assignment12_category_0016

Important!

*** Start by reading all the instructions carefully. Don't worry if they seem long; just take it one step at a time. Make a plan on how to do each task and take notes as you go. Break down the big jobs into smaller ones to make it easier. The instructions have a lot of details to save your time from waiting in the support session, but don't be scared. If something is confusing, write it down, and you can ask for help later. Stay organized, take your time, and you'll do great ***

Medical Camp Management System

Project Overview:

The Medical Camp Management System (MCMS) is an user-friendly MERN (MongoDB, Express.js, React, Node.js) application meticulously crafted to revolutionize the efficiency of organizing and managing medical camps. This comprehensive system is tailored to meet the diverse needs of three pivotal user roles: Organizers, Healthcare Professionals, and Participants, ensuring seamless coordination and optimal functionality.

If we have any update we will mention it here]. Do check frequently to see if any updates have been made.

Main Requirements:

1. Unique Design:

- o Develop a distinct and innovative design for the website.
- o Avoid similarities with previous assignments or demo projects.

2. User-Friendly Interface:

o Create an intuitive and easy-to-navigate interface.

3. **Design Resources:**

• Utilize free resources for design from platforms like theme forest.

4. Mutations and Alerts:

- Implement Create, Update, Delete operations with visual alerts.
- Avoid using browser alert() to prevent mark reduction.

Navbar Design:

1. Logo:

 Incorporate the company's logo in the navbar for brand representation.

2. Conditional Buttons:

Register/Login Button:

i. Display "Register" and "Login" buttons conditionally based on whether the visitor is logged in or not.

User Photo and Logout Button:

- i. If a user is logged in, display their profile photo.
- ii. Clicking the user photo reveals a functional "Logout" button.

3. Navigation Items:

- Home:
 - i. Direct users to the home page.

Available Camps (Private):

- i. Visible to logged-in users (Participants, Healthcare Professionals, Organizers).
- ii. Redirects to the "Available Camps" page, showing a list of all available medical camps.

Dashboard (Private):

- i. Visible to logged-in users (Participants, Healthcare Professionals, Organizers).
- ii. Redirects to the dashboard, providing personalized information based on the user's role.

Contact Us (Public):

- i. Accessible to all visitors.
- ii. Directs to the "Contact Us" page for public inquiries.

Home Page:

In the web app's Home Page, the design aims for a captivating and informative layout.

1. Banner / Slider:

- Display a visually appealing slider showcasing the success stories and impactful moments of past medical camps.
- Highlight achievements, positive outcomes, or any relevant achievements of the Medical Camp Management System.

2. Popular Medical Camps Section:

Overview:

i. Showcase a carefully curated collection of popular medical camps(add at least six camps), each offering a comprehensive set of details. This includes Camp Name, Image, Camp Fees, Scheduled Date and Time, Venue Location, Specialized Services Provided, Healthcare Professionals in Attendance, and Target Audience.

Sorting Functionality:

- Implement an interactive sorting feature that allows users to rearrange the list of popular camps based on participant engagement.
- The participant count starts at zero for each camp and increases with each registration.(see camp details page for more registration details)

Participant Count Display:

- Display the current participant count for each camp alongside its details.
- ii. This count dynamically updates as participants register for the respective camps.

Details Page Link:

- Add a dedicated button or link associated with each popular camp, directing users to a detailed information page for that specific camp.
- ii. Maintain the existing route structure (e.g., /camp-details/:campId) and ensure associated components effectively showcase detailed camp information.

Note: Consider adding a "See All Camps" button to redirect users to a page named "Available Camps" after fulfilling the above requirements. (See Available Camps page for more details)

3. Testimonials:

- Expand the testimonials slider to showcase feedback and ratings from participants .
- Include relevant details such as Camp Name, Date, and the feedback/rating provided.

 Ensure that the slider dynamically updates to display the latest and most relevant testimonials.

4. Additional Sections:

- Integrate two more sections that enhance the user's understanding of the system.
- Consider adding a section "Upcoming Camps" to provide a glimpse of future events.

5. Navbar and Footer:

- Include a Navbar for easy navigation and consistent access to key functionalities.
- Add a Footer to provide essential links, contact information, or quick access to other parts of the web app.

Layout:

Ensure that the Navbar and Footer are designed to be visible and accessible from anywhere within the web app, providing a seamless and user-friendly experience.

Camp Details Page:

1. Camp Details Section:

- Display camp details prominently, including name,image, date, venue, and audience.
- Provide a brief and informative description of the camp's purpose and benefits.

2. Join Camp Button:

- o Enable a "Join Camp" button for Participants.
- For Participants: Activate the button for registration.
- For Organizers and Healthcare Professionals: Keep it inactive or replace it with admin actions.

3. Registration Modal:

- Implement a modal for participant registration. Clicking the Join Camp Button will pop up the modal.
- Collect necessary info like participant details(Name, Age, Phone,Gender, Address), Camp Fees(Fee Amount readonly), and requirements(Any specific health-related information,Emergency contact).

• Ensure user-friendly input fields and clear instructions.

4. Submit and Save Data:

- Save entered data in a dedicated database collection.
- o Implement backend logic for data submission and storage.

5. **Route:**

• Maintain the route structure (e.g., /camp-details/:campId).

Available Camps Page:

1. Camp Information:

 Display essential details for each camp, such as Camp Name, image, Scheduled Date and Time, Venue Location, Specialized Services Provided, Healthcare Professionals in Attendance, Target Audience, Comprehensive Description.

2. Join Camp Button:

 Implement a prominent "Details" button next to each listed camp, which will navigate to the details page, enabling participants to express interest and register from the details page using the **Join** Camp Button .

3. Integration with Details Page:

 Establish a clear connection between the "Available Camps" page and the detailed information on the corresponding Details Page.

Additional Features: (Optional but Highly Recommended)

• Search Functionality:

• Introduce a search bar allowing users to search for specific camps based on keywords, dates, or other relevant criteria.

• Sort by Field:

- Enable users to sort the list of available camps based on different criteria, such as:
- Most Registered
- Target Audience
- Alphabetical Order (Camp Name)

Dashboard (Private)

Tailors pages dynamically to user roles—Organizers, Healthcare Professionals, and Participants. Each role enjoys a personalized experience with relevant functionalities based on responsibilities and permissions.

Organizer Routes (Private)

Organizer Profile Management:

Route Endpoint: / organizer-profile

• Purpose:

Enable organizers to manage their profile information, displaying and allowing updates for contact details and preferences. Additional sections highlight the organizer's impact, showcasing organized camps, success stories, and participant feedback.

• User Interaction:

Organizers initiate profile updates by clicking the "Update" button, triggering a modal/form displaying current information. Validation checks ensure data accuracy. Upon successful updates, a success alert is shown; errors are handled appropriately.

Add A Camp:

- Route Endpoint: /add-a-camp
- Form Fields: Camp Name, Image (Upload camp image using ImgBB or other hosting platform), Camp Fees, Scheduled Date and Time, Venue Location, Specialized Services Provided, Healthcare Professionals in Attendance, Target Audience, Comprehensive Description.
- Form Submission: Validate and save entered data in the database.
- Utilize a form handling library such as <u>Formik</u> or <u>React Hook Form</u> for efficient form validation and submission.
- Validate the entered data to ensure accuracy and completeness.

Manage Camps:

Route Endpoint: /manage-camps

Camp List Display: Present a tabular view of all camps created by the specific organizer, displaying essential details for each camp.

Details and Actions:

- For each camp, show details such as Camp Name, Scheduled Date and Time, Venue Location, Specialized Services Provided, Healthcare Professionals in Attendance, Target Audience, and Comprehensive Description.
- Include action buttons for each camp:
 - **Update:** Allows the organizer to modify the camp's information.
 - Update Route Endpoint: /update-camp/:campId
 - **Delete:** Permits the organizer to remove the camp from the system.
 - Delete Route Endpoint: /delete-camp/:campId

• Table Structure:

 Organize and display camp details in a tabular view. Each row represents a camp, with columns containing specific details.

• Update Process:

- Clicking the "Update" button opens a modal or form with pre-filled information for the selected camp.
- o Organizer can modify the necessary details and save changes.

• Delete Process:

- Clicking the "Delete" button prompts a confirmation dialogue to ensure the organizer's intent.
- o Upon confirmation, remove the camp from the system.

Manage Registered Camps:

Route Endpoint: /manage-registered-camps

• Table Structure:

- Organize and display camp details in a clear table format using the "react-table" library. Each row represents a camp, with columns containing specific details. Recommended: Install the library with npm install react-table.(Optional but highly recommended)
- Include columns such as Camp Name, Date and Time, Venue, Camp Fees, Payment Status, Confirmation Status, and Actions.

Additional Fields:

 Payment Status: Clearly indicate whether participants have paid or are yet to pay, offering transparency in financial transactions. Confirmation Status: Initially set to "Pending," and upon clicking the "Pending" button, update to "Confirmed" after successful payment.

Actions:

• Provide intuitive action buttons for each registered camp row:

Cancel:

- Make the "Cancel" button visible only if the participant has completed the payment for the registered camp.
- This ensures that organizers can manage and cancel registrations effectively.

• Payment Handling:

- Implement the Stripe.js integration to securely manage participant payments.
- Initialize the payment status as "Unpaid," indicating pending payments.
- Upon successful payment via Stripe.js, seamlessly update the payment status to "Paid."

• Confirmation Status Update:

Facilitate a streamlined process where clicking the "Pending"
 button triggers an update to "Confirmed" post successful payment.

• Cancellation:

- Empower organizers with the ability to cancel a participant's registration.
- Implement a user-friendly confirmation dialog for cancellation actions.
- Upon confirmation, promptly update the status and ensure real-time reflection of changes in the table for enhanced transparency and control.

Participant Routes (Private)

Profile Management:

• Route: /participant-profile

• Purpose:

Participants access this page to view and edit profile details, including contact information and personal preferences. Unique touches include fields for attended camps and interests in specific medical areas.

• User Interaction:

Update your profile effortlessly by clicking the "Update" button. A modal/form displays current information for easy modification. The system ensures data accuracy through validation checks. Click "Save" or "Update" for changes, receiving a success alert for successful updates; errors are handled accordingly.

Registered Camps:

• Route: /registered-camps

• Table Structure:

- Display a tabular view of camps that the participant has registered for.
- Include columns such as Camp Name, Date and Time, Venue, Camp Fees, Payment Status, Confirmation Status, Actions.
- Utilize the "react-table" library for a dynamic and responsive table.

• Payment Status:

- Display the payment status for each registered camp (Unpaid/Paid).
 - For unpaid camps, include a "Pay" button that redirects participants to the payment page.
 - After successful payment via Stripe, show a notification with the transaction ID.
 - Change the "Pay" button to "Paid," indicating a successful payment, and make it non-clickable.

Confirmation Status:

 Initially set to "Pending" and updated to "Confirmed" after approval by organizers.

• Actions:

- o Include action buttons for each registered camp row:
 - **Cancel:** Allow participants to cancel their registration for a specific camp. Cancel can be possible before payments only, after payment you can disabled the button.
 - Implement a confirmation dialog for cancel actions.
 - On confirmation, update the confirmation status and reflect changes in the table.

<u>Payment History:</u>

Route Endpoint: /payment-history

• Table Structure:

 Display a tabular view of essential fields such as Camp Name, Date and Time, Venue, Camp Fees, Payment Status, and Confirmation Status.

• Payment History Collection:

- After successful payment, record transaction details in a dedicated collection for the participant's payment history.
- Fields include Camp Name, Date and Time, Venue, Camp Fees,
 Payment Status, and Confirmation Status.

• User Interaction:

 Allow participants to conveniently view and track their payment history, gaining insights into past and present financial transactions related to registered camps.

Feedback and Ratings:

Route Endpoint: /feedback-and-ratings

• Page Features:

 Participants can submit comprehensive feedback and ratings for medical camps they have attended.

• Table Structure:

 Display a tabular view of present detailed information on attended medical camps, showcasing fields such as Camp Name, Date and Time, Venue, Camp Fees, Payment Status, Confirmation Status, and Actions. Display only the camps that have been paid for and completed.

• Feedback and Ratings:

 Allow participants to share their experiences through written feedback and provide numerical ratings for various aspects of the medical camps.

• Review Button:

 Introduce an intuitive "Review" button within the actions for each attended camp. Clicking the "Review" button opens a modal or form for participants to input their feedback and ratings, creating a seamless and user-friendly experience.

• Review Submission:

- Upon submitting a review, save the entered data in a dedicated collection, associating it with the specific medical camp and participant.
- Encourage participants to upload images or testimonials along with their feedback, adding a visual and personalized touch to the testimonials.

• Testimonial Section Update:

- Dynamically update the **Home Page** testimonial section to showcase participant reviews.
- Display relevant details such as Camp Name, Date, and participant's feedback alongside any uploaded images or testimonials.

Healthcare Professionals Routes (Private)

Healthcare Professional Profile Management:

• Route: /professional-profile

• Purpose:

Empower healthcare professionals to manage their profile, showcasing details like medical specialty, certifications, and contact information. Additional sections highlight their impact, including camps of interest and accepted camps.

• User Interaction:

Effortlessly update profiles by clicking "Update," triggering a modal with current information. Modify contact details and preferences, ensuring data accuracy with validation checks. Click "Save" for successful updates, and errors are handled appropriately.

Additionals Requirements:

1. Responsive Design

- 2. Use the Environment variable to hide the Firebase config keys and Mongodb credentials.
- 3. Minimum 25 meaningful git commits on the client-side repository.
 - Minimum 15 meaningful commits on the server-side repository.
 - reate a readme on your client-side and add the following info:
 - * Your project name, credentials & live link.
 - # 5 main Features of your projects with bullet points.
- 4. Add a 404 page(not found page)
- 5. Give your website a name. The website title will be changed according to your route/page. Suppose your website name is PHero. Then, on the 'login' router/page, your website title will be 'PHero | Login'
- 6. Elevate the functionality of the Dashboard routes by seamlessly incorporating a Search Bar (search by Camp names or any other field), providing users with a convenient tool for effortless navigation through the displayed tabular data.

Bonus Task:

1. *** Please extend the dashboard functionality by implementing three additional routes. Two of them should be categorized under the Organizers Dashboard, while the third should be part of the Healthcare Professionals Dashboard. ***

Organizer Routes (Private) <u>Add Upcoming Camps:</u>

Route Endpoint: /add-upcoming-camp

Form Fields: Camp Name, Image (Upload using ImgBB or other hosting platform), Camp Fees, Scheduled Date and Time, Venue Location, Specialized Services Provided, Target Audience, Comprehensive Description

Form Submission:

- Validate and save entered data, including the scheduled date and time, in the database.
- Utilize a form handling library such as <u>Formik</u> or <u>React Hook Form</u> for efficient form validation and submission.

Database Storage:

Save the entered data in a dedicated collection in the database.

Upcoming Camps Display:

- The added data will be showcased in the "Upcoming Camps" section on the homepage with some relevant fields. (See *Additional Sections* in the Home Page)
- Each upcoming camp entry will have a "Details" button leading to a page displaying all information about the camp.

Role-Based Buttons:

For Participants:

- In the **Details Page**, A "Join Upcoming Camp" button will be available.
 - Clicking this button opens a modal prompting participants to provide additional information:
 - Participant Details: Collect necessary info like participant details(Name, Age, Phone,Gender, Address), Camp Fees(Fee Amount readonly), and requirements(Any specific health-related information,Emergency contact).
 - Camp-Specific Information: Any specific health-related information or requirements.
 - Upon submission, the participant is added to the growing participant list. Initially the interested participant list count will be zero.
 - Participant details are stored in a collection for organizers' management.

For Healthcare Professionals:

- In the **Details Page**, An "Interested Upcoming" button will be available.
 - Clicking this button opens a modal prompting professionals to provide additional information:
 - Professional Details: Name, Specialization, Contact Information.

- Camp-Specific Information: Any specific requirements or areas of interest.
- Upon submission, the professional's details are saved for organizers to review.
- The number of interested professionals is a condition for organizers to decide whether to publish the upcoming camp.

Manage Upcoming Camps:

Route Endpoint: /manage-upcoming-camps

Table Structure:

 Display a tabular view which includes fields such as Camp Name, Date and Time, Venue, Target Audience, Participant Count, Interested Professionals Count, and Actions.

Actions:

- For each upcoming camp, organizers can:
 - Review Participants: View the list of participants who have expressed interest in joining.
 - Participants' details will include: Name, Age, Phone, Gender, Address, Emergency Contact, and any specific health-related information or requirements.
 - Organizers can use this information to verify and ensure participant readiness for the camp.
 - Review Professionals: View the details of healthcare professionals interested in the camp.
 - Professionals' details will include: Name, Specialization,
 Contact Information, and any specific requirements or areas of interest.
 - Organizers can assess the professional background and suitability for the camp.
 - Accept Participants: Clicking this button will register the participants for the camp.

- Participants will move from the "Interested" list to the registered list of camps.
- Accept Professionals: Clicking this button will change the status of professionals from "Pending" to "Accepted."
- Publish: Activate the upcoming camp for public viewing. The camp becomes visible to both participants and professionals.
 - The "Publish" button is conditional and will be visible only if the participant count is more than 10 and the number of interested professionals is more than 3.
 - Clicking the "Publish" button will move the camp to the "Popular Medical Camps" section on the home page with the updated field Healthcare Professionals in Attendance . You must update the Healthcare Professionals in Attendance field for general Participants registration.

Note: Make any random Healthcare Professionals between the interested persons. Also You can Change the publish button condition as your own too. (example: It could be 5 participant and 1 Healthcare Professional)

- The camp will be removed from the "Upcoming Camps" section.
- Update: Clicking this button will open a modal or form with pre-filled information for the selected upcoming camp.
 - Organizers can modify details such as Camp Name, Date and Time, Venue, Target Audience, etc.
 - After making changes, organizers can click the "Save" or "Update" button within the modal/form.
- Delete: Clicking this button will prompt a confirmation dialog to ensure the organizer's intent.
 - Upon confirmation, the upcoming camp will be deleted from the system.

Healthcare Professionals Routes (Private)

Accepted Camps:

Route Endpoint: /accepted-camps

List of Accepted Camps:

Display a comprehensive list of all upcoming camps for which the professional's interest has been accepted. Include details such as Camp Name, Date and Time, Venue, Target Audience, and Acceptance Status. Utilize a clear and organized tabular format for easy comprehension.

Actions:

Provide intuitive action buttons for each accepted camp row:

View Details:

- Allow professionals to view comprehensive information about the accepted camp.
- Implement a button or link leading to the detailed information page for the specific camp.

2. JWT for Sensitive Routes:

- Implement JSON Web Token (JWT) authentication for routes containing sensitive data.
- Explore JWT libraries for both the server and client-side to handle authentication.

3. Pagination for Tables :

- Add pagination functionality at the footer of tables to enhance user experience.
- Utilize a library or implement a custom solution to manage and display paginated data.

4. Tanstack Query Implementation:

- Implement Tanstack Query for efficient data fetching and state management.
- Utilize React **Tanstack** hooks to seamlessly manage and update data on the client-side.
- Explore the use of query caching and invalidation to optimize data retrieval.
- Enhance the overall performance of the web app by leveraging the capabilities of **Tanstack** Query.

Login System Requirements

Authentication and Authorization:

Implement a secure authentication and authorization system using Firebase Authentication to manage user access based on their roles (Organizers, Healthcare Professionals, and Participants).

Firebase Authentication:

1. User Registration:

- Participants, Healthcare Professionals, and Organizers should be able to register with the system.
- Use Firebase Authentication to securely handle user registration.
- Capture necessary user information during registration (e.g., email, password).

2. User Login:

- o Provide a login functionality for users to access their accounts.
- Leverage Firebase Authentication for secure user login.
- Authenticate users based on their email and password.

3. Role-Based Access:

- Implement role-based access control to ensure that users have access to features and data based on their roles.
- Assign roles to users during registration or account creation.

4. Logout:

Implement a secure logout mechanism using Firebase
 Authentication to terminate user sessions

Role-Based Access Control (RBAC):

1. Assign Roles during Registration:

- Assign roles (Organizer, Healthcare Professional, Participant) to users during the registration process.
- Store user roles in the database.

2. Role-Based Redirects:

- Redirect users to specific dashboard pages based on their roles after successful login.
- For example, after login, an Organizer should be redirected to the Organizer Dashboard, a Healthcare Professional to the Professional Dashboard, and a Participant to the Participant Dashboard.

Optional Tasks

- 1) Added animation on your website using (<u>framer motion</u>, or <u>AOS Animate</u> on scroll library
- 2) Implement Axios interceptor.
- 3) Implement an Admin Dashboard Where Admin can manage Organizer, Participant and Healthcare Professional.
- 4) Avoid incorporating Daisy UI; instead, opt for alternative UI component libraries like Material Tailwind, Flowbite, or Ant Design. You can choose any UI library, excluding Daisy UI.
- 5) In Contact Us Page implement a section of your website address with this react-leaflet
- 6) Add one extra feature of your own. This will help you in the future to differentiate your project from others.

What to Submit

- 1. Assignment Category: assignment12_category_0016
- 2. Front-end Live Site Link:
- 3. Client Side Github Repository Link:
- 4. Server Side Github Repository Link: