



Universiteit Antwerpen
| Faculteit Wetenschappen

Project Software Engineering

Metro simulator

Algemene vereisten

Tools

- **Buildsysteem (CMake, CLion IDE)**
- **Tests (Google Test)**
- **Contracten (DesignByContract.h)**
- **XML input files (TinyXML)**
- **Documentatie (Doxygen)**
- **Samenwerken (GitHub)**

Evaluatiecriteria

- **Voorbeeld evaluatie formulier (op Ansymore -> “Project”)**
- **Niet-functionele vereisten (op Ansymore -> “Project”)**
- **Kwalitatieve code: OO design, code conventions (zie Ansymore)**
- **Goed project management: Documentatie, planning**
- **Kwaliteitscontrole: automatische testen, contracten**

Voorbeeld van evaluatieformulier

▪ Te vinden op Ansymore -> "Project"

Algemeen

Samenwerking: blijven samenwerken
 Beheersing werktuigen: makefiles gtest tinyxml cmake clion doxygen fileserver github
 Commentaar:

Tests

O Nt aanwezig (= geen tests)	O Vrkrd. gebruikt (= manuele test)	O Beperkt (= nt. alles getest) - domein tests	O Voldoende (= invoer) - invoer tests - meerder fouten	O Goed (= uitvoer) - uitvoertests - scenarios	O Excellent (= testcode goed) - subklassen - code duplicatie
---------------------------------	--	--	---	--	---

Commentaar:

Contracten

O Nt aanwezig (= geen assert)	O Vrkrd. gebruikt (= geen pre-post) - niet in .h file - private/protected	O Beperkt (= minimum) - properlyInit	O Voldoende (= alleen pre) - getters	O Goed (= soms pre-post) - setters - output	O Excellent (= soms pre-post) - domeinmodel bewegingen
----------------------------------	--	--	--	--	---

Plan

O Nt aanwezig (= geen planning ingediend)	O Vrkrd. gebruikt (= te laat)	O Beperkt (= nt. alles)	O Voldoende (= alle verplicht)	O Goed (= meer) - nt. verplichte - zoals beloofd	O Excellent (= bijna alles) - graf. impressie - graphics engine
---	-------------------------------------	----------------------------	-----------------------------------	--	--

Commentaar:

Objectgericht Ontwerp

O Nt aanwezig (= geen klassen) - alles in mastodont files	O Vrkrd. gebruikt (= geen pointers) - duplicated code - 0,1 infinity principle - inefficiënte data opslag	O Beperkt (= geen encapsulatie) - direct (write) accessors - long parameter lists - gebruik van lijsten in interface
O Voldoende (= domeinmodel) - klassen modeleren het domein + klassenamen = substantieven - ADT minimaal (minstens in 1 klasse)	O Goed (= ADT) - ADT volledig (elke klasse) + functienamen = werkwoorden - functionaliteit bij data (cohesion) - correct gebruik inheritance	O Excellent (= objecten collaboratie) - hollywood principe - input/output is duidelijk afgescheiden + ook in de testcode

Metro simulator

Simulator voor tram- en metronetwerk:

- **Trams**
- **Haltes**
- **Defecten**
- **Onderhoudskosten**
- **Functionaliteit beschreven in use-cases**
 - Specificatie 1.0 (nu) en 2.0 (komt later)

Use-cases voor specificatie 1.0

Use-Case	Prioriteit
<i>1: Invoer</i>	
Use case 1.1: Trams en stations inlezen	VERPLICHT
<i>2: Uitvoer</i>	
Use case 2.1: Simpele uitvoer	VERPLICHT
<i>3: Simulatie</i>	
Use case 3.1: Rijden van trams	VERPLICHT
Use case 3.2: Automatische simulatie	BELANGRIJK

Planning

Planning

- **Specificatie 1.0**

- Belofde functionaliteit vermelden in plan 1.0
- **Systeem 1.0** demonstreren tijdens evaluatie

- **Specificatie 2.0**

- Belofde functionaliteit in plan 2.0
- **Systeem 2.0** demonstreren tijdens evaluatie
- Tijd over? => Extra functionaliteit beloven in plan 2.1
- **Systeem 2.1** demonstreren tijdens finale projectverdediging

Planning

- Zie Ansymore website -> “Tijdsschema”

Dag	Deadline
Do 9 maart	Indienen plan 1.0
Wo 22 maart, do 23 maart	Evaluatie systeem 1.0
Do 30 maart	Indienen plan 2.0
Wo 10 mei, do 11 mei	Evaluatie systeem 2.0
Do 18 mei	Indienen plan 2.1

