

Functioneel ontwerp

November 2019

# 

# Voorwoord

Dit functioneel ontwerp bevat informatie over de functionele aspecten van ons project: de Update Tracker.

Huidige situatie: er is bijvoorbeeld software wat een bedrijf niet vaak gebruikt maar wel cruciaal is voor de veiligheid is van de systemen, doordat de software niet vaak wordt gebruikt wordt de software niet vaak geupdate en ontstaat er een veiligheidsrisico.

Gewenste situatie: gebruiker (bijvoorbeeld bedrijf) geeft aan welke software hij gebruikt en welke versie hij er momenteel van heeft. Als er een nieuwere softwareversie is krijgt de gebruiker een mailtje met daarin de info (welke software moet worden geupdate en naar welke versie) de gebruiker kan vervolgens aangeven dat hij hem heeft geupdate naar de nieuwste versie. Op die manier weet het systeem altijd de juiste versie die de gebruiker heeft.

Als de klant niet betaald voor het product kan hij maar twee software systemen toevoegen aan de update tracker en krijgt hij maar één mailtje per maand. Als de gebruiker wel betaald kan hij unlimited software systemen toevoegen en krijgt hij elke dag een mailtje als dat nodig is.

# 

# Inhoudsopgave

[**Voorwoord**](#_t4ebjjtroxpo) **1**

[**Inhoudsopgave**](#_kcgtui27zbj2) **2**

[**Samenvatting**](#_e0u8nlwvtsto) **3**

[**Analyse huidige situatie**](#_30voj2qkpkjq) **3**

[**Analyse gewenste situatie**](#_dhtpx9b1jqxz) **3**

[Requirements](#_4yxu305bmvb5) 4

[Must haves:](#_908wadi5d9xu) 4

[Should haves:](#_f2rz6mgtxst5) 6

[Could haves:](#_1xx5k6ekaphl) 6

[Won’t haves:](#_prvx1btuhhv6) 6

[Inforamtieverwerking](#_9l59589cu8gu) 7

[Use cases gebruiker](#_96y891mhmq9l) 7

[Use cases admin](#_4n3lx3u2o8vc) 8

[Use case systeem](#_tn3pb8wwrhh1) 9

[**Use case tabellen**](#_omqfrmmq0cby) **10**

[Login:](#_84neitlybypx) 10

[Uitloggen](#_18qcujjcfe8c) 11

[Account aanmaken](#_l3tqc7kikcs0) 12

[Add software](#_2sg0htsxlfn1) 13

[Remove software](#_mjpwpj8x9mox) 14

[Remove user](#_76e9so1jj5ol) 15

[Edit software](#_v3gol8gumce6) 16

[Edit user (gebruiker)](#_g6seaet748ub) 17

[Edit user (admin)](#_td5fq8gsz82m) 18

[Toon data](#_58wqnhkjxqe2) 19

[Get software update](#_q9hmjs2cl78p) 20

[Api data ophalen](#_hxmdz1oualcw) 21

[Controleren of user software update heeft](#_1u5ii29uvmbu) 22

[Mail versturen](#_7k8ezfxtc9z) 23

[Paying customer](#_7xfm0etwhnk7) 24

[**Interfaces**](#_s5we7949ucun) **25**

# 

# Samenvatting

Ik ga een systeem maken waar mensen hun software kunnen toevoegen aan een lijst. Het systeem haalt nieuwe software versies op uit een API van derden. Vervolgens vergelijkt het systeem de gebruiker zijn software met de nieuwe software versies. Als er een nieuwe softwareversie is voor een gebruiker dan krijgt de gebruiker een mailtje met het bericht dat de gebruiker moet updaten wegens veiligheidsredenen. In het mailtje zal een knop zitten waarmee de gebruiker kan aangeven dat hij/zij zijn software heeft geupdate.

# 

# Analyse huidige situatie

De huidige situatie valt het beste te omschrijven met een voorbeeld. Er is bijvoorbeeld software wat een bedrijf niet vaak gebruikt maar wel cruciaal is voor de veiligheid is van de systemen, doordat de software niet vaak wordt gebruikt wordt de software niet vaak geupdate en ontstaat er een veiligheidsrisico.

# 

# Analyse gewenste situatie

Gebruiker (bijvoorbeeld bedrijf) geeft aan welke software hij gebruikt en welke versie hij er momenteel van heeft. Als er een nieuwere softwareversie is krijgt de gebruiker een mailtje met daarin de info (welke software moet worden geupdate en naar welke versie) de gebruiker kan vervolgens aangeven dat hij hem heeft geupdate naar de nieuwste versie. Op die manier weet het systeem altijd de juiste versie die de gebruiker heeft.

## Requirements

MosCoW is een samenstelling van beginletters die ergens voor staan. De 2x o zijn ingevoegd om het woord ‘moscow’ leesbaar te maken, zonder dat ze betekenis hebben. De **M** staat voor **Must-haves**, de **S** staat voor **Should-haves**, de **C** staat voor **Could-haves** en de **W** staat voor **Won’t-haves**. De ‘moscow’ analyse gebaseerd op de vragen die eerder zijn gesteld aan de opdrachtgever.

### Must haves:

* Account aanmaken
  + De gebruiker voert zijn e mail, username, wachtwoord en wachtwoord in. vervolgens drukt de user op sign up. Als de user 2 keer hetzelfde wachtwoord heeft ingevuld wordt er een account aangemaakt. Anders krijgt de gebruiker een foutmelding en wordt hem gevraagd de gegevens opnieuw in te loggen. Als de gebruiker succesvol een account heeft aangemaakt wordt de gebruiker doorgestuurd naar de inlogpagina.
* Inloggen
  + Als de gebruiker in het bezit is van een account kan hij/zij zijn username en password invullen. Vervolgens kan de gebruiker op login drukken als de gegevens juist zijn dan wordt er een sessie aangemaakt en kan de gebruiker in het systeem. Anders krijgt de gebruiker een foutmelding.
* Uitloggen
  + Als de gebruiker is ingelogd dan kan de gebruiker op de uitlog knop drukken. Vervolgens beëindigd het systeem de sessie en stuurt het systeem de gebruiker door naar de inlogpagina.
* Add software
  + Als de gebruiker is ingelogd kan de user software toevoegen. De gebruiker vult software naam en softwareversie in. vervolgens drukt de gebruiker op voeg toe. Als de gebruiker geen velden heeft leeg gelaten wordt de software toegevoegd en wordt de gebruiker doorgestuurd naar de home pagina. Anders krijgt de gebruiker een foutmelding.
* Remove user
  + Als de gebruiker is ingelogd kan de user software verwijderen. De gebruiker drukt op delete en krijgt de vraag weet u het zeker. Als de user op ja drukt wordt de software verwijderd en wordt de user doorgestuurd naar de home pagina. Anders wordt de software niet verwijderd.
* Edit software
  + Als de gebruiker is ingelogd kan de user software aanpassen. De user drukt op edit en wordt doorgestuurd naar edit pagina. De user veranderd de software naam en/of software versie en drukt op opslaan. Als alle velden gevuld zijn wordt de verandering opgeslagen. Anders krijgt de gebruiker een foutmelding en moet de gebruiker het opnieuw proberen.
* Edit user (gebruiker)
  + Als de gebruiker is ingelogd kan de user zijn eigen account data aanpassen. De user drukt op profiel en wordt doorgestuurd naar zijn profielpagina. De user veranderd de usernaam en/of wachtwoord en drukt op opslaan. Als alle velden gevuld zijn wordt de verandering opgeslagen. Anders krijgt de gebruiker een foutmelding en moet de gebruiker het opnieuw proberen.
* Edit user (admin)
  + Als de gebruiker is ingelogd als admin kan de gebruiker iemand anders zijn account aanpassen. De gebruiker drukt op edit en wordt doorgestuurd naar een aanpas pagina. De gebruiker veranderd de usernaam en/of wachtwoord en drukt op opslaan. Als alle velden gevuld zijn wordt de verandering opgeslagen. Anders krijgt de gebruiker een foutmelding en moet de gebruiker het opnieuw proberen.
* Toon data
  + Als de gebruiker is ingelogd kan de gebruiker data aanvragen bij de database.
* Get software update
  + Als de gebruiker zijn software een update heeft dan krijgt de gebruiker een mailtje. Vervolgens update de user en geeft door aan het systeem dat hij/zij heeft geupdate. User zijn huidige softwareversie wordt geupdate. Als de user niet aangeeft dat hij /zij heeft geupdate update het systeem de huidige softwareversie niet.
* Api data ophalen
  + Het systeem haalt 1 keer per dag data uit de api. En slaat het op in de database
* Controleren of user software update heeft
  + Het systeem controleert 1 keer per dag of een user zijn huidige software een update heeft.
* Mail versturen
  + Het systeem verstuurt 1 keer per dag een mailtje met info waarin staat welke updates een user moet doen. In het mailtje staat ook een linkje daar kan een user op klikken om aan te geven dat de user zijn software heeft geupdate.

### Should haves:

* Duidelijke en gebruiksvriendelijke styling (gebruiker moet zijn weg goed kunnen vinden)
  + Het is belangrijk dat de styling gebruiksvriendelijk is. Zonder gebruiksvriendelijke styling kan de gebruiker een slecht gevoel krijgen bij het gebruiken van de website. Om er achter te komen wat gebruiksvriendelijke styling is ga ik het testen in de testperiode.

### Could haves:

* Styling na eigen smaak (wel overleggen met opdrachtgever of hij het goed keurt)
  + Styling is uitgetekend in adobe xd zie functioneel ontwerp.

### Won’t haves:

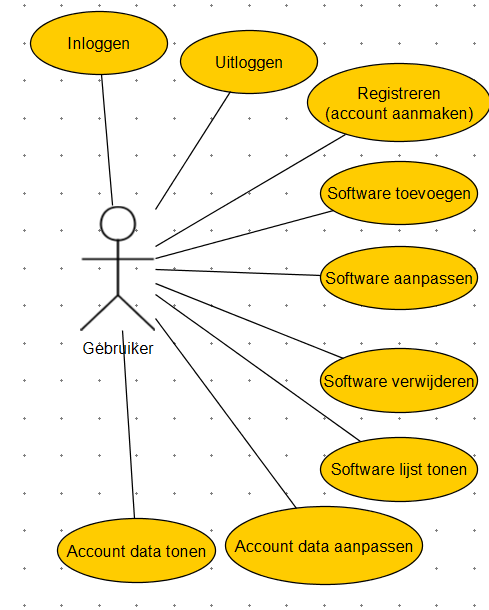
### 

## 

## Inforamtieverwerking

### Use cases gebruiker

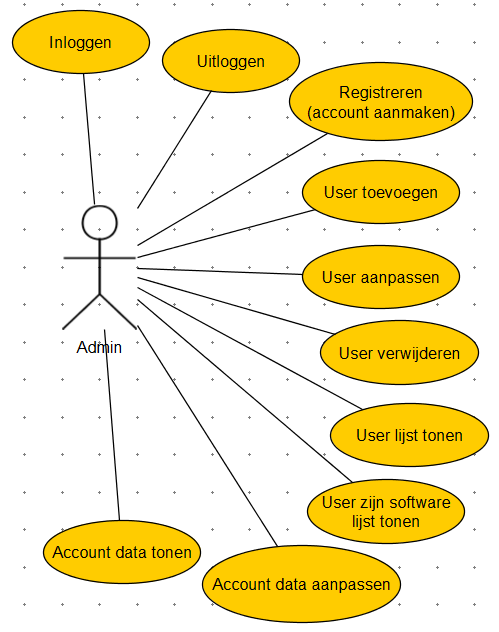
In deze use case is omschreven welke functionaliteiten de normale gebruiker heeft. De gebruiker heeft toegang tot inloggen, uitloggen, registreren (account aanmaken), software toevoegen, software aanpassen, software verwijderen, software lijst tonen, account data aanpassen en account data tonen.



### 

### Use cases admin

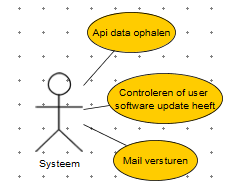
In deze use case is omschreven welke functionaliteiten de admin heeft. De admin heeft toegang tot inloggen, uitloggen, registreren (account aanmaken), user toevoegen, user aanpassen, user verwijderen, user lijst tonen, user zijn software lijst tonen, account data aanpassen en account data tonen.



### 

### Use case systeem

In deze use case is omschreven welke functionaliteiten het systeem heeft. Het systeem runt de volgende functionaliteiten automatisch; Api data ophalen, controleren of user software update heeft en mail versturen.



# 

# Use case tabellen

## Login:

Als de gebruiker in het bezit is van een account kan hij/zij zijn username en password invullen. Vervolgens kan de gebruiker op login drukken als de gegevens juist zijn dan wordt er een sessie aangemaakt en kan de gebruiker in het systeem. Anders krijgt de gebruiker een foutmelding.

|  |  |
| --- | --- |
| Naam | Inloggen |
| Versie | 1.0 |
| Actor | Gebruiker |
| Preconditie | Klant is bezit van account |
| Scenario | Gebruiker logt in   1. Voer username in. 2. Voer password in. 3. Druk op login |
| Uitzonderingen | Gebruiker voert niet de juiste gegevens in   1. Voer username in. 2. Voer password in. 3. Druk op login 4. Melding verkeerde gegevens |
| Niet functionele eisen | Gebruiker moet binnen 20 seconden ingelogd zijn. |
| Postconditie | Sessie wordt aangemaakt. User kan nu in het systeem. |

## Uitloggen

Als de gebruiker is ingelogd dan kan de gebruiker op de uitlog knop drukken. Vervolgens beëindigd het systeem de sessie en stuurt het systeem de gebruiker door naar de inlogpagina.

|  |  |
| --- | --- |
| Naam | Uitloggen |
| Versie | 1.0 |
| Actor | Gebruiker |
| Preconditie | Klant is ingelogd |
| Scenario | Gebruiker logt uit   1. Gebruiker drukt op uitloggen 2. Sessie wordt beëindigd 3. Gebruiker wordt doorgestuurd naar inlog pagina |
| Uitzonderingen |  |
| Niet functionele eisen | Gebruiker moet binnen 20 seconden ingelogd zijn. |
| Postconditie | Sessie wordt aangemaakt. User kan nu in het systeem. |

## Account aanmaken

De gebruiker voert zijn e mail, username, wachtwoord en wachtwoord in. vervolgens drukt de user op sign up. Als de user 2 keer hetzelfde wachtwoord heeft ingevuld wordt er een account aangemaakt. Anders krijgt de gebruiker een foutmelding en wordt hem gevraagd de gegevens opnieuw in te loggen. Als de gebruiker succesvol een account heeft aangemaakt wordt de gebruiker doorgestuurd naar de inlogpagina.

|  |  |
| --- | --- |
| Naam | Account aanmaken |
| Versie | 1.0 |
| Actor | Gebruiker |
| Preconditie |  |
| Scenario | Account aanmaken   1. Voert email in 2. Voert username 3. Voert wachtwoord in 4. Voert wachtwoord in 5. Drukt op sign up |
| Uitzonderingen | Gebruiker vult niet 2 keer dezelfde wachtwoorden in   1. Voert email in 2. Voert username 3. Voert wachtwoord in 4. Voert wachtwoord in 5. Drukt op sign up 6. Toon melding dat het aanmaken van account niet gelukt is 7. Begin weer bij stap 1 |
| Niet functionele eisen | Het moet duidelijk zijn wat de gebruiker allemaal moet invullen |
| Postconditie | Gebruiker wordt doorgestuurd naar inlogpagina zodat hij/zij kan inloggen met zijn nieuwe account. |

## Add software

Als de gebruiker is ingelogd kan de user software toevoegen. De gebruiker vult software naam en softwareversie in. vervolgens drukt de gebruiker op voeg toe. Als de gebruiker geen velden heeft leeg gelaten wordt de software toegevoegd en wordt de gebruiker doorgestuurd naar de home pagina. Anders krijgt de gebruiker een foutmelding.

|  |  |
| --- | --- |
| Naam | Add software |
| Versie | 1.0 |
| Actor | Gebruiker |
| Preconditie | Moet ingelogd zijn |
| Scenario | Add software   1. Vult in software naam 2. Vult in software versie 3. Drukt op voeg toe |
| Uitzonderingen | Laat veld leeg   1. Vult in software naam 2. Drukt op voeg toe 3. Geef melding dat alle velden ingevuld moeten zijn   Heeft max bereikt   1. Vult in software naam 2. Drukt op voeg toe 3. Geef melding dat max is bereikt |
| Niet functionele eisen | Het moet duidelijk zijn wat de gebruiker allemaal moet invullen |
| Postconditie | Gebruiker wordt doorgestuurd naar home pagina |

## Remove software

Als de gebruiker is ingelogd kan de user software verwijderen. De gebruiker drukt op delete en krijgt de vraag weet u het zeker. Als de user op ja drukt wordt de software verwijderd en wordt de user doorgestuurd naar de home pagina. Anders wordt de software niet verwijderd.

|  |  |
| --- | --- |
| Naam | Remove software |
| Versie | 1.0 |
| Actor | Gebruiker |
| Preconditie | Moet ingelogd zijn |
| Scenario | Remove software   1. Drukt op delete 2. Krijgt vraag weet u het zeker 3. Vult ja in 4. Software wordt verwijderd |
| Uitzonderingen | User drukt op nee   1. Drukt op delete 2. Krijgt vraag weet u het zeker 3. Vult nee in 4. Software wordt niet verwijderd |
| Niet functionele eisen | Het verwijderen moet snel en soepel verlopen |
| Postconditie | Home pagina word opnieuw geladen. |

## Remove user

Als de gebruiker is ingelogd als admin kan de user users verwijderen. De user drukt op delete en krijgt de vraag weet u het zeker. Als de user op ja drukt wordt de user verwijderd en wordt de user doorgestuurd naar de home pagina. Anders wordt de user niet verwijderd.

|  |  |
| --- | --- |
| Naam | Remove user |
| Versie | 1.0 |
| Actor | Admin |
| Preconditie | Moet ingelogd zijn |
| Scenario | Remove software   1. Drukt op delete 2. Krijgt vraag weet u het zeker 3. Vult ja in 4. User wordt verwijderd |
| Uitzonderingen | User drukt op nee   1. Drukt op delete 2. Krijgt vraag weet u het zeker 3. Vult nee in 4. User wordt niet verwijderd |
| Niet functionele eisen | Het verwijderen moet snel en soepel verlopen |
| Postconditie | Home pagina word opnieuw geladen. |

## Edit software

Als de gebruiker is ingelogd kan de user software aanpassen. De user drukt op edit en wordt doorgestuurd naar edit pagina. De user veranderd de software naam en/of software versie en drukt op opslaan. Als alle velden gevuld zijn wordt de verandering opgeslagen. Anders krijgt de gebruiker een foutmelding en moet de gebruiker het opnieuw proberen.

|  |  |
| --- | --- |
| Naam | Edit software |
| Versie | 1.0 |
| Actor | Gebruiker |
| Preconditie | Moet ingelogd zijn |
| Scenario | Edit software   1. Druk op edit (potloodje) 2. User word doorgestuurd naar edit pagina. 3. Verander de velden die de user wil veranderen 4. Druk op edit |
| Uitzonderingen | User laat een veld leeg   1. Druk op edit (potloodje) 2. User word doorgestuurd naar edit pagina. 3. Verander de velden die de user wil veranderen 4. Laat 1 veld leeg 5. Druk op edit 6. Krijg melding dat alle velden gevuld moeten zijn |
| Niet functionele eisen | Het bewerken moet snel en soepel verlopen |
| Postconditie | User wordt doorgestuurd naar home pagina |

## Edit user (gebruiker)

Als de gebruiker is ingelogd kan de user zijn eigen account data aanpassen. De user drukt op profiel en wordt doorgestuurd naar zijn profielpagina. De user veranderd de usernaam en/of wachtwoord en drukt op opslaan. Als alle velden gevuld zijn wordt de verandering opgeslagen. Anders krijgt de gebruiker een foutmelding en moet de gebruiker het opnieuw proberen.

|  |  |
| --- | --- |
| Naam | Edit user (gebruiker) |
| Versie | 1.0 |
| Actor | Gebruiker |
| Preconditie | Moet ingelogd zijn |
| Scenario | Edit user (gebruiker)   1. Druk op profile knop 2. Verander de username of wachtwoord 3. Druk op opslaan |
| Uitzonderingen | User laat een veld leeg   1. Druk op profile knop 2. Verander de username maar laat wachtwoord leeg 3. Druk op opslaan 4. Geef melding dat alle velden ingevuld moeten zijn |
| Niet functionele eisen | Het bewerken moet snel en soepel verlopen |
| Postconditie | User wordt doorgestuurd naar home pagina |

## Edit user (admin)

Als de gebruiker is ingelogd als admin kan de gebruiker iemand anders zijn account aanpassen. De gebruiker drukt op edit en wordt doorgestuurd naar een aanpas pagina. De gebruiker veranderd de usernaam en/of wachtwoord en drukt op opslaan. Als alle velden gevuld zijn wordt de verandering opgeslagen. Anders krijgt de gebruiker een foutmelding en moet de gebruiker het opnieuw proberen.

|  |  |
| --- | --- |
| Naam | Edit user (admin) |
| Versie | 1.0 |
| Actor | Admin |
| Preconditie | Moet ingelogd zijn |
| Scenario | Edit user (admin)   1. Druk op edit (potloodje) 2. Admin wordt doorgestuurd naar edit pagina. 3. Verander de velden die de admin wil veranderen 4. Druk op edit |
| Uitzonderingen | User laat een veld leeg   1. Druk op edit (potloodje) 2. Admin wordt doorgestuurd naar edit pagina. 3. Verander de velden die de admin wil veranderen 4. Laat 1 veld leeg 5. Druk op edit 6. Krijg melding dat alle velden gevuld moeten zijn |
| Niet functionele eisen | Het bewerken moet snel en soepel verlopen |
| Postconditie | User wordt doorgestuurd naar home pagina |

## Toon data

Als de gebruiker is ingelogd kan de gebruiker data aanvragen bij de database.

|  |  |
| --- | --- |
| Naam | Toon data |
| Versie | 1.0 |
| Actor | Gebruiker |
| Preconditie | Moet ingelogd zijn |
| Scenario | Toon data   1. User gaat naar home pagina 2. Toon data |
| Uitzonderingen |  |
| Niet functionele eisen | De data moet snel geladen worden |
| Postconditie |  |

## Get software update

Als de gebruiker zijn software een update heeft dan krijgt de gebruiker een mailtje. Vervolgens update de user en geeft door aan het systeem dat hij/zij heeft geupdate. User zijn huidige softwareversie wordt geupdate. Als de user niet aangeeft dat hij /zij heeft geupdate update het systeem de huidige softwareversie niet.

|  |  |
| --- | --- |
| Naam | Get software update |
| Versie | 1.0 |
| Actor | Gebruiker |
| Preconditie |  |
| Scenario | Get software update   1. User krijgt mailtje met info dat hij/zij software moet updaten 2. User update 3. User geeft door aan het systeem dat hij/ zij heeft geupdate 4. User zijn huidige software versie geupdate |
| Uitzonderingen | Gebruiker geeft niet aan dat hij/ zij heeft geupdate   1. User krijgt mailtje met info dat hij/zij software moet updaten 2. User update 3. User geeft niet aan dat hij/zij heeft geupdate |
| Niet functionele eisen |  |
| Postconditie | Systeem update user zijn huidige software versie |

## Api data ophalen

Het systeem haalt 1 keer per dag data uit de api. En slaat het op in de database

|  |  |
| --- | --- |
| Naam | Api data ophalen |
| Versie | 1.0 |
| Actor | Systeem |
| Preconditie |  |
| Scenario | Api data ophalen   1. Het systeem haalt 1 keer per dag data uit de api 2. Het systeem slaat de gekregen data op in de database |
| Uitzonderingen | Er zijn geen updates   1. Het systeem haalt 1 keer per dag data uit de api 2. Het systeem slaat de gekregen data op in de database maar er zijn geen veranderingen |
| Niet functionele eisen |  |
| Postconditie |  |

## Controleren of user software update heeft

Het systeem controleert 1 keer per dag of een user zijn huidige software een update heeft.

|  |  |
| --- | --- |
| Naam | Controleren of user software update heeft |
| Versie | 1.0 |
| Actor | Systeem |
| Preconditie |  |
| Scenario | Controleren of user software update heeft   1. Het systeem Vergelijkt de api data met de user zijn software data 1 keer per dag 2. Als er software tussen staat die een update nodig heeft stuur een mailtje |
| Uitzonderingen | Er is geen software update   1. Het systeem Vergelijkt de api data met de user zijn software data 1 keer per dag |
| Niet functionele eisen |  |
| Postconditie |  |

## Mail versturen

Het systeem verstuurt 1 keer per dag een mailtje met info waarin staat welke updates een user moet doen. In het mailtje staat ook een linkje daar kan een user op klikken om aan te geven dat de user zijn software heeft geupdate.

|  |  |
| --- | --- |
| Naam | Mail versturen |
| Versie | 1.0 |
| Actor | Systeem |
| Preconditie |  |
| Scenario | Mail versturen   1. Het systeem verstuurt 1 keer per dag als als er een software update beschikbaar is voor een user 2. User kan vervolgens aangeven dat de user heeft geupdate. 3. De huidige software versie word geupdate. |
| Uitzonderingen | User geeft niet aan dat de user heeft geupdate   1. De huidige software versie word niet geupdate 2. De user blijft mailtjes ontvangen dat hij/ zij moet updaten |
| Niet functionele eisen |  |
| Postconditie |  |

## 

## Paying customer

User wilt meer functionaliteiten en dus wil hij gaan betalen voor het systeem. De user drukt op “buy” en wordt doorgestuurd naar invulformulier. Als de gebruiker alles goed invult krijgt de gebruiker meer rechten op de website.

|  |  |
| --- | --- |
| Naam | Paying customer |
| Versie | 1.0 |
| Actor | Gebruiker |
| Preconditie | Moet in bezit zijn van account en geld |
| Scenario | Gebruiker wil meer functies   1. User drukt op knop “buy” 2. Vult bank gegevens in formulier 3. User drukt op buy |
| Uitzonderingen | Gebruiker laat velden leeg   1. User drukt op knop “buy” 2. Vult bank gegevens in formulier 3. User drukt op buy 4. Systeem geeft foutmelding dat alle velden gevuld moeten zijn |
| Niet functionele eisen | Gebruiker verliest geld |
| Postconditie | User krijgt meer functies op website |

# 

# Interfaces

Hier laat ik zien hoe het systeem er uiteindelijk uit komt te zien. Dat doe ik doormiddel van adobe xd. Het ontwerp is te zien via de link. Verder leg ik de knoppen uit per pagina.

<https://xd.adobe.com/view/1a88d63f-c0f1-4b26-65ec-1ea7fc1e0d87-6cb9/>

1. User login knoppen
   1. Login
      1. Stuurt user door naar home page
         1. Als het een admin wordt hij doorgestuurd naar admin pagina
   2. Sign up
      1. Stuurt user door naar sign up page
2. User sign up knoppen
   1. Sign up
      1. Maakt nieuwe user aan en stuurt user door naar inlogpagina
3. User add software
   1. Add
      1. Voegt ingevulde velden toe aan gebruiker zijn softwarelijst en stuurt door naar home scherm
4. User edit software
   1. Edit
      1. Verandert ingevulde velden van specifiek item en stuurt door naar home scherm
5. Admin add user
   1. Add
      1. Voegt user toe aan database en stuurt door naar home scherm
6. Profile page admin and user
   1. Verandert ingevulde velden van specifiek item en stuurt door naar home scherm
7. Tabellen
   1. Plus
      1. Wordt gebruikt om software toe te voegen aan lijst
   2. Potlood
      1. Wordt gebruikt om huidige data te veranderen
   3. Prullenbak
      1. Wordt gebruikt om huidige data te verwijderen
   4. Oogje
      1. Wordt gebruikt om details te tonen van huidig item
8. Navbar
   1. Huisje
      1. Gaat naar home screen
   2. Poppetje
      1. Gaat naar ingelogde user zijn accountgegevens
   3. Deur
      1. Logt huidige user uit
9. Tripple dot mobiel
   1. Laat drop down zien van alle mogelijkheden.
10. User home screen
    1. Buy
       1. User kan met behulp van deze knop extra functionaliteiten krijgen