

BACHELORPROEF

OPEN WEBSLIDES

ANNOTATOR

TEAM WEBSLIDES-01

Bavo DEPREZ
Kurt D'HAENE
Phaedra NEIRINCK
Simeon BRUYLAND

Het project rond Open Weblides is in onderzoeksfase aan de universiteit Gent. Er wordt gezocht naar een gebruiksvriendelijk, gratis platform voor het delen van cursusmateriaal. Dit wordt gerealiseerd via een mini-forum rond weblides waarbij interactie tussen studenten en professoren mogelijk is. Met de annotatietool kunnen opmerkingen verbonden worden aan cursuselementen. Vervolgens kan er gemakkelijk genavigeerd worden tussen die opmerkingen via verschillende functies. De weblides zijn een betere manier om informatie te bundelen en kunnen hierbij andere platformen integreren.

Inhoudsopgave

Hoofdstuk 1: Inleiding	1
1.1 Context	1
1.2 Probleemstelling.....	1
1.3 Doelstelling	2
1.4 Structuur van het verslag.....	2
Hoofdstuk 2: Gebruikersaspecten	3
2.1 High-level requirements	3
2.2 Use case diagrammen.....	3
2.3 Volledige feature list	9
Hoofdstuk 3: Systeemarchitectuur.....	9
3.1 High-level systeemmodel.....	9
3.2 Klassendiagrammen	11
3.3 Sequentiediagrammen	33
3.4 Databank.....	36
3.5 Activiteitendiagrammen	36
Hoofdstuk 4: Testplan.....	38
4.1 Backend	38
4.2 Frontend	38
4.2.1 Geautomatiseerd: Unit en integration testen	38
4.2.2 Manueel: Usability en Integration testen.....	40
4.3 Platform testen.....	44
Hoofdstuk 5: Evaluatie en discussies.....	46
5.1 Performantie	46
5.1.1 Componentniveau	46
5.1.2 Systeemniveau	47
5.2 Beveiliging	47
5.3 Schaalbaarheid	47
5.4 Problemen en geleerde lessen	47
Hoofdstuk 6: Installatiehandleiding en gebruikershandleiding	50
6.1 Installatiehandleiding	50
6.1.1 Pre-setup	51
6.1.2 Setup	53

6.1.3 Jest installeren.....	55
6.2 Gebruikershandleiding	56
6.2.1 Openen van de annotatietool	56
6.2.2 Navigatieknoppen	56
6.2.3 Annotaties zoeken en bekijken	57
6.2.4 Annotaties plaatsen	58
6.2.5 Annotaties aanpassen en verwijderen	59
6.2.6 Reactie op een annotatie plaatsen en aanpassen	60
6.2.7 Notificaties.....	60
6.2.8 Annotaties en reacties beoordelen	60
6.2.9 Tooltips	60
Hoofdstuk 7: Besluit	61
Referentielijst.....	62

Overzicht van de toegevoegde figuren

Figuur 1: Use case diagram.....	8
Figuur 2: Deployment-diagram.....	10
Figuur 3: Klassendiagram met inhoud van Component-module	26
Figuur 4: Klassendiagram met inhoud van Manager-, Drawer- en Nav-modules	27
Figuur 5: Klassendiagram met inhoud van AnnotationOverview- en AnnotationForm-modules	28
Figuur 6: Klassendiagram met inhoud van Caller-, Config-, Sidebar- en LocalData-modules....	29
Figuur 7: Klassendiagram backend.....	32
Figuur 8: Sequentiediagram van AnnotationOverview-module	34
Figuur 9: Sequentiediagram van AnnotationForm-module	35
Figuur 10: Activiteitendiagram van AnnotationForm-module	37
Figuur 11: Inhoud van Caller_loadAnnotationsFromServer.test.js	39
Figuur 12: Tooltip	40
Figuur 13: Foutmelding	40
Figuur 14: Helpcontainer	40
Figuur 15: Foutmelding	41
Figuur 16: Foutmelding	41
Figuur 17: Foutmelding	41
Figuur 18: Standaardtags met kleur	49
Figuur 19: Zoekbalk met input	49
Figuur 20: Zoekbalk zonder input.....	49
Figuur 21: Voorbeeld van het .yaml-bestand	51
Figuur 22: Voorbeeld van een dockerfile	52
Figuur 23: Voorbeeld van een .properties-bestand	52
Figuur 24: Voorbeeld inhoud config.js	53
Figuur 25: Voorbeeld commando's om installatiescript uit te voeren	53
Figuur 26: Structuur in WebApplicationApp	54
Figuur 27: Uitvoeren script om containers te starten.....	54
Figuur 28: Navigatieknoppen in het Default-tabblad	56
Figuur 29: Overview-tabblad.....	58
Figuur 30: Add-tabblad.....	59
Figuur 31: Annotatie met reacties.....	60
Figuur 32: Tooltip	60

Overzicht van de toegevoegde tabellen

Tabel 1: Backlog	9
Tabel 2: Leden Annotation-klasse	11
Tabel 3: Leden Reaction-klasse	12
Tabel 4: Leden Voter-klasse	13
Tabel 5: Leden LocalData-klasse.....	13
Tabel 6: Leden Drawer-module	14
Tabel 7: Leden Caller-module	16
Tabel 8: Leden Config -module.....	17
Tabel 9: Leden Nav-module	18
Tabel 10: Leden Manager-module.....	19
Tabel 11: Leden AnnotationOverview-module	20
Tabel 12: Leden AnnotationForm-module	22
Tabel 13:Leden Sidebar module	24
Tabel 14: Relaties tussen klassen en modules	24
Tabel 15: Leden AnnotationsService-klasse.....	30
Tabel 16: Leden AnnotationsCRUD-klasse.....	30
Tabel 17: Leden AnnotationsAdapter-klasse.....	31
Tabel 18: Testplan.....	44

Hoofdstuk 1: Inleiding

1.1 Context

Open Weblides is een project in onderzoeksfase aan de universiteit Gent. De *weblides* kunnen visueel vergeleken worden met een online PowerPointpresentatie, vergezeld van extra functionaliteit. Deze *slides* gebruiken namelijk enkel JavaScript, HTML en CSS waardoor ze steeds aanpasbaar zijn, net zoals gewone websites. Er moeten geen speciale programma's gedownload worden, want ze zijn zowel mobiel als op vaste toestellen beschikbaar in een browser. Het sleutelwoord "open" verwijst naar het gratis platform en de keuze die de eindgebruiker heeft tussen verschillende browsers. Diegenen die hun informatie online zetten op *weblides* krijgen ook zelf de keuze over de beschikbaarheid van deze info.

Op dit moment zijn de *weblides* al in gebruik door de ontwikkelaars bij het lesgeven, wel zonder bepaalde *features*. Vorig jaar werd er een bachelorproef ingediend rond het converteren van documenten naar *webslide*-formaat. Zo kunnen alle cursussen met gemak toegevoegd worden. Ook is het reeds mogelijk bekende platformen als *YouTube* te implementeren in die *weblides*. In de toekomst zouden nog enkele *features* bijgevoegd worden, zoals het annoteren van *slides* en het slim afdrukken. Via een annotatietool kunnen er annotaties aangemaakt worden die achteraf gefilterd, geordend, beoordeeld en door middel van *cherrypicking* verkozen kunnen worden tot favoriet. Deze annotaties kunnen dan weer handig mee afgedrukt worden, zodat de student een mooi overzicht krijgt van de cursus met annotaties.

Naast die mogelijke toekomstige *features*, is het basisidee al volledig uitgewerkt. De studenten zijn via hun UGent-account verbonden en kunnen discussies starten over verschillende *topics* rond hun cursusmateriaal. Door feedback van de professoren en medestudenten wordt dit uitgebouwd tot een heus forum, wat een positief effect heeft op toekomstige studenten.

1.2 Probleemstelling

Wanneer studenten vragen hebben in verband met de leerstof, raadplegen ze wiki's, sociale media en fora of sturen ze mails naar de professor. De benodigde informatie staat vervolgens op verscheidene bronnen waardoor het een hele opgave is die te bundelen. Ook is het op die platformen niet gemakkelijk om te verwijzen naar delen uit de cursus. Het zou handig zijn indien die referenties integreerbaar zouden zijn, want dan hoeven die pagina- en slidenummers niet meer vermeld te worden en staat alles mooi samen.

Vaak worden er vragen gesteld in de les die niet hoorbaar zijn voor het hele auditorium of enkel privé worden beantwoord in de pauze. De antwoorden van de professor zijn nuttig voor alle aanwezige studenten, dus het zou een verbetering zijn indien die bijgehouden kunnen worden op een online prikbord.

Wanneer bepaalde cursusonderwerpen niet genoeg uitgediept worden tijdens de les of complex zijn, zoeken de studenten naar aanvullende informatie. De gevonden info is vaak fout of niet meer geldig, wat niet altijd duidelijk is zonder inbreng van de professor. Deze foute informatie wordt steeds verder gedeeld tussen studenten en wordt als waarheid beschouwd. Het komt terecht in wanordelijke documenten vol vragen en discussies. Ook worden er statussen over

verschillende *topics* gestart op sociale media, dit leidt tot “vakgroepen” die bijgehouden worden doorheen de jaren. Ze bevatten vaak informatie over een verouderde cursus en na veel zoekwerk blijkt er meestal niet veel nuttigs terug te vinden. Indien elke cursusversie een eigen plek zou hebben voor opmerkingen met steeds een betrouwbaar antwoord, zouden deze problemen verholpen kunnen worden. Hierdoor heeft de prof zicht op die gedeelde info en kan hij bijsturen waar nodig, in het openbaar zodat elke student er nut aan heeft.

1.3 Doelstelling

In dit project werd er gekozen om de *feature* omtrent annotatietools verder uit te werken en te implementeren in bestaande *webslides*. Er zou op deze manier een oplossing komen om zowel voor professoren als studenten op één plek alle informatie te verzamelen. De openbare cursussen kan het publiek raadplegen, kunnen de professoren inhoudelijk aanpassen en kunnen opmerkingen krijgen door studenten. Hieronder wordt weergegeven hoe de annotatietools gebruikt moeten kunnen worden.

Wanneer de gebruiker zich begeeft naar een formulier op een specifieke dia, kan hij een opmerking toevoegen. Er zijn verschillende categorieën om uit te kiezen en *tags* kunnen bijgevoegd worden om extra informatie over die annotatie weer te geven. De zichtbaarheid van zo’n annotatie kan ingesteld worden om opmerkingen voor zichzelf te houden of publiek te zetten. Elke annotatie kan daarnaast ook verbonden worden met een element op de *webslide* om de context van die opmerking duidelijk te maken. Op die annotaties kunnen er reacties geplaatst worden en beiden kunnen door de schrijver steeds gewijzigd of verwijderd worden. Om de *input* van de professor zo opvallend mogelijk te maken, wordt zijn gebruikersnaam in een andere kleur weergegeven. Ook krijgen professoren extra moderatorfunctionaliteit door mogelijkheid tot verwijderen van alle annotaties en reacties. Op deze manier wordt er een mini-forum gecreëerd tussen medestudenten, met betrouwbare informatie door inbreng van de professor. Elke annotatie en reactie heeft nog een beoordelingssysteem en daarnaast kunnen er favoriete annotaties bijgehouden worden. De gebruiker kan van alle annotaties een mooi overzicht verkrijgen, waarbij een handige zoekfunctie is voorzien met verschillende sorteermogelijkheden.

Er zijn wekelijkse meldingen per presentatie beschikbaar, hierdoor kan de student steeds zien welke annotaties of reacties laatst toegevoegd werden, met mogelijkheid tot navigatie naar die specifieke *webslide*. De webapplicatie kan ook gebruikt worden door professoren tijdens hun spreekuren door de annotatietools tijdelijk te sluiten en tijdens de pauze om alle nieuwste vragen met één enkele knop op te halen. Ten slotte moet er mogelijkheid zijn tot veranderen van de taal zodat internationalisering ondersteund wordt.

De *Open Webslides* zijn een gebruiksvriendelijk, gratis platform voor het delen van cursusmateriaal en helpen de toekomstige studenten en professoren bij het leerproces.

1.4 Structuur van het verslag

In vorige paragrafen werd het idee achter *Open Webslides* aangehaald en welke doelen er voorop gesteld worden. In hoofdstuk 2 worden de gebruikersaspecten overlopen en in hoofdstuk 3 komt de systeemarchitectuur aan bod. Het testplan wordt besproken in hoofdstuk 4 met daaropvolgend de evaluatie in hoofdstuk 5. In hoofdstuk 6 staan de handleidingen met het besluit in hoofdstuk 7.

De oorspronkelijke vraag van de klant wordt eerst weergegeven in paragraaf 2.1. De features werden uitgetekend in het *use case* diagram, dat zich met verdere uitleg in paragraaf 2.2 bevindt. Vanuit de *backlog* met noodzakelijke *features* en nieuwe ideeën, werden drie sprints opgesteld. Het complete overzicht en de gekozen opdeling hiervan worden vermeld in 2.3.

Doorheen hoofdstuk 3 wordt de werking van de webapplicatie verduidelijkt met verschillende figuren, startend met het *high-level* systeemmodel (3.1) en vervolgens klasse- (3.2), sequentie- (3.3), databank- (3.4) en activiteitendiagrammen (3.5).

Welke tests en hoe ze uitgevoerd worden, wordt besproken in het testplan in paragraaf 4.

In hoofdstuk 5 wordt het project geëvalueerd door middel van performantie (5.1), beveiliging (5.2) en schaalbaarheid (5.3). Vervolgens volgt een bespreking van onvoorziene problemen in paragraaf 5.4.

De gebruiker en klant kunnen respectievelijk een gebruiks- en installatiehandleiding terugvinden in paragrafen 6.1 en 6.2.

Hoofdstuk 2: Gebruikersaspecten

2.1 High-level requirements

Het *Open Weblides team* had al een welbepaald idee achter het project en vele *features* daarvan werden al toegevoegd. De annotatietool is een nieuw idee dat de basis vormt voor dit project en kan indien nodig verborgen worden. Die annotaties zouden door de gebruiker volledig naar eigen zin gevormd kunnen worden, met gekozen zichtbaarheidsniveau, *tags*, titel, commentaar, categorie en visuele aanbeng op de *slides* zelf. Zulke opmerkingen kunnen aangevuld worden met reacties en beiden kunnen een beoordeling krijgen. Via ordenings- en filterfuncties zijn annotaties gemakkelijk op te sporen en via *cherrypicking* kunnen favoriete annotaties bijgehouden worden door elke gebruiker. Aanpassing en verwijdering van annotaties en reacties is mogelijk, maar steeds moet de *flow* behouden worden waarbij het zichtbaar is dat deze actie is uitgevoerd. Hiernaast zijn een overzichtspagina en een *alert*-container met meldingen voorzien met navigatiemogelijkheid naar annotatie van keuze. Ten slotte zou internationalisering verder uitgebreid moeten kunnen worden in de toekomst. Bovenal moet de *userinterface* gebruiksvriendelijk zijn zodat iedereen ermee overweg kan.

2.2 Use case diagrammen

Het *use case* diagram toont de verschillende *features* voor de gebruikers, bevindt zich op Figuur 1 en wordt hieronder verduidelijkt.

Use case 1

Naam: annotaties bekijken

Doelstelling: De gebruiker kan de annotaties van één *webslide* bekijken op het *default*-tabblad of de annotaties van de hele presentatie bekijken op het *overview*-tabblad. Zowel minimale (*by default*) als maximale versie van de annotatie kunnen bekomen worden.

Actoren: primair: gebruiker

Precondities: Er moet verbinding met het internet en de databank zijn zodat de annotaties

opgevraagd kunnen worden.

Postcondities: /

Successscenario: Na klikken op het plusicoon wordt de annotatie gemaximaliseerd, dit bevat dan de schrijver, categorie, titel, commentaar, *tags*, reacties en hun beoordeling. Wanneer de annotatie geminimaliseerd wordt door klikken op het minicoon, worden enkel schrijver, categorie, titel, *tags* en aantal reacties getoond.

Alternatieve scenario's: Er is geen connectiviteit waardoor er een foutmelding verschijnt of er zijn geen annotaties die weergegeven kunnen worden. Bij de laatste mogelijkheid volgt er een passend bericht.

Use case 2

Naam: annotatie plaatsen

Doelstelling: De gebruiker kan een annotatie opstellen door het kiezen van een categorie (*remark, extra, question*), titel, zichtbaarheidsniveau (*private, public*), *tags*, commentaar en het aanduiden van een bijpassend element in de *webslide*. Enkel het invullen van commentaar en titel is verplicht. Het formulier wordt verstuurd via de 'stuur door'-knop.

Actoren: primair: gebruiker

Precondities: De gebruiker klikt op het *add*-tabblad in de navigatiebalk. Er moet ook verbinding zijn met het internet en de databank zodat de annotatie verstuurd kan worden.

Postcondities: /

Successscenario: De gebruiker plaatst een annotatie en vindt deze terug op het *default*- en *overview*-tabblad.

Alternatieve scenario's: De gebruiker krijgt een foutmelding wanneer hij geen commentaar en/of titel meegegeven heeft of wanneer er geen verbinding is waardoor de annotatie niet geplaatst kan worden. Die foutmeldingen kunnen steeds weggeklikt worden door de gebruiker. De plaatsing kan geannuleerd worden door terug te keren naar het *default*-tabblad via de pijl of door het sluiten van de *sidebar*.

Use case 3

Naam: reactie plaatsen

Doelstelling: De gebruiker kan een reactie opstellen door het invullen van het tekstvak onder een annotatie en verstuurt deze door te klikken op de pijl.

Actoren: primair: gebruiker

Precondities: De gebruiker maximaliseert een annotatie om het tekstvak voor het plaatsen van reacties te kunnen raadplegen. Er moet ook verbinding zijn met het internet en de databank zodat annotaties opgevraagd kunnen worden voor het weergeven van de maximale annotatie en de reactie verzonden kan worden.

Postcondities: /

Successscenario: De gebruiker plaatst een reactie en vindt deze terug op het *default*- of *overview*-tabblad na maximalisatie van de annotatie.

Alternatieve scenario's: De gebruiker krijgt een foutmelding wanneer er geen verbinding is waardoor de reactie niet geplaatst kan worden of er gebeurt niks omdat er geen tekst werd ingevuld in het tekstvak. Die foutmeldingen kunnen steeds weggeklikt worden door de gebruiker. De plaatsing kan geannuleerd worden door niet op de pijl te klikken, het *default*-

tabblad te vernieuwen, naar het *overview*- of *add*-tabblad te navigeren of door het sluiten van de *sidebar*.

Use case 4

Naam: annotatie of reactie aanpassen

Doelstelling: De gebruiker kan een annotatie of reactie aanpassen door te klikken op de *edit*-knop. Wanneer het een annotatie is, wordt er genavigeerd naar een formulier op het *add*-tablad met de huidige waarden van de annotatie. Wanneer het een reactie is, wordt het tekstvak waarin de reactie staat aanpasbaar. Nadien moet de aanpassing verstuurd worden door te klikken op de ‘stuur door’-knop of de pijl.

Actoren: primair: gebruiker

Precondities: De gebruiker maximaliseert een annotatie om de *edit*-knop te kunnen raadplegen op zelfgeschreven annotaties en reacties. Er moet ook verbinding zijn met het internet en de databank zodat annotaties opgevraagd kunnen worden en de aanpassing verzonden kan worden.

Postcondities: /

Successscenario: De gebruiker vindt zijn annotatie of reactie aangepast terug op het *default*- of *overview*-tabblad. Ook staat er een klikbaar klokicoon naast deze annotatie of reactie zodat iedereen steeds de bewerkingsgeschiedenis kan opvolgen.

Alternatieve scenario's: De gebruiker krijgt een foutmelding wanneer er geen verbinding is waardoor de annotatie of reactie niet aangepast kan worden. Die foutmeldingen kunnen steeds weggeklikt worden door de gebruiker. Net zoals bij het aanmaken van annotaties of reacties kan de aanpassing steeds stopgezet worden. De gebruiker kan de verandering van een annotatie annuleren door terug te keren naar het *default*-tabblad via de pijl of door het sluiten van de *sidebar*. De aanpassing van een reactie kan stopgezet worden door niet op de pijl te klikken, het *default*-tabblad te vernieuwen, naar het *overview*- of *add*-tabblad te navigeren of door het sluiten van de *sidebar*.

Use case 5

Naam: annotatie of reactie verwijderen

Doelstelling: De gebruiker kan een annotatie of reactie verwijderen door te klikken op de *delete*-knop.

Actoren: primair: gebruiker

Precondities: De gebruiker maximaliseert een annotatie om de *delete*-knop te kunnen raadplegen op zelfgeschreven annotaties en reacties. Elke professor krijgt moderator-functionaliteit en kan daardoor alle annotaties of reacties verwijderen. Er moet ook verbinding zijn met internet en de databank zodat annotaties opgevraagd kunnen worden en de verwijdering verzonden kan worden.

Postcondities: /

Successscenario: De gebruiker vindt zijn annotatie of reactie wel terug na verwijdering op het *default*- of *overview*-tabblad, maar deze bevat de tekst “Dit werd verwijderd door gebruiker of professor” of overeenkomstige tekst naargelang de taalinstelling. Het is niet mogelijk de oorspronkelijke inhoud op te vragen.

Alternatieve scenario's: De gebruiker krijgt een foutmelding wanneer er geen verbinding is

waardoor de annotatie of reactie niet verwijderd kan worden. Die foutmeldingen kunnen steeds weggeklikt worden door de gebruiker.

Use case 6

Naam: annotatie of reactie beoordelen

Doelstelling: De gebruiker kan een annotatie of reactie beoordelen door te klikken op de bijhorende duimen. Hierdoor verandert het weergegeven aantal *upvotes* en/of *downvotes* en deze aanpassing zal zichtbaar zijn telkens de gebruiker de annotatie of reactie herbekijkt.

Actoren: primair: gebruiker

Precondities: De gebruiker maximaliseert een annotatie of reactie om de *upvotes* en *downvotes* te kunnen raadplegen. Er moet ook verbinding zijn met het internet en de databank zodat annotaties opgevraagd kunnen worden en de aanpassing verzonden kan worden.

Postcondities: /

Successscenario: De gebruiker vindt de *upvotes* en/of *downvotes* van de annotatie of reactie aangepast terug op het *default*- of *overview*-tabblad.

alternatieve scenario's: Wanneer er geen verbinding is, verschijnt er een foutmelding die door de gebruiker weggeklikt kan worden. Indien de gebruiker zelfgeschreven annotaties of reacties probeert te beoordelen, zal dit niet lukken en verschijnt er ook geen foutmelding.

Use case 7

Naam: annotatie toevoegen aan favorieten en favorieten ophalen (*cherry picking*)

Doelstelling: De gebruiker kan een annotatie aan zijn favorieten toevoegen en nadien alle favorieten opvragen op het *overview*-tabblad door op het stericoon te klikken.

Actoren: primair: gebruiker

Precondities: De gebruiker maximaliseert een annotatie om te checken of die annotatie al behoort tot zijn favorieten. Wanneer dit wel zo is, is de ster geel ingekleurd, anders is ze grijs. Er moet ook verbinding zijn met het internet en de databank zodat annotaties opgevraagd kunnen worden en de aanpassing verzonden kan worden.

Postcondities: /

Successscenario: De gebruiker vindt de ster van de annotatie aangepast terug op het *default*- of *overview*-tabblad. Ook bevindt deze annotatie zich tussen zijn favorieten wanneer ze later opgevraagd worden.

alternatieve scenario's Wanneer er geen verbinding is, verschijnt er een foutmelding die door de gebruiker weggeklikt kan worden. Het kan ook zijn dat de gebruiker geen favorieten heeft, dan wordt dit verduidelijkt bij opvragen op het *default*-tabblad.

Use case 8

Naam: annotaties filteren

Doelstelling: De gebruiker kan de annotaties van de hele presentatie filteren op het *overview*-tabblad. In de zoekbalk kan er gezocht worden op *op*, *person*, *commentary*, *text*, *contentTags*, *category* en *title*. Respectievelijk wil dit zeggen dat de gebruiker kan filteren op de schrijver van een annotatie en reactie, de inhoud van een annotatie en reactie, de *tags* die meegegeven werden en de gekozen categorie en titel.

Actoren: primair: gebruiker

Precondities: Er moet verbinding zijn met het internet en de databank zodat de annotaties opgevraagd kunnen worden. Tussen elke zoekterm moet er een komma staan, daarnaast is de zoekfunctie niet hoofdlettergevoelig en worden er enkel annotaties weergegeven die matchen met elk element uit de zoekbalk. Er kan door de gebruiker gekozen worden of hij ook tussen de reacties (schrijver en inhoud) wil zoeken of dit niet wil doen door het aanpassen van de optie “inclusief reacties”.

Postcondities: /

Successscenario: De gebruiker krijgt na elke ingevulde letter een geüpdatete gefilterde lijst met annotaties. Wanneer er bewerkingsgeschiedenis is, wordt er enkel gezocht tussen de laatste aanpassing en bij verwijdering worden de oorspronkelijke waarden gebruikt.

Alternatieve scenario's: De gebruiker krijgt geen lijst terug wanneer er geen verbinding is of wanneer er geen annotaties matchen met de vraag.

Use case 9

Naam: annotaties ordenen

Doelstelling: De gebruiker kan de annotaties van de hele presentatie of zijn zoekopdracht op het *overview*-tabblad ordenen. In de *drop-down list* kan er gekozen worden tussen “*slides*”, “*top*”, “*new*” en “*hot*” waarbij er respectievelijk gesorteerd wordt op de naam van de slides (alfabetisch), de totale beoordeling (aantal *upvotes* verminderd met aantal *downvotes*), de datum (recentste eerst) en het aantal reacties (hoogste aantal bovenaan).

Actoren: primair: gebruiker

Precondities: Er moet verbinding zijn met het internet en de databank zodat de annotaties opgevraagd kunnen worden.

Postcondities: /

Successscenario: De gebruiker krijgt na klikken op een sorteervolgorde een gesorteerde lijst met annotaties.

Alternatieve scenario's: Indien de gebruiker geen wijziging opmerkt na klikken op een element uit de lijst, dan werd de sortering wel uitgevoerd, maar werd hetzelfde resultaat bekomen. Het valt op te merken dat ook verwijderde annotaties en reacties meegesorteerd worden zodat de *flow* behouden kan worden.

Use case 10

Naam: *alerts* bekijken

Doelstelling: De gebruiker kan veranderingen uit de huidige presentatie van de voorbije week gemakkelijk opvragen. Dit kunnen nieuwe annotaties of reacties zijn, maar de aanpassingen van reeds geschreven inhoud of beoordelingen worden niet gemeld. Rechtsboven bevindt zich de *alert*-knop die voor de betrokken annotaties en reacties steeds zegt wie ze toegevoegd heeft, wat de titel is van de *webslide* en de annotatie en de gebruiker na klikken, brengt naar de besproken *slide*.

Actoren: primair: gebruiker

Precondities: Er moet verbinding zijn met het internet en de databank zodat de annotaties opgevraagd kunnen worden.

Postcondities: /

Successscenario: De gebruiker krijgt na klikken een venster met opsommingen van namen, *slide*-

titels en linken.

Alternatieve scenario's: Indien er de voorbije week geen toevoegingen gebeurden, wordt de tekst "niets nieuws deze week" getoond.

Use case 11

Naam: taal veranderen

Doelstelling: De gebruiker kan kiezen in welke taal hij de extra uitleg in de *tooltips* en foutmeldingen wil.

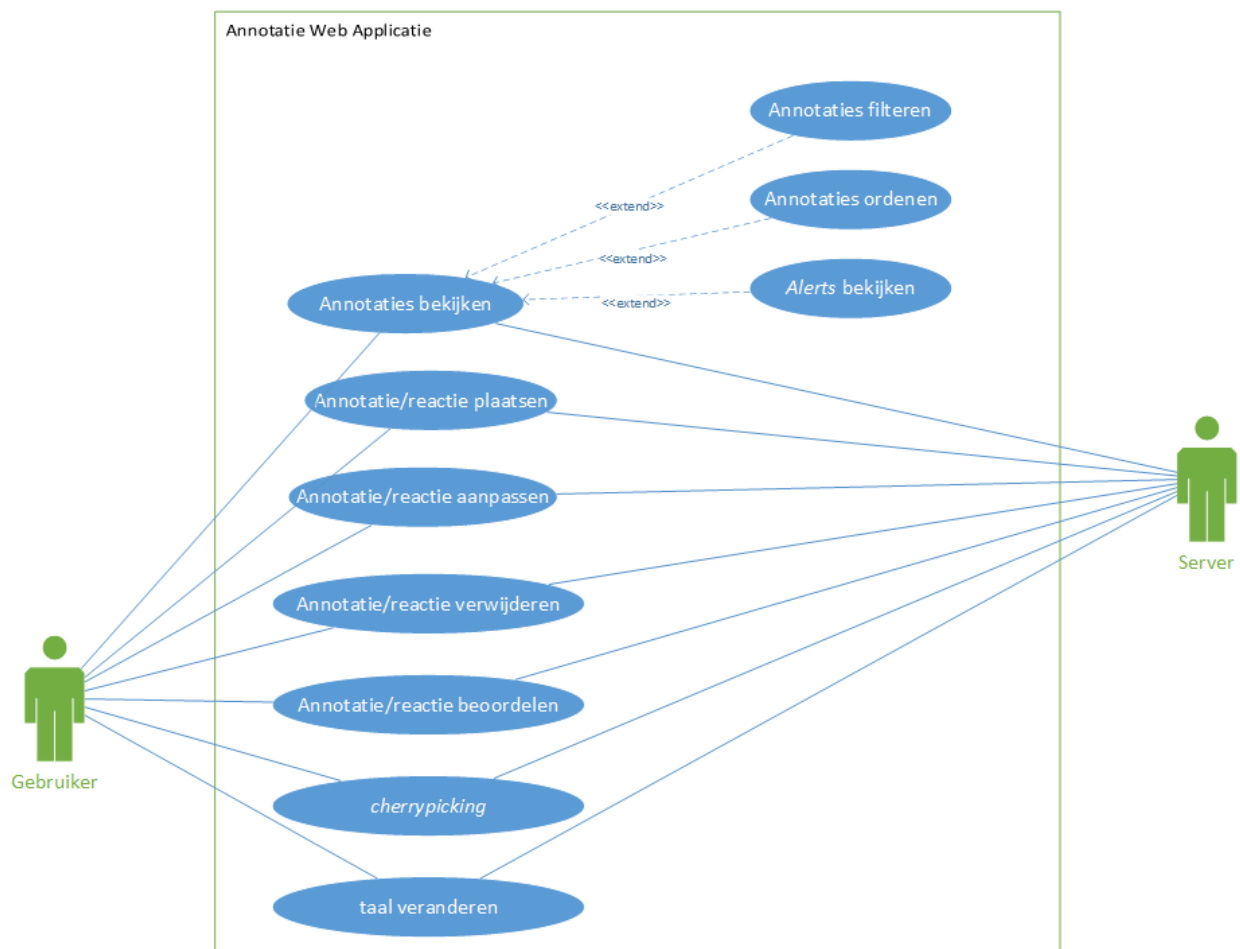
Actoren: primair: gebruiker

Precondities: Rechts bovenaan de zijbalk kan de gebruiker op "Nederlands" (*by default*) klikken om een overzicht met taalkeuzes te krijgen. Vervolgens moet er verbinding zijn met het internet en de databank om de *sidebar* leeg te halen en een vertaalde *sidebar* op te vullen.

Postcondities: /

Successscenario: De gebruiker krijgt een vertaalde *sidebar* na klikken op een taal in de *drop-down list*.

Alternatieve scenario's: Er is geen verbinding waardoor de *sidebar* niet hervuld kan worden, de vertaling zelf kan zonder verbinding.



Figuur 1: Use case diagram

2.3 Volledige feature list

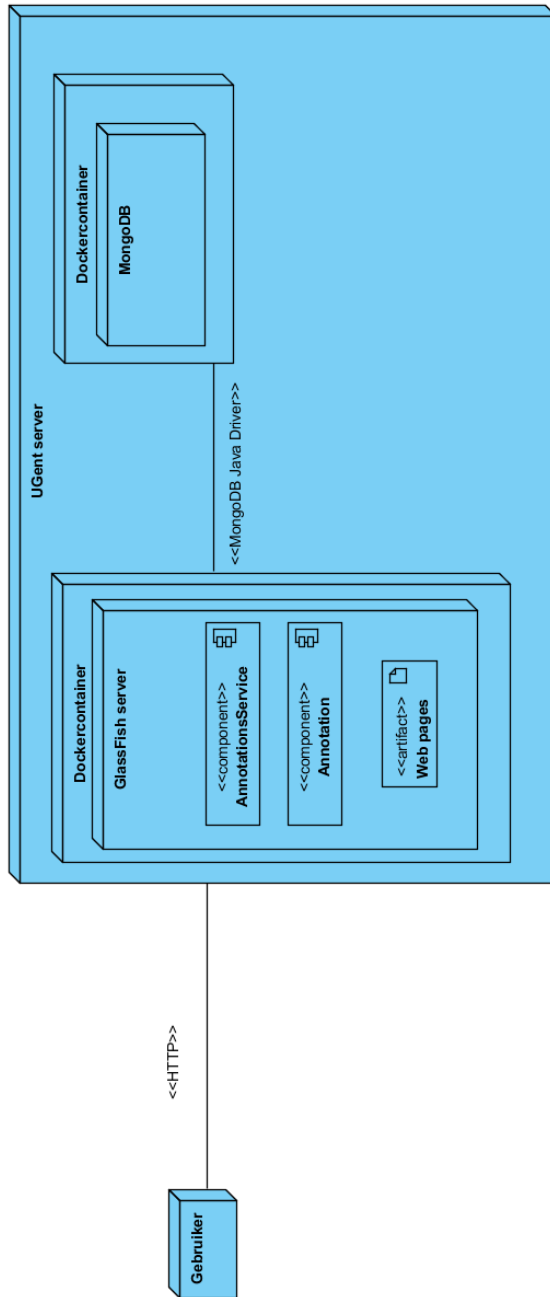
Tabel 1: Backlog

NR	Feature	Sprint	Prioriteit	Complexiteit
1	De gebruiker kan de <i>sidebar</i> openen en verslepen	Sprint1	Hoog	Hoog
2	De gebruiker kan gegenereerde annotaties bekijken in de <i>sidebar</i>	Sprint1	Hoog	Laag
3	De gebruiker kan annotaties <i>cherrypicken</i> , <i>upvoten</i> en <i>downvoten</i>	Sprint1	Hoog	Hoog
4	De gebruiker kan zelf annotaties en reacties aanmaken.	Sprint1	Hoog	Hoog
5	Annotaties en reacties worden bijgehouden in de databank.	Sprint2	Hoog	Hoog
6	De gebruiker kan annotaties of reacties aanpassen of verwijderen.	Sprint2	Hoog	Hoog
7	De <i>sidebar</i> werkt op verschillende browsers.	Sprint2	Laag	Laag
8	Er is een overzichtspagina met filter- en ordeningsfunctie.	Sprint2	Hoog	Hoog
9	Integratie met bestaande <i>webslides</i> is mogelijk.	Sprint2	Hoog	Hoog
10	De gebruiker kan zijn favorieten opvragen (<i>cherrypicking</i>)	Sprint2	Hoog	Laag
11	De gebruiker kan kiezen tussen Engels en Nederlands.	Sprint3	Laag	Hoog
12	De gebruiker kan meldingen voor de voorbije week opvragen.	Sprint3	Laag	Laag
13	De professor krijgt moderatorfunctionaliteit.	Sprint3	Laag	Hoog
14	De gebruiker kan navigeren naar de <i>slide</i> met een welbepaalde annotatie.	Sprint3	Hoog	Laag
15	De gebruiker kan <i>tooltips</i> bekijken voor uitleg.	Sprint3	Hoog	Hoog

Hoofdstuk 3: Systeemarchitectuur

3.1 High-level systeemmodel

Er is gekozen om Docker-containers te gebruiken als productie-omgeving. Deze lichtgewicht virtuele machine draait een versie van een GlassFish-server, waarin de webpagina's zich bevinden die naar de browser van de gebruiker gestuurd worden. Deze server zal ook reageren op oproepen om annotaties te laden en communiceren met de *MongoDB*-databank, die ook in een Docker-container geplaatst wordt. *MongoDB* is een *NoSQL*-databank. Het *deployment* diagram wordt weergegeven op Figuur 2 onderaan deze alinea.



Figuur 2: Deployment-diagram

3.2 Klassendiagrammen

De frontend is opgesplitst in klassen en modules naargelang de functionaliteit. Het klassendiagram in Figuur 3 bevat de klassen *Annotation*, *Reaction* en *Voter*. Deze zitten allen in de *Component*-module.

Tabel 2: Leden Annotation-klasse

Leden van Annotation	Beschrijving	
id	<i>id</i> is een unieke identificatie dat gebruikt wordt voor de AJAX-calls (<i>get</i> , <i>post</i> , <i>put</i> , <i>delete</i>) met de databank. Bij aanmaak is de <i>id</i> nog null, want het wordt pas ingevuld na interactie met de databank.	
presentationId	<i>presentationId</i> stelt de unieke naam van de presentatie van <i>webslides</i> voor.	
slideNumber	<i>slideNumber</i> verwijst naar een specifieke <i>webslide</i> in de presentatie. Dit komt overeen met een id in de HTML van een <i>webslide</i> en kan dus zowel getallen als tekst bevatten.	
title	<i>title</i> is de titel die door de gebruiker aan de annotatie wordt gegeven.	
element	<i>element</i> is een HTML-element waarop de annotatie van toepassing is. Dit kan optioneel geselecteerd worden.	
op	<i>op</i> (<i>original poster</i>) is de gebruiker die de annotatie opstelde.	
content	<i>content</i> werd als anoniem object opgesteld omdat een eigen klasse onnodig is. Dit omvat de inhoud van een annotatie.	
	commentary	<i>commentary</i> is de commentaar die door de <i>op</i> werd ingegeven.
	markType	<i>markType</i> staat <i>by default</i> ingesteld op “fluo”, maar zou als uitbreiding zelf gekozen kunnen worden en stelt de manier van weergeven van het geselecteerde element voor. Andere voorgestelde mogelijkheden zijn onderlijning en omkadering.
	category	<i>category</i> is de soort annotatie die gemaakt werd. Er is momenteel enkel keuze uit “ <i>Extra</i> ”, “ <i>Question</i> ” en “ <i>Remark</i> ”.
contentTags	<i>contentTags</i> zijn de <i>tags</i> die meegegeven worden door de <i>op</i> bij creatie. De standaardtags “ <i>important</i> ”, “ <i>solved</i> ”, “ <i>info</i> ” en “ <i>example</i> ” worden gekleurd weergegeven, maar er kunnen steeds nieuwe tags toegevoegd worden zonder kleur.	
view	<i>view</i> is de zichtbaarheid en kan op “ <i>private</i> ” en “ <i>public</i> ” ingesteld worden.	
cherrypicking	<i>cherrypicking</i> stelt een dynamische favorietenlijst voor. Elke gebruiker die een annotatie wilt bijhouden, maakt ze tot favoriet. Hierdoor bevat <i>cherrypicking</i> alle gebruikers die in die specifieke annotatie geïnteresseerd zijn.	
rating	<i>rating</i> is een anoniem object, stelt het beoordelingssysteem voor van een annotatie en wordt bij aanmaak ingesteld op een lijst met nul <i>upvotes</i> en <i>downvotes</i> .	

	thumbsUp/ thumbsDown	<i>thumbsUp/thumbsDown</i> bevat alle gebruikers die een <i>upvote/downvote</i> gaven.
date	<i>date</i> is de datum van plaatsing en wordt automatisch ingevuld.	
reactions	<i>reactions</i> bevat de reacties van andere gebruikers of de <i>op</i> zelf en is oorspronkelijk leeg. Deze behoren tot de <i>Reaction</i> -klasse.	
status	<i>status</i> bevat oorspronkelijk de waarde “ <i>used</i> ” om aan te duiden dat het nog geen aanpassingen kreeg of nog niet verwijderd werd. Wanneer de eerste aanpassing wordt gedaan, verandert de status naar “ <i>changed</i> ” en bij verwijdering verandert ze naar “ <i>removed</i> ”.	
upvote/downvote/ removeVote	Via <i>upvote/downvote/removeVote</i> worden <i>upvote/downvote/removeVote</i> uit <i>Voter</i> opgeroepen.	
deleteReaction	<i>deleteReaction</i> zorgt voor het verwijderen van reacties uit <i>reactions</i> van een annotatie.	
cherrypick	Via <i>cherrypick</i> worden gebruikers in en uit de favorietenlijst <i>cherry-picking</i> gezet.	

Tabel 3: Leden Reaction-klasse

Leden van Reaction	Beschrijving	
person	<i>person</i> is de gebruiker die de reactie opstelde.	
date	<i>date</i> is de datum van plaatsing en wordt automatisch ingevuld.	
rating	<i>rating</i> is een anoniem object, stelt het beoordelingssysteem voor van een reactie en wordt bij aanmaak ingesteld op een lijst met nul <i>upvotes</i> en <i>downvotes</i> .	
	thumbsUp/ thumbsDown	<i>thumbsUp/thumbsDown</i> bevat alle gebruikers die een <i>upvote/downvote</i> gaven.
text	<i>text</i> is de reactie die door <i>person</i> werd ingegeven.	
status	<i>Status</i> bevat oorspronkelijk de waarde “ <i>used</i> ” om aan te duiden dat het nog geen aanpassingen kreeg of verwijderd werd. Wanneer de eerste aanpassing wordt gedaan, verandert de status naar “ <i>changed</i> ” en wanneer het verwijderd wordt naar “ <i>removed</i> ”.	
upvote/downvote/ removeVote	Via <i>upvote/downvote/removeVote</i> worden <i>upvote/downvote/removeVote</i> uit <i>Voter</i> opgeroepen.	
hasUpvoted	<i>HasUpvoted</i> roept <i>HasUpvoted</i> op uit <i>Voter</i> .	
hasDownvoted	<i>hasDownvoted</i> roept <i>hasDownvoted</i> op uit <i>Voter</i> .	

Voter bevat statische methodes waar zowel *Annotation* als *Reaction* gebruik van maken om duplicatie te vermijden. Op vlak van werking is er geen verschil tussen *upvotes* en *downvotes* bij annotaties en reacties waardoor ze voor het grootste deel dezelfde hulpmethoden kunnen gebruiken. Hierbij wordt er gecontroleerd of de gebruiker zijn eigen annotatie of reactie niet probeert te beoordelen.

Tabel 4: Leden Voter-klasse

Leden van Voter	Beschrijving
upvote/downvote/ removeVote (statisch)	Via <i>upvote/downvote/removeVote</i> worden namen toegevoegd/verwijderd uit de <i>thumbsUp</i> - en <i>thumbsDown</i> -lijsten in <i>rating</i> .
hasUpvoted	<i>HasUpvoted</i> checkt of de gebruiker voorkomt in de <i>thumbsUp</i> -lijst in <i>rating</i> .
hasDownvoted	<i>HasDownvoted</i> checkt of de gebruiker voorkomt in de <i>thumbsDown</i> -lijst in <i>rating</i> .

De *LocalData*-module bevat de gelijknamige klasse *LocalData*. Hierin zitten verscheidene methodes om lokaal de annotaties te kunnen bijhouden en bewerken. Op deze manier hoeft men niet steeds alles op te halen van de databank, maar toch wordt er op de nodige momenten data ververst. Uit de *annotationList* van *LocalData* kunnen specifieke annotaties opgevraagd worden door het *id* mee te geven dat werd opgegeven in de databank na creatie. De reacties krijgen geen *id* in de databank, maar hun plaats in de lijst *reactions* van een annotatie blijft steeds gelijk en wordt daarom als *id* beschouwd. Ook de *LocalData*-klasse kan teruggevonden worden in het klassendiagram op Figuur 6.

Tabel 5: Leden LocalData-klasse

Leden van LocalData	Beschrijving
annotationList	<i>annotationList</i> is een lokale lijst met alle annotaties na een laatste <i>refresh</i> uit de databank.
getAnnotations	Via <i>getAnnotations</i> wordt de <i>annotationList</i> opgehaald.
addAnnotation	<i>addAnnotation</i> voegt een nieuwe annotatie toe aan de lokale <i>annotationList</i> . De nieuwe annotatie wordt niet in de databank gestoken via deze methode.
clearData	<i>annotationList</i> kan geleegd worden via <i>clearData</i> .
findAnnotationByHTMLId/ findReactionByHTMLId	Via <i>findAnnotationByHTMLId</i> / <i>findReactionByHTMLId</i> kan een annotatie/reactie opgevraagd worden uit de lokale data.

In de *Drawer*-module zitten methodes die reacties en annotaties maken en uittekenen. Er bestaan twee formaten van de annotaties: ze kunnen geminimaliseerd zijn waarbij ze enkel hun *op*, *title*, *category*, *contentTags* en het aantal reacties erop weergeven. Dit geminimaliseerd formaat is de standaardweergave van een annotatie. Wanneer de annotatie gemaximaliseerd is, zijn ook de *commentary*, *reactions* (waaronder *person* en *text*) en *rating* zichtbaar. Enkel in dit formaat kunnen extra reacties toegevoegd worden. Daarnaast kan er aan de kleur van de symbolen ook gezien worden of de ingelogde gebruiker zelf een *upvote* of *downvote* gaf of die annotatie als favoriet beschouwt. Op zelfgeschreven annotaties en reacties is er een *edit*- en *delete*-knop aanwezig. Een professor krijg moderatorfunctionaliteit en kan vervolgens de *delete*-knop op elke annotatie of reactie gebruiken. Deze module wordt in het klassendiagram op Figuur 4 weergegeven.

Tabel 6: Leden Drawer-module

Leden van Drawer	Beschrijving
voteColor/favoriteColor/iconColor	<i>voteColor/favoriteColor/iconColor</i> bepalen de standaardkleuren van respectievelijk de duimen, sterren en algemene iconen.
TAGCOLORS	<i>TAGCOLORS</i> is een lijst die de standaardtags mapt op hun overeenkomstige kleuren.
drawAnnotations/ drawAnnotationsMax/ drawAnnotationsMin	Alle annotaties die meegegeven worden aan <i>drawAnnotations</i> worden uitgetekend in minimale versie <i>by default</i> . Wanneer je erna de annotatie wilt maximaliseren/minimaliseren, worden <i>drawAnnotationsMax/drawAnnotationsMin</i> gebruikt om de reeds uitgetekende HTML-code aan te passen. Deze methoden roepen <i>makeContent/makeContentMax</i> en <i>fillTags</i> op. De maximale versie van een annotatie heeft ook nog <i>fillReactions</i> en <i>makeNewReaction</i> nodig.
makeNewReaction	Wanneer de annotatie in maximaal formaat getoond wordt, staat er onderaan een extra tekstvak om reacties toe te voegen. Dat tekstvak wordt getekend door <i>makeNewReaction</i> .
drawReaction	Nadat een nieuwe reactie opgesteld is en doorgestuurd wordt met een AJAX-call, wordt ze uitgetekend in HTML via <i>drawReaction</i> en roept ze daarbij <i>makeReaction</i> op. <i>drawReaction</i> verwacht het <i>annotation</i> -object waarvan de reactie afkomstig is, het <i>reaction</i> -object dat moet worden gevisualiseerd en een HTML-object als <i>target</i> waar de nieuwe HTML-reactie aan moet worden toegevoegd.
fillReactions	<i>fillReactions</i> splitst de lijst <i>reactions</i> van een annotatie op in aparte <i>reaction</i> -objecten en tekent ze elk uit in HTML in de HTML-annotatie via <i>makeReaction</i> .
makeReaction	<i>makeReaction</i> krijgt een <i>reaction</i> -object mee en zal aan de hand hiervan een HTML-container teruggeven die dan aan de HTML-pagina kan worden toegevoegd. De tweede parameter is een nummer, dat duidt op de hoeveelste reactie dit is op de annotatie en wordt gebruikt als id.
makeContent/ makeContentMax	Bij een geminimaliseerd formaat (dus ook <i>default</i> -versie) wordt de inhoud van de annotatie via <i>makeContent</i> opgebouwd, anders wordt <i>makeContentMax</i> gebruikt. Deze methode zorgt voor de HTML-opbouw van de binnenste delen van de HTML-annotatie.
fillTags	<i>fillTags</i> splitst <i>contentTags</i> op in aparte woorden en tekent ze uit in HTML-code in de HTML-annotatie, die als <i>target</i> wordt meegegeven.
refreshRatingButtons	<i>refreshRatingButtons</i> is een hulpmethode die gebruikt wordt om naast de duimen het bijhorende aantal gebruikers te plaatsen uit de <i>rating</i> -lijst van de meegegeven annotatie of reactie.
refreshRatingButtons- Style	<i>refreshRatingButtonsStyle</i> stelt de juiste kleur in van de <i>up</i> - en <i>downvote</i> -knoppen (duimen). Dit gebeurt aan de hand van de variabelen <i>voteColor</i> en <i>iconColor</i> .
refreshFavoriteButtons- Style	<i>refreshFavoriteButtonsStyle</i> stelt de juiste kleur in van de <i>favorite</i> -knop. Dit gebeurt aan de hand van de variabelen <i>favoriteColor</i> en <i>iconColor</i> .

favorite	Via <i>favorite</i> wordt het stericoon van kleur veranderd zodat de gebruiker ziet of hij de annotatie aan zijn favorieten toegevoegd heeft en wordt de methode <i>cherrypick</i> van de annotatie opgeroepen.
annotationUpvote/ annotationDownvote/ reactionUpvote/ reactionDownvote	Deze methodes worden opgeroepen bij het klikken op een duimicoon. Ze zorgen voor het aanpassen van de kleuren van de duimen en roepen de bijhorende methodes van het annotatie- of reactieobject op om die annotatie of reactie te beoordelen. Deze methodes passen enkel het aangeklikte icoon aan terwijl <i>setAnnotationByttonsStyle</i> alle iconen in de volledige HTML-pagina bij het opstarten of maximaliseren van een annotatie zal controleren op de juiste kleur.
maximize/minimize	Wanneer op de knoppen met een plus- of minicoon, linksboven de annotaties, geklikt wordt, wordt de methode <i>maximize/minimize</i> opgeroepen die de <i>drawAnnotationsMax/drawAnnotationsMin</i> oproepen om de reeds uitgetekende HTML-code aan te passen.
annotationEdit/ ReactionEdit	De gebruiker kan steeds zijn eigen annotaties/reacties aanpassen door op de <i>edit</i> -knop te klikken in de gemaximaliseerde versie van de annotatie. Deze knop roept <i>annotationEdit/ReactionEdit</i> op.
ReactionDelete/ annotationDelete	De gebruiker kan steeds zijn eigen annotaties/reacties verwijderen door op de <i>delete</i> -knop te klikken in de gemaximaliseerde versie van de annotatie. Deze knop wordt door de professor op elke annotatie/reactie teruggevonden. De <i>delete</i> -knop roept <i>annotationDelete/ReactionDelete</i> op.
drawError/ drawReactionError	<i>drawError</i> is een algemene methode waarmee fouten kunnen weergegeven worden aan de gebruikers. Er komt een gekleurde balk tevoorschijn, met extra uitleg, die door de gebruiker gesloten kan worden. Deze methode wordt gebruikt in het formulier wanneer een annotatie gemaakt of aangepast moet worden. <i>drawReactionError</i> tekent een icoon met uitroepteken zodat de gebruiker ziet dat er iets verkeerd is gebeurd. Dit wordt gebruikt bij het maken of aanpassen van een reactie.
drawEmptyAnnotations	Wanneer er door slechte connectie geen annotaties opgehaald kunnen worden van de databank, of er geen annotaties zijn om weer te geven, wordt er een melding in de plaats gegeven door <i>drawEmptyAnnotations</i> .
setToDefault	Wanneer de gebruiker de <i>sidebar</i> sluit en terug opent, het formulier afsluit, vanuit een annotatie naar de juiste <i>slide</i> springt of terugkeert naar de <i>default view</i> , worden via <i>setToDefault</i> zowel het formulier geledigd als de oude annotaties weggehaald om alles opnieuw te tekenen zoals de begintoestand.
TagNavigation	<i>TagNavigation</i> wordt opgeroepen wanneer er geklikt wordt op een <i>tag</i> in een annotatie. Er wordt vervolgens genavigeerd naar het <i>overview</i> -tabblad met als zoekterm de geklikte <i>tag</i> . Zo kunnen alle soortgelijke annotaties gevonden worden.
showAnnotationOnDia	<i>showAnnotationOnDia</i> zorgt ervoor dat het HTML-element, die bij een bepaalde annotatie hoort, aangeduid wordt op de <i>webslide</i> . Afhankelijk van het formaat (geminimaliseerd of gemaximaliseerd) wordt het element respectievelijk omkaderd of gefluoriseerd.

hideAnnotationOnDia	<i>hideAnnotationOnDia</i> zorgt ervoor dat het HTML-element, dat bij een bepaalde annotatie hoort, niet langer aangeduid wordt op de <i>webslide</i> .
giveValidElement	<i>giveValidElement</i> geeft een HTML-element terug die bij de meegeven annotatie hoort.
startLoadingAnimation	<i>startLoadingAnimation</i> start de animatie die aanduidt dat iets aan het laden is.
stopLoadingAnimation	<i>stopLoadingAnimation</i> stopt de animatie die aanduidt dat iets aan het laden is.
getScaling	Afhankelijk van de positie in de <i>sidebar</i> kan de schaling van de <i>webslides</i> via een ander element berekend worden om zo de grootte of functionaliteit van de <i>sidebar</i> erop af te stemmen. De berekende schaling is anders bij weergave van één slide, bij het verkleinen van het browsertabblad of bij het openen van de console.
openHistoryContainer	Wanneer een annotatie of reactie aangepast is, bevat het in maximale vorm een klokicoon. Indien de gebruiker op dat icoon klikt, wordt <i>openHistoryContainer</i> opgeroepen en opent dat een venster met de geschiedenis van het gevraagde element.
closeHistoryContainer	Wanneer de gebruiker de bewerkingsgeschiedenis van een element wil sluiten, kan er op het klokicoon of op het venster geklikt worden om <i>closeHistoryContainer</i> op te roepen en dat venster te sluiten.

De *Caller*-module bevat de *Caller*-klasse en maakt interactie met de databank via asynchrone *AJAX-calls* (*get*, *post*). Aan een object van de *Caller*-klasse kunnen methodes worden meegegeven die voor en na een asynchrone *call* worden uitgevoerd. Aan iedere methode van de *Caller*-klasse kunnen *callback*- en *fail*-methodes worden meegegeven die na de asynchrone *call* zullen uitgevoerd worden naargelang de *call* respectievelijk is gelukt of een fout is opgetreden. Er wordt geen *delete-call* uitgevoerd omdat annotaties en reacties bij verwijdering in de databank gehouden worden, zodat bij weergave de *flow* behouden blijft. Een *caller*-object heeft steeds een *LocalData*-object nodig. Deze module wordt in het klassendiagram op Figuur 6 weergegeven.

Tabel 7: Leden *Caller*-module

Leden van <i>Caller</i>	Beschrijving
<i>localData</i>	<i>localData</i> is een object van de <i>LocalData</i> -klasse waarin de <i>Caller</i> de opgehaalde annotaties in het lokale geheugen zal opslaan. Een <i>localData</i> -object is nodig om alle gegevens lokaal bij te kunnen houden voor interactie met de databank.
<i>before</i>	Alles meegegeven aan de <i>before</i> -methode wordt voor iedere <i>call</i> sowieso uitgevoerd.
<i>after</i>	Alles meegegeven aan de <i>after</i> -methode wordt na iedere <i>call</i> sowieso uitgevoerd.
<i>serverAddress</i>	<i>serverAddress</i> krijgt de gelijknamige waarde van het <i>serverSettings</i> -element.
<i>restUrl</i>	<i>restUrl</i> krijgt de gelijknamige waarde van het <i>serverSettings</i> -element.

url	<i>url</i> bevat de volledige URL waar de <i>AJAX-calls</i> naartoe verlopen en bestaat uit de concatenatie van <i>serverAddress</i> en <i>restURL</i> .
loadAnnotationsFromServer	<i>loadAnnotationsFromServer</i> haalt alle annotaties op via een <i>get-call</i> in <i>JSON</i> -formaat, splitst ze op en steekt ze in annotatie-objecten. Bij succes wordt de <i>callback</i> opgeroepen, bij falen wordt de <i>fail</i> opgeroepen.
sendAnnotationToServer	<i>sendAnnotationToServer</i> stuurt een <i>post-call</i> om een annotatie toe te voegen aan de databank. Bij succes wordt de <i>callback</i> opgeroepen, bij falen wordt de <i>fail</i> opgeroepen.
sendReaction	<i>sendReaction</i> stuurt een <i>post-call</i> om een reactie toe te voegen aan de databank en doet daarbij hetzelfde als <i>sendAnnotationToServer</i> . Bij succes wordt de <i>callback</i> opgeroepen, bij falen wordt de <i>fail</i> opgeroepen.
deleteAnnotation	<i>deleteAnnotation</i> roept <i>sendAnnotationToServer</i> op en wordt gebruikt om de <i>status</i> -wijziging naar “removed” door te geven.
deleteReaction	<i>deleteReaction</i> roept <i>sendReaction</i> op en wordt gebruikt om de <i>status</i> -wijziging naar “removed” door te geven.
sendVote	<i>sendVote</i> zendt een annotatie naar de server bij wijzigen van een beoordeling op een annotatie of reactie.

In de module *Config* worden de instellingen van het pad naar de server bijgehouden. Deze module wordt in het klassendiagram op Figuur 6 weergegeven.

Tabel 8: Leden Config -module

Leden van Config	Beschrijving	
ServerSettings	<i>ServerSettings</i> bevat de instellingen voor de server.	
	serverAddress	<i>serverAddress</i> bevat de URL waarop de server zich bevindt. Tijdens het coderen werden soms dummy-waarden gebruikt via de <i>MyJson</i> -website. Voor productie kan dit veranderd worden naar <i>http://webslides-01.project.tiwi.be</i> en voor testen met Docker kan dit op <i>Linux</i> vervangen worden door <i>http://localhost:9003</i> of op <i>Windows</i> door <i>http://192.168.99.100:9003</i> .
	restPath	<i>restPath</i> is het gedeeltelijke <i>REST service</i> -pad waar de server naar luistert.

De *sidebar* met de navigatiebalk wordt geregeld door de *Nav*-module. De zijbalk heeft een minimale absolute grootte (*default*-grootte) en kan door de gebruiker uitgetrokken worden tot gewenste grootte. De grootte wordt sowieso naar *full screen-mode* aangepast wanneer de gebruiker naar de *overview* navigeert. Deze module wordt in het klassendiagram op Figuur 4 weergegeven.

Tabel 9: Leden Nav-module

Leden van Nav	Beschrijving
GROOTTE_SIDEBAR	<i>GROOTTE_SIDEBAR</i> geeft het percentage weer van het scherm die door de <i>sidebar</i> wordt bedekt.
GROOTTE_SIDEBAR_ABSOLUUT	<i>GROOTTE_SIDEBAR_ABSOLUUT</i> bepaalt de effectieve breedte die de <i>sidebar</i> moet innemen op basis van <i>GROOTTE_SIDEBAR</i> en <i>MIN_ABSOLUTE_SIZE</i> .
MIN_ABSOLUTE_SIZE	<i>MIN_ABSOLUTE_SIZE</i> geeft de minimale absolute breedte in pixels weer van de <i>sidebar</i> .
open_sidebar	De <i>sidebar</i> wordt geopend door op het tekstvaksymbool langs de rechterkant te klikken. Dit roept <i>open_sidebar</i> op.
close_sidebar	De <i>sidebar</i> wordt gesloten door op het kruisje rechtsboven te klikken, dit roept <i>close_sidebar</i> op.
makeSidebarResizable	<i>makeSidebarResizable</i> merkt op wanneer de gebruiker het browsertabblad vergroot of verkleint zodat de <i>sidebar</i> mee kan schalen en andere nodige aanpassingen gedaan kunnen worden.
onDropDown	<i>onDropDown</i> zorgt ervoor dat het <i>drop-down menu</i> met de talen zichtbaar wordt.
closeDropDown	<i>closeDropDown</i> zorgt ervoor dat het <i>drop-down menu</i> met de talen onzichtbaar wordt.
goToOverview	<i>goToOverview</i> behoort tot de <i>show</i> -functionaliteit, maar werd apart gezet zodat het gemakkelijk opgeroepen kan worden wanneer de gebruiker klikt op een <i>tag</i> in een annotatie.
setReadyForForm	<i>setReadyForForm</i> stelt de zichtbaarheid in van de navigatieknoppen wanneer de <i>form</i> wordt geopend.
setToDefaultButtons	Wanneer de gebruiker de <i>sidebar</i> sluit en terug opent, het formulier afsluit, vanuit een annotatie naar de juiste <i>slide</i> springt of terugkeert naar de <i>default-view</i> , wordt er gesprongen naar de <i>default-view</i> en wordt de navigatiebalk hersteld met de juiste knoppen.
show	<i>show</i> wordt opgeroepen telkens er op een knop in de navigatiebalk wordt geklikt en zorgt daarbij voor het (on)zichtbaar maken van bepaalde elementen.

De *Manager*-module zorgt voor de basiszaken. Het regelt de eerste initialisatie van de *sidebar*, checkt welke gebruiker ingelogd is en waar de gebruiker zich bevindt: welke presentatie en specifieke *webslide*. Het zorgt voor instanties van de *LocalData*- en *Caller*klasse om data te kunnen doorgeven aan de databank. De *Manager* module vormt de *root* van het project en wordt in het klassendiagram op Figuur 4 weergegeven.

Tabel 10: Leden Manager-module

Leden van Manager	Beschrijving
presentationId/ getPresentationId	Na het opvragen van de huidige presentatie via <i>getPresentationId</i> , wordt het bijgehouden in <i>presentationId</i> .
slideNumber/ getSlideNumber	Na het opvragen van de huidige <i>webslide</i> van de presentatie via <i>getSlideNumber</i> , wordt het bijgehouden in <i>slideNumber</i> .
user/getUser	De ingelogde gebruiker wordt via een <i>prompt</i> naar zijn gebruikersnaam gevraagd die vervolgens wordt bijgehouden in <i>user</i> en opgevraagd kan worden via <i>getUser</i> .
getAccountFunction	Wanneer de gebruiker een professor is, moet hij moderator-functionaliteit krijgen en herkend worden. Dit gebeurt via <i>getAccountFunction</i> .
localData	Er wordt een instantie van de <i>LocalData</i> -klasse bijgehouden in <i>localData</i> waarin een <i>annotationList</i> zit met lokale annotaties.
caller	Er wordt een instantie van de <i>Caller</i> -klasse bijgehouden in <i>caller</i> , die AJAX-calls zal uitvoeren als interactie met de server.
MutationObserver	De <i>MutationObserver</i> is een <i>observer</i> die wijzigingen in de webpagina en opmaak kan detecteren. Deze variabele verwijst naar de <i>mutationObserver</i> van de <i>window</i> .
mutationObserverConfig	Dit is de configuratie van de <i>MutationObserver</i> . Deze zal enkel naar de HTML-elementen, inclusief hun attributen, waaraan hij is toegevoegd kijken en niet naar hun kinderelementen.
previousActiveSlideID	<i>previousActiveSlideID</i> wordt ingevuld met het id van de laatst actieve <i>slide</i> . Dit wordt gebruikt om te checken of de code na detectie van slidewissel niet onnodig wordt opgeroepen als het om een wissel naar dezelfde slide gaat.
Observer	In <i>Observer</i> wordt een nieuwe instantie van de <i>MutationObserver</i> aangemaakt. Deze krijgt de methode <i>detectActiveSlideChange</i> mee als parameter en zal deze uitvoeren bij het detecteren van een wijziging aan de elementen waarnaar hij luistert.
detectActiveSlideChange	Deze methode zorgt ervoor dat de <i>sidebar</i> opnieuw wordt opgevuld, maar enkel met de annotaties van de huidige actieve <i>slide</i> .
addSlideChangeMutation-Observer	<i>addSlideChangeMutationObserver</i> voegt de <i>MutationObserver</i> toe aan alle elementen met de CSS-klasse <i>slide</i> . Deze moet bij opstarten van de <i>sidebar</i> worden opgeroepen.
main	De <i>main</i> wordt opgeroepen wanneer alle bestanden zijn geladen en dit gebeurt slechts één keer.
refreshTable	<i>refreshTable</i> roept steeds <i>loadAnnotationsFromServer</i> op om alle annotaties van de server op te halen en zal dan de <i>sidebar</i> opnieuw laten tekenen.
noHtml	De gebruiker mag bij het weergeven van annotaties geen HTML-code te zien krijgen die door andere gebruikers, met slechte bedoeling, werd toegevoegd. <i>noHtml</i> wordt gebruikt om

	bij weergave van tekst, alle speciale tekens in HTML-code te vervangen.
getAnnotationsById-Overview	Uit een gegeven lijst worden via <i>getAnnotationsByIdOverview</i> alle annotaties gehaald die behoren tot een specifieke presentatie.
getAnnotationsById	Uit een gegeven lijst worden via <i>getAnnotationsById</i> alle annotaties gehaald die behoren tot een specifieke presentatie en <i>webslide</i> .
stopKeyEventPropagation	<i>stopKeyEventPropagation</i> kan aan een HTML-element worden toegevoegd en zorgt ervoor dat events van toetsaanslagen niet door het DOM worden doorgegeven in dit element. Dit wordt gebruikt bij tekstvakken in de <i>sidebar</i> zodat typen geen events van de <i>webslides</i> oproept.
idToewijzing	<i>idToewijzing</i> geeft een gegenereerd id aan elementen in het DOM die nog geen id hebben.
recursieveIdToewijzing	<i>recursieveIdToewijzing</i> is een hulpfunctie die gebruikt wordt om het DOM recursief af te lopen bij het toekennen van een id.
alertsThisWeek	Wanneer de gebruiker op het <i>alert</i> -icoon klikt, worden de nieuwe meldingen voor de voorbije week getoond. Dit bevat schrijvers van nieuwe annotaties en reacties met de titel van de <i>webslide</i> en annotatie en de mogelijkheid om naar die <i>webslide</i> te navigeren.
openAlertContainer	<i>openAlertContainer</i> opent de <i>alert</i> -container en wordt opgeroepen bij klikken op het <i>alert</i> -icoon.
closeAlertContainer	<i>closeAlertContainer</i> sluit de <i>alert</i> -container en wordt opgeroepen bij klikken op het <i>alert</i> -icoon of op het geopende venster.
changeLanguage	De <i>sidebar</i> wordt leeggemaakt en terug opgebouwd wanneer er een andere taal uit de <i>drop-down list</i> geselecteerd wordt via <i>changeLanguage</i> .
removeSidebar	<i>removeSidebar</i> leegt de sidebar na aanpassen van de taal.

De *AnnotationOverview*-module zorgt voornamelijk voor de zoekfuncties in het *overview*-tabblad waarmee er gezocht kan worden op *commentary*, *contentTags*, *category* en *cherrypicking*. Deze module wordt in het klassendiagram op Figuur 5 weergegeven.

Tabel 11: Leden *AnnotationOverview*-module

Leden van Annotation-Overview	Beschrijving
viewAnnotation	In de <i>overview</i> wordt via <i>viewAnnotation</i> de zoeklijst opgebouwd uit de <i>input</i> in de zoekbalk, zonder rekening te houden met hoofdletters. Vervolgens wordt de

	sorteringsvolgorde gecheckt en alles in kolommen uitgeschreven via <i>putResults</i> .
viewAnnotationEnter	Via “enter” wordt de zoek- <i>input</i> verstuurd. Deze methode is overbodig aangezien de zoekresultaten automatisch geüpdatet worden.
searchEverything	<i>searchEverything</i> roept voor elke term in de zoekbalk <i>helpFilterEverything</i> op. Er wordt steeds een lijst met annotaties verkregen die als <i>input</i> wordt gebruikt voor de volgende zoekterm.
helpFilterEverything	In <i>helpFilterEverything</i> worden alle helpfuncties afgelopen voor een gegeven zoekterm om elke matchende annotatie toe te voegen aan de tijdelijke annotatielijst. Er wordt geen rekening gehouden met hoofdletters in de helpfuncties.
helpTagFilterOne	Uit een gegeven annotatielijst worden via <i>helpTagFilterOne</i> alle annotaties gehaald die de gegeven <i>contentTag</i> hebben.
helpCategoryFilterOne	Uit een gegeven annotatielijst worden via <i>helpCategoryFilterOne</i> alle annotaties gehaald die tot de gegeven <i>category</i> behoren.
helpTitleFilterOne	Uit een gegeven annotatielijst worden via <i>helpTitleFilterOne</i> alle annotaties gehaald die een deel van de gegeven <i>title</i> hebben.
helpCommentaryFilterOne	Uit een gegeven annotatielijst worden via <i>helpCommentaryFilterOne</i> alle annotaties gehaald die een deel van de gegeven <i>commentary</i> hebben.
helpTextFilterOne	Uit een gegeven annotatielijst worden via <i>helpTextFilterOne</i> alle annotaties gehaald die een deel van de gegeven <i>text</i> hebben in een reactie.
helpOpFilterOne	Uit een gegeven annotatielijst worden via <i>helpOpFilterOne</i> alle annotaties gehaald die door <i>op</i> geschreven zijn.
helpPersonFilterOne	Uit een gegeven annotatielijst worden via <i>helpPersonFilterOne</i> alle annotaties gehaald die een reactie hebben die door <i>person</i> geschreven is.
helpCherryPicking	Uit een gegeven annotatielijst worden via <i>helpCherryPicking</i> alle annotaties gehaald die favoriet zijn voor een persoon uit de meegegeven personenlijst. Deze helpfunctie wordt niet gebruikt in de zoekfunctie.
favoriteOverview	Wanneer de gebruiker op het stericoon klikt, wordt <i>favoriteOverview</i> opgeroepen dat via <i>helpCherryPicking</i> de favorieten van de gebruiker zoekt.
helpView	Uit een gegeven annotatielijst worden via <i>helpView</i> alle annotaties gehaald die “ <i>public</i> ” zijn of zelfgeschreven zijn door de gebruiker en “ <i>private</i> ” zijn.
orderOverview	Afhankelijk van sorteringsvolgorde worden de indices anders berekend voor de <i>dictionary</i> en de uiteindelijke lijst anders opgebouwd via <i>orderOverview</i> .

putResults	Via <i>putResults</i> worden alle gevonden annotaties na zoeken verdeeld over een aantal kolommen en uitgetekend, <i>by default</i> in minimaal formaat.
------------	--

In de *AnnotationForm*-module zitten methodes die helpen bij de opbouw van het formulier om nieuwe annotaties toe te voegen. Wanneer *tags* worden toegevoegd door de gebruiker, krijgt elke *tag* een eigen knop. De gebruiker kan die *tags* verwijderen door de *button* aan te klikken. Wanneer de gebruiker over de elementen in de *webslide* *hoovert*, gaan ze één voor één gekleurd worden in fluo en sluit de *sidebar*. Door te klikken op zo'n element, bevestigt de gebruiker welk element bij de annotatie hoort. De gebruiker kan zijn keuze veranderen door dit proces te herhalen. Door op eender welke toets te drukken (zoals de toets 'S'), verschijnt de *sidebar* opnieuw en kan er verdergegaan worden met het schrijven van een annotatie. Naast het formulier voor de annotaties bevat dit ook de functies die gebruikt worden voor het mini-formulier van reacties. Deze module wordt in het klassendiagram op Figuur 5 weergegeven.

Tabel 12: Leden *AnnotationForm*-module

Leden van <i>AnnotationForm</i>	Beschrijving
selectEnabled/ setSelectedEnabled	<i>selectEnabled</i> bepaalt of het selecteren op een <i>webslide</i> momenteel mogelijk is.
selectedElement/ setSelectElement	<i>selectedElement</i> houdt het geselecteerde HTML-element bij.
isInvalidSelection	<i>isInvalidSelection</i> controleert of het meegegeven HTML-element gebruikt kan worden als een element voor de selectie.
isSelectedElement	<i>isSelectedElement</i> checkt of het meegegeven element hetzelfde is als het <i>selectedElement</i> .
firstHover	Wanneer de gebruiker een element wil toevoegen aan het formulier, dan wordt bij het <i>hoveren</i> op de <i>slide</i> , <i>firstHover</i> op "false" gezet. Dit gebeurt bij het kiezen van een element en na keuze wordt dit element terug op "true" gezet.
hooveredElement	<i>hooveredElement</i> houdt het element bij waar laatst over <i>gehoovert</i> werd. Op deze manier kan de kleur weggewerkt worden wanneer de gebruiker onverwachts de dia verlaat.
openForm	Wanneer de gebruiker een nieuwe annotatie wilt maken, moet hij een nieuw formulier invullen dat via <i>openForm</i> geopend wordt vanuit het <i>add</i> -tabblad.
showtag	Er wordt een <i>button</i> met <i>tag</i> toegevoegd in de <i>form</i> via <i>showtag</i> .
showTagEnter	<i>showTagEnter</i> is een <i>listener</i> op de <i>enter</i> -knop op het <i>input</i> -veld die de <i>tag</i> toevoegt in de <i>form</i> , zoals in <i>showtag</i> .
drawTagInForm	<i>drawTagInForm</i> is een hulpmethode die de opmaak en structuur van een <i>tag</i> vastlegt en gebruikt wordt bij het weergeven van <i>tags</i> in de <i>form</i> .
deletetag	Er wordt een <i>button</i> met <i>tag</i> verwijderd via <i>deleteTag</i> in de <i>form</i> .

initEventListenersToSlide	<i>initEventListenersToSlide</i> zorgt voor het kleuren van <i>webslide</i> -elementen in fluo, het bevestigen van het uiteindelijke element en het hernemen van die keuze. Dit roept <i>showSidebarWhenHover/hideSidebarWhenHover</i> op wanneer de gebruiker zich op een formulier op een specifieke <i>webslide</i> bevindt.
editStyle	<i>editStyle</i> verandert de stijl van de elementen waarover <i>gehoverd</i> wordt naar fluo. Dit zou uitgebreid kunnen worden met onderlijning en omkadering.
clearSelection	<i>clearSelection</i> maakt het onmogelijk om verder te gaan met selectie van een element.
hideSidebarWhenHover	<i>hideSidebarWhenHover</i> maakt <i>firstHoover</i> “false” en sluit de <i>sidebar</i> zodat de gebruiker de gehele <i>webslide</i> kan zien.
showSidebarWhenHover	Na klikken op een toets wordt de <i>firstHoover</i> -waarde opnieuw “true” en kan de <i>sidebar</i> opnieuw geopend worden via <i>showSidebarWhenHover</i> .
showSidebarWhenEscape	<i>showSidebarWhenEscape</i> maakt de <i>sidebar</i> terug zichtbaar wanneer in het <i>add</i> -tabblad de <i>escape</i> -knop wordt ingedrukt en het overzicht van <i>webslides</i> terug zichtbaar wordt.
submitAnnotation	Alle velden van het formulier worden ingelezen door <i>submitAnnotation</i> zodat er een annotatie-object opgesteld en doorgegeven kan worden aan de databank.
submitAnnotationAgain	Wanneer de gebruiker terugkeert naar een eerder gemaakte annotatie om die aan te passen, zal het alle info behalve <i>title</i> , <i>contentTags</i> , <i>category</i> , <i>commentary</i> , <i>view</i> en <i>element</i> behouden. De <i>status</i> van de annotatie wordt veranderd naar “changed”. Ook wordt het formulier ingevuld met de oude waarden en gebeurt het aanpassen van de annotatie via <i>submitAnnotation-Again</i> .
cancelAnnotation	De gebruiker kan het formulier verlaten via <i>cancelAnnotation</i> . Dit zorgt ervoor dat het selecteren van elementen wordt afgebroken.
postReaction	Wanneer een reactie gepost wordt in het tekstvak onder de gemaximaliseerde annotatie, wordt <i>postReaction</i> opgeroepen bij drukken op de pijl. Dit roept <i>sendReaction</i> op en tekent die reactie uit via <i>drawReaction</i> .
postReactionAgain	Als een reactie aangepast wordt, moet enkel de <i>text</i> veranderd worden en een AJAX-call gestuurd worden met de ouder-annotatie om die te overschrijven via <i>postReactionAgain</i> . De <i>status</i> van de reactie wordt veranderd naar “changed”. <i>postReactionAgain</i> roept <i>sendReaction</i> op van de <i>caller</i> .

Een laatste module, de module *Sidebar*, bevat de methodes die de HTML van de lege *sidebar* voorzien, met de mogelijkheid om verschillende talen te kiezen. Deze module zorgt ervoor dat er geen rechtstreekse aanpassingen in de HTML nodig zijn om de *sidebar* te gebruiken buiten enkele *scripts* en *CSS-tags*. Deze module wordt in het klassendiagram op Figuur 6 weergegeven.

Tabel 13:Leden Sidebar module

Leden van Sidebar	Beschrijving
drawSidebar	<i>drawSidebar</i> geeft de HTML-code terug van een lege <i>sidebar</i> .
drawOpenSidebarButton	<i>drawOpenSidebarButton</i> geeft de HTML-code terug van de knop om de <i>sidebar</i> te openen.
initIndex	<i>initIndex</i> voegt de <i>sidebar</i> en een openknop toe aan de webpagina en zorgt voor correcte invulling van de optie voor taalselectie.
uppercaseFirstLetter	<i>uppercaseFirstLetter</i> is een hulpmethode om hoofdletters toe te voegen.
changeDropdownExceptFor	<i>changeDropdownExceptFor</i> is een hulpmethode bij het invullen van de taalkeuzeselectie.
clearDropdown	<i>clearDropdown</i> is een hulpmethode bij de opmaak van de taalselectieselectie.
makeOption	<i>makeOption</i> is een hulpmethode bij het invullen van de taalkeuzeselectie.

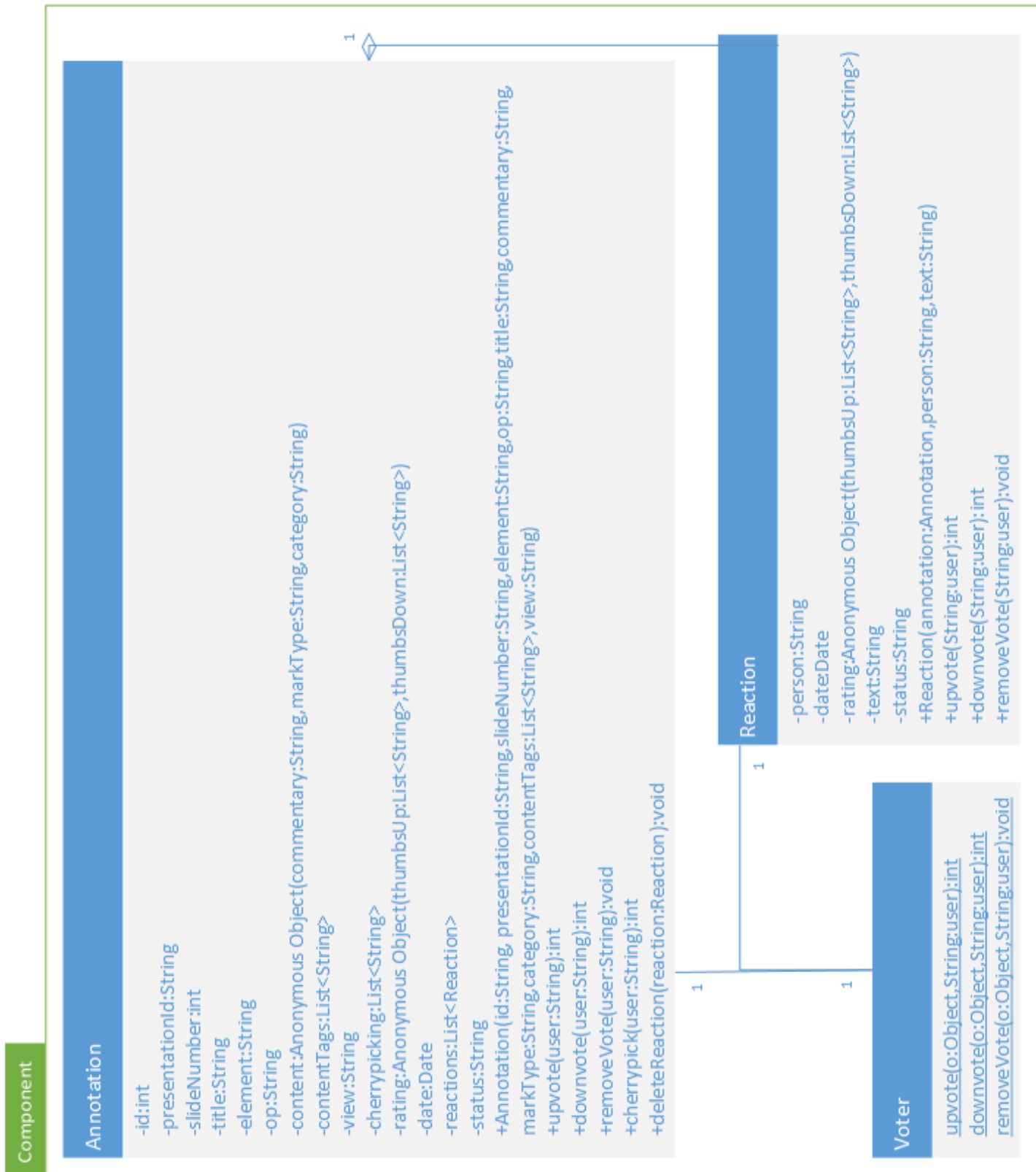
Tussen deze modules en klassen komen verschillende relaties voor.

Tabel 14: Relaties tussen klassen en modules

Aggregation (bezitter en bezit):	
Annotation en Reaction	Een annotatie kan leven zonder reacties, maar een reactie hoort altijd bij een annotatie.
Caller en LocalData	Een <i>Caller</i> -object moet steeds een <i>LocalData</i> -instantie meekrijgen in de constructor.
Association (gebruikte en gebruiker)	
Voter en Annotation, Reaction	Annotaties en reacties gebruiken beiden de statische methode uit <i>Voter</i> voor hun beoordelingssysteem.
Annotation en AnnotationOverview	In de <i>overview</i> worden alle zoekfuncties op annotaties toegepast.
Annotation, Reaction, LocalData, Caller en AnnotationForm	Bij het versturen/aanpassen van annotaties en reacties in <i>AnnotationForm</i> wordt een <i>LocalData</i> -object gebruikt om lokale (reeds opgehaalde) data aan te passen en een <i>caller</i> -instantie gebruikt voor de AJAX-calls.
Caller en Manager	Wanneer annotaties en hun reacties ververst worden in de <i>Manager</i> , wordt de <i>Caller</i> -instantie gebruikt voor de AJAX-call.
Nav en AnnotationForm, Drawer, AnnotationOverview	Bij veranderen van tabblad in de navigatiebalk, worden elementen via methodes in Drawer, Nav, AnnotationForm en AnnotationOverview (on)zichtbaar.
Annotation, Reaction, Caller, LocalData, Manager en Drawer	Voor het opvragen en uittekenen van annotaties/reacties in <i>Drawer</i> , wordt de lokale

Manager, AnnotationForm en Sidebar	data opgevraagd via <i>AJAX-calls</i> . Om de kleur van de iconen te achterhalen, wordt de <i>Manager</i> geraadpleegd.
Config en Caller	Als eerste stap zullen de methodes van <i>Manager</i> de <i>sidebar</i> moeten tekenen in de <i>webslides</i> . In <i>annotationForm</i> moet de zijbalk gesloten kunnen worden tijdens selectie van het bijpassend element. Het <i>Caller</i> -object gebruikt de <i>serverSettings</i> uit de <i>Config</i> .

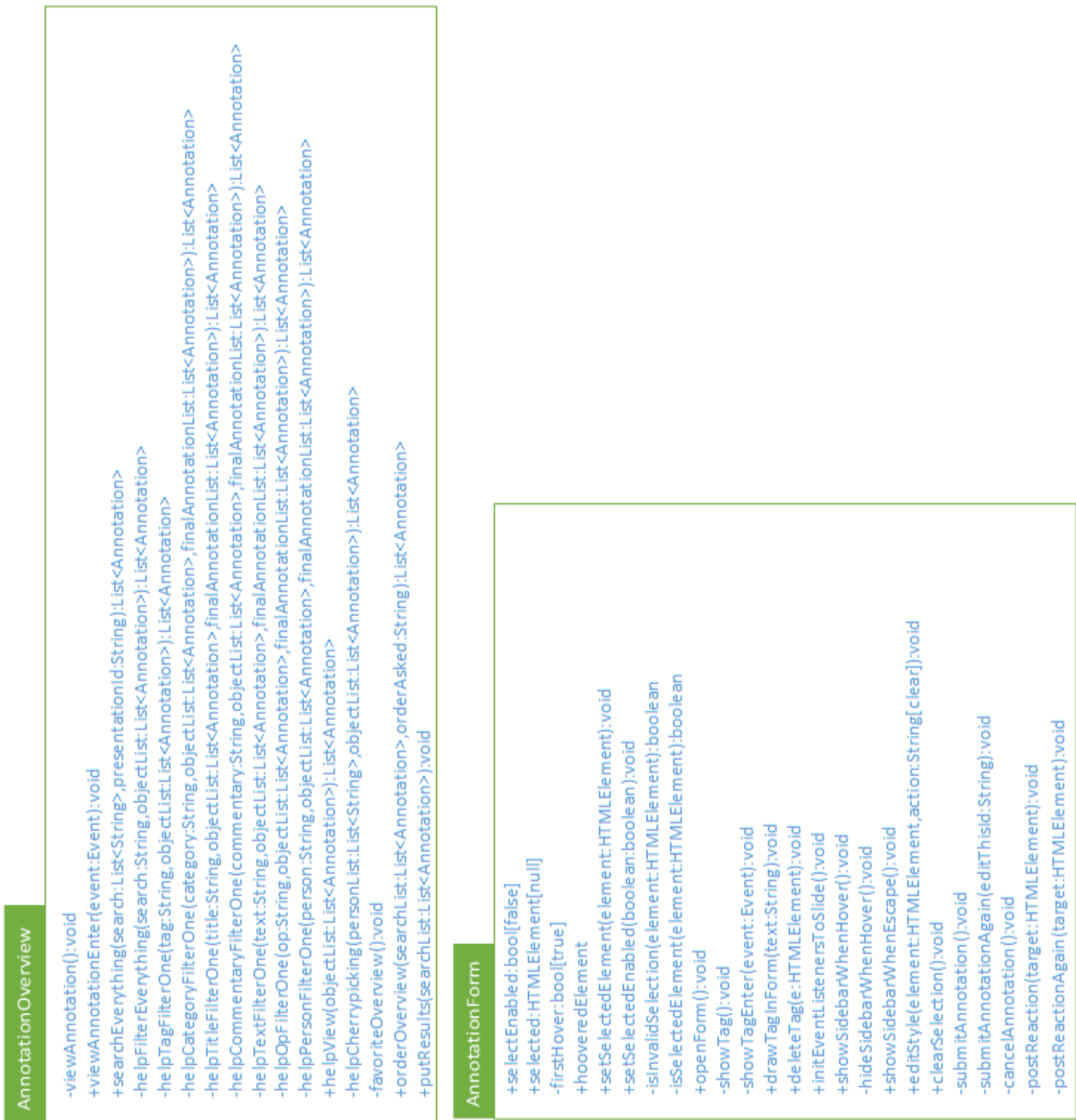
Hieronder, op Figuren 3, 4, 5 en 6 werden vier klassendiagrammen bijgevoegd waarin de relaties tussen de verschillende modules en klassen duidelijk gemaakt worden.



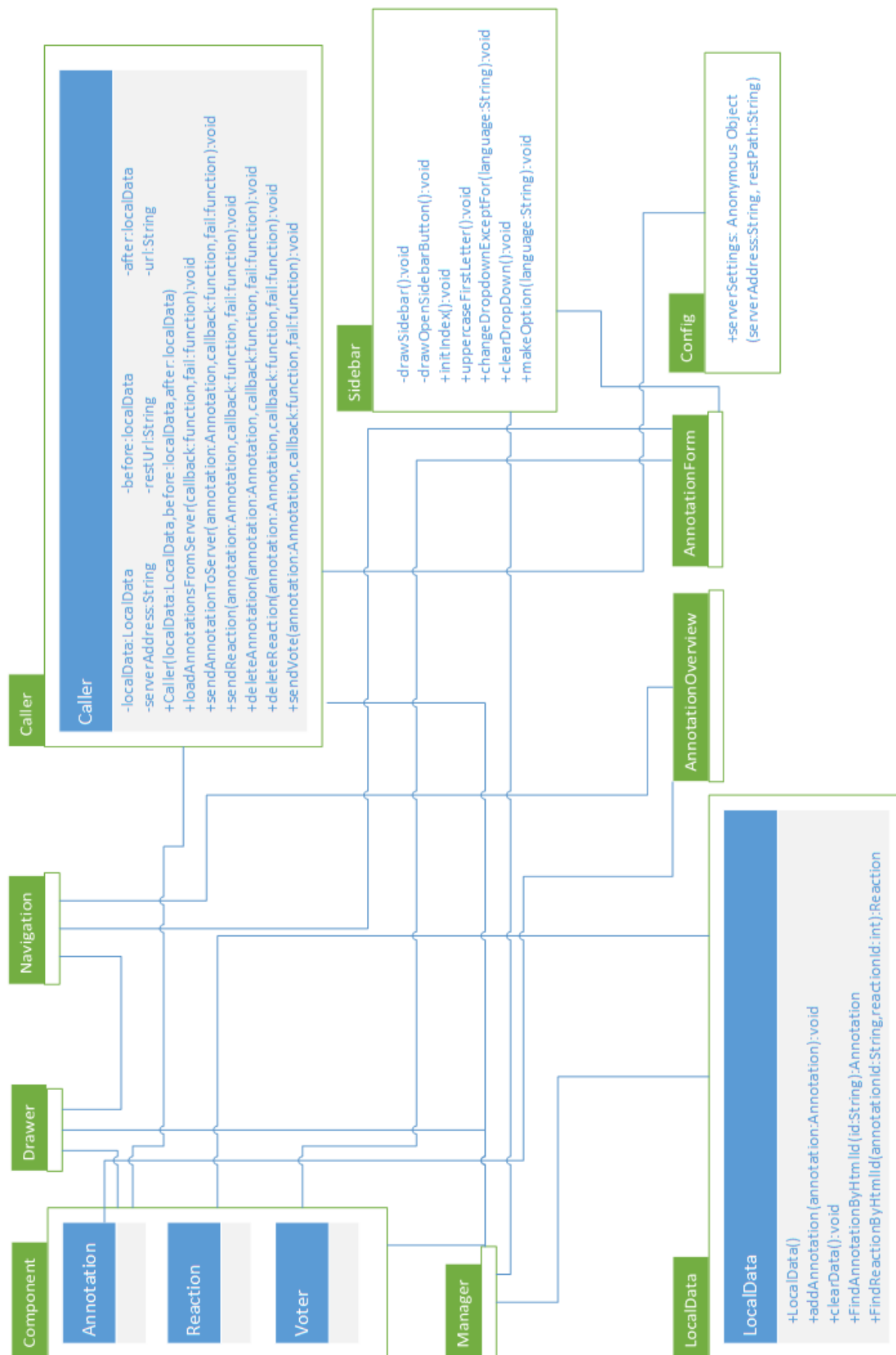
Figuur 3: Klassendiagram met inhoud van Component-module



Figuur 4: Klassendiagram met inhoud van Manager-, Drawer- en Nav-modules



Figuur 5: Klassendiagram met inhoud van AnnotationOverview- en AnnotationForm-modules



Figuur 6: Klassendiagram met inhoud van Caller-, Config-, Sidebar- en LocalData-modules

De backend heeft ook meerdere klassen om het gebruik van de annotatietool te vergemakkelijken. Deze zijn in Figuur 7 als een klassendiagram voorgesteld.

De klasse *AnnotationsService* regelt alle HTTP-requests.

Tabel 15: Leden *AnnotationsService*-klasse

Leden van <i>AnnotationsService</i>	Beschrijving
<i>getJson</i>	<i>getJson</i> haalt elke annotatie in de databank op en verstuurt die naar de aanvrager. Dit reageert op een <i>GET</i> -aanvraag.
<i>putJson</i>	<i>putJson</i> verwacht een annotatie in JSON-formaat en stuurt deze door om in de databank op te slaan. Dit reageert op een <i>POST</i> -aanvraag.
<i>deleteItem</i>	<i>deleteItem</i> verwacht een <i>id</i> van een annotatie en verwijdert die uit de databank. Dit reageert op een <i>DELETE</i> -aanvraag.
<i>getJsonAndPutOneDummy</i>	<i>getJsonAndPutOneDummy</i> maakt één dummy-annotatie aan, plaatst die in de databank en geeft vervolgens elke annotatie in de databank terug. Dit gebeurt via een <i>POST</i> -aanvraag.
<i>getJsonAndPutManyDummy</i>	<i>getJsonAndPutManyDummy</i> maakt meerdere dummy-annotaties aan, plaatst die in de databank en geeft vervolgens elke annotatie in de databank terug. Dit gebeurt via een <i>POST</i> -aanvraag.
<i>dropCollection</i>	<i>dropCollection</i> verwijdert elke annotatie in de databank. Dit gebeurt via een <i>DELETE</i> -aanvraag.

Deze klasse moet dus kunnen communiceren met de databank, hiervoor is een aparte klasse *AnnotationsCRUD* voorzien. Deze kan verbinding maken met de databank, om bv. annotaties op te halen of te verwijderen. Aangezien het belangrijk is dat deze verbindingen gesloten worden wanneer ze niet meer gebruikt worden, is gekozen voor een *singleton*-patroon.

Tabel 16: Leden *AnnotationsCRUD*-klasse

Leden van <i>AnnotationsCRUD</i>	Beschrijving
<i>instance</i>	Op elk gegeven moment zou <i>instance</i> de enigste instantie van deze klasse mogen zijn.
<i>connString</i>	<i>connString</i> is de connectiestring die gebruikt wordt om met de databank te connecteren.
<i>dbNaam</i>	<i>dbNaam</i> is de naam die men aan de databank wil geven (en staat <i>by default</i> op “BachelorproefDB”).

collNaam	<i>collNaam</i> is de naam van de collectie (te vergelijken met een <i>table</i> in SQL) (en staat <i>by default</i> op “Annotaties”).
getInstance	<i>getInstance</i> verkrijgt de instantie van de klasse.
deleteAnnotationById	<i>deleteAnnotationById</i> zoekt in de databank naar een annotatie met hetzelfde <i>id</i> dat meegegeven werd, en indien dit bestaat, wordt deze verwijderd.
getItems	<i>getItems</i> levert een lijst van elke annotatie uit de databank op.
putJson	<i>putJson</i> slaat de meegegeven annotatie op in de databank.

Voordat de annotaties opgeslagen worden, stelt men ze eerst voor als object. Hiervoor is de *Annotation*-klasse voorzien. Deze klasse dient als voorstelling van de eerdergenoemde *Annotation* van de frontend en is een opsomming van zijn leden, identiek aan Tabel 2, maar dan zonder de bijhorende methodes.

De *AnnotationsCRUD* moet vlot kunnen converteren tussen JSON, *Annotation* en *Document* (de klasse die een object in de databank voorstelt). Hiervoor is een aparte klasse voorzien: de *AnnotationsAdapter*.

Tabel 17: Leden *AnnotationsAdapter*-klasse

Leden van <i>AnnotationsAdapter</i>	Beschrijving
toAnnotation	<i>toAnnotation</i> kan ofwel een <i>Document</i> -instantie, ofwel een JSON-bestand omzetten naar <i>Annotation</i> .
toDocument	<i>toDocument</i> zet ofwel een annotatie, ofwel een JSON-bestand om naar een <i>Document</i> -instantie.
toJSON	<i>toJSON</i> kan één of meerdere annotaties omzetten naar JSON.

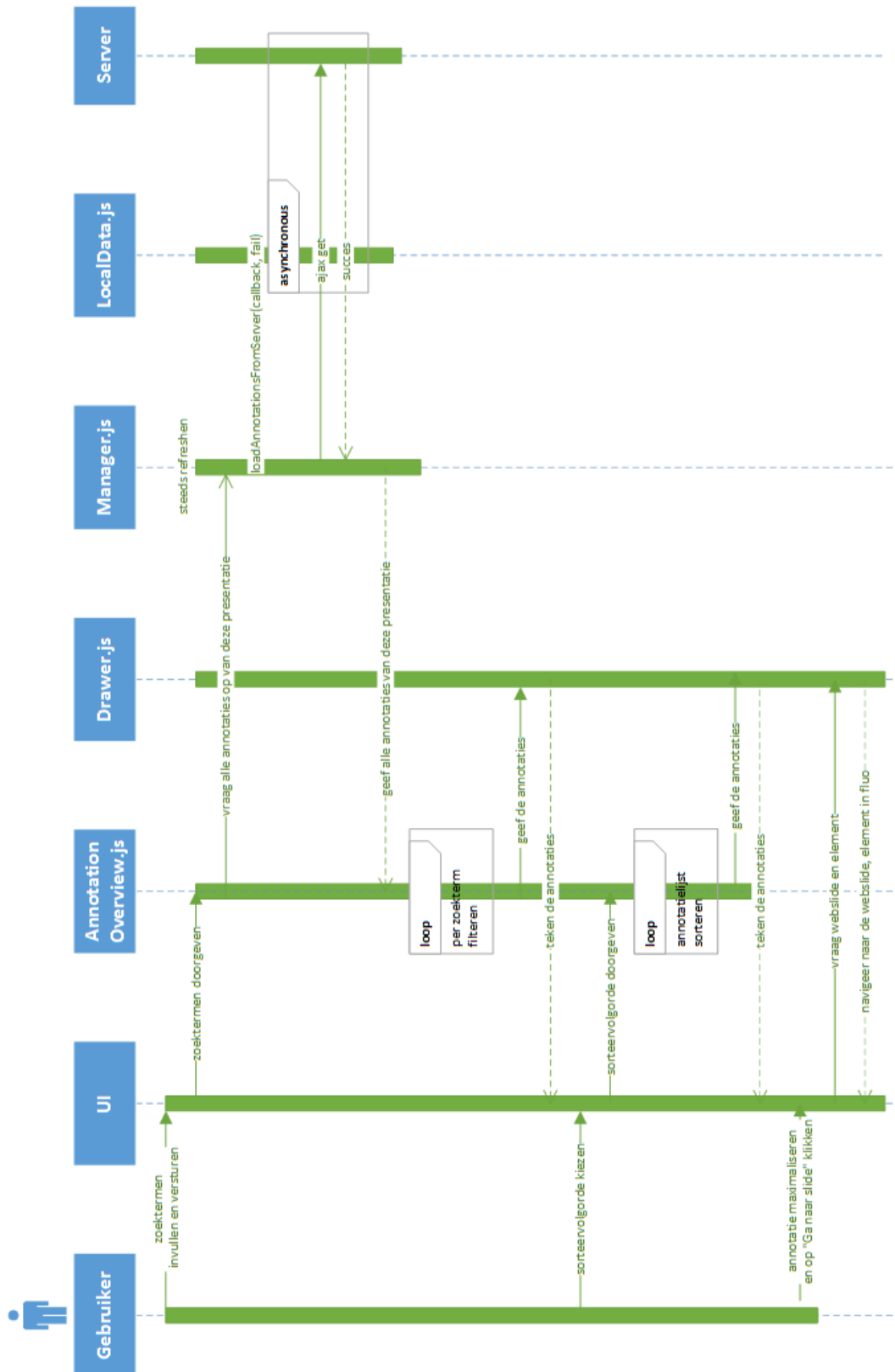


Figuur 7: Klassendiagram backend

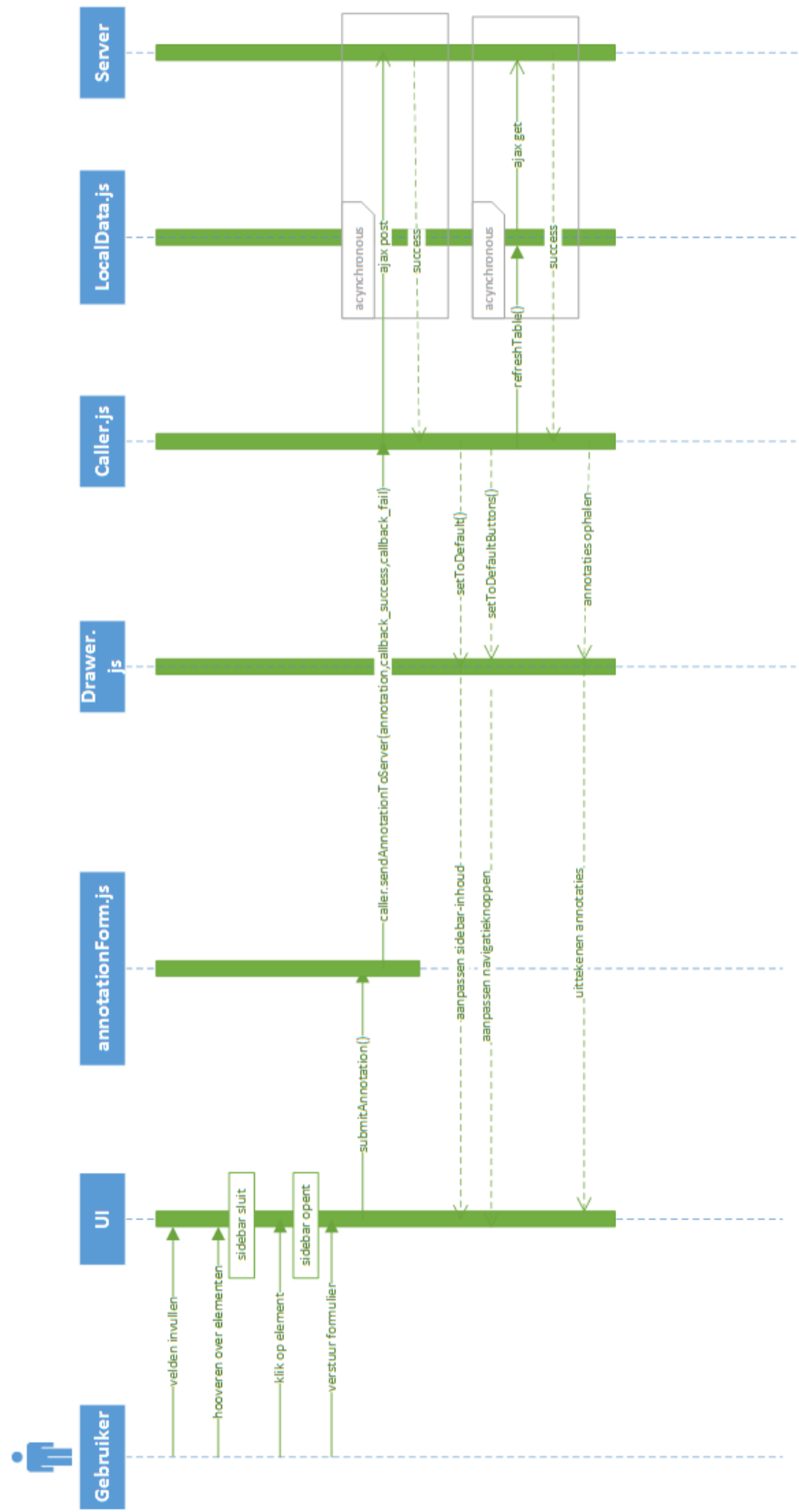
3.3 Sequentiediagrammen

Op Figuur 8 staat een sequentiediagram over het zoeken van specifieke annotaties. In het *overview*-tabblad kan de gebruiker meerdere zoektermen invullen, gescheiden door komma's. Per term wordt er een lijst aan helpfuncties overlopen, zo wordt er steeds gezocht naar een match tussen de zoekterm en de velden *op*, *person*, *commentary*, *text*, *contentTags*, *title* en *category* uit de gegeven annotaties. Het eerste zoekwoord zal filteren op alle annotaties uit de presentatie waardoor er een gefilterde annotatielijst verkregen wordt die aan de volgende term meegegeven wordt als *input*-lijst. Dit gaat zo door tot alle zoektermen overlopen zijn en er een annotatielijst verkregen wordt die matcht met elk gegeven zoekwoord. Net zoals bij zoeken via *Google Search* worden de uitgetekende annotaties geüpdatet per ingevulde letter in de zoekbalk. Wanneer er op geen enkele term gefilterd wordt, worden alle annotaties teruggegeven en wanneer er niks gevonden wordt na filtering, zal er ook niks weergegeven worden. Er zal dan een opmerking verschijnen die meldt dat er geen matches zijn. De gebruiker kan de gefilterde annotatielijst ordenen door een sorteervolgorde te kiezen uit de *drop-down list*. Deze sortering kan op datum ("NEW"), op beoordeling ("TOP", aantal upvotes verminderd met aantal downvotes), op populariteit ("HOT", aantal reacties) en alfabetisch op dianaam ("SLIDES") gebeuren. Indien de gebruiker de *webslide* bij een annotatie wilt zien, klikt hij op de "Ga naar slide"-link. Er wordt vervolgens automatisch genavigeerd naar de juiste *webslide* in de presentatie met het bijhorende element in het lichtblauw.

Op Figuur 9 staat een sequentiediagram over het invullen van het formulier. De gebruiker moet minstens de titel en commentaar invullen, anders verschijnen er foutmeldingen. Daarnaast kunnen ook zichtbaarheidsniveau, categorie en *tags* aangepast worden voor het versturen van het formulier. Er wordt samengewerkt tussen verschillende modules zodat de gemaakte annotatie naar de databank gestuurd kan worden en achteraf te zien is op de lijsten in het *default*- en *overview*-tabblad. De meeste interacties in de databank verlopen op een soortgelijke manier, waarbij de asynchrone *AJAX-calls* bij succes resulteren in het aanpassen van de weergaven in HTML.



Figuur 8: Sequentiedigram van AnnotationOverview-module



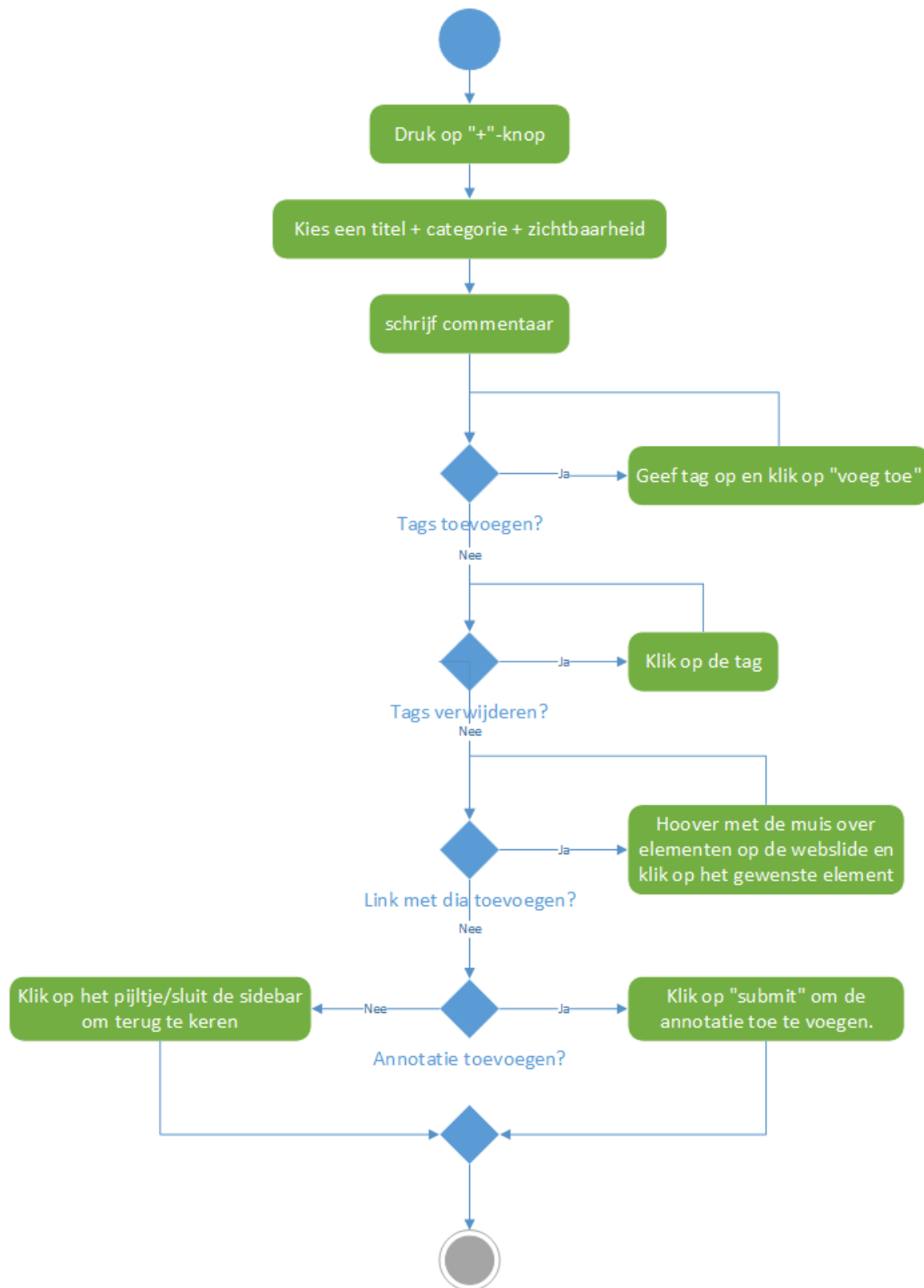
Figuur 9: Sequentiediagram van AnnotationForm-module

3.4 Databank

Er werd gekozen voor de *NoSQL*-databank *mongoDB* (*opensource*) omdat het data bijhoudt in JSON-formaat. Dit formaat werd al gebruikt in de *AJAX-calls* naar de server. Deze databank is zeer flexibel en zorgt ervoor dat de velden kunnen variëren tussen verschillende documenten. Distributie gebeurt hierdoor met gemak tussen verschillende computers.

3.5 Activiteitendiagrammen

De gebruiker kan op een specifieke *webslide* een annotatie toevoegen door te navigeren naar het *add*-tabblad (met het plusicoon). De enige verplichting is het invullen van de commentaar en titel, want versturen zonder invullen van deze velden resulteert in een foutmelding bovenaan het formulier. Zo'n melding kan door de gebruiker steeds weggeklikt worden en houdt hem/haar niet tegen om datzelfde of een aangepast formulier opnieuw door te sturen. Indien de gebruiker het formulier volledig wil invullen, zal hij een categorie (*remark (by default)*, *question*, *extra*), *tags* en zichtbaarheidsniveau (*public*, *private (by default)*) moeten kiezen. Bij de keuze van die *tags* worden alle reeds gebruikte termen uit die presentatie voorgesteld. Zo is het gemakkelijk voor de gebruiker om passende *tags* te kiezen en kan de orde bewaard worden. Ook kan hij de annotatie linken met de hele dia of een zelfgekozen *DOM*-element. Wanneer de gebruiker de muis over de bijhorende *webslide* *hoovert*, sluit de *sidebar* en zullen de elementen één voor één oplichten in het geel. Hieruit kan er een passend element gekozen worden door te klikken, deze wordt dan lichtblauw en de *sidebar* opent opnieuw. Ook zal, door om het even welke toets aan te klikken, de *sidebar* heropenen, zonder verplichte selectie. Indien nodig kan de aanduiding van het element steeds herdaan worden door opnieuw te *hoveren* en te klikken. Op Figuur 10 staat een activiteitendiagram over het invullen van het formulier.



Figuur 10: Activiteitendiagram van AnnotationForm-module

Hoofdstuk 4: Testplan

Door rekening te houden met de *use cases* werden volgende testen voorzien: toevoegen, verwijderen en aanpassen van een annotatie en een reactie. Naast deze automatische tests, werden er natuurlijk ook manuele voorzien. Hierbij werd rekening gehouden met de *pop-ups* die verschijnen wanneer de verbinding opeens zou wegvallen, de complexiteit van het eindproduct en de ondersteuning op andere *browsers*. In het testplan in tabel 18 staan de uit te voeren testen en hun scenario's.

4.1 Backend

De backend server is geschreven in Java en de functionaliteiten ervan worden getest aan de hand van *integration* testen. Hierbij worden vooral de methodes voor toevoegen en ophalen uit de databank getest.

Voor de backend zijn twee testklassen voorzien in de package *AnnotationToolTests*. De *AdapterTests*-klasse gaat de correcte werking na van de omzetting van een annotatie in JSON-formaat naar een *Annotation*-object. De *AnnotationCRUDTests*-klasse voorziet de *integration tests*, die toegang tot en operaties in de databank nagaan.

4.2 Frontend

4.2.1 Geautomatiseerd: Unit en integration testen

Er werd gebruik gemaakt van het Jest *framework* om de frontend van de webapplicatie uit te testen.

In paragraaf 6.1.6 worden de nodige installaties voor het uitvoeren van jest-testen overlopen. Volgende beschrijving gaat ervan uit dat Jest correct is geïnstalleerd. De testen bevinden zich in *sidebar/scripts/tests* en eindigen met de extensie “.test.js”. Jest zal deze bestanden automatisch herkennen. De testbestanden staan gegroepeerd in een submap van de testmap per afzonderlijke module die specifiek getest wordt, dit zijn de *Caller*-, *Component*- en *LocalData*-modules.

Het bestand *test_Environment_Setup.js* bevat code om *jQuery* te herkennen bij de testen die gebruik maken van het *jQuery framework*.

De overige bestanden in de testmap, *temp.js* en *temp.test.js* bevatten een basistest die als controle van het Jest *framework* wordt gebruikt. Deze test zou altijd moeten slagen en indien dit niet het geval is, dan is waarschijnlijk het Jest *framework* foutief geïnstalleerd.

De testen kunnen worden uitgevoerd door in de console de map *sidebar* te openen en het volgend commando in te geven:

```
npm test
```

Jest zal dan alle testen overlopen en de bijhorende berichten weergeven. Indien alle testen succesvol zijn, zullen deze met een groene kleur worden weergegeven. Testen die faalden, worden in rood weergegeven en tonen door Jest gegenereerde foutberichten.

Ieder testbestand verwacht een methode met de naam “test”. Deze methode heeft twee parameters. De eerste parameter is een stukje tekst die wordt getoond bij het uitvoeren van de test, de tweede parameter is een functie die de testmethode zal uitvoeren en die werkelijke code en vergelijkingen bevat om te controleren.

Als voorbeeld zal de test in het bestand *Caller_loadAnnotationsFromServer.test.js* worden besproken. Deze test bevindt zich in de submap “Caller” van de testmap en controleert of annotaties van de server kunnen worden gehaald, want dit behoort tot de functionaliteit van de *Caller*-klasse. Aangezien dit asynchrone uitvoer verwacht, wordt gewerkt met een *callback*-functie. In deze functie wordt gecontroleerd of de vooraf *false*-gedeclearde variabelen a, b en c nu wel *true* zijn. Het woord *done* laat weten dat Jest hierna niet meer moet wachten en de test is afgerond. Met de *expect*-methode kan worden aangegeven wat verwacht wordt. Zo wordt verwacht dat als de *callback* wordt opgeroepen, de variabelen a, b en c *true* zijn door de code “expect(variabele).toBeThruthy()”. Met “.equals()” kan men aangeven dat twee zaken gelijk moeten zijn en met “.toBeDefined()” kan man aangeven dat een bepaalde variabele moet gedefinieerd zijn.

Er wordt een nieuw object van de klasse “Caller” aangemaakt. De eerste parameter van de *constructor* verwacht een object van de klasse “LocalData”, de tweede en derde verwachten methodes die respectievelijk voor en na een bepaalde *call* worden uitgevoerd. Met de methode “caller.loadAnnotationsFromServer(...)” zal de *call* worden uitgevoerd die annotaties van de server zal ophalen. Hierna wordt de *callback* opgeroepen die de variabelen vergelijkt. Indien er iets fout is gelopen, zullen deze niet *true* zijn en zal de test falen.

```
1      import {Caller} from "../../data/Caller";
2      import {LocalData} from "../../data/LocalData";
3      import "../../test_Environment_Setup";
4
5      test("load annotation", done => {
6          let callback = () => {
7              expect(c).toBeTruthy();
8              expect(a).toBeTruthy();
9              expect(b).toBeTruthy();
10             done();
11         };
12         let local = new LocalData();
13         expect(local).toBeDefined();
14         let a = false;
15         let b = false;
16         let c = false;
17         let caller = new Caller(local, () => {a = true;}, () => {b = true; callback()});
18         expect(caller).toBeDefined();
19         expect(caller.localData).toEqual(local);
20         expect(caller.before).toBeDefined();
21         expect(caller.after).toBeDefined();
22         expect(caller.serverAddress).toBeDefined();
23         expect(caller.url).toBe(caller.serverAddress+caller.restUrl);
24         caller.loadAnnotationsFromServer(() => {c = true;}, () => {c = false;});
25     });
```

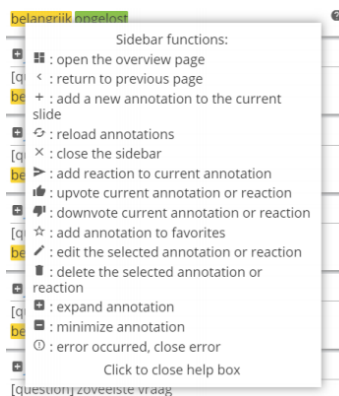
Figuur 11: Inhoud van *Caller_loadAnnotationsFromServer.test.js*

4.2.2 Manueel: Usability en Integration testen

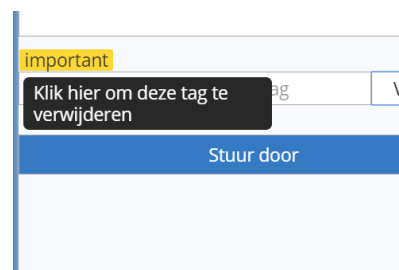
Het project werd doorheen de drie sprints aan verscheidene proefpersonen voorgelegd om te controleren of die courante versie van de annotatietool niet te complex was. Deze personen werden gevraagd enkele simpele opdrachten uit te voeren zoals het plaatsen van een annotatie, het beoordelen van een reactie, het opzoeken van een specifieke annotatie en het navigeren naar die *webslide*. Door de moeilijkheden die ze ondervonden en hun opmerkingen werden aanpassingen gedaan die leidden tot de laatste versie. Er moest een evenwicht gevonden worden tussen de simpele lay-out zonder beschrijvingen, gevraagd door de klant en een duidelijke lay-out met uitleg, gevraagd door de gebruiker. Om de website zo simpel mogelijk te laten ogen, werden beschrijvingen weggelaten en vervangen door iconen, zoals een duim voor de beoordeling en een ster als favoriet. Deze symbolen komen vaak voor op sociale media, maar zijn niet door iedereen gekend. Als oplossing werd oorspronkelijk een helpcontainer voorzien die elk icoon een korte beschrijving geeft, dit wordt weergegeven op Figuur 14. De proefpersonen raadpleegden meerdere keren die container wat het hele proces vertraagde. Om die reden werd de helpfunctie vervangen door *tooltips* waarin een korte beschrijving verschijnt vanaf de muis een seconde over dat icoon *hoovert*. (zie Figuur 12).

Wanneer de internetverbinding van de gebruiker opeens wegvalt, kan geen data meer doorgegeven of opgehaald worden van de databank. Bij het aanmaken/aanpassen/verwijderen van een annotatie of reactie zal een wegklikbare melding verschijnen (zie Figuren 13, 15, 16 en 17). Ook bij het *upvoten*, *downvoten* en toevoegen aan favorieten zullen de gemaakte wijzigingen niet doorgaan en wordt er een gepaste melding gegeven.

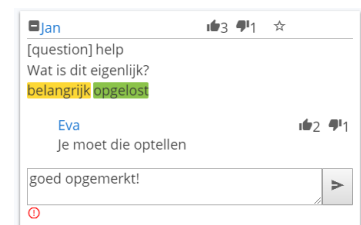
In onderstaande paragrafen worden enkele acties overlopen die gebruikt zijn om de *flow* en werking van de *sidebar* te testen.



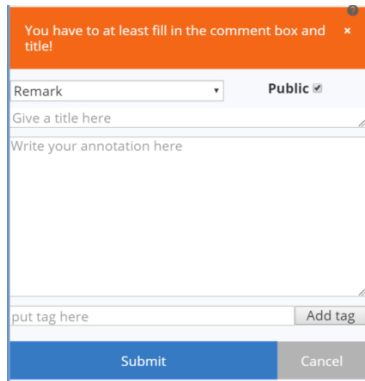
Figuur 14: Helpcontainer



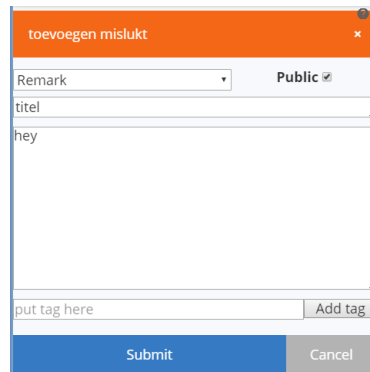
Figuur 12: Tooltip



Figuur 13: Foutmelding



Figuur 17: Foutmelding



Figuur 16: Foutmelding



Figuur 15: Foutmelding

4.2.2.1 Openen sidebar

Klik aan de rechterkant op de blauwe rechthoek met het berichticoon. De *sidebar* zou nu moeten worden weergegeven aan de rechterkant van het scherm. In eerste instantie zal, als de *server* ook draait, rechtsonderdaan een ronddraaiende animatie afspelen gedurende de tijd van de *call* naar de server en het opbouwen van de annotaties. Wanneer de server correct geladen is, worden de annotaties zichtbaar in de *sidebar*. Indien de server niet draait, de *call* mislukt of er geen annotaties zijn, verschijnt er een foutmelding of opmerking. Wanneer de *sidebar* voor het eerst geopend wordt tijdens de sessie, is er nog geen specifieke *slide* aangeklikt of geselecteerd. Nu worden alle annotaties van de gehele presentatie weergegeven. In dit geval is er geen enkele *slide* actief en kunnen geen annotaties worden toegevoegd, dus de *add*-knop wordt niet weergegeven.

4.2.2.2 Sluiten sidebar

Klik in de open *sidebar* rechtsboven op het kruisje om de *sidebar* te sluiten. Indien dit in de vorm gebeurt, wordt de selectie op een *slide* afgebroken.

4.2.2.3 Klikken op een slide/navigeren tussen slides

Indien vanuit het overzicht van *webslides* op een specifieke *slide* wordt geklikt of de pijltjestoetsen worden gebruikt voor navigatie tussen *slides*, dan zullen enkel annotaties van die specifieke dia worden weergegeven. Bovenaan de *sidebar* staat de tekst "all annotations" wanneer men zich op de gehele presentatie bevindt en "Webslide:slidenaam" indien niet. Indien men zich op zo'n dia bevindt, zal de *add*-knop wel weergegeven worden als tweede knop in de *sidebar* bovenaan. Bij het drukken op de pijltjestoetsen zal de *sidebar* de annotaties van de huidige zichtbare *slide* ophalen uit lokale opslag en weergeven.

4.2.2.4 Annotatie toevoegen

De gebruiker klikt op een *webslide* en op de *add*-knop (tweede knop van links, bovenaan in de *sidebar*) waardoor het formulier in de zijbalk wordt weergegeven. In de *combobox* kan een categorie gekozen worden. Daarnaast staat een *toggle* met de keuze voor "private" of "public" (naargelang taalinstelling). Indien "public" werd gekozen, staat de *toggle* naar rechts en is ze blauw gekleurd, anders staat ze links en is ze grijs. Klikken op de *toggle* zal dynamisch de kleur en de tekst wijzigen.

In de volgende tekstvakken kunnen respectievelijk de titel en de inhoud van de annotatie worden opgegeven. Indien een tekstvak focus heeft, zullen *keyevents* (toetsaanslagen) met speciale acties op de *webslides*, niet worden doorgegeven doorheen het DOM. Het laatste tekstvak dient om *tags* in te geven, hierin kunnen met de pijl de aanwezige tags in de presentatie worden weergegeven. De *tag* kan toegevoegd worden door op de knop “voeg toe” te klikken of door de *enter*-toets in te drukken. De *tags* verschijnen dan als knoppen boven het tekstvak, met een specifieke kleur indien “*info*”, “*solved*”, “*important*” en/of “*example*” werden opgegeven. Klikken op een *tag* verwijdert deze terug.

Door met de muis uit de *sidebar* te navigeren en over de slide te *hoveren*, zal de *sidebar* onzichtbaar worden. Ze komt terug door op een element te klikken in de *slide* of eender welke toets aan te slagen zonder selectie. Aanklikbare elementen worden bij *hoveren* geel weergegeven en krijgen een blauwe kleur indien ze zijn aangeklikt. De selectieprocedure kan steeds herhaald worden en het is ook niet verplicht een element uit te kiezen. Let op, de aanklikbare elementen kunnen enkel worden aangeklikt als de muis een tekstselectie-*cursor* is. Indien de muis een handje is, dan zal klikken op een element de bijhorende klikactie van dat element oproepen en zal zo bijvoorbeeld naar een link genavigeerd worden. Door tijdens selectie op de *escape*-toets te drukken, zal terug het overzicht van de *slides* worden getoond en is de *sidebar* terug zichtbaar. De gebruiker zit nu nog altijd in de form, dus klikken op dezelfde *webslide* geeft terug de *slide* met zijbalk en het formulier weer. Klikken op een andere slide zal de *sidebar* terug resetten naar zijn standaardinstelling.

Door op de 'stuur door'-knop te drukken, wordt de annotatie toegevoegd aan de *slide*. Indien er verbinding is met de server, wordt de *sidebar* op de *default*-tabblad van de huidige *webslide* getoond. Als er geen verbinding is, dan wordt er een foutmelding gegeven en blijft de *form* weergegeven. Zo'n foutmelding wordt ook verkregen wanneer de titel en inhoud niet werden ingevuld.

4.2.2.5 Annotatie verwijderen

Bij eigen annotaties van de gebruiker of alle annotaties indien er moderatorfunctionaliteit (bv. professor) is, wordt een vuilbakicoon weergegeven, rechts van iedere annotatie. Als er verbinding is met de server, dan zal klikken op dat icoon de annotatie verwijderen. De inhoud en reacties op een annotatie zijn dan niet meer zichtbaar. Er wordt wel in de annotatie meegedeeld dat de annotatie verwijderd is met een cursief bericht. Als er geen verbinding is met de server, dan wordt een gepaste melding weergegeven.

4.2.2.6 Annotatie aanpassen

Bij eigen annotaties van de gebruiker wordt rechts een potloodicoon weergegeven. Klikken op dit icoon opent terug het formulier en laat de gebruiker toe om de annotatie te bewerken. De tekstvakken in de *form* staan dan al ingevuld met de vorige inhoud van de annotatie. De categorie, *tags* en de *toggle* staan ook correct, overeenkomstig met de annotatie die bewerkt wordt. Deze elementen kunnen net zoals bij een nieuwe annotatie worden aangepast. Op de 'stuur door'-knop drukken, voert de wijzigingen door als er verbinding is met de server, anders wordt een gepaste melding gegeven. Eenmaal een annotatie is bewerkt, staat naast het potloodicoon een klokicoon. Klikken op dit icoon opent een kader met de volledige bewerkingsgeschiedenis van

die inhoud van de annotatie. Deze container wordt gesloten door daarop of op het klokicoon te klikken.

4.2.2.7 Annotatie beoordelen

Een annotatie kan worden beoordeeld door op de duim omhoog of omlaag te klikken. De gebruiker kan zijn eigen annotaties niet beoordelen dus klikken op de duimen heeft dan geen effect. Als er verbinding is met de server, dan zal klikken op een duim deze blauw kleuren en, indien voorheen de andere duim was gekleurd, de kleur daar weghalen. Een tweede keer op dezelfde duim klikken, haalt de kleur en beoordeling terug weg. Als er geen verbinding is met de server, dan worden de duimen niet gekleurd en wordt een gepaste melding gegeven. Op de ster klikken, voegt die annotatie toe aan de favorieten van de gebruiker. Als er verbinding is met de server, dan zal dit de ster geel inkleuren en is het toevoegen gelukt. Een tweede keer op dezelfde ster klikken, haalt de kleur terug weg en verwijdert de annotatie uit de favorieten van de gebruiker. Indien er geen verbinding is met de server, wordt er een gepaste melding gegeven.

4.2.2.8 Annotatie vergroten

Door op het plusicoon links van de auteur of op de auteur en titel zelf te klikken, wordt de annotatie vergroot weergegeven. Nu zijn auteur, categorie, titel, *tags* en inhoud van de annotatie en de reacties weergegeven met hun beoordeling. Het plusicoon wordt nu een minicoon.

4.2.2.9 Annotatie verkleinen

Door op het minicoon links van de auteur te klikken, wordt de annotatie verkleind weergegeven. Hierbij zijn enkel de naam, categorie, titel, aantal reacties en *tags* zichtbaar. Het minicoon wordt nu een plusicoon.

4.2.2.10 Reactie plaatsen

Door op een annotatie te klikken en die met het plusicoon te vergroten, worden de reacties zichtbaar. Onderaan de annotatie staat ook een tekstvak waarin een nieuwe reactie kan worden geschreven. Als er verbinding is met de server en het tekstvak is niet leeg, dan zal op de pijl klikken de reactie toevoegen aan de annotatie. Anders wordt een gepaste melding gegeven. De nieuw toegevoegde reactie staat onderaan de huidige reacties van de annotatie.

4.2.2.11 Reactie verwijderen

Bij eigen reacties van de gebruiker of indien er moderatorfunctionaliteit (bv. professor) is, wordt een vuilbakicoon weergegeven, rechts van iedere reactie. Als er verbinding is met de server, dan zal klikken op het icoon de reactie verwijderen. De inhoud is dan niet meer zichtbaar en in de plaats wordt een cursieve tekst meegegeven die meedeelt dat de reactie is verwijderd. Als er geen verbinding is met de server, dan wordt een gepaste melding weergegeven.

4.2.2.12 Reactie aanpassen

Bij eigen reacties van de gebruiker wordt een potloodicoon weergegeven rechts van iedere reactie. Door op dit icoon te klikken, wordt op de plek van de reactie een tekstvak weergegeven met een nieuwe verzendknop. Dit tekstvak is ingevuld met de inhoud van de reactie. Als er verbinding is met de server en het tekstvak is niet leeg, dan zal klikken op de verzendknop de reactie opslaan. Eenmaal een reactie is bewerkt, staat naast het potloodicoon een klokicoon.

Klikken op dit icoon opent een kader met de volledige bewerkingsgeschiedenis van die reactie. Deze container wordt gesloten door daarop of op het klokicoon te klikken.

4.2.2.13 Reactie beoordelen

Een reactie kan worden beoordeeld door te klikken op de duim omhoog of omlaag. De gebruiker kan zijn eigen reacties niet beoordelen, klikken op de duimen heeft dan geen effect. Als er verbinding is met de server, dan zal klikken op een duim die duim blauw kleuren en, indien voorheen de andere duim was gekleurd, de kleur daar weghalen. Een tweede keer op dezelfde duim klikken, haalt de kleur en beoordeling terug weg. Als er geen verbinding is met de server, dan worden de duimen niet gekleurd en wordt er een gepaste melding gegeven.

4.2.2.14 Opmaak

- De naam van een student staat in een annotatie of reactie altijd blauwgekleurd, bij een professor staat deze steeds in het groen.
- Verkleinde annotaties bestaan uit de gebruikersnaam, berichticoon met aantal reacties, categorie, titel en optionele *tags*. Links naast de gebruikersnaam staat een plusicoon.
- Vergrote annotaties bestaan uit de gebruikersnaam, categorie, titel, de inhoud met beoordeling (duimen, ster) en optionele *tags*. Naast de gebruikersnaam staat een minicoon. Onderaan staan de reacties met een tekstveld om nieuwe reacties toe te voegen en een verzendknop.
- In de *sidebar* zijn bovenaan navigatieknoppen voorzien. De linkse knop is afhankelijk van huidig tabblad een pijl- of *overview*-icoon. De tweede knop is een plusicoon en is enkel zichtbaar indien een dia actief is. De derde knop is een *refresh*-icoon en de vierde en laatste knop is een kruisicoon dat de *sidebar* sluit.
- Op een annotatie of reactie kunnen voor dezelfde gebruiker nooit de twee duimiconen tegelijkertijd blauw ingekleurd staan.
- De *overview*-pagina heeft 4 kolommen. Als het scherm te klein is, zoals bij klikken op een *slide* of openen van de console, zijn er maar 2 kolommen.

4.3 Platform testen

Het project werd op verschillende browsers uitgetest en werkt op recente versies van *Chrome* (66.0.3359.139) en *Firefox* (59.0.3). Om het project volledig te doen werken in *Firefox* wordt verwacht dat *dom.moduleScripts.enabled* op *true* wordt gezet in *about:config*, de instellingen van *Firefox*.

In onderstaande tabel worden de verscheidene testen opgesomd. “US” staat voor *usability test*, “UN” voor *Unit test* en “IN” voor *Integration test*. “AU” wijst op een geautomatiseerde test terwijl “MA” wijst op een manuele test.

Tabel 18: Testplan

Wat	Soort	Frequentie	Wie
Frontend			
Constructor van Caller klasse	UN, AU	Na aanpassing code	Ontwikkelaar

loadAnnotationsFromServer van Caller klasse	IN, AU	Na aanpassing code	Ontwikkelaar
sendAnnotationToServer van Caller klasse	IN, AU	Na aanpassing code	Ontwikkelaar
sendReaction van Caller klasse	IN, AU	Na aanpassing code	Ontwikkelaar
sendVote van Caller klasse	IN, AU	Na aanpassing code	Ontwikkelaar
Annotation_cherrypick in Component klasse	IN, AU	Na aanpassing code	Ontwikkelaar
Annotation_downvote in Component klasse	IN, AU	Na aanpassing code	Ontwikkelaar
Annotation_upvote in Component klasse	IN, AU	Na aanpassing code	Ontwikkelaar
NewAnnotation in Component klasse	UN, AU	Na aanpassing code	Ontwikkelaar
NewReaction in Component klasse	UN, AU	Na aanpassing code	Ontwikkelaar
Constructor van LocalData klasse	UN, AU	Na aanpassing code	Ontwikkelaar
addAnnotation van LocalData klasse	IN, AU	Na aanpassing code	Ontwikkelaar
clearData van LocalData klasse	IN, AU	Na aanpassing code	Ontwikkelaar
findAnnotationByHtmlId van LocalData klasse	IN, AU	Na aanpassing code	Ontwikkelaar
getAnnotation van LocalData klasse	IN, AU	Na aanpassing code	Ontwikkelaar
Zoeken van annotatie	US, IN, MA	Continu	Gebruiker
Foutmeldingen	US, IN, MA	Continu	Gebruiker
HTML-code invoeren	US, IN, MA	Continu	Gebruiker
Openen sidebar	US, IN, MA	Continu	Gebruiker
Sluiten sidebar	US, IN, MA	Continu	Gebruiker
Klikken op een slide/navigeren tussen slides	US, IN, MA	Continu	Gebruiker
Annotatie toevoegen	US, IN, MA	Continu	Gebruiker
Annotatie verwijderen	US, IN, MA	Continu	Gebruiker
Annotatie aanpassen	US, IN, MA	Continu	Gebruiker
Annotatie beoordelen	US, IN, MA	Continu	Gebruiker
Annotatie vergroten	US, IN, MA	Continu	Gebruiker
Annotatie verkleinen	US, IN, MA	Continu	Gebruiker
Reactie plaatsen	US, IN, MA	Continu	Gebruiker
Reactie verwijderen	US, IN, MA	Continu	Gebruiker
Reactie aanpassen	US, IN, MA	Continu	Gebruiker
Reactie beoordelen	US, IN, MA	Continu	Gebruiker
Opmaak	US, IN, MA	Continu	Gebruiker
Ondersteuning van browsers	US, IN, MA	Continu	Ontwikkelaar
Backend			

Annotatie verwijderen met specifieke <i>id</i>	IN, AU	Na aanpassing code	Ontwikkelaar
Alle annotaties verwijderen	IN, AU	Na aanpassing code	Ontwikkelaar
Annotatie toevoegen	IN, AU	Na aanpassing code	Ontwikkelaar
Alle annotaties ophalen	IN, AU	Na aanpassing code	Ontwikkelaar
Annotatie ophalen met specifieke <i>id</i>	IN, AU	Na aanpassing code	Ontwikkelaar

Hoofdstuk 5: Evaluatie en discussies

5.1 Performantie

5.1.1 Componentniveau

5.1.1.1 Databank

De gebruikte databank is een van de bekendste NoSQL-voorbeelden: MongoDB. Het grootste verschil met een traditioneel RDMS zoals MySQL is dat er geen vast schema voorzien is voor objecten die opgeslagen moeten worden. Dit betekent dat dynamisch andere objecten kunnen opgeslagen worden, zonder deze op voorhand te definiëren. Intern worden objecten in MongoDB opgeslagen als BSON (Binary JSON), een JSON-gebaseerd formaat.

Aangezien de databank in dit project gebruikt wordt om annotaties op te slaan en die annotaties in JSON-formaat doorgestuurd worden, was de keuze voor de hand liggend. Enerzijds biedt MongoDB *Java Drivers* de mogelijkheid om snel Java-objecten en JSON-objecten om te zetten naar BSON. Anderzijds is het flexibele schema van MongoDB een groot voordeel tijdens het ontwikkelingsproces, waar nieuwe features sneller kunnen toegevoegd en getest worden.

Alle data wordt bij elke *refresh* in de *localData*-variabele opgeslagen waardoor updates niet *real-time* zullen gebeuren. Dit zorgt voor betere performantie.

De zoekfunctie splitst de zoek-*input* op in verschillende elementen en steekt ze in een lijst. Elke zoekterm zal apart een hele weg afleggen waarbij het samen met de helpfuncties matchen filtert. Het eerste zokelement kiest zijn matches uit alle beschikbare annotaties uit de presentatie, maar elk volgend zokelement zal een meer beperkte *input* krijgen, afhankelijk van de *output* van de vorige zoekopdracht. Zo wordt er uiteindelijk een groep van annotaties verkregen die aan elke zoekterm voldoet. Bij elke volgende letter die ingetypt wordt, wordt er opnieuw gezocht, zoals bij *Google Search*, hierdoor wordt deze reeks helpfuncties steeds opnieuw opgeroepen. Dit maakt de zoekfunctie minder performant.

Aangezien de klant had gevraagd voor een beperkte backend was het niet mogelijk *real-time* meldingen zoals in *Facebook* te repliceren. Daarom werd gekozen voor wekelijkse meldingen zonder de nood om extra programma's te downloaden. Op deze manier krijgt de gebruiker toch de laatste toegevoegde annotaties en reacties te zien door vergelijking van data.

5.1.2 Systeemniveau

De webapplicatie is enkel afhankelijk van de internetverbinding en de databank. Indien de internetconnectie plots wegvalt, kan er nog steeds verder gewerkt worden met de oude opgehaalde annotaties. De enige operaties die dan wegvallen, zijn *refreshen*, *upvoten*, *downvoten*, toevoegen aan favorieten, annotaties of reacties *posten*, aanpassen of verwijderen. De gebruiker krijgt dan een foutmelding en kan verder surfen na het sluiten.

5.2 Beveiliging

Op vlak van beveiliging werd er in het project slechts een klein gedeelte voorzien. In de tekstvelden wordt er namelijk een *noHtml*-functie opgeroepen zodat gebruikers zelf geen HTML-codefragmenten te zien kunnen krijgen, die met slechte bedoelingen door andere gebruikers werden meegegeven aan annotaties of reacties. De authenticatie zal door de klant zelf geregeld worden waardoor er enkel een *prompt* voorzien werd. Deze vraagt aan de gebruiker zijn loginnaam, zonder wachtwoord. Wanneer de gebruiker “professor” ingeeft, zal hij de rol van de professor verkrijgen met moderatorfunctionaliteit. Als backend wordt verwacht dat de huidige implementatie door de klant wordt vervangen door hun eigen systeem. De huidige backend geldt vooral als illustratie van de werking van de *sidebar*. Op de backend werden om deze reden geen extra veiligheidschecks of controles op consistentie uitgevoerd.

5.3 Schaalbaarheid

De MongoDB-databank is uitermate geschikt om grote datasets te verwerken, dus het opslaan van annotaties kan gemakkelijk grote aantallen aan. Bij het project wordt echter verwacht dat er niet meer dan een vijftal annotaties per *slide* zullen komen met zelfs mogelijkheid tot lege *slides*. Door de veronderstelling van de weinige annotaties werd gekozen om alle verkregen annotaties van de databank onmiddellijk in een lokale *JavaScript*-klasse op te slaan en vervolgens alle annotaties uit te tekenen. Er zouden al heel veel annotaties moeten voorkomen om ervoor te zorgen dat de lokale klasse een onaanvaardbare invloed op het geheugengebruik zou hebben. De implementatie van het weergeven van annotaties is echter meer beperkt.

Bij meer dan 100 annotaties op een presentatie is er al een merkbaar, weliswaar klein, snelheidsverlies van de *sidebar*. Dit komt doordat iedere annotatie uit een twintigtal HTML-tags bestaat en deze allemaal simultaan moeten worden getekend. Hiervoor werd een laadbalk voorzien die duidelijk maakt dat de *sidebar* nog bezig is met het tekenen van de annotaties.

In de toekomst wordt aangeraden om, indien meer dan 100 annotaties per presentatie worden verwacht, een dynamisch systeem van tekenen te introduceren. Hierbij zouden dan bijvoorbeeld enkel de zichtbare annotaties of een vast aantal getoond worden. Aangezien bij de huidige implementatie heel weinig annotaties worden verwacht, is de implementatiekost hoger dan de huidige opbrengst die deze implementatie zou opleveren.

5.4 Problemen en geleerde lessen

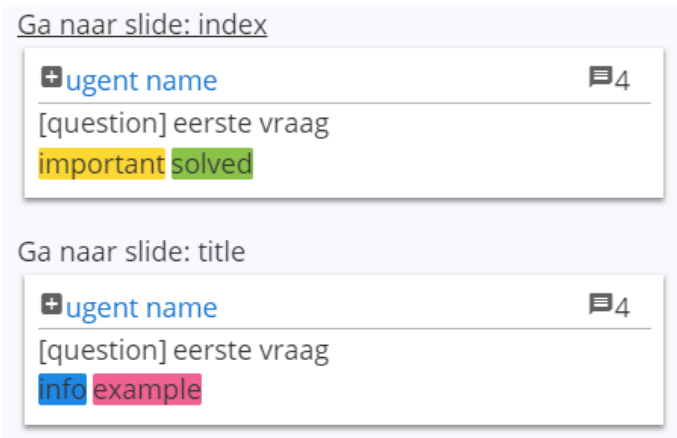
Onze *sidebar* werd geïmplementeerd in de bestaande *webslides* van de klant die ons limiteerden en uiteindelijk onze grootste uitdaging vormden. Ook was het niet mogelijk aan alle voorstellen van de klant te voldoen door de limieten van bestaande HTML-elementen bij het maken van de

drop-down list voor de zoekbalk. Ten slotte zorgde de simpele backend ook voor minder mogelijkheden.

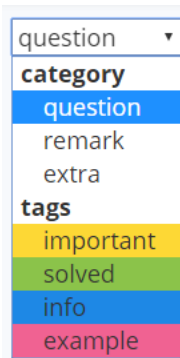
Wanneer de gebruiker zich op een specifieke *webslide* bevindt, wordt deze vergroot via schaling met zwarte randen langs weerskanten. Hierdoor wordt de *sidebar* zelf ook herschaald en beperkt tot de nieuwe voorziene ruimte. De plaatsing komt helemaal in het midden en bedekt hierdoor een deel van de dia's waardoor ze onleesbaar worden. Oorspronkelijk was het doel de vergrote *webslide* naar links te verschuiven en de gebundelde zwarte velden dan aan de rechterkant te vervangen door de *sidebar*, maar dit bleek onmogelijk zonder wijziging van de broncode. De *sidebar* kan natuurlijk steeds gesloten worden, dus dit vormde enkel een probleem bij het invullen van het formulier op het *add*-tabblad waarbij de *sidebar* open moet blijven staan. Hierdoor zijn een deel van de elementen op de *webslide* bedekt en wordt het onmogelijk een bedekt element te selecteren die moet worden gekoppeld aan een annotatie. Om de gebruiker toch alle elementen te tonen, wordt de *sidebar* tijdelijk gesloten tijdens het *hoveren* over de verschillende elementen. Na het drukken op eender welke toets of klikken op een element, wordt de *sidebar* terug geopend. Bij het *hoveren*, wordt het geselecteerde DOM-element geel gekleurd en na selectie wordt deze lichtblauw. Er is steeds herselectie mogelijk, zonder verplichting tot het kiezen van een element.

Om te checken of de *sidebar* zich op een enkele dia bevindt, wordt er gewerkt met schaling aangezien de broncode hier geen methode voor bevat. Achteraf gezien, was dit geen goede keuze doordat computers met kleinere schermgrootte een kleinere schaling bevatten en de *sidebar* hierdoor niet weet dat hij zich op een enkele *slide* bevindt. Dit heeft als effect dat de zijbalk niet sluit bij kiezen van een element op zulke schermen. De schaling werd ook gebruikt om de *overview* in minder kolommen weer te geven indien het venster wordt verkleind of de console wordt geopend.

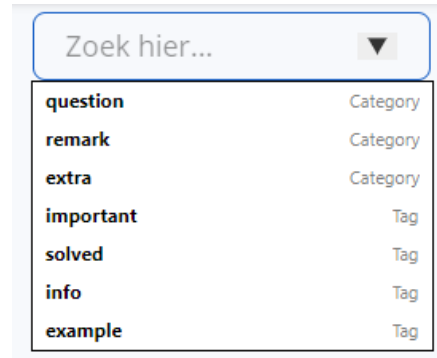
De vier standaardtags krijgen hun eigen kleur: “*info*” (blauw), “*example*” (roze), “*important*” (geel) en “*solved*” (groen) (zie Figuur 18). Er werd door de klant gevraagd om deze tags in de zoekbalk voor te stellen en daarbij het kleur weer te geven zoals op Figuur 20. Die kleuren zijn enkel mogelijk wanneer er geen *datalist* wordt gebruikt, dus wanneer geen *input* nodig is. Toch zou de gebruiker moeten kunnen filteren op zelfgekozen waarden waardoor er met zo'n zoeklijst niet aan de vraag voldaan kon worden. Het is slechts mogelijk de lay-out van de *drop-down list* weer te geven zoals op Figuur 19.



Figuur 18: Standaardtags met kleur



Figuur 20: Zoekbalk zonder input



Figuur 19: Zoekbalk met input

Doordat er maar een simpele backend werd gerealiseerd, zijn er inconsistenties in de code. Indien twee gebruikers gelijktijdig aan dezelfde annotatie of reactie bezig zijn, wordt er geen tijdelijke *lock* voorzien. Zo kan het gebeuren dat de professor een annotatie verwijdert tijdens een aanpassing door een gebruiker op het formulier. Hierdoor wordt de status veranderd naar “*removed*” en nadien nog gewijzigd naar “*changed*” waardoor de annotatie blijft bestaan. Dit zou in een uiteindelijke implementatie gecheckt moeten worden in de backend voor de wijziging mag doorgaan. Omdat bij de klant de nadruk op lay-out werd gelegd en niet zozeer op de implementatie van de backend, werd gekozen om aan deze kleine inconsistenties geen verdere aandacht meer te besteden.

Hoofdstuk 6: Installatiehandleiding en gebruikershandleiding

6.1 Installatiehandleiding

Hier vindt de klant een stappenplan voor het installeren van de nodige programma's zodat er in de toekomst gemakkelijk aanpassingen gedaan kunnen worden aan de code of dit geïntegreerd kan worden in een reeds bestaand project. Er wordt meegegeven hoe de backend gerealiseerd werd (Docker en MongoDB) en hoe de testen (Jest) uitgevoerd kunnen worden.

De webapplicatie dient om annotaties aan te maken op de *WebSlides* en deze op een later ogenblik weer opnieuw te kunnen opvragen. De gegevens dienen dan ook persistent opgeslagen te worden. Hiervoor werd gekozen voor de NoSQL-databank MongoDB.

Anderzijds worden de annotaties over het Internet naar de databank gestuurd via een GlassFish Server. Deze reageert op inkomende oproepen om bijvoorbeeld annotaties op te halen of op te slaan.

Men kan via Docker deze software in een container starten, hierdoor dient men deze GlassFish en MongoDB niet lokaal zelf te installeren. Deze containers doen dienst als een soort lichtgewicht virtuele machine, waardoor de omgeving altijd dezelfde is. Verder wordt er verondersteld dat Docker geïnstalleerd is; dit kan op

Windows (https://docs.docker.com/toolbox/toolbox_install_windows/),

Linux (<https://docs.docker.com/install/linux/docker-ce/ubuntu/>) of

MacOS (<https://docs.docker.com/docker-for-mac/install/>).

De annotatietool heeft een databank nodig om annotaties op te slaan en een webserver om de presentatie met annotatietool te sturen naar de gebruiker. Vooraleer presentaties dus online gezet kunnen worden, moeten deze vereisten in orde zijn. Deze twee services draaien elk in een Docker-container en dit moet ook opgezet worden. Concreet bestaat de pre-setup uit:

- Configureren van de Docker-containers
- Configureren van de Servlet

Gebruikt men voor elke presentatie dezelfde webserver en databank, dan hoeft de pre-setup maar één keer te gebeuren. De setup zelf bestaat uit:

- Het toevoegen van de annotatietool aan een presentatie
- De presentatie toevoegen aan de webserver
- Deployen van de webserver via een distributiebbestand

De setup voert men elke keer uit als een presentatie toegevoegd of verwijderd moet worden.

Verder bevinden zich in de root van de repository ook de map *docker/* en de map *Servlets/*. Deze kunnen als template gebruikt worden, om over te nemen of gemakkelijk eigen configuratie uit te voeren.

6.1.1 Pre-setup

6.1.1.1 Configureren van de Docker-containers

In deze stap kan men de naam van de containers aanpassen, in welke map verdere configuratie gezocht moet worden en welke uitgaande poort van de containers afgebeeld wordt op een poort van de host. Verder vindt men hier ook de optie om te kiezen welke poort opengezet wordt van de container, zodat die communicatie toelaat.

Vooraleer de containers opgestart worden, moet Docker weten hoeveel containers nodig zijn, en waar verdere configuratie te vinden is. Het `.yaml`-bestand beschrijft deze containers. Men kan de `docker/`-map in de repository gebruiken als template om de containers te configureren.

Bijvoorbeeld ziet het bestand `webslides-01.yaml` er zo uit:

```
version: "2" # "3" if beta release (swarm mode)
services:
  webslides-01-glassfish:
    build:
      context: webslides-01-glassfish/
      dockerfile: Dockerfile
    ports:
      - "9003:8080"
  webslides-01-mongodb:
    build:
      context: webslides-01-mongodb/
      dockerfile: Dockerfile
```

Figuur 21: Voorbeeld van het `.yaml`-bestand

De lijn na `services` bepaalt de naam van de container, en de `context` bepaalt in welke map Docker moet zoeken om de `dockerfile` met bijkomende beschrijving van de container te vinden.

In Figuur 21 worden dus twee services gedefinieerd: `webslides-01-glassfish` is dan de webservercontainer en `webslides-01-mongodb` is de databankcontainer. De bijhorende `dockerfiles` zijn terug te vinden in respectievelijk `webslides-01-glassfish/` en `webslides-01-mongodb/`.

Belangrijk is hier de lijn na `ports`. Dit duidt aan welke poort van de container afgebeeld wordt op de poort van de host. In het voorbeeld is dit dus poort “8080” van de container op de poort “9003” van de host. Hier kan men dus de container bereiken vanuit de host via poort “9003”.

Verder staat in de individuele `Dockerfile`-bestanden (in de template zijn dit `docker/webslides-01-glassfish/Dockerfile` en `docker/webslides-01-mongodb/Dockerfile`) nog welke poort communicatie van buiten zijn container toelaat. Dit gebeurt via het `EXPOSE` sleutelwoord. In de

template is dit “8080” voor de webserver, en “27017” voor de databank (de default poort van MongoDB).

Een voorbeeld van zo’n *dockerfile* vindt men terug in Figuur 22.

```
# Latest glassfish image
FROM glassfish:latest

# Copy WAR file and startup script, make startup script executable
COPY AnnotationToolServer.war /
COPY start.sh /
RUN chmod +x /start.sh

# Expose HTTP port
EXPOSE 8080

# Execute startup script
ENTRYPOINT ["/start.sh"]
```

Figuur 22: Voorbeeld van een *dockerfile*

6.1.1.2 Configureren van de Servlet

De broncode van de Servlet vindt men terug in de map *Servlets/*. De enige configuratie van de Servlet t.o.v. de databank bevindt zich in de het bestand

Servlets/AnnotationToolServer/src/java/Annotations/resources/prod.properties.

Het bestand bevat de connectiestring van de databank, de databanknaam en de collectienaam, bijvoorbeeld:

```
connString=mongodb://webslides-01-mongodb:27017
dbNaam=BachelorproefDB
collNaam=Annotaties
```

Figuur 23: Voorbeeld van een *.properties*-bestand

Om de Servlet correct te laten verbinden met de databank, dient men de connectiestring correct in te vullen. De connectiestring heeft een vaste vorm: *mongodb://host:poort*. Als host kiest u de naam van de Dockercontainer van de webserver die u in de vorige stap gekozen heeft, als poort de *port* die u in de vorige stap heeft opengezet van de databankcontainer (standaard is dit 27017). De databanknaam en bijhorende collectienaam hoeven niet op voorhand te bestaan in de instantie van MongoDB.

6.1.2 Setup

6.1.2.1 De annotatietool toevoegen aan een presentatie

De annotatietool is zo ontworpen om te kunnen inhaken op een bestaande WebSlidespresentatie. Zo'n WebSlidespresentatie is compact in één *.html*-bestand voor te stellen (vanaf hier noemen we dit HTML-bestand *index.html*). Meer info over het maken van een eigen WebSlides-presentatie vindt men op <https://openwebslides.github.io/BroodjeMartinnovatie/#GitHubDemo>.

Om de annotatietool nu toe te voegen aan een eigen WebSlidespresentatie, downloadt men het bestand *import.zip* uit de map */Import* en pakt men het uit op dezelfde plaats waar het *.html* bestand staat van uw presentatie (deze map bevat nu dus *index.html*, de map *Sidebar* en het scriptbestand *import.sh*). Voordat de annotatietool toegevoegd wordt aan de presentatie, dient men in *Sidebar/scripts/data/config.js* met een simpele tekstverwerker een aanpassing te doen. In dit bestand staat de URL waarop de server te vinden is, en die dus gebruikt wordt om de annotaties op te vragen en door te sturen. Het bestand ziet er als volgt uit:

```
export let serverSettings = {  
  "serverAddress": "webslides-01.project.tiwi.be"  
  , "restPath" : "/webresources/api"  
};
```

Figuur 24: Voorbeeld inhoud *config.js*

Op de plaats van “*webslides-01.project.tiwi.be*” staat dus de URL van de eigen server. Op de plaats van “*/webresources/api*” komt het eigen gekozen REST-pad dat aan de URL toegevoegd wordt. Als de server niet verandert voor volgende presentaties, kan men altijd een kopie van *config.js* of van de volledige *Sidebar*-map bijhouden, zodat deze aanpassing niet met elke volgende presentatie moet doorgevoerd worden.

Vervolgens voert men het script *import.sh* uit. Dit kan door gebruik te maken van eender welk bash-gebaseerde terminal, bijvoorbeeld de Docker Quickstart Terminal. Om dit te doen, navigeert men in deze terminal eerst naar de map waar u zojuist de inhoud van *import.zip* naartoe kopiëerde, met het commando *cd*. Daarna voert men het script uit. Bijvoorbeeld:

```
$ cd /usr/webslides/presentation  
$ ./import.sh mijn_index.html
```

Figuur 25: Voorbeeld commando's om installatiescript uit te voeren

Als argument neemt het script de naam van het *.html*-bestand van de presentatie en vervangt u *mijn_index.html* dus door de naam van uw eigen bestand. Wordt geen argument meegegeven, dan kiest het script standaard voor *index.html*.

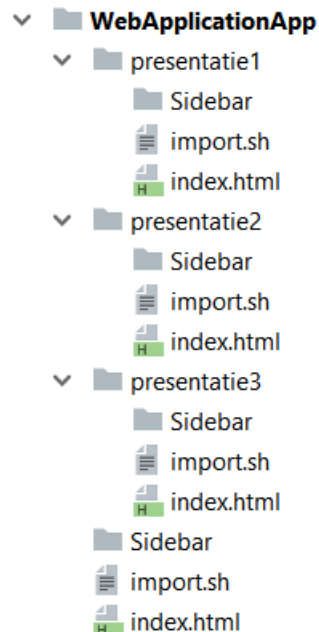
Hierdoor wordt al het nodige toegevoegd aan de presentatie zelf. Er wordt een back-up genomen van het originele bestand, met dezelfde naam met een “~” erna. Dus in het vorige voorbeeld zou de originele presentatie terug te vinden zijn in het bestand *mijn_index.html~*.

6.1.2.2 De presentatie toevoegen aan de webserver

De presentaties aan de webserver toevoegen, is simpel, maar eerst moet een beperking in acht genomen worden: elke presentatie moet in een eigen map zitten, met unieke mapnaam. Dit komt omdat men het navigeren vanuit de ene presentatie naar de andere presentatie kan doen op basis van de mappenstructuur.

Is aan de vereiste voldaan, dan kopieert men elke presentatiemap naar de map */Servlets/AnnotationToolServer/web/WebApplicationApp/*. Als dat nog niet het geval is, wordt best een *index.html* voorzien in deze map. Die kan bijvoorbeeld dienen als overzichtspagina tussen de verschillende presentaties. Natuurlijk kan een presentatie ook gebruikt worden hiervoor. De structuur in */Servlets/AnnotationToolServer/web/WebApplicationApp/* is nu dus bijvoorbeeld die van Figuur 26.

Is dit correct gebeurd, dan maakt men een distributiebestand (met extensie *.war*) van de map *Servlets/AnnotationToolServer/*. Dit ene bestand stelt de volledige logica van de webserver voor en wordt door de Dockercontainer gebruikt. Het distributiebestand aanmaken kan bijvoorbeeld door de map *Servlets/AnnotationToolServer/* te openen in NetBeans, en de optie "Clean and Build" te laten uitvoeren. Het *.war* bestand bevindt zich dan in *Servlets/AnnotationToolServer/dist/*.



Figuur 26: Structuur in *WebApplicationApp*

6.1.2.3 Deployen van de webserver aan de hand van een distributiebestand

Het distributiebestand dat men in de vorige stap verkregen heeft, kan men nu dus gebruiken om de webserver te *deployen*. Volgt men de *docker-template*, dan plaatst men deze in *docker/webslides-01-glassfish/*, met de naam *AnnotationToolServer.war*.

De containers starten

Als een correct distributiebestand toegevoegd is, is alles klaar om de twee containers te starten. Deze Dockercontainers zijn de omgeving waarin de webserver en de databank zullen werken. Gebruikt men de *docker-template*, dan plaatst men het script *buildandrun.sh* (te vinden in de root van de repository) buiten de map *docker/* en voert men het uit. Dit kan via de Docker Quickstart Terminal.

```
$ ./buildandrun.sh
```

Figuur 27: Uitvoeren script om containers te starten

Volgt men de template niet, dan verandert men best in het script het argument *docker/webslides-01.yml* in een verwijzing naar het voorgenoemde .yml bestand. Negeer in dat geval de uitvoer van het script.

Als test dat de webserver én de databank correct gestart zijn, kan men al eens naar <http://192.168.99.100:XXXX> surfen (waarbij XXXX het gekozen poortnummer is). Standaard staat deze momenteel op <http://192.168.99.100:9003> ingesteld. Indien men hier begroet wordt door de gekozen presentatie met annotatietool, dan is de frontend van de applicatie in orde. Indien u annotaties kunt toevoegen en verwijderen, dan is de backend dat ook! Meer info over het gebruik van de annotatietool vindt men in **6.2 Gebruikershandleiding**.

6.1.3 Jest installeren

Alvorens de testen via Jest uitgevoerd kunnen worden, moeten enkele installaties gedaan worden. Ten eerste moet *Node.js* reeds aanwezig zijn op de computer waarin de *Node Package Manager (npm)* geïncludeerd is. Om dit te checken, worden volgende commando's uitgevoerd na het openen van de opdrachtprompt.

```
<<node -v  
<<npm -v
```

Indien deze een foutmelding geven, worden ze beiden geïnstalleerd via www.nodejs.org/en. Indien deze een specifieke versie weergeven, is alles al geregeld en kan Jest gedownload worden. Navigeer hiervoor naar de *Sidebar*-map. Deze is te vinden in de map “WebFundamentals-master”. De Node Modules zullen hier worden geïnstalleerd. Dit kan vervolgens via het commando:

```
<<npm install --save-dev jest
```

Op sommige toestellen werpt Jest een fout op in verband met Babel. Uit voorzorg is het best om ook volgend commando uit te voeren:

```
<<npm install babel-preset-env --save-dev
```

Sommige testen maken gebruik van jQuery. Hiervoor verwacht npm dat jQuery beschikbaar is. Dit kan via volgende regel:

```
<<npm install jQuery
```

Nu alles geïnstalleerd is, kunnen de bestaande *tests* uitgevoerd worden. Navigeer hiervoor naar de map “Sidebar/scripts/tests” en voer volgend commando uit:

```
<<npm test
```

In paragraaf 4 wordt het testplan weergegeven met *unit*, *integration* en *usability tests*, hun scenario en frequentie. Hierin wordt het formaat van een test besproken met een voorbeeld zodat nieuwe testen gemakkelijk opgesteld kunnen worden.

6.2 Gebruikershandleiding

Functionaliteit:

- 0: Inloggen
- 1: Openen van de annotatietool
- 2: Navigatieknoppen
- 3: Annotaties zoeken en bekijken
- 4: Annotaties plaatsen
- 5: Annotaties aanpassen en verwijderen
- 6: Reactie op een annotatie plaatsen en aanpassen
- 7: Notificaties
- 8: Annotaties en reacties beoordelen
- 9: Tooltips

6.2.0 Inloggen

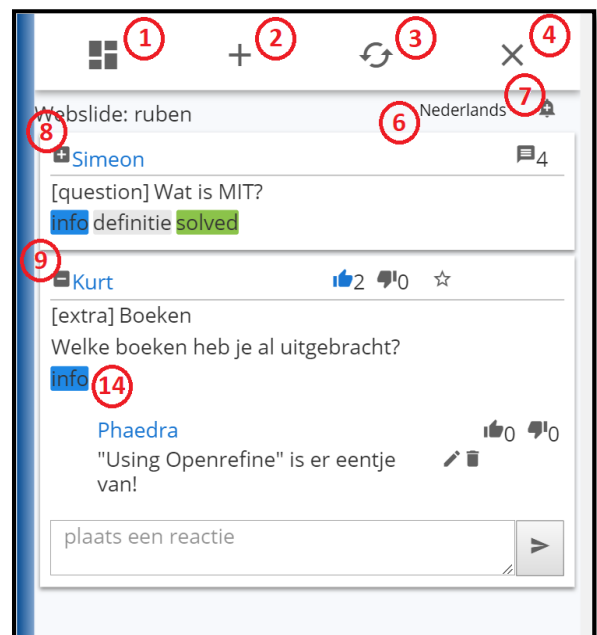
Wanneer u surft naar de website, krijgt u een *prompt* te zien die vraagt achter uw loginnaam. Afhankelijk van de *input* zullen er andere *features* mogelijk zijn in de *sidebar*. Zo zullen professoren alle annotaties en reacties kunnen verwijderen en verschijnt hun gebruikersnaam in een opvallende kleur. U kan als professor inloggen door "professor" in te geven in de *prompt*, anders wordt u als student beschouwd.

6.2.1 Openen van de annotatietool

Om de annotatietool te openen, klikt u op het blauwe icoon aan de rechterkant van het scherm. Door de blauwgekleurde balk aan de linkerkant van de annotatietool te verslepen, kan men de grootte die hij inneemt op het scherm aanpassen.

6.2.2 Navigatieknoppen

Via het *overview*-icoon wordt het *overview*-tabblad geopend waarin men kan filteren op annotaties van de gehele cursus (Zie 3: Annotaties zoeken en bekijken). (1) Via het plusicoon wordt het *add*-tabblad getoond en kan men zelf een annotatie schrijven en toevoegen (Zie 4: Annotaties plaatsen). (2) Het *refreshen* van de annotaties doet men door op het *refresh*-icoon te klikken. (3) Met behulp van het kruisicoon sluit men de annotatietool. (4) Om terug te gaan naar de *default*-tabblad klikt u op de terugknop. (5) Om een andere taal te selecteren, klikt u op de huidige taal en selecteert u in het *drop-down* menu de gewenste vertaling. (6) Het allerlaatste icoon wordt besproken onder de titel 7: Notificaties. (7)



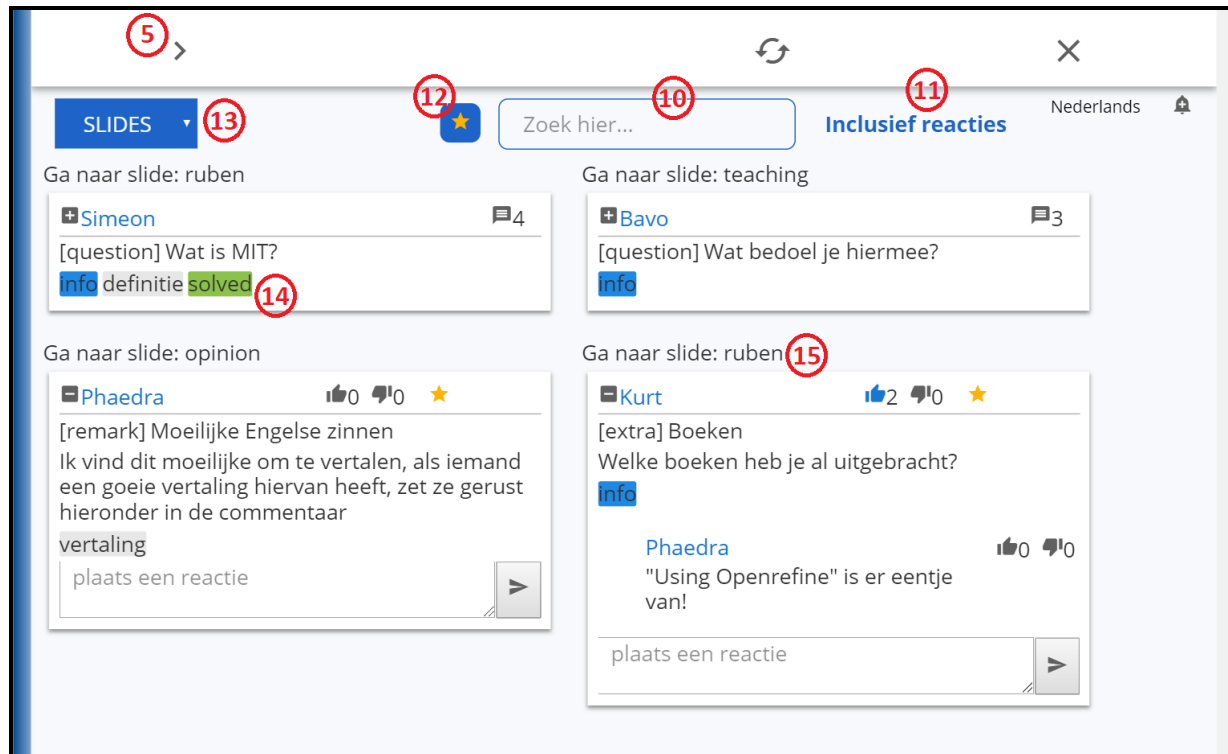
Figuur 28: Navigatieknoppen in het Default-tabblad

6.2.3 Annotaties zoeken en bekijken

Wanneer de annotatietool is geopend, worden *by default* alle annotaties die bij de huidige *webslide* horen onder elkaar geminimaliseerd weergegeven. In deze geminimaliseerde versie zijn enkel de auteur, titel, categorie, tags en het aantal reacties te zien van de annotatie. Om de overige informatie zoals de inhoud van zo'n annotatie en de effectieve reacties erop te bekijken, klikt u op het plusicoon naast de auteur. (8) Om de annotatie opnieuw te minimaliseren, klikt u op het minicoon. (9) In het *overview*-tabblad (Zie 2: Navigatieknoppen) kan men de annotaties van de gehele cursus bekijken en filteren. Hierin kan men eveneens op dezelfde manier een annotatie vergroten en verkleinen.

In het *overview*-tabblad kan er zowel op inhoud als op een categorie, titel, naam of *tag* gefilterd worden door in het zoekveld een term op te geven en op de toets *Enter* te drukken. Ook meerdere zoektermen kunnen meegegeven worden, gescheiden door een komma. De output matcht dan met elk opgegeven zoekwoord, zonder rekening te houden met hoofdletters. (10) *By default* wordt ook een annotatie gematcht wanneer een gedeelte van een reactie (inhoud of *poster*) overeenkomt met het opgegeven stukje tekst. Dit kan aan- of uitgezet worden door op de blauwe tekst te klikken naast de zoekbalk. (11) Men kan eveneens favoriete annotaties opvragen, hiervoor klikt u op de *favorite*-knop naast de zoekbalk (Zie 8: Annotaties en reacties beoordelen). (12) Nadat men op de toets *Enter* of de *favorite*-knop heeft gedrukt, worden alle gematchte annotaties weergegeven onder de zoekbalk. De volgorde waarin al deze annotaties worden weergegeven, kan ingesteld worden door een andere sorteervolgorde aan te duiden in het *drop-down* menu. (13) Standaard staat deze op *SLIDES* (ordenen volgens de naam van de *webslide*). Andere opties zijn *TOP* (sorteren op beoordeling door middel van aantal *upvotes* verminderd met aantal *downvotes*), *HOT* (sorteren op populariteit met aantal reacties) en *NEW* (sorteren op datum met vooraan de recentste annotaties).

Naast het filteren op een *tag* kan men ook binnen een annotatie op een *tag* klikken om automatisch naar het *overview*-tabblad te navigeren met soortgelijke annotaties. (14) Klik op 'Jump to slide' (of een vertaling hiervan) boven een annotatie in de lijst om naar de bijhorende *slide* te gaan. (15)



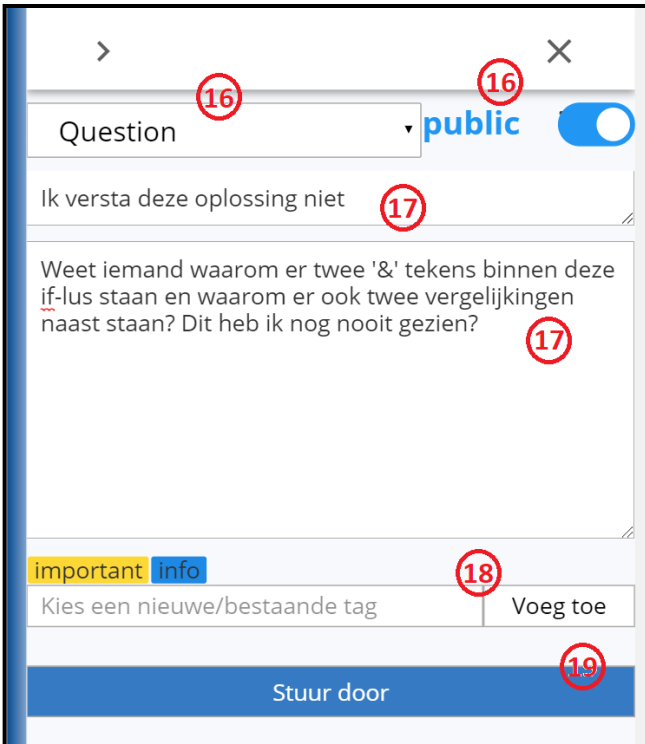
Figuur 29: Overview-tabblad

6.2.4 Annotaties plaatsen

Om een annotatie te plaatsen bij een bepaalde *slide*, gaat u naar het *add*-tabblad (Zie 2: Navigatieknoppen). Er moet weliswaar eerst een *slide* geselecteerd zijn om een annotatie toe te voegen. Men kan een categorie kiezen via het *drop-down* menu en aanduiden of de annotatie publiek (voor iedereen zichtbaar) of privaat (enkel voor uzelf zichtbaar) moet worden geplaatst. (16) In de invoegkaders eronder geeft u een titel aan de annotatie en schrijft u de inhoud ervan. (17) Er kunnen eventueel *tags* worden toegevoegd door een tag op te geven en hierna op de *add*-knop te klikken. Deze opgegeven *tag* verschijnt boven het invoegbalkje. Om een *tag* te verwijderen, hoeft u er gewoon op te klikken. (18)

Normaliter wordt een annotatie gekoppeld aan de gehele *slide*. Om een annotatie aan een specifiek deel van de *slide* te koppelen, klikt u op het gewenste element in de *webslide*. Hierbij zal de annotatietool tijdelijk verborgen worden om over de gehele *slide* aan te kunnen duiden. Wanneer u een element heeft aangeduid, verschijnt het in het lichtblauw en komt de zijbalk opnieuw tevoorschijn. Om de sidebar op te roepen zonder een verplichte selectie, kunnen alle toetsen ingedrukt worden, behalve de *escape*-toets aangezien deze een speciale functie heeft binnen de *websites* zelf).

Om de ingevulde annotatie toe te voegen, klikt u op de *submit*-knop. (19) Klik op de terugknop links bovenaan om het maken van annotatie te annuleren en terug te keren naar het *default*-tabblad (Zie 2: Navigatieknoppen).



Figuur 30: Add-tabblad

6.2.5 Annotaties aanpassen en verwijderen

Wanneer men in de het *overview*-tabblad of *default*-tabblad (Zie 2: Navigatieknoppen) een annotatie maximaliseert, zijn er drie klikbare iconen te zien aan de rechterkant: Een klokicoon, potloodicoon en vuilnisemmericoon. (20) Het klokicoon staat enkel bij annotaties die in het verleden aangepast zijn door hun *original poster*. Met behulp van dit icoon kan men de geschiedenis van zo'n bewerkte annotatie zien. Met het potloodicoon kan men een annotatie bewerken, het *add*-tabblad wordt dan geopend met daarin de velden reeds ingevuld met huidige info (Zie 4: Annotaties plaatsen). **Belangrijk:** Om de wijziging ook effectief door te voeren, klikt u op de *submit*-knop. (19) Om een annotatie te verwijderen, klikt u op het vuilnisemmericoon.

6.2.6 Reactie op een annotatie plaatsen en aanpassen



Figuur 31: Annotatie met reacties

Wanneer een annotatie gemaximaliseerd is, (Zie 3: Annotaties zoeken en bekijken) kan men er een reactie op plaatsen door deze reactie in te vullen in het invoegkader en daarna op het pijl-icoon te klikken. (21) De reactie verschijnt nu onderaan de lijst van reacties op deze annotatie. Net zoals bij een annotatie (Zie 5: Annotaties aanpassen en verwijderen) kan men een eigen reactie bewerken en verwijderen met behulp van het potloodicoon en vuilnisemmericoon. Met het klokicoon kan men eveneens de geschiedenis van een reactie bekijken als deze aangepast is. (22)

6.2.7 Notificaties

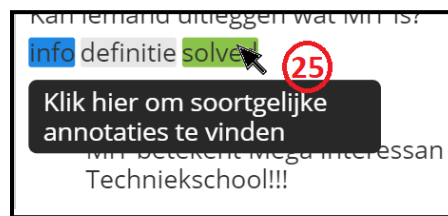
Wanneer iemand een nieuwe annotatie of reactie heeft geplaatst binnen de laatste week, dan zal dit verschijnen in de notificatielijst. Deze lijst kan men bekijken door op het belicoon te klikken onder de navigatiebalk (Zie 2: Navigatieknoppen). (7) Hierin kan men opnieuw (zoals in 3: Annotaties zoeken en bekijken) door op 'Jump to slide' (of een vertaling hiervan) te klikken in de lijst naar de bijhorende *slide* en annotatie navigeren.

6.2.8 Annotaties en reacties beoordelen

Wanneer een annotatie gemaximaliseerd is, (Zie 3: Annotaties zoeken en bekijken) kan u uw mening geven over een annotatie of reactie erop. Om een annotatie of reactie te *up-* of *downvoten*, klikt u op de overeenkomstige duim omhoog of omlaag. (23) Om een annotatie toe te voegen aan uw favorieten, klikt u op de ster (24) Uw favoriete annotaties kan u bekijken in het *overview*-tabblad (Zie 3: Annotaties zoeken en bekijken).

6.2.9 Tooltips

Als er toch iets van deze gebruikershandleiding u ontglipt is, zijn er nog steeds enkele *tooltips* die helpen bij het ontcijferen van de verscheidene iconen en knoppen. Blijf met de muis even op een knop of icoon staan om een *tooltip* tevoorschijn te laten komen. (25)



Figuur 32: Tooltip

Hoofdstuk 7: Besluit

Tot op vandaag hebben studenten en professoren nog steeds geen manier gevonden om cursussen en bijhorende informatie te bundelen op een dynamische cursus met mogelijkheid tot integratie van externe platformen. *Open Weblides* bieden de mogelijkheid de studiepakketten als PowerPointpresentatie *online* te zetten en uit te bouwen tot een mini-forum. Hierdoor verloopt communicatie tussen leerling en professor vlotter en kunnen medestudenten meegenieten van vragen, antwoorden en opmerkingen of extra bronnen, gedeeld door anderen en goedgekeurd door de professor.

In dit project werd een annotatietool als *feature* geïmplementeerd in reeds bestaande *weblides*. Dit biedt de mogelijkheid tot plaatsen van annotaties en reacties, met een beoordelingssysteem voor betrouwbaarheid. Elke annotatie behoort tot een bepaalde categorie en krijgt tags om het filteren te vergemakkelijken. Via zoekopdrachten kunnen studenten gemakkelijker op zoek gaan naar nodige info en door middel van *cherrypicking* kunnen bepaalde annotaties zelfs als favoriet opgeslagen worden. De opmerkingen kunnen gesorteerd worden en via wekelijkse meldingen blijft de gebruiker geüpdatet. Het navigeren tussen de verschillende *weblides* loopt vlot en maakt het geheel meer gebruiksvriendelijk.

Met deze extra *feature* is het *Open Weblides*-project één stap dichterbij voltooiing.

Referentielijst

Material Design. Geraadpleegd op 8 maart 2018 via <https://material.io/tools/icons/?style=baseline>

Myjson. Geraadpleegd op 22 februari 2018 via <http://myjson.com/>

RubenVerborgh . Open Weblides WebFundamentals. Geraadpleegd op 13 maart 2018 via <https://github.com/RubenVerborgh/WebFundamentals>

NodeJs. Geraadpleegd op 19 maart 2018 via www.nodejs.org/en

Jest 22.4. (JavaScript Testing). Geraadpleegd op 19 maart 2018 via <https://facebook.github.io/jest/en/>

Babel. Geraadpleegd op 30 mei 2018 via <https://babeljs.io/docs/setup/#installation>

NPM, jQuery. Geraadpleegd op 30 mei 2018 via <https://www.npmjs.com/package/jquery>

Docker. Geraadpleegd op 21 maart 2018 via <https://www.docker.com/>

MongoDB. Geraadpleegd op 1 maart 2018 via <https://www.mongodb.com/>

MongoDB Java Drivers. Geraadpleegd op 6 maart 2018 via <https://mongodb.github.io/mongo-java-driver/>

JavaScript api. Geraadpleegd op 24 februari 2018 via <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference>

HTML, Javascript, CSS. Geraadpleegd op 20 februari 2018 via <https://www.w3schools.com/>