**Final Setup and Deployment**

**Prerequisites**

Before deploying the application, ensure you have the following:

1. Visual Studio 2022 (any edition)
2. .NET 8.0 SDK installed
3. SQL Server 2022 (or SQL Server Express) installed ( SQL Database)
4. Basic knowledge of C# and ASP.NET Core MVC

**Local Development Setup**

**Setting Up the Environment**

1. Clone or download the repository to your local machine
2. Create an SQL Database.
3. Open the solution file (AgriEnergyConnect.sln) in Visual Studio 2022
4. Restore NuGet packages (Visual Studio should do this automatically)
5. Connect your database to SQL Server 2022 and copy the connection string
6. Paste your connection string in appsettings.json and update it to match your SQL Server instance

**Creating the Database**

1. Open Package Manager Console (Tools > NuGet Package Manager > Package Manager Console)
2. Run the following command to create the database:

Add-Migration [MigrationName]

Update-Database

1. This will create the database and all required tables based on your models

**Running the Application**

1. Press F5 or click the "IIS Express" button to run the application
2. The application should launch in your default web browser
3. Register new users to test different roles (Farmer and Employee)

**Maintaining the Application**

**Adding New Features**

1. Always create a backup or use source control before making significant changes
2. Add new models as needed
3. Update the database using Entity Framework migrations:
4. Add-Migration [MigrationName]

Update-Database

1. Test thoroughly before deploying updates to production

**Securing the Application**

1. Consider implementing additional security measures:
   * HTTPS enforcement
   * Password complexity requirements
   * Failed login attempt tracking
   * Cross-Site Request Forgery (CSRF) protection (already included in ASP.NET Core)
   * Regular security updates
2. Update the authorization policies as needed to secure new features or controllers

**Backing Up the Database**

1. Set up regular SQL Server database backups
2. Store backups in a secure location, preferably off-site
3. Test database restoration periodically to ensure backups are working correctly