality of the application under test. The test plan serves as a blueprint to conduct software testing activities as a defined process, minutely monitored and controlled by the test manager.

In addition, it should also have the following objectives:

- To find defects that may get created by the
- To make sure that the result meets the business and user requirements.
- To ensure that it satisfies the SRS that is System Requirement Specifications.

2. Testing method

<table>
<tr><td rowspan="2">Unit Test</td><td colspan="2">White-box testing</td></tr>
<tr><td colspan="2">Automatic test (Jest framework)</td></tr>
<tr><td rowspan="2">System Test</td><td rowspan="2">Blackbox testing</td><td>Function test</td></tr>
<tr><td>GUI test</td></tr>
</table>

3. Intended Audience

The mainly intended audience of this document is all significant stakeholders, which include the development team, the project owner, testers, the senior project advisor (Dr. Jayakrit Hirisajja), and anyone evaluating the project.

4. Project Scope

develop a web-based COVID-19 vaccination tracking application include a front-end user interface that displays a list of vaccinated individuals, their names, surnames, and vaccine status. When a user clicks on a name, they can view the individual's personal information, including name, surname, age, and hometown, along with their vaccination details. The back-end system will handle user authentication and access permissions, with admin users having full access to patient data and the ability to add vaccine information, while regular users can only view their own data. Doctors can add comments to their patients' profiles, and the system includes a registration page for new users. Admins can designate users as doctors or regular users, and doctors can view and add comments to their patients' profiles.

5. Acronyms

SRS - Software Requirement Specification

URS - User Requirement Specification

UC - Use Case

UI - User Interface

UTC - Unit Test Case

STC - System Test Case

IT- Integration test

6. Definition

Name	Definition
Feature	Transformation of input parameters to output parameters based on a specified algorithm. It describes the functionality of the product in the language of the product. Used for requirements analysis, design, coding, testing or maintenance. [IEEE90]
IEEE	Institute for Electrical and Electronics Engineers. Biggest global interest group for engineers of different branches and computer scientists. [IEEE90]
Requirement	(1) A condition or capability needed by the user to solve a problem or achieve an objective. (2) A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document. (3) A documented representation of a condition or capability as in definition (1) or (2). [IEEE90]
Specification	Precise description of an activity or work product that serves as the basic or input for further activities or work product. A specification can comprise requirements for a product and how they will be solved. Different parts of a specification (e.g., what is to be done, how it will be done) must not be mixed. [IEEE90]
Use case	(1) Concept to describe a system based on usage of system resources by its environment. Characterized by an objective-set of interactions within and at the borders of that system. (2) Notation from UML for describing a scenario (Usage approach, operational scenario) from the perspective of this user. [IEEE90]

Test Plan and Test Procedure

1. Test Objectives

The objectives of Vaccination Record Software System are:

1. All bugs or defects are detected
2. Those bugs or defects are fixed.
3. Functionalities and user interface covered the requirements.
4. All functionalities and features must be there.

Scope of Testing

Vaccination Record Software System will be tested the unit testing and system testing and record the test results in the test record.

Test Duration

Progress	Date and duration
Term project Progress Report	**Perform date:** March 1 – 10 2023 **Duration:** 10days

Test Responsibility

Item	Responsibility
Test unit test of web application	YYH
Record unit test of web application	YYH
Test system test of web application	ZYL,SS
Record system test of web application	ZYL,SS
Test Integration test of web application	JZ,HZL
Record Integration test of web application	JZ,HZL

Test Strategy

COVID-19 Vaccination Record Software System test will be following by:

1. Design test case for each feature.
2. Prepare test data for each feature.
3. Determine expected result.
4 . Perform testing on individual features.
5. Result of testing will be recorded.

2. Result of Testing

In the test record, the test result will separate into two parts, which are:

1. Actual output: The actual outputs that are performed by each test case.
2. Pass/Fail criteria:
   - Pass: the result of actual result is the same as expected result
   - Fail: the result of actual result is not the same as expected result.

3. Test Environment

Laptops

MacBook Pro (M1, 2022)

Processor: Apple M1 chip with 8-core CPU and 8-core GPU
Memory: 8GB
Graphics: 8-core GPU
Operating System: macOS Big Sur

ROG Q533Q

Processor: Intel Core i7-1165G7 Processor, 2.8 GHz
Memory: 32 GB DDR4 3200MHz SDRAM
Graphics: NVIDIA GeForce RTX2070 , with 8GB GDDR6 VRAM
Operating System: Windows 10 Home

Internet

Unit test

Unit Test Case 001 (UTC-001):

Method name: getVaccine(@PathVariable("id") Long id)

Description: This unit test is for the getVaccine method in the VaccineController class. It tests whether the method returns the expected result when given a valid or invalid ID.

Data for testing:

ID	Valid or Invalid
1	Valid
100	Invalid

Test Case：

ID	Description	Input	Expected Result
1.	Enter vaccine with ID to test whether the ID can return the correct vaccine	1	The method should return a ResponseEntity with the VaccineDTO for the ID 1
2.	Enter vaccine with ID to test whether the ID can return the correct vaccine	100	The method should return a "The given id is not found"

Unit Test Case 002 (UTC-002):

Method name: getDoctor(@PathVariable("id") Long id)

Description: This code is a method for getting a doctor by their ID in the system. This unit test is to ensure that the method returns the expected result when given a valid or invalid ID.

Data for testing:

ID	Valid or Invalid
1	Valid
32	Invalid

Test Case：

ID	Description	Input	Expected Result
1.	Enter doctor with ID to test whether the ID can return the correct doctor	1	The method should return a ResponseEntity with the DoctorDTO for the ID 1
2.	Enter doctor with ID to test whether the ID can return the correct doctor	32	The method should return a "The given id is not found"

Unit Test Case 003 (UTC-003):

Method name: getPatient(@PathVariable("id") Long id)

Description: This code is a method for getting a patient by their ID in the system. This unit test is to ensure that the method returns the expected result when given a valid or invalid ID.

Data for testing:

ID	Valid or Invalid
1	Valid
88	Invalid

Test Case：

ID	Description	Input	Expected Result
1.	Enter patient with ID to test whether the ID can return the correct patient	1	The method should return a ResponseEntity with the PatientDTO for the ID 1
2.	Enter patient with ID to test whether the ID can return the correct patient	88	The method should return a "The given id is not found"

Unit Test Case 004 (UTC-004):

Method name: Authority.builder().name( ).build()

Description: It is used to detect different login user identities, thus giving users different permissions

Data for testing:

Email	User identity
admin@admin.com	admin
enabled@user.com	user
doctor@doctor.com	doctor

Test Case：

ID	Description	Input	Expected Result
1.	After entering the email, the system will detect and assign different user permissions	.email("admin@admin.com")	.name(AuthorityName.ROLE_ADMIN)
2.	After entering the email, the system will detect and assign different user permissions	.email("enabled@user.com ")	.name(AuthorityName.ROLE_DOCTOR)
3	After entering the email, the system will detect and assign different user permissions	.email("doctor@doctor.com ")	.name(AuthorityName.ROLE_USER)

Unit Test Case 005 (UTC-005):

Method name: addCommentPatient(@RequestBody Patients patient)

Description: Test whether the function that only doctors can leave messages runs successfully

Data for testing:

Email	User identity
admin@admin.com	admin
enabled@user.com	user
doctor@doctor.com	doctor

Test Case：

ID	Description	Input	Expected Result
1.	Use the login account to enter the website respectively, and then check whether the patient can leave a message	.email("admin@admin.com")	null
2.	Use the login account to enter the website respectively, and then check whether the patient can leave a message	.email("enabled@user.com ")	null
3	Use the login account to enter the website respectively, and then check whether the patient can leave a message s	.email("doctor@doctor.com ")	setDoctor_comm(patient.getDoctor_comm ());

Unit Test Case 006 (UTC-006):

Method name: createAuthenticationToken() throws AuthenticationException

Description: This code is a method for creating an authentication token for a user in the system. It takes in a username and password as parameters and generates a token using JWT (JSON Web Token) for user authentication.

Data for testing:

ID	username	password	Valid or Invalid
1	testuser	testpassword	Valid
2	invaliduser	testpassword	Invalid
3	testuser	invalidpassword	Invalid

Test Case：

ID	Description	Input	Expected Result
1.	Should generate an authentication token for a valid user	JwtAuthenticationRequest(username="testuser", password="testpassword"), Device	The method should return a ResponseEntity with a token and user information
2.	Should throw AuthenticationException for an invalid user	JwtAuthenticationRequest(username="invaliduser", password="testpassword"), Device	The method should throw an AuthenticationException
3	Should throw AuthenticationException for an invalid password	JwtAuthenticationRequest(username="testuser", password="invalidpassword"), Device	The method should throw an AuthenticationException
4	Should throw IllegalArgumentException with null authenticationRequest	JwtAuthenticationRequest(username=null, password=null), Device	The method should throw an IllegalArgumentException

Unit Test Case 006 (UTC-006):

Method name:getPatients()

Description: This method retrieves a list of all patients in the system if the user is an admin user, and retrieves only the data of the currently logged in user if the user is a normal user.

Data for testing:

Email	User identity
admin@admin.com	admin
enabled@user.com	user

Test Case：

ID	Description	Input	Expected Result
1.	Should return a list of all patients when called by an admin user	.name(AuthorityName.ROLE_ADMIN)	The method should return a ResponseEntity with a list of all PatientDTOs
2.	Should return only the data of the currently logged in user	.name(AuthorityName.ROLE_USER)	The method should return a ResponseEntity with the PatientDTO for the currently logged in user

Unit Test Case 007 (UTC-007):

Method name: addVaccine()

Description: This method adds vaccine data to a user's profile in the system, but only if the user making the request has admin privileges.

Data for testing:

Email	User identity
admin@admin.com	admin
enabled@user.com	user

Test Case：

ID	Description	Input	Expected Result
1.	Should add vaccine data to the user's profile when called by an admin user	.name(AuthorityName.ROLE_ADMIN)	The method should return a ResponseEntity with a success message
2.	Should throw ResponseStatusException when called by a normal user	.name(AuthorityName.ROLE_USER)	The method should throw a ResponseStatusException And prompt that “you do not have permission to operate”

Unit Test Case 007 (UTC-007):

Method name: register(username:String,password:String):String

Description: Test whether this method can be used to detect whether user account is registered.

Data for testing:

Username	Account registration status
enabled@user.com	Not registered
Enabled2@user.com	registered

Test Case：

ID	Description	Input	Expected Result
1.	The user enters an unregistered user name, enters a password for the user name, and the format of the user name meets the requirements of the software. Then click the Registration button	“Username”:” enabled@user.com”, “password”:”123456”	"message":" Account created successfully ! "
2.	The user enters an user name have already registered, enters a password for the user name. Then click the Registration button	“Username”:”Enabled2@user.com” ,“password”:”123456”	"message":" Registration failed. The Email may have been registered"

Unit Test Case 008 (UTC-008):

Method name: addDoctor(),addPat()

Description: Test whether the administrator correctly uses the permissions of the user group when assigning the registered account to different user permissions

Data for testing:

username	User identity assigned
1	doctor
2	user

Test Case：

ID	Description	Input	Expected Result
1.	When a user is assigned to the doctor user group, check whether they have all the permissions of the doctor user group	“Username”:”1”, “User identity assigned”:” doctor”	User group permissions: doctor
2.	When a user is assigned to the user group, check whether they have all the permissions of the user group	“Username”:”2” “User identity assigned”: “user”	User group permissions: user

System Test

System Testing Feature4: Users register for vaccination appointments and record vaccination times.

System test case 1 (STC_1): Test doctor and customer Sign up.
Description: This system testing is used for test Signup recording information to data storage. Signup separates two parts which are patient and doctor. Else ,Doctor and patient have different databases. The system shall provide the message alert which is “Registration successful”.

This system testing refers to URS_1: Users can click the “Signup” button and skip to the registration page for the account register. Else the register of patients and doctor have different databases.

Prerequisite or input required
The user who is a doctor or patient Signup to the system. The test data is available at Appendix A. This system testing doctor and patient registration on the web application.

Test step

1.The user clicks the sign up button and the system will skip to the registration page.
2.The user registration in doctor or patient and fill the registration list will record into the database.
3.The system will check the information format.
4.When the format and list fill click “Sign up” will receive a message alert which is “Registration successful” else will receive “Format wrong”.

Test case:

ID	Description	Input					Expect Result
		Username	Email	Password	firstname	Lastname	
STC-1.1	Registration with doctor or patient	admin					The system checks the format and records to the database.
STC_1.2	Registration with doctor or patient		admin1@gmail.com				The system checks the format and records to the database.
STC_1.3	Registration with doctor or patient			admin			The system checks the format and records to the database
STC_1.4	Registration with doctor or patient				docter1		The system checks the format and records to the database
STC_1.5	Registration with doctor or patient					DC	The system checks the format and records to the database
STC_1.6	Registration with doctor or patient						The system record information {String username = admin,String Email = admin@gmial.com,String Password = admin ,String first name = doctor, String Lastname DC}.The system send message alert “Registration successful”
STC_1.7	Registration with doctor or patient		12315				Format wrong The system send message alert “Format wrong”

System test case 2 (STC_2): Test doctor and patient Login

Description: This system testing is used for test Login that's account matched information with data storage. Login separates two parts which are patient and doctor. Else ,Doctor and patient have different databases. If the account is wrong. The system shall provide the message alert which is "Account or password wrong".

This system testing refers to URS_1: Users can click the "Login" button and skip to the Login page for the account Login. Else the login of patients and doctor have different databases.

Prerequisite or input required:
The user who is a doctor or patient Login to the system. The test data is available at Appendix A. This system testing doctor and patient Login on the web application.
Test step

1. The user selects the "Login" button for the top right of the web site to skip the login page.
2. The user fills in "username" and "password" for account login.
3. The user clicks the "Login" button at the option on the list of login.
4. The system checks if the account is correct if not send the message "Account or Password wrong".

Test case:

ID	Description	Input			Expected Result
		Username	Password		
STC_2.2	User fill username in Login list	admin			The system check the account format.String “User name = admin”and provide the account information{String user name = “admin,String password = admin}
STC_2.3	User fill password in the Login list		admin		The system checks the account format. String “Password= admin” and provide account information{String user name = “admin”,String password = admin}
STC_2.4	Users click the “Login ” button. {String user name = “admin”,String password = admin}			Select the login button.	The system matched the account {String user name = “admin”,String password = admin}.Enter the app page.
STC_2.5	User fill password in the Login list		1234		The system matched the account {String user name = “admin”,String password = admin}.Send the message “Account or password wrong”.

System test case 3 (STC-3): Tests whether the administrator can view patient and doctor information

Description: This system test will test whether the patient system can invoke data to display patient information after logging in to the system as an administrator and whether the doctor system can invoke data to display doctor information.

This system testing refers from URS-3: Users can log in and register with their own accounts to store their personal information, and doctors can register as doctors and store their personal information. The administrator can view the information of doctors and patients for easy management.

Prerequisite or input required
The user, patient and administrator who have to register and save their personal information. The test data is available at Appendix A. This system test provides patient and doctor buttons and displays their personal information.\

Test step

1.Users use their username and password to log into the account and set their name and surname and vaccination status.
2.The system match the information that the user required with the information on data storage.
3.The system retrieves username, password, name, surname and vaccination status from the database. for providing.
4.If the information does not match the data store, the user and doctor information will be incorrect or not displayed.

Test case:

ID	Description	Input					Expected Result
		Username	Password	Name	Surname	Vaccination status	
STC-2.1	Register User name	Jack					The system provides the job assignment,from{ Username=Jack , Password=password,Name=Jack, Surename=Wang, Vaccination Status=First does.}
STC-2.2	Register Password		password				The system provides the job assignment,from{Password= password,Name=Jack, Surename=Wang, Vaccination Status=First does.}
STC-2.3	Save name			Jack			The system provides the job assignment,from{,Name=Jack, Surename=Wang, Vaccination Status=First does.}
STC-2.4	Save Surname				Wang		The system provides the job assignment,from {Surename=Wang, Vaccination Status=First does.}
STC-2.5	Save Vaccination Status					First does	The system provides the job assignment,from{ Username=Jack , Password=password, Name=Jack, Surename=Wang,Vaccination Status=First does.}

System test case 4 (STC-4): Test whether the doctor can view the patient's information, whether the doctor can update the patient's vaccination status and add comments

Description: This system test will test whether the doctor can see the patient's First and Second Does Time after logging in to the system, and whether the doctor can edit the vaccination date. Can I click the update button to update. Whether you can comment after the update is complete.

This system testing refers from URS-4: Doctors can log into their accounts and view patient vaccination information, update patient information and add comments

Prerequisite or input required
Doctors can log into the system, which has recorded vaccination data and can be called, with the ability to update and add comments. The test data is available at Appendix A. This system test Display information, provide update buttons and add comments.

Test step

1： After logging in, doctors can check the patient's first vaccination time, second vaccination time and vaccination status in the information provided by the system, and can add comments.
2. The system match the information that the user required with the information on data storage.
3: The system retrieves from the data store, first vaccination time, second vaccination time, vaccination status, comment status. For providing.
4: If the information is not match from data storage. The system displays None and no comments

Test Case

ID	Description	Input	Expect result
STC-3.1	The patient click the "vaccine" button{String name = admin, String vaccine_status = First dose,String First dose name= inactivated vaccine,String Firstdose time =02/09/2022,String sconddosename = null String second dose time = null}	Click the "vaccine" button	The system provide the information of patient.{String name = admin, String vaccine_status = First dose,String First dose name= inactivated vaccine,String Firstdose time =02/09/2022,String sconddosename = null String second dose time = null}
STC-3.2	The doctor click the "vaccine" button{String name = admin, String vaccine_status = First dose,String First dose name= inactivated vaccine,String Firstdose time =02/09/2022,String sconddosename = null String second dose time = null,String name = a, String vaccine_status = A,String First dose name= a,String Firstdose time =02/09/2022,String sconddosename = b String second dose time = 03/09/2022…..}	Click the "vaccine" button.	The system provide the information of patient.{String name = admin, String vaccine_status = First dose,String First dose name= inactivated vaccine,String Firstdose time =02/09/2022,String sconddosename = null String second dose time = null,String name = a, String vaccine_status = A,String First dose name= a,String Firstdose time =02/09/2022,String sconddosename = b String second dose time = 03/09/2022…..}
STC-3.3	Doctor click the "Update" button{String name = admin, String vaccine_status = First dose,String First dose name= inactivated vaccine,String Firstdose time =02/09/2022,String	Doctor choose one patient for update vaccine information Doctor click the "Update" button{String name = admin, String vaccine_status = First dose,String First dose name= inactivated vaccine,String	The system receive update information {String name = admin, String vaccine_status = First dose,String First dose name= inactivated vaccine,String Firstdose time =02/09/2022,String

	sconddosename = null String second dose time = null,String name = a, String vaccine_status = A,String First dose name= a,String Firstdose time =02/09/2022,String sconddosename = b String second dose time = 03/09/2022…..}	Firstdose time =02/09/2022,String sconddosename = invactivated String second dose time = 05/08/2023,String name = a, String vaccine_status = A,String First dose name= a,String Firstdose time =02/09/2022,String sconddosename = b String second dose time = 03/09/2022…..}	sconddosename = invactivated String second dose time = 05/08/2023,String name = a, String vaccine_status = A,String First dose name= a,String Firstdose time =02/09/2022,String sconddosename = b String second dose time = 03/09/2022…..}
STC-3.4	The doctor who click at the option of update{String name = admin, String vaccine_status = First dose,String First dose name= inactivated vaccine,String Firstdose time =02/09/2022,String sconddosename = invactivated String second dose time = 05/08/2023,String name = a, String vaccine_status = A,String First dose name= a,String Firstdose time =02/09/2022,String sconddosename = b String second dose time = 03/09/2022…..}	The information update in database {String name = admin, String vaccine_status = First dose,String First dose name= inactivated vaccine,String Firstdose time =02/09/2022,String sconddosename = invactivated String second dose time = 05/08/2023,String name = a, String vaccine_status = A,String First dose name= a,String Firstdose time =02/09/2022,String sconddosename = b String second dose time = 03/09/2022…..}	The database update{String name = admin, String vaccine_status = First dose,String First dose name= inactivated vaccine,String Firstdose time =02/09/2022,String sconddosename = invactivated String second dose time = 05/08/2023,String name = a, String vaccine_status = A,String First dose name= a,String Firstdose time =02/09/2022,String sconddosename = b String second dose time = 03/09/2022…..}

System test case 5 (STC-5): Test whether the doctor can view the patient's information, whether the doctor can update the patient's vaccination status and add comments

Description: This system test will test whether the doctor can see the patient's First and Second Does Time after logging in to the system, and whether the doctor can edit the vaccination date. Can I click the update button to update. Whether you can comment after the update is complete.

This system testing refers from URS-5: Doctors can log into their accounts and view patient vaccination information, update patient information and add comments

Prerequisite or input required
Doctors can log into the system, which has recorded vaccination data and can be called, with the ability to update and add comments. The test data is available at Appendix A. This system test Display information, provide update buttons and add comments.

Test step

1： After logging in, doctors can check the patient's first vaccination time, second vaccination time and vaccination status in the information provided by the system, and can add comments.
2. The system match the information that the user required with the information on data storage.
3: The system retrieves from the data store, first vaccination time, second vaccination time, vaccination status, comment status. For providing.
4: If the information is not match from data storage. The system displays None and no comments

Test Case

ID	Description	Input				Expected Result
		First Does	Second Does	Update	Comment	
SCT-5.1	System record the first does date	2022.7.8				The system provides the job assignment From{First does date=2022.7.8),second does=2023.8.9,update= 2023.8.9,Comment=Already done.Patient feel good.}
SCT-5.2	System record the second does date		2023.8.9			The system provides the job assignment From {First does date=2022.7.8}
SCT-5.3	System provide a update button update does information			2023.12.5		The system provides the job assignment From {First does date=2022.7.8),second does=2023.8.9,update= 2023.8.9}
SCT-5.4	Doctor can add a comment				Already done .Patient feel good	The system provides the job assignment From{First does date=2022.7.8),second does=2023.8.9,update= 2023.8.9,Comment=Already done.Patient feel good.}

System test case 6 (STC-6): Tests whether the administrator can see patient details and doctor details after logging in.

Description: This system test will test Whether the administrator can view the doctor and patient details after logging in by clicking Patientdetail and doctordetain buttons. The system will call the required information.

This system testing refers from URS-6: The administrator account is logged in with the ability to view doctor details, Name,Surname,Age,Home Town, patient Name, Surname, and The status of vaccined.

Prerequisite or input required
Doctors have saved information about names, ages, and hometowns, and patients have been systematically logged for vaccination dates and have saved information about names and vaccinations.

Test step

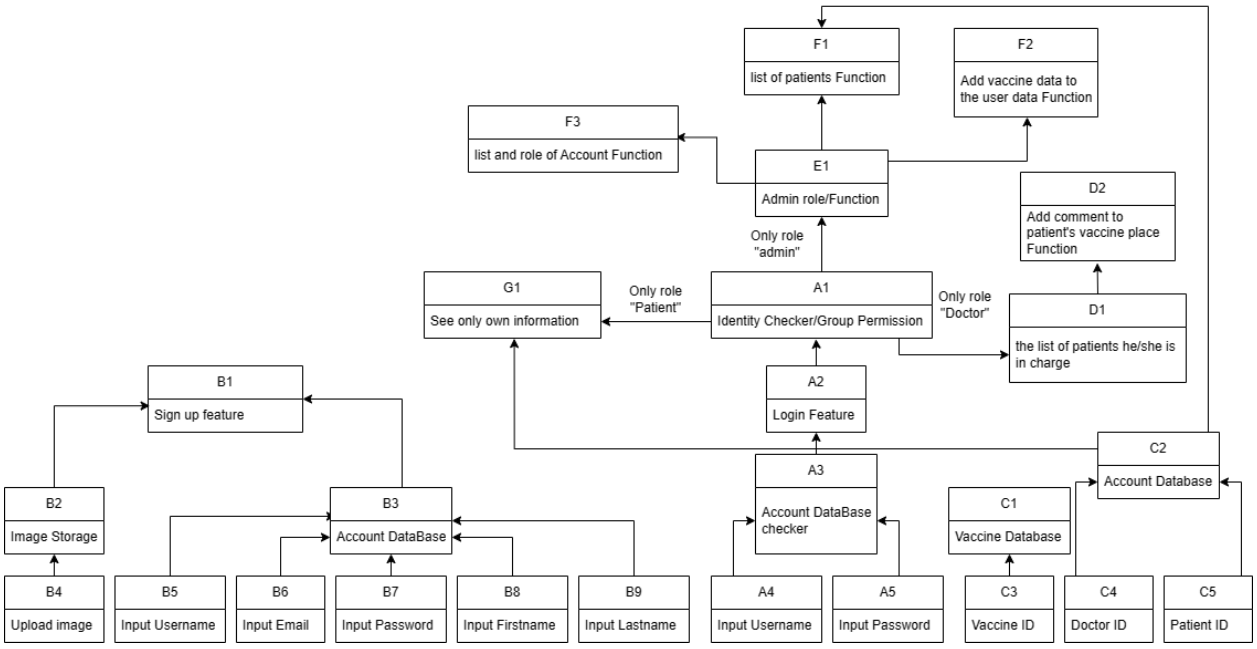
1: After the administrator logs in, the system displays the doctor's name, Surname, Age,and Home Town. Patient's vaccination status.
2. The system match the information that the user required with the information on data storage.
3: The system retrieves from the data store, Name, Surname, Home Town, Age, Patient’s Name Surname , and The status of vaccined.
4: If the information is not match from data storage. The information will not be display .

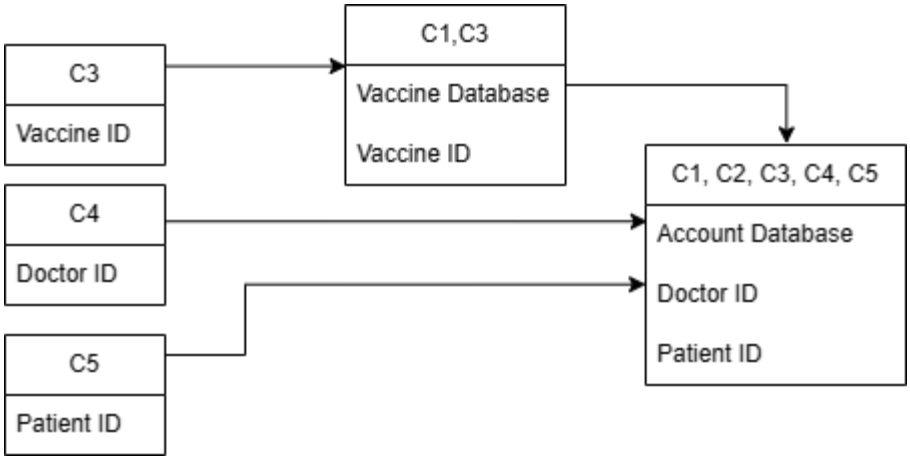
Test Case

ID: Doctor	Description	Input				Expected Result
		Name	Surname	Age	Home Town	
SCT-6.1	System record the Doctor name try to search	Mike				The system provides the job assignment From{Name=Mike,Surname=Wang\ ,Age=18,Home Town=Chaing Mai.}
SCT-6.2	System record the Doctor Surname try to search		Wang			The system provides the job assignment From { Surname=Wang\ ,Age=18,Home Town=Chaing Mai }
SCT-6.3	System record the Doctor Age try to search			18		The system provides the job assignment From { Age=18,Home Town=Chaing Mai }
SCT-6.4	System record the Doctor Home Town try to search				Chiang Mai	The system provides the job assignment From{First does date=2022.7.8),second does=2023.8.9,update= 2023.8.9,Comment=Already done.Patient feel good.}

ID: Patient	Description	Input			Expected Result
		Name	Surname	The status of vaccined	
SCT-6.5	System record the patient name try to search	Jcay			The system provides the job assignment From{Name=jacy,Surname=Wang,The status of vacciend =Second Time done. }
SCT-6.6	System record the patient Surname try to search		Wang		The system provides the job assignment From{Surname=Wang,The status of vacciend =Second Time done. }
SCT-6.7	System record the patient status of vacciend try to search			Seconde Time done.	The system provides the job assignment From{Name=jacy,Surname=Wang,The status of vacciend =Second Time done. }

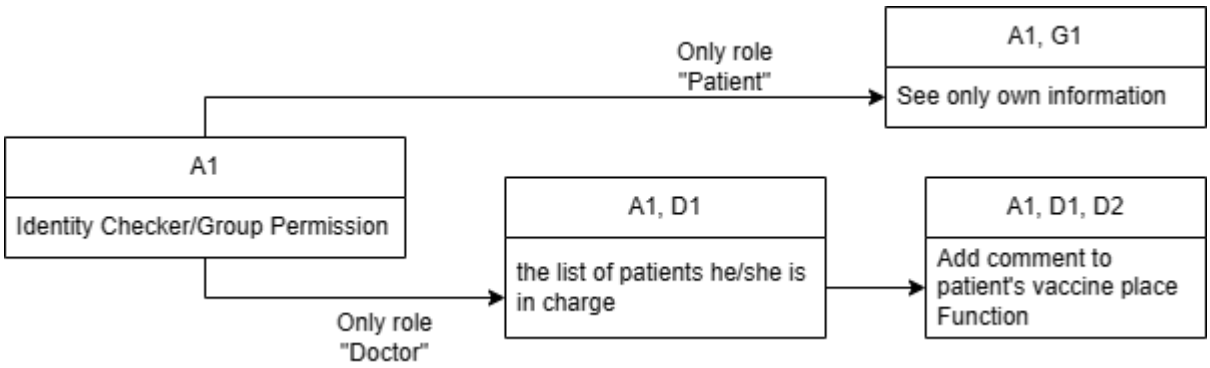
Integration Test





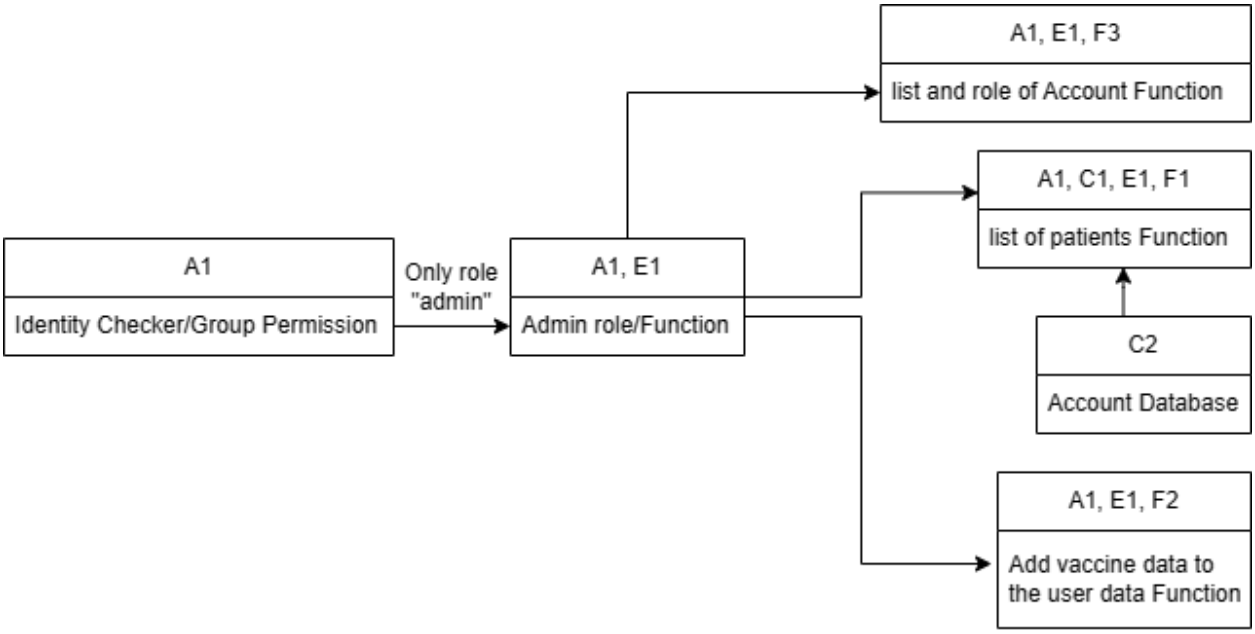
Integration Test Case 001

- From Unit Test Case 001 (UTC-001) - Unit Test Case 003 (UTC-003)
- Description :Get Vaccine ID from Vaccine Database and get DoctorID and Patient ID from Account Database. Verify that the data is stored correctly in the database.



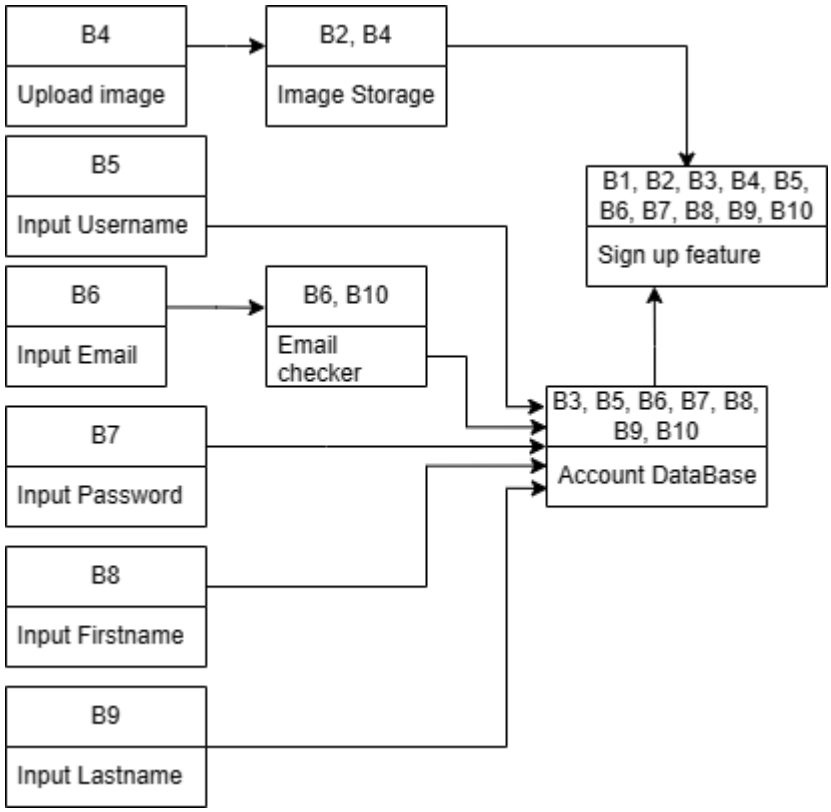
Integration Test Case 002

- From Unit Test Case 004 (UTC-004) - Unit Test Case 005 (UTC-005), Unit Test Case 009 (UTC-009)
- Description :Login with different user identities and system can give users different permissions or not. Test only doctors can leave messages. Test user registered status.



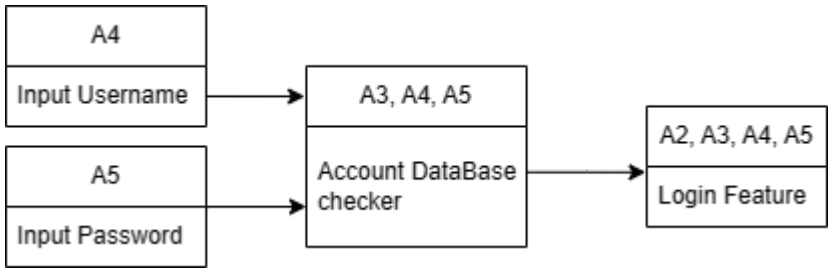
Integration Test Case 003

- From Unit Test Case 007 (UTC-007) - Unit Test Case 008 (UTC-008)
- Description : Test user is admin user or normal user, if is admin user, can add vaccine data to a user's profile in the system.



Integration Test Case 004

- From Unit Test Case 009 (UTC-009)
- Description :Test whether this user name is registered or not. If not registered and registered successfully the system shows “Account created successfully!”. If the username already exists, the system shows: “Registration failed. The Email may have been registered”.



Integration Test Case 005

- Description : login information check. Verify that the data is stored correctly in the database.

Integration			
Test case ID	Test case Objective	Test case Description	Expected Result
IT-1	System database(vaccine database and account database)	Get Vaccine ID from Vaccine Database and get DoctorID and Patient ID from Account Database. Verify that the data is stored correctly in the database.	return a ResponseEntity with the PatientDTO
IT-2	User account(admin,user,doctor)	Login with different user identities and systems can give users different permissions or not. Test only doctors can leave messages. Test user registered status.	only doctors can leave messages runs successfully
IT-3	Vaccine and patients database	Test user is admin user or normal user, if is admin user, can add vaccine data to a user's profile in the system.	Admin user add vaccine data to a user's profile
IT-4	Account database	Check whether user account is registered	Return message: “Account created successfully !” or “Registration failed. The Email may have been registered.”
IT-5	Account database	Test if login information is correct or not.	Return message:”Login success!” or “Login failed”