

Assignment 2 (E1800176 & E1800182)

by I Made Siva Aditya Surya

Submission date: 09-Apr-2021 10:28PM (UTC+0800)

Submission ID: 1554583037

File name: Assignment_2_-_E1800176_E1800182.docx (4.1M)

Word count: 5758

Character count: 30061

BIT302 SOFTWARE ENGINEERING

Assignment 2

**HELP COVID-19 Testing Information System
(CTIS) Website – Design Specification Document**



Team Leader

I Made Siva Aditya Surya / E1800176

mdsivaaditya@gmail.com

Member

Ida Bagus Gede Awidya Andika / E1800182

awidyaandika@gmail.com

2021

8
Table of Contents

1.	Introduction.....	1
1.1.	Requirements Specifications	2
1.2.	Use Case Diagram	5
2.	Design Specification	6
2.1.	Architectural Design	6
2.1.1.	Design Class Diagram.....	6
2.1.2.	ERD	7
2.2.	System Design	16
2.2.1.	System Sequence Diagram + Contract.....	16
2.2.2.	Database Design & Data Dictionary.....	20
3.	Iteration 1	24
3.1.	Identification	24
3.2.	Test Objectives	25
3.3.	Test Plan	25
3.4.	Test Results.....	26
3.4.1.	Unit Testing	26
3.4.2.	Integration Testing.....	32
3.4.3.	System Testing.....	36
3.5.	Test Analysis Report.....	47
3.6.	Updated Gantt Chart.....	48
3.7.	Collaboration via GitHub.....	49
	Bibliography	51
	Appendix A: Peer Presentation Evaluation.....	52

1. Introduction

HELP CTIS Website project was initiated with aims to develop a website information system to administers test and enable direct-to-patient report of COVID-19 test results. Some of our project objectives is to create a design specification and test plan, which will be included in this document.

The design specification document includes all the requirements of the project, architectural design of the system, and system design of the system. The architectural design defines how the relationship/structures of the system components and how they interact with each other. We listed design class diagram, entity relationship diagram, site map, and generic user experience/wireframes in this document as part of our architectural design. The system designs of the project in form of system sequence diagrams, contracts, and database design (data dictionary) are included in this document to ensure the system component would meet all the requirements which are determined in assignment 1 document.

We have done the testing of on the first iteration prototype our system and compiled the result in this document. Test plan is done to finding errors, check the quality of the system, and make sure that the end result meets the functional and non-functional requirements.

1.1. Requirements Specifications

There were no updates on any of the requirements during the first iteration of this project.
The functional and non-functional requirements stated on the first assignment is put into table for better identification, as follows:

Functional Requirements

Functional Requirement No.	Description	Associated User
FR1	A menu to login to the system with username and password verification.	All
FR2	View testing history menu that shows the patient's Covid-19 test data and result in chronological order.	Patient
FR3	Test report menu that shows a list of all test that have been administered in the test centre that the officer is employed at are shown in chronological order.	Test Centre Officer
FR4	Test centre officer menu that shows a list of all test centre officer (regardless of position) that is employed in the test centre.	Test Centre Officer
FR5	Test kit menu that shows a list of all test kit that the test centre has and the stock.	Test Centre Officer
FR6	A form in patient test menu to record new test data with appropriate label and column to fill out patient data, patient type, and symptoms and a button to record the data.	Tester
FR7	Update test result function in patient test menu that shows a list of test data in chronological order with a button to edit test result, the button will navigate to update test result page with appropriate form to fill out the result data and a button to update the test result.	Tester

FR8	Register test centre page when the manager first login to system, to verify and record the manager's test centre into database.	Test Centre Manager
FR9	Record officer function in test centre officer menu that includes a form with appropriate label and column to fill out new officer data, a username verification, and a button to record the data.	Test Centre Manager
FR10	Record tester function in test centre officer menu that includes a button to update the officer position to tester if the manager wants to change the position of an existing officer to tester, or a form to fill out officer data if the manager wants to record new officer with the position of tester.	Test Centre Manager
FR11	Manage test kit menu that includes a list of all test kit available with its stock, a button to input new test kit, and a button to update test kit details (stock).	Test Centre Manager
FR12	A button to log out from the system.	All

Non-functional Requirements

Non-functional Requirement No.	Name	Description
NF1	Correctness	All the information that is recorded to the system and provided by the system must be accurate, complete, and up-to-date. To support this requirement, the system must have a data verification every time a data is about to be recorded to the database.
NF2	Efficiency	The system should be lightweight and does not require extensive amount of hardware resources to perform a function.
NF3	Privacy of Information	Patient and officer personal data that is stored in the database must be properly consent and regulated.
NF4	Security & Integrity	The system must control unauthorized person from accessing and changing the data. The only one who can record and update test centre officer data and position is the manager. The

		only who can record and update patient's test data and test result is the tester.
NF5	Usability	The system should be easy to learn, understand, and operate all the features. To support this requirement, a good design for the website with appropriate icon for buttons and menu should be implemented.
NF6	Reliability	The system should perform all its functions as intended without failure.

1.2. Use Case Diagram

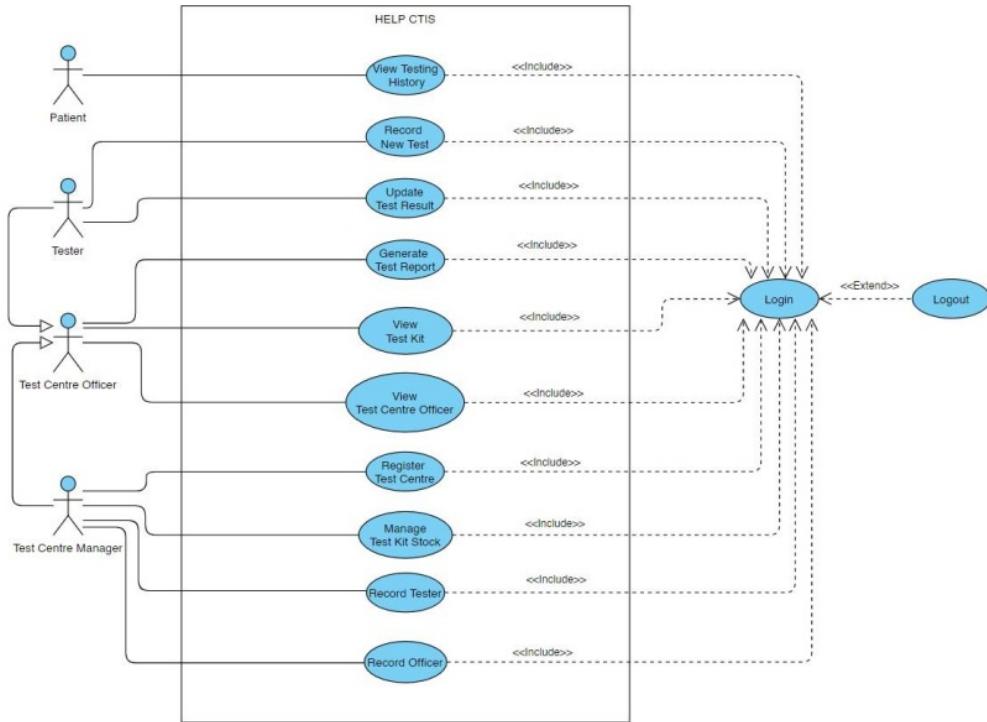


Figure 1. Use Case Diagram

There were no updates on the use case diagram during the first iteration of this project. The project will still go with the 12 use cases that were already stated on the first assignment. Three out of 12 use cases, namely: login, logout, and register test centre were developed and tested in the first iteration of the first project. The remaining use cases are to be developed in the second iteration of the project.

2. Design Specification

2.1. Architectural Design

2.1.1. Design Class Diagram

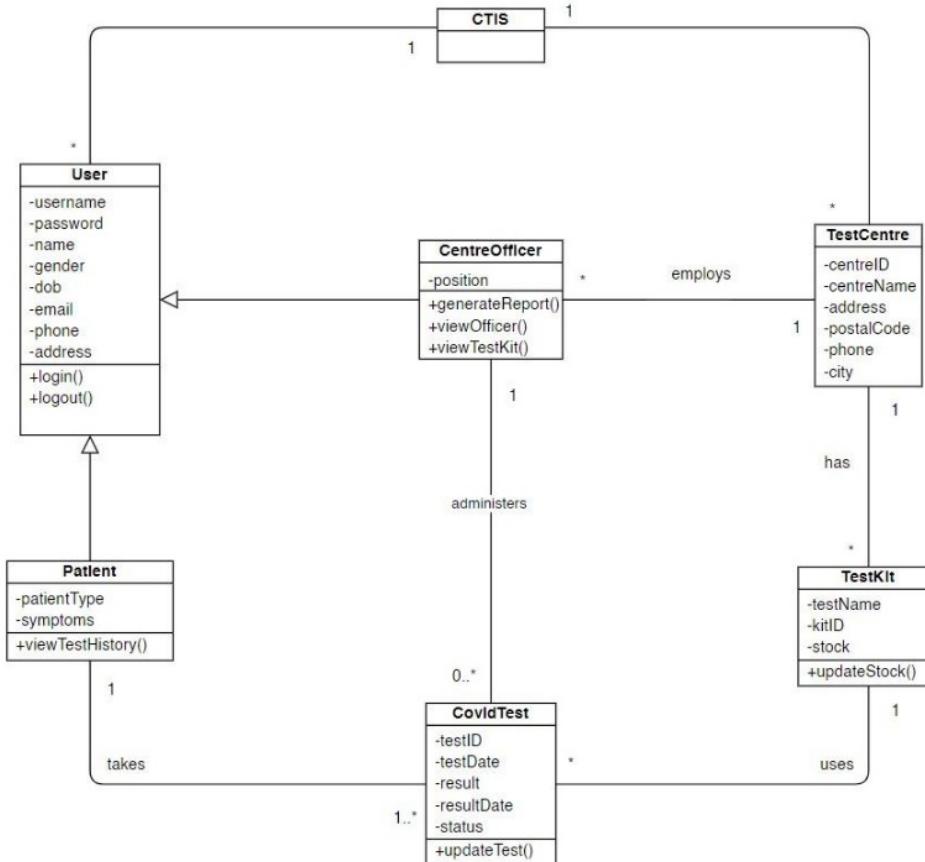


Figure 2. Design Class Diagram

There were no updates on the design class diagram during the first iteration of this project. Three out of 12 use cases that were developed in the first iteration of the first project requires no changes to the class diagram. The remaining use cases that are to be developed in the second iteration of the project may change the design and details of the class diagram, further details of the changes to the diagram will be described in the next iteration documents.

2.1.2. ERD

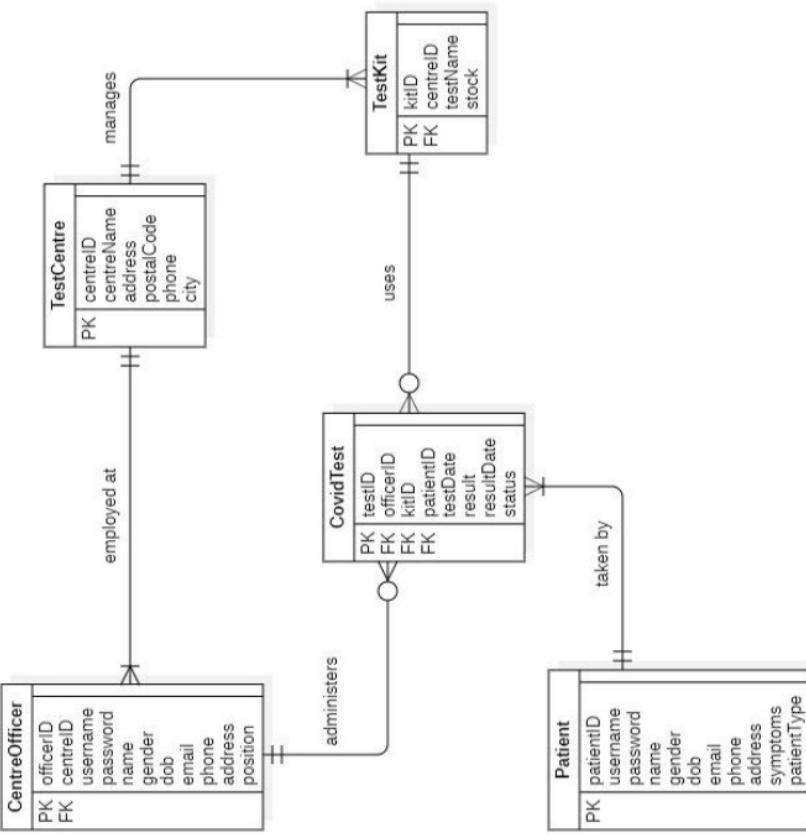


Figure 3. Entity Relationship Diagram

2.1.3. Site Map

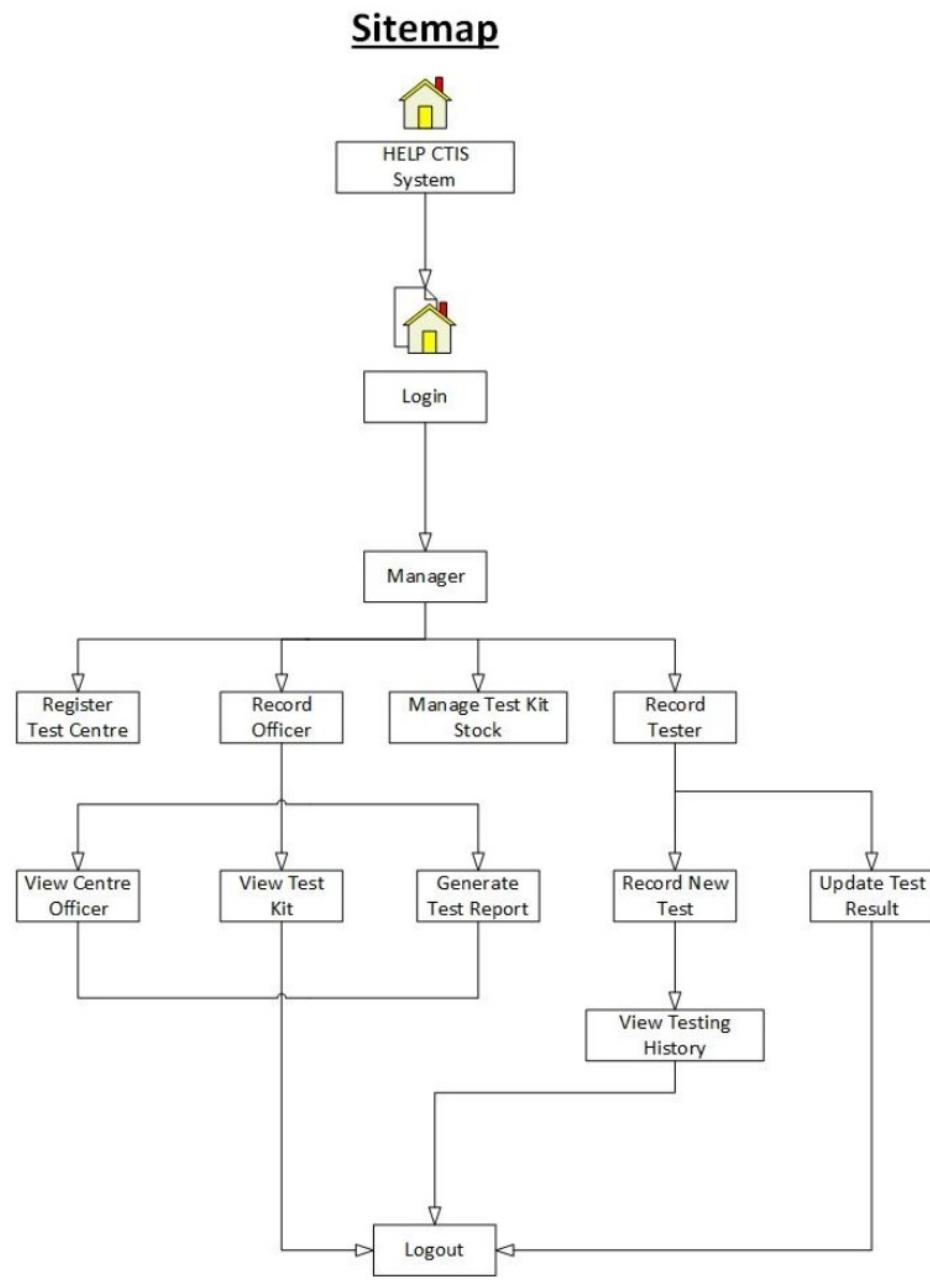


Figure 4. HELP CTIS Sitemap

2.1.4. Generic User Experience/Wireframe

1. Login

The wireframe shows a simple login interface. At the top center is the text "HELP CTIS". Below it are two input fields: the top one is labeled "Username or Email" and the bottom one is labeled "Password". At the bottom center is a dark rectangular button with the text "SIGN IN" in white.

Figure 5. Login Wireframe

2. Dashboard (All User)

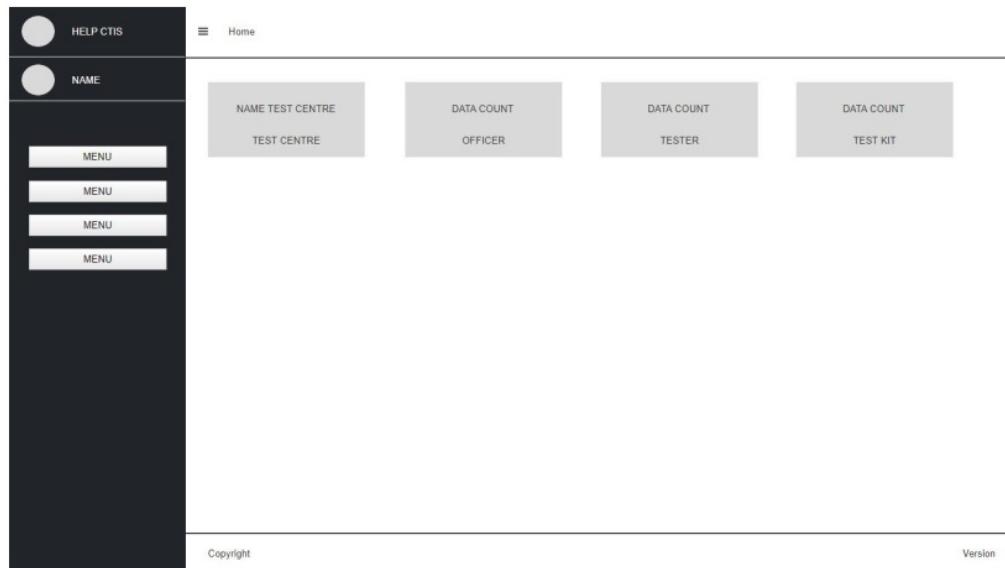
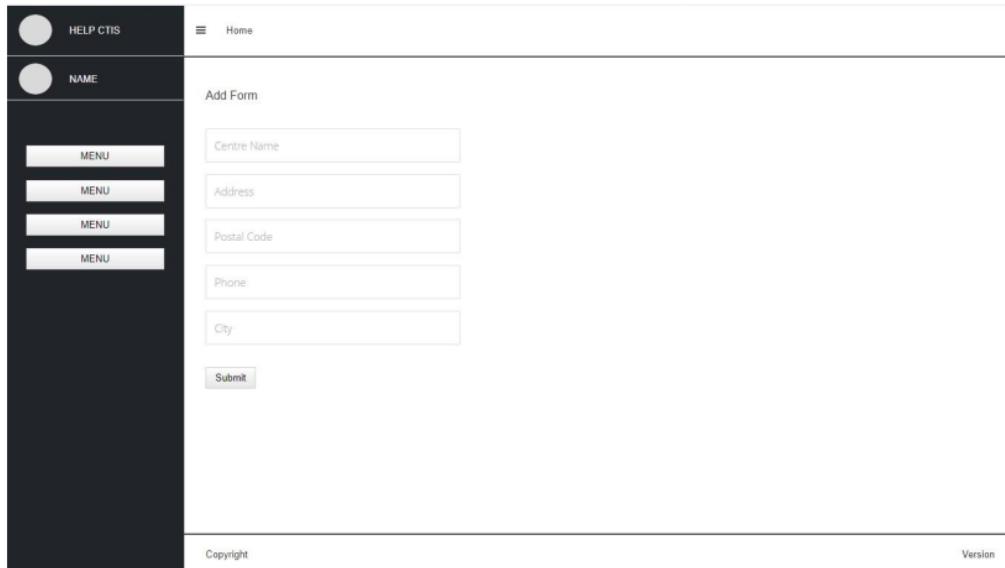


Figure 6. Dashboard Wireframe

3. Register Test Centre



The wireframe for the 'Register Test Centre' page features a dark sidebar on the left containing icons for 'HELP CTIS' and 'NAME', and four 'MENU' items. The main content area has a header with a 'Home' link and a 'Add Form' section. The 'Add Form' section includes fields for 'Centre Name', 'Address', 'Postal Code', 'Phone', and 'City', followed by a 'Submit' button. At the bottom of the page are 'Copyright' and 'Version' links.

Figure 7. Register Test Centre Wireframe

4. Record Officer



The wireframe for the 'Record Officer' page features a dark sidebar on the left containing icons for 'HELP CTIS' and 'NAME', and four 'MENU' items. The main content area has a header with a 'Home' link and an 'Add Form' section. The 'Add Form' section includes fields for 'Username', 'Password', 'Name', 'Gender', 'Date of Birth', 'Email', 'Phone', and 'Address', followed by a 'Submit' button. At the bottom of the page are 'Copyright' and 'Version' links.

Figure 8. Record Officer Wireframe

5. Record Tester



The wireframe for the Record Tester page features a dark sidebar on the left containing icons for 'HELP CTIS' and 'NAME', and four 'MENU' buttons. The main content area has a header with a 'Home' button and a 'Add Form' section. This section includes three input fields labeled 'Username', 'Name', and 'Position', followed by a 'Submit' button. At the bottom of the page are 'Copyright' and 'Version' links.

Figure 9. Record Tester Wireframe

6. View Test Kit



The wireframe for the View Test Kit page follows a similar structure to Figure 9. It includes a dark sidebar with 'HELP CTIS' and 'NAME' icons, and four 'MENU' buttons. The main area has a 'Data Table Test Kit' section with a table showing three rows of sample data. Each row contains three text fields ('Title') and one row of buttons ('Edit' and 'Delete'). A 'Add Test Kit' button is located at the top right of the table area. The bottom of the page includes 'Copyright' and 'Version' links.

Title	Title	Title	Action
sample text	sample text	sample text	Edit Delete
sample text	sample text	sample text	Edit Delete

Figure 10. View Test Kit Wireframe

7. Manage Test Kit

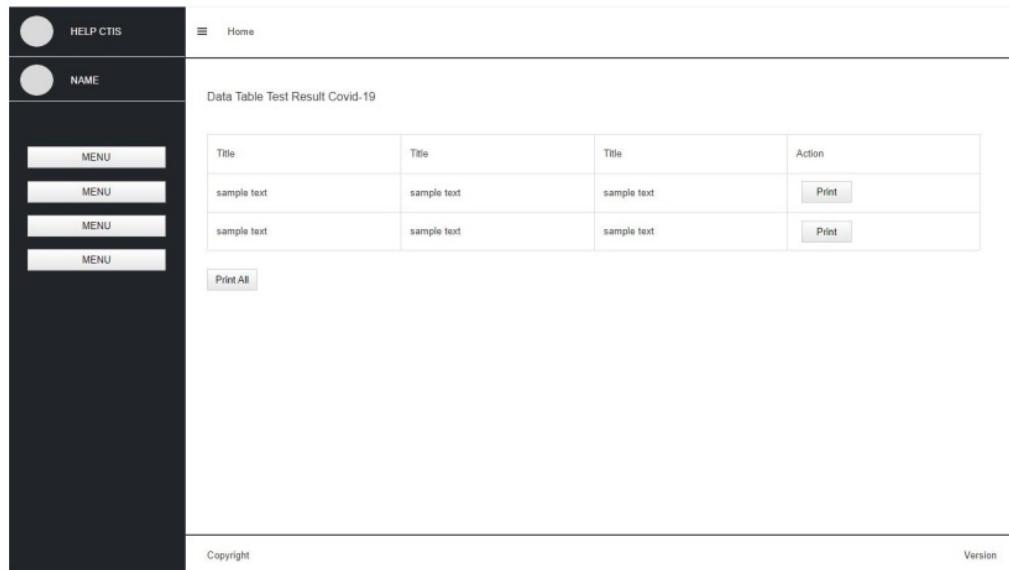
The wireframe for the Manage Test Kit (Add) screen features a dark sidebar on the left containing icons for 'HELP CTIS' and 'NAME', and four 'MENU' items. The main area has a header with a menu icon and 'Home'. Below it is a form titled 'Add Form' with fields for 'Test Name' and 'Stock', and a 'Submit' button. At the bottom are 'Copyright' and 'Version' links.

Figure 11. Manage Test Kit (Add) Wireframe

The wireframe for the Manage Test Kit (Edit) screen is similar to Figure 11, but the form title is 'Edit Form'.

Figure 12. Manage Test Kit (Edit) Wireframe

8. Generate Test Report



The wireframe for the 'Generate Test Report' page features a dark sidebar on the left containing icons for 'HELP CTIS' and 'NAME', and four 'MENU' items. The main content area has a header with a 'Home' button and a title 'Data Table Test Result Covid-19'. Below the title is a table with four rows and four columns. The first three rows have sample text in all columns, while the fourth row has 'sample text' in the first two columns and 'sample text' in the last two columns. Each row contains a 'Print' button in the last column. A 'Print All' button is located below the table. The footer includes 'Copyright' and 'Version'.

Title	Title	Title	Action
sample text	sample text	sample text	Print
sample text	sample text	sample text	Print
sample text	sample text	sample text	Print

Figure 13. Generate Test Report Wireframe

9. View Test Centre Officer



The wireframe for the 'View Test Centre Officer' page follows a similar structure to Figure 13. It includes a dark sidebar with 'HELP CTIS' and 'NAME' icons, and four 'MENU' items. The main area has a 'Home' button in the header and a title 'Data Table Officer'. On the right side of the title is a 'Add Officer' button. Below the title is a table with four rows and four columns. The first three rows have sample text in all columns, while the fourth row has 'sample text' in the first two columns and 'sample text' in the last two columns. Each row contains 'Edit' and 'Delete' buttons in the last column. A 'Print All' button is located below the table. The footer includes 'Copyright' and 'Version'.

Title	Title	Title	Action
sample text	sample text	sample text	Edit Delete
sample text	sample text	sample text	Edit Delete
sample text	sample text	sample text	Edit Delete

Figure 14. View Test Centre Officer Wireframe

10. Record New Test

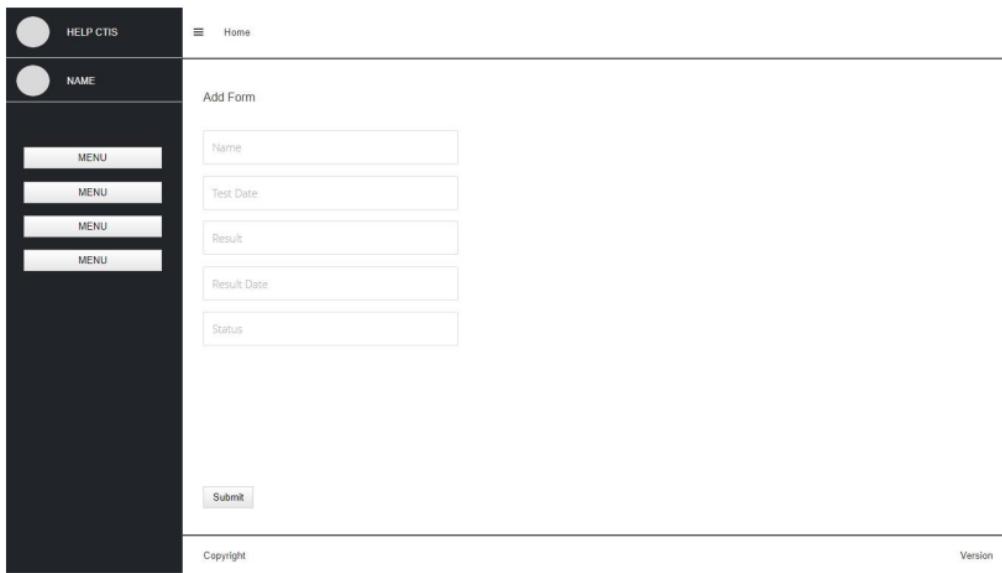


The wireframe for 'Record New Test' features a dark sidebar on the left containing icons for 'HELP CTIS' and 'NAME', and four 'MENU' buttons. The main area has a header with a menu icon and 'Home'. Below is a form titled 'Add Form' with fields for Name, Gender, Email, Phone, Date of Birth, Symptoms, Patient Type, and Status, followed by a 'Submit' button. The footer includes 'Copyright' and 'Version'.

Name	Gender	Email	Phone	Date of Birth	Symptoms	Patient Type	Status
<input type="text"/>							

Figure 15. Record New Test Wireframe

11. Update Test Result



The wireframe for 'Update Test Result' is similar to Figure 15, with a dark sidebar, a header with a menu icon and 'Home', and a form titled 'Add Form' with fields for Name, Test Date, Result, Result Date, and Status, plus a 'Submit' button. The footer includes 'Copyright' and 'Version'.

Name	Test Date	Result	Result Date	Status
<input type="text"/>				

Figure 16. Update Test Result Wireframe

12. View Testing History

The wireframe for the 'View Testing History' page features a dark sidebar on the left containing icons for 'HELP CTIS' and 'NAME', and four 'MENU' buttons. At the top right, there's a 'Home' button and a horizontal line. Below the title 'Data Table Test History Covid-19', there's a data table with four columns: 'Title', 'Title', 'Title', and 'Action'. The first row contains 'sample text' in all three 'Title' columns and a 'Print' button in the 'Action' column. The second row also contains 'sample text' in all three 'Title' columns and a 'Print' button in the 'Action' column. At the bottom, there are 'Copyright' and 'Version' links.

Title	Title	Title	Action
sample text	sample text	sample text	<input type="button" value="Print"/>
sample text	sample text	sample text	<input type="button" value="Print"/>

Figure 17. View Testing History Wireframe

2.2. System Design

2.2.1. System Sequence Diagram + Contract

1. Login

Prepared by: Ida Bagus Gede Awidya Andika

- System Sequence Diagram

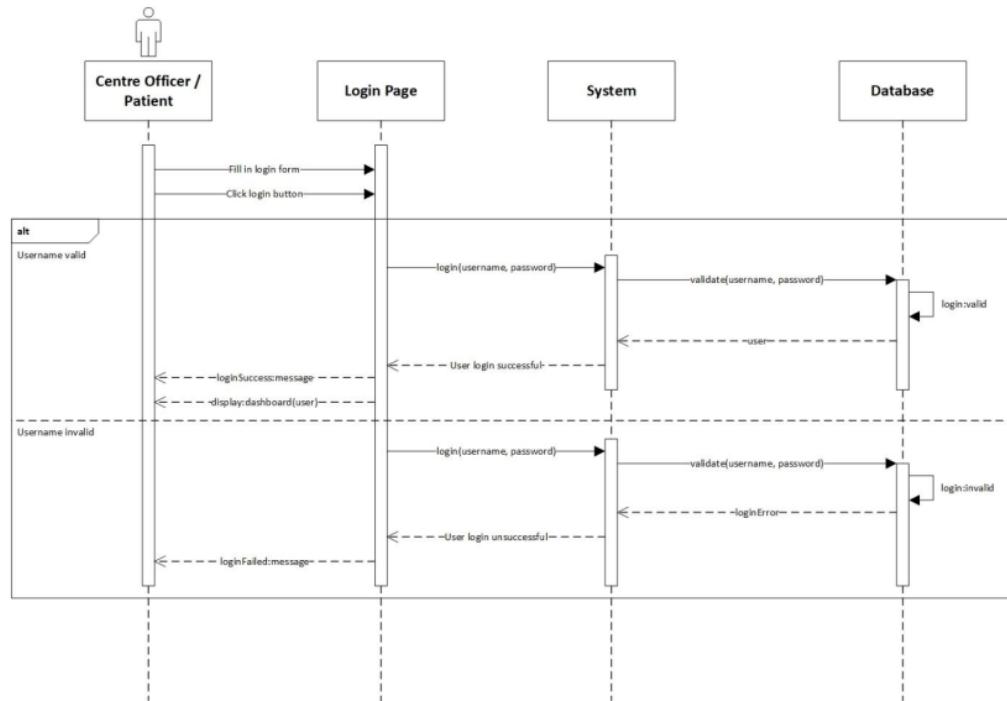


Figure 18. Login System Sequence Diagram

- Contract

Cross References	Login
Operation	<code>login(username, password)</code>
Responsible	To access the user's dashboard.
Pre-conditions	<ul style="list-style-type: none"> • User object must be available • A valid username and password

Post-conditions	<ul style="list-style-type: none"> Session was created (instance creation) Session was associated with the User (association formed)
Cross References	Login
Operation	login(username, password)
Responsible	To access the user's dashboard.

2. Logout

Prepared by: Ida Bagus Gede Awidya Andika

- System Sequence Diagram

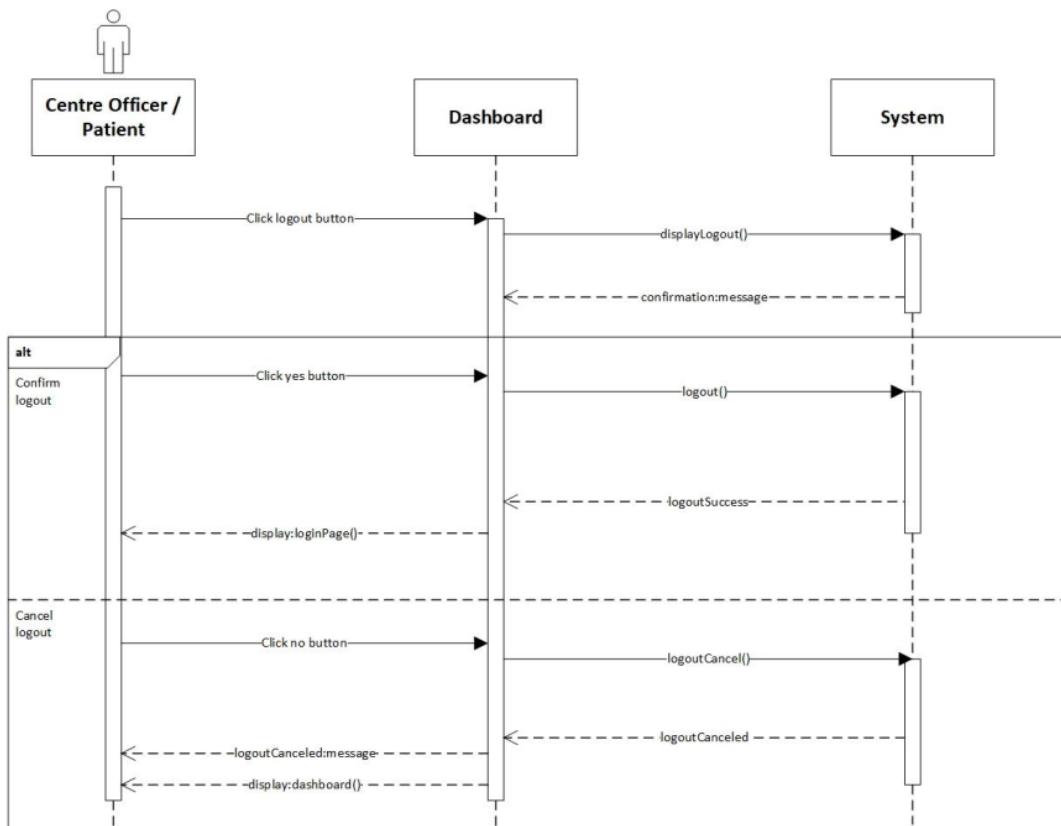


Figure 19. Logout System Sequence Diagram

- Contract

Cross References	Logout
Operation	logout()
Responsible	To log out or exit from the system
Pre-conditions	<ul style="list-style-type: none"> • User object must be available • Session must already be created • Confirm to logout
Post-conditions	Session was destroyed (instance deletion)

Cross References	Logout
Operation	logoutCancel()
Responsible	To cancel log out or exit from the system
Pre-conditions	<ul style="list-style-type: none"> • User object must be available • Session must already be created • Confirm to cancel logout
Post-conditions	Session was not destroyed

3. Register Test Centre

Prepared by: Ida Bagus Gede Awidya Andika

- System Sequence Diagram

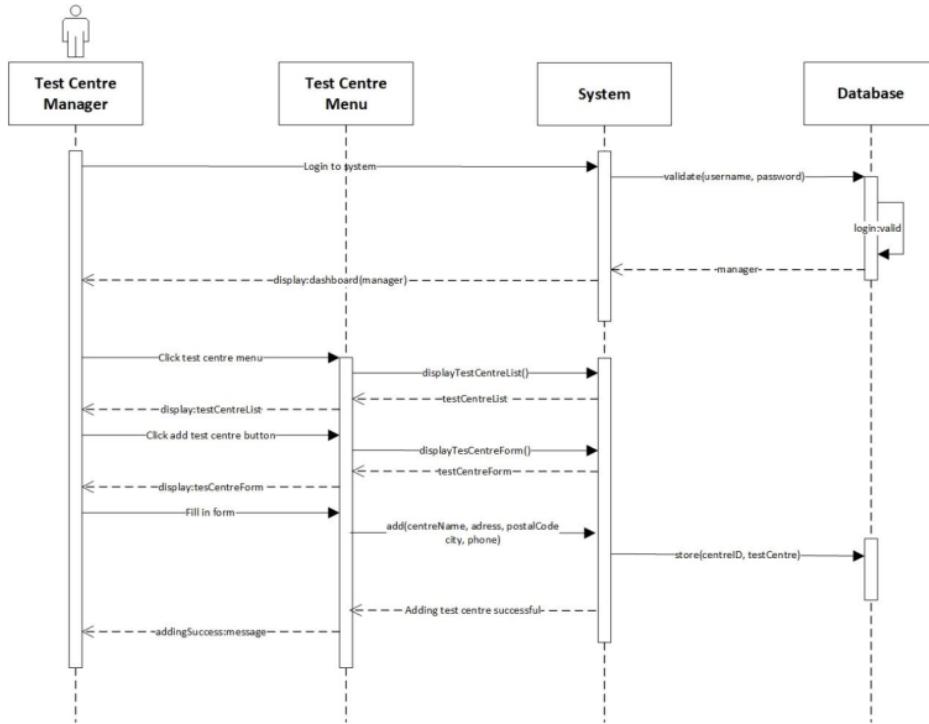


Figure 20. Register Test Centre System Sequence Diagram

- Contract

Cross References	Register Test Centre
Operation	login(username, password)
Responsible	To access the manager's dashboard.
Pre-conditions	<ul style="list-style-type: none"> Manager object must be available A valid username and password
Post-conditions	<ul style="list-style-type: none"> Session was created (instance creation) Session was associated with the Manager (association formed)
Cross References	Register Test Centre
Operation	add(centreName, address, postalCode, city, phone)

Responsible	To register test centre.
Pre-conditions	<ul style="list-style-type: none"> Manager object must be available Session must already be created Test centre data is available
Post-conditions	<ul style="list-style-type: none"> TestCentre was created (instance creation) centreId was generated (attribute modification) Manager was associated with the TestCentre (association formed)

2.2.2. Database Design & Data Dictionary

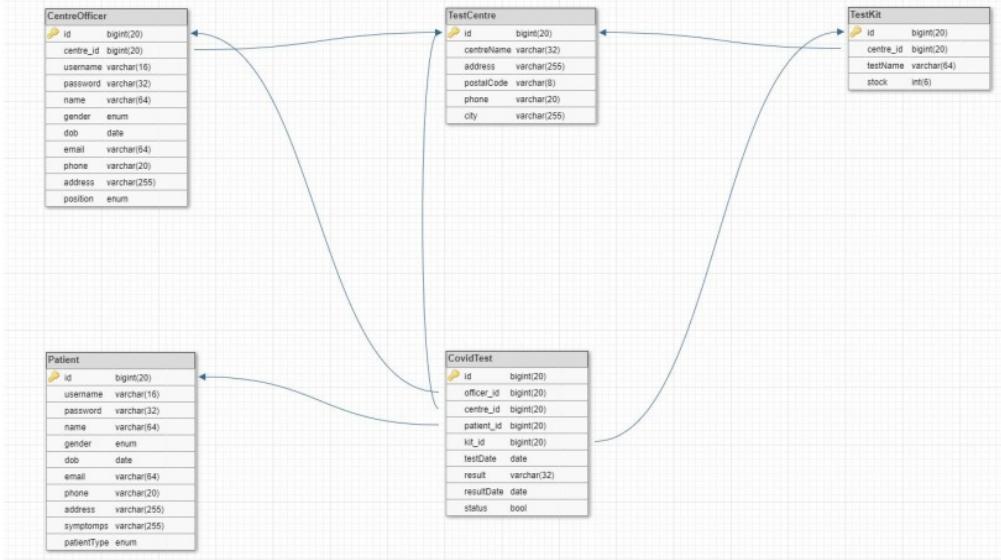


Figure 21. Database Design

a. CentreOfficer Table

Field Name	Data Type	Field Size	Description	Example
id	bigint	20	A test centre officer's unique ID number	1

centre_id	bigint	20	ID number of the test centre that a test centre officer is employed at	1
username	varchar	16	Test centre officer unique account username for login	manager
password	password	32	Test centre officer account password for login	mangacan
name	varchar	64	Name of test centre officer	Manager
gender	enum		Gender of test centre officer (Male or Female)	Male
dob	date		Birth date of test centre officer	1998-01-03
email	varchar	64	Email of test centre officer	manager@gmail.com
phone	varchar	20	Phone number of test centre officer	085734903945
address	varchar	255	Home address of test centre officer	Jl. Perum Permai, No: 22
position	enum		Employee position of test centre officer	manager

3
b. TestCentre Table

Field Name	Data Type	Field Size	Description	Example
id	bigint	20	A test centre's unique ID number	1
centreName	varchar	32	Name of test centre	Bali Med
address	varchar	255	Address of test centre	Jl. Mahendra 12x
postalCode	varchar	8	Postal code of the address	80119
phone	varchar	20	Hotline call centre number of test centre	(0361) 423215
city	varchar	50	City/region where the test centre is located	Denpasar

³
c. TestKit Table

Field Name	Data Type	Field Size	Description	Example
id	bigint	20	A test kit's unique ID number	1
centre_id	bigint	20	ID number of the test centre that owns the test kit	1
testName	varchar	64	Name of test kit	COVID-19 Antigen Rapid Testing (RTK-Ag)
stock	int	6	Available stock of test kit	100

³
d. Patient Table

Field Name	Data Type	Field Size	Description	Example
id	bigint	20	A patient's unique ID number	1
username	varchar	16	Patient unique account username for login	roger
password	varchar	32	Patient account password for login	captainroger
name	varchar	64	Name of patient	Roger Fritz
gender	enum		Gender of patient (Male or Female)	Male
dob	date		Birth date of patient	1991-04-03
email	varchar	64	Email of patient	rogercpt@gmail.com
phone	varchar	20	Phone number of patient	08573492304
address	varchar	255	Home address of patient	Jl. Beringin no: 17
symptoms	varchar	255	COVID-19 symptoms of patient	Fever, sore throat, loss of taste and smell, dry cough, and difficulty of breathing.

patientType	enum		Type of patient (“returnee”, “quarantined”, “close contact”, “infected”, or “suspected”)	Returnee
-------------	------	--	--	----------

3
e. CovidTest Table

Field Name	Data Type	Field Size	Description	Example
id	bigint	20	A test centre's unique ID number	1
officer_id	bigint	20	ID number of the test centre officer who administered the test	1
patient_id	bigint	20	ID number of the patient who took the test	1
kit_id	bigint	20	ID number of test kit which is used to perform the test	1
testDate	date		The date on which the test is performed	2021-04-03
result	varchar	100	Result of the test	HEALTHY and NEGATIVE rapid test results with the COVID-19 Rapid Test
resultDate	date		The date on which the test results are issued	2021-04-05
status	enum		Status of the test (“pending” or “completed”)	Completed

3. Iteration 1

3.1. Identification

For the first iteration of this project, we developed and tested three out of 12 use cases namely login, logout, and register test center which all are developed by Ida Bagus Gede Awidya Andika. All the remaining use cases will be developed and tested during the second iteration of our project

No	Use Case	Developed by	Development & Testing
1	Login	Ida Bagus Gede Awidya Andika	Iteration 1
2	Logout	Ida Bagus Gede Awidya Andika	Iteration 1
3	Register Test Centre	Ida Bagus Gede Awidya Andika	Iteration 1
4	Manage Test Kit Stock	Ida Bagus Gede Awidya Andika	Iteration 2
5	Record Officer	I Made Siva Aditya Surya	Iteration 2
6	Record Tester	I Made Siva Aditya Surya	Iteration 2
7	View Test Kit	Ida Bagus Gede Awidya Andika	Iteration 2
8	View Test Centre Officer	Ida Bagus Gede Awidya Andika	Iteration 2
9	Generate Test Report	I Made Siva Aditya Surya	Iteration 2
10	Record New Test	I Made Siva Aditya Surya	Iteration 2
11	Update Test Result	I Made Siva Aditya Surya	Iteration 2
12	View Testing History	I Made Siva Aditya Surya	Iteration 2

3.2. Test Objectives

The test objectives of the testing for the result of the first iteration of our project are to test the three use cases developed in this iteration based on the functional and non-functional requirements. Here is the list of the objectives that have to be fulfilled for the first iteration testing of this project:

- The system can be used on various computer devices such as PCs, laptops and smartphones
- The user interface runs according to its function, from buttons, links, forms, pages according to the role / position of the user
- The system can direct users to their dashboards when logged in based on their position
- All users can end their session via the logout process
- Manager can add Test Center data

3.3. Test Plan

Type of Testing	Strategy/Approach	Tools (if any)	Schedule
Unit Testing - Login - Record Test Centre - Logout	Black Box	PHP Storm	From gantt chart 4/5/21 – 4/6/21
Integration Testing	Black Box	PHP Storm	From gantt chart 4/7/21 – 4/7/21
Functional Testing	Black Box	PHP Storm	From gantt chart 4/7/21 – 4/7/21
Non-Functional Testing	Black Box	Google Chrome	From gantt chart 4/8/21 – 4/8/21

3.4. Test Results

3.4.1. Unit Testing

1. Login

The findUsername function located in the loginController functions to record email and username data entered by the user.

```
public function findUsername()
{
    $login = request()->input( key: 'login');

    $fieldType = filter_var($login, filter: FILTER_VALIDATE_EMAIL) ? 'email' : 'username';

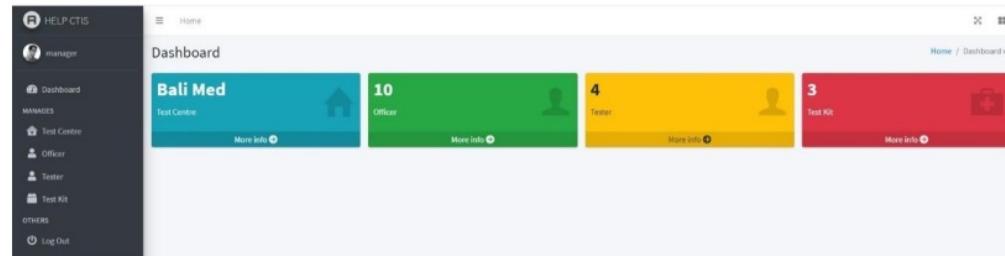
    request()->merge([$fieldType => $login]);

    return $fieldType;
}
```

Then if the data entered by the user in accordance with the database, then the user will be direct to the dashboard according to registered role / position (this function is located in HomeController).

```
public function index()
{
    if (auth()->user()->position == 'manager') {
        return redirect()->route( route: 'manager.route');
    }else if(auth()->user()->position == 'officer'){
        return redirect()->route( route: 'officer.route');
    }else{
        return redirect()->route( route: 'home');
    }
}
```

Manager Dashboard



Test ID	R-001			
Use Case	Login			
Test Description	Allow the user for login their account 4			
Source Code	<pre> if(auth()->attempt(array(\$fieldType => \$inputVal['username'], 'password' => \$inputVal['password']))){ if (auth()->user()->position == 'manager') { return redirect()->route('manager-home'); }else if(auth()->user()->position == 'officer'){ return redirect()->route('officer-home'); }else{ return redirect()->route('home'); } }else{ return redirect()->route('login') ->with('error','Email & Password are incorrect.'); } </pre>			
	Test Data	Expected Output	Actual Output	Remark(s)
Test Case 1:	Username : Password:	A warning notification will be displayed	A warning notification will be displayed	Pass
Test Case 2:	Username: manager Password: manager	Directed to the dashboard page	Directed to the dashboard page	Pass
Test Case 3:	Username: user Password: manager	A warning message that user input wrong username will be notify	A warning message that user input wrong username will be notify	Pass
Test Case 4:	Username: manager Password: user	A warning message that user input wrong password will be notify	A warning message that user input wrong password will be notify	Pass

2. Register Test Centre

The `create_test_centres_table` function as a Test Center data declaration which will be used to create tables in the database.

```

public function up()
{
    Schema::create('test_centres', function (Blueprint $table) {
        $table->id();
        $table->unsignedBigInteger('user_id')->nullable();
        $table->foreign('user_id')->references('id')->on('users')->onDelete('cascade');
        $table->string('centreName', 64);
        $table->string('address')->unique();
        $table->string('postalCode', 8);
        $table->string('phone', 20);
        $table->string('city', 50);
        $table->timestamps();
    });
}

```

In the TestCentreController, there is store function, which is useful for storing all user input data that will later be sent to the database. Users can only enter Test Centre data once.

```

public function store(Request $request)
{
    $id = auth()->user()->id;
    $data=db::table( table: 'test_centres')->where( column: 'user_id', $id)->first();

    if($data){
        return redirect()->route( route: 'testCentre.index')
            ->with('success', 'Sorry, you already created test centre.');
    }
    else{
        $request->validate([
            'user_id',
            'centreName' => 'required',
            'address' => 'required',
            'postalCode' => 'required',
            'phone' => 'required',
            'city' => 'required',
        ]);

        TestCentre::create($request->all());

        return redirect()->route( route: 'testCentre.index')
            ->with('success', 'Test Centre created successfully.');
    }
}

```

The \$fillable that is located in the TestCentre model functions to add attributes according to the column name contained in the database.

```

class TestCentre extends Model
{
    use HasFactory;

    protected $fillable = [
        'user_id',
        'centreName',
        'address',
        'postalCode',
        'phone',
        'city',
    ];
}

```

Then for the Register Test Center display located in the addtestcentre.blade.php file where the name of each attribute on the form is adjusted to the \$fillable which is located in the TestCentre model.

```
<form action="{{ route('testCentre.store') }}" method="post">
  @csrf
  <div class="card-body">
    <input type="hidden" class="form-control" id="exampleInputText" value="{{Auth::user()->id}}" name="user_id">
    <div class="form-group">
      <label for="exampleInputEmail">Test Centre Name</label>
      <input type="text" class="form-control" id="exampleInputText" placeholder="Enter Test Centre name" name="centreName">
    </div>
    <div class="form-group">
      <label for="exampleInputEmail">Address</label>
      <textarea class="form-control" id="exampleFormControlTextarea1" rows="3" name="address"></textarea>
    </div>
    <div class="form-group">
      <label for="exampleInputEmail">Postal Code</label>
      <input type="text" class="form-control" id="exampleInputText" placeholder="Enter postal code" name="postalCode">
    </div>
    <div class="form-group">
      <label for="exampleInputEmail">Phone</label>
      <input type="text" class="form-control" id="exampleInputText" placeholder="Enter phone" name="phone">
    </div>
    <div class="form-group">
      <label for="exampleInputEmail">City</label>
      <input type="text" class="form-control" id="exampleInputText" placeholder="Enter city" name="city">
    </div>
  </div>
  <!-- /.card-body -->
  <div class="card-footer">
    <button type="submit" class="btn btn-primary">Submit</button>
  </div>
</form>
```

Test Centre Dashboard

The screenshot shows a dashboard interface for managing test centers. On the left, there is a sidebar with navigation links: Home, manager, Dashboard, MANAGES, Test Centre, Officer, Tester, Test Kit, others, and Log Out. The main area is titled "Dashboard" and contains a table titled "DataTable Test Centre". The table has columns: No, Centre Name, Address, Postal Code, Phone, City, and Action. One entry is visible: "1 Bali Med Jl. Buana Kubu No.65 DPS, BR/Link, Seri Buana 80119 08585839746 Denpasar". There are buttons for "Add Test Centre" and "Edit Test Centre". At the bottom, it says "Showing 1 to 1 of 1 entries" and has "Previous" and "Next" buttons.

Test ID	R-002
Use Case	Register Test Centre
Test Description	Allow the user to register for a new account.
Source Code	<pre>\$request->validate(['centreName' => 'required', 'address' => 'required', 'postalCode' => 'required', 'phone' => 'required', 'city' => 'required',]); TestCentre::create(\$request->all());</pre>

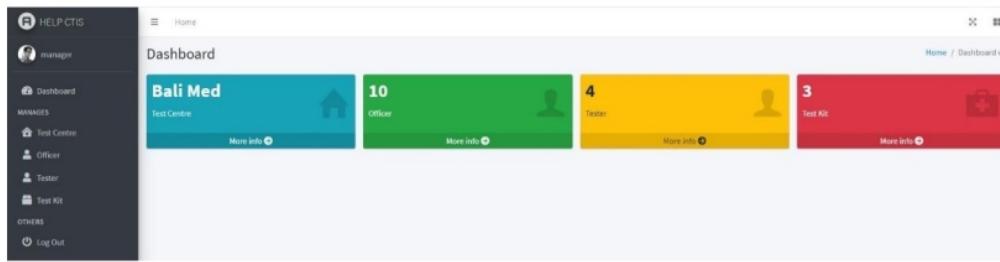
	<pre>User::where('id', \$request->user()['id'])->update(['centre_name' => \$request->centreName]); 13 return redirect()->route('testCentre.index') ->with('success', 'Test Centre created successfully.');</pre>			
	Test Data	Expected Output	Actual Output	Remark(s)
Test Case 1:	Test Centre Name: Address: Postal Code: Phone: City:	A warning notification will be displayed	A warning notification will be displayed	Pass
Test Case 2:	Test Centre Name: Bali Med Address: Jl. Buana Kubu No. 65 DPS, BR/Link. Sari Buana Postal Code: 80119 Phone: 085858839746 City: Denpasar	The data will be displayed on the Test Centre Dashboard	The data will be displayed on the Test Centre Dashboard	Pass
Test Case 3:	Test Centre Name: Bali Med Address: Jl. Buana Kubu No. 65 DPS, BR/Link. Sari Buana Postal Code: 80119 Phone: 085858839746 City: Denpasar	A warning message already created Test Centre will be notify	A warning message already created Test Centre will be notify	Pass

3. Logout

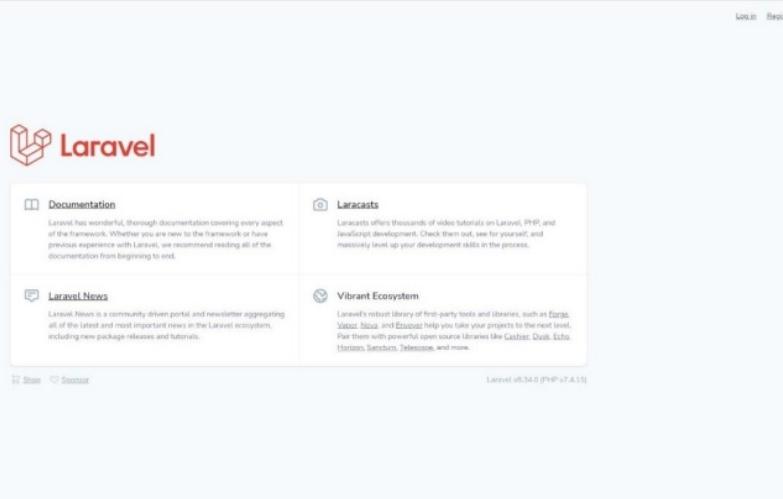
In the navbar display, there is a logout function which is used to end a session of a user who has logged in.

```
<div class="modal-footer">
    <input type="button" class="btn btn-default" data-dismiss="modal" value="Cancel">
    <a href="{{ route('logout') }}" class="btn btn-primary" onclick="event.preventDefault();
        document.getElementById('logout-form').submit();">Log Out</a>
    <form id="logout-form" action="{{ route('logout') }}" method="POST" class="d-none">
        @csrf
    </form>
</div>
```

Manager Dashboard



Back to Dashboard page



Test ID	R-003			
Use Case	Logout			
Test Description	Allow the user to logout from the application 6			
Source Code	 {{ __('Logout') }} 			
Test Data	Expected Output	Actual Output	Remark(s)	
Test Case 1:	Click log out button	Confirmation message will be displayed	Confirmation message will be displayed	Pass
Test Case 2:	Click cancel button in the confirmation message	Return to previous page	Return to previous page	Pass
Test Case 3:	Click log out button in the confirmation message	Directed to login page	Directed to login page	Pass

3.4.2. Integration Testing

We used the Selenium IDE in Chrome browser Extension. The application will run based on the functions available in each use case. The results of the tests will be exported to a Python file (.py).

1. Login

```

1 # Generated by Selenium IDE
2 import pytest
3 import time
4 import json
5 from selenium import webdriver
6 from selenium.webdriver.common.by import By
7 from selenium.webdriver.common.action_chains import ActionChains
8 from selenium.webdriver.support import expected_conditions
9 from selenium.webdriver.support.wait import WebDriverWait
10 from selenium.webdriver.common.keys import Keys
11 from selenium.webdriver.common.desired_capabilities import DesiredCapabilities
12
13 class TestLogin():
14     def setup_method(self, method):
15         self.driver = webdriver.Chrome()
16         self.vars = {}
17
18     def teardown_method(self, method):
19         self.driver.quit()
20
21     def test_login(self):
22         self.driver.get("http://127.0.0.1:8000/login")
23         self.driver.set_window_size(516, 556)
24         self.driver.find_element(By.ID, "login").send_keys("ma")
25         self.driver.find_element(By.CSS_SELECTOR, ".login-box-msg").click()
26         self.driver.find_element(By.ID, "login").click()
27         self.driver.find_element(By.ID, "login").send_keys("manager")
28         self.driver.find_element(By.ID, "password").click()
29         self.driver.find_element(By.ID, "password").send_keys("mangakan")
30         self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
31

```

Running 'Login'

1. open on /login OK
2. setWindowSize on 516x556 OK
3. type on id=login with value ma OK
4. click on css=.login-box-msg OK
5. click on id=login OK
6. type on id=login with value manager OK
7. click on id=password OK
8. type on id=password with value mangakan OK
9. click on css=.btn OK

'Login' completed successfully

Manager : Login

Test ID	R-001			
Use Case	Manager : Login			
Test Description	Allow the user for login their account			
	Test Data	Expected Output	Actual Output	Remark(s)
Test Case 1:	Username : Password:	A warning notification will be displayed	A warning notification will be displayed	Pass
Test Case 2:	Username: manager Password: manager	Directed to the dashboard page	Directed to the dashboard page	Pass
Test Case 3:	Username: user Password: manager	A warning message that user input wrong username will be notify	A warning message that user input wrong username will be notify	Pass
Test Case 4:	Username: manager Password: user	A warning message that user input wrong password will be notify	A warning message that user input wrong password will be notify	Pass

2. Register Test Centre

```

1  # Generated by Selenium IDE
2  import pytest
3  import time
4  import json
5  from selenium import webdriver
6  from selenium.webdriver.common.by import By
7  from selenium.webdriver.common.action_chains import ActionChains
8  from selenium.webdriver.support import expected_conditions
9  from selenium.webdriver.support.wait import WebDriverWait
10 from selenium.webdriver.common.keys import Keys
11 from selenium.webdriver.common.desired_capabilities import DesiredCapabilities
12
13 class TestRegisterTestCentre():
14     def setup_method(self, method):
15         self.driver = webdriver.Chrome()
16         self.vars = {}
17
18     def teardown_method(self, method):
19         self.driver.quit()
20
21     def test_registerTestCentre(self):
22         self.driver.get("http://127.0.0.1:8000/manager/home")
23         self.driver.set_window_size(1936, 1096)
24         self.driver.find_element(By.LINK_TEXT, "Test Centre").click()
25         self.driver.find_element(By.LINK_TEXT, "Add Test Centre").click()
26         self.driver.find_element(By.NAME, "centreName").click()
27         self.driver.find_element(By.NAME, "centreName").send_keys("Bali Med")
28         self.driver.find_element(By.ID, "exampleFormControlTextarea1").click()
29         self.driver.find_element(By.ID, "exampleFormControlTextarea1").send_keys("Jl. Buana Kubu No.65 DPS, BR/Link. Sari Buana")
30         self.driver.find_element(By.NAME, "postalCode").send_keys("80119")
31         self.driver.find_element(By.NAME, "phone").send_keys("085858839746")
32         self.driver.find_element(By.NAME, "city").send_keys("Denpasar")
33         self.driver.find_element(By.CSS_SELECTOR, ".btn-primary:nth-child(1)").click()
34

```

Running 'Register Test Centre'

1. open on http://127.0.0.1:8000/manager/home OK
2. setWindowSize on 1936x1096 OK
3. click on linkText=Test Centre OK
4. click on linkText=Add Test Centre OK
5. click on name=centreName OK
6. type on name=centreName with value Bali Med OK
7. click on id=exampleFormControlTextarea1 OK
8. type on id=exampleFormControlTextarea1 with value Jl. Buana Kubu No.65 DPS, BR/Link. Sari Buana OK
9. type on name=postalCode with value 80119 OK
10. type on name=phone with value 085858839746 OK
11. type on name=city with value Denpasar OK
12. click on css=.btn-primary:nth-child(1) OK

'Register Test Centre' completed successfully

Manager : Login → Register Test Centre

Test ID	R-002			
Use Case	Manager : Dashboard → Register Test Centre			
Test Description	Allow the user to register for a new account.			
	Test Data	Expected Output	Actual Output	Remark(s)
Test Case 1:	Test Centre Name: Address: Postal Code: Phone:	A warning notification will be displayed	A warning notification will be displayed	Pass

	City:			
Test Case 2:	Test Centre Name: Bali Med Address: Jl. Buana Kubu No. 65 DPS, BR/Link. Sari Buana Postal Code: 80119 Phone: 085858839746 City: Denpasar	The data will be displayed on the Test Centre Dashboard	The data will be displayed on the Test Centre Dashboard	Pass
Test Case 3:	Test Centre Name: Bali Med Address: Jl. Buana Kubu No. 65 DPS, BR/Link. Sari Buana Postal Code: 80119 Phone: 085858839746 City: Denpasar	A warning message already created Test Centre will be notify	A warning message already created Test Centre will be notify	Pass

3. Logout

```

1 # Generated by Selenium IDE
2 import pytest
3 import time
4 import json
5 from selenium import webdriver
6 from selenium.webdriver.common.by import By
7 from selenium.webdriver.common.action_chains import ActionChains
8 from selenium.webdriver.support import expected_conditions
9 from selenium.webdriver.support.wait import WebDriverWait
10 from selenium.webdriver.common.keys import Keys
11 from selenium.webdriver.common.desired_capabilities import DesiredCapabilities
12
13 class TestLogout():
14     def setup_method(self, method):
15         self.driver = webdriver.Chrome()
16         self.vars = {}
17
18     def teardown_method(self, method):
19         self.driver.quit()
20
21     def test_logout(self):
22         self.driver.get("http://127.0.0.1:8000/manager/home")
23         self.driver.set_window_size(1920, 1080)
24         self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(8) p").click()
25         self.driver.find_element(By.CSS_SELECTOR, ".modal-content > .modal-footer > .btn-primary").click()
26

```

Running 'Logout'

1. open on <http://127.0.0.1:8000/manager/home> **OK**
2. setWindowSize on 1920x1080 **OK**
3. click on `css=.nav-item:nth-child(8) p` **OK**
4. Trying to find `css=.modal-content > .modal-footer > .btn-primary...` **OK**

'Logout' completed successfully

Manager: Login → Logout

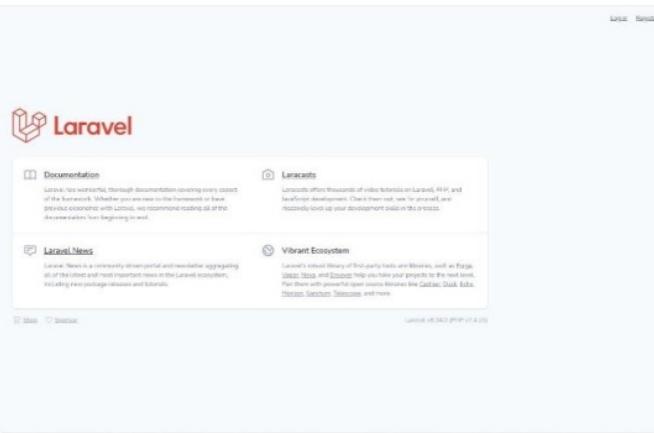
Test ID	R-003			
Use Case	Manager : Login → Logout			
Test Description	Allow the user to register for a new account.			
	Test Data	Expected Output	Actual Output	Remark(s)
Test Case 1:	Click log out button	Confirmation message will be displayed	Confirmation message will be displayed	Pass
Test Case 2:	Click cancel button in the confirmation message	Return to previous page	Return to previous page	Pass
Test Case 3:	Click log out button in the confirmation message	Directed to login page	Directed to login page	Pass

3.4.3. System Testing

Functional Requirements

1. Login

The first test starts when the user enters the URL, then the dashboard page will appear and the user selects the login page.



Login Page

The screenshot shows the HELPCTIS login interface. At the top center, the text "HELPCTIS" is displayed. Below it, a white rectangular box contains the text "Sign in to start your session". Inside this box are two input fields: "Username or Email" and "Password", each accompanied by a small icon (a person for the first and a lock for the second). Below the password field is a checkbox labeled "Remember Me" and a blue "Sign In" button. At the bottom of the box are links for "I forgot my password" and "Register a new membership".

- a. User leaves the form blank and notification will be appear.

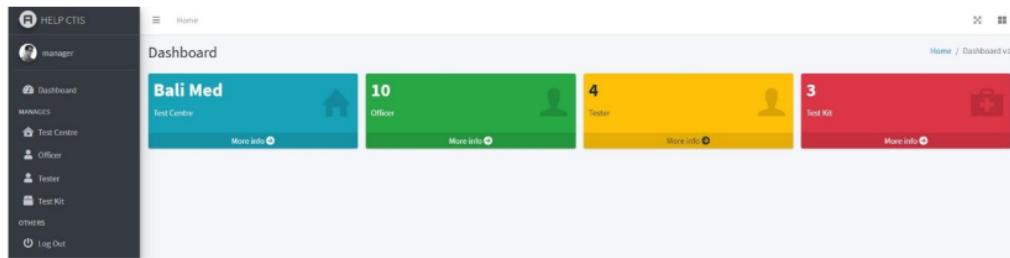
This screenshot is identical to the one above, showing the HELPCTIS login page. However, the "Password" input field now has a yellow border and a red exclamation mark icon. A red tooltip message "Please fill out this field." is overlaid on the right side of the password field. The rest of the interface remains the same, including the "Sign In" button and the "Remember Me" checkbox.

- b. User input wrong data (username, email or password)

The screenshot shows the HELPCTIS sign-in interface. At the top, it says "HELPCTIS". Below that is a box containing the text "Sign in to start your session". Inside the box, there is a user input field with "manager" typed into it, accompanied by a help icon (a question mark inside a circle) and a user icon. Below this field, a red error message reads "These credentials do not match our records.". There is also a "Password" input field with a lock icon. To the left of the "Sign In" button is a "Remember Me" checkbox. Below the input fields are links for "I forgot my password" and "Register a new membership".

- c. If user input correct data, it will immediately directed to the dashboard page according to the role.

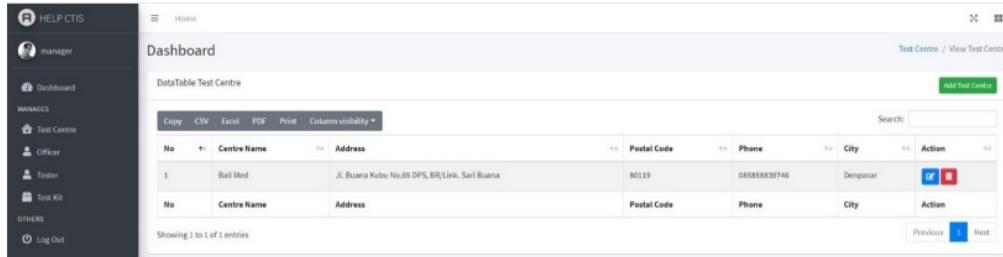
The screenshot shows the same HELPCTIS sign-in interface as the previous one, but with correct credentials entered. The "manager" username is now highlighted in blue, indicating it is selected. The "Password" field contains several dots ("....."). The rest of the interface, including the "Sign In" button and other links, remains the same as in the previous screenshot.



Test ID	R-001			
Use Case	Login			
Test Description	Testing login process by the user. To ensure the user can login to the system.			
	Test Data	Expected Output	Actual Output	Remark(s)
Test Case 1:	Login page	Show the login page	Show the login page	Pass
Test Case 2:	User fill the form with valid username and password 1. Username : "manager" 2. Password: "manager"	Successfully login and directed to the dashboard page	Successfully login and directed to the dashboard page	Pass
Test Case 3:	User fill the form with invalid username 1. Username : "user" 2. Password: "manager"	A warning message that user input wrong username will be notify	A warning message that user input wrong username will be notify	Pass
Test Case 4:	User fill the form with invalid password 1. Username : "manager" 2. Password: "user"	A warning message that user input wrong password will be notify	A warning message that user input wrong password will be notify	Pass
Test Case 5:	User do not fill the form 1. Username : 2. Password:	A warning message that the user must fill the field will be displayed	A warning message that the user must fill the field will be displayed	Pass

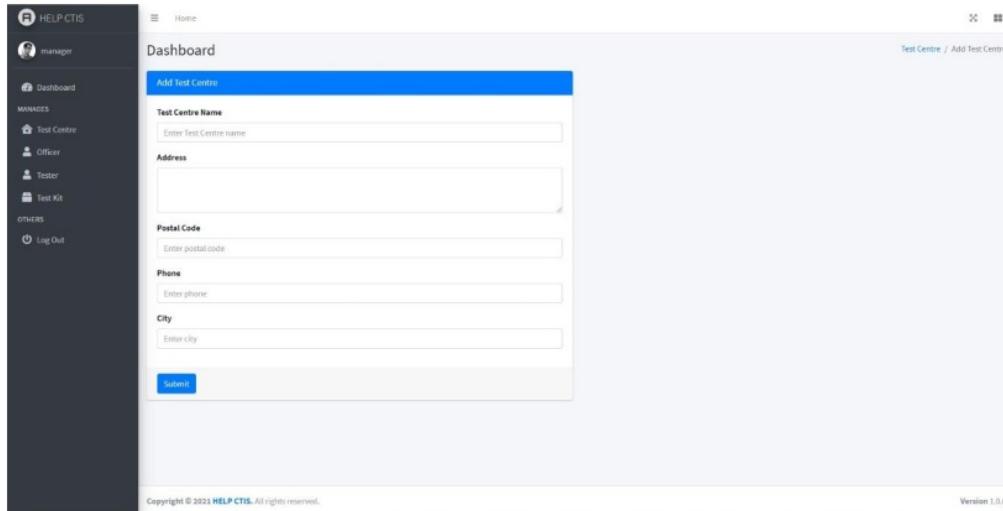
2. Register Test Centre

Register Test Center can be done when the user has selected the Test Center menu on the navbar and then press Add Test Center button on the dashboard page.



The screenshot shows the HELP CTIS dashboard with the 'Test Centre' menu item selected. A table titled 'DataTable Test Centre' displays one entry: 'Bali Med' located at 'Jl. Buana Kubu No.65 DPS, BR/Linik, Sari Buana'. The table includes columns for No, Centre Name, Address, Postal Code, Phone, City, and Action (with edit and delete icons). Below the table, a message says 'Showing 1 to 1 of 1 entries'.

Register Test Centre Page



The screenshot shows the HELP CTIS dashboard with the 'Add Test Centre' button highlighted in blue. A modal window titled 'Add Test Centre' contains fields for 'Test Centre Name' (with placeholder 'Enter Test Centre name'), 'Address' (with placeholder 'Enter address'), 'Postal Code' (with placeholder 'Enter postal code'), 'Phone' (with placeholder 'Enter phone'), and 'City' (with placeholder 'Enter city'). A 'Submit' button is at the bottom of the form.

- a. User input blank data form and notification would be appear.

The screenshot shows the 'Add Test Centre' form on the 'Dashboard'. A red validation error box at the top states: 'Whoops! There were some problems with your input.' It lists five required fields: 'Test Centre Name', 'Address', 'Postal Code', 'Phone', and 'City'. Below the error box are four input fields: 'Test Centre Name' (placeholder 'Enter Test Centre name'), 'Address' (placeholder 'Address'), 'Postal Code' (placeholder 'Enter postal code'), and 'Phone' (placeholder 'Enter phone'). A 'City' input field is partially visible below. At the bottom is a blue 'Submit' button.

- b. When the user succeeds in entering the Test Centre Data.

The screenshot shows the 'DataTable Test Centre' table on the 'Dashboard'. The table has columns: No, Centre Name, Address, Postal Code, Phone, City, and Action. One row is present: No 1, Centre Name 'Bali Med', Address 'Jl. Buana Kubu No.65 DPS, BR/Link, Sari Buana', Postal Code '80119', Phone '085658839746', City 'Denpasar', and Action buttons. The table includes standard data table controls like copy, CSV, Excel, PDF, Print, Column visibility, and search.

- c. When the user has previously created Test Center data

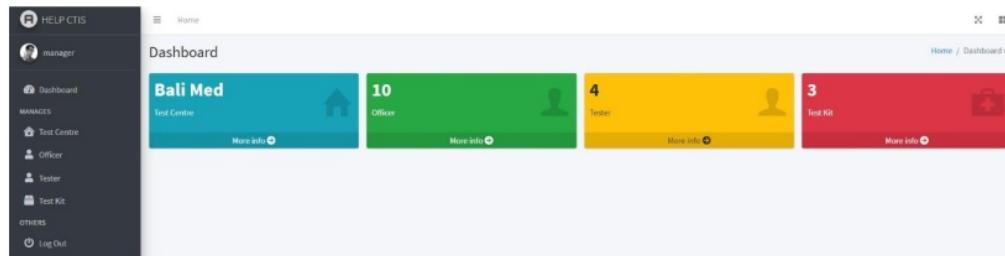
The screenshot shows the 'DataTable Test Centre' table on the 'Dashboard'. A green success message at the top says 'Sorry, you already created test centre.' The table structure is identical to the previous screenshot, showing one row of data for 'Bali Med'. The table includes standard data table controls like copy, CSV, Excel, PDF, Print, Column visibility, and search.

Test ID	R-002			
Use Case	Register Test Centre			
Test Description	Testing register process for new account. To ensure the user register to the system.			
	Test Data	Expected Output	Actual Output	Remark(s)
Test Case 1:	Register Test Centre	Click the Add Test Centre button and show the register form	Click the Add Test Centre button and show the register form	Pass
Test Case 2:	User fill the form with valid data 1. Test Centre Name: "Bali Med" 2. Address: "Jl. Buana Kubu No. 65 DPS, BR/Link. Sari Buana" 3. Postal Code: "80119" 4. Phone: "085858839746" 5. City: "Denpasar"	Successfully register will be notify and displayed on the Test Centre Dashboard	Successfully register will be notify and displayed on the Test Centre Dashboard	Pass
Test Case 3:	User fill the form with same data that already registered 1. Test Centre Name: "Bali Med" 2. Address: "Jl. Buana Kubu No. 65 DPS, BR/Link. Sari Buana" 3. Postal Code: "80119" 4. Phone: "085858839746" 5. City: "Denpasar"	A warning message already created Test Centre will be notify	A warning message already created Test Centre will be notify	Pass
Test Case 4:	User do not fill the form 1. Test Centre Name: 2. Address: 3. Postal Code:	A warning notification will be displayed that the user must fill the field	A warning notification will be displayed that the user must fill the field	Pass

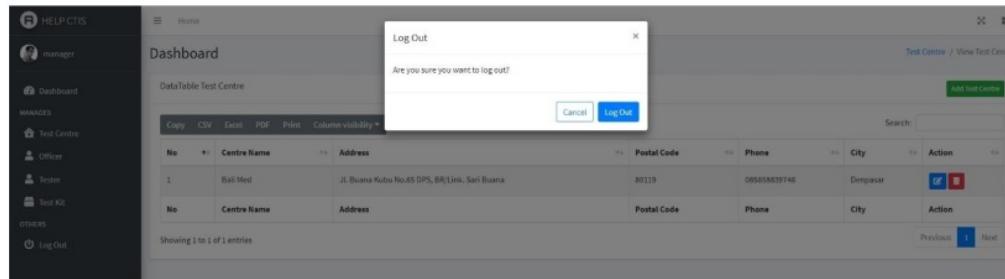
	4. Phone:		
	5. City:		

3. Logout

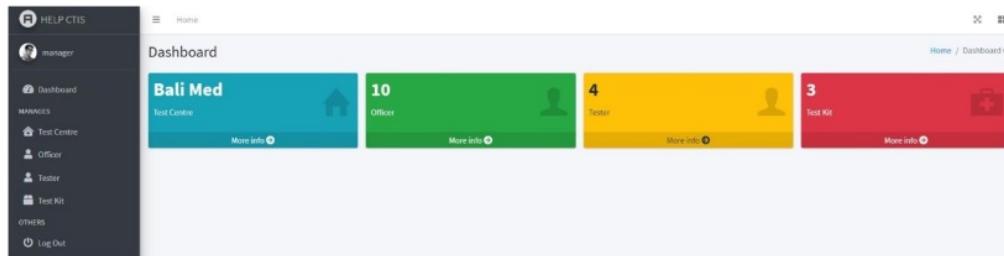
When the user wants to end the session, the user can log out of the account through the Log Out menu on the navbar



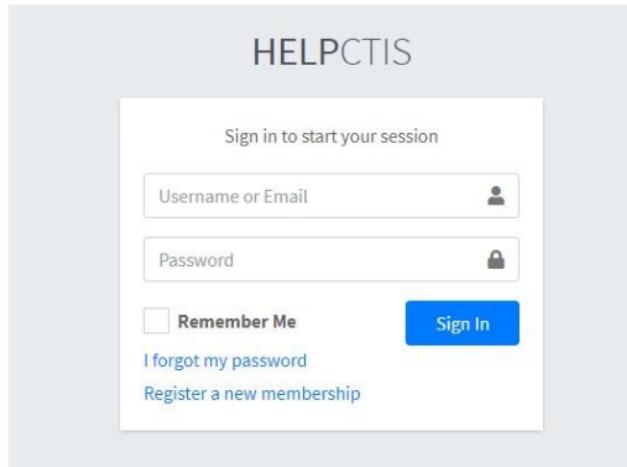
Confirmation message when the user press the Log Out button



- If the user press the cancel button on the confirmation message, will return to the previous page.



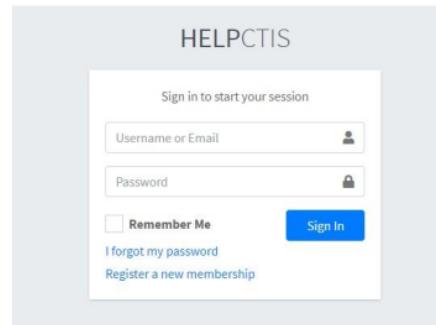
- b. If the user press the log out button on the confirmation message, will be directed to the login page.



Test ID	R-003			
Use Case	Logout			
Test Description	Testing the logout function. User able to logout from the system			
	Test Data	Expected Output	Actual Output	Remark(s)
Test Case 1:	Click logout button	Confirmation message will be displayed	Confirmation message will be displayed	Pass
Test Case 2:	Click cancel button in the confirmation message	The system will return to previous page	The system will return to previous page	Pass
Test Case 3:	Click log out button in the confirmation message	The system will show the login page	The system will show the login page	Pass

Non Functional Requirements

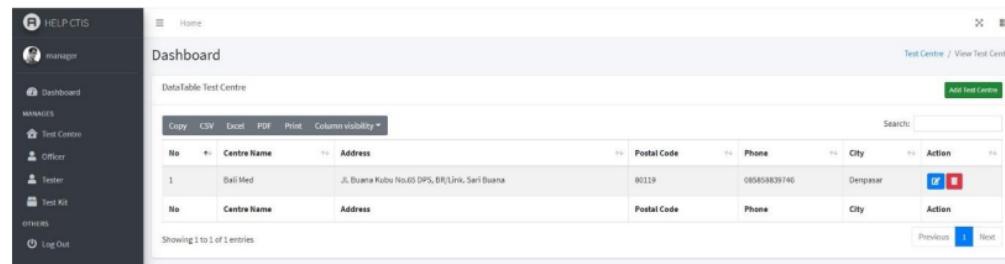
1. Security & Integrity: The login feature to identify users who are connected to the HELP CTIS system.

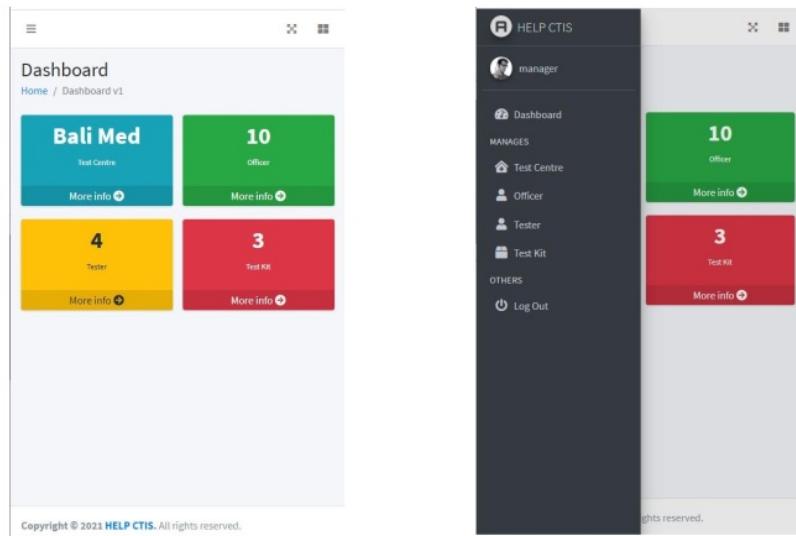


Non-Func – run on Google Version 89.0.4389.114 (Official Build) (64-bit)

Test ID	R-001			
Use Case	Password for login is encrypted			
Test Description	Password login will be encrypted			
	Test Data	Expected Output	Actual Output	Remark(s)
Test Case 1:	1. Username = “manager” 2. Password = “manager”	Password in the text box will be “*****” and in the database table the password is encrypted	Password in the text box will be “*****” and in the database table the password is encrypted	Pass

2. Usability: Website system that is easy to access anywhere on PC, laptop, or smartphone.





3. Correctness: Data alignment where the data entered by the user will be complete sent to the database.

Data entered by the user.

Add Test Centre	
Test Centre Name	Bali Med
Address	Jl. Buana Kubu No.65 DPS, BR/Link. Sari Buana
Postal Code	80119
Phone	085858839746
City	Denpasar
Submit	

Data stored in the database.

	<input type="button" value="←"/>	<input type="button" value="→"/>	<input type="button" value="▼"/>	id	centreName	address	postalCode	phone	city	
	<input type="checkbox"/>	 Edit	 Copy	 Delete	43	Bali Med	Jl. Buana Kubu No.65 DPS, BR/Link, Sari Buana	80119	085858839746	Denpasar

3.5. Test Analysis Report

We have run all of the necessary testing for the system on the functional and non-functional requirements. Based on the testing results, we can conclude that the system is operationally stable and followed all the requirements. The system and codes are functional and have passed the unit, integration, and system the testing phase. We did not find any bugs or errors on our system during our testing. We can ensure that all functions perform flawlessly in accordance with the requirements that have been assigned to them.

3.6. Updated Gantt Chart

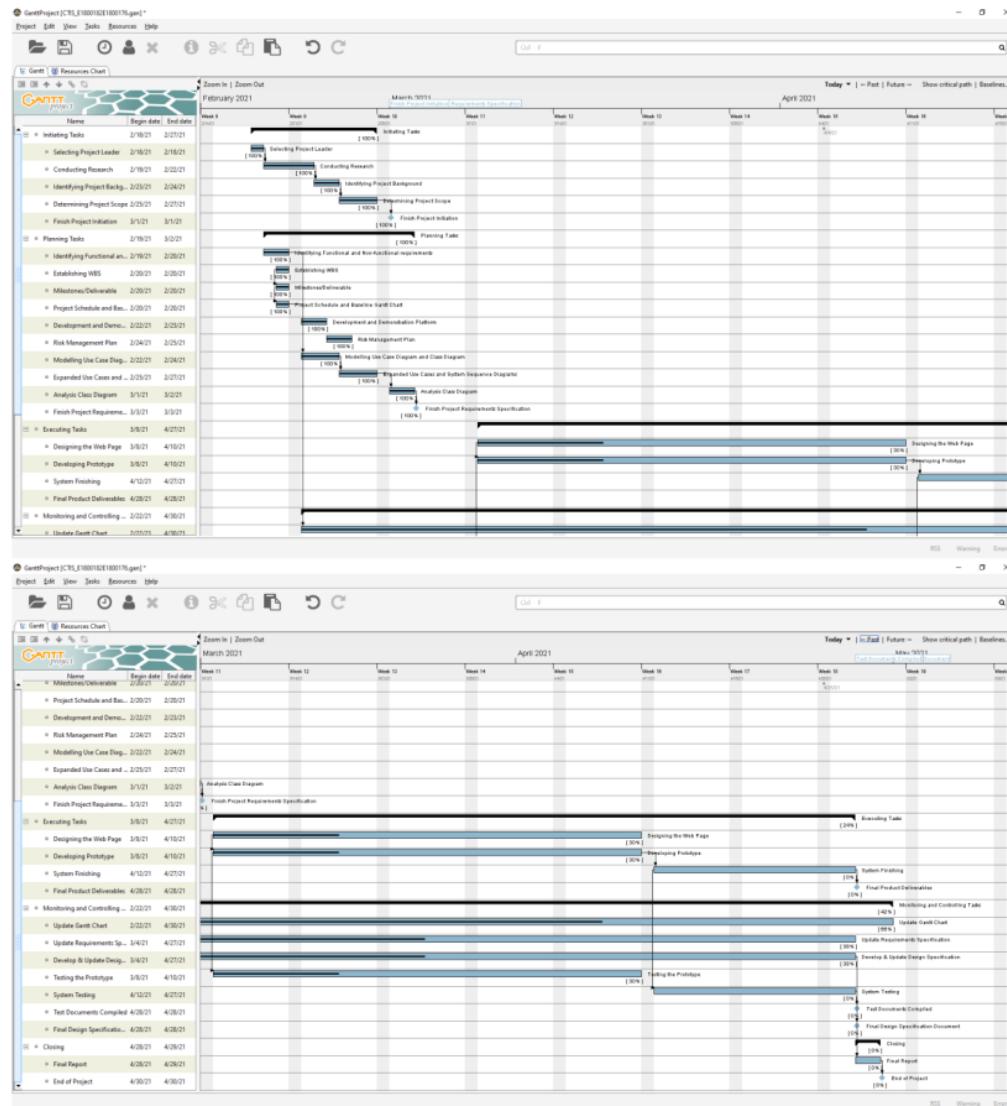


Figure 22. Gantt Chart

3.7. Collaboration via GitHub

GitHub page link: <https://github.com/awidyaandika/help-ctis>

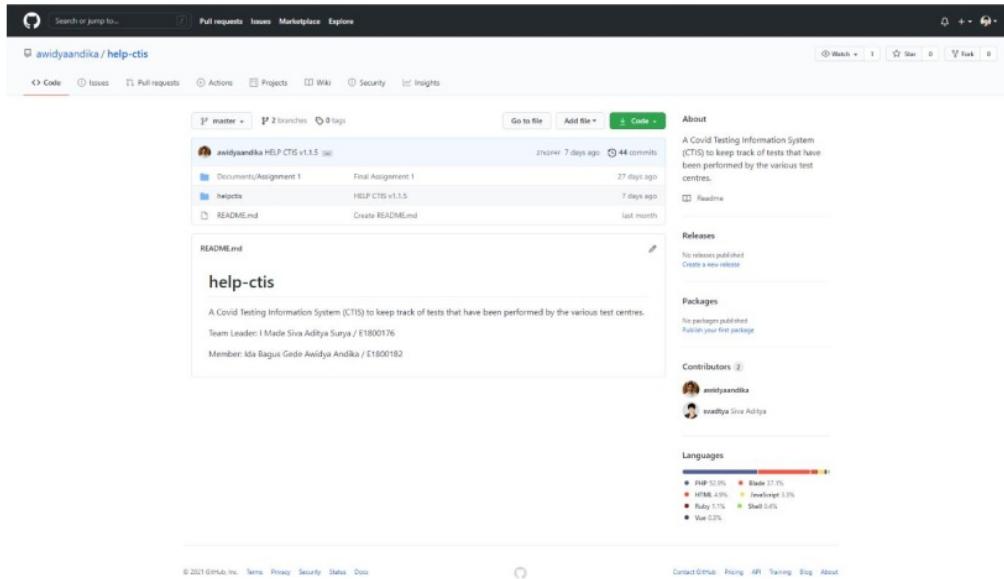


Figure 23. GitHub Main Page

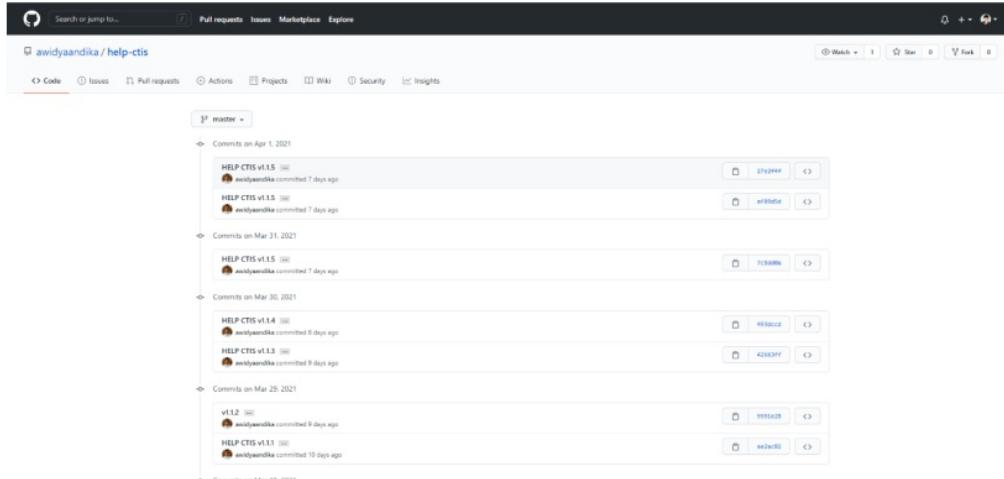


Figure 24. GitHub Commits History

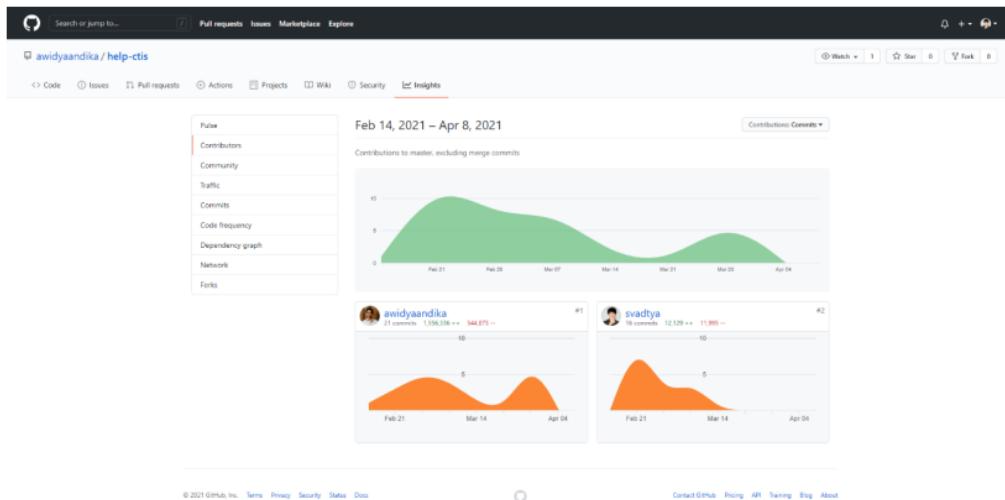


Figure 25. GitHub Contributors Insights

Bibliography

10

Sommerville, I. (2016). *Software engineering 10th Edition*. Pearson Education.

Appendix A: Peer Presentation Evaluation

Reviewer: Ida Bagus Gede Awidya Andika (E1800182)

Reviewing: I Made Siva Aditya Surya (E1800176)

1 Skills	4 Advanced - exceeds expectations	3 Competent - meets expectations	2 Progressing - does not fully meet expectations	1 Beginning - does not meet expectations	Score
Contributions & Attitude	Always cooperative. Routinely offers useful ideas. Always displays positive attitude.	Usually cooperative. Usually offers useful ideas. Generally displays positive attitude.	Sometimes cooperative. Sometimes offers useful ideas. Rarely displays positive attitude.	Seldom cooperative. Rarely offers useful ideas. Is disruptive.	4
Cooperation with Others	Did more than others. Highly productive. Works extremely well with others.	Did own part of workload. Cooperative. Works well with others.	Could have shared more of the workload. Has difficulty. Requires structure, directions, and leadership.	Did not do any work. Does not contribute. Does not work well with others.	3
Focus, Commitments	Tries to keep people working together. Almost always focused on the task. Is very self-directed.	Does not cause problems in the group. Focuses on the task most of the time. Can count on this person.	Sometimes focuses on the task. Not always a good team member. Must be prodded and reminded to keep on task.	Often is not a good team member. Does not focus on the task. Lets others do the work.	3
Ability to Communicate	Always listens to, shares with, and supports the efforts of others.	Usually listens to, shares with, and supports the efforts of others. Usually does most of the talking. Sometimes talks too much. Provides some effective feedback. Relays some basic information that relates to the topic.	Rarely listens to, shares with, or supports the efforts of others. Is always talking and never listens to others. Provides no feedback. Does not relay any information to teammates.		3

Accuracy - documentation/coding etc	Work is complete, well-organized, error-free, and done on time or early.	Work is generally complete, meets the requirements of the task, and is mostly done on time.	Work tends to be disorderly, incomplete, inaccurate, and is usually late.	Work is generally sloppy and incomplete, contains excessive errors, and is mostly late.	3	
				TOTAL	16/20	

Questions:

- 2
1. How effectively did your group work?

In my opinion, our group has been working effectively. This can be achieved due to the neatly arranged and planned division of tasks from the start. For example, whether I or my group friends complement each other if there are problems that cannot be resolved individually, we also provide input and suggestions to each other so that this project can be better.

- 5
2. Were the behaviors of any of your team members particularly valuable or detrimental to the team? Explain.

In my opinion, my partner named Siva, has a perfectionist attitude, where he always looks at little things in detail. From this attitude, it has both positive and negative impacts. The positive thing is to minimize carelessness and mistakes when making a project, but the negative impact of this slows down the completion of a work on the project due to having to revise it to get the appropriate results. But I personally do not feel affected by his attitude because Siva and I have often worked on projects together.

- 2
3. What did you learn about working in a group from this project that you will carry into your next group experience (e.g. assignment 3)?
I have high appreciation commitment for this group has working on HELP CTIS project. With Siva in this group, it makes me feel motivated to carry out this project in an efficient, detailed and on time.

Reviewer: I Made Siva Aditya Surya (E1800176)

Reviewing: Ida Bagus Gede Awidya Andika (E1800182)

1 Skills	4 Advanced - exceeds expectations	3 Competent - meets expectations	2 Progressing - does not fully meet expectations	1 Beginning - does not meet expectations	Score
Contributions & Attitude	Always cooperative. Routinely offers useful ideas. Always displays positive attitude.	Usually cooperative. Usually offers useful ideas. Generally displays positive attitude.	Sometimes cooperative. Sometimes offers useful ideas. Rarely displays positive attitude.	Seldom cooperative. Rarely offers useful ideas. Is disruptive.	4
Cooperation with Others	Did more than others. Highly productive. Works extremely well with others.	Did own part of workload. Cooperative. Works well with others.	Could have shared more of the workload. Has difficulty. Requires structure, directions, and leadership.	Did not do any work. Does not contribute. Does not work well with others.	4
Focus, Commitments	Tries to keep people working together. Almost always focused on the task. Is very self-directed.	Does not cause problems in the group. Focuses on the task most of the time. Can count on this person.	Sometimes focuses on the task. Not always a good team member. Must be prodded and reminded to keep on task.	Often is not a good team member. Does not focus on the task. Lets others do the work.	4
1 Ability to Communicate	Always listens to, shares with, and supports the efforts of others. Provides effective feedback. Relays a lot of relevant information.	Often listens to, shares with, and supports the efforts of others. Sometimes talks too much. Provides some effective feedback. Relays some basic information that relates to the topic.	Rarely listens to, shares with, or supports the efforts of others. Usually does most of the talking. Rarely listens to others. Provides little feedback. Relays very little information that relates to the topic.	Provides no feedback. Does not relay any information to teammates.	3

Accuracy - documentation/coding etc	Work is complete, well-organized, error-free, and done on time or early.	Work is generally complete, meets the requirements of the task, and is mostly done on time.	Work tends to be disorderly, incomplete, inaccurate, and is usually late.	Work is generally sloppy and incomplete, contains excessive errors, and is mostly late.	3	TOTAL	18/20
--	--	---	---	---	---	-------	-------

Questions:

- 2**
1. How effectively did your group work?

Our group has worked on this project quite effectively. We always carry out discussions and brainstorms before working on a task to equalize concepts among team members and avoid misunderstandings. In my opinion, we have divided the work of the task between the team members very well, which is in accordance with the abilities and experience of the team members so that all members can execute tasks properly.

- 2**
2. Were the behaviors of any of your team members particularly valuable or detrimental to the team? Explain.

My team member Awidya Andika is a very cooperative person, hardworking, committed, and very knowledgeable in website development. All the tasks he does are always completed on time, sometimes even ahead of the set schedule. Even though his work is not always perfect, his persistence and hard work really helped me as the leader of this project. Without the contribution from Awidya Andika, the quality of our project would have been considerably diminished.

- 2**
3. What did you learn about working in a group from this project that you will carry into your next group experience (e.g. assignment 3)?

I learned a lot from my team member Awidya Andika who is diligent and hardworking. As a leader, I often can't complete assignments on

time. I began to reflect on myself and focus on improving my time management skills. So that the implementation of tasks in the next assignment can be carried out properly according to the schedule plan

Assignment 2 (E1800176 & E1800182)

ORIGINALITY REPORT



PRIMARY SOURCES

1	teaching.cornell.edu Internet Source	12 %
2	www.coursehero.com Internet Source	1 %
3	es.scribd.com Internet Source	1 %
4	blog.basaksecmen.com Internet Source	1 %
5	docplayer.net Internet Source	<1 %
6	phpinfo.in Internet Source	<1 %
7	Fu-Shing Sun, Xueying Kong, Yi-Hua Weng. "An Interactive Web System for Group Project Management and Peer Evaluation", Proceedings of the 2nd International Conference on Information System and Data Mining - ICISDM '18, 2018 Publication	<1 %

8	140.115.155.41 Internet Source	<1 %
9	researchstation.jp Internet Source	<1 %
10	m.scirp.org Internet Source	<1 %
11	myassignmenthelp.com Internet Source	<1 %
12	repository.um.edu.my Internet Source	<1 %
13	www.itsolutionstuff.com Internet Source	<1 %
14	www.mdpi.com Internet Source	<1 %
15	Juan Carlos Cabanillas-Noris, Mario Ivan Martinez-Hernandez, Ildefonso Leon-Monzon, Juan Manuel Mejia-Camacho et al. "Analysis and Perspectives of Requirements for Detector Control Systems in High- Energy Physics Experiments", 2020 8th International Conference in Software Engineering Research and Innovation (CONISOFT), 2020 Publication	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		