MOHAMED HAFLAN - panurgic

Challenges Faced and Solutions Implemented

1. **Frontend and Backend Communication Issue**
    - **Challenge:** The React frontend was running on port 5173, while the Spring Boot backend was running on port 8081. Due to this, requests from the frontend were not reaching the backend because of Cross-Origin Resource Sharing (CORS) restrictions.
    - **Solution:** This issue was resolved by adding a CORS configuration in the `SecurityConfig` file of the Spring Boot backend. This allowed the frontend to successfully communicate with the backend.

2. **Restricting Certain Parts of the Application to Admin Users**
    - **Challenge:** Some parts of the application were meant to be accessible only to admin users, but there was no clear way to enforce this restriction in the frontend.
    - **Solution:** The user's role was stored in a global context in React. This made it possible to access and check the role from anywhere in the frontend, ensuring that only authorized users could access restricted areas.

3. **Backend Access Control Based on User Role**
    - **Challenge:** Even though the frontend restricted access based on roles, there was still a need to enforce role-based access control at the backend to prevent unauthorized access.
    - **Solution:** The user's role was included in the JWT token, which was stored in a secure HTTP-only cookie. Upon each request, the backend extracted the role from the token and applied the necessary access control logic to restrict endpoints accordingly.

4. **Validating Each Request to the Backend**
    - **Challenge:** It was necessary to ensure that every request reaching the backend was valid and secure, checking for token presence and validity before proceeding.
    - **Solution:** This was accomplished by extending the `OncePerRequestFilter` class in Spring Security. This filter checked the availability and validity of the JWT token in every request before allowing it to be processed by the controllers, ensuring secure request handling.

Enhancements for Future Improvements

1. **Email Notification Service**
    - Implement an email service to send automatic confirmation emails to users when they book a service or perform specific actions within the application.
2. **Improved UI/UX Design**
    - Enhance the overall design of the application to make it more visually appealing and user-friendly. This includes refining the color scheme, typography, responsiveness, and animations.
3. **User Profile Management**
    - Add a profile management section where users can update their personal information, upload profile pictures, and manage other account details.

By implementing these enhancements, the application will offer a better user experience and improved functionality while maintaining security and access control.