Portfolio semester 3

# Individueel

## Week 1

In week 1 was het vooral een opstart week en heb ik me vooral ingelezen op canvas. Ook heb ik veel nagedacht over wat ik dit semester wil gaan maken, hier heb ik nu een redelijk beeld bij. Op vrijdag heb ik alvast gekeken naar de leerdoelen en de technieken die ik wil gaan gebruiken. Dit is nog niet helemaal gelukt en zal ik volgende week mee verder gaan.

## Week 2

Deze week heb ik een planning gemaakt en aan mijn plan voor mijn individueel project gewerkt.

## Week 3

Vorige week had ik mijn plan afgemaakt en daarom kon ik deze week aan de slag met mijn story mapping en mijn backlog.

## Week 4

Deze week was de sprint oplevering van sprint 1. Daarom heb ik deze week mijn architectuur document opgesteld. Ik heb alle 4 mijn diagrammen gemaakt en toelichting bij gegeven ook heb ik antwoord gegeven op de vragen die in de module staat.

## Week 5

In week ben ik eindelijk begonnen aan een klein vue projectje om er mee te oefenen, zelf vond ik dit eigenlijk te laat en had ik graag vorige week willen doen. Ook heb ik deze week gekeken naar ORM en databases

## Week 6

Ik heb en vue tutorial gevolgd en een front end gemaakt voor mijn eigen project. En onderzoek gedaan naar ORM

## Week 7

Deze week heb ik een ORM database opgezet en begonnen aan een back end. Deze heb ik geprobeerd om te koppelen met mijn front end, dit is nog niet helemaal gelukt en ga ik volgende week meteen aan beginnen

## Week 8

Deze week heb ik een gesprek gehad met patrick, hierna heb ik bij mijn get een post methode toegevoegd maar deze zit nog niet in mijn front-end. Ook heb ik een 2e tabel gemaakt,

## Week 9

In de vakantie heb ik gekeken naar foreign keys via entity framework en heb ik gekeken naar websockets. Deze week ben ik verder gegaan met websockets. Ik heb nu 3 servers opgezet om er een gevoel voor te krijgen. Volgende week wil ik de berichten laten weergeven in plaats van de nu staande get met een database. Dit betekent wel dat mijn rest nu vervangen moet worden samen met de tussen tabel. Ik kan dit wel later terug brengen met accounts en users. Voor volgende week wil ik meteen beginnen aan het laten zien op mijn Front end en een test framework zoals cypress toepassen.

## Week 10

Deze week heb ik me meer op andere leerdoelen gefocust, ik heb eerst wat gedaan aan mijn websocket. Op de dinsdag ben ik bezig geweest met cypress en het opzetten van unit testen voor mijn back end. Op de vijdag heb ik gekeken naar ci/cd en heb ik een pipeline en een runner toegevoegd op mijn github. Voor volgende week wil ik meteen een gesprek met Patrick zodat hij tips kan geven over hoe nu verder te gaan.

# Groep

## Week 1

De eerste week voor de groep was vooral kennis maken met elkaar en de product owner. We hebben als groep ook al gekeken naar de opstart en een trello opgezet. Ook hebben we vast het gemaakte werk wat wij moeten uitbreiden doorgespit om zo tot antwoorden te komen over hoe de applicatie in elkaar zit.

## Week 2

In week 2 heb ik met mijn groep al de docker containers opgezet een gekeken of we het programma aan de praat konden krijgen. Dit was helaas nog niet bij iedereen gelukt.

## Week 3

In deze week hebben we voor iedereen alles aan de praat gekregen en hebben we het lokaal laten runnen op een eigen database en met behulp van docker kunnen we nu ook alles tegelijk laten draaien.

## Week 4

Voor onze groepsopdracht heb ik deze week hebben we een verbeterde readme geschreven en we hebben alles draaiend gekregen zonder docker, in Visual Studio.

## Week 5

Voor Ipost hebben we wat kleine issues gevonden en in groepjes opgelost. Ik ben met Axel bezig geweest met de pdf/ QR code en hier gaan we volgende week mee verder.

## Week 6

Voor het groepsproject heb ik samen met axel de qr code afgemaakt en aan Jacques laten zien.

## Week 7

Deze week heb ik voor het groepsproject samen met Stan gewerkt aan het steden overzicht om dit er duidelijker uit te laten zien en zodat er meerdere tabellen weergegeven kunnen worden

## Week 8

Ik heb met Stan verder gewerkt aan het overzicht, dit is nu af en we zijn begonnen aan het gebouwen overzicht.

## Week 9

Deze week was sprint oplevering, in het begin hebben stan en ik verder gewerkt aan de overzichten. En zijn we bezig geweest om een database toe te voegen aan de personeel service. Hier zijn we volgende week wel mee klaar

## Week 10

Stan en ik hebben verder gewerkt aan de database toevoegen, daarna hebben we al een klein beetje gekeken naar de nfc scanner en het uitlezen van een pasje.

# Feedback



# Leerdoelen

You design and build user-friendly, full stack web applications.

* Er moet een applicatie worden neergezet met een database, front end en back end waarvan de front in Javascript framework en de back end een OO taal is. Deze applicatie moet duidelijk en gebruiksvriendelijk zijn.
* Ik heb meerdere diagrammen gemaakt die aangeven dat ik een full-stack applicatie ga maken. Ik heb een componenten diagram gemaakt die duidelijk weergeeft welke onderdelen ik ga gebruiken. Ik heb een back-end gemaakt in .NET en een front-end in Vue.js. Mijn database is met entity framework gemaakt. Er word gebruik gemaakt van REST en van websockets.
* De front end moet gemaakt worden in een JavaScript framework.

De meest populairste frameworks zijn Angular, React en Vue.js.

|  |  |  |  |
| --- | --- | --- | --- |
| **Criteria** | **React** | **Angular** | **Vue** |
| Onderhoudbaarheid | Niet heel erg onderhoudbaar | Goed onderhoudbaar | Goed onderhoudbaar |
| Leerbaarheid/bruikbaarheid | Niet moeilijk maar ook niet makkelijk te leren/gebruiken | Lastig te leren/ gebruiken | Makkelijk te leren/te gebruiken (moet het leren voor groepsproject |

Ik heb gelezen dat Vue makkelijk te onderhouden. Dit is bij React niet altijd het geval maar bij Angular wel. Daarom valt React al af bij mij. Voor nu lijkt blijft dan Angular en Vue over. Vue is makkelijk te gebruiken, dit is iets waar Angular iets minder makkelijk in is. In de groepsopdracht wordt met Vue wordt gewerkt. Ik weet van mezelf dat als ik teveel talen en technieken tegelijk ga leren dat het dan voor mij niet bij te houden is. Daarom ga wegens bovenstaande redenen gebruik maken van Vue.js bij mijn project.

Voor de back-end moet een OO taal worden gebruikt. Deze wordt waarschijnlijk met het framework van .Net, dit komt omdat ik vorig semester hier ook mee heb gewerkt. Ik heb dus al enige ervaring met dit framework, ook heb ik vorige semester wat fouten gemaakt hiermee en er waren volgens Jelle Oosterkamp (mijn docent vorig semester) dus nog wat verbeter punten. Ik denk daarom dat nog een poging op asp een goede stap is. Ook zie ik na onderzoek dat de veiligheid van java lager is dan bij .Net In beide back-end talen kunnen goede frameworks worden toegevoegd dus hier is in principe geen probleem

You use software **tooling and methodology** that continuously monitors and improve the software quality during software development.

* De applicatie moet onderhoud baar zijn doormiddel van testen, veiligheid en duidelijke code.
* Ik heb onderzoek gedaan naar de beste frameworks om te gebruiken in mijn applicatie.
* Ik heb gekeken naar test. Voor mijn front end test wil ik gebruik gaan maken voor cypress omdat dit ook in het groepsproject wordt gebruikt en het er niet heel lastig uitzag. Een ander voorbeeld voor de front end is Jest. *Jest is a JavaScript test framework that is focused on simplicity. One of its unique features is the ability to take snapshots of tests in order to provide an alternative means of verifying units of your application*. Een ander voorbeeld is Mocha *Mocha is a JavaScript test framework that is focused on being flexible. Because of this flexibility, it allows you to choose different libraries to fulfill other common features such as spying (e.g., Sinon) and assertions (e.g., Chai). Another unique feature of Mocha is that it can also execute tests in the browser in addition to Node.js.* Voor de back end tests ga ik schrijven in unit test omdat dit handig is voor .Net en ik er al ervaring mee heb

You **design and implement** a (semi)automated software release process that matches the needs of the project context.

* De applicatie wordt onderhouden op bijvoorbeeld docker of git om nieuwe versies te pushen en continuous intergration aan te houden.
* Ik heb een deployment diagram gemaakt om aan te geven hoe mijn applicatie werkt als hij gedeployed
* Verder heb ik meerdere versies staan in git en push ik regelmatig wat ik recent heb gedaan. Zo is versie beheer duidelijk te zien
* Ik wil nog gaan kijken naar het deployen in docker en misschien naar Jenkins. Hiermee kun je ook continious intergration gebruiken.

You act in a **professional manner**during software development and learning.

* Er wordt op een profesionele wijze gedaan naar groepsgenoten en docenten. Er wordt geluisterd naar de feedback van stakeholders en deze wordt verwerkt in de applicatie en er moet voldoende feedback worden gevraagd
* De architectuur wordt netjes beschreven
* Ik heb mijn vier diagrammen gemaakt om zo mijn architectuur duidelijk aan te kunnen tonen.