bworkzlive.com Documentation

Project Overview

bworkzlive.com is a cutting-edge platform designed for individuals who aspire to become certified instructors in disciplines such as Zumba. It offers training programs and certifications that enable users to gain the skills required to become professional instructors. The platform simplifies the journey from learning to certification, providing a seamless experience for both students and instructors.

Technology Stack

- **Frontend**: React.js
- **Backend**: Serverless Framework using AWS Lambda
- **Database**: AWS DynamoDB
- **Authentication**: AWS Cognito and Google Auth (email-password mechanism)
- **Payment Gateways**: Razorpay and PayPal

Backend Functionality

The backend for bworkzlive.com is developed using a serverless architecture, leveraging AWS Lambda functions to handle various processes. This design ensures scalability, reliability, and cost-efficiency.

Folder Structure

Functions Folder:

- Divided into three subfolders: `admin`, `any`, and `user`.
  - **Subfolders**:
    - `handlers`: Manages HTTP requests and routes.
    - `services`: Encapsulates core business logic.
    - `middleware`: Provides middleware logic for request handling.
    - `utils`: Contains utility/helper functions.

Deployed Lambda Functions

- **Attendance & Scheduling**:

  - `prodMonthlyAttendanceReport`: Generates monthly attendance reports.
  - `prodClassScheduleAutomation`: Automates class scheduling tasks.
- **Data Upload & Signup**:

  - `prodLeadsDataUpload`: Handles bulk lead data uploads.
  - `prodPreSignUp`: Manages pre-signup activities.
  - `prodGooglePreSignUp`: Handles Google-based pre-signup flows.
- **Authentication**:

  - `prodCreateAuthChallenge`: Creates authentication challenges for login.
  - `prodDefineAuthChallenge`: Defines the logic for authentication challenges.
  - `prodVerifyAuthChallengeResponse`: Verifies responses to authentication challenges.
- **File Management**:

  - `prodTimeTableUpload`: Uploads timetables for class schedules.

Endpoints

Admin Endpoints

- Manage profiles:

  - `GET /admin/profile-list/{cognitoId}`: Retrieve profiles.
  - `POST /admin/create-user`: Create a new user.
  - `PATCH /admin/update-user`: Update user details.
  - `PUT /admin/delete-user`: Delete a user.
- Certification management:

  - `POST /admin/create-certificate/{institution}`: Create a new certificate.
  - `PUT /admin/update-certificate/{institution}`: Update an existing certificate.
  - `PUT /admin/update-certified-status/{institution}`: Update certification status.
- Video management:

  - `POST /admin/upload-class-videos`: Upload new class videos.

- ○ `DELETE /admin/delete-class-videos/{institution}`: Delete class videos.

General (Any) Endpoints

- User checks:
  - ○ `POST /any/user-exists/{institution}`: Verify if a user exists.
  - ○ `GET /any/certifiedmemberget/{institution}`: Fetch certified members.
- Product and class data:
  - ○ `GET /any/products/{institution}`: Retrieve product details.
  - ○ `GET /any/instructor-list/{institution}`: Retrieve instructor details.
  - ○ `GET /any/get-current-class/{institution}`: Get details of the current class.

User Endpoints

- Schedule management:

  - ○ `GET /user/previous-schedule`: Fetch past schedules.
  - ○ `GET /user/upcoming-schedule`: Fetch future schedules.
- Profile and payment:

  - ○ `PUT /user/profile-create-or-update`: Create or update a profile.
  - ○ `PUT /user/payment-initiate`: Initiate payment.
  - ○ `PUT /user/payment-verify`: Verify payment.
  - ○ `PATCH /user/cancel-payment-razorpay`: Cancel a Razorpay payment.
  - ○ `PATCH /user/cancel-payment-paypal`: Cancel a PayPal payment.
- Video management:

  - ○ `GET /user/fetch-videos`: Retrieve a list of videos.
  - ○ `GET /user/get-playlist`: Retrieve playlists.

Authentication and Authorization

- **Mechanisms**:
  - ○ AWS Cognito for user authentication and management.
  - ○ Google Auth for email-password-based authentication.

Third-Party Integrations

- **Payment Gateways**:
  - ○ Razorpay and PayPal for subscription and payment processing.

Frontend Functionality

The frontend is designed to provide a user-friendly interface for both instructors and users. The platform ensures a smooth experience for accessing courses, managing subscriptions, and engaging with content.

Behavior and Flow

Initial API Calls

Upon visiting bworkzlive.com, the following API calls are triggered:

- `GET /any/products/Bworkz`: Retrieves a list of available products.
- `GET /any/instructor-list/Bworkz`: Fetches the list of instructors.

User Authentication

- **Signup/Login**:
  - `POST /any/user-exists/Bworkz`: Verifies whether the user already exists in the system.
  - `GET /user/profile/Bworkz`: Fetches user profile data upon login.

Subscription Management

If the user's status is inactive, they are redirected to the `/subscription` page to choose a subscription plan:

- Plans: Monthly or yearly.
- Payment methods: Razorpay or PayPal.

Upon successful payment:

- The **Payment Table** is updated with the transaction details like `cognitoId`, `paymentId`, `subscriptionId`, `amount`, `currency`, `active`, `productId`, `institution`, `subscriptionType`, `planId`, `paymentMode`, `paymentDate`, `renewDate`, `emailId`, `isVerified`.
- The **User Profile Table** is updated to mark the user as "Active," with the subscription ID, payment ID, product ID, and renewal date.

Post-Subscription Features

Once the user becomes active, the dashboard is unlocked. It includes:

- **Upcoming Classes**: Displays future classes with details (date, instructor, class type, time, and Zoom link).
- **Previous Classes**: Lists all completed classes.
- **Videos Section**: Provides access to videos and curated playlists.

Additional Pages

- `/aboutus`: Provides information about the platform.
- `/instructor`: Displays details of all instructors.

Admin Functionality

Admins have extensive control over the platform's operations, including user and content management.

Key Features

1. **Member Management**:

   - Create, update, and list all members.
   - Certify members and manage their certification status.
2. **Dashboard Management**:

   - Add upcoming classes.
   - Upload and manage videos.
   - Create and manage playlists.
3. **Video Management**:

   - Upload and delete videos in the video section.

Instructor Functionality

Instructors have access to functionalities related to managing classes and certifications.

Key Features

1. **Class Management**:

   - Add or update classes in the upcoming section.

○ Add videos to the previous classes section.
2. **Certifications**:
 ○ Certify members and manage their certification status.

Limitations

● Instructors cannot access the member list.
● They cannot upload videos to the platform.

Future Enhancements

1. creating an automated lambda function that will fetch the user transaction from razorpay and paypal and automatically add those data in the payment table and update the user profile table . it will make the user inActive and calculate the due amount as well by using the event bridge we will run the function every day.
2. UI enhancement of all the pages.
3. Enhancement of videos section in dashboard adding audio download option and ui enhancement.