# Technische Onderbouwing

**Keuzes**

* Server-side
* .Net
* C#
* MVC
* Microsoft SQL
* LINQ
* Bootstrap
* xUnit
* AAA-pattern
* Mocking

**Onderbouwing**

**Server-side**: We hebben gekozen voor server-side ontwikkeling omdat het de meest geschikte aanpak is voor het bouwen van een webapplicatie zoals die van CM.com. Deze aanpak stelt ons in staat om complexe logica, gegevensbeheer en veiligheidsmaatregelen op de server uit te voeren, waardoor we de gebruikerservaring kunnen optimaliseren en de vereiste prestaties en beveiliging kunnen bieden.

**.NET**: We hebben gekozen voor .NET omdat het een bewezen technologie is die een breed scala aan tools en libraries biedt om snel en efficiënt krachtige server-side webapplicaties te ontwikkelen. Onze eerdere ervaring met .NET stelt ons in staat om snel aan de slag te gaan en te profiteren van de voordelen die het platform biedt. Bovendien is .NET een aantrekkelijke keuze voor onze toekomstige carrièremogelijkheden, omdat er een grote vraag is naar .NET-ontwikkelaars met concurrerende salarissen.

**C#**: We hebben C# gekozen als onze programmeertaal vanwege zijn krachtige mogelijkheden en naadloze integratie met .NET. C# is een populaire keuze voor het ontwikkelen van webapplicaties vanwege zijn sterke typen, moderne syntaxis en ondersteuning voor object georiënteerd programmeren. Dit stelt ons in staat om efficiënte en onderhoudbare code te schrijven voor onze webapplicatie.

**MVC-architectuur**: De keuze voor het MVC-architectuurpatroon is strategisch, omdat het ons in staat stelt om de logica van onze webapplicatie te structureren en te scheiden in drie afzonderlijke componenten: het model, de view en de controller. Dit vergroot de modulariteit en onderhoudbaarheid van onze code, waardoor het gemakkelijker wordt om functies toe te voegen en te onderhouden.

**Microsoft SQL**: Microsoft SQL Server is onze databasekeuze vanwege de naadloze integratie met .NET en C#. Het biedt uitstekende prestaties, gegevensintegriteit en beveiliging, wat cruciaal is voor het beheren van de reisinformatie van medewerkers en het optimaliseren van rittenplanning.

**LINQ**: We maken gebruik van LINQ als een krachtige taal geïntegreerde querytaal om gegevens te manipuleren en te beheren. Dit zal ons helpen bij het efficiënt verwerken van de reisinformatie en het genereren van nuttige inzichten voor medewerkers.

**Bootstrap CSS-framework**: Voor de gebruikersinterface vertrouwen we op het Bootstrap CSS-framework, omdat het ons in staat stelt om snel responsieve en gebruiksvriendelijke interfaces te creëren zonder uitgebreide handmatige styling. Dit zal de ontwikkelingstijd verkorten en zorgen voor een consistente en aantrekkelijke gebruikerservaring op verschillende apparaten en browsers.

**Testen**: We hebben voor xUnit gekozen voor het testen van onze webapplicatie vanwege zijn gestructureerde aanpak met het AAA-pattern. Mocking wordt toegepast om externe afhankelijkheden te isoleren en de betrouwbaarheid te vergroten. We schrijven zowel unit- als integratietesten om de betrouwbaarheid te waarborgen en een stabiele applicatie te garanderen, wat cruciaal is voor het succes van de ritplanningsfunctionaliteit en het bevorderen van efficiëntie en duurzaamheid.

Met deze technologische keuzes kunnen we een efficiënte, milieuvriendelijke en gebruiksvriendelijke webapplicatie ontwikkelen die het plannen en delen van ritten voor CM.com-medewerkers mogelijk maakt, waardoor zowel duurzaamheid als een gezellige werkomgeving worden bevorderd.