

Solvas Fleet - Use Cases

Steven Bastiaens
Benjamin Cousaert
Nils Mak
Niko Strijbol
Karel Vandenbussche
David Vandorpe
Domien Van Steendam

Contents

Algemene Usecases	2
Aanmaken van een [Object]	2
Wijzigen van een [Object]	3
Verwijderen/Archiveren van een [Object]	4
Oplijsten van [Objecten]	5
Gebruiker	5
Wijzigen van profiel	5
Wijzigen van wachtwoord van niet-ingelogde gebruiker	6
Premies	7
Use case: Opvragen van factuur van de polis door klant	7
Use case: Raadplegen van gegevens (van polis) door klant	7
Use case: Opvragen van de premie van een specifieke verzekeraar door klant	8
Use case: Opvragen premie van een subvloot door klant	8
Use case: Opvragen premie van een vloot door klant	9
Use case: Opvragen premie van een voertuig door klant	10
Use case: Simuleren van de premiewijzigingen bij het veranderen van verzekering en/of toevoegen van voertuigen	11
Voertuig	12
Aanmaken van een voertuig	12
Subvloot	12
Aanmaken van een subvloot	12

Verzekeringsmaatschappij	12
Use case: Facturen van een verzekeringsmaatschappij opvragen	12
Use case: Nog niet betaalde facturen van een verzekeringsmaatschappij bekijken	13
Extra	14
Use case: Importeren van Data	14
Use case: Exporteren van Data	15
Use case: afsluiten van een verzekering	15

Bij alle usecases gaan we bij pre-condities ervan uit dat de persoon die een bepaalde actie probeert uit te voeren ingelogd is in het systeem, en dat hij de nodige rechten heeft om een bepaalde actie uit te voeren.

Indien de persoon niet ingelogd is, wordt hij doorgestuurd naar een inlogpagina met de melding dat hij ingelogd moet zijn. Na het succesvol inloggen, kan hij verder gaan met zijn acties.

Indien de persoon niet de nodige rechten heeft, wordt hij doorgestuurd naar de index pagina met de melding dat hij niet de nodige rechten bezit voor wat hij wou doen.

Algemene Usecases

Hieronder worden enkele algemene usecases opgelijst. Zo staan hieronder de usecases om een [Object] aan te maken, te wijzigen en te verwijderen uit de databank, en om alle [Objecten] op te lijsten. Dit is een veralgemening voor de usecases voor gebruikers, klanten, vloten, subvloten en voertuigen.

De workflow voor deze specifieke usecases zal hetzelfde blijven, er zullen enkel een paar kleine verschillen zijn in de links waarop geklikt moet worden.

Aanmaken van een [Object]

Primaire actor: Gebruiker

Objectief: De gebruiker moet een [Object] kunnen toevoegen aan de databank

Normale flow:

1. De gebruiker geeft aan dat hij/zij een nieuwe [Object] wil toevoegen
2. De gebruiker krijgt een formulier voor een nieuw [Object] te zien
3. De gebruiker vult de gegevens in voor het [Object] dat hij wil toevoegen
4. De gebruiker submit het formulier
5. Het syteem vraagt de gebruiker om de gegevens te bevestigen

6. De gebruiker bevestigt de gegevens van het nieuwe [Object]
7. De gebruiker krijgt een bevestiging dat het [Object] werd toegevoegd

Pre-conditions:

- De gebruiker beschikt over de nodige informatie over het [Object]

Post-conditions:

- Het [Object] werd succesvol toegevoegd aan de databank

Alternatieve flow:

5. De ingevulde gegevens zijn fout (niet uniek, fout formaat, ...) of onvolledig
 - a. De gebruiker krijgt het formulier te zien met de ingevulde gegevens, waarbij aangeduid staat welke gegevens fout zijn
 - b. Na het aanpassen van het formulier kan de gebruiker het formulier weer submitten (stap (4))
6. De gebruiker annuleert de toevoeging
 - a. De gebruiker wordt doorgestuurd zonder zijn wijzigingen op te slaan

Exceptionele flow

7. Het [Object] wordt niet opgeslaan (wegens onvoorziene fout)
 - a. De gebruiker krijgt een foutmelding te zien, met de boodschap opnieuw te proberen

Wijzigen van een [Object]

Primaire actor: Gebruiker

Objectief: De gebruiker moet de gegevens van een [Object] kunnen wijzigen

Normale flow:

1. De gebruiker gaat naar de infopagina van het [Object]
2. De gebruiker geeft aan het [Object] te willen wijzigen
3. De gebruiker krijgt een formulier te zien voor het bestaande [Object], waarbij de gegevens al zijn ingevuld
4. De gebruiker wijzigt de gegevens van het [Object]
5. De gebruiker submit het formulier
6. Het systeem vraagt de gebruiker om de wijzigingen te bevestigen
7. De gebruiker bevestigt de wijzigingen van het [Object]
8. De gebruiker krijgt een bevestiging dat het [Object] gewijzigd werd

Post-conditions:

- Het [Object] werd gewijzigd

Alternatieve flow

6. De ingevulde gegevens zijn fout (niet uniek, fout formaat, ...) of onvolledig
 - a. De gebruiker krijgt het formulier te zien met de ingevulde gegevens, waarbij aangeduid staat welke gegevens fout zijn
 - b. Na het aanpassen van het formulier kan de gebruiker het formulier weer submitten (stap (4))
7. De gebruiker annuleert de wijzigingen
 - a. De gebruiker wordt doorgestuurd zonder zijn wijzigingen op te slaan

Exceptionele flow

7. Het [Object] wordt niet gewijzigd (wegens onvoorziene fout)
 - a. De gebruiker krijgt een foutmelding te zien, met de boodschap opnieuw te proberen

Verwijderen/Archiveren van een [Object]

Primaire actor: Gebruiker

Objectief: De gebruiker moet een [Object] kunnen verwijderen/archiveren

Normale flow:

1. De gebruiker gaat naar de infopagina van het [Object]
2. De gebruiker geeft aan het [Object] te willen verwijderen/archiveren
3. Het systeem vraagt om bevestiging
4. De medewerker bevestigt de verwijdering
5. De gebruiker krijgt een bevestiging dat het [Object] verwijderd werd

Post-conditions:

- Het [Object] werd verwijderd

Alternatieve flow

2. Het [Object] heeft nog relaties openstaan, bv een klant die nog verzekeringen heeft
 - a. De gebruiker wordt verwittigd dat deze relaties ook verwijderd/gearchiverd zullen worden
 - b. Ga verder met stap (3)

4. De gebruiker annuleert de verwijdering
 - a. De gebruiker wordt doorgestuurd

Oplijsten van [Objecten]

Primaire actor: Gebruiker

Objectief: De gebruiker moet [Objecten] kunnen oplijsten

Normale flow:

1. De gebruiker geeft aan de [Objecten] te willen oplijsten
2. De gebruiker krijgt een oplijsting van alle [Objecten] te zien, met mogelijkheid om te filteren
3. De gebruiker verfijnt zijn zoekopdracht, gebruikmakend van de filters
4. De gebruiker bevestigt de zoekopdracht
5. De gebruiker krijgt een oplijsting te zien van alle [Objecten] die aan de zoekopdracht voldoen te zien

Post-condities:

- De gebruiker krijgt een lijst van de [Objecten] te zien

Alternatieve flow:

3. De gebruiker wist de zoekopdracht
 - a. Ga naar stap (2)

Gebruiker

Wijzigen van profiel

Primaire actor: Gebruiker

Objectief: De gebruiker moet zijn eigen gegevens kunnen aanpassen

Normale flow

1. De gebruiker gaat naar zijn instellingen
2. De gebruiker past zijn gegevens aan
3. De gebruiker bevestigt de wijzigingen
4. De gebruiker krijgt een bevestiging dat zijn instellingen zijn aangepast

Post-condities

De instellingen van de gebruiker zijn aangepast

Alternatieve flow

2. Indien de gebruiker zijn wachtwoord wil aanpassen, moet hij eerst zijn huidige wachtwoord ingeven
3. Indien de gebruiker zijn email wil aanpassen, zal hij zijn nieuwe emailadres eerst moeten bevestigen
4. De gebruiker annuleert de wijzigingen
 - a. De gebruiker wordt doorgestuurd

Wijzigen van wachtwoord van niet-ingelogde gebruiker

Primaire actor: Gebruiker

Objectief: De gebruiker moet zijn wachtwoord kunnen aanpassen zonder in te loggen indien hij vergeten is

Normale flow:

1. De gebruiker slaagt er niet in om in te loggen
2. De gebruiker geeft aan dat hij zijn wachtwoord vergeten is
3. De gebruiker vult zijn emailadres in
4. De gebruiker bevestigt zijn emailadres
5. De gebruiker krijgt de melding dat hij een email met een wachtwoord reset link heeft gekregen
6. De gebruiker volgt deze link
7. De gebruiker kan een nieuw wachtwoord kiezen door het tweemaal in te vullen
8. De gebruiker bevestigt zijn nieuwe wachtwoord
9. De gebruiker krijgt een bevestigingsemail en wordt doorgestuurd naar de inlogpagina

Pre-condities:

- De gebruiker is niet ingelogd op de webapplicatie

Post-condities:

- De gebruiker heeft een nieuw wachtwoord ingesteld

Alternatieve flow:

4. Het emailadres is niet gekend in de databank
 - a. Stap (5) wordt nog steeds uitgevoerd
 - b. De gebruiker zal niet verder kunnen aangezien hij geen email heeft ontvangen
8. De wachtwoorden komen niet overeen
 - a. De gebruiker krijgt een foutboodschap te zien
 - b. Ga naar stap (7)

Premies

Use case: Opvragen van factuur van de polis door klant

Primaire actor: De klant (een bedrijf)

Objectief: De klant moet in staat zijn om elke factuur van zijn polis te raadplegen

Normale flow:

1. De gebruiker selecteert de datum van de factuur die hij wil raadplegen
2. Het systeem haalt de factuur van de gebruiker op de gekozen datum uit het archief en geeft deze weer aan de gebruiker

Pre-condities: De klant is ingelogd op de webapplicatie

Post-condities: De klant heeft zijn factuur ontvangen

Alternatieve flow:

- (1). De gebruiker selecteert geen datum
 - a. Het systeem vult de huidige datum in
 - b. Het systeem berekent de nieuwe factuur voor de huidige periode en geeft deze weer
- (1). De gebruiker selecteert de huidige datum
 - a. Het systeem berekent de nieuwe factuur voor de huidige periode en geeft deze weer

Use case: Raadplegen van gegevens (van polis) door klant

Primaire actor: De klant (een bedrijf)

Objectief: De klant moet in staat zijn om alle nodige informatie over zijn polis op een bepaalde datum te bekijken

Normale flow:

1. De gebruiker selecteert de datum van de gegevens die hij wil raadplegen
2. Het systeem geeft de gegevens weer van de gebruiker op de gekozen datum

Pre-conditions: De klant is ingelogd op de webapplicatie

Post-conditions: De klant krijgt zijn gegevens te zien

Alternatieve flow:

- (1). De gebruiker selecteert geen datum
 - a. Het systeem vult de huidige datum in
 - b. Ga naar (2)

Use case: Opvragen van de premie van een specifieke verzekeraar door klant

Primaire actor: De klant (een bedrijf)

Objectief: De klant moet in staat zijn om de verschillende premies van zijn verzekeraars te raadplegen

Normale flow:

1. De gebruiker selecteert de verzekeraar waarvan hij de premie wil krijgen
2. Het systeem geeft de premie weer voor de klant van de gekozen verzekeraar

Pre-conditions: De klant is ingelogd op de webapplicatie

Post-conditions: De klant heeft de premie van een bepaalde verzekeraar verkregen

Alternatieve flow:

- (1). De gebruiker selecteert geen verzekeraar
 - a. Het systeem geeft een melding van het probleem
 - b. ga naar (1)

Use case: Opvragen premie van een subvloot door klant

Primaire actor: De klant (een bedrijf)

Objectief: De klant moet in staat zijn de premie van een subvloot op te vragen

Normale flow:

1. De gebruiker selecteert een vloot
2. Het systeem toont informatie over de vloot
3. De gebruiker selecteert een subvloot van de vloot
4. Het systeem toont informatie over de subvloot
5. De gebruiker vraagt de premie op van de subvloot
6. Het systeem geeft de premie weer van de subvloot

Pre-condities: De klant is ingelogd op de webapplicatie

Post-condities: De klant kent de premie van een bepaalde subvloot

Alternatieve flow:

- (5). Niet alle voertuigen hebben een premie waardoor de subvloot-premie niet weergegeven kan worden
 - a. Het systeem laat weten welke voertuigen nog geen premie hebben en geeft een voorlopige premie weer
- (5). Er zijn nog geen voertuigen met een premie waardoor de subvloot-premie niet weergegeven kan worden
 - a. Het systeem laat weten dat niet alle voertuigen een premie hebben

Exceptionele flow:

- (1). De gebruiker heeft geen vloten
 - a. Het systeem geeft een melding dat er eerst vloten toegekend moeten worden
- (3). De vloot heeft geen subvloten
 - a. Het systeem geeft een melding dat er eerst subvloten toegekend moeten worden aan de vloot

Use case: Opvragen premie van een vloot door klant

Primaire actor: De klant (een bedrijf)

Objectief: De klant moet in staat zijn de premie van een vloot op te vragen

Normale flow:

1. De gebruiker selecteert een vloot
2. Het systeem toont informatie over de vloot
3. De gebruiker vraagt de premie op van de vloot
4. Het systeem geeft de premie weer van de vloot

Pre-condities: De klant is ingelogd op de webapplicatie

Post-condities: De klant kent de premie van een bepaalde vloot

Alternatieve flow:

- (3). Niet alle voertuigen hebben een premie waardoor de vloot-premie niet weergegeven kan worden
- a. Het systeem laat weten welke voertuigen nog geen premie hebben en geeft een voorlopige premie weer
- (3). Er zijn nog geen voertuigen met een premie waardoor de vloot-premie niet weergegeven kan worden
- a. Het systeem laat weten dat nog niet alle voertuigen een premie hebben

Exceptionele flow:

- (1). De gebruiker heeft geen vloten
- a. Het systeem geeft een melding dat er eerst vloten toegekend moeten worden

Use case: Opvragen premie van een voertuig door klant

Primaire actor: De klant (een bedrijf)

Objectief: De klant moet in staat zijn de premie van een voertuig op te vragen

Normale flow:

1. De gebruiker selecteert een vloot
2. Het systeem toont informatie over de vloot
3. De gebruiker selecteert een subvloot van de vloot
4. Het systeem toont informatie over de subvloot
5. De gebruiker selecteert een voertuig van de subvloot
6. Het systeem toont informatie over dit voertuig
7. De gebruiker vraagt de premie op van dit voertuig
8. Het systeem geeft de premie weer van het voertuig

Pre-condities: De klant is ingelogd op de webapplicatie

Post-condities: De klant kent de premie van een bepaald voertuig

Alternatieve flow:

- (7). Het voertuig heeft geen premie waardoor deze niet weergegeven kan worden
- a. Het systeem laat weten dat het voertuig nog geen premie heeft

Exceptionele flow:

- (1). De gebruiker heeft geen vloten
 - a. Het systeem geeft een melding dat er eerst vloten toegekend moeten worden
- (3). De vloot heeft geen subvloten
 - a. Het systeem geeft een melding dat er eerst subvloten toegekend moeten worden aan de vloot
- (5). De subvloot heeft geen voertuigen
 - a. Het systeem geeft een melding dat er eerst voertuigen toegevoegd moeten worden aan de subvloot

Use case: Simuleren van de premiewijzigingen bij het veranderen van verzekering en/of toevoegen van voertuigen

Primaire actor: De klant (een bedrijf)

Objectief: De klant moet in staat zijn te weten wat er wijzigt bij het overschakelen naar een andere verzekering of bij het toevoegen van een voertuig

Normale flow:

1. De gebruiker geeft aan dat hij een simulatie wil starten
2. Het systeem vraagt om de huidige verzekering te wijzigen in een andere verzekering
3. De gebruiker geeft een andere verzekering op
4. Het systeem geeft de nieuwe premie weer, berekend met de gekozen verzekering in de simulatie en vraagt de gebruiker een voertuig toe te voegen, te verwijderen, te wijzigen of een andere verzekering te kiezen
5. De gebruiker voegt een voertuig toe
6. Het systeem geeft de nieuwe premie weer, berekend met de gekozen verzekering en het gewijzigde aantal en type voertuigen

Pre-condities: De klant is ingelogd op de webapplicatie

Post-condities: De klant heeft een simulatie van mogelijke premie wijzigingen gemaakt

Alternatieve flow:

- (3). De gebruiker geeft geen andere verzekering op
 - a. Het systeem geeft de huidige premie weer en vraagt de gebruiker een voertuig toe te voegen of te verwijderen, of een andere verzekering te kiezen

- b. Ga naar 5
 - (5). De gebruiker verwijdert een voertuig
- a. Ga naar 6
 - (5). De gebruiker wijzigt het type van een voertuig
- a. Ga naar 6
 - (5). De gebruiker wijzigt de verzekering opnieuw
- a. Ga naar 4

Voertuig

Aanmaken van een voertuig

Het aanmaken van een voertuig is een uitbreiding van de algemene toevoeg usecase. Om aan te geven dat de gebruiker een nieuw voertuig wil toevoegen, moet hij wel eerst de vloot en subvloot selecteren waaraan hij dit voertuig wil toevoegen.

Subvloot

Aanmaken van een subvloot

Het aanmaken van een subvloot is een uitbreiding van de algemene toevoeg usecase. Om aan te geven dat de gebruiker een nieuwe subvloot wil toevoegen, moet hij wel eerst de vloot selecteren waaraan hij deze subvloot wil toevoegen.

Verzekeringsmaatschappij

Use case: Facturen van een verzekeringsmaatschappij opvragen

Primaire actor: Medewerker verzekeringsmaatschappij (voortaan gewoonweg medewerker genoemd)

Objectief: De verzekeringsmaatschappij moet een overzicht kunnen krijgen met alle facturen van alle aangesloten makelaars.

Normale flow:

1. De medewerker geeft aan dat hij de facturen wil opvragen.

2. Er wordt gevraagd naar een start en einddatum waartussen hij wil zoeken (dit is verplicht). Hierbovenop kunnen optionele parameters zoals een bepaalde makelaar, betaalstatus (betaald of nog niet betaald) of bedragintervaal kunnen meegegeven worden.
3. De medewerker bevestigt de parameters en geeft deze aan het systeem door.
4. Het systeem stelt intern deze lijst op en toont deze aan de medewerker.
5. De zoekparameters kunnen veranderd worden waardoor men terugvalt naar stap 2 of men kan deze lijst printen.

Pre-conditions: De medewerker is ingelogd op de webapplicatie en is verbonden met een verzekeringsmaatschappij.

Post-conditions: De medewerker ontvangt een lijst van de facturen waarvoor hij geïnteresseerd is.

Alternatieve flow:

- (2). De gegevens zijn fout vb (einddatum < startdatum,..)
 - a. Het systeem toont de foutmelding.
 - b. Ga naar stap 2 en probeer opnieuw.
- (5). De printer werd niet gevonden of er vond een andere IO-fout plaats.
 - a. Het systeem toont de foutmelding afkomstig van de printer.
 - b. Ga naar stap 5 en probeer opnieuw.

Exceptionele flow:

- (1). De medewerker heeft geen rechten om de facturen te bekijken.
 - a. Het systeem toont een foutmelding, met de mogelijkheid om rechten aan te vragen aan een overste van de verzekeringsmaatschappij.
- (4). De medewerker heeft geen rechten om de facturen te bekijken.
 - a. Het systeem toont een foutmelding, met de mogelijkheid om rechten aan te vragen aan een overste van de verzekeringsmaatschappij.

Use case: Nog niet betaalde facturen van een verzekeringsmaatschappij bekijken

Primaire actor: Medewerker verzekeringsmaatschappij (voortaan gewoonweg medewerker genoemd)

Objectief: De verzekeringsmaatschappij moet kunnen zien welke facturen nog niet betaald geweest zijn door de aangesloten makelaars.

Normale flow:

1. <<include “facturen opvragen”>> De medewerker geeft aan het systeem de waarde “niet betaald” mee voor de zoekparameter “betaalstatus”.
2. Het systeem toont de gewenste lijst aan de medewerker.

Pre-condities: De medewerker is ingelogd op de webapplicatie en is aangesloten als werknemer bij een verzekeringsmaatschappij.

Post-condities: De medewerker ziet een lijst van alle niet-betaalde facturen.

Exceptionele flow:

- (1). De medewerker heeft geen rechten om de facturen te bekijken.
- a. Het systeem toont een foutmelding, met de mogelijkheid om rechten aan te vragen aan een overste van de verzekeringsmaatschappij.

Extra

Use case: Importeren van Data

Primaire actor: Gebruiker

Objectief: De gebruiker moet data via een csv (of excel) data in bulk kunnen importeren in het database systeem.

Normale flow:

1. De gebruiker geeft aan dat hij een import wil starten
2. De gebruiker kiest zijn bestand om te importeren
3. Het systeem toont welke wijzigingen aangebracht zullen worden aan het databanksysteem
4. De gebruiker bevestigt de wijzigingen
5. De gebruiker wordt doorgestuurd naar een bevestigingspagina, en zal later een notificatie krijgen dat de wijzigingen succesvol zijn doorgevoerd

Pre-condities: De gebruiker is ingelogd op de webapplicatie en heeft de nodige rechten

Post-condities: De gebruiker heeft zijn gegevens geïmporteerd

Alternatieve flow: * (3). Het bestand bevat foute gegevens

- a. Het systeem toont de foute gegevens, met een melding
 - b. Ga naar 2
- (4). De gebruiker aanvaardt de wijzigingen niet

- a. Ga naar 1

Use case: Exporteren van Data

Primaire actor: Gebruiker

Objectief: De gebruiker moet (een deel van) het databasesysteem kunnen exporteren naar een csv

Normale flow:

1. De gebruiker geeft aan dat hij een export wil starten
2. De gebruiker kiest de gegevens die hij wil exporteren
3. De gebruiker kiest het formaat waarin hij de data wil exporteren
4. De gebruiker bevestigt zijn export
5. Het systeem stuurt de gebruiker door naar een bevestigingspagina
6. Het systeem stuurt een mail met de export naar de gebruiker

Pre-conditions: De gebruiker is ingelogd op de webapplicatie en heeft de nodige rechten

Post-conditions: De gebruiker heeft zijn data geëxporteerd

Use case: afsluiten van een verzekering

Primaire actor: De beheerder

Objectief: Er wordt een verzekering afgesloten tussen de klant en een verzekeringsmaatschappij over een voertuig.

Normale flow:

1. De beheerder selecteert het account van de klant.
2. De beheerder selecteert het voertuig waarvoor de klant een verzekering wilt hebben door de juiste vloot, subvloot te kiezen.
3. De klant geeft aan welk type verzekering hij wilt voor het voertuig.
4. De klant kan uit een lijst kiezen welke verzekering hij/zij wilt.
5. De beheerder selecteert deze verzekering en slaat de upgedate gegevens op

Pre-conditions: De klant heeft een profiel bij solvas en heeft minstens 1 voertuig in zijn vloot alsook is de beheerder ingelogd in het systeem.

Post-conditions: Het voertuig heeft een nieuwe verzekering

Alternatieve flow: * (1). De beheerder heeft de rechten niet om het account van de klant te beheren

- a. De beheerder vraagt deze rechten aan aan de administrator
- b. Ga naar 2
 - (3). Het gekozen voertuig heeft al een verzekering voor op het voertuig voor het gekozen type
- a. Het systeem geeft een melding van het probleem
- b. Het systeem zet de huidige verzekering stop na bevestiging
- c. Ga naar 4