



Bakgrund

IT-gurun och superentreprenören Marcus Friberg har börjat spendera frukten av sina framgångar på exklusiva bilar. Hur mycket sonen Malte än gillar pappas nya hobby så är resten av familjen (sambon Maria och dottern Malin) inte lika entusiastiska över att gården fylls med bilar. Som den kreativa problemlösare han är har Marcus nu registrerat bolaget Friberg Car Rentals AB för att legitimera sitt beteende.

Marcus har kontaktat din arbetsgivare, Friberg IT-Solutions för att beställa en webbapplikation för det nya bolagets räkning. Eftersom hela upplägget är en fasad för att legitimera Marcus nya hobby finns egentligen inte någon ambition om att hyra ut några bilar, men applikationen behöver kunna hantera det. Eftersom Marcus har god insyn i Friberg IT-Solutions har du fått instruktioner av din närmaste chef att endast leverera lösningar på det som framgår av beställningen. God arkitektur bör tillämpas och standarder följas men features som inte nämns får du inte lägga tid på.

Beställning av webbapplikation

Tekniska krav

Produkten ska

- utvecklas utifrån ett projekt av typen ASP.NET Core Web App (Model-View-Controller)
- använda en SQL Server databas
- inte innehålla några features som inte är nödvändiga för att tillgodose de user stories som presenteras nedan (ex. loggning, formvalidering m.m.)
- inte använda några färdiga CSS/JS/HTML-bibliotek så som bootstrap, detta ska utvecklas från grunden och fungera bra i webbläsare på datorn, eventuell mobilanpassning beställs senare om det ens blir aktuellt.

Sidan bör se snygg och seriös ut, logotyp för företaget finns med i det här dokumentet och den ska användas. Det finns inte budget för någon funktion för bilduppladdning, men Marcus har bilder på alla bilar online så varje bil som läggs till kan ha en eller flera strängar med en länkar till bilden/bilderna.

Sidan bör byggas med god struktur, bilar, kunder och beställningar bör utgöras av Model-klasser, Repository bör användas så att så mycket som möjligt går att återanvända vid en eventuell ytterligare beställning.

Användarkrav – User Stories

Följande user stories har tagits fram och tydliggör vad kunder ska kunna göra och vad administratören ska kunna göra.

Som kund vill jag kunna

- lista alla bilar som finns att hyra
- se en eller flera bilder för bilen jag funderar på att hyra
- skapa en bokning för en viss bil med ett startdatum och ett slutdatum
- få en bekräftelse på bokningen
- se mina tidigare bokningar
- ta bort en kommande bokning.

Som administratör vill jag kunna

- lista, lägga till, ändra och ta bort bilar
- lista, lägga till, ändra och ta bort kunder
- lista och ta bort beställningar.

Övrig information från beställaren

Kunder och Administratörer bör vara olika models, inloggning sker med e-post och ett lösenord, lösenordet kan sparas i klartext i databasen, jag vill inte ha någon dyr säkerhet som Oauth2 eller liknande. Om en besökare listar bilar och klickar på "Boka" för en bil ska sidan visa två kolumner, i en av kolumnerna ska kunden kunna logga in och i den andra registrera sig som ny kund. Sedan ska bokningsprocessen fortgå. Administratörerna ska kunna logga in genom att besöka adressen med

tillägget /admin/, det bör alltså finnas en egen controller för admins där index metoden visar inloggningsformulär för administratörer.

Out of character (info från chefen)

I övrigt vill jag som din närmaste chef bara påminna om syftet med den här sidan. Marcus hoppas helt ärligt att ingen vill hyra hans bilar men han vill kunna legitimera mina inköp (eller som sambon och dottern kallar det, hans vansinne) och visa upp en fungerande om än ej blomstrande verksamhet. Jag förväntar mig att ni håller er till kravspecifikationen och löser varje krav med minsta möjliga nedlagda tid. Det blir inga plus i kanten för extra lister och krom som Marcus inte beställt 😊

Vill också tydliggöra att både tekniska och användar/admin krav kan komma att ändras under projektets gång.

Lycka till och happy hacking 😊