Personal Development Document

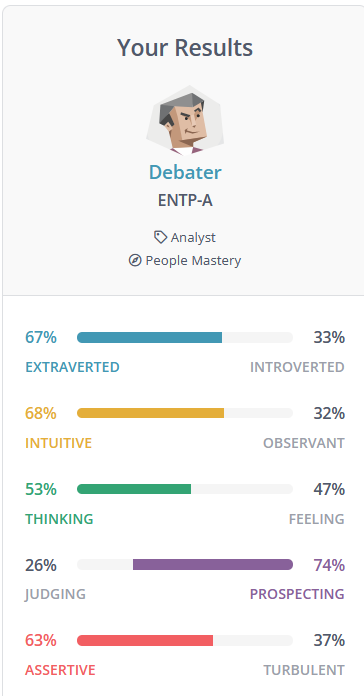
Tiemon Steeghs

 Van: <https://www.freepik.com/free-photos-vectors/technology>

# Persona

Ik ben Tiemon Steeghs, ik ben 17 jaar oud en ik heb voordat ik aan deze opleiding ben begonnen Havo afgerond. Ik woon in Zaltbommel waarvandaan ik elke dag met de trein na school reis. Ik houd onder andere van basketballen, muziek luisteren, gamen, Netflix kijken en chillen. Op mijn middelbare school had ik het combi profiel NG/NT gekozen. Hier heb ik mijn interesse in ICT ontdekt. Ik had toen namelijk het vak informatica gekozen en dit vak sprak mij toen goed aan. Ik heb veel geleerd over de verschillende vakken die ik nu heb. Bij informatica heb ik bijvoorbeeld gewerkt met databases en het maken van een website. Deze dingen zie ik dan weer terug in Media en Business op het Fontys. Na mijn middelbare school ben ik opzoek gegaan naar een opleiding die deze interesses in ICT vervult. Na verschillende opleidingen bekeken te hebben, ben ik bij het Fontys uitgekomen. Het Fontys sprak mij het meeste aan door de uitgebreide keuze die ze aanboden binnen de ICT. Ik wist namelijk nog niet precies wat ik wou doen in de ICT en deze opleiding leek mij het perfecte hulpmiddel hiervoor.

Persoonlijkheids type:



# Project

## Assigned project: Fischertechnik Warehouse

Voor de proftaak ga ik samen met mijn vier groepsgenoten werken aan het Warehouse gedeelte van de fischertechnik fabriek.

## Personal project

# Learning outcome table with proof

## Software Development / Design

### Elevator Challenge

Deze challenge draait om het maken van designs voor een gegeven beschrijving. In mijn git kan je deze vinden in de software development map.

**Locatie:***t-db3-2223vj-tiemon-steeghs\Sofware Design\Elevator Challenge*

### Car Rental

Deze challenge gaat over het coderen in C++. Ik heb in deze opdracht kennis gemaakt met OOP in C++ en heb hierbij gecontrolleerd op memoryleaks. Deze opdracht is te vinden in de software development map.

**Locatie** *t-db3-2223vj-tiemon-steeghs\Sofware Design\Car rental challenge*

### Car Truck

De Car Truck wheels challenge gaat verder in het programmeren in C++ en brengt ook een unit-test element erbij. In de software development map kan je hem vinden.

**Locatie:** *t-db3-2223vj-tiemon-steeghs\Sofware Design\Car Truck wheel*

### States – Breadbaker

Bij de breadbaker opdracht moest ik een state machine maken. Hierbij kon ik gebruik maken van code die deels gegeven was en een grotendeels complete klassendiagram. In de states map van software development kan je de opdracht vinden.

**Locatie:** *t-db3-2223vj-tiemon-steeghs\Sofware Design\States*

### Algoritmiek – Part 1

Bij het eerste deel van algoritmiek heb ik gewerkt met sorteer algoritmes en het toepassen ervan op data bestanden. De opdracht is te vinden in mijn git in de algoritmiek map.

**Locatie:** *t-db3-2223vj-tiemon-steeghs\Sofware Design\Algoritmiek*

|  |  |  |  |
| --- | --- | --- | --- |
| **Learning outcome** | **Description** | **Proof** | **Rating** |
| **Software Development** |  |  | **Docent / Zelf** |
| analyse-SD-BA | You analyse the required functionality for a software/embedded system. You analyse the general performance aspects for a software system. You analyse existing components that can contribute to your solution. | Elevator challenge, Car rental, Car truck, States, Algoritmiek Part 1 | Orienting / Beginning |
| advice -SD-BA | You advise your stakeholder about impact of your design alternatives. | Algoritmiek part 1 |  |
| design -SD-BA | You create a design for a software system that meets the following design requirements: o OS and hardware independence o Modularity o Testability o Performance o Memory impact |  |  |
| realisation -SD-BA | You create an implementation that fully adheres to the design. You test your implementation using unit and integration tests. You implement your solution according to given quality and coding standards using defensive programming practices. |  |  |
| manage and control -SD-BA | You setup and use version control. You use a test framework |  |  |

## Communications

### Hardware UART

Voor de hardware UART opdracht heb ik UART leren toepassen met gebruik te maken van de hardware component die UART support. Hierbij heb ik handmatig naar registers geschreven en geen gebruik gemaakt van libraries. Deze opdracht kan je vinden in de communicatie map in mijn git.

**Locatie:** *t-db3-2223vj-tiemon-steeghs\Communicatie*\*Redboard uart challenge*

### Software UART

Bij de software UART challenge heb ik geleerd een UART communicatie op aan de hand van software. Hierbij heb ik dus geen gebruik gemaakt van al bestaande hardware onderdelen die UART ondersteunen. Deze opdracht kan je vinden in de communicatie map in git.

**Locatie:** *t-db3-2223vj-tiemon-steeghs\Communicatie*\*Software uart challenge*

### IOT Protocols

De IOT protocols opdracht heeft laten kennis maken met verschillende IOT protocollen. Ik heb voor deze opdracht een complexe situatie bedacht en vervolgens gekeken welke van de onderzochte IOT protocollen het best bij de bedachte situatie past. Deze opdracht vind je ook in de communicatie map van mijn git.

**Locatie:** *t-db3-2223vj-tiemon-steeghs\Communicatie*\*Communication Protocols for IOT*

|  |  |  |  |
| --- | --- | --- | --- |
| **Learning outcome** | **Description** | **Proof** | **Rating** |
| **Communications** |  |  | **Docent / Zelf** |
| analyse-COM-BA | You perform an analysis yielding suitable communication for the problem domain and justify your choices. | Hardware UART, Software UART en IOT protocols | Beginning / Beginning |
| advice-COM-BA | You advise on protocol choices, taking into account resources, performance and security of the component communication. | IOT Protocols | Beginning / Beginning |
| design-COM-BA | You design communication among multiple components based on an analysis. You elaborate on the chosen synchronisation mechanism if multithreading or multiprocessing is used. | Hardware UART, Software UART en IOT protocols | Beginning / Beginning |
| Realisation-COM-BA | You apply multithreading or multiprocessing and related synchronisation mechanisms. You implement serial communication that fully adheres to the design. You implement a solution for IoT communication application level protocol that fully adheres to the design. | Hardware UART, Software UART en IOT protocols | Orienting / Beginning |

## Embedded systems

### GPIO Challenge

De GPIO challenge heeft gezorgd voor de eerste kennismaking met het STM32 Nucleo bord. In deze opdracht heb ik een led aangestuurd aan de hand van een knop. Hierbij was het belangrijk om interrupts toe te passen, zodat er een verschil in functionaliteit was afhankelijk van hoelang de knop was ingedrukt.

**Locatie:** *t-db3-2223vj-tiemon-steeghs\Embedded systems*\*GPIO challenge*

### Timers

Voor de timers challenge ben ik te werk gegaan met de verschillende timers die beschikbaar zijn op het STM32 Nucleo bord. Deze timers heb ik gebruikt om een PWM signaal te genereren zodat ik servo motoren kan aansturen op mijn eigen project.

**Locatie:** *t-db3-2223vj-tiemon-steeghs\Embedded systems*\*Timers challenge*

### Closed loop

De closed loop challenge is een grote opdracht. Ik heb van deze opdracht nu onder andere het timer gedeelte gemaakt waarbij ik een ultrasonische sensor uitlees. Deze opdracht kan je in mijn git vinden in het closed loop mapje.

**Locatie:** *t-db3-2223vj-tiemon-steeghs\Embedded systems*\*Closed loop challenge*

|  |  |  |  |
| --- | --- | --- | --- |
| **Learning outcome** | **Description** | **Proof** | **Rating** |
| **Embedded systems** |  |  | **Docent / Zelf** |
| Analyse-ES-BA | You analyse the required functionality for a software/embedded system. You analyse datasheets and use these to develop low level software that controls hardware components. You analyse signals and protocols with lab equipment. | GPIO Challenge, Timers en Closed loop | Orienting / Beginnning |
| Advice-ES-BA | You offer a technical advice for the hardware and software components of an embedded system. | Closed loop, Proftaak | Undefined / Orienting |
| Design-ES-BA | You design an embedded system based on the requirements. You design a suitable hardware abstraction layer for your embedded system. | GPIO Challenge, Timers en Closed loop | Orienting / Beginning |
| Realisation-ES-BA | • You create an implementation that fully adheres to the design.    • You test your implementation using unit and integration tests.  • You implement your solution according to given quality and coding standards using defensive programming practices. | GPIO Challenge, Timers en closed loop. | Orienting / Beginning |

# Evaluation and reflection on the whole semester

# Feedpulse