Maintenance Guide for "Miracle of Luck"

1. About:

"Miracle of Luck" is a Vite stack application with multiple layers, utilizing various technologies:

1. **Front End:** We built the web application using React and deployed it on Vercel, accessible at https://miracle-of-luck.vercel.app.
2. **Back End:** We created the server using Node.js with the Express.js framework and hosted it on Render, accessible at https://miracle-of-luck.onrender.com, with a MongoDB database on Atlas.
3. **Authentication:** We manage user authentication using Auth0, a secure identity platform that supports social login (e.g., Google, Facebook), email/password authentication, and passwordless login. It ensures secure access to the application through token-based authentication and supports features like multifactor authentication (MFA).
4. **Responsive Design:** We designed the website to adapt to various screen sizes, such as phones, tablets, and computers, using a CSS framework.

Front-End Structure:

We developed the front end using React, with the following technologies and tools:

- **CSS and Material-UI:** We use these for styling and ensuring a consistent look and feel across the application.
- **Form Validation:** We handle form validation with React Hook Form to validate user input.
- **Auth0:** We integrated Auth0 for secure user authentication.
- **API:** We handle API requests and interactions with the backend, providing a centralized location for managing all server communications.

Back-End Structure:

We follow the MVC (Model-View-Controller) pattern in the back end:

- **`index.js`:** We use this as the main server entry point to handle HTTP requests and connect to MongoDB with Express.js.
- **Controllers:** We store functions that handle requests and return responses in the Controllers.
- **Routes:** We define the routes for server-side logic.
- **Prisma:** We use Prisma as an ORM (Object-Relational Mapping) tool to interact with the database in a type-safe and efficient way, providing a powerful data modeling and querying layer on top of MongoDB.
- **Config:** We maintain configuration files, such as `prismaconfig` for setting up and managing Prisma connections and `auth0config` for configuring Auth0 authentication.

Additional Aspects:

Database Management:

- **Backups and Restoration:** We ensure regular backups of the MongoDB database on Atlas and have a plan in place for data restoration in case of data loss or corruption.
- **Environment Variables:** We ensure that all necessary environment variables (e.g., database URIs, Auth0 keys) are securely set in the deployment environments.

Security:

- **Security Practices:** We follow security best practices, such as using HTTPS, implementing content security policies, sanitizing user inputs, and securing cookies.
- **Auth0 Security Features:** We utilize Auth0's security features, such as anomaly detection, brute-force protection, and token expiration.

Dependencies Management:

- **Dependency Updates:** We regularly update dependencies to patch vulnerabilities and ensure compatibility with the latest versions.

Documentation and Comments:

- We maintain good code documentation and comments, especially for complex logic or algorithms used in the project.

Running the Project

To run and update both the front-end and back-end parts of the project:

1. **Install Prerequisites:**
   - We use **Visual Studio Code (VSC)**, which can be downloaded from the Visual Studio Code website.
   - We use **Node.js and npm**, which can be downloaded from the Node.js official website. We ensure both are added to your system's PATH.
2. **Clone the Repository:**
   - We download or clone the project repository from your version control system (e.g., GitHub).
3. **Install Dependencies:**
   - **Server:** We navigate to the server directory (`cd server`) and run `npm install` to install server-side dependencies.
   - **Client:** We navigate to the client directory (`cd client`) and run `npm install` to install client-side dependencies.
4. **Start the Server:**
   - We open a terminal, navigate to the server directory, and run the server using the command `npm run start`.
5. **Start the Client:**

    - We open a new terminal window or tab, navigate to the client directory, and run the client using the command `npm run dev`.
6. **Access the Application:**
    - We open a web browser and navigate to https://miracle-of-luck.vercel.app.

Deploying the Application:

- **Server Deployment:** We have deployed the backend server on Render, and it can be accessed at https://miracle-of-luck.onrender.com.
- **Client Deployment:** We have deployed the frontend client on Vercel, and it is available at https://miracle-of-luck.vercel.app.

To deploy updates, we push changes to the respective repositories linked to Render and Vercel. Both platforms automatically handle the build and deployment process.

By following this guide, you can efficiently maintain, operate, and deploy your "Miracle of Luck" project. If you need further details or any specific customizations, feel free to reach out!