

# Project Report for DCS-Assignment-3

## Enhancements

### Frontend

1. **CRUD Operations:** Implemented Create, Read, Update, and Delete functionalities for managing family data.
2. **Form Handling:** Improved form handling for editing and adding family members dynamically.
3. **API Integration:** Successfully integrated frontend with backend API for real-time data interaction.

### Backend

1. **Node.js & Express:** Established a server using Node.js and Express.
2. **MongoDB Integration:** Utilized Mongoose for efficient database operations with MongoDB.
3. **API Endpoints:** Developed RESTful API endpoints for comprehensive family management.

### Deployment

**Render:** Deployed the application on Render for seamless continuous deployment.

## Challenges Faced and Solutions

### Challenge 1: State Management in React

**Problem:** Managing complex state for family members and handling dynamic form fields.

**Solution:** Utilized React hooks (`useState`, `useEffect`) to manage state efficiently. Ensured that changes to form fields were correctly reflected in the state, allowing for smooth addition and removal of family members.

### Challenge 2: Dynamic Form Fields

**Problem:** Handling dynamic form fields for family members.

**Solution:** Created functions to dynamically add and remove form fields. Managed the state of these fields using arrays, which allowed for easy updates and maintenance.

### Challenge 3: Asynchronous API Calls

**Problem:** Ensuring synchronous state updates with asynchronous API calls.

**Solution:** Used `async/await` for API calls and ensured state updates were performed after data was successfully fetched or modified, preventing inconsistencies.

### Challenge 4: Deployment Issues

**Problem:** Initial deployment on Render faced configuration issues.

**Solution:** Thoroughly reviewed the deployment settings and environment variables. Corrected the configuration and ensured that the application was correctly connected to the MongoDB instance.

## Conclusion

The project successfully implements a full-stack web application with robust CRUD functionalities, dynamic form handling, and seamless integration between frontend and backend. Despite facing several challenges, they were effectively overcome, resulting in a well-functioning and user-friendly application.

## Design

The design for this project was created using [Figma](#).

For more details, visit the [GitHub repository](#).