

Front End Programming

Functioneel Ontwerp

Niels van Brakel, Robin van Vlijmen, Soufian Tichattibin

31 mei 2019 - v1.0

Versiebeheer

Versie	Datum	Auteur	Wijzigingen
1.0	31-05-2019	Iedereen	Eerste versie

Inhoudsopgave

Versiebeheer	1
Inhoudsopgave	2
Inleiding	3
Life-Cycle / Casus	3
Request	4
Item	5
Subject	6
User	6
Use Case Diagram	7
Domein klassendiagram	10
Volledigheids Analyse	11
Actoren	13
Sitemap	15
Use Case 1: Inloggen/uitloggen	16
Use Case 2: Item aanvragen	18
Use Case 3: Aanvragen inzien	21
Use Case 4: Item terugbrengen	23
Use Case 5: Beschikbare items inzien	24
Use Case 6: Aanvraag annuleren	26
Use Case 7: Item reserveren	28
Use Case 8: Verzoek behandelen	30
Use Case 9: Uitgeleende items inzien	32
Use Case 10: Gebruiker registreren	34

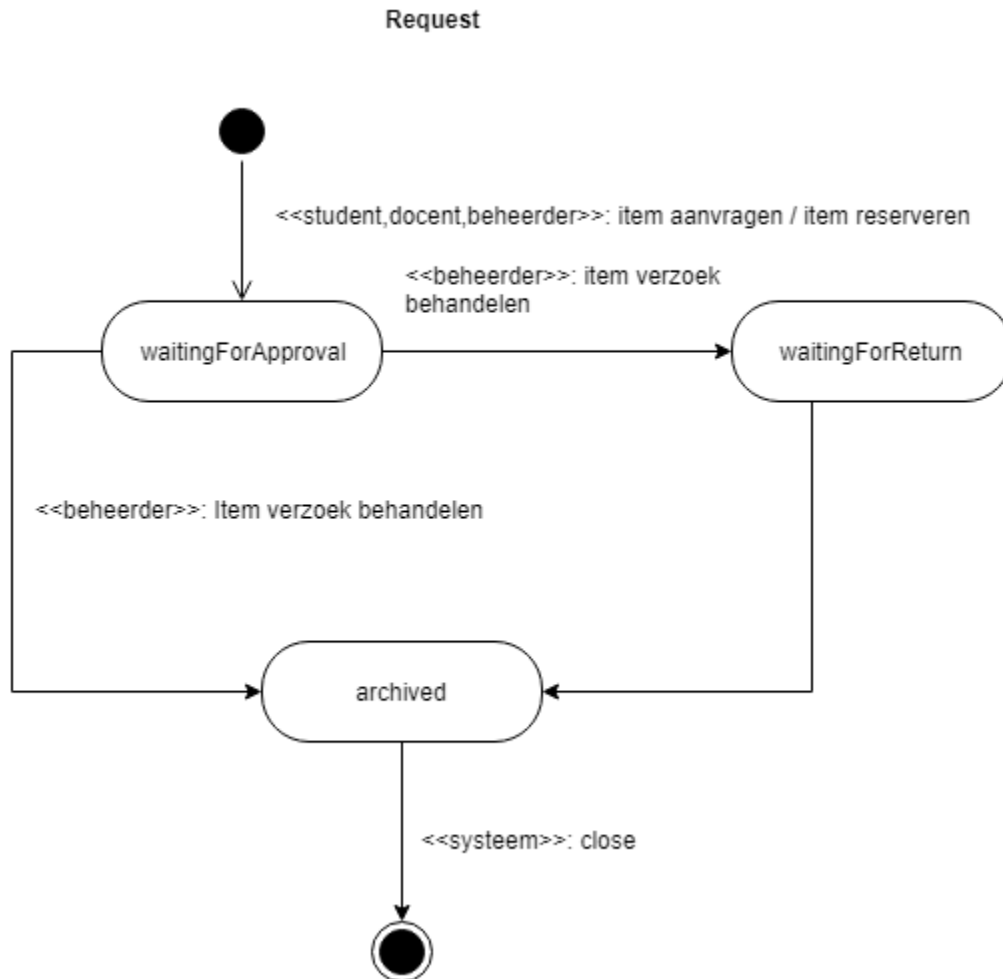
Inleiding

Voor het vak Front End Development (FEP) moeten wij als projectgroep een project verwezenlijken gebruikmakende van Angular en Scrum. Dit document laat zien dat wij de benodigde kennis hebben om correcte functionele documentatie op te leveren.

Life-Cycle / Casus

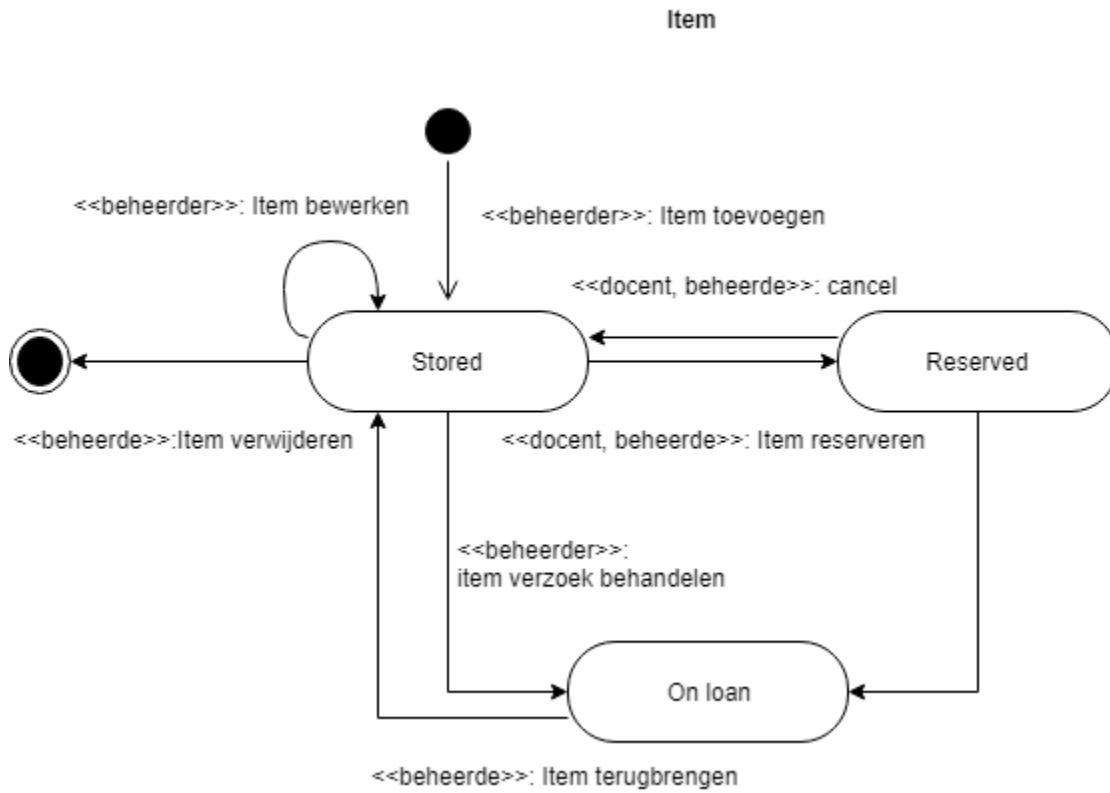
Tim le Febre, een docent van de SNE afstudeerrichting beheerd naast het SNE lab ook hardware die door studenten geleend kunnen worden. In het verleden was dat met name hardware die binnen de SNE afstudeerrichting zelf gebruikt werd, maar de laatste jaren kwam al snel hardware van andere opleidingen erbij (denk bijvoorbeeld aan de Raspberry Pi sets, IoT devices zoals Eddystone Beacons, ...). Vanwege deze groei is er behoefte aan een (simpele) app voor Tim als ook voor de studenten ter ondersteuning voor het uitleen proces.

Request



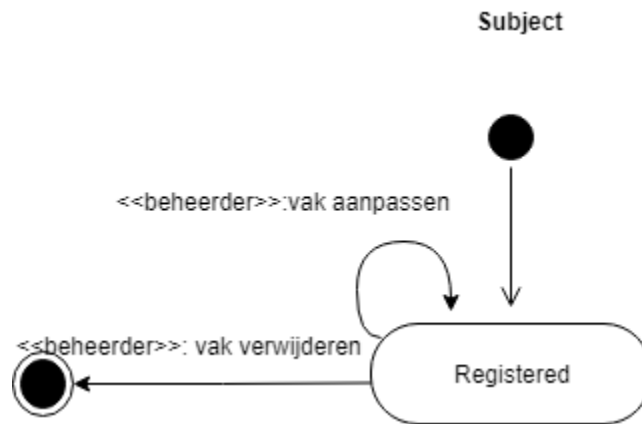
Figuur 1: Request state diagram

Item



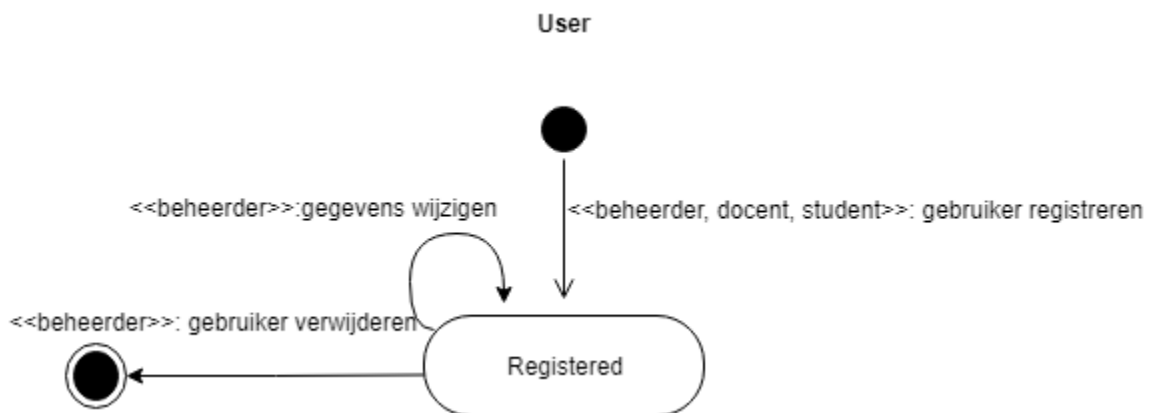
Figuur 2: item state diagram

Subject



Figuur 3: Subject state diagram

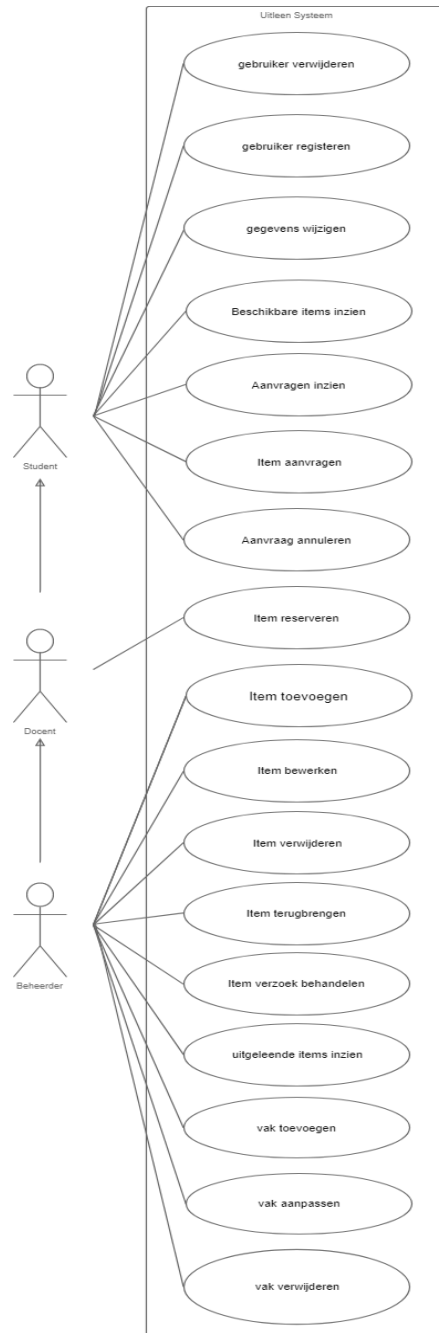
User



Figuur 4: User state diagram

Use Case Diagram

Op basis van de analyse van het vorige hoofdstuk (Life-Cycle / Casus) zijn wij tot het volgende use case diagram gekomen:



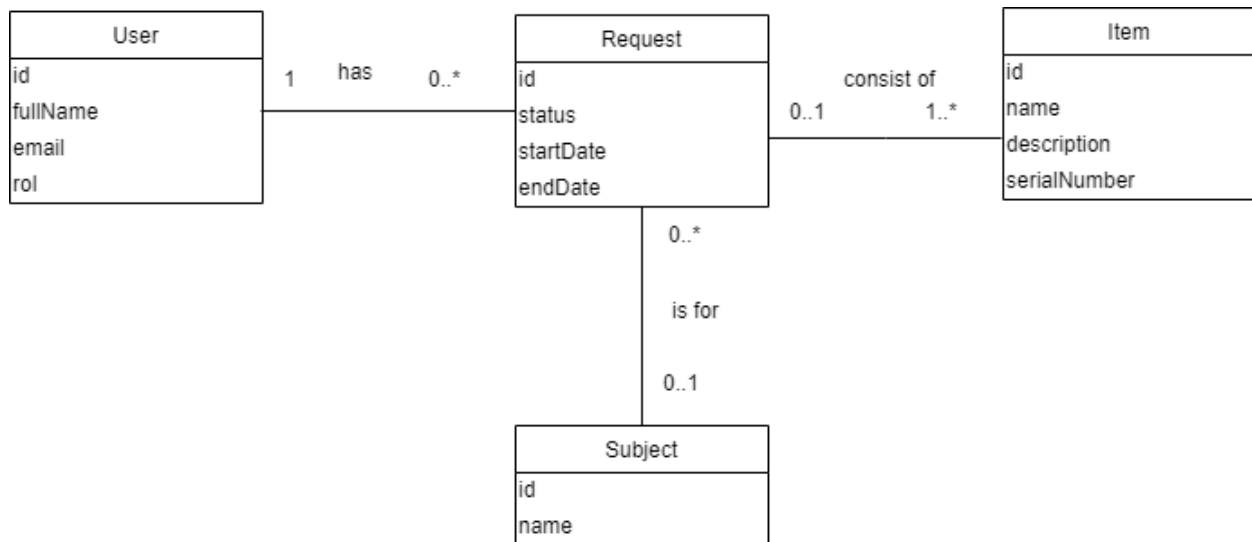
Figuur 5: Use Case Diagram

Use Case	Actor	Samenvatting
Beschikbare items inzien	Student, Docent, Beheerder	Het systeem toont de gebruiker alle items die nog niet uitgeleend zijn. Gereserveerde items horen hier ook bij maar daar staat ook bij dat het gereserveerd is (aanvraag status). (out) Verder wordt er per item de naam (out) en omschrijving (out) van het item getoond.
Aanvragen inzien	Student, Docent, Beheerder	Het systeem toont de gebruiker een lijst aanvragen (bestaande uit de namen van de items (out), status van de aanvraag (out), startdatum van aanvraag(out), einddatum van de aanvraag(out), naam van de aanvrager (out) en het serienummer (out)).
Item terugbrengen	Beheerder	Dit is een vervolg de use case Aanvragen inzien. Bij aanvragen waar de status "waitingForReturn" is. Het systeem geeft de gebruiker de mogelijkheid om een vooraf gekozen aanvraag terug te brengen.
Item aanvragen	Student, Docent, Beheerder	Dit is een vervolg op de use case Beschikbare items inzien. Het systeem toont de gebruiker de mogelijkheid om een item aan te vragen. Hierbij vraagt het systeem de actor om een hoeveelheid (in), startdatum van aanvraag (in), einddatum van aanvraag (in) en om een bevestiging.
Aanvraag annuleren	Student, Docent, Beheerder	Dit is een vervolg op de use case Aanvragen inzien. Bij aanvragen waar de status "waitingForApproval" geeft het systeem de mogelijkheid de aanvraag te verwijderen.
Item reserveren	Docent, Beheerder	Dit is een vervolg op de use case Beschikbare items inzien. Het systeem toont de gebruiker de mogelijkheid om een item te reserveren. Hierbij vraagt het systeem de actor om een hoeveelheid (in), startdatum van aanvraag (in), einddatum van aanvraag (in), een lijst van vakken (bestaande uit naam(out)) waaruit een moet worden gekozen (in) en om een bevestiging.
Verzoek behandelen	Beheerder	Het systeem toont een overzicht van alle aanvragen (bestaande uit naam van de item (out), naam van de aanvrager (out), startdatum van de aanvraag (out), einddatum van de aanvraag (out), de hoeveelheid (out)) die status "waitingForApporval" hebben, de actor kan deze kan goed of afkeuren.
Item toevoegen	Beheerder	Het systeem zal de gebruiker om naam (in) en een omschrijving(in) vragen, deze attributen zijn verplicht. Tevens zal het systeem de mogelijkheid bieden om een serienummer(in) toe te voegen. de actor heeft ook de mogelijkheid om meerdere items in een keer toe te voegen.
Item bewerken	Beheerder	Het systeem zal de gebruiker de mogelijkheid bieden om attribuutwaarden van een item te veranderen.
Item verwijderen	Beheerder	Het systeem laat een overzicht zien van alle items die er zijn, de actor kan een selectie maken en de items verwijderen uit het systeem, indien deze niet uitgeleend zijn.

Uitgeleende items inzien	Beheerder	De actor krijgt een overzicht van alle items die op dit moment als uitgeleend geregistreerd zijn (bestaan uit de naam van de item(out), naam van lener (out), startdatum van de lening(out) de einddatum van de lening(out), serienummer (out)
Vak toevoegen	Beheerder	De actor vult naam (in) en het systeem creëert het vak.
Vak aanpassen	Beheerder	De actor krijgt een overzicht van bestaande vakken(out) en de actor kiest het vak dat hij wil wijzigen, hierna voert hij de wijzigingen door en slaat hij deze op.
Vak verwijderen	Beheerder	De actor krijgt een overzicht van bestaande vakken(out) en selecteert het vak dat hij wil verwijderen, hierna verwijderd hij het vak.
Gebruiker registreren	Beheerder, Docent, Student	Het systeem toont de gebruiker een formulier waarin hij zijn gegevens kan invoeren bestaande uit: naam(in) en email(in) , hierna wordt het account aangemaakt.
Gegevens aanpassen	Beheerder, Docent, Student	Het systeem toont de gebruiker zijn gegevens (bestaande uit naam(out), email(out) en zijn rol (out)) en geeft daarbij de mogelijkheid om de naam(in) en email(in) aan te passen.
Gebruiker verwijderen	Beheerder	Het systeem toont de gebruiker een overzicht van alle gebruikers (bestaande uit naam(out) en email (out)) en geeft daarbij de mogelijkheid om een gebruiker te verwijderen (in).

Figuur 6: Use Case Samenvattingen

Domein klassendiagram



Figuur 7: Domein klassendiagram

User:

Dit object wordt gebruikt om de gegevens van de gebruikers op te slaan. Hierdoor kan er binnen de applicatie ook onderscheid worden gemaakt tussen de rollen (voor autorisatie).

Request:

Dit object wordt gebruikt om de aanvragen die door de gebruikers gemaakt wordt op te slaan. Hiervoor is er een verbinding met het user object om de aanvrager te identificeren, een verbinding met subject om het bijbehorende vak aan te geven als het een reservering is en tenslotte item om naar het geleende/aangevraagde item te verwijzen.

Subject:

Dit object wordt gebruikt om een vak op te slaan die gebruikt wordt als een docente rol een reservering maakt.

Item:

Dit object bevat de items die aangevraagd kunnen worden in de applicatie. Hierbij kan er optioneel een serienummer worden opgegeven als het over waardenvolle materiaal gaat.

Volledigheids Analyse

Attributen check

Naam attribuut	Type	Voorkomende Use Cases
(Item) name	Tekst	<ul style="list-style-type: none"> - Beschikbare items inzien - Aanvragen inzien - Item toevoegen - Item bewerken - Uitgeleende items inzien - Verzoek behandelen
(Item) description	Text	<ul style="list-style-type: none"> - Beschikbare items inzien - Item toevoegen - Item bewerken
(Request) status	Text	<ul style="list-style-type: none"> - Beschikbare items inzien - Aanvragen inzien - Item terugbrengen - Aanvraag annuleren - Verzoek behandelen - Uitgeleende items inzien
(Item) serialNumber	Text	<ul style="list-style-type: none"> - Aanvragen inzien - Item toevoegen - Item bewerken - Uitgelezen items inzien
amount	Nummer	<ul style="list-style-type: none"> - Item aanvragen - Item reserveren - Verzoek Behandelen
(Request) startDate	Datum	<ul style="list-style-type: none"> - Aanvragen inzien - Item aanvragen - Item reserveren - Verzoek behandelen - Uitgeleende items inzien.
(Request) endDate	Datum	<ul style="list-style-type: none"> - Aanvragen inzien - Item aanvragen - Item reserveren

		<ul style="list-style-type: none"> - Verzoek behandelen - Uitgeleende items inzien.
(Subject) name	Text	<ul style="list-style-type: none"> - Item reserveren - Vak toevoegen - Vak verwijderen
(User) name	Text	<ul style="list-style-type: none"> - Gebruiker registreren - Gegevens aanpassen - Gebruiker verwijderen - Uitgeleende items inzien - Verzoek behandelen - Aanvragen inzien
(User) email	Text	<ul style="list-style-type: none"> - Gegevens aanpassen - Gebruiker verwijderen - Gebruiker registreren
(User) rol	Text	<ul style="list-style-type: none"> - Gegevens aanpassen

Figuur 8: Attributen check

Functionaliteiten check

Uit ons Use Case diagram kunnen we ontnemen dat er drie actoren zijn. Dit vereist dat het te realiseren systeem weet door welke actor het bediend wordt. Een oplossing voor deze issue is het realiseren van een dedicated systeem per onderkende actor, de andere oplossing voor deze issue is het uitbreiden van het systeem met een use case 'inloggen'.

We kiezen hier voor het toevoegen van de use case 'inloggen'. Een use case die deel uitmaakt van de superuser 'anonieme gebruiker', een user van wie alle users erven en die we expliciet niet in het Use Case Diagram hebben opgenomen.

Het toevoegen van de use case inloggen heeft echter ten gevolge dat we niet weten welke rol een gebruiker heeft die ingelogd is. Deze issue lossen we voor nu op door de in het systeem bekende users en hun rol hardcoded in het systeem te implementeren.

Conclusie

De applicatie zal de drie rollen (student, docent en beheerder) ondersteunen samen met de bovengenoemde use cases in de komende iteraties. Hierbij zal er een hoge prioriteit worden gegeven aan de use cases die met de core van de applicatie belang hebben.

Actoren

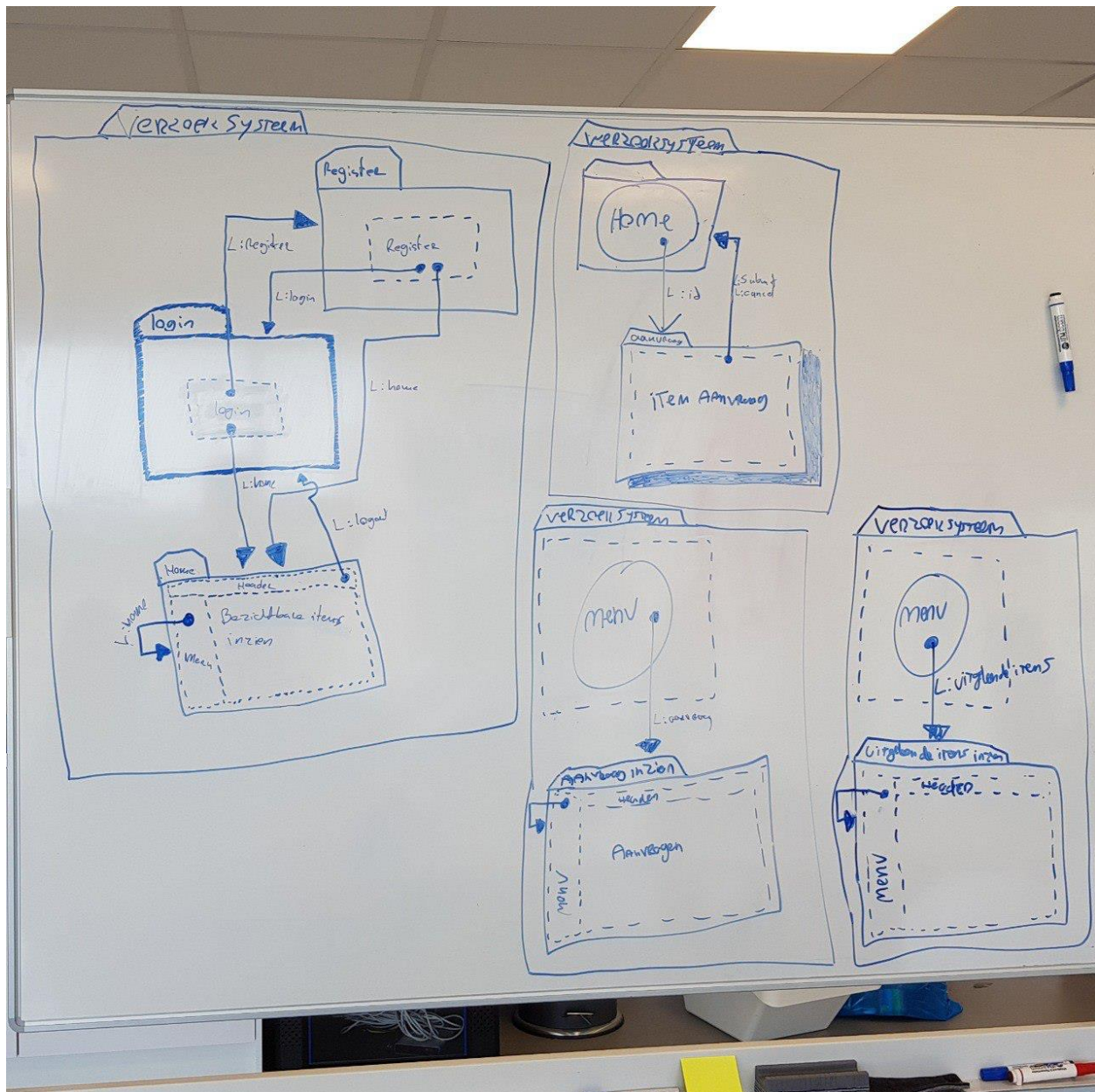
Uit het Use Case Diagram van het vorige hoofdstuk kunnen de we volgende actoren extraheren: Student, Docent en Beheerder. Hierbij geldt dat een Beheerder alles mag wat een Docent mag en een Docent alles mag wat een Student mag.

Project	Uitleensysteem	
Rolnaam	Student	
Rolbeschrijving	Een student kan erg verschillend zijn in gedrag en ze kunnen zich specialiseren op ieder onderwerp wat er wordt aangeboden door de school waar ze studeren. In het algemeen heeft een student soms items nodig voor projecten of lessen die hij/zij zelf niet tot de beschikking heeft. In het uitleensysteem kan vervolgens gekeken worden of het beschikbaar is en geleend kan worden. Buiten het lenen kan een student zijn aanvragen inzien, annuleren en natuurlijk zijn gegevens bewerken.	
Personen	Naam	Niels van Brakel
	Email	niels.vanbrakel@student.hu.nl
	Functie	student
	Aantekening	

Project	Uitleensysteem	
Rolnaam	Docent	
Rolbeschrijving	Een docent geeft les aan een groep studenten om ze kennis bij te brengen, hiervoor moet hij soms een zekerheid hebben dat items beschikbaar zijn voor een bepaalde les. Hiervoor heeft hij buiten alles wat een student mag ook de mogelijkheid om items te reserveren.	
Personen		
	Naam	Robin van Vlijmen
	Email	robin.vanvlijmen@student.hu.nl
	Functie	Docent
	Aantekening	

Project	Uitleensysteem	
Rolnaam	Beheerder	
Rolbeschrijving	De beheerder wie meestal een docent is die het als een extra taak op zich heeft genomen heeft de belangrijkste rol binnen de applicatie. Hij/Zij beheert namelijk alle items en kan ze dus toevoegen, aanpassen of verwijderen. Dat geldt ook voor de vakken. Hij is ook de enige die items kan terugbrengen of een account kan verwijderen.	
Personen		
	Naam	Soufian Tichattibin
	Email	soufian.tichattibin@student.hu.nl
	Functie	Beheerder
	Aantekening	

Sitemap



Figuur 9: Sitemap

Use Case 1: Inloggen/uitloggen

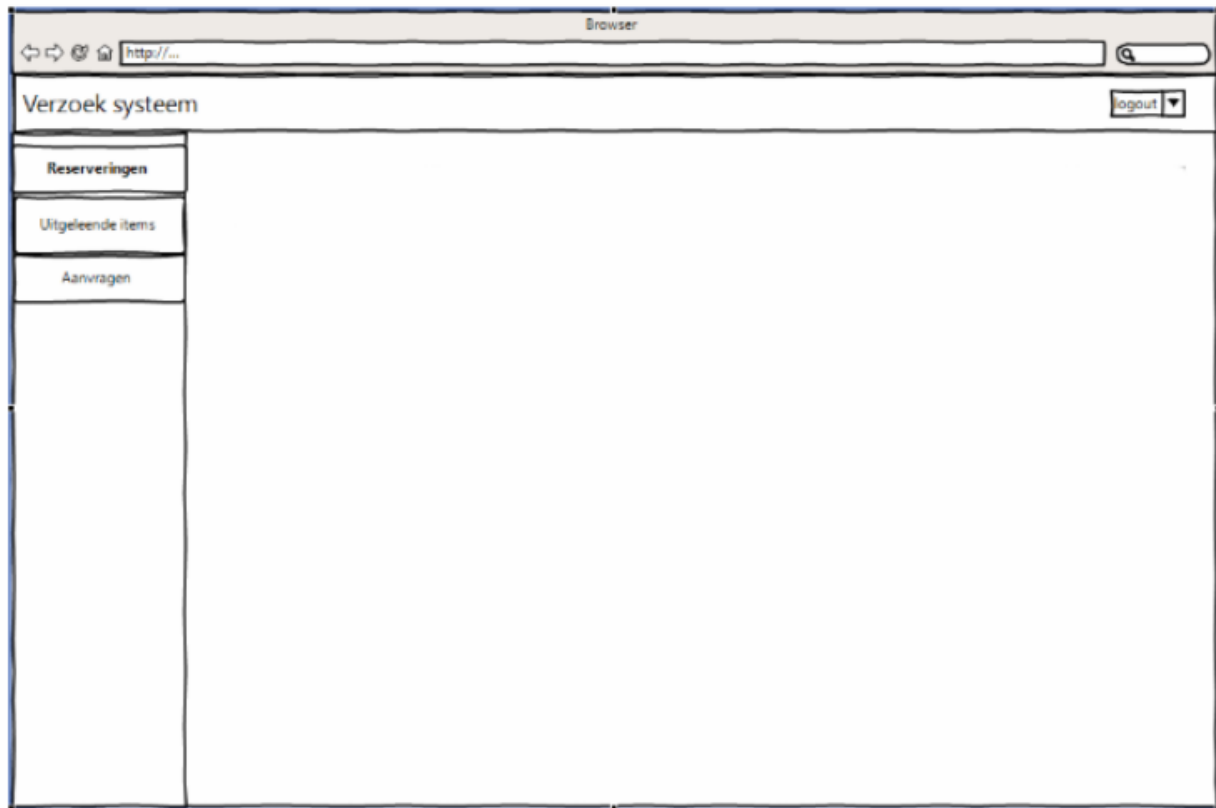
Dit is een functionaliteit die vereist is, maar die niet verder uitgewerkt zal worden. Vandaar dat er alleen de wireframes gegeven zullen worden om deze functionaliteit toe te lichten. Omdat we vereisen dat de actor zich identificeert, zal als de actor nog niet in het systeem bekend is, een scherm getoond worden die de actor in staat stelt om zich in te loggen. (Zie figuur 10)

Dit maakt geen deel uit van het Functioneel Ontwerp. Technisch zal deze use case door een extern systeem (firebase) worden afgehandeld.

The image shows a hand-drawn wireframe of a login page. It is enclosed in a rectangular frame representing a browser window. At the top of the frame is a header bar labeled "Browser". Below the header is a search bar containing the text "http://...". In the center of the page is a smaller rectangular box titled "Login". Inside this box, there are two input fields: "Email address" with the value "Jan@jan.nl" and "Password" with three asterisks "***". Below these fields are two buttons: "Login" and "Register".

Figuur 10: Wireframe 1-1 Inlog pagina

Nadat de actor bekend is in het systeem is er in ieder scherm een menu balk verwerkt waar rechtsboven een log uit knop staat. Zodra hier op geklikt wordt wordt de actor weer uitgelogd. (Zie figuur 11)



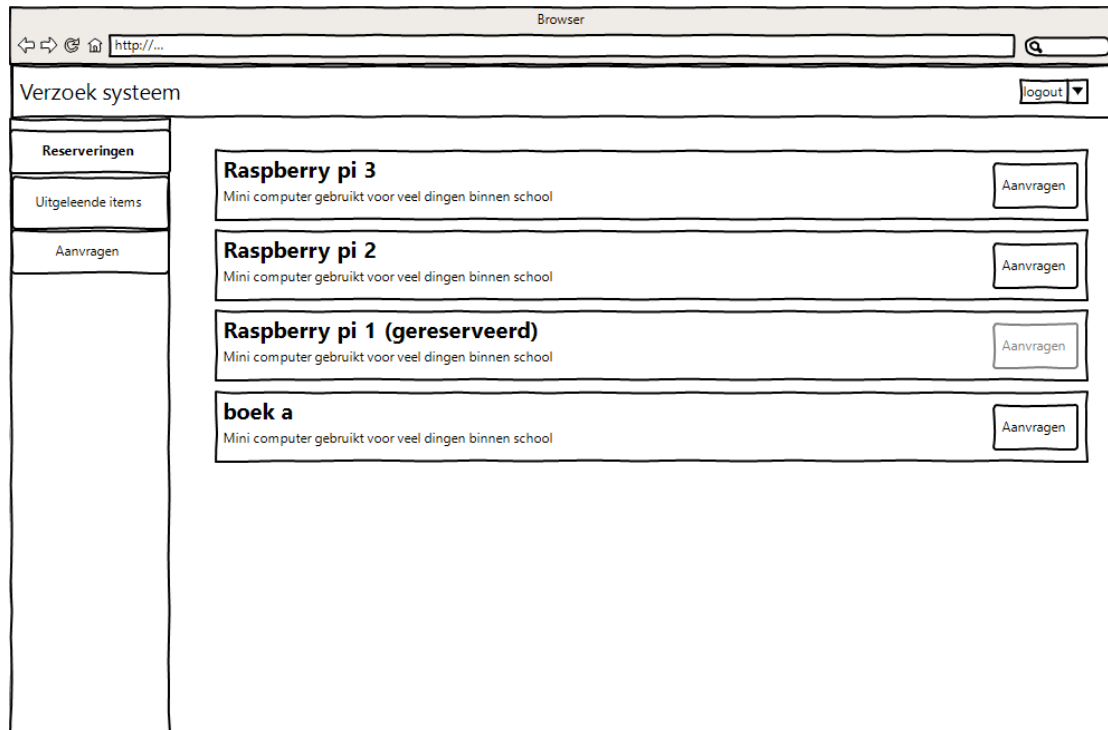
Figuur 11: Wireframe 1-2 Menubalk

Use Case 2: Item aanvragen

Dit is een vervolg op de use case Aanvragen inzien. Het systeem toont de gebruiker de mogelijkheid om een item aan te vragen. Hierbij vraagt het systeem de actor om een hoeveelheid (in) en om een bevestiging.

Use Case Nr.:	UC-2	
Use Case Naam:	Item Aanvragen	
Actoren:	Student, Beheerder, Docent	
Samenvatting:	De systeem heeft toont een optie om een item aan te vragen, wanneer de actor hier op klikt moet deze alle informatie invoeren benodigd om een item aan te vragen: Einddatum lening(in) en begindatum lening(in). Na het doorgeven van deze informatie wordt de aanvraag gedaan.	
Pre Conditie:	De actor is ingelogd, er zijn items beschikbaar om uit te lenen.	
Main Scenario:	<ol style="list-style-type: none"> 1. Het systeem haalt alle items op die uitgeleend kunnen worden. 2. De actor kiest een van de items. 3. Het systeem vraagt om gegevens die nodig zijn om een aanvraag in te dienen, start(in) en einddatum(in). 4. De actor geeft gevraagde gegevens in. Start(in) en einddatum(in) 5. het systeem vraagt de actor deze te bevestigen. 6. De actor bevestigt zijn invoer. 7. Het systeem maakt een aanvraag aan. <i>Usecase beëindigd.</i>	
Post Conditie:	Er staat een nieuwe aanvraag in het systeem.	
Alternatief Scenario 1:	m.b.t. stap 1 van het main scenario.	
	Trigger:	Er zijn geen items die uitgeleend kunnen worden.
	Scenario:	<ol style="list-style-type: none"> 1. Het systeem toont een melding dat er geen beschikbare items zijn.
	Post Conditie:	geen verandering in het systeem.
Alternatief Scenario 2:	m.b.t. stap 6 van het main scenario.	

	Trigger:	De startdatum ligt na de einddatum.
	Scenario:	1. Het systeem geeft een melding dat de startdatum voor de einddatum moet liggen.
	Post Conditie:	geen verandering in het systeem.



Figuur 12: Wireframe 2-1 overzicht items

Browser

http://...

Verzoek systeem [logout](#)

Reserveringen

Uitgeleende items

Aanvragen

Item:

omschrijving:

startdatum:

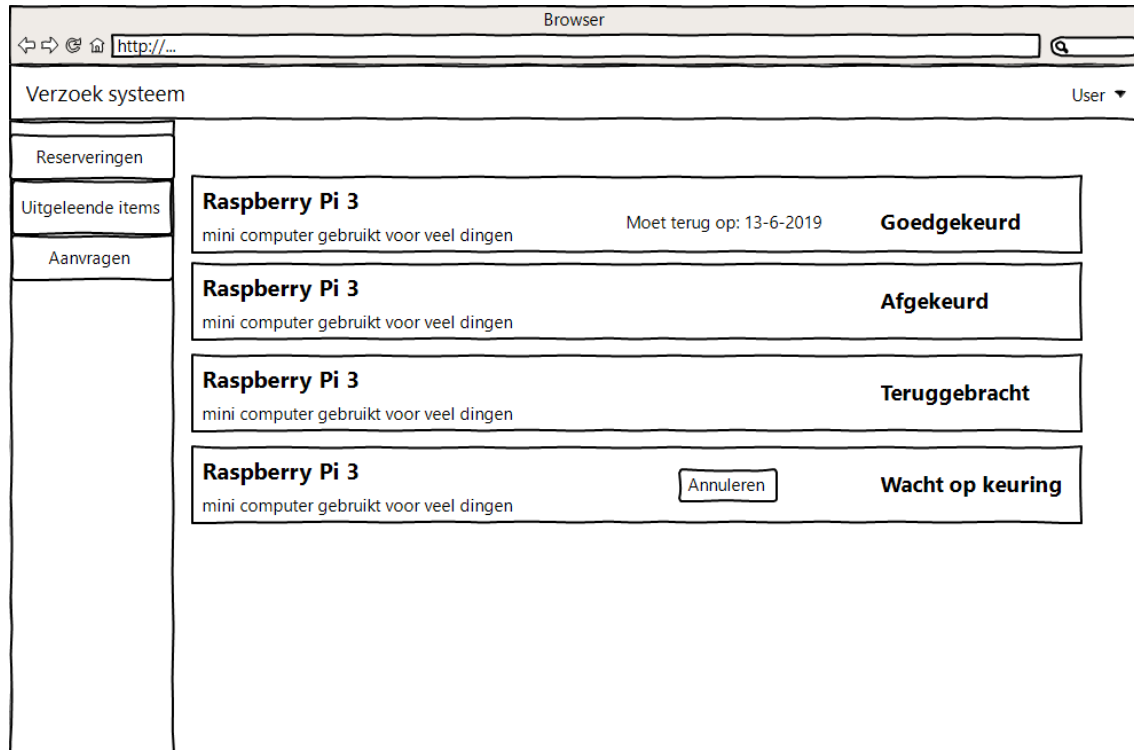
eindatum:

Figuur 13: Wireframe 2-2 form item aanvraag

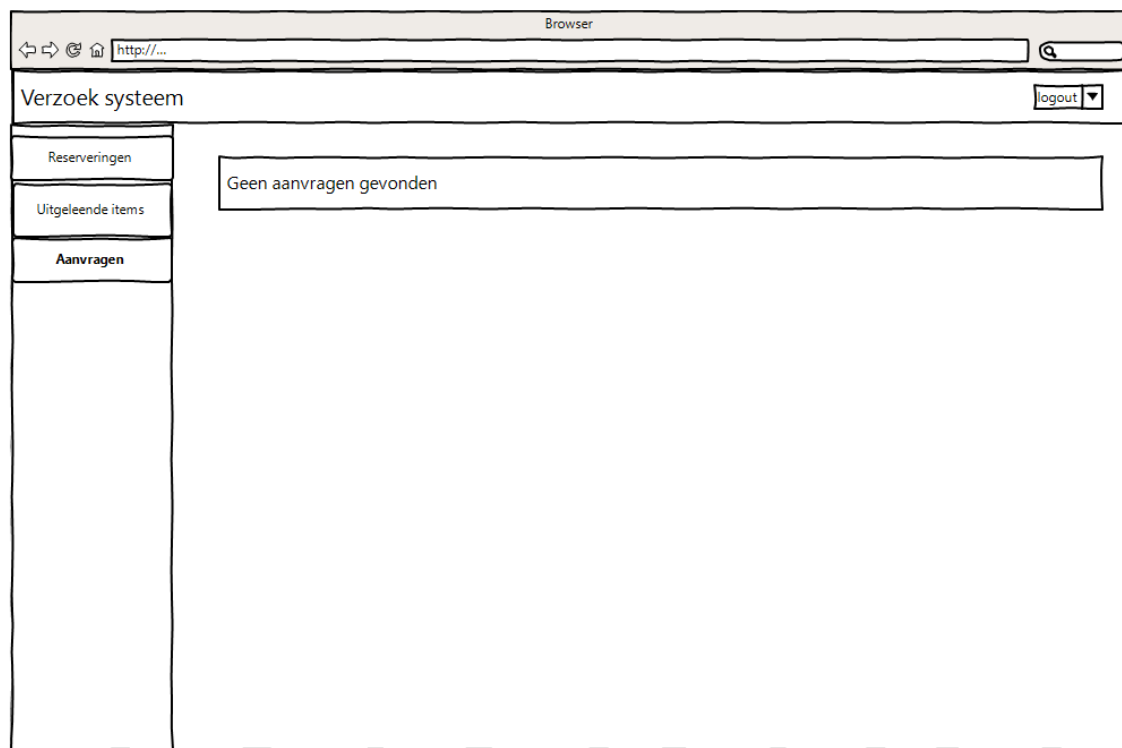
Use Case 3: Aanvragen inzien

Het systeem toont de gebruiker een lijst aanvragen (bestaande uit de namen van de items (out), status van de aanvraag (out), startdatum van aanvraag(out), einddatum van de aanvraag(out) en het serienummer (out)).

Use Case Nr.:	UC-3	
Use Case Naam:	Aanvragen inzien	
Actoren:	Student, Beheerder, Docent	
Samenvatting:	Het systeem toont de gebruiker een lijst van al zijn of haar aanvragen, met de volgende informatie per aanvraag: naam van het item (out), status van de aanvraag (out), startdatum van aanvraag (out), einddatum van de aanvraag(out) en het serienummer (out), naam van de aanvrager(in)	
Pre Conditie:	De actor is ingelogd, er is op zijn minst, één aanvraag gedaan door de actor, als de actor een beheerder is krijgt deze een overzicht van alle lopende aanvragen.	
Main Scenario:	<ol style="list-style-type: none"> 1. Systeem haalt alle aanvragen op waar de actor verantwoordelijk voor is. 2. Systeem toont alle aanvragen (itemnaam (out), status van de aanvraag (out), startdatum van aanvraag (out), einddatum van de aanvraag(out) en het serienummer (out), naam van de aanvrager(in)) <i>Usecase beëindigd</i>	
Post Conditie:	Er is geen verandering in het systeem	
Alternatief Scenario 1:	Trigger:	Er zijn geen aanvragen gedaan door de actor
	Scenario:	<ol style="list-style-type: none"> 1. Het systeem geeft een melding dat er geen aanvragen gedaan zijn.
	Post Conditie:	geen verandering in het systeem.



Figuur 14: Wireframe 3-1 aanvraag inzien



Figuur 15: Wireframe 3-2 aanvraag inzien geen items

Use Case 4: Item terugbrengen

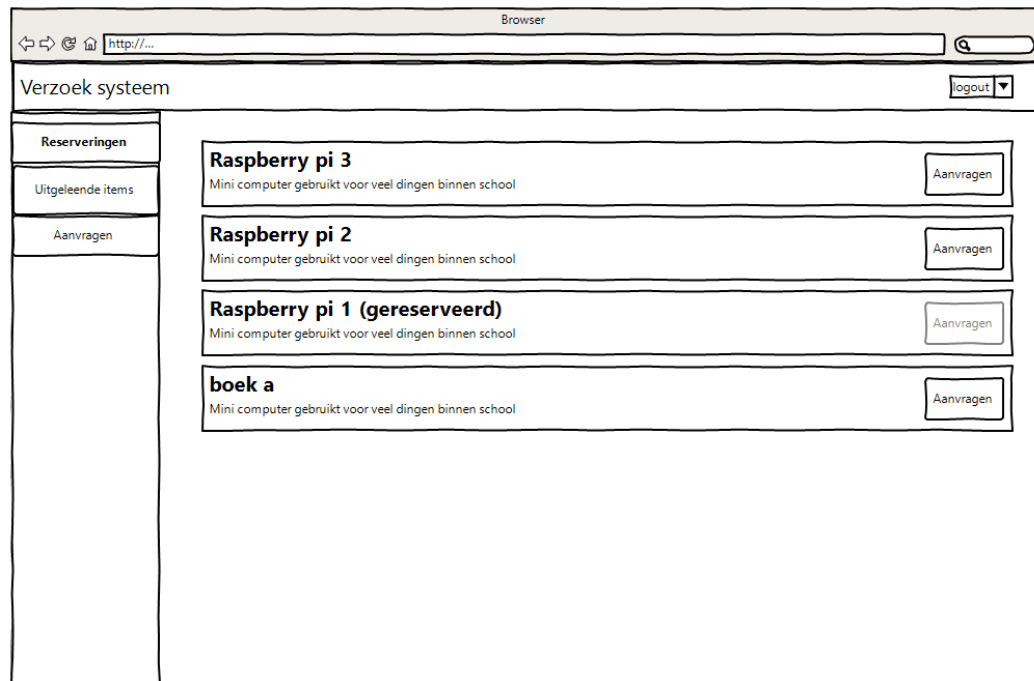
Dit is een vervolg de use case Aanvragen inzien. Bij aanvragen waar de status “waitingForReturn” is. Het systeem geeft de gebruiker de mogelijkheid om een vooraf gekozen aanvraag terug te brengen.

Use Case Nr.:	UC-4	
Use Case Naam:	Item terugbrengen	
Actoren:	Beheerder	
Samenvatting:	Het systeem geeft de gebruiker de optie om een item ‘terug te brengen’, dit is een use case die verder gaat op aanvragen inzien en is alleen beschikbaar voor de Beheerder, deze kan een aanvraag selecteren en een aangeven dat het item is teruggebracht en weer beschikbaar is om uit te lenen.	
Pre Conditie:	De actor is ingelogd, de actor is een beheerder, daarnaast is er op zijn minst één aanvraag gedaan door hem of een andere actor.	
Main Scenario:	<ol style="list-style-type: none"> 1. De actor kiest ervoor om een item terug te brengen. 2. Het systeem toont de actor de gegevens van de aanvraag: naam van het item (out), status van de aanvraag (out), startdatum van aanvraag (out), einddatum van de aanvraag(out) en het serienummer (out), naam van de aanvrager(in) 3. De actor kiest dat het item in goede orde terug is.’ <i>Usecase beëindigd</i>	
Post Conditie:	Het teruggebrachte item kan weer geleend worden.	
Alternatief Scenario 1:	m.b.t. stap 3	
	Trigger:	Het item kan, om wat voor reden dan ook, niet terug gebracht worden.
	Scenario:	<ol style="list-style-type: none"> 1. De actor kiest er voor dat het item niet teruggebracht kan worden en geeft dit aan.
	Post Conditie:	geen verandering in het systeem.

Use Case 5: Beschikbare items inzien

Dit is een vervolg van de UC-1 waarin de gebruiker is ingelogd en de beschikbare producten wilt inzien. Zie afbeelding 17 voor de gebruikte wireframe.

Use Case Nr.:	UC-5	
Use Case Naam:	Beschikbare items inzien	
Actoren:	Student, Beheerder, Docent	
Samenvatting:	Het systeem toont de gebruiker alle items die nog niet uitgeleend zijn. Gereserveerde items horen hier ook bij maar daar staat ook bij dat het gereserveerd is (aanvraag status). (out) Verder wordt er per item de naam (out) en omschrijving (out) van het item getoond.	
Pre Conditie:	De actor is ingelogd	
Main Scenario:	1. Het systeem toont een lijst van alle beschikbare en gereserveerde items. Bestaande uit naam (out) en omschrijving (out). <i>Usecase beëindigd</i>	
Post Conditie:	Er zijn geen veranderingen in het systeem	
Alternatief Scenario 1:	Mbt tot stap 1	
	Trigger:	Er zijn geen items om getoont te worden
	Scenario:	1. Het systeem toont de actor dat er geen beschikbare items zijn.
	Post Conditie:	geen verandering in het systeem.



Figuur 17: Wireframe 5-1 Beschikbare items inzien

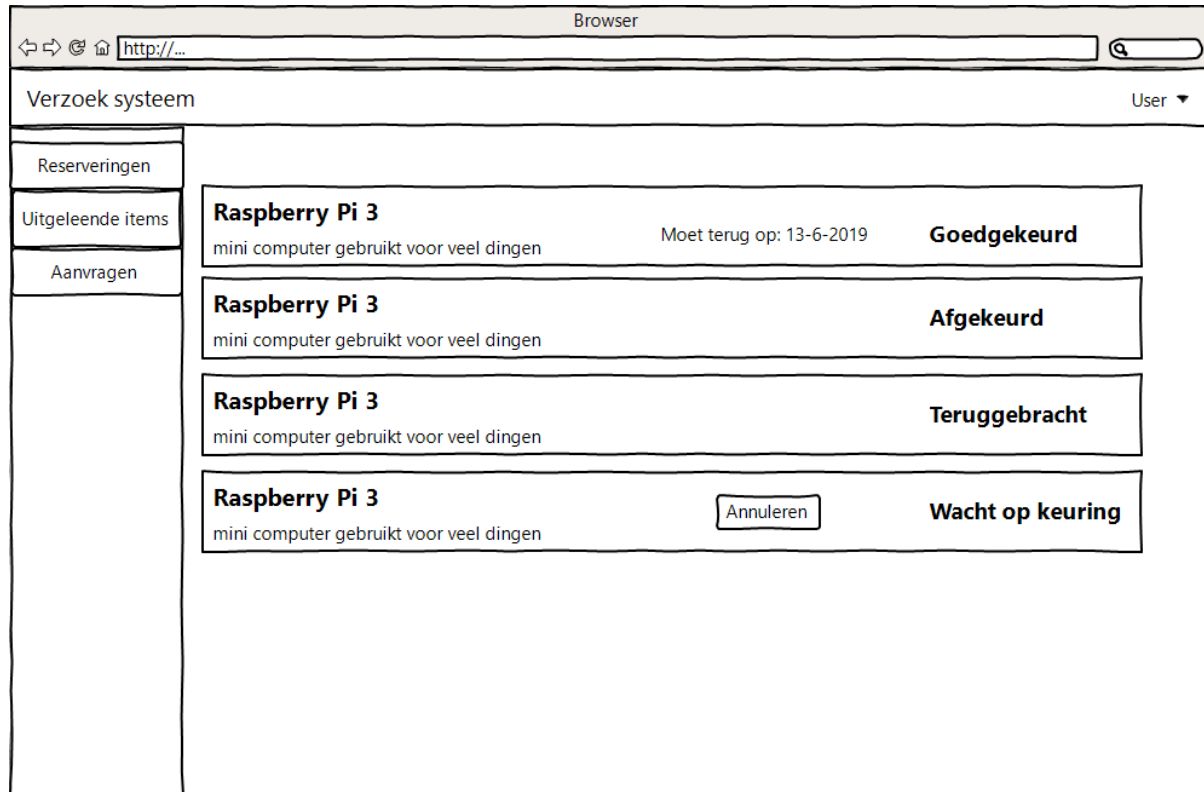


Figuur 18: Wireframe 5-2 Beschikbare items inzien geen beschikbaar

Use Case 6: Aanvraag annuleren

Dit is een vervolg op de use case Aanvragen inzien. Bij aanvragen waar de status “waitingForApproval” geeft het systeem de mogelijkheid de aanvraag te annuleren.

Use Case Nr.:	UC-6
Use Case Naam:	Aanvraag annuleren
Actoren:	Student, Beheerder, Docent
Samenvatting:	Het systeem geeft de gebruiker de optie om een aanvraag te annuleren’, dit is een use case die verder gaat op aanvragen inzien en is, de actor kan op een aanvraag met de status ‘waitingForApproval’ op een knop annuleren klikken (out). Deze knop is bij een andere status niet beschikbaar.
Pre Conditie:	De actor is ingelogd en heeft minimaal een aanvraag gedaan die nog niet is gecontroleerd door een beheerder.
Main Scenario:	<ol style="list-style-type: none">1. Systeem toont bij eigen aanvragen met de status ‘waitingForApproval’ een knop met de tekst annuleren (out).2. De actor selecteert een aanvraag om te annuleren.3. Het systeem verwerkt de keuze van de actor.<ol style="list-style-type: none">a. Het systeem annuleert de aanvraag.b. Het systeem toont de actor een succes bericht.. <p><i>Usecase beëindigd</i></p>
Post Conditie:	De aanvraag is geannuleerd



Figuur 19: Wireframe 6-1 knop annuleren

Use Case 7: Item reserveren

Dit is een vervolg op de use case Beschikbare items inzien.

Use Case Nr.:	UC-7	
Use Case Naam:	Item reserveren	
Actoren:	Beheerder, Docent	
Samenvatting:	Het systeem toont de gebruiker de mogelijkheid om een item te reserveren. Hierbij vraagt het systeem de actor om een hoeveelheid (in), startdatum van aanvraag (in), einddatum van aanvraag (in), een lijst van vakken (bestaande uit naam(out)) waaruit een moet worden gekozen (in) en om een bevestiging.	
Pre Conditie:	De actor is ingelogd, zijn items beschikbaar om uit te lenen en UC-5 is uitgevoerd.	
Main Scenario:	<ol style="list-style-type: none"> 1. Het vraagt de actor om een startdatum(in), einddatum(in), hoeveelheid (in), vak(in) en reservering(in). 2. De actor verstrekt de gegevens. 3. Het systeem vraagt de actor om een bevestiging. 4. De actor bevestigt zijn gegevens. 5. Het systeem maakt een aanvraag aan. <i>Usecase beëindigd</i>	
Post Conditie:	Er staat een aanvraag in het systeem.	
Alternatief Scenario 1:	m.b.t. stap 4 van het main scenario.	
	Trigger:	De actor annuleert de bevestiging.
	Scenario:	1. Het systeem gaat terug naar ms-1.
	Post Conditie:	geen verandering in het systeem.

Browser

http://...

Verzoek systeem logout

Reserveringen

Uitgeleende items

Aanvragen

Item:

omschrijving:

startdatum:

eindatum:

☐ Reservering

Hoeveelheid:

Vak:

Figuur 20: Wireframe 7-1 Item reserveren

Use Case 8: Verzoek behandelen

Deze use case gaat over het behandelen van nieuwe vragen.

Use Case Nr.:	UC-8						
Use Case Naam:	Verzoek behandelen						
Actoren:	Beheerder						
Samenvatting:	Het systeem toont de actor een overzicht van de aanvragen (bestaande uit naam van de item (out), naam van de aanvrager (out), startdatum van de aanvraag (out), einddatum van de aanvraag (out), de hoeveelheid (out)) die de status "waitingForApporval" hebben. Het systeem geeft hierbij de actor de optie een verzoek te keuren en verwerkt vervolgens zijn keuze.						
Pre Conditie:	De gebruiker is ingelogd, er zijn openstaande verzoeken aanwezig.						
Main Scenario:	<ol style="list-style-type: none"> 1. Het systeem toont de actor een overzicht van de aanvragen ((bestaande uit naam van de item (out), naam van de aanvrager (out), startdatum van de aanvraag (out), einddatum van de aanvraag (out), de hoeveelheid (out)) met de status "waitingForApporval". Hierbij geeft het systeem de actor de optie om een aanvraag af of goed te keuren. 2. De actor keurt een aanvraag goed. 3. Het systeem verwerkt de keuze van de actor. <ol style="list-style-type: none"> a. Het systeem zet de status van de aanvraag op "waitingForReturn". b. Het systeem toont de actor een succes bericht. <i>Use case beëindigd.</i>						
Post Conditie:	De status van de aanvraag is gezet op "waitingForReturn"						
Alternatief Scenario 1:	<p>m.b.t. stap 2 van het main scenario.</p> <table> <tr> <td>Trigger:</td><td>De actor keurt de aanvraag af.</td></tr> <tr> <td>Scenario:</td><td> <ol style="list-style-type: none"> 1. Het verwerkt de keuze van de actor. <ol style="list-style-type: none"> a. Het systeem zet de status van de aanvraag op "Archived". b. Het systeem toont de actor een succes bericht. <i>Use case beëindigd.</i> </td></tr> <tr> <td>Post Conditie:</td><td>De status van de aanvraag is gezet op "Archived "</td></tr> </table>	Trigger:	De actor keurt de aanvraag af.	Scenario:	<ol style="list-style-type: none"> 1. Het verwerkt de keuze van de actor. <ol style="list-style-type: none"> a. Het systeem zet de status van de aanvraag op "Archived". b. Het systeem toont de actor een succes bericht. <i>Use case beëindigd.</i>	Post Conditie:	De status van de aanvraag is gezet op "Archived "
Trigger:	De actor keurt de aanvraag af.						
Scenario:	<ol style="list-style-type: none"> 1. Het verwerkt de keuze van de actor. <ol style="list-style-type: none"> a. Het systeem zet de status van de aanvraag op "Archived". b. Het systeem toont de actor een succes bericht. <i>Use case beëindigd.</i>						
Post Conditie:	De status van de aanvraag is gezet op "Archived "						

Browser

[↶](#)
[↷](#)
[🔍](#)
[🏠](#)

Verzoek systeem

User ▼

Reserveringen

Uitgeleende items

Aanvragen

Raspberry Pi 3
1X
Jan Jansen
01/01/1990 - 30/30/2040

Goedkeuren

Afkeuren

Wacht op keuring

mini computer gebruikt voor veel dingen

Raspberry Pi 3
1X
Jan Jansen
01/01/1990 - 30/30/2040

Goedkeuren

Afkeuren

Wacht op keuring

mini computer gebruikt voor veel dingen

Raspberry Pi 3
1X
Jan Jansen
01/01/1990 - 30/30/2040

Goedkeuren

Afkeuren

Wacht op keuring

mini computer gebruikt voor veel dingen

Raspberry Pi 3
1X
Jan Jansen
01/01/1990 - 30/30/2040

Goedkeuren

Afkeuren

Wacht op keuring

mini computer gebruikt voor veel dingen

Raspberry Pi 3
1X
Jan Jansen
01/01/1990 - 30/30/2040

Goedkeuren

Afkeuren

Wacht op keuring

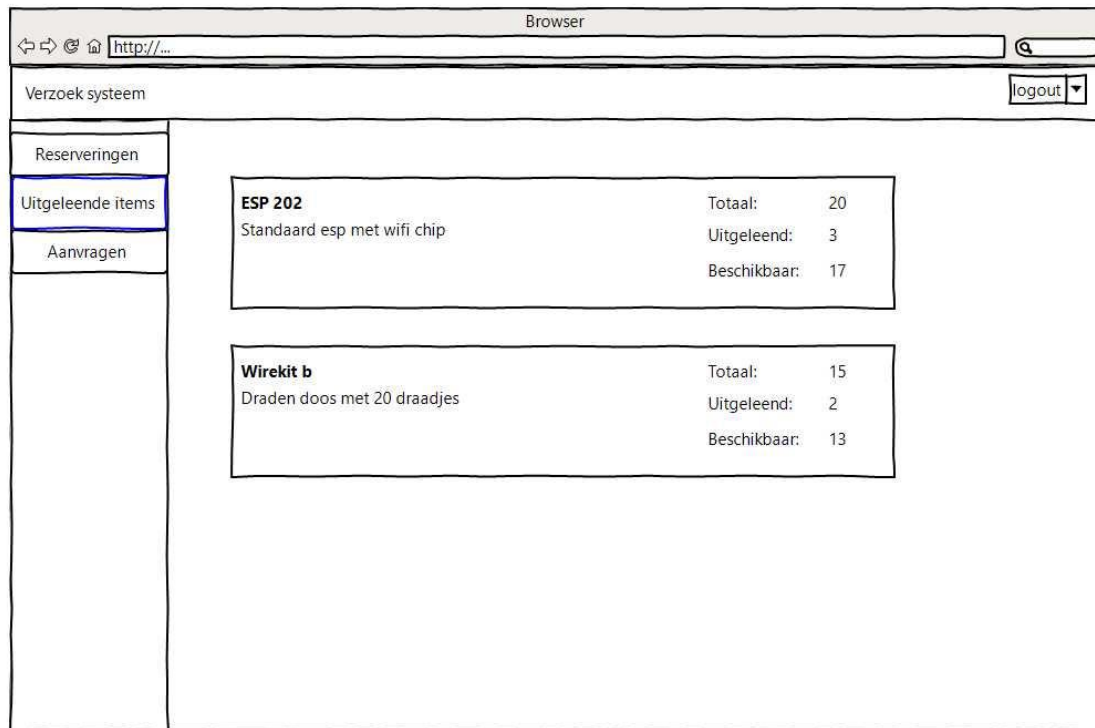
mini computer gebruikt voor veel dingen

Figuur 21: Wireframe 8-1 Verzoek behandelen

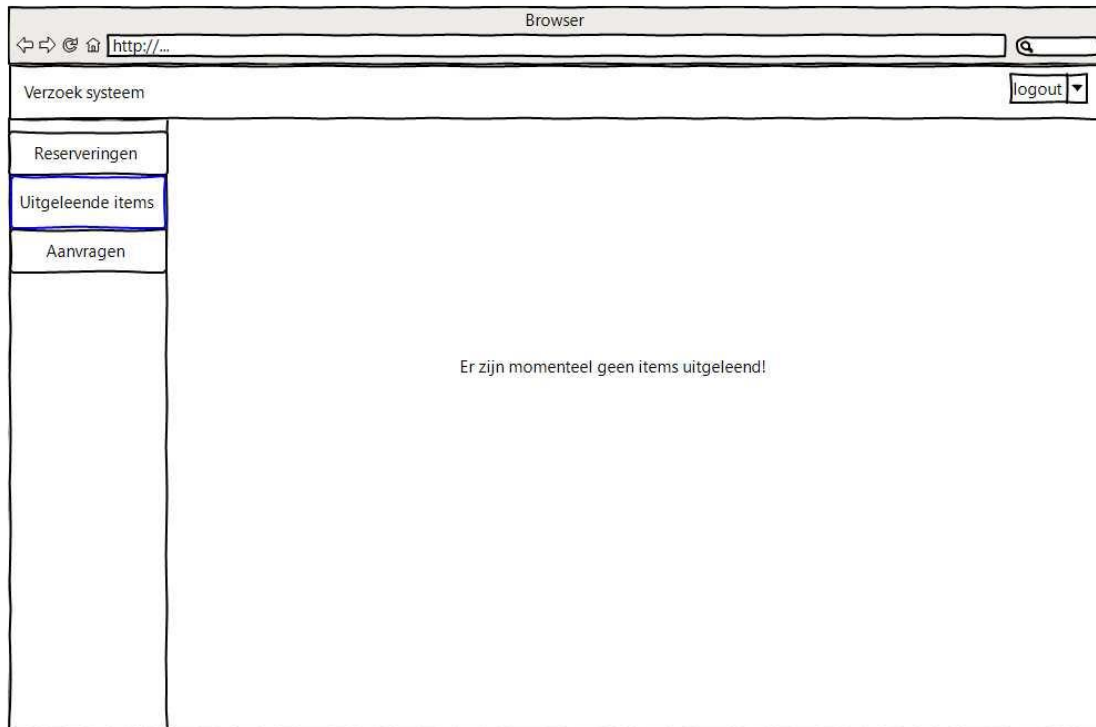
Use Case 9: Uitgeleende items inzien

De actor krijgt een overzicht van alle items die op dit moment als uitgeleend geregistreerd zijn (bestaan uit de naam van de item(out), naam van lener (out), startdatum van de lening(out) de einddatum van de lening(out) en serienummer (out))

Use Case Nr.:	UC-9						
Use Case Naam:	Uitgeleende items inzien						
Actoren:	Beheerder						
Samenvatting:	De actor krijgt een overzicht van alle uitgeleende items op dit moment met de volgende informatie: naam van de item(out), naam van lener (out), startdatum van de lening(out) de einddatum van de lening(out) en serienummer (out)						
Pre Conditie:	de actor is een beheerder						
Main Scenario:	<ol style="list-style-type: none"> 1. Het systeem toont een optie om de lopende leningen in te zien. 2. De actor selecteert een lening. 3. Het systeem toont een overzicht van alle uitgeleende items met de volgende informatie: naam van de item(out), naam van lener (out), startdatum van de lening(out) de einddatum van de lening(out) en serienummer (out) <p><i>Usecase beëindigd</i></p>						
Post Conditie:	Het systeem blijft onveranderd						
Alternatief Scenario 1:	<p>m.b.t. stap 1 van het main scenario.</p> <table> <tr> <td>Trigger:</td><td>Het systeem heeft geen uitgeleende items</td></tr> <tr> <td>Scenario:</td><td>Het systeem toont een melding dat er geen uitgeleende items zijn op dit moment.</td></tr> <tr> <td>Post Conditie:</td><td>Het systeem blijft onveranderd.</td></tr> </table>	Trigger:	Het systeem heeft geen uitgeleende items	Scenario:	Het systeem toont een melding dat er geen uitgeleende items zijn op dit moment.	Post Conditie:	Het systeem blijft onveranderd.
Trigger:	Het systeem heeft geen uitgeleende items						
Scenario:	Het systeem toont een melding dat er geen uitgeleende items zijn op dit moment.						
Post Conditie:	Het systeem blijft onveranderd.						



Figuur 22: Wireframe 9-1 Uitgeleende items inzien



Figuur 23: Wireframe 9-2 Geen uitgeleende items beschikbaar

Use Case 10: Gebruiker registreren

Het systeem toont de gebruiker een formulier waarin hij zijn gegevens kan invoeren bestaande uit: naam(in) en email(in) , hierna wordt het account aangemaakt.

Use Case Nr.:	UC-10	
Use Case Naam:	Gebruiker registreren	
Actoren:	Student, Beheerder, Docent	
Samenvatting:	De actor voert zijn gegevens in: naam(in) en email(in) en registreert zichzelf daarna.	
Pre Conditie:	geen.	
Main Scenario:	<ol style="list-style-type: none">1. Het systeem toont de optie om te registreren2. De actor selecteert het.3. Het systeem vraagt de actor om een naam(in) en email(in).4. De actor verstrekt zijn gegevens en bevestig deze.5. Het systeem maakt de gebruiker in het systeem en de actor wordt ingelogd. <i>Usecase beëindigd</i>	
Post Conditie:	Er is een nieuwe gebruiker in het systeem	
Alternatief Scenario 1:	m.b.t. stap 4 van het main scenario.	
	Trigger:	De actor vult invalide gegevens in.
	Scenario:	<ol style="list-style-type: none">1. De actor vult invalide gegevens in2. Het systeem geeft hier een melding van.
	Post Conditie:	Het systeem is onveranderd.

A hand-drawn wireframe of a web browser window. The browser's address bar at the top shows "http://...". In the center of the page is a rectangular box titled "Register". Inside this box, there are two input fields: "Email address" with the text "Jan@jan.nl" and "Password" with three asterisks "***". Below these fields are two buttons: "Register" and "Login".

Figuur 24: Wireframe 10-1 Registratie actor

Technische Requirements

De applicatie dient te draaien in de firebase omgeving gebruikmakend van de database opties die daar worden aangeboden. De applicatie dient goed te werken in web weergaven. Mobiel is hierbij niet verplicht.