



TECHNISCH ONTWERP

1

Internet of Things: GitHub

Shared Service Center
Scalda Vlissingen
Edisonweg 4a

Projectleider	: Cihan Toksöz
Projectleden	: Maikel Evegroen
Datum	: 20 november 2019
Klas	: ICO43C



Inhoudsopgave

1 Inleiding	3
2 Fysiek ontwerp	4
Opslagruimte	4
CPU 5	
Geheugen	5
Database	5
Opsomming te verrichte activiteiten	6
Te gebruiken apparaten en / of omgevingen	6
3 Inrichting omgeving	6
Installatieoverzicht	7
Inrichting permissies	7
Applicaties	7
4 Inrichting services en gebruikers	8
Personal user account	8
Organization accounts	9
Enterprise accounts	13
Overdracht van repositories	14
5 Testplan	15



1 Inleiding

Binnen in het Shared Service Center zou er een behoefte zijn voor een nieuwe GitHub omgeving. De GitHub omgeving dient software te waarborgen door middel van versiebeheersysteem, waardoor GitHub alle mogelijkheden van Git en eigen toevoegingen aanbiedt met bepaalde functionaliteiten. Technisch gerealiseerd worden

Dit technisch ontwerp geeft antwoord op de vraag hoe kan het met de opdrachtgever afgestemde functioneel ontwerp ten behoeve van een acceptatieplan of een concept, welke functionaliteiten willen we gebruiken en hoe kunnen de functionaliteiten getest worden in een simpele GitHub omgeving.

Het technisch ontwerp geeft een beeld met van de functionaliteiten die met de opdrachtgever zijn vastgelegd, geeft een gedetailleerde beschrijving van de infrastructuur die de functionaliteit mogelijk maakt en geeft aan hoe de totaaloplossing bij de opdrachtgever kan worden getest.



2 Fysiek ontwerp

Ontwerpen doe je vaak samen met andere mensen. Daarbij is het van belang dat alle activiteiten en beschikbare informatie overzichtelijk en duidelijk worden weergegeven. Zo weet iedereen welke informatie er is, wat er gedaan is, en wat er nog moet gebeuren. Een fysiek model is een realistisch en tastbaar driedimensionaal model van een ontwerp.

ESP MQTT JSON Multisensor

This project shows a super easy way to get started with your own IoT Multisensor to use with Home Assistant, a nice open source Home Automation platform that can do just about anything.

Bonus: This project requires no soldering and no breadboards - just header wires and the development board!

Video Tutorial: <https://youtu.be/gpRr5m>

The code provided in this repository will run Home Assistant's MQTT JSON light Component, MQTT Sensor Component, and a NodeMCU ESP8265 development board.

Supported Features Include

- DHT22 temperature sensor
- DHT22 humidity sensor
- AM2312 PHS precision sensor
 - photodiode or TEM1600 light sensor
 - RGB led with support for color, flash, fade, and transition
 - Over-the-Air (OTA) upload from the Arduino IDE

OTA Uploading

This code also supports remote uploading to the ESP8266 using Arduino's OTA library. To utilize this, you'll need to first upload the sketch using the traditional USB method. However, if you need to update your code after that, your WiFi-connected ESP should show up as an option under Tools -> Boot -> Upload to your platform's IDE. More information on OTA uploading can be found here. Note: You cannot access the serial monitor over WiFi at this point.

Parts List

Amazon Prime (fast shipping)

- NodeMCU v2
- SparkFun Module
- 10K Potentiometer

ESP MQTT JSON Multisensor

Het plan zoals beschreven staat in het functioneel ontwerp is om een versiebeheersysteem te ontwikkelen voor de kenniscgroep IOT die als doel heeft om files als README, CONFIG files of andere bestanden in mappenstructuur te kunnen waarborgen onder de server van GitHub. Bij mogelijke uitbreiding moet er gebruik gemaakt kunnen worden in heel het Shared Service Center.

GitLab is ontwikkeld voor op Linux gebaseerde besturingssystemen. GitLab vereist Ruby (MRI) 2.6. Ondersteuning voor Ruby-versies onder 2.6 (2.4, 2.5) stopt met GitLab 12.2. Ruby is een programmeertaal. De taal is ontworpen om snel en makkelijk object georiënteerd te programmeren.

Opslagruimte

De benodigde ruimte op de harde schijf hangt grotendeels af van de grootte van de repo's die u in GitLab wilt opslaan, maar als vuistregel moet u ten minste evenveel vrije ruimte hebben als al uw repo's samen in beslag nemen.

Als je in de toekomst flexibel wilt zijn in het uitbreiden van de harde schijfruimte, overweeg dan om deze met LVM te monteren zodat je meer harde schijven kunt toevoegen wanneer je ze nodig hebt.

Behalve een lokale harde schijf kun je ook een volume mounten dat het NFS-protocol (Network File System) ondersteunt. Dit volume bevindt zich mogelijk op een bestandsserver, een NAS-apparaat (Network Attached Storage), SAN (Storage Area Network) of op een Elastic Block Store (EBS) -volume van Amazon Web Services (AWS).

Als je genoeg RAM-geheugen en een recente CPU hebt, wordt de snelheid van GitLab voornamelijk beperkt door de zoektijden van de harde schijf. Het hebben van een snelle



schijf (7200 RPM en hoger) of een solid-state schijf (SSD) zal de responsiviteit van GitLab verbeteren.

CPU

Dit is de aanbevolen minimale hardware voor een handvol voorbeelden van GitLab-gebruikersbasisformaten. Uw exacte behoeften kunnen meer zijn, afhankelijk van je werklast. Je werklast wordt beïnvloed door factoren zoals - maar niet beperkt tot - hoe actief je gebruikers zijn, hoeveel automatisering je gebruikt, mirroring en repo / verander grootte.

- 1 kern ondersteunt maximaal 100 gebruikers, maar de applicatie kan wat langzamer zijn omdat alle werknemers en achtergrondtaken op dezelfde kern worden uitgevoerd
- 2 cores is het aanbevolen minimum aantal cores en ondersteunt maximaal 100 gebruikers
- 4 cores ondersteunt maximaal 500 gebruikers
- 8 cores ondersteunen maximaal 1.000 gebruikers
- 32 cores ondersteunt maximaal 5.000 gebruikers

Geheugen

Je hebt minimaal 8 GB adresseerbaar geheugen (RAM + swap) nodig om GitLab te installeren en te gebruiken! Het besturingssysteem en alle andere actieve toepassingen gebruiken ook geheugen, dus houd er rekening mee dat u minimaal 4 GB beschikbaar moet hebben voordat je GitLab uitvoert. Met minder geheugen geeft GitLab vreemde fouten tijdens de herconfiguratie en 500 fouten tijdens het gebruik.

- 4 GB RAM + 4 GB swap ondersteunt maximaal 100 gebruikers, maar het zal erg langzaam zijn
- 8 GB RAM is de aanbevolen minimale geheugengrootte voor alle installaties en ondersteunt maximaal 100 gebruikers
- 16 GB RAM ondersteunt maximaal 500 gebruikers
- 32 GB RAM ondersteunt maximaal 1.000 gebruikers
- 128 GB RAM ondersteunt tot 5.000 gebruikers

Database

De server die de database uitvoert, moet ten minste 5-10 GB opslagruimte beschikbaar hebben, hoewel de exacte vereisten afhankelijk zijn van de grootte van de GitLab-installatie (bijvoorbeeld het aantal gebruikers, projecten, enz.).

De volgende database wordt ondersteunt:

- PostgreSQL

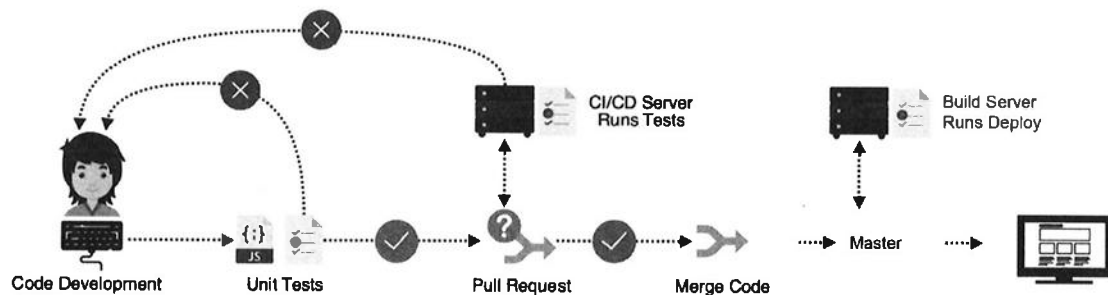


Opsomming te verrichte activiteiten

- Repository aanmaken voor het organiseren van een project
- Branch aanmaken om tegelijkertijd aan verschillende versies van een repository te werken
- Commit change aanbrengen bij opgeslagen wijzigingen waarin wordt uitgelegd waarom een bepaalde wijziging is aangebracht
- Pull request aanvragen om de wijzigingen te laten bekijken door andere gebruikers
- In de laatste stap is het tijd om je wijzigingen samen te brengen. ReadME en edit branch komen samengevoegd in de master branch

Te gebruiken apparaten en / of omgevingen

Het idee is om een GitHub omgeving te ontwikkelen die wij in het Shared Service Center kunnen gaan gebruiken. Het moet beschikken over toegangscontrole en verschillende samenwerkingsfunctie, zoals issue tracker, een forum voor het aanvragen van functies, takenlijsten en wiki's voor ieder project. Vervolgens is het bedoeld om bestaande documentatie zoals de functionele en technische ontwerpen, leeroffertes en handleidingen vanuit SharePoint. De gebruikelijke books uit de kennisbank en misschien bepaalde dingen vanuit de servicedesk over te zetten naar de nieuwe omgeving die we in heden willen gaan gebruiken. Het openbaar gebruik (local) van GitHub, zal ervoor zorgen dat alle projecten makkelijk te bereiken zullen zijn. GitHub rijkt uit met vele details in vergelijking met de opgeslagen data 's binnenin SharePoint.





3 Inrichting omgeving

Installatieoverzicht

GitHub is een tool dat makkelijk te benaderen valt. Via de website van GitHub kun je makkelijk te werk gaan, maar er is ook een applicatie die te downloaden is via de site van GitHub. Voordat de GitHub Desktop ingesteld wordt, moet er een GitHub- of Github Enterprise account hebben. GitHub Desktop is te installeren op Microsoft Windows 7 of hoger of op een macOS 10.10 of hoger

Waarschuwing: je moet een 64-bits besturingssysteem hebben om GitHub Desktop uit te voeren.

Mac

- 1 Visit the GitHub Desktop download page
- 2 Choose Download for Mac
- 3 In your computer's Downloads folder, double-click the GitHub Desktop zip file
- 4 After the file has been unzipped, double-click GitHub Desktop

Windows

- 1 Visit the GitHub Desktop download page
- 2 Choose Download for Windows
- 3 In your computer's Downloads folder, double-click GitHub Desktop
- 4 In the pop-up window, click install
- 5 After the program has been installed, click Run

Opmerking: als je een netwerkbeheerder bent, kun je het Windows-installatiepakket van GitHub Desktop gebruiken om Github Desktop te implementeren.

Nadat je GitHub Desktop hebt gestart, kun je ervoor kiezen om GitHub Desktop meteen in te stellen, of je kunt het installatieproces overslaan.

Om het wat makkelijker te maken met het uploaden van sommige data's en het bewerken van codes hebben wij via internet een software gevonden.

Eventuele softwares:

- Visual Code
- Gitkraken
- Git Bash

Inrichting permissies

De gebruikers van een repository zullen de permissie hebben om de documentatie te plaatsen en te verwijderen die binnen hun project wordt bewerkt en andere wijzigingen te voldoen. Er zal een kennisgroep binnen IOT aanwezig zijn die het GitHub omgeving zal beheren.

Applicaties

Tijdens de test en gebruik maken wij gebruik van de software Visual Studio Code. Visual Studio Code is een krachtige broncode-editor die op de bureaublad draait en beschikbaar is voor Windows, macOS en Linux. Het wordt ondersteunt door JavaScript, TypeScript en Node.js en heeft een rijk ecosysteem van extensies voor andere talen.



4 Inrichting services en gebruikers

Voor de inrichting van de services en gebruikers zijn er een paar eisen gesteld waar het aan zou moeten voldoen. De te gebruiken services in ons GitHub project zijn: code, issues, pull request, projects, wiki en een gebruikershandleiding, die zal uitleggen hoe GitHub beheert wordt. Voor de gebruikte hard- en software is een gratis licentie beschikbaar als je een student bent met een schoolaccount die je dan vervolgens moet koppelen aan GitHub, zodat je gratis licentie krijgt met de functies. Hoewel er lees- / schrijftoegang verleend kan worden aan bijdragers in een persoonlijk repository, kunnen leden van het Shared Service Center gedetailleerde toegangsmachtigingen hebben voor de repository. Binnen GitHub zijn 3 soorten account die verleend kunnen worden:

- Personal user accounts
- Organization accounts
- Enterprise accounts

Personal user account

Een repository die eigendom is van een gebruikersaccount heeft twee machtigingsniveaus: de repository-eigenaar en bijdragers. Als er gedetailleerde lees- / schrijftoegang nodig zal zijn tot een repository die eigendom is van de gebruikersaccount, kan het overwogen worden de repository over te dragen aan een organisatie. De eigenaar van de repository heeft volledige controle over de repository. Naast alle machtigingen die door repository leden zijn toegestaan, kan de eigenaar van de repository:

- Medewerkers uitnodigen
- De zichtbaarheid van de repository wijzigen (van openbaar naar privé of van privé naar openbaar)
- Beperk interacties met een repository
- Een pull-aanvraag samenvoegen op een beveiligde branch, zelfs als er geen goedkeurende beoordelingen zijn
- Verwijder de repository
- Beheer de onderwerpen van een repository
- Schakel de afhankelijkheidsgrafiek in voor een privérepository
- Pakketten verwijderen
- Maak en bewerk repository sociale kaarten
- Maak van de repository een sjabloon
- Ontvang beveiligingswaarschuwingen voor kwetsbare afhankelijkheden in een repository
- Meld u aan voor gegevensgebruik voor een privérepository
- Definieer code-eigenaren voor de repository
- Archiefrepository's
- Maak beveiligingsadviezen
- Toon een sponsorknop

Er is maar een eigenaar van een repository die eigendom is van een gebruikersaccount: deze toestemming kan niet worden gedeeld met een ander gebruikersaccount. In een privérepository kunnen eigenaren van repositories alleen schrijfrechten verlenen aan bijdragers. Leden kunnen geen alleen-lezen toegang hebben tot repositories die eigendom zijn van een gebruikersaccount.



Medewerkers op een persoonlijke repository kunnen:

- Druk op (schrijven), trek uit (lezen) en vork (kopieer) de repository
- Labels, mijlpalen maken, toepassen en verwijderen
- Problemen openen, sluiten, opnieuw openen en toewijzen
- Bewerk en verwijder opmerkingen over commits, pull-aanvragen en problemen
- Markeer een probleem of pull-verzoek als duplicaat
- Openingsverzoeken openen, samenvoegen en sluiten
- Voorgestelde wijzigingen toepassen op pull-aanvragen
- Stuur pull-aanvragen van vorken van de repository
- Pakketten publiceren, bekijken en installeren
- Wiki's maken en bewerken
- Maak en bewerk releases
- Verwijder zichzelf als bijdragers aan de repository
- Dien een beoordeling in voor een pull-aanvraag die de samenvoegbaarheid beïnvloedt
- Fungeren als een aangewezen code-eigenaar voor de repository
- Vergrendel een gesprek
- Meld misbruik aan GitHub Support of GitHub Premium Support
- Breng een probleem over naar een andere repository

9

Organization accounts

Nadat er een organisatie is gemaakt, moeten de eigenaarsrechten gegeven worden aan een kleine groep mensen die het organisatieaccount zullen beheren. De organisatie leden kunnen verdeeld worden onder de eigenaar van het project, leden en billing manager.

- Eigenaars hebben volledige beheerderstoegang tot de organisatie. Deze rol moet worden beperkt tot een paar mensen in de organisatie.
- Billing managers kunnen een persoon factureringinstellingen beheren. Dat is niet iets wat van toepassing zal zijn
- Normale leden hebben de standaardrollen

Organisatie actie	Eigenaren	leden	Factureringsmanagers
Maak repositories	X	X	
Factureringsinformatie bekijken en bewerken	X		X
Nodig mensen uit om lid te worden van de organisatie	X		
Bewerk en annuleer uitnodigingen om lid te worden van de organisatie	X		
Verwijder leden uit de organisatie	X		
Herstel voormalige leden van de organisatie	X		
Voeg mensen toe en verwijder ze uit alle teams	X		
Promoot organisatieleden tot <i>teambeheerder</i>	X		
Configureer toewijzingen voor codereview	X		
Geplande herinneringen instellen	X		
Voeg bijdragers toe aan alle repositories	X		
Open het auditlogboek van de organisatie	X		
Bewerk de profielpagina van de organisatie	X		
Controleer de domeinen van de organisatie	X		
Verwijder alle teams	X		
Verwijder het organisatieaccount, inclusief alle repositories	X		



Maak teams	X	X	
Teams verplaatsen in de hiërarchie van een organisatie	X		
Maak projectborden	X	X	
Bekijk alle organisatieleden en teams	X	X	
@vermelding elk zichtbaar team	X	X	
Kan een <i>teambeheerder</i> worden	X	X	
Bekijk organisatie-inzichten	X	X	
Bekijk en post openbare teamdiscussies voor alle teams	X	X	
Privé-teamdiscussies bekijken en posten voor alle teams	X		
Bewerk en verwijder teamdiscussies in alle teams	X		
Verberg opmerkingen over commits, pull-aanvragen en problemen	X	X	
Schakel teamdiscussies uit voor een organisatie	X		
Beheer de weergave van inzichten in afhankelijkheid van organisaties	X		
Stel een teamprofielfoto in alle teams in	X		
SAML single sign-on inschakelen en afdwingen	X		
Beheer de SSH-certificaatautoriteiten van een organisatie	X		
Repositories overdragen	X		



Koop, installeer, beheer facturering voor en annuleer GitHub Marketplace-apps	X		
Lijst met apps in GitHub Marketplace	X		
Ontvang beveiligingsmeldingen over kwetsbare afhankelijkheden voor alle repositories van een organisatie	X		
Beheer geautomatiseerde beveiligingsupdates	X		
Vorken toestaan of uitschakelen voor alle privérepository's in een organisatie	X		
Beperk activiteit in openbare repositories in een organisatie	X		
Trek (lees), push (schrijf) en kloon (kopieer) <i>alle repositories</i> in de organisatie	X		
Converteer organisatieleden naar externe medewerkers	X		
Bekijk mensen met toegang tot een organisatie-repository	X		
Een lijst met mensen met toegang tot een organisatierepository exporteren	X		
Beheer standaardlabels	X		
Teamsynchronisatie inschakelen	X		



Enterprise accounts

Enterprise-accounts worden geleverd met een set beheerdersrollen die u kunt toewijzen aan gebruikers van uw onderneming. Elke beheerdersrol wordt toegewezen aan bedrijfsfuncties en biedt machtigingen voor het uitvoeren van specifieke taken binnen het enterprise-account.

Momenteel zijn er twee beheerdersrollen beschikbaar in enterprise-accounts: bedrijfseigenaren, die toegang hebben tot en toegang hebben tot alle instellingen in het account, en factureringsbeheerders, die toegang hebben tot alleen de factureringsinstellingen van het enterprise-account en deze kunnen beheren. Ondernemingseigenaren kunnen ook alle leden en externe medewerkers zien voor elke organisatie die eigendom is van het ondernemingsaccount.

Enterprise eigenaar

Ondernemers hebben volledige controle over het bedrijfsaccount en kunnen elke actie ondernemen, waaronder:

- Beheer beheerders
- Organisaties toevoegen en verwijderen van en naar de onderneming
- Bedrijfsinstellingen beheren
- Beleid afdwingen tussen organisaties
- Factureringsinstellingen beheren

Ondernemers van een bedrijf hebben geen toegang tot organisatie-instellingen of inhoud tenzij ze eigenaar zijn van een organisatie of directe toegang krijgen tot een repository van de organisatie. Evenzo hebben eigenaren van organisaties in uw enterprise-account geen toegang tot het enterprise-account zelf, tenzij u hen enterprise-eigenaars maakt.

Je kunt zoveel enterprise-eigenaren toevoegen als u wilt aan uw enterprise-account. Ondernemers moeten een persoonlijk account hebben op GitHub.

Enterprise leden

Leden van organisaties die eigendom zijn van de enterprise-account zijn ook automatisch lid van de enterprise-account. Leden kunnen samenwerken in organisaties en kunnen organisatie-eigenaren zijn, maar leden hebben geen toegang tot of het configureren van enterprise account-instellingen, inclusief factuurinstellingen.

Mensen in uw enterprise-account kunnen verschillende toegangsniveaus hebben tot de verschillende organisaties die eigendom zijn van uw enterprise-account en tot repositories binnen die organisaties. U kunt de bronnen bekijken waartoe elke persoon toegang heeft. Mensen met externe medewerkerstoegang tot opslagplaatsen die eigendom zijn van uw organisatie, worden ook vermeld op het tabblad Mensen van uw enterprise-account, maar zijn geen enterprise-leden en hebben geen toegang tot de enterprise-account.



Overdracht van repositories

Wanneer een repository overgedragen wordt aan een nieuwe eigenaar, kunnen deze onmiddellijk de inhoud, problemen, pull-aanvragen, releases, projectboards en instellingen van de repository beheren.

Vereisten voor repository overdracht:

- Wanneer u een repository die je bezit naar een ander gebruikersaccount overbrengt, ontvangt de nieuwe eigenaar een bevestigingsmail. De bevestigingsmail bevat instructies voor het accepteren van de overdracht. Als de nieuwe eigenaar de overdracht niet binnen één dag accepteert, vervalt de uitnodiging.
- Om een repository die je bezit naar een organisatie over te dragen, moet u toestemming hebben om een repository in de doelorganisatie te maken.
- Het doelaccount mag geen repository met dezelfde naam of een vork in hetzelfde netwerk hebben.
- De oorspronkelijke eigenaar van de repository wordt toegevoegd als bijdrager aan de overgedragen repository. Andere bijdragers aan de overgedragen repository blijven intact.
- Privévorken kunnen niet worden overgedragen.

Als u een privérepository overdraagt naar een gebruikersaccount met GitHub Free, verliest het account toegang tot functies zoals GitHub-pagina's. Wanneer u een repository overdraagt, worden de problemen, pull-aanvragen, wiki, sterren en watchers ook overgedragen. Als de overgedragen repository webhooks, services of implementatiesleutels bevat, blijven deze gekoppeld nadat de overdracht is voltooid. Je kunt uw repository overbrengen naar elk gebruikersaccount dat uw repository-overdracht accepteert. Wanneer een repository wordt overgedragen tussen twee gebruikersaccounts, worden de oorspronkelijke repository-eigenaar en bijdragers automatisch als bijdragers aan de nieuwe repository toegevoegd. Als je een GitHub-pagina site in een privérepository hebt gepubliceerd en een aangepast domein hebt toegevoegd, voordat u de repository overdraagt, wilt u misschien uw DNS-records verwijderen of bijwerken om het risico van een domeinovername te voorkomen.



5 Testplan

Ons project wordt uiteindelijk getest omdat 'GitHub' één groot platform is waar allemaal projecten aangemaakt worden en daarin allemaal verschillende documenten en informatie worden bewaard die goed beheert moet blijven worden, waar gebruikers van die projecten elke keer bij moeten kunnen komen zonder problemen. Aan het begin van een project stel je een testplan op. dit is vergelijkbaar met een plan van aanpak die je opstelt. Het doel van het Testplan is om de aanpak van het testen binnen het programma, project, release of change te beschrijven. En alles wat nodig is om die aanpak uit te kunnen voeren op het juiste moment en met welke mensen en middelen.

De onderdelen die behandeld kunnen worden zijn:

- Aanleiding (waarom)
- Aanpak (hoe)
- Benodigde mensen en middelen (wie en wat)
- Benodigde tijd (planning) (wat)
- Op te leveren producten (wat)



Wij als projectgroep van GitHub dat geleid wordt door Cihan Toksöz met zijn projectlid Maikel Evegroen, zullen onder de projectleider Jeroen van Os de testomgeving testen voor mogelijke feedback. De testen zullen na de opgestelde documentaties getest worden. De laatste documentatie (technisch ontwerp) zal in de 50^e week opgeleverd worden, waarna gelijk een acceptatietest gestart zal worden. De testopstelling zal uitgevoerd worden waar het mogelijk gebruikt zal worden. GitHub is een makkelijk bereikbaar platform, wat overal getest kan worden. GitHub zal te zonder enige probleem te bereiken zijn via het netwerk van Scalda. In heel het Shared Service Center is het mogelijk om de test uit te voeren. De mogelijkheid zal er ook zijn om te testen op thuisbasis via het eigen netwerk. De test zal uitgevoerd worden om te kijken of het hele versiebeheerplatform 'GitHub' van het project te benaderen valt.

Testnummer	Test	Uitvoering	Resultaat
1	Alle documenten en projecten zijn terug te vinden die aangemaakt zijn		
2	Issues kunnen gemeld worden		
3	Er kunnen contributors (leden) toegevoegd worden en verwijderd worden		
4	Er kan een repository gemaakt worden		
5	De projecten zijn local bereikbaar		
6	Kan er gemakkelijk een pull request gemaakt worden?		
7	Het verwerken van codes lukt soepel		
8	Het gebruiken van een workflow is beschikbaar		
9	De projecten kunnen privé gezet worden		



Handtekeningen	
Student: Naam : Cihan Toksöz	
Projectcoach: Naam : Jeroen van Os	
Opdrachtgever: Naam : Kees Krijger	
9 december 2019	