# Operation contracts

## Initialisaties

**Contract CO001** : Initialiseer data

**Operation** :InitData(Exparc system)

**Crosse reference** :-

**Precondition**: Een medewerker wil het systeem beginnen te gebruiken

**Postconditions** : Een instantie van KlantenBestand kb wordt gemaakt.

Een instantie van TevredenheidsOnderzoekenBestand tb wordt gemaakt.

**Contract CO002** : Initieren van een sessie

**Operation** :login(username,wachtwoord)

**Crosse reference** :-

**Precondition**: Een medewerker wil in het systeem inloggen

**Postconditions** :

**Contract CO003** : Inloggen in het systeem

**Operation** :login(username,wachtwoord)

**Crosse reference** :-

**Precondition**: Een medewerker wil in het systeem inloggen

**Postconditions** :

## UC-1 Registeren parkeerbeurt

**Contract CO1** : Registreer Parkeerbeurt

**Operation** :registreerParkeerBeurt(nummerplaat,inRijMoment, uitRijMoment)

**Crosse reference** :UC1 : registreer parkeerbeurt

**Precondition** : Een wagen wordt geïdentificeerd aan de bareel

**Postconditions** : Het systeem detecteert of de wagen binnen of buitenrijdt. Case : binnenrijden:

Het systeem controleert de nummerplaat

Case : nummerplaat herkend:

* Een instantie van een Parkeerbeurt pb wordt aangemaakt.
* Indien registreerToestemming = TRUE, wordt pb geassocieerd met betreffende Abonnee ab instantie.
* pb.inRijMoment wordt huidige tijdstip

Case : nummerplaat niet herkend:

* Een instantie van een Parkeerbeurt pb wordt aangemaakt.
* pb.motor = typemotor
* pb.merk = merk
* pb.model = model
* Een instantie van ParkeerBewijs pbw wordt aangemaakt.
* pbw wordt geassocieerd met pb.
* pbw.bewijsidentificatie krijgt een waarde (en een fysieke vorm i.e. ticketje)
* pbw.betaald = FALSE
* pb.inRijMoment wordt huidige tijdstip
* pbw wordt geassocieerd met pb

Case : buitenrijden:

Het systeem controleert de nummerplaat

Case : nummerplaat herkend:

* pb.uitRijMoment wordt huidige tijdstip
* ab.totaalbeurten wordt verminderd met 1

Case : nummerplaat niet herkend:

* Betalen parkeerbeurt (aparte CO)
* pb.uitRijMoment wordt huidige tijdstip

## UC-2 Betalen parkeerbeurt

**Contract CO2** : Toon mogelijke betaalwijzen

**Operation** :toonBetaalWijzen()

**Crosse reference** :UC2 : betalen parkeerbeurt

**Precondition**: Een wagen wordt geïdentificeerd aan de bareel adhv nummerplaat en ParkeerBewijs identificatie en de nummerplaat wordt niet herkend als Abonnee instantie ab.nummerplaat attribuut

**Postconditions** : Systeem toont als output de enum<BetaalWijze>

**Contract CO3** : Toon totaal

**Operation** :toonTotaal()

**Crosse reference** :UC2 : betalen parkeerbeurt

**Precondition** : Parkeerder heeft een betaalwijze geselecteerd

**Postconditions** :

* + - * Instantie Parkeerbeurt pb wordt gezocht
      * Ophalen pb.inRijMoment
      * Set pb.uitRijMoment
      * Bereken tijdsverschil in de juiste eenheid en verdisconteer met de kostprijs per tijdseenheid.
      * Systeem toont totaal via output

**Contract CO4** : Stuur betaalgegevens naar betaalservice

**Operation** :stuurBetaalGegevens()

**Crosse reference** :UC2 : betalen parkeerbeurt

**Precondition** : Totaalbedrag werd berekend

**Postconditions** :

Case : Cash

* Parkeerder voert geld in
* Systeem berekent restbedrag tot deze 0 bedraagt of negatief en berekend in dat geval de teruggave
* Parkeerbewijs pbw.betaald = TRUE
* Systeem opent bareel

**Contract CO5** : controleer pincode

**Operation** :controleerPin()

**Crosse reference** :UC2 : betalen parkeerbeurt

**Precondition**: Een wagen wordt geïdentificeerd aan de bareel adhv nummerplaat en ParkeerBewijs identificatie en de nummerplaat wordt niet herkend als Abonnee instantie ab.nummerplaat attribuut en de parkeerder heeft betalingswijze met de kaart gekozen

**Postconditions** : Systeem stuurt ingevoerde PIN gegevens naar BetalingsService met een maximum van 3 keer. Indien de juiste PIN werd ingegeven, wordt pbw.betaald = TRUE

## UC-3 Registreren tevredenheidsonderzoek

**Contract CO6** : Registreren van een tevredenheidsonderzoek

**Operation** :registreerTevredenheidsOnderzoek()

**Crosse reference** :UC3 : Registeren tevredenheidsonderzoek

**Precondition**: Instantie van Klasse Medewerker wenst tevredenheidsonderzoek in te voeren

**Postconditions** :

* Instantie van TevredenheidsOnderzoek to wordt aangemaakt.
* To.naam krijgt een waarde

**Contract CO7** : Aanbieden lijst klanten

**Operation** :getKlantenLijst()

**Crosse reference** :UC3 : Registeren tevredenheidsonderzoek

**Precondition**: Instantie TevredenheidsOnderzoek to bestaat

**Postconditions** :

* Systeem toont alle klanten en er wordt een keuze gemaakt

**Contract CO8** : Registreren van een klantencontact

**Operation** :registreerKlantenContact()

**Crosse reference** :UC3 : Registeren tevredenheidsonderzoek

**Precondition**: Instantie TevredenheidsOnderzoek to bestaat en medewerker wenst een klantencontact in te geven

**Postconditions** :

* Instanctie van een KlantenContact kc wordt gemaakt
* kc.onderwerp krijgt een waarde
* kc.tijdstip krijgt huidige tijdstip
* kc.medewerker krijgt de Medewerker zijn naam
* kc.typeContact krijgt een waarde uit Enum<TypeContact>

**Contract CO9** : Aanbieden keuzecriteria

**Operation** :getCriteria()

**Crosse reference** :UC3 : Registeren tevredenheidsonderzoek

**Precondition**: Instantie TevredenheidsOnderzoek to bestaat

**Postconditions** :

* Systeem toont enum<Criterium>
* Instantie van TevredenheidsCriteria tc wordt aangemaakt
* tc.naam = naam
* tc.criterium = gekozen criterium
* associatie tussen tc en to wordt gemaakt

**Contract C10** : Registreren score

**Operation** :registreerScore()

**Crosse reference** :UC3 : Registeren tevredenheidsonderzoek

**Precondition**: Instantie TevredenheidsOnderzoek to, TevredenheidsCriteria tc en KlantenContact kc bestaat

**Postconditions** :

* Instantie van Score sc wordt gemaakt
* Sc.scoreBehaald = waarde
* Associaties tussen sc , tc, kc en to worden gemaakt

## UC-4 Registreren klant

**Contract C11** : Invoeren van gegevens

**Operation** :inputGegevens()

**Crosse reference** :UC4 : Registeren klant

**Precondition**: Medewerker wenst klant te registreren

**Postconditions**:

Medewerker geeft klantgegevens in in het system

**Contract C12** : Opvragen nummerplaat

**Operation** :zoekNummerplaat(nummerplaat)

**Crosse reference** :UC4 : Registeren klant

**Precondition**: Medewerker heeft klantgegevens ingegeven

**Postconditions**:

* + - * Input van nummerplaat
      * Systeem geeft nummperplaat door aan verkeersbelastingsinspectie
      * Systeem ontvangt gegevens

**Contract C13** : Registratie klant

**Operation** :registreerKlant(email, gsmnr,nummerplaat,automerk,automodel,Motor typeMotor)

**Crosse reference** :UC4 : Registeren klant

**Precondition**: Systeem heeft gegevens ontvangen

**Postconditions**:

* + - * Aanmaak instantie van Abonnee ab
      * ab. email = email
      * ab.gsmnr = gsmnr
      * ab.nummerplaat = nummerplaat
      * ab.automerk = automerk
      * ab.automodel = model
      * ab.typeMotor = motor

**Contract C14** : Bevestigen aanmaken klant aan medewerker

**Operation** :bevestigingMedewerker()

**Crosse reference** :UC4 : Registeren klant

**Precondition**: Instantie van Abonnee ab bestaat

**Postconditions**:

Systeem stuurt bevestiging aan medewerker

**Contract C15** : Bevestigen aanmaken klant aan abonnee

**Operation** :bevestigingKlantEmail()

**Crosse reference** :UC4 : Registeren klant

**Precondition**: Instantie van Abonnee ab bestaat

**Postconditions**:

* + - * Systeem zoekt ab.email
      * Systeem stuurt email naar ab.email