# Projectplanning

## Risico’s

* Connectie met het RDW
  + Het is belangrijk dat dit in het systeem zal zitten.
* CRM pakket
  + Het CRM pakket zelf zal niet in de scope van het project zitten. We nemen wel maatregelen om de migratie naar een CRM packet vloeiend te verlopen

## Sprintduur

Het project loop van woensdag tot de donderdag de week daarop. Wij hebben 6 sprints gepland. Voor sprint 0 rekenen wij 3 uur. Voor sprint 1 rekenen wij 4 uur. Sprint 0 en 1 worden beide op de eerste woendag gedaan. Voor alle andere sprints rekenen wij een dag.

## Scope

Het doel van het systeem is om van de excel sheets af te komen en het process te automatiseren. Ook is een het belangrijk om gegevens naar de RDW te kunnen sturen voor 1 Januari.

Het systeem moet minimaal gegevens van klanten en autos kunnen opslaan en dit tonen. Ook moet de onderhoudsopdrachten worden ingevuld worden.

Het systeem hoeft geen authenticatie te bevatten.

## Oplossingsrichting

Het syteem zal worden gebouwd met een SOA architectuur. De technieken die hierbij worden gebruikt zijn WCF, ASP.NET, MVC en EF.

# Requirements

## Functional Requirements

1. Een auto moet voor onderhoud aangeboden kunnen worden. De onderhoudsopdracht moet dan, samen met de overige klantinformatie, in het systeem gezet worden.
2. Een monteur moet aan kunnen geven met een nieuwe opdracht te beginnen
3. Een monteur moet aan kunnen geven klaar te zijn. Indien het een APK-keuring betreft, wordt er dan een automatisch bericht naar het RDW gestuurd. Dat kan betekenen dat de auto uitgekozen wordt voor een steekproef van het RDW.
4. Een monteur moet de auto klaar kunnen melden (na de steekproef)
5. De receptioniste moet klantgegevens kunnen wijzigen
6. Er moet overzicht zijn van de status van de auto’s die momenteel in onderhoud zijn.
7. De receptioniste moet een klantoverzicht kunnen krijgen
8. De receptioniste moet een overzicht van auto’s kunnen krijgen
9. De receptioniste moet de lijst met automerken en types kunnen bijwerken
10. De receptioniste moet een overzicht van reparaties kunnen krijgen

## Non-functional Requirements

1. De communicatie met het RDW dient te worden gelogd in een database
2. Alle technische fouten dienen worden gelogd in een log-file
3. Er moet een backup-strategie voor de database(s) komen
4. De applicatie moet door de MITS in beheer genomen worden en moet dus voldoen aan de Endeavour-richtlijnen m.b.t. code- en teststandaarden
5. De applicatie hoeft (nog) niet beveiligd te zijn, maar als er nog tijd over is, dan is dat wel fijn

# Architectuur

Er is gekozen voor een ruim opgezette SOA-Architectuur. Dit is pas de eerste release van de applicatie. We verwachten dat het systeem in de toekomst veel uitgebreider gaat worden.

ISRijksdienstWegverkeer

FEGarageManagementSysteem

Service Bus

BSVoertuigEnKlantbeheer

PcSOnderhoud

De **FEGarageManagementSysteem** implementeert het User Interface voor zowel de *receptioniste* als de *monteur*. Op het hoofdscherm staat een menulijstje met taken (use cases).

De **PcSOnderhoud** implementeert het bedrijfsproces van het doen van onderhoud (aanmelden, onderhoud starten, evt. contact opnemen met het RDW, voertuig klaarmelden). In deze service moet worden bewaard: de onderhoudsopdracht en de status van het voertuig dat in onderhoud is.

De **BSVoertuigEnKlantbeheer** beheert de primaire data van het garagebedrijf: de klantgegevens, de autogegevens en de onderhoudsgeschiedenis (niet de opdracht, maar wel de uitgevoerde wertkzaamheden).

De **ISRijksdienstWegverkeer** regelt de communicatie met het RDW.

In de toekomst komt klantinformatie mogelijk in een CRM-pakket te zitten. Het zou fijn zijn als de BSVoertuigEnKlantbeheer in de toekomst gesplitst kan worden in BSKlantbeheer en BSVoertuigbeheer.