

**KISI-KISI LKS
WEB TECHNOLOGIES**



**PROVINSI JAWA TIMUR
2022**

CLIENT SIDE MODULE

CONTENTS

This module has the following files:

1. MODULE_CLIENT_SIDE.doc
2. MODULE_CLIENT_SIDE_MEDIA.zip

INTRODUCTION

In recent years, people are intrigued to know about playing music in their devices. But, sometimes just playing music out loud is not enough to entertain people nowadays. People are more exhilarated if they can play a music game on their devices.

You are asked to develop a game called **Piano** using HTML and CSS and develop client-side programming using JavaScript. Some media files are available to you in a zip file. You can create more media and modify anything in the media if you want. Your game needs to be developed in a tablet resolution (960x600 pixels). In bigger resolution, the game must be centred in the screen both horizontally and vertically.

DESCRIPTION OF PROJECTS AND TASKS

This is a module of 4 hours. Your first 2 hours must be used to create the design of the game in three PNG images and the initial layout using HTML/CSS. Your layout should follow the design that you created. The final 2 hours you will create the functionality of the game using JavaScript that allows the game to work correctly in different web browsers.

Piano game screen should have meet these requirements below:

1. Game title
2. 4 lane block
3. Piano keys in the bottom
4. A border between keys and lane
5. Total time elapsed
6. Total score percentage
7. Timer

Design and Initial layout

1. **Deliver at least 3 PNG images that present:**
 - a. Game instruction: The first screen of the game presents the game icon, instructions of gameplay, a select option to choose the music to play, and the Play button.
 - b. Game screen: It must present all the elements described above in the game screen.
 - c. Game icon which represents the game immediately.
2. **Develop the initial markup (HTML + CSS) of your game application.** Overall screen must be within 1024x1024 pixels and centered on the screen.
3. **The design should be delivered in dark mode color.** You are free to choose dark color as long as it has the user convenience.
4. **Piano keys should be blue colored.** The background for the keys must be #3e8db1 and the hover state must be #66CFFF.
5. **You are free to decorate** the game screen design as long as it meets the requirements.
6. **The HTML and CSS** code must be valid in the W3C standards for HTML5 and CSS3 rules in accordance with the WCAG and standard ARIA (Accessible Rich Internet Applications Suite)

You are given music.json file for every song provided.

Game functionalities

1. **Show the game icon** as a favicon.
2. **Show game instruction** in the centre after page are loaded.
3. **Show select field for player** to choose song. The song list are included in media files.
4. **Player can start game** after click "Play" button at the bottom of game instruction.
5. **The "Play" button should be disabled** if the user did not select the song.
6. **Show countdown for three seconds in the center of screen** after user clicked the play button before the game started playing.
7. **Tiles should move and hit** the border at the exact given time.
8. **To click the keys**, player can use "DFJK" keys with the following (from the left key):
 - a. 'D' for the first key
 - b. 'F' for the second key
 - c. 'J' for the third key
 - d. 'K' for the fourth key
9. **The score will be increased** if player click the keys when the tiles is touched the border.
10. **Score** is served as percentage from total tiles clicked divided by total tiles passed.
11. **The tile will be vanished** when player clicked at the right timing or it passed the border without clicked.
12. **Click tolerance** for click distance is permitted for 100ms back and forth from the tile.

13. **Lights appeared above the border** when player clicked the key. The light should be appeared which is a gradient from transparent to solid color vertically. The light is located above the border which the key is clicked.
14. **Player fails** if the score below 50% in the end.
15. **Show alert after song finished** to display the score, high score, and winning status.
16. **Save the score to the browser.**
17. **Maintain your HTML/CSS and JavaScript code organized and clean to facilitate future maintenance.** Use correct indentation and comments. Use meaningful variable names and document your code as much as possible so another developer would be able to modify your work in the future.
18. The game needs to work correctly in **Google Chrome** and **Mozilla Firefox**.
19. **Player can change the tile move speed**
20. Even if the move speed increased, the tile must be **consistently** hit the border at the given time.
21. **Player can increase or decrease the speed** by clicking CTRL and (+) or CTRL and (-). An indicator should appear to indicate that the speed is changed.
22. **Player can pause** the game
23. **Press Esc** to open the pause popup. Player can change the volume by moving the range. The game should be in paused state when opening the popup.
24. **Press Esc again** to continue or click the continue button.
25. **Restart button is exists in the popup** for user to restart the game.
26. **Display countdown from 3 at the center of screen** before playing after user continue or restart the game.

INSTRUCTION FOR COMPETITORS

1. The media files are available in the ZIP file. You can modify the supplied files and create new media files to ensure the correct functionality and improve the application.
2. Save your design files in a folder called "**client-side/design**".
3. The entry file should be '**client-side/index.html**'
4. You should create additional images for each of the requested resolutions to highlight hidden elements, animations, interactions, or any additional information that will assist in the presentation of the game design.
5. Additional file names
 - a. Instructions: XX_instructions_1.png, XX_instructions_2.png
 - b. Game board: XX_game_screen_1.png, XX_game_screen_2.png
 - c. Game icon: XX_game_icon.png
6. Save any image source files to a folder named "**source**" inside the "**client-side/design**" folder. The source files are the files that contain the design layers, development files, i.e. xd, fig, psd, ai, or svg.
7. Save the working game to the directory on the server named "**client-side**". Be sure that your main file is called index.html.

8. You are responsible for the time management in your development. If you finalize some tasks you can continue to other tasks.

Example

These following images are for example purpose only. You may design your own game layout.

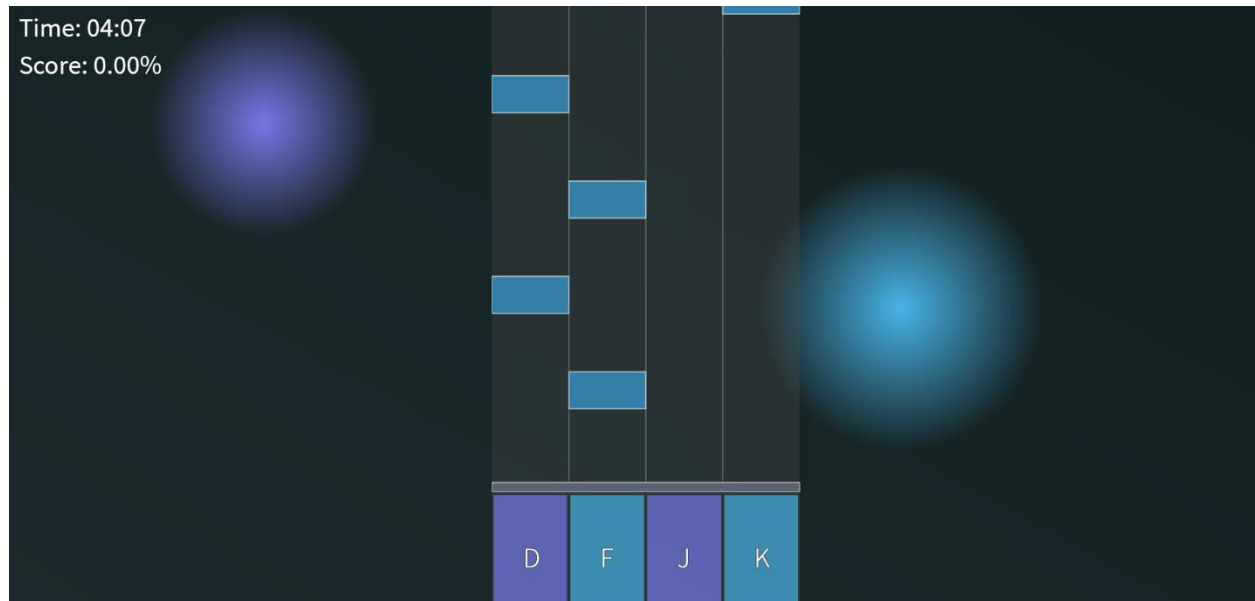


Figure 1 Gameplay Example

SERVER-SIDE MODULE

CONTENTS

This module has the following files:

1. MODULE_SERVER_SIDE.doc
2. MODULE_SERVER_SIDE_MEDIA.zip

INTRODUCTION

We know that the pandemic is not over yet and vaccination is something that must be done to be able to reduce the death rate against the COVID-19 attack. So the government appoints you as a competent person to create a vaccination registration platform.

The goal of this platform is so that societies can register vaccinations on the platform without having to come to the vaccine spot directly and can also see when and where society can come to the vaccine spot.

DESCRIPTION OF PROJECTS AND TASKS

Vaccination registration involves the society platform (request consultation and register first/second vaccination), doctor platform (see and accept/reject consultation from society), and officer platform (update vaccination data from society such as giving vaccines and determining who the vaccinator is).

But your scope of work on this module is only to **create a society platform**. This module is divided into two phases:

- Phase one for creating a Society Vaccination REST API using one of provided PHP frameworks
- Phase two for building a Frontend using provided **JavaScript** libraries (react/angular/vue) of Society Vaccination Platform

Phase one - Vaccination REST API

Authentication

Before a society performs any activity on the society platform, they must be logged in using their **ID card number and password**. After successfully logging in, the token will be automatically generated from the **md5 of society ID card number**.

Specifications :

Feature: A1 - Login and Logout as society: as a visitor, i want to login so that i can enter the system and register a vaccination.

Scenario: Society Login

Request:

URL: [domain]/api/v1/auth/login

Method: POST

Header:

Body: { id_card_number: "SomeText", password: "SomeText" }

Response:

A1a - If login success

Header: Response code: 200

Body: {

 "name": "Siti Puspita",

 "born_date": "1974-10-22",

 "gender": "male"|"female",

 "address": "Ki. Raya Setiabudhi No. 790",

 "token": "e96aaafb6f2f76460b8cc93723bd030e",

 "regional": {

 "id": 1,

 "province": "DKI Jakarta",

 "district": "Central Jakarta"

```
}
```

```
}
```

A1b - If ID Card Number or Password incorrect

Header: Response code: 401

Body: { "message": "ID Card Number or Password incorrect" }

Scenario: Society Logout

Request:

URL: [domain]/api/v1/auth/logout

Params: { token }

Method: POST

Header:

Body:

Response:

A1c - If logout success

Header: Response code: 200

Body: { "message": "Logout success" }

A1d - If logout invalid token

Header: Response code: 401

Body: { "message": "Invalid token" }

Request Consultation

To be able to register a vaccine, the society must consult with a doctor on the platform first. The society can view consultation status and make consultation requests. The society can only make one vaccination request.

Specifications:

Feature: A2 - Request Consultation: As a society, I want to request consultation so that I can register for vaccination when my consultation is accepted.

Scenario: Request consultation

Request:

URL: [domain]/api/v1/consultations

Params: { token }

Method: POST

Header:

Body: { disease_history: "SomeText", current_symptoms:
"SomeText" }

Response:

A2a - If request consultation success

Header: Response code: 200

Body: { "message": "Request consultation sent successful" }

A2b - If invalid token

Header: Response code: 401

Body: { "message": "Unauthorized user" }

Scenario: Get society consultation

Request:

URL: [domain]/api/v1/consultations

Params: { token }

Method: GET

Header:

Body:

Response:

A2c - If success

Header: Response code: 200

Body: {

```
      "consultation": {
        "id": 1,
        "status": "pending"|"accepted"|"rejected",
        "disease_history": "SomeText"|null,
        "current_symptoms": "SomeText"|null,
        "doctor_notes": "SomeText"|null,
        "doctor": { DOCTOR }|null
      }
    }
  }
```

A2d - If invalid token

Header: Response code: 401

Body: {"message": "Unauthorized user"}

Vaccination Spot

When the consultation has been accepted by the doctor and society will register the vaccination, the society can choose the vaccination spot based on the society region. List of available vaccines at each vaccine spot should also be displayed.

In the spot detail, calculate the number of vaccination registrations at that spot and on the requested date in the date param (optional and default today).

Specifications:

Feature: A3 - Vaccination Spot: As a society, I want to see vaccination spots in my region.

Scenario: Get all vaccine spots by society region

Request:

URL: [domain]/api/v1/spots

```
Params: { token }

Method: GET

Header:

Body:

Response:

A3a - If success

Header: Response code: 200

Body: {
  "spots": [
    {
      "id": 1,
      "name": "Purnawati Hospital",
      "address": "Jr. Gotong Royong",
      "serve": 1|2|3,
      "capacity": 15,
      "available_vaccines": {
        "Sinovac": true,
        "AstraZeneca": false,
        "Moderna": true,
        "Pfizer": true,
        "Sinnopharm": false
      }
    },
    ...,
```

```
    ]  
  }  
  
A3b - If invalid token  
  
Header: Response code: 401  
  
Body: { "message": "Unauthorized user" }
```

Scenario: Get spot detail by spot ID and date

Request:

```
URL: /api/v1/spots/<spot_id>  
  
Params: { token, date (optional and default today) }  
  
Method: GET  
  
Header:  
  
Body:
```

Response:

```
A3c - If success  
  
Header: Response code: 200  
  
Body: {  
  "date": "October 21, 2021",  
  "spot": {  
    "id": 1,  
    "name": "Purnawati Hospital",  
    "address": "Jr. Gotong Royong No. 264",  
    "serve": 2,  
    "capacity": 15
```

```
    },  
    "vaccinations_count": 12  
  }  
}
```

A3d - If invalid token

Header: Response code: 401

Body: { "message": "Unauthorized user" }

Vaccination Registration

The society can register for vaccination if the society consultation has been accepted by the doctor. At the vaccination register, if the society has finished the first vaccination, it will automatically register the second vaccination.

Society can register for a second vaccination on condition that they have to wait 30 days after the first vaccination day and each society can only register for vaccination 2 times. Society can be said to be vaccinated if it has received a dose of the vaccine.

Spot ID and date must be filled where the date format is *YYYY-MM-DD*.

Specifications:

Feature: A4 - Vaccination: as a society, i want to register vaccination and get all vaccination data.

Scenario: Register Vaccination

Request:

URL: /api/v1/vaccinations

Params: { token }

Method: POST

Header:

Body: { spot_id: 1, date: "2021-10-26" }

Response:

A4a - If success

Header: Response code: 200

Body: { "message": "First|Second vaccination registered
successful" }

A4b - If invalid token

Header: Response code: 401

Body: { "message": "Unauthorized user" }

A4c - If the society consultation hasn't accepted by doctor

Header: Response code: 401

Body: { "message": "Your consultation must be accepted by
doctor before" }

A4d - If invalid fields

Header: Response code: 401

Body: {
 "message": "Invalid field",
 "errors": {
 "date": [
 "The date does not match the format Y-m-d."
],
 "spot_id": [
 "The spot id field is required."
]
 }
}

A4e - If 2nd vaccination < 30 days from 1st vaccination

Header: Response code: 401

Body: { "message": "Wait at least +30 days from 1st
Vaccination" }

A4f - If have been 2x vaccinated

Header: Response code: 401

Body: { "message": "Society has been 2x vaccinated" }

Scenario: Get all of society vaccinations

Request:

URL: [domain]/api/v1/vaccinations

Params: { token }

Method: GET

Header:

Body: Response:

A4g - If success

Header: Response code: 200

Body: {

 "vaccinations": {

 "first": {

 "queue": 1,

 "dose": 1,

 "vaccination_date": "2021-10-21",

 "spot": {

 "id": 1,

 "name": "Purnawati Hospital",

 "address": "Jr. Gotong Royong",

 "serve": 2,

 "capacity": 15,

 "regional": {

 "id": 1,

 "province": "DKI Jakarta",

 "district": "Central Jakarta"

 }

 },


```

        "status": "done",

        "vaccine": {

            "id": 1,

            "name": "Sinovac"

        },

        "vaccinator": {

            "id": 1,

            "role": "doctor",

            "name": "Dr. Halima Yuniar"

        }

    },

    "second": <VACCINATION>|null

}

}

```

A4h - If invalid token

Header: Response code: 401

Body: { "message": "Unauthorized user" }

Phase two - Society Vaccination Platform (Frontend)

In this phase, you can use the gui template provided in the media files to build a society platform. Use the REST API following :

- URL : https://localhost/xx_module_server

Implement REST API into the frontend. All response formats must be in JSON form.

Society login and logout

To be able to enter the system, the society must first login using the ID card number and password. When the society has entered the system, the logout button will appear, and the society can exit the system by clicking the logout button

Specifications:

Feature: B1 - Society login and logout: as a visitor, i want to login into society vaccination platform and logout to exit the system.

Scenario: B1a - Society login success

Given I am on the login page

And I enter correct id_card_number and password

Then I redirected to society dashboard page

Scenario: B1b - Society login fails

Given I am on the login page

And I enter wrong id_card_number or password

Then I stay in the login page and I see error message "ID Card Number or Password incorrect"

Scenario: B1c - Society logout success

Given I am on the any page that has been logged in

And I press logout button

Then I get message "logout success" and redirected to login page

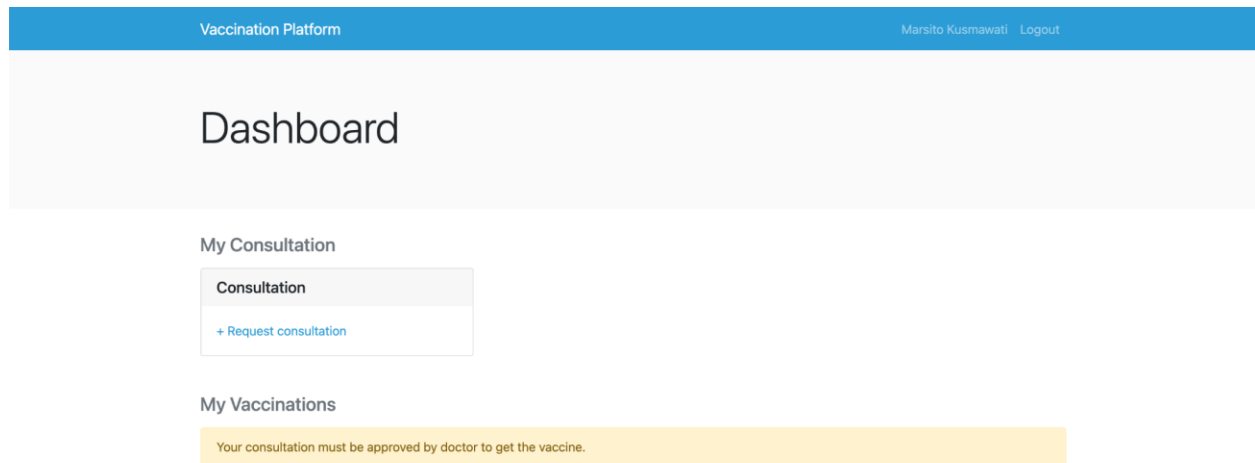
And I cannot reach the previous page by going back

Society Dashboard

In the dashboard page, there are two sections, consultation and vaccination section. where the society must request a consultation and the consultation is approved by the doctor to be able to register for vaccination.

Specifications:

Feature: B2 - Society Dashboard: as a society, i want to see consultation info and my vaccination list in the dashboard page.



Scenario: B2a - Consultation section is displayed

Given I am on dashboard page

Then I see the consultation section with "My Consultation" title

Scenario: B2b - Link to request consultation appears when society hasn't consultation before

Given I am on dashboard page

And I haven't request consultation before

Then I see request consultation link in the consultation section

Scenario: B2c - Show consultation info

My Consultation

Consultation	
Status	Accepted
Disease History	diabetes
Current Symptoms	flu
Doctor Name	Dr. Ratna Pradipta
Doctor Notes	ok

Given I am on dashboard page

And my request consultation is sent or accepted/rejected by doctor

Then I see consultation info in the consultation section

Scenario: B2d - Vaccination list section is displayed

Given I am on dashboard page

Then I see the vaccination list section with "Vaccinations" title

Scenario: B2e - Show vaccination info message

Given I am on dashboard page

And I haven't request consultation before

Then I see "Your consultation must be approved by doctor to get the vaccine" in the vaccination list section

Scenario: B2f - The first vaccination box is displayed when society consultation have been accepted by doctor

My Vaccinations

First Vaccination

[+ Register vaccination](#)

Given I am on dashboard page

And I my request consultation has been accepted by doctor

Then I see the first vaccination box with link is displayed in the vaccination list section

Scenario: B2g - The second vaccination box is displayed when the first vaccination is finished

My Vaccinations

First Vaccination	
Status	Vaccinated
Date	October 27, 2021
Spot	Usamah Hospital
Vaccine	Sinovac
Vaccinator	Dr. Raina Pradipta

Second Vaccination

[+ Register vaccination](#)

Given I am on dashboard page

And my first vaccination is finished

Then I see the second vaccination box with link is displayed in the vaccination list section

Scenario: B2h - The vaccinations box show vaccination info correctly when vaccination is finished

Given I am on dashboard page

And my first or second vaccination is finished

Then I see vaccination info (status, date, spot, vaccine, and vaccinator) in the vaccination box

Request Consultation

In the consultation request page, the society can fill in the history of the disease and current symptoms if they have by selecting "Yes, I have" in each dropdown question.

Specifications :

Feature: B3 - Society Consultation: as a society, i want to request consultation.

The screenshot shows the 'Request Consultation' page of the 'Vaccination Platform'. At the top, there is a blue header bar with 'Vaccination Platform' on the left and 'Login' on the right. Below the header, the page title 'Request Consultation' is centered in a large, dark font. The main content area contains a form with the following elements: a dropdown menu labeled 'Do you have disease history?' with 'Yes, I have' selected; a large text area labeled 'Describe your disease history'; a dropdown menu labeled 'Do you have symptoms now?' with 'No' selected; and a blue button labeled 'Send Request' at the bottom.

Scenario: B3a - Consultation form is displayed correctly

When I press "Request consultation" button in the dashboard page

Then I see the consultation form (disease history and current symptoms) is displayed

Scenario: B3b - Show textarea form by choosing "Yes, I have"

Given I am on request consultation page

Then I select "Yes, I have" of dropdown in each consultation form,
then the textarea will be displayed.

Scenario: B3c - Request consultation successful

Given I am on request consultation page

And I press request button

Then the request consultation is sent, the "Request consultation
successful" message is appear, and I redirected to dashboard page

Vaccination Spots List

When the society registers for vaccination by pressing the "Register vaccination" button on the dashboard page, the society is directed to the list vaccination spot page to select a vaccination spot. In the item spot, display the vaccines available in that spot correctly.

Vaccination spot are displayed by society region. In each place the vaccine has a different service where :

- Serve 1 : only first vaccination
- Serve 2 : only second vaccination
- Serve 3 : both

So, make the vaccination spot unavailable when the vaccination registration cannot be served by the vaccination spot.

Specifications :

Feature: B4 - Vaccination Spots List: as a society, I want to see a
list of vaccination spots by my region.

First Vaccination

List Vaccination Spots in Central Jakarta

Usamah Hospital Ds. Abdullah No. 31, DKI Jakarta	Available vaccines Sinovac, Moderna, Pfizer.	Serve Only first vaccination
Nasyidah Hospital Ki. Bakau Griya Utama No. 476, DKI Jakarta	Available vaccines Sinovac, AstraZeneca, Moderna, Pfizer.	Serve Only second vaccination
Napitupulu Hospital Jln. Laswi No. 228, DKI Jakarta	Available vaccines Sinovac, AstraZeneca, Sinnopharm.	Serve Both vaccination

Scenario: B4a - Show list vaccination spots by society region

Given I am on vaccination spot list page

Then I see a list of vaccination spots in my region

Scenario: B4b - Show spot information for each spot item

Given I am on vaccination spot list page

Then I see spot information (spot name, address, available vaccines, and serve) in each spot item.

Scenario: B4c - Spot item unavailable when the spot can't be serve

Given I am on vaccination spot list page

Then I see spot that can't be serve unavailable

Detail Vaccination Spot

In the vaccination spot detail page, there are the 3 sessions :

- Session 1 : 09.00 - 11.00

- Session 2 : 13.00 - 15.00
- Session 3 : 15.00 - 17.00

In each section box, there is a queue order that has been divided based on the number of capacity at the vaccination spot. For example, if the capacity of the vaccination spot is 15, so in each section box there are 5 slots.

Specifications :

Feature: B5 - Detail Vaccination Spot: as a society, I want to see detail of vaccination spot and see how many registrations at the spot based on the selected date.

The screenshot displays the 'Vaccination Platform' interface for Napitupulu Hospital. The header includes the hospital name and address, a 'Register vaccination' button, and user information (Marsito Kusmawati, Logout). Below the header, a date selector is set to '26/10/2021'. Three sessions are listed:

- Session 1** (09:00 - 11:00): Slots #1, #2, #3, #4, #5.
- Session 2** (13:00 - 15:00): Slots #6, #7, #8 (highlighted in blue), #9, #10.
- Session 3** (15:00 - 17:00): Slots #11, #12, #13, #14, #15.

Scenario: B5a - Three sections and slots displays correctly

Given I am on vaccination spot detail page

Then I see three sections with slots that have been divided according to the capacity of the vaccine spot is displayed correctly. For example, if spot capacity is 15, so, in each section each has 3 slots.

Scenario: B5b - Can see vaccinations info by choosing date

Given I am on vaccination spot detail page

And I select a date

Then I see how many vaccination slots have been booked on the selected date at that spot

Scenario: B5c - Mark other society vaccinations by green border

Given I am on vaccination spot detail page

Then I see the slots that filled with other society's vaccinations green border

Scenario: B5d - Mark my vaccination slot by blue background

Given I am on vaccination spot detail page

Then I see the slot that i will is blue background

Scenario: B5e - Register vaccination

Given I am on vaccination spot detail page

And I press "Register vaccination" button

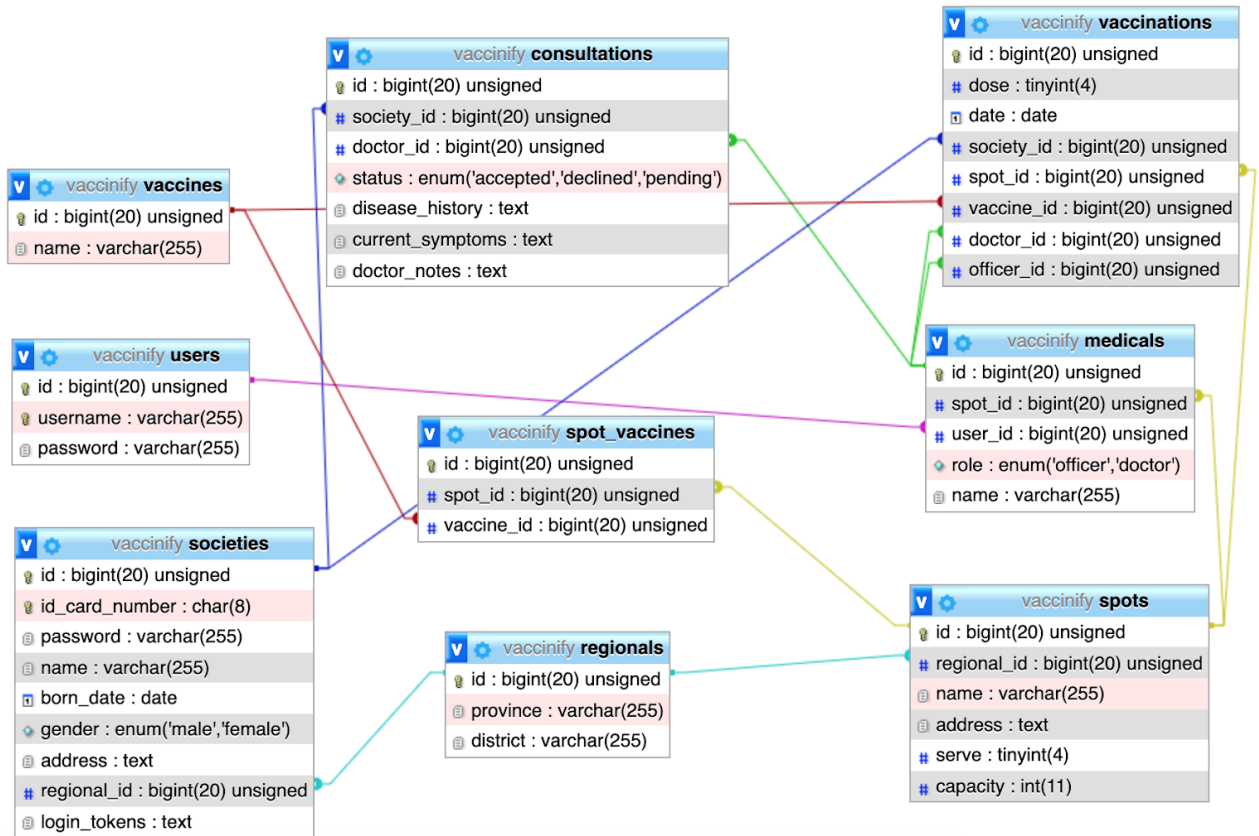
Then the success or error message is appear and I redirected to dashboard page

Notes:

- For each response from the REST API must be displayed.
- Build the frontend into production mode first before your frontend project is pushed and the REST API must be consumed from https://localhost/xx_module_server

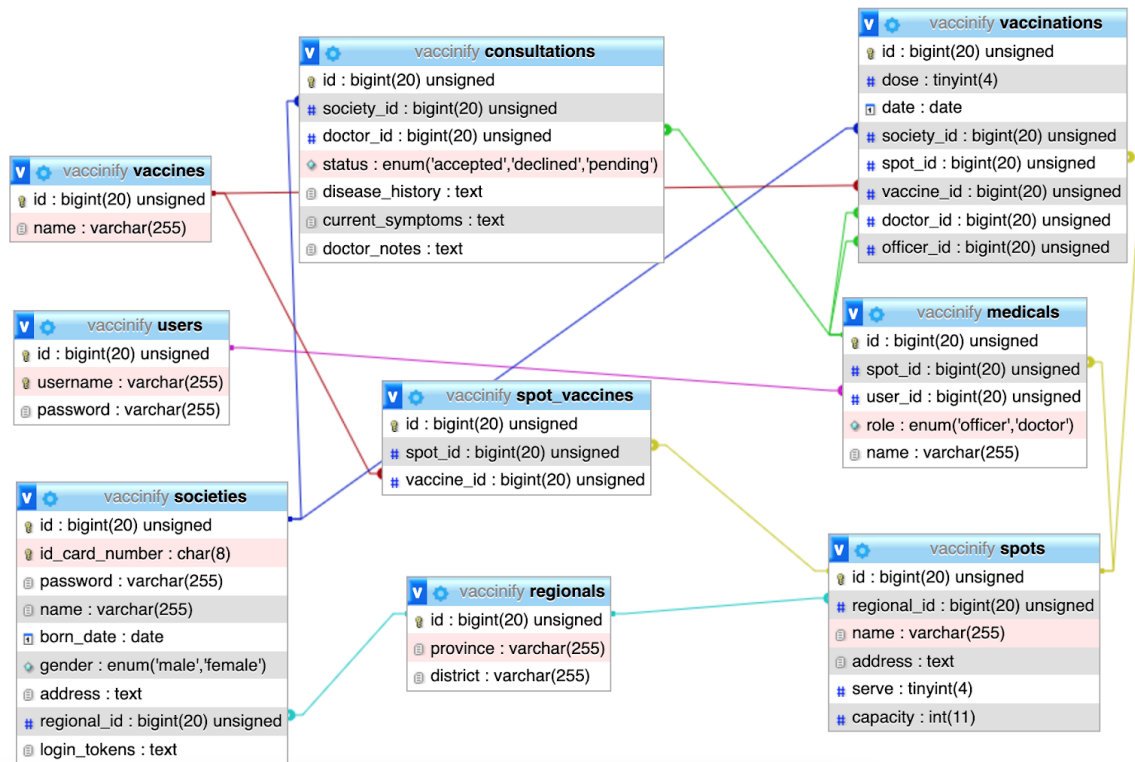
ERD

You can use and improve ERD below:



INSTRUCTION FOR COMPETITORS

- Import the database that has been provided in db-dump in the media files folder.
- Save your work in the "serverside" folder :
 - Vaccination REST API : serverside/backend/
 - Society Vaccination Platform : serverside/frontend/
- For the frontend, you must be built to production and save in serverside/frontend/dist/ folder.
- Postman environment and collection provided in SERVER_SIDE_MEDIA/postman/
- You can check community accounts based on vaccination stages by looking at the accounts that have been provided in the folder SERVER_SIDE_MEDIA/account/
- Use the gui templates provided in the media files.
- Create/generate a DB-diagram named "db-diagram.xxx" (xxx is the extension/type of the file eg. pdf or jpg) and put it into the directory mentioned above. Example:



- For this module, you must use one of the three available frameworks provided. Applications developed without use of any of these frameworks will not be considered. You should take advantage of the framework as much as possible.