

Dokumentation av Projektet (ASP.NET Core MVC)

Syfte

Projektet är ett webbapplikationssystem som hanterar projekt och användare med olika roller. Användare kan skapa, redigera, ta bort och visa projekt beroende på behörighet. Systemet stödjer även inloggning, registrering, extern autentisering, val av dark mode och profilhantering.

Arkitektur

PresentationLayer

Innehåller alla controllers som hanterar användargränssnitt och sidnavigering. Här finns även alla ViewModels som binder data mellan formulär och applikationen.

ApplicationLayer (Service Layer)

Innehåller affärslogiken, t.ex. vem som får skapa projekt, hämta rätt användare, kontrollera roller och hantera datalogik. All affärslogik ligger i olika tjänster, t.ex. ProjectService och UserService.

Cross-cutting Concerns

Här finns alla Data Transfer Objects (DTOs) som används för att flytta data mellan lager. Även gemensamma gränssnitt och hjälpklasser som hanterar t.ex. användarstatus, dark mode och formulärdata finns här.

InfrastructureLayer

Innehåller implementationen av alla repositories som kommunicerar med databasen. Detta lager är kopplat till domänmodellerna och ansvarar för att hämta och spara data.

DataAccessLayer

Här finns konfigurationer och DbContext som mappar databasen. Den använder Entity Framework Core för att hantera tabeller för projekt, kunder och teammedlemmar.

SecurityLayer

Hantera autentisering, externa login-leverantörer och claims. Använder Microsoft Identity för användarhantering och inloggning.

Roller och behörighet

Roll	Skapa/Redigera	Visa Projekt	Ta bort
Admin	Ja	Alla	Ja
Manager	Ja	Alla	Ja
TeamMember	Endast egna	Egna	Nej
Customer	Nej	Egna	Nej
User	Nej	Nej	Nej

Flöde: Skapa projekt

1. Användaren fyller i formulär med projektnamn, kundens e-post, datum, budget, beskrivning och medlemmar.
2. Systemet sparar eventuell bild och skapar ett dataobjekt.
3. Affärslogiken validerar inmatningen och kopplar rätt kund och medlemmar.
4. Projektet sparas till databasen via ett repository.

Flöde: Inloggning och registrering

Användaren loggar in med e-post och lösenord. Om uppgifterna är korrekta, sätts användarens session, roll, dark mode-inställning och andra claims. Vid registrering skapas användaren med rollen "User" och positionen "New Member".

Flöde: Extern inloggning

Användaren kan logga in via externa tjänster som Google eller GitHub. Om användaren inte finns skapas ett nytt konto automatiskt med rollen "User". Profilbilden kan hämtas automatiskt från den externa tjänsten.

Designbeslut

- Projektbeskrivning stöder HTML och lagras som formaterad text.
- Bilder laddas upp och lagras under wwwroot-mappen.
- Användargränssnittet tillåter sökning och val av medlemmar via ett taggsystem.
- Dark mode sparas per användare och aktiveras via claim.

Säkerhet

- Alla sidor är skyddade med auktorisering.
- Roller styr vilken controller eller funktion man får använda.
- Systemet kontrollerar att endast behöriga användare kan utföra vissa åtgärder, t.ex. ta bort projekt eller redigera andra användare.

Slutsats

Systemet är designat med tydlig lagerindelning för att följa principen separation of concerns. Det är skalbart, säkert och har stöd för både intern och extern användarhantering. Rollbaserad åtkomstkontroll gör systemet flexibelt för olika typer av användare.

Vidareutveckling

Om projektet skulle vidareutvecklas finns flera naturliga förbättringar att överväga:

- **E-postbekräftelse och glömt lösenord:** Genom att implementera e-postverifiering kan man möjliggöra återställning av lösenord via länk.
- **"Kom ihåg mig"-funktion:** Förbättrad hantering av användarsessioner med längre inloggningstid.
- **Global sökfunktion:** En sida som visar sökresultat över både projekt och användare i ett helskärmsläge med klickbara kort.
- **Notifieringar:** Skapa stöd för meddelanden vid nya projekt, ändringar eller när användare läggs till.

- **Adminsida för systemstatus:** En vy för admin att se statistik och manuellt ändra statusar, roller eller information.

Källor

Projektet har byggts med stöd från följande resurser:

- **ChatGPT 4o** – för idébearbetning, felsökning, och strukturella förslag.
- **Builder.io** – för layoutstruktur och gränssnittsinspiration.
- **Microsoft Docs** – för referens till ASP.NET Core MVC, Identity och Entity Framework.
- **Mozilla Developer Network (MDN)** – för JavaScript, HTML och CSS-beteenden i webbläsare.