DESIGN DOCUMENT

1. Architecture Overview

NattyWorld is built upon the MERN (MongoDB, Express.js, React.js, Node.js) stack:

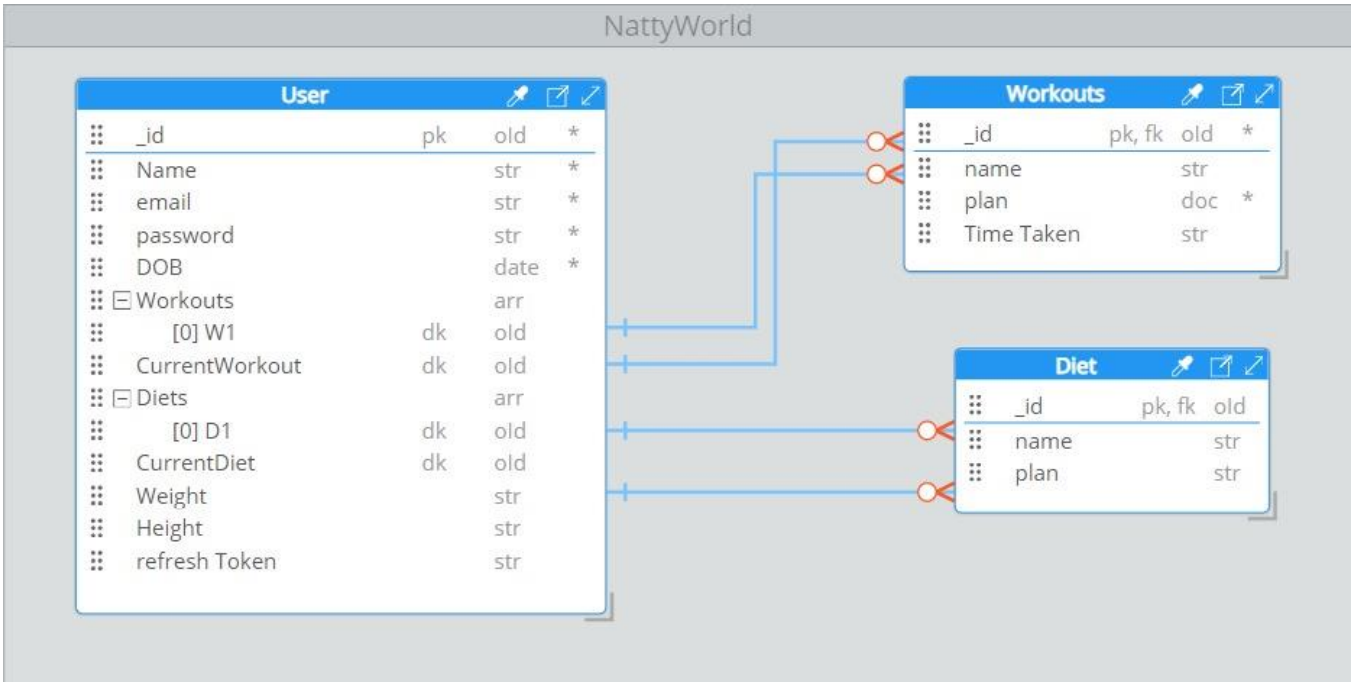
- **Frontend:** Developed with React.js, NattyWorld offers interfaces for users to engage with personalized diet plans, workout routines, BMI calculation, gym location services, and user authentication.
- **Backend:** Powered by Express.js and Node.js, the backend provides RESTful APIs for user authentication, diet planning, workout planning, BMI calculation, gym location services, and AI integration for generating personalized plans.
- **Database:** MongoDB serves as the database to store user data, diet plans, workout plans, gyms, and related information.

2. Database Design

NattyWorld's MongoDB database consists of the following collections:

- **Users:** Stores user information, including name, email, password hash, and preferences.
- **Diet Plans:** Contains personalized diet plans generated for users, comprising meal schedules, nutritional information, and recipes.
- **Workout Plans:** Stores personalized workout plans generated for users, detailing exercises, sets, reps, and rest intervals.
- **Gyms:** Contains information about nearby gyms, such as name, address, contact information, and facilities.

3. Schema Modals



4. API Design

NattyWorld backend offers RESTful APIs for various functionalities:

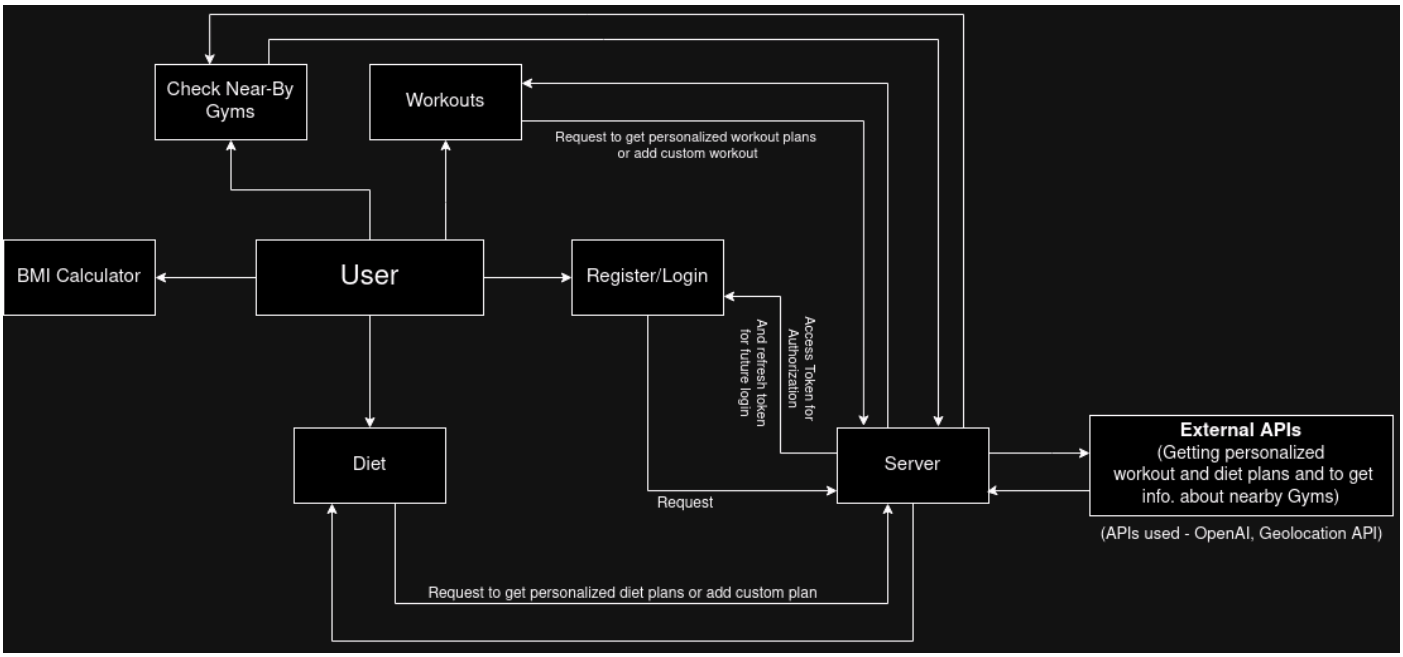
- **Authentication APIs:**
    - /api/auth/register: Register a new user.
    - /api/auth/login: Authenticate user credentials and generate a JWT token.
    - /api/auth/user: Retrieve current user details.
- **Diet Planner APIs:**
    - /api/diet/plans: Generate personalized diet plans for users.
    - /api/diet/plans/:id: Retrieve, update, or delete diet plans by ID.
- **Workout Planner APIs:**
    - /api/workout/plans: Generate personalized workout plans for users.
    - /api/workout/plans/:id: Retrieve, update, or delete workout plans by ID.
- **BMI Calculator APIs:**
    - /api/bmi/calculate: Calculate BMI based on user input.
- **Gym Locator APIs:**
    - /api/gyms: Fetch nearby gyms based on user's location.
    - /api/gyms/:id: Retrieve details of a specific gym by ID.

5. Frontend Design

NattyWorld's frontend, developed using React.js and styled with CSS, encompasses various components:

- **Homepage:** Introduces the website and its features.
- **User Authentication:** Facilitates sign up, login, and logout functionalities.
- **Dashboard:** Personalized dashboard exhibiting diet plans, workout plans, progress, etc.
- **Diet Planner:** Allows users to input dietary preferences, view personalized diet plans, and access recipes.
- **Workout Planner:** Enables users to specify fitness goals, view personalized workout plans, and access exercise demonstrations.
- **BMI Calculator:** Calculates BMI based on user input and presents the results.
- **Gym Locator:** Utilizes Google Maps integration to find nearby gyms.

6. Data Flow Diagram



7. Deployment

NattyWorld can be deployed on cloud platforms like Heroku, AWS, or DigitalOcean. Frontend hosting services such as Netlify or Vercel, coupled with MongoDB Atlas for database hosting, offer a robust deployment solution.

8. Testing

Unit tests for frontend components using Jest and backend APIs using tools like Mocha/Chai ensure robustness. Additionally, integration tests with Cypress cover end-to-end scenarios.

9. Scalability and Performance

NattyWorld is designed for scalability, employing caching mechanisms, load balancing, and horizontal scaling strategies. Performance optimizations guarantee fast response times and efficient resource utilization.

10. Security

NattyWorld prioritizes security by encrypting sensitive data, implementing input validation, safeguarding against common web vulnerabilities, and utilizing secure authentication mechanisms like JWT tokens.

11. Accessibility

Adhering to web accessibility standards ensures usability for all users. Features such as alternative text for images, keyboard navigation support, and semantic HTML markup enhance accessibility.

12. Maintenance and Support

Regular maintenance, updates, and a robust support system address user inquiries, bug reports, and feature requests promptly, ensuring a seamless user experience.

13. Conclusion

NattyWorld endeavors to provide a comprehensive fitness platform, empowering users to achieve their fitness goals effectively. By embracing the outlined design principles and implementing suggested features, NattyWorld aims to foster a healthy lifestyle journey for its users.