Technical Handover Notes - Dama Finance

Project Overview

This project consists of a **React.js frontend** (using TypeScript) and an **ASP.NET Core Web API backend** with a PostgreSQL database. Authentication and authorization are handled by a separate **Identity Server 4** instance. The frontend, backend API, and identity provider are distinct solutions and are hosted on GitHub.

Technology Stack

Frontend

- **Framework:** React.js (v17.0.2)
- **Language:** TypeScript
- **State Management:** Redux Toolkit
- **UI Library:** Ant Design (AntD)
- **Form Handling:** Formik
- **Node.js Version:** 20.9.0

Backend

- **Framework:** ASP.NET Core Web API
- **Database:** PostgreSQL
- **Authentication & Authorization:** Identity Server 4
- **ORM:** Entity Framework Core (EF Core)
- **Repository Pattern:** Generic Repository Pattern
- **CQRS & Mediator Pattern:** Uses MediatR for implementing the CQRS pattern.
- **Hosting:** Linux-based cloud hosting

Repository Structure

Frontend Repository

```
frontend/
├── src/
│   ├── components/          # Reusable UI components
│   ├── constants/           # App constants, API, Identity Provider etc
│   ├── contexts/            # Context API Logic
│   ├── hooks/               # Custom React hooks
│   ├── layout/              # App Layout
│   ├── models/              # Data shape models
│   ├── pages/               # Page components
│   ├── redux/               # Redux slices anad store configuration
│   ├── utils/               # Helper functions
│   ├── services/            # API request handling using Axios
│   ├── scss/                # SCSS modules (BEM naming convention used)
│   ├── translations/        # Translation module
│   ├── app.component.tsx    # Main App component
│   ├── index.tsx            # Application entry point
│
├── public/
├── .env                     # Environment variables
├── package.json             # Project dependencies and scripts
```

```
├──    tsconfig.json                # TypeScript configuration
├──    README.md                    # Project documentation
```

Hosting & Domain Information

- Online services were purchased using my personal email.
- The domain name transfer needs to be initiated with the preferred provider. The transfer process usually takes about **five days** to complete, depending on the hosting provider.
- The current hosting service is tied to my **personal email**.
- There are **no upfront or outstanding payments** on this account, so the new team can determine their own hosting preferences.

Environment Setup

Frontend Setup

1. Clone the repository:
2. `git clone` https://github.com/dama-finance-hiv-free/budget-monitoring-app.git
3. `cd frontend`
4. Install dependencies:
5. `npm install`
6. Set up api and Idenitity Provider variables in `constants folder`:
7. Start the development server:
8. `npm start`

Backend Setup

1. Clone the repository:
2. `git clone https://github.com/dama-finance-hiv-free/budget-monitoring-api.git`
3. `cd backend-api`
4. Configure the database connection in `appsettings.json`:

```json
{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft": "Warning",
      "Microsoft.Hosting.Lifetime": "Information"
    }
  },
  "AllowedOrigins": "http://localhost:3000",
  "ConnectionStrings": {
    "MonitoringData": "User ID=userid;
                      Password=******;
                      Server=localhost;Port=5432;
                      Database=dbname;
                      Integrated Security=true;
                      Pooling=true;",
  },
  "AppSettings": {
    "AuthorizationServer": "https://localhost:5001",
    "EnableDiagnostics": "true"
```

5. Apply migrations:

```
6. dotnet ef database update
```

7. Run the API:

```
8. dotnet run
```

Identity Server Setup

1. Clone the repository:

```
2. git clone https://github.com/dama-finance-hiv-free/budget-monitoring-identity-
   provider.git
```

```
3. cd identity-server
```

4. Configure the `appsettings.json`:

5. Run the Identity Server:

```
6. dotnet run
```

Common Issues & Troubleshooting

CORS Errors

- If the frontend cannot communicate with the backend, update CORS settings in `Startup.cs`:
- `services.AddCors(options =>`
- `{`
- `    options.AddPolicy("AllowSpecificOrigins",`
- `    builder =>`
- `    {`
- `        builder.WithOrigins("<frontend-url>")`
- `                .AllowAnyMethod()`
- `                .AllowAnyHeader()`
- `                .AllowCredentials();`
- `    });`
- `});`

Database Connection Issues

- Ensure PostgreSQL is running and the connection string is correct.
- Run migrations if necessary:
- `dotnet ef database update`

Identity Server Not Issuing Tokens

- Check `appsettings.json` to ensure valid client configurations.

- Ensure Identity Server is running and accessible from the frontend.
- Ensure you correctly link to the wildcard.dama.pfx file for signing credentials

Github Repositories

Organization: https://github.com/dama-finance-hiv-free

Identity Provider: https://github.com/dama-finance-hiv-free/budget-monitoring-identity-provider

API: https://github.com/dama-finance-hiv-free/budget-monitoring-api

React Frontend: https://github.com/dama-finance-hiv-free/budget-monitoring-app

Data & Notes: https://github.com/dama-finance-hiv-free/data-notes.git