Separation Agreement Wizard - Project Report (Frontend)

Project Overview

The Separation Agreement Wizard is an Angular-based web application designed to assist users in creating customized separation agreements. This wizard-style application guides users through a step-by-step process to collect information about the separating parties, select appropriate legal clauses, and generate a legally formatted separation agreement document.

Technical Stack

- **Frontend Framework**: Angular (latest version with standalone components)
- **UI Component Library**: PrimeNG - providing rich UI components including:
- Tree component for hierarchical clause selection
- Tables for data display
- Dropdown menus
- Calendar for date selection
- Form controls
- Buttons and other UI elements
- **Styling**:
- SCSS for component styling
- Responsive design using grid layout
- Animations using animate.css library
- **Routing**: Angular Router for navigation between wizard steps
- **State Management**: Service-based state management with RxJS Observables
- **Data Persistence**: Session storage for maintaining wizard state between page refreshes

Application Architecture

Component Structure

The application follows a modular architecture with these key components:

1. **App Component**: Root component providing the base layout with header, content area, and footer
2. **Party Info Component**: Collects information about both parties, their roles, support arrangements, important dates, and children
3. **Clause Selection Component**: Allows users to select legal clauses to include in the agreement from a categorized tree structure
4. **Review Document Component**: Displays a preview of the collected information and selected clauses before generating the final document

Services

The application uses a central service-based approach for data management:

- **WizardService**: Core service that:
- Manages party information and selected clauses
- Handles data persistence to session storage
- Loads clauses from a JSON file
- Communicates with backend API for document generation
- Provides methods for adding/removing/ordering clauses

Data Models

Well-defined TypeScript interfaces are used for data consistency:

1. **PartyInfo**: Contains information about the separating parties including:

- Personal details (names)
- Relationship roles (Mother, Father, Wife, Husband)
- Support arrangements (payor/recipient)
- Important dates (marriage, cohabitation, separation)
- Children information

2. **Clause**: Represents legal clauses with:

- ID
- Category

- Text content
- Selection status
- Label

Workflow

The application implements a guided workflow:

1. **Party Information Entry**:

- Users enter details for both parties
- Record support arrangements (spousal/child)
- Enter important relationship dates
- Add information about children (if applicable)

2. **Clause Selection**:

- Clauses are organized in a hierarchical tree by category
- Users can select individual clauses or entire categories
- Selected clauses are displayed in a separate panel
- Search functionality allows finding specific clauses

3. **Document Review**:

- Preview all entered information
- Review selected clauses with text personalized with party names
- Navigate back to previous steps for corrections

4. **Document Generation**:

- Send compiled data to backend API
- Generate and download final document in DOCX format

Key Features

1. **Multi-step Wizard Interface**:

- Intuitive navigation between steps
- Persistent data storage between steps

- Form validation to ensure required information is provided

2. **Smart Clause Management**:

- Hierarchical organization of clauses by legal categories
- Ability to select/deselect individual or grouped clauses
- Automatic ordering of clauses by predefined legal category sequence

3. **Dynamic Form Handling**:

- Support for adding multiple children with individual details
- Role selection for parties
- Support arrangement configuration

4. **Document Personalization**:

- Automatic replacement of placeholders with actual party names
- Context-aware display of relevant sections
- Formatted according to legal document standards

5. **Responsive Design**:

- Mobile-friendly interface
- Adaptive layout for different screen sizes
- Touch-friendly controls for mobile users

6. **User Experience Enhancements**:

- Visual feedback during loading and processing
- Form validation with error messages
- Animations for smoother transitions
- Modern UI with consistent styling

Integration Points

The application integrates with a backend service through HTTP calls:

- Fetches clause data from a JSON file (assets/clauses.json)
- Sends compiled agreement data to a .NET backend API for document generation

- Downloads the generated document as a DOCX file

Technical Implementation Details

1. **State Management**:

- Uses RxJS BehaviorSubjects for reactive state management
- Automatically persists state to session storage
- Implements proper data transformation and ordering

2. **Form Handling**:

- Reactive forms with validation
- Dynamic form arrays for children information
- Custom validation logic

3. **Tree Component Implementation**:

- Hierarchical display of clauses by category
- Custom selection logic for parent-child relationships
- Search and filtering capabilities

4. **Document Generation**:

- Assembles selected clauses in proper legal order
- Replaces placeholders with actual party information
- Handles downloading of generated document

Conclusion

The Separation Agreement Wizard is a sophisticated web application that streamlines the complex process of creating legal separation agreements. It combines a user-friendly interface with robust functionality to guide users through collecting necessary information and selecting appropriate legal clauses. The application's modular architecture, responsive design, and thoughtful UX make it accessible to users without legal expertise while producing professionally formatted legal documents.The codebase demonstrates good Angular practices including standalone components, service-based state management, and reactive programming principles. The use of PrimeNG components provides a polished, professional UI while custom styling creates a cohesive brand experience.

Separation Agreement Document Generation API Project Report (Backend)

Project Overview

This project is a backend API service built with ASP.NET Core that dynamically generates separation agreement legal documents in Microsoft Word format.
The system takes structured data about separating parties and their agreement terms, then produces professionally formatted legal documents that are ready for review and signing.

Technology Stack

- **Framework**: ASP.NET Core
- **Document Generation**: OpenXML (DocumentFormat.OpenXml)
- **API Documentation**: Swagger/OpenAPI
- **Frontend Integration**: CORS-enabled for Angular client application

Core Components

Models

1. **DocumentRequest**: The main data container for generating a document
   - Contains party information and selected legal clauses
2. **PartyInfo**: Stores information about both parties in the separation
   - Names of both parties (first, middle, last)
   - Information about support arrangements (spousal and child support)
   - Key dates (marriage, cohabitation, separation)
   - Child information
3. **Clause**: Represents legal clauses to be included in the document
   - Organized by category
   - Contains the legal text with placeholders for party information
   - Selection status

Services

1. **DocumentGenerationService**: Core service that produces the document

- Generates a Word document (.docx) using OpenXML
- Formats document with professional styling (headers, spacing, fonts)
- Creates structured sections with proper legal formatting
- Dynamically replaces placeholders with actual party information
- Handles complex document features:
- Different first page headers
- Page numbering
- Signature lines
- Custom formatting for schedules and clauses
- Page breaks where needed

Controllers

1. **DocumentController**: Exposes API endpoints

- POST endpoint for document generation (/api/Document/generate)
- Returns generated document as a downloadable file
- Includes error handling and appropriate HTTP status codes

Program Configuration

- CORS configuration to allow requests from specific origins:
- Local development (http://localhost:4200)
- Production (https://sepration-agreement-wizard.vercel.app)
- Service registration for dependency injection
- Swagger configuration for API documentation
- Standard ASP.NET Core middleware configuration

Document Generation Features

The service produces professionally formatted legal documents with:

- Consistent styling using Calibri font and appropriate spacing
- Professional header with page numbering
- Formatted party information and legal clauses
- Proper indentation and formatting for legal text
- Dynamic replacement of placeholders with party information
- Signature lines for both parties
- Formatted schedules with appropriate page breaks
- Footer with generation timestamp

Integration Points

The API is designed to work with an Angular frontend application. The frontend would:

1. Collect party information
2. Allow selection of appropriate legal clauses
3. Send structured data to the API
4. Receive and present the generated document to the user

Deployment Considerations

The application is configured for both development and production environments with appropriate security measures including CORS restrictions to specific origins.

Security and Compliance

The application handles sensitive legal information, so proper hosting with appropriate security measures is essential. No user authentication is visible in the current codebase, suggesting this might be handled by the frontend or a separate service.

Conclusion

This backend service is a specialized document generation API for creating separation agreements. It demonstrates a clean separation of concerns with models, controllers, and services, and leverages the OpenXML library to produce professional legal documents based on user input from a frontend application.