Programmeerproject 2:

Voorstudie (fase 1)

Maxim Brabants

0576581

Maxim.Lino.Brabants@vub.be

Academiejaar 2021-2022



Inhoudsopgave

[1 Korte inleiding 3](#_Toc85207098)

[2 Beschrijving ADTs 3](#_Toc85207099)

[2.1 Locomotief 3](#_Toc85207100)

[2.2 Wissel 4](#_Toc85207101)

[2.3 Detectieblok 4](#_Toc85207102)

[3 Afhankelijkheidsdiagram 4](#_Toc85207103)

[4 Werktermijn 4](#_Toc85207104)

# Korte inleiding

Voor het tweede programmeerproject binnen het academiejaar 2021-2022 wordt er van ons verwacht dat we een volledig controlesysteem gaan uitbouwen dat zal instaan voor de aansturing van een modelspoor inclusief de treinen die daarover zullen kunnen rijden. Dit alles zal volledig programmatorisch moeten gebeuren met een opsplitsing in enkele onderdelen. Eerst een kort overzicht van welke onderdelen precies aanwezig moeten zijn en welke aspecten per onderdeel van belang zijn (op deze manier denk ik na over wat precies de bevoegdheid is van elk onderdeel en wat de mogelijkheden zijn):

* Allereerst moet er iets aanwezig zijn dat rechtstreeks met de spoorelementen kan communiceren zodat er op bepaalde momenten een verandering in de spoorsituatie kan plaatsvinden. Hiervoor zal het **Command Station (Z21)** verantwoordelijk zijn. Deze zal a.d.h.v. een bepaald protocol met de hardware communiceren (deze software is reeds voorzien). Dit station gaan we kunnen aansturen door er van buitenaf commando’s naar te sturen.
* Vervolgens hebben we nog een permanent draaiend component dat zal communiceren met de modelbouw hardware. Die software zal voortdurend moeten actief blijven omdat deze juist de signalen zal sturen naar het Command Station. Dit component kunnen we laten draaien op onze computer of afzonderen op een raspberry Pi. **(Infrabel)**
* Als allerlaatst hebben we nog een component met een grafische interface die eigenlijk los moet staan van de logica voor het aansturen van de sporen, maar wel moet instaan voor het berekenen van bijvoorbeeld trajecten. **(NMBS)**

De grote nadruk zal ongetwijfeld ook liggen op de assemblage van al deze elementen tot één groot geheel. We mogen namelijk geen elementen gaan mixen met elkaar, want als we dat gaan doen, wordt het zowel voor ons als voor andere developers onduidelijk waar we elk element kunnen terugvinden.

# Beschrijving ADTs

## Locomotief

Locomotieven vormen een integraal onderdeel binnen dit project. Het algemene idee zegt ons dat we ten alle tijden moeten trachten de garantie te bewaren dat een trein veilig van punt A naar punt B kan reizen. M.a.w. dat een trein zonder enige obstakels (botsingen, ontsporingen,…) bepaalde trajecten moet kunnen afleggen. Qua gedrag denk ik aan het kunnen starten of stoppen van een trein, alsook het versnellen en vertragen van een rijdende locomotief en ten slotte ook de bepaling van de huidige rijrichting.

|  |  |
| --- | --- |
| Operaties | Signatuur |
| start! | (ф 🡪 ф) |
| stop! | (ф 🡪 ф) |
| versnel! | (ф 🡪 ф) |
| vertraag! | (ф 🡪 ф) |
| veranderRijrichting! | (ф 🡪 ф) |

* **start!** zal het signaal geven dat de locomotief in beweging is naar de huidige richting toe.
* **stop!** zal dan weer aangeven dat de trein niet meer rijdt.

## Wissel

## Detectieblok

# Afhankelijkheidsdiagram

Er zijn bepaalde types die niet correct gaan kunnen werken zonder het bestaan van andere. Er is dus een bepaalde afhankelijkheid die dient gerespecteerd te worden om een goede werking van het gehele programma te kunnen garanderen.

# Werktermijn

Een overzicht van de uitgevoerde taken per week :

|  |  |
| --- | --- |
| Week | Taak |
| Fase 1 | |