

Java Traffic Simulator

Dokument	Anforderungsdokument
Klassifikation	Keine
Status	Verifikation
Projektname	Java Traffic Simulator
Projektverantwortlicher	Peter Schwab (PS)
Projektteam	Mathias Winker (MW), Mischa Lehmann (ML)
Zuhanden	Rolf Gasenzer (RG)

Management summary

Der Java Traffic Simulator (JTS) ist eine Agent-basierte Mikrosimulation auf realen Verkehrsnetzen für den einfachen User. Das Projekt wird im Rahmen des Modules: Projekt 1 der Berner Fachhochschulen Abteilung Technik und Informatik realisiert.

Dokumentversion

Version	Datum	Beschreibung	Autor
0.1	7.10.2014	Initialisierung	ML
0.2	8.10.2014	Review	MW
1.0	15.10.2014	Finalisieren / Abgabe	ML

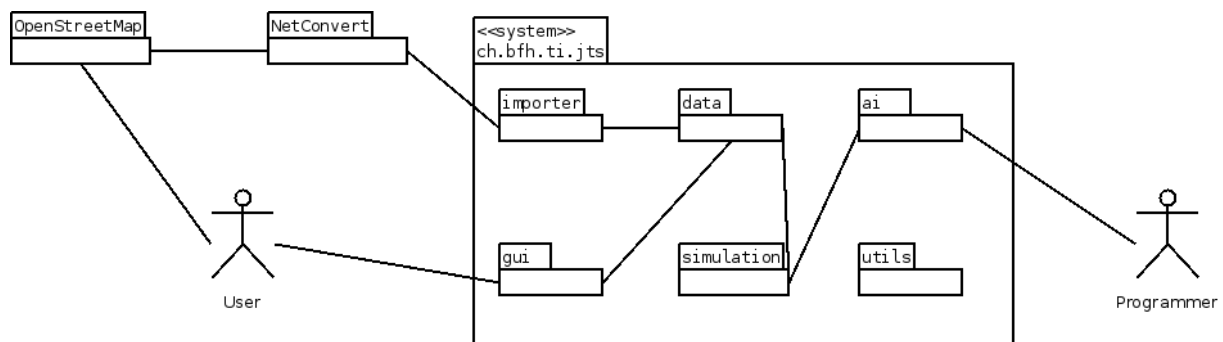
1 Index

1	Index	2
2	Ziele	3
3	Systemkontext	3
4	Scope	4
5	Anforderungen.....	4
5.1	Detailbeschreibung Anforderungen: Datenmodell	4
5.2	Detailbeschreibung Anforderungen: Simulator-Engine	4
5.3	Detailbeschreibung Anforderungen: GUI.....	5
5.4	Detailbeschreibung Anforderungen: Importer/Parser.....	5
5.5	Detailbeschreibung Anforderungen: KI.....	5
6	Verweise	6

2 Ziele

Haupt- ziel	Unter- ziel	Beschreibung
1		Ziels ist, dass nach Beendigung des Projekts von einem Anwender Stauherde erkannt werden können.
	1.1	Unterziels ist, dass dem Anwender die jeweiligen Simulationszustände in einem GUI visualisiert wird.
	1.2	Unterziel ist, dass dem Anwender Informationen zu Geschwindigkeit der Fahrzeuge über das GUI präsentiert wird.
	1.3	Unterziel ist, dass der Anwender über das GUI Verkehrsszenarien definieren kann.
2		Ziels ist, dass nach Beendigung des Projekts von einem Anwender Statistische Daten über den Verkehrsfluss pro Zeit über ein bestimmtes Verkehrsszenario erhoben werden können.
	2.1	Unterziel ist, dass der Anwender über das GUI Messpunkte setzen und konfigurieren kann.
	2.2	Unterziel ist, dass der Anwender über das GUI Daten zu den Messpunkten auslesen kann.
3		Ziels ist, dass nach Beendigung des Projekts von einem Anwender reale Strassennetze mit hilfe eines Konvertors [2] von OpenStreetMap importiert werden können.
	3.1	Unterziel ist, dass dem Anwender der Konvertor [2] zur verfügung gestellt wird.
	3.2	Unterziel ist, dass der Anwender über das GUI Strassennetze importieren kann.
4		Ziels ist, dass nach Beendigung des Projekts von einem Programmierer flexibel neue Verhaltensmuster für Fahrertypen eingeführt werden können.
	4.1	Unterziel ist, dass Programmierer eine Schnittstellt zum Abfragen von situationspezifischen Simulationsdaten für den Agent zur verfügung steht.
	4.2	Unterziel ist, dass Programmierer eine fest definierte Schnittstellt zum einfügen neuer Verhaltensmuster zur verfügung steht.
	4.3	Unterziel ist, dass Programmierer im GUI neue Verhaltensmuster erfassen können.

3 Systemkontext



4 Scope

Beschreibung	Typ
Überholen nicht über Gegenfahrbahn möglich.	Funktional
Kein Editor. Daten werden importiert von Openstreetmap mit [1] .	Interface
Datenformat für Weltdefinition gemäss [2].	Interface
Visualisierung zweckmässig, nicht Hauptfokus der Arbeit.	Funktional

5 Anforderungen

Für eine strukturierte Auflistung aller Anforderungen mit Risikobeurteilung und Status siehe Dokument: Anforderungskatalog.xlsx, Mappe: Anforderungen.

5.1 Detailbeschreibung Anforderungen: Datenmodell

Nummer	Detailbeschreibung
1	Datenmodell
1.1	Das Datenmodell ermöglicht es das Strassen wie Kanten auf einem gerichteten Graphen modelliert werden können.
1.2	Das Datenmodell ermöglicht es das Verzweigungen wie Vertices auf einem gerichteten Graphen modelliert werden können.
1.3	Das Datenmodell ermöglicht es das Mehrspurige Strassen mit mehreren Listen pro Kante auf einem gerichteten Graphen modelliert werden können.
1.4	Das Datenmodell ermöglicht es das Fahrzeuge wie Elemente der Listen pro Kante auf einem gerichteten Graphen modelliert werden können.
1.4.1	Fahrzeuge sollen simulationsrelevante Attribute wie: Position, Geschwindigkeit, Beschleunigung speichern können.
1.4.1	Attribute der Fahrzeuge sollen mit jedem Simulationsschritt verändert werden können.
1.5	Das Eingabedatenformat sieht einen gerichteten Graphen als Datengrundlage vor.

5.2 Detailbeschreibung Anforderungen: Simulator-Engine

Nummer	Detailbeschreibung
2	Simulator-Engine
2.1	Die Simulation wird für fest definierte Zeitintervalle berechnet. Von den berechneten Simulationsschritten ausgehend sollen für die Visualisierung interpolierungen gemacht werden damit eine für den Betrachter flüssig laufende Simulation entsteht.
2.2	Die Simulation soll zu jedem Zeitpunkt neu gestartet werden können und ohne Veränderung der Rahmenbedingungen zu denselben Ergebnissen führen.
2.3	Die Simulation soll so aufgebaut werden das diese mit vertikal skaliert werden kann.
2.4	Simulation bewegt die Fahrzeuge und passt ihre Eigenschaften an
2.5	Der Seed für den Pseudorandom-Generator soll vom User vorgegeben werden können. (Rahmenbedingung)
2.6	Die Grösse der Simulationsschritte soll konfigurierbar sein.
2.7	

5.3 Detailbeschreibung Anforderungen: GUI

Nummer	Detailbeschreibung
3	GUI
3.1	Die Berechnung der Simulation und die dazu gehörige Visualisierung soll jederzeit angehalten oder neu gestartet werden können.
3.2	Ein gestoppter Simulationszustand soll exportiert werden können.
3.3	Ein exportierter Simulationszustand soll wieder importiert werden und weiter simuliert werden können.
3.4	Auf der Visualisierung der Simulation sollen schematisch Strassen, Verzweigungen und Fahrzeuge erkennbar sein.
3.4.1	Strassen werden als ein Netz von Rechtecken dargestellt.
3.4.2	Fahrzeuge werden als sich bewegende Rechtecke dargestellt.
3.5	Auf einer laufenden Simulation soll der User in Helikopterperspektive navigieren können.
3.6	Die Visualisierung bietet eine Möglichkeit Openstreetmap-Daten konvertiert zum SUMO-Datenformat zu importieren.

5.4 Detailbeschreibung Anforderungen: Importer/Parser

Nummer	Detailbeschreibung
4	Importer/Parser
4.1	Weltdaten können von Openstreetmap exportiert werden
4.2	Das Datenformat für den Import sind SUMO generierte XML-Dokumente.

5.5 Detailbeschreibung Anforderungen: KI

Nummer	Detailbeschreibung
5	KI
5.1	Jeder Agent soll mit einer minimalen künstlichen Intelligenz Entscheidungen über sein Verhalten selbst treffen können.
5.2	Die KI soll über ein Interface ihrer Umgebung erfahren können.
5.2.1	Die KI alle Eigenschaften der Umgebenden Fahrzeuge (vorne, hinten, Seiten) abrufen können.
5.2.2	Die KI soll Informationen zu kommenden Verzweigungen abrufen können.
5.3	Die KI soll Entscheide treffen können die das Verhalten des Agents beeinflussen.
5.3.1	Beschleunigung/Geschwindigkeit verändern
5.3.2	Entscheidung treffen bei Verzweigungen

6 Verweise

Nr	Beschreibung
[1]	http://sumo.dlr.de/wiki/Networks/Import/OpenStreetMap
[2]	http://sumo.dlr.de/wiki/NETCONVERT