|  |
| --- |
|  |
| Cloud Applications |
|  |
| Automatisering BAP |

September 26, 2016

Opgesteld door: Bryan Lameir, Kay Elst

Cloud Applications

Automatisering BAP

# Project Omschrijving

**Probleemstelling:**

Ten eerste moeten de promotors van de bachelorproeven manueel de gegevens van de studenten, en hun corresponderende bachelorproef opzoeken in een excelfile. Vervolgens kunnen de promotors op deze manier de link naar een repository op github openen om hier de nodige commentaren of issues te maken. Dit is niet efficiënt aangezien ze deze acties moeten uitvoeren voor elke repository. Aangezien het aantal studenten elk jaar groter wordt, wordt dit een langdurige vervelende opdracht. Tenslotte hebben we nog het opvolgen van de repository. Het is tijdrovend om zaken op te zoeken zoals: het aantal commits sinds laatste login, aantal openstaande issues, laatste logentry etc.

**Doel van het project:**

De doel van het project is om deze opdrachten zoveel mogelijk te automatiseren of zelfs overbodig te maken. We gaan een website maken die je kan openen in de browser. Alle handelingen die in bovenstaande probleemstelling zijn vernoemd zullen uitvoerbaar zijn via de website. Een extra feature die we gaan invoegen is dat er verschillende blokken commentaar omgevormd zullen worden tot een issue indien deze in de scriptie gemaakt zijn. Op deze manier zullen de promotors één platform hebben in plaats van alles manueel uit te voeren en/of op te zoeken. De Excelfile zal vervangen worden door een database waarin alle gegevens van de studenten, en wie hun promotor is wordt opgeslagen. De interactie tussen de site en de repository op github zal gebeuren via de github API. Er zal in één oogopslag de recente activiteit gezien kunnen worden zodat de promotor de student gemakkelijk kan opvolgen. Verder zal voor het gemak van de leerkracht alle github namen herleid worden naar de echte namen zodat de docenten een beter overzicht hebben. De logs worden naar git gepushed met een tag, vanaf dat de github API deze tag registreerd zal de laatste log van de branch gehaald worden en er wordt een mail naar de leerkrachten gestuurd zodat deze hiervan op de hoogte zijn. Als de repo niet is ingedeeld zoals het voorbeeld zal er een error te zien zijn bij deze persoon zodat de docenten de leerling hierop kunnen wijzen. Alle Markdown files zullen ook verwerkt worden zodat ze gelezen kunnen worden via de website. Zoals de Markdown files verwerkt worden zal er ook een Code Viewer geïntegreerd worden zodat er commentaar bijgeschreven kan worden. De Readme Zal uitgelezen worden en de inhoud wordt mee verwerkt in de layout van de website. Hierdoor zullen bijvoorbeeld de naam van de student en de namen van de promotors kunnen worden uitgelezen op een automatische manier. Tenslotte willen we er ook graag voor zorgen dat de studenten een mailtje krijgen als er iets veranderd in het originele studentenrepo, op deze manier zijn ze op de hoogte gebracht en kunnen ze aanpassingen maken.

**Methodologie:**

Voor dit project gaan we de SCRUM methodologie toepassen. De SCRUM methodologie wordt vooral toegepast indien er nog geen exact beeld is van het eindproduct maar dat er wel requirements zijn waaraan het eindproduct moet voldoen. SCRUM bevat een backlog waarin alle user stories zitten, deze user stories moeten tegen het einde van het project ook daadwerkelijk waar zijn. De tijdspan waarover het project zicht strekt wordt opgedeeld in sprints. Deze sprints zijn korte periodes van enkele weken waarin enkele doelen worden opgesteld die gerealiseerd moeten worden tegen het einde van de sprint, dit is de planning van de sprint. Ons project is opgedeeld in sprints die 2 weken duren, gedurende deze weken moeten we de planning realiseren. Na elke sprint wordt er samengezeten met de opdrachtgevers om te overleggen over de vooruitgang en de status van het eindproduct en om te kijken of er eventueel plannen zijn veranderd of ideeën zijn bijgekomen. Op deze manier wordt het eindproduct na elke planning concreter en zal het na de laatste sprint een eindproduct zijn die voldoet aan de eisen van de opdrachtgever, met behulp van zijn input na elke sprint.

**Verwachte Resultaten:**

Het verwachte eindresultaat is een website te creëren waarop alle handelingen, die in de probleemstelling staan, uitgevoerd kunnen worden. Op deze manier moet het leven van de promotors gesimplificeerd worden, tevens zou het ook eenvoudiger en minder tijdrovend moeten zijn dan de manuele variant als er meer studenten zijn. De website zal volledig op zichzelf werken zodat er niet meer naar github gegaan hoeft te worden.

# Technologie

*Beschrijving en of schematische voorstelling van de gebruikte technologieën*

*We gaan een webapplicatie maken met behulp van volgende Technologieën:*

*Voor de taal van het coderen gaan we gebruik maken van Java. We hebben gekozen voor Java omdat we hier ervaren in zijn en het is onze favoriete taal voor het coderen van websites. Voor de Lay-out gaan we bootstrap en HTML gebruiken omdat dit een goede optie is voor non-designers om hun website op een eenvoudige manier mooi te maken. We gaan Angular gebruiken voor het binden van onze data in de HTML wegens zijn compatibiliteit met Javascript. De Node en express-server gaan communiceren met de API en de database, we gebruiken deze omdat we hier al ervaring mee hebben en het wordt geschreven in dezelfde taal als onze algemene code. Voor access tot de API gebruiken we OAuthorization en de database is MongoDB. Met github gaan we onze progress bijhouden en samen aan hetzelfde project werken en eventueel om oude commits op te vragen. Tenslotte gebruiken we balsamiQ om onze visie van de website en een eerste layout van het eindproduct aan de opdrachtgevers te tonen.*

# Uitkomst voor de maatschappij

*Omschrijving van max. 1500 lettertekens (inclusief spaties).*

*Sector/partners, maatschappelijke dienstverlening, bijdrage op het vlak van duurzame ontwikkeling,…*

*Ons Project moet promotors helpen om op een eenvoudige manier de stages op te volgen. Het hoofddoel is dat de promotors efficiënter met hun tijd zullen kunnen omgaan en dus diepgaander kunnen werken over eenzelfde tijdspanne. Door veel kleine vervelende zaken te automatiseren en veel te automatiseren gaat de promotor ook minder opzien tegen het opvolgen van repositories wat op zijn beurt niet zorgt voor een burnout.*

# User stories / Actoren

|  |
| --- |
| As a Teacher I want to be able to create a single issue for multiple comments on the scription. |
| As a Teacher I want to be able to see information about a repo on the site with a single buttonpress. |
| As a Teacher I want to be able to see how active the student is within the repo. |
| As a Teacher I want to be able to open repos without having to look up the link. |
| As a Teacher I want to be able to write comments on a students’ commit on the website. |
| As a Teacher I want to be able to Login / Filter the students by promotor so I don’t have to search for them | |
| As a Teacher I want to be able to search for Students in the list of all my students | |
| As a Teacher I want to be able to add/remove/edit students from the list. | |
| As a Teacher I want to have a good looking website with a good overview. | |
| As a Teacher I want to be able to create notes for my students only visible by me. | |
| As a Teacher I want to receive a mail when a new logfile is pushed to github. | |
| As a Teacher I want to be able to read and edit code using the website | |
| As a Teacher I want to be able to read markdown files on the website. | |
| As a Developer I want basic information to be extracted automatically. | |
| As a Student I want to be notified when the original repo changes | |

# Mockups

BalsamiQ

# Links

* *Github*
* *Github projects*