**Presentie Informatiesysteem**

Chi-kin, Christian, Jorrit, Matthijs & Pasoon

Groep 2

**Inleiding**

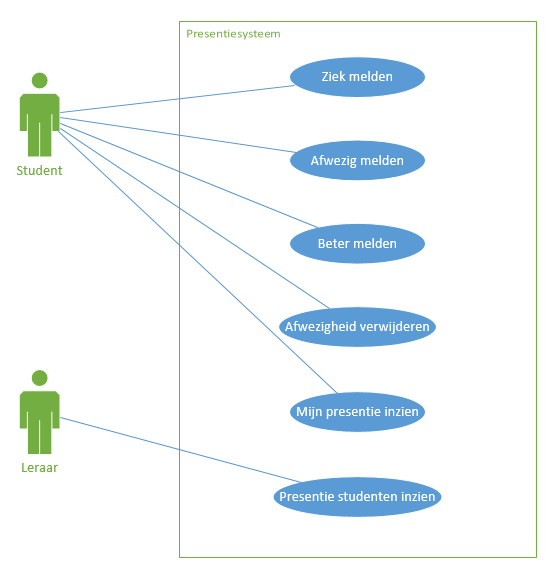
De aanleiding van deze paper is het analyseren en realiseren van een presentie informatiesysteem, die als doel heeft het huidige systeem dat gebruik maakt van ouderwetse presentielijsten, te moderniseren. Als eerst worden de Use Cases in kaart gebracht, die de belangrijkste processen automatiseren. Er wordt gekeken naar welke actoren toegang krijgen tot het systeem en welke acties ze mogen uitvoeren. De Uses Cases worden volledig uitgewerkt in een Use Case Template en op basis hiervan worden er activiteit diagrammen opgesteld. Daarna worden, door middel van een klassendiagram, de klassen die het systeem vormen omschreven. Hieruit kunnen de onderlinge associaties tussen de klassen worden afgeleid. Op basis van dit technisch ontwerp, wordt het presentie informatiesysteem geschreven door gebruik te maken van HTML/Polymer en Java. Ten slotte wordt het proof-of-concept presentie informatiesysteem getest, op basis van vooraf opgestelde testscenario’s.

**Probleemstelling**

De afgelopen jaren is de Hogeschool Utrecht bezig geweest met het verhogen van het studiesucces van haar studenten. In opdracht van OCW is er een onderzoek gedaan naar de relatie tussen studieduur en de tijd die studenten aan de studie besteden. Hieruit is gebleken dat contacttijd een positieve invloed heeft op de betrokkenheid van studenten en dus het studiesucces. Studenten die vaak afwezig zijn leveren namelijk weinig tot geen bijdrage aan het collectieve leerproces. Met presentielijsten wordt de aanwezigheid van studenten bijgehouden. Deze lijsten spelen steeds meer een prominente rol binnen de onderwijsketen van de HU. Helaas hebben deze lijsten nogal veel gebreken. Zo worden de presentielijsten vaak in losse bestanden gemaakt en bijgehouden door verschillende docenten. Ze worden hierdoor niet up-to-date gehouden en kunnen ook niet centraal ingezien worden.

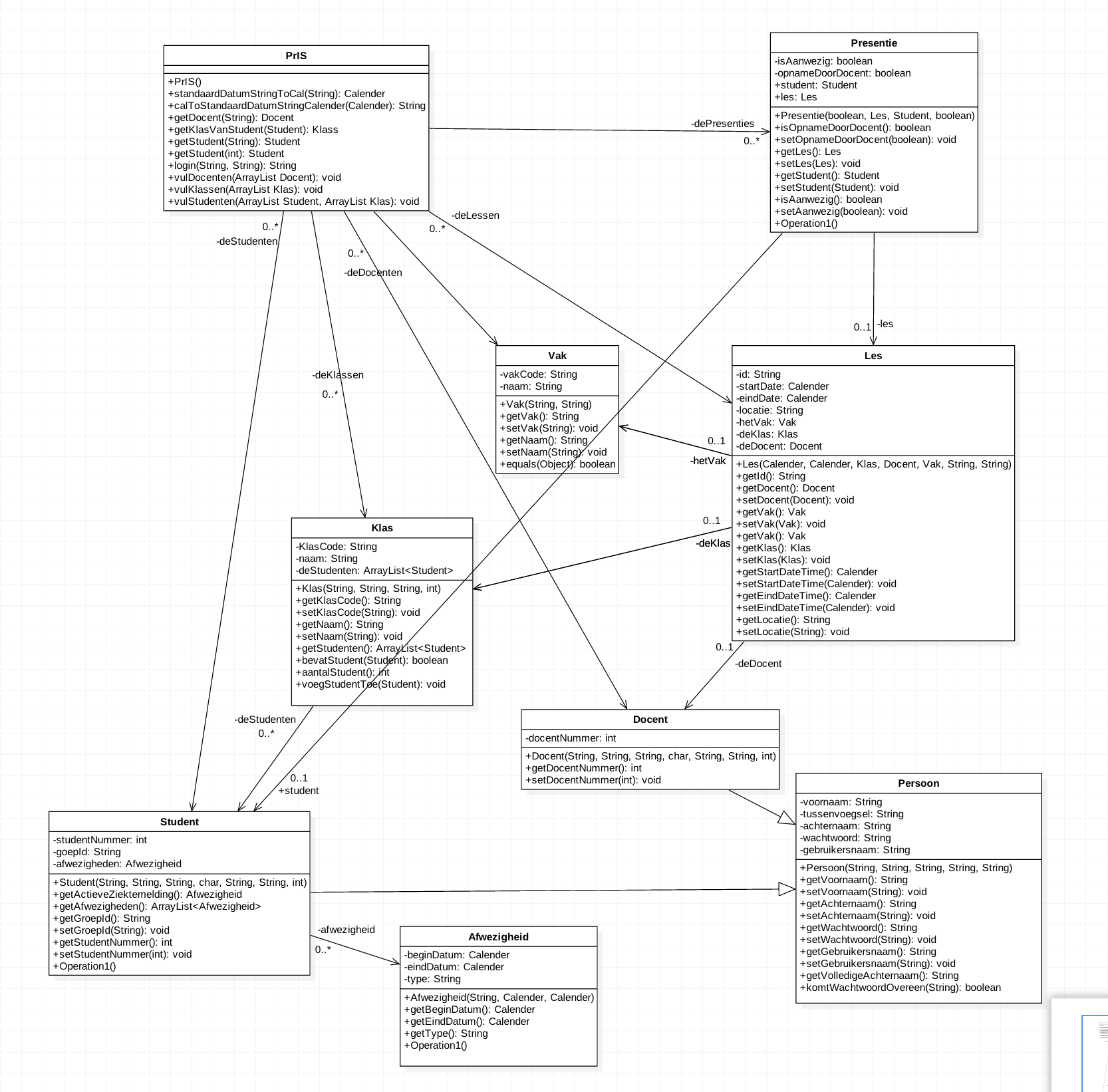
**Aanpak**

Om achter de functionaliteiten van het systeem te komen, zijn er Use Cases opgesteld. Hierbij wordt er gekeken naar de belangrijkste processen van het systeem en welke actor toegang krijgt voor een Use Case. Uit de casus kwamen twee actoren naar voren: de student en de leraar. De Use Cases die naar voren kwamen uit de casus zijn: “ziekmelden”, “afwezig melden”, “beter melden”, “afwezigheid verwijderen” en “mijn presentie inzien” voor de student en “de presentie van studenten inzien” voor de docent. Deze Use Cases zijn uitgewerkt in Use Case Templates en op basis van het scenario zijn er activiteiten-diagrammen opgesteld. De activiteiten die beschreven staan in de activiteiten-diagrammen zijn de stappen die het systeem moet uitvoeren. De volledig uitgewerkte Use Case Templates en activiteit-diagrammen zijn te vinden in de bijlage.



Figuur : de Use Case Diagram

Na het in kaart brengen van de Use Cases hebben we UML-klassendiagrammen gemaakt. Aan de hand van deze klassendiagrammen hebben we de Javacode geschreven van het presentiesysteem.



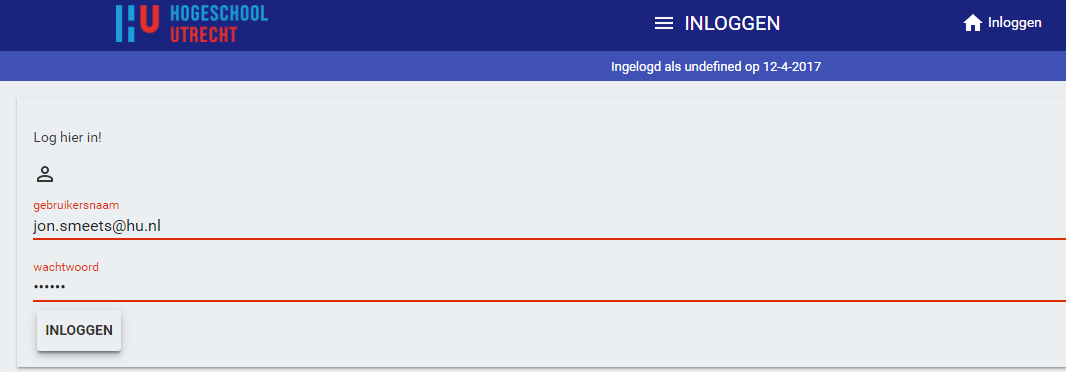
Figuur : UML klassendiagrammen

**Resultaten**

De actor kan inloggen via het inlogscherm. Er kunnen in dit scherm geen andere acties worden uitgevoerd, behalve inloggen en annuleren, zolang de actor niet de juiste gegevens heeft ingevuld.

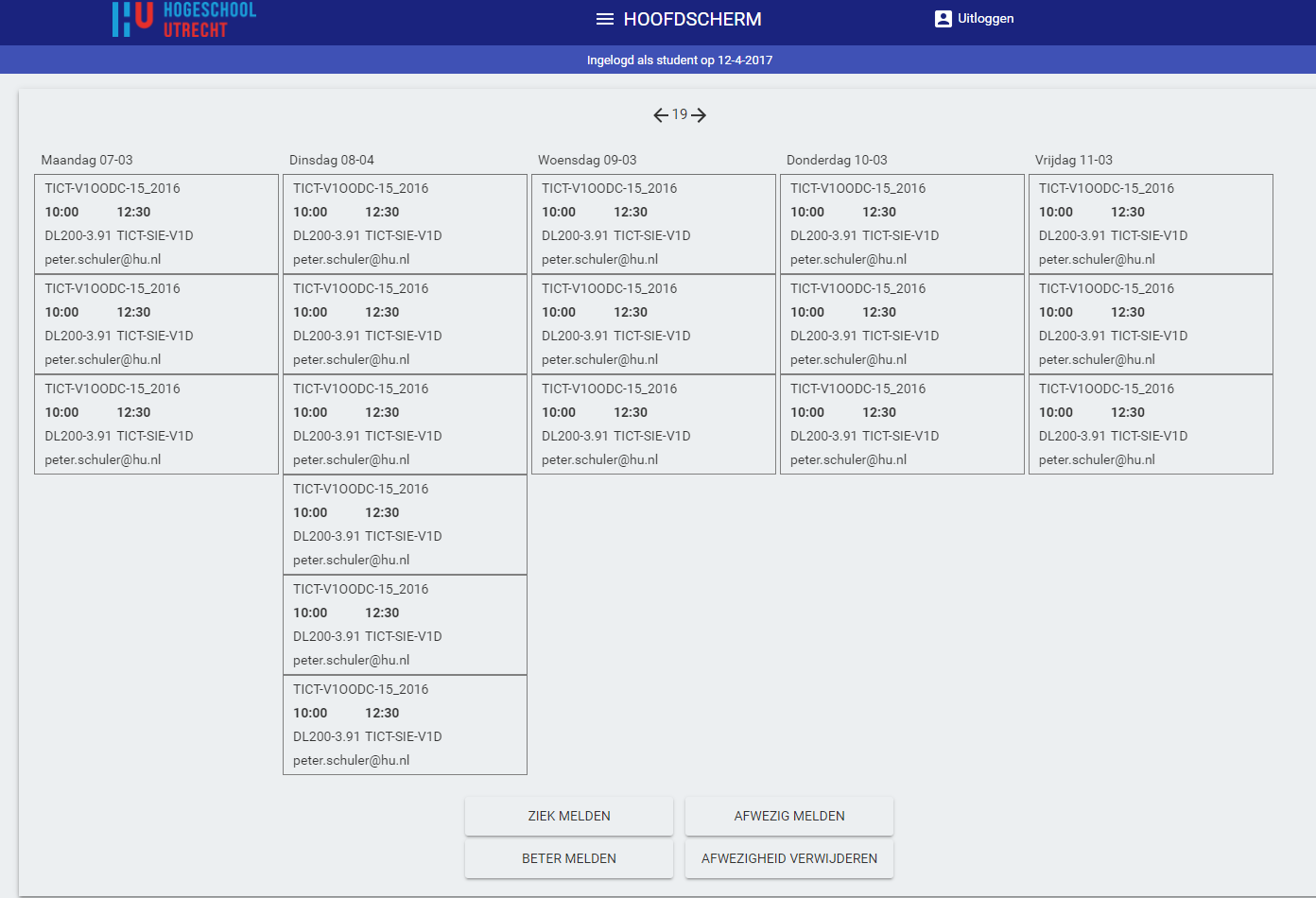


Figuur : Juiste inloggegevens ingevuld



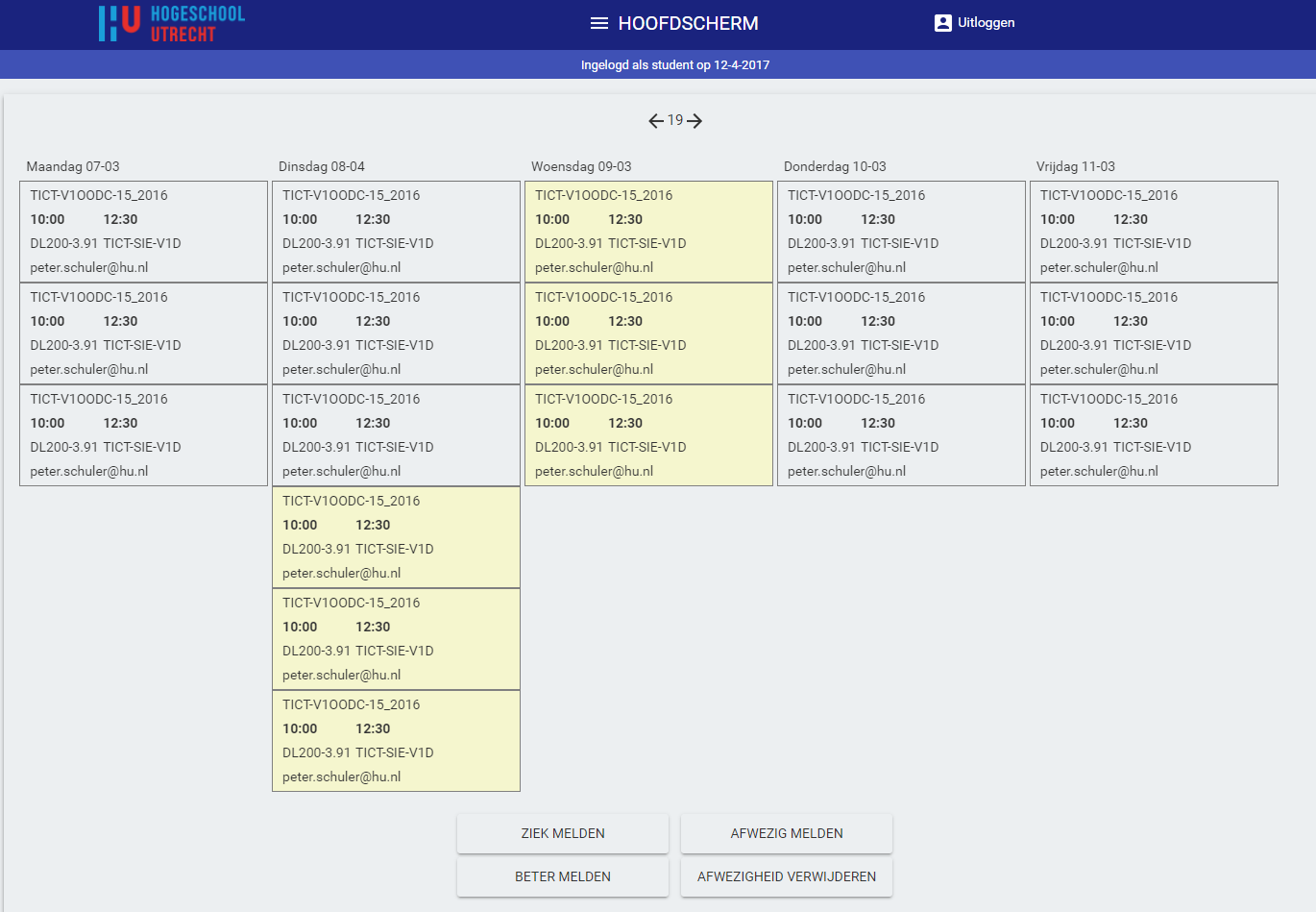
Figuur : Onjuiste inloggegevens ingevuld

Wanneer de actor succesvol is ingelogd, wordt het lesrooster getoond in het hoofdscherm, samen met 4 opties: ziekmelden, afwezig melden, beter melden en afwezigheid verwijderen.



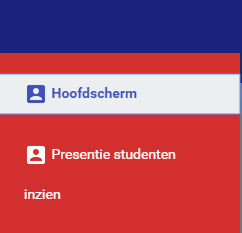
Figuur : Het hoofdscherm voor studenten

De actor kan een actie uitvoeren, door lessen in het rooster te selecteren en om daarna een keuze te maken uit de vier mogelijke opties.

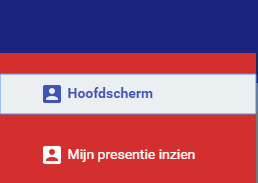


Figuur : Lessen selecteren

Door op de knop naast het hoofdscherm te klikken, wordt er een aparte navigatiemenu getoond. Vanuit hier kan de actor zijn eigen presentie inzien wanneer hij is ingelogd als student of de presentie van studenten inzien wanneer de actor is ingelogd als docent.



Figuur : Actor is ingelogd als docent



Figuur : Actor is ingelogd als student

Aan de hand van de activiteit diagrammen hebben we de testpaden in kaart gebracht. Voor elke Use Case zijn de testscenario’s gemaakt, waarin wordt beschreven op welke manieren het systeem wordt getest. Deze testscenario’s zijn te vinden in de bijlage.

**Conclusie**

Deze opdracht was een grote uitdaging met veel nieuwe en uitdagende elementen voor ons. We hadden problemen om elke hoek met het realiseren van ons project. Echter het bedenken van de applicatie lay-out en elementen ging ons goed af. De front-end en backend zijn afzonderlijk goed gelukt. De communicatie tussen de font-end en backend lukte niet zoals wij hadden gehoopt, wij hadden niet genoeg kennis vergaard om dit deel goed uit te voeren. De communicatie tussen de teamleden had beter gekund maar was niet per definitie slecht. Allemaal vinden wij het jammer dat niet alles is gelukt wat we bedacht hadden.

**Bijlage**

De Use Case Templates:

|  |  |  |
| --- | --- | --- |
| **Use Case nummer** | | UC-1 |
| **Use Case naam** | | Ziekmelden |
| **Actor** | | Student |
| **Samenvatting** | | Het systeem toont het persoonlijke rooster**(o)** en vijf opties:  UC-1: Ziekmelden  UC-2: Afwezig melden  UC-3: Beter melden  UC-4: Afwezigheid verwijderen  De actor selecteert lessen/dagen/weken naar zijn/haar keuze en vervolgens de gewenste actie die uitgevoerd moet worden op deze lessen/dagen/weken.  Of: UC-5: Mijn presentie inzien, dat verder geen interactie heeft met het rooster. |
| **Pre-conditie** | | N.v.t. |
| **Main Scenario** | | 1. Het systeem toont het persoonlijke rooster**(o)**, de vier mogelijke bewerkingen en 'Mijn presentie inzien' 2. A. De actor selecteert de lessen om een bewerking op uit te voeren**(i)**.   B. De actor selecteert de optie om zich ziek te melden.  3. Het systeem zet de geselecteerde lessen op geoorloofd afwezig met als reden: 'ziek'**(o).** |
| **Post-Conditie (Main Scenario)** | | Er zijn (minstens 1) nieuwe afwezigheden opgeslagen. |
| **Alternatief Scenario 1** | · **Trigger:**  · **Scenario:**  · **Post-conditie:** | · Na stap 2B van het main scenario  [De actor is al als ziek gemarkeerd die dag]  · 1. Systeem geeft foutmelding dat de actor al is afgemeld die 2. dag Systeem keert terug naar stap 1 van het main scenario.  · Er zijn geen wijzigingen in het systeem/op het rooster. |
| **Alternatief scenario 2** | · **Trigger:**  · **Scenario:**  · **Post-conditie:** | · Na stap 2B van het main scenario  [(verwacht) tijdsbestek geen datum/ligt in het verleden/einddatum ligt voor begindatum/*te lang*]  · 1. Systeem meldt dat het tijdsbestek niet in orde is.  2. Systeem keert terug naar stap 2 van het main scenario.  · Er zijn geen wijzigingen in het systeem/op het rooster. |
| **Alternatief scenario 3** | · **Trigger:**  · **Scenario:**  · **Post-conditie:** | · Na stap 2B van het main scenario  [Er zijn geen lessen geselecteerd]  · 1. Systeem meldt dat er geen lessen zijn geselecteerd.  2. Systeem keert terug naar stap 2 van het main scenario.  · Er zijn geen wijzigingen in het systeem/op het rooster. |

|  |  |  |
| --- | --- | --- |
| **Use Case nummer** | | UC-2 |
| **Use Case naam** | | Afwezig melden |
| **Actor** | | Student |
| **Samenvatting** | | Het systeem toont het persoonlijke rooster**(o)** en vijf opties:  UC-1: Ziekmelden  UC-2: Afwezig melden  UC-3: Beter melden  UC-4: Afwezigheid verwijderen  De actor selecteert lessen/dagen/weken naar zijn/haar keuze en vervolgens de gewenste actie die uitgevoerd moet worden op deze lessen/dagen/weken.  Of: UC-5: Mijn presentie inzien, dat verder geen interactie heeft met het rooster. |
| **Pre-conditie** | | N.v.t. |
| **Main Scenario** | | 1. Het systeem toont het persoonlijke rooster**(o)**, de vier mogelijke bewerkingen en 'Mijn presentie inzien' 2. A. De actor selecteert de lessen om een bewerking op uit te voeren**(i)**.   B. De actor selecteert de optie om zich afwezig te melden.  3. 1. Het systeem geeft een popup met de vraag een reden te geven(  2. Het systeem zet de 'afwezig gemelde lessen' op 'geoorloofd afwezig' met reden: x **(o).** |
| **Post-Conditie (Main Scenario)** | | Er zijn (minstens 1) nieuwe afwezigheden opgeslagen. |
| **Alternatief Scenario 1** | · **Trigger:**  · **Scenario:**  · **Post-conditie:** | · Na stap 2B van het main scenario  [De actor is al als 'afwezig' gemarkeerd die dag]  · 1. Systeem geeft foutmelding dat de actor al als 'afwezig' staat die dag.  2. Systeem keert terug naar stap 1 van het main scenario.  · Er zijn geen wijzigingen in het systeem/op het rooster. |
| **Alternatief scenario 2** | · **Trigger:**  · **Scenario:**  · **Post-conditie:** | · Na stap 2B van het main scenario  [Er zijn geen lessen geselecteerd]  · 1. Systeem meldt dat er geen lessen zijn geselecteerd.  2. Systeem keert terug naar stap 2 van het main scenario.  · Er zijn geen wijzigingen in het systeem/op het rooster. |

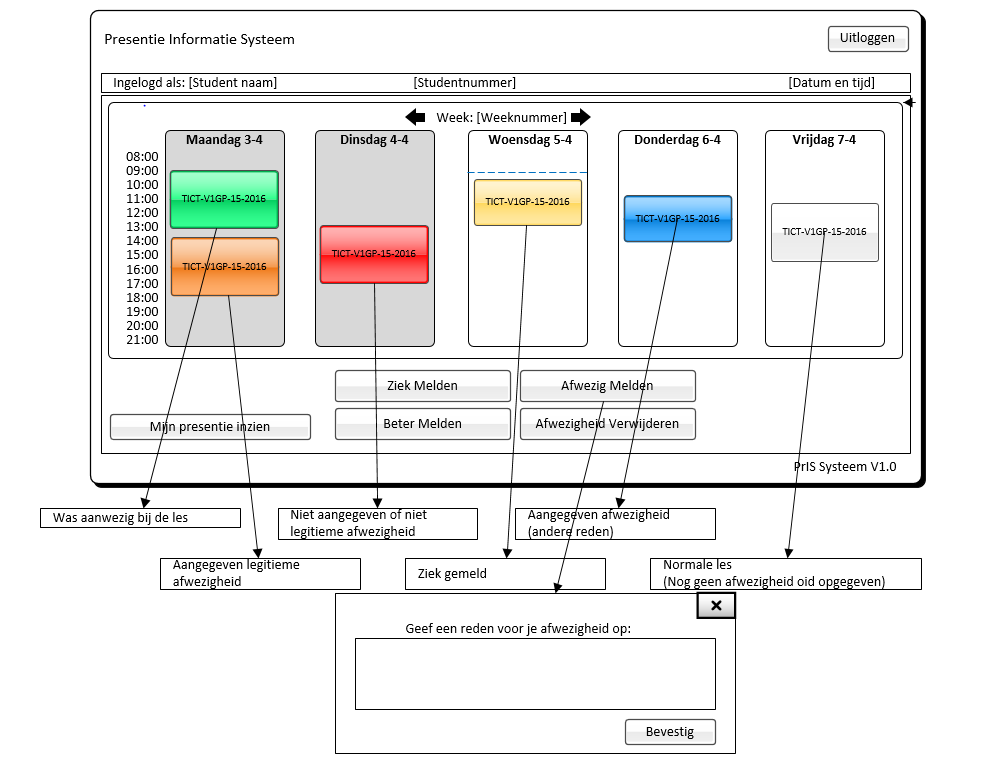
|  |  |  |
| --- | --- | --- |
| **Use Case nummer** | | UC-3 |
| **Use Case naam** | | Beter melden |
| **Actor** | | Student |
| **Samenvatting** | | Het systeem toont het persoonlijke rooster**(o)** en vijf opties:  UC-1: Ziekmelden  UC-2: Afwezig melden  UC-3: Beter melden  UC-4: Afwezigheid verwijderen  De actor selecteert lessen/dagen/weken naar zijn/haar keuze en vervolgens de gewenste actie die uitgevoerd moet worden op deze lessen/dagen/weken.  Of: UC-5: Mijn presentie inzien, dat verder geen interactie heeft met het rooster. |
| **Pre-conditie** | | N.v.t. |
| **Main Scenario** | | 1. Het systeem toont het persoonlijke rooster**(o)**, de vier mogelijke bewerkingen en 'Mijn presentie inzien' 2. A. De actor selecteert de lessen om een bewerking op uit te voeren**(i)**.   B. De actor selecteert de optie om zich beter te melden.  3. Het systeem zet de 'ziekgemelde lessen' op 'aanwezig'**(o).** |
| **Post-Conditie (Main Scenario)** | | Er zijn (minstens 1) nieuwe afwezigheden beter gemeld. |
| **Alternatief Scenario 1** | · **Trigger:**  · **Scenario:**  · **Post-conditie:** | · Na stap 2B van het main scenario  [De actor is al als 'aanwezig' gemarkeerd die dag]  · 1. Systeem geeft foutmelding dat de actor al als 'aanwezig' staat die dag  2. Systeem keert terug naar stap 1 van het main scenario.  · Er zijn geen wijzigingen in het systeem/op het rooster. |
| **Alternatief scenario 2** | · **Trigger:**  · **Scenario:**  · **Post-conditie:** | · Na stap 2B van het main scenario [Er zijn geen lessen geselecteerd]  · 1. Systeem meldt dat er geen lessen zijn geselecteerd.  2. Systeem keert terug naar stap 2 van het main scenario.  · Er zijn geen wijzigingen in het systeem/op het rooster. |

|  |  |  |
| --- | --- | --- |
| **Use Case nummer** | | UC-4 |
| **Use Case naam** | | Afwezigheid verwijderen |
| **Actor** | | Student |
| **Samenvatting** | | Het systeem toont het persoonlijke rooster**(o)** en vijf opties:  UC-1: Ziekmelden  UC-2: Afwezig melden  UC-3: Beter melden  UC-4: Afwezigheid verwijderen  De actor selecteert lessen/dagen/weken naar zijn/haar keuze en vervolgens de gewenste actie die uitgevoerd moet worden op deze lessen/dagen/weken.  Of: UC-5: Mijn presentie inzien, dat verder geen interactie heeft met het rooster. |
| **Pre-conditie** | | N.v.t. |
| **Main Scenario** | | 1. Het systeem toont het persoonlijke rooster**(o)**, de vier mogelijke bewerkingen en 'Mijn presentie inzien' 2. A. De actor selecteert de lessen om een bewerking op uit te voeren**(i)**.   B. De actor selecteert de optie om zijn/haar afwezigheid te verwijderen.  3. Het systeem zet de 'afwezig gemelde lessen' op 'aanwezig'**(o).** |
| **Post-Conditie (Main Scenario)** | | Er zijn (minstens 1) nieuwe afwezigheden present gemeld. |
| **Alternatief Scenario 1** | · **Trigger:**  · **Scenario:**  · **Post-conditie:** | · Na stap 2B van het main scenario  [De actor is al als 'aanwezig' gemarkeerd die dag]  · 1. Systeem geeft foutmelding dat de actor al als 'aanwezig' staat die dag  2. Systeem keert terug naar stap 1 van het main scenario.  · Er zijn geen wijzigingen in het systeem/op het rooster. |
| **Alternatief scenario 2** | · **Trigger:**  · **Scenario:**  · **Post-conditie:** | · Na stap 2B van het main scenario [Er zijn geen lessen geselecteerd]  · 1. Systeem meldt dat er geen lessen zijn geselecteerd.  2. Systeem keert terug naar stap 2 van het main scenario.  · Er zijn geen wijzigingen in het systeem/op het rooster. |

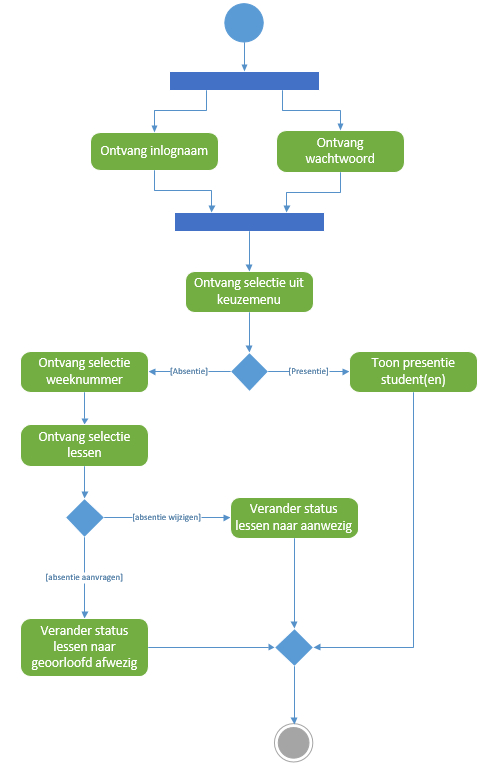
|  |  |  |
| --- | --- | --- |
| **Use Case nummer** | | UC-5 |
| **Use Case naam** | | Mijn presentie inzien |
| **Actor** | | Student |
| **Samenvatting** | | Het systeem toont het persoonlijke rooster**(o)** en vijf opties:  UC-1: Ziekmelden  UC-2: Afwezig melden  UC-3: Beter melden  UC-4: Afwezigheid verwijderen  De actor selecteert lessen/dagen/weken naar zijn/haar keuze en vervolgens de gewenste actie die uitgevoerd moet worden op deze lessen/dagen/weken.  Of: UC-5: Mijn presentie inzien, dat verder geen interactie heeft met het rooster. |
| **Pre-conditie** | | N.v.t. |
| **Main Scenario** | | 1. Het systeem toont het persoonlijke rooster**(o)**, de vier mogelijke bewerkingen en 'Mijn presentie inzien' 2. De actor selecteert de optie 'Mijn presentie inzien'. 3. Het systeem geeft een scherm met grafieken en percentages over de presentie van de actor. |
| **Post-Conditie (Main Scenario)** | | N.v.t. |
| **Alternatief Scenario 1** | · **Trigger:**  · **Scenario:**  · **Post-conditie:** | · Na stap 2 van het main scenario [Het persoon heeft nog geen gegevens over presentie]  · 1. Systeem geeft foutmelding dat de actor nog geen gegevens heeft om te laten zien  2. Systeem keert terug naar stap 1 van het main scenario.  · Er zijn geen wijzigingen in het systeem/op het rooster. |
|  |  |  |

|  |  |
| --- | --- |
| **Use Case nummer** | UC-6 |
| **Use Case naam** | Studenten presentie inzien(Leraar) |
| **Actor** | Leraar |
| **Samenvatting** | Het systeem toont het persoonlijke rooster**(o)** en de optie:  UC-6:Studenten presentie inzien  De actor selecteert krijgt een lijst met vakken tezien waar hij/zij er een van selecteert. Nu ziet de actor een lijst met klassen die dit vak volgen. Zodra de actor een klas selecteert ziet de actor de presentie van alle leerlingen in deze klas voor dit vak. |
| **Pre-conditie** | N.v.t. |
| **Main Scenario** | 1. Het systeem toont het persoonlijke rooster**(o)** en de optie Studeten presentie inzien 2. De actor selecteert de optie 'Studenten presentie inzien'. 3. Het systeem geeft een lijst met vakken 4. De actor selecteert een vak 5. Het systeem geeft een lijst met klassen die dit vak volgen 6. De actor selecteert een klas 7. Het systeem laat een lijst zien van alle leerlingen in de klas en hun presentiepercentage. |
| **Post-Conditie (Main Scenario)** | N.v.t. |

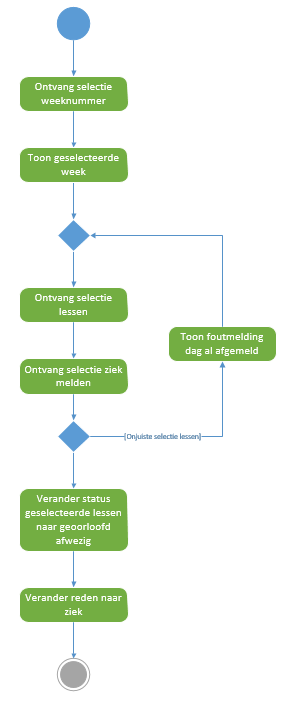
De wireframe:



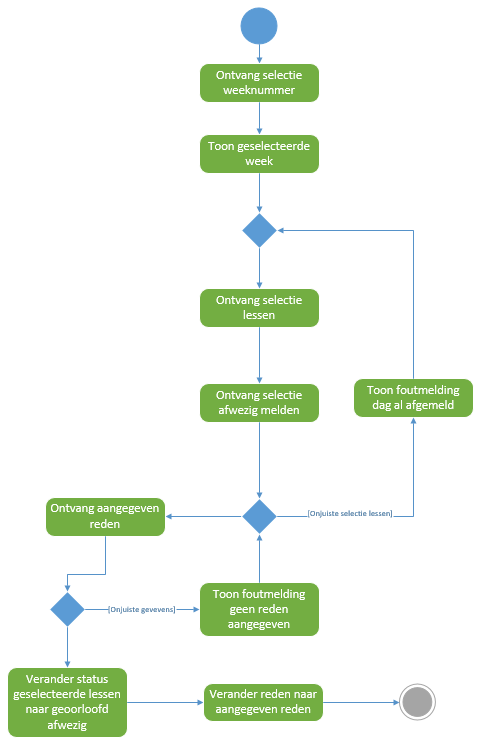
De activiteit diagrammen:



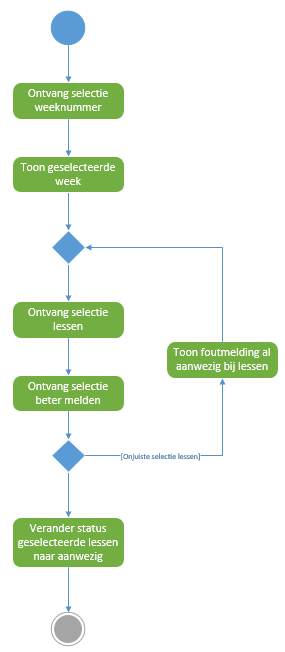
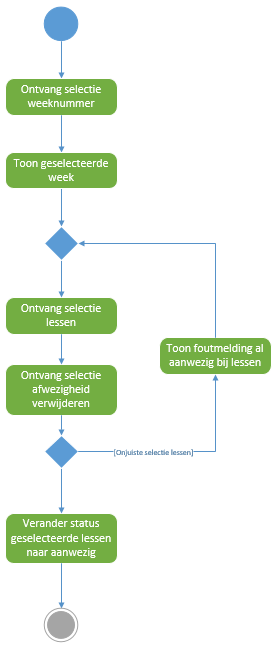
**Ziekmelden:**



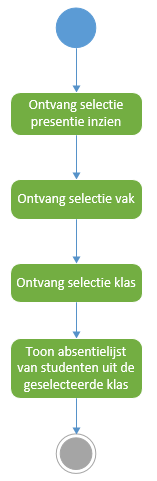
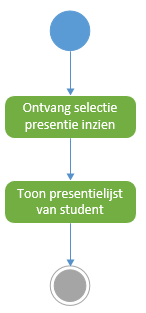
**Afwezigheid Melden:**



**Presentie beter melden: Afwezigheid verwijderen:**

**Mijn presentie inzien: Student presentie inzien(Leraar):**



De testscenario’s:

Activiteiten diagram **ziekmelden (TC 1)**

Procescyclus **ziekmelden**

|  |  |
| --- | --- |
|  | **A** |
| **Inkomend** | **1** |
| **uitgaand** | **2, 3** |

A: 1-2 & 1-3

Testpaden:

Pad 1: 1-3

Pad 2: 1-2

Activiteiten diagram **presentie inzien (TC 2)**





Procescyclus **presentie inzien**

Er is maar 1 pad mogelijk.

Pad 3: 1

Activiteiten diagram **beter melden (TC 3)**

Procescyclus **beter melden**

|  |  |
| --- | --- |
|  | **A** |
| **Inkomend** | **1** |
| **uitgaand** | **2, 3** |

A: 1-2 & 1-3

Testpaden:

Pad 4: 1-3

Pad 5: 1-2

Activiteiten diagram **afwezigheid melden (TC 4)**



Procescyclus **afwezigheid melden**

|  |  |  |
| --- | --- | --- |
|  | **A** | **B** |
| **Inkomend** | **1, 4** | **3** |
| **Uitgaand** | **2, 3** | **4, 5** |

A: 1-2, 1-3, 4-2 & 4-3

B: 3-4 & 3-5

Testpaden:

Pad 6: 1-2-1-3-5

Pad 7: 1-3-4-3-5

Pad 8: 1-3-4-2-1-3-5

Pad 9: 1-3-5

Activiteiten diagram **afwezigheid verwijderen (TC 5)**



|  |  |
| --- | --- |
|  | **A** |
| **Inkomend** | **1** |
| **uitgaand** | **2, 3** |

A: 1-2 & 1-3

Testpaden:

Pad 10: 1-3

Pad 11: 1-2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Testpad pad | Test case NR | Uitgangssituatie | Testwaarden | Verwachte resultaten | Werkelijke resultaten | Test geslaagd |
| 1 | 1 | Rooster/ hoofdscherm  opgestart. | Actor selecteert een of meerdere lessen en meld zich hiervoor af. | De naam van de student + de lessen waar hij zich voor ziek meld worden in een CSV bestand geschreven. De vakken waarvoor hij zich ziekmeld worden geel. | Zoals verwachten resultaten | X |
| 2 | 1 | Rooster/ hoofdscherm  opgestart. | Actor selecteert een dagdeel waarvoor hij al is afgemeld. | Systeem toont een foutmelding en voert geen wijzigingen door. Stuurt actor terug naar het hoofdscherm. | Er gebeurt niks. | X |
| 3 | 2 | Rooster/ hoofdscherm  opgestart. | - | Actor ziet kleuren op lessen. | Zoals verwachte resultaten. | X |
| 4 | 3 | Rooster/ hoofdscherm  opgestart. | Actor selecteert enkele lessen en meld zich hiervoor beter. | De naam van de student + de lessen waar hij zich voor afmeld + de reden worden uit het CSV bestand geschreven. De kleur verdwijnt weer. | Zoals verwachte resultaten. | X |
| 5 | 3 | Rooster/ hoofdscherm  opgestart. | Actor selecteert een dagdeel waarvoor hij al aanwezig is. | Systeem toont een foutmelding en voert geen wijzigingen door. Stuurt actor terug naar het hoofdscherm. | Er gebeurt niks. | X |
| 6 | 4 | Rooster/ hoofdscherm  opgestart. | Actor meld zich af voor een dagdeel waarvoor hij al is afgemeld. | Systeem toont een foutmelding en voert geen wijzigingen door. Stuurt actor terug naar het hoofdscherm. | Er gebeurt niks. | X |
| 7 | 4 | Rooster/ hoofdscherm  opgestart.  Scherm geef reden op is opgestart. | Geen reden opgeven. | Systeem toont foutmelding en stuurt actor terug naar scherm “geef reden op”. | Er gebeurt niks. | X |
| 8 | 4 | Rooster/ hoofdscherm  opgestart. | Zie testpad pad 6+7. | Zie testpad pad 6+7. | Er gebeurt niks. | X |
| 9 | 4 | Rooster/ hoofdscherm  opgestart. | Actor selecteert dag waarop hij nog niet afgemeld is + een reden. | De naam van de student + de lessen waar hij zich voor afmeld + de reden worden in een CSV bestand geschreven. De lessen worden blauw. | Zoals verwachte resultaten. | X |
| 10 | 5 | Rooster/ hoofdscherm  opgestart. | Actor selecteert dagdeel waarop hij is afgemeld en selecteert afwezigheid verwijderen. | De naam van de student + de lessen waar hij zich voor afmeld + de reden worden uit het CSV bestand geschreven. De blauwe kleur verdwijnt weer. | Zoals verwachte resultaten. | X |
| 11 | 5 | Rooster/ hoofdscherm  opgestart. | Actor selecteert dagdeel waarop hij al aanwezig is. | Systeem toont foutmelding en stuurt actor terug naar hoofdscherm. | Er gebeurt niks. | X |