

FH - Studiengang für
Informationstechnik und System-Management
Salzburg

ITS

Software Architektur Labor - Review

Es wurde ein Review der Version 6 gemacht.

Version:	2
Datum:	13.07.2017
Autoren:	Stephanie Kaschnitz, Christopher Wieland, Martin Wieser, Andreas Lippmann, Hannes Kleiner
Zu evaluierende Gruppe:	Martin Uray, Elisabeth Birnbacher, Dieter Draxler, Fellner Tobias, Waldner Simon

Versionshistorie

Version	Datum	Änderungen
1	25.06.17	Erstellen der Struktur und Bewertungskriterien
2	10.07.17	Ergebnisse des Reviews

Inhaltsverzeichnis

1	Zweck des Architektur-Reviews	4
1.1	Aufgabenstellung	4
2	Bewertungskriterien und Ziele des Reviews.....	4
2.1	Bewertungskriterien - Qualitätsziele	4
2.2	Bewertungskriterien – Funktionale Anforderungen	5
3	Ergebnisse und Verbesserungsvorschläge	7
3.1	Ergebnisse.....	7
3.2	Verbesserungsvorschläge	7
3.3	Sonstiges	8
4	Aktionsplan	Fehler! Textmarke nicht definiert.

Tabellenverzeichnis

Tabelle 1: Qualitätsziele.....	4
Tabelle 2: Bewertungskriterien	7
Tabelle 3: Verbesserungsvorschläge	8

1 Zweck des Architektur-Reviews

Zweck dieses Dokuments ist die Erstellung eines Architekturreviews im Rahmen des Software Architektur Labors.

1.1 Aufgabenstellung

Ein Teil der im Rahmen des Labors erstellten Verkehrssimulation ist die Durchführung eines Reviews. Dabei werden zwischen den Gruppen die jeweilige Architekturdokumentation sowie die Projektmappe mit der zugehörigen Visual Studio Solutions ausgetauscht. Nun muss diese auf Grundlage der definierten Anforderungen und Ziele an das Projekt geprüft und bewertet werden.

Bereitgestellte Dokumente: Dokumentation der Softwarearchitektur, Source Code des Projekts (Projektmappe mit Verkehrssimulation und Ampelsteuerung)

2 Bewertungskriterien und Ziele des Reviews

Die folgenden Bewertungskriterien wurden von der zu evaluierenden Gruppe festgelegt und sind im Appendix des Dokuments angehängt.

2.1 Bewertungskriterien - Qualitätsziele

Tabelle 1 zeigt die zu bewertenden Qualitätsziele der Verkehrssimulation. Diese werden nun analysiert und gegen die vorliegende Architektur geprüft. Dabei wird auf jedes Kriterium eingegangen und die Auswirkungen der darauf basierenden Architekturentscheidung bewertet. Anhang A liefert eine genauere Beschreibung der Kriterien.

Verweisnummer #	Kriterium	Bewertung
1	Wartbarkeit	
2	Bedienbarkeit	
3	Verfügbarkeit	
4	Verständlichkeit	

Tabelle 1: Qualitätsziele

2.2 Bewertungskriterien – Funktionale Anforderungen

Die folgende Tabelle 2 beinhaltet die festgelegten funktionalen Anforderungen an das Projekt. Diese werden nun mit der umgesetzten Architektur verglichen. Anhang B liefert eine genauere Beschreibung der Kriterien.

No	Kriterium	Bewertung	Erfüllungsgrad
1	Verteiltes System	<p>Verkehrssimulation und Ampelsteuerung sind in diesem Fall getrennte Komponenten, die über ein Interface (IPC) kommunizieren. Öffnet man das Projekt mit Visual Studio findet man zwei Visual Studio Projektmappen: LightControlUnit.sln und TrafficSim.sln.</p> <p>Beide Projekte können gleichzeitig und unabhängig von Fremdsystemen ausgeführt werden.</p>	Die Anforderung wurden vollständig erfüllt.
2.1	Komponentenbasiert: „Traffic Simulation“	<p>Wie die bereitgestellte Architekturdokumentation zeigt, gibt es in der Komponente „Traffic Simulation“ ein zentrales Modul: den Simulator (TrafficSim), der gleich mehrere Aufgaben übernimmt (Simulation & Darstellung der Fahrzeuge). Um komponentenbasiert zu Entwickeln wäre hier die Trennung von Logik und Darstellung hilfreich. Eine Entkopplung der GUI von der logischen Komponente hätte zusätzlich den Vorteil, dass die Benutzerschnittstelle jederzeit angepasst oder ausgetauscht werden könnte. Die übrigen „Traffic Simulation“ – Komponenten wurden sinnvoll gewählt und bieten soweit es die Aufgabenstellung erlaubt eine große Austauschbarkeit/Parametrierbarkeit.</p>	Die Anforderung wurde teilweise erfüllt.
2.2	Komponentenbasiert: „Traffic Light Controller“	<p>Die TrafficLightControlServer Komponente stellt über eine Konsolenanwendung ein Menü zur Verfügung, über den die Ampelschaltungen der</p>	Die Anforderung wurden vollständig erfüllt.

		Verkehrssimulation konfiguriert werden können.	
3	Kreuzungen	Die Simulation beinhaltet mehrere geregelte und eine ungeregelte Kreuzung. Die geregelte Kreuzung sind durch Ampeln geschaltet und von den Verkehrsteilnehmern beachtet.	Die Anforderung wurden vollständig erfüllt.
4	Verkehrsregeln	Die ungeregelte Kreuzung wurde implementiert jedoch keine Verkehrsregeln dabei festgelegt (z.B. Rechts-vor-links). Die Verkehrsregeln werden ebenfalls von der zentralen Steuerungskomponente „Traffic Sim“ umgesetzt und somit nicht ohne weiteres auszutauschen.	Die Anforderung wurden nicht erfüllt.
5	Parametrisierbarkeit der Verkehrssimulation	In diesem Fall wurde das Kriterium vollends erfüllt. Über eine eigene Komponente ist es dem Benutzer möglich, die Geschwindigkeit, Anteil an LWs, Anzahl an einfahrenden Fahrzeugen und das einfahren von Fahrzeugen einer anderen Gruppe zu steuern.	Die Anforderung wurde mehr als erfüllt.
6	Parametrisierbarkeit der Ampelsteuerung	Wie in Punkt 2.2 beschrieben, erweist sich die Parametrisierung der Ampelsteuerung als sehr umständlich durch Bedienung eines Menüs über die Konsole.	Die Anforderung wurde teilweise erfüllt.
7	Hindernisse	Über die Komponente Obstacles ist es möglich Hindernisse in der Verkehrssimulation zu verteilen. Sie erlaubt es dem Benutzer ein Hindernis auszuwählen, dieses im Straßennetz zu platzieren und bei Bedarf wieder zu entfernen.	Die Anforderung wurde mehr als erfüllt.
8	Vernetzung mit weiteren Simulationen anderer Gruppen	Auch hier wurden die Anforderungen erfüllt, indem eine Schnittstelle zum zentralen RabbitMQ-Server implementiert wurde.	Die Anforderung wurde mehr als erfüllt.

Opt	Optionale Regelungen (siehe Anhang A)	Das Fahrverhalten kann nicht parametrisiert werden.	Die Anforderung wurde teilweise erfüllt.
-----	--	---	--

Tabelle 2: Bewertungskriterien

3 Ergebnisse und Verbesserungsvorschläge

Im diesem Kapitel werden die Ergebnisse noch einmal wiedergegeben und etwaige Verbesserungsvorschläge eingebracht. Zudem wird auf die Gründe hinter den architekturbeeinflussenden Entscheidungen eingegangen.

3.1 Ergebnisse

Komponentenbasierte Architektur:

- Auf die Aufteilung der hier vorliegenden Architektur wurde bereits im Kapitel 2.2 eingegangen. Hierbei hat sich gezeigt, dass der Fokus auf einer komponentenbasierten Entwicklung des Systems lag, allerdings durch den Einsatz des Frameworks „Mono-Game“, sich Simulation und Darstellung des Projekts auf eine zentrale Komponente konzentriert hat. Dies hat eine große Abhängigkeit von dieser Komponente zur Folge. Die Entscheidung für das „Mono-Game“ wurde zusätzlich noch in der Architekturdokumentation begründet und die verschiedenen Vor- und Nachteile genannt. Es handelt sich hierbei um einen Trade-off zwischen dem Gaming Framework, dass viele Funktionen und eine attraktive Benutzeroberfläche zur Verfügung stellt und andererseits einer vorgegebenen Struktur, die die Flexibilität einschränkt.

Zusammenhängendes Verkehrsnetz:

- Nach Starten der Anwendung erscheint ein Fenster, dass die Verkehrssimulation zeigt. Alle Straßen sind hierbei zusammenhängend, wobei es einen zentralen Ein- und Ausfahrtpunkt für alle Fahrzeuge gibt. Hier wurde durch die Komponente „Road System“ sichergestellt, dass sich diese leicht austauschen lässt was bei einer eventuellen Neugestaltung des „Straßennetzes“ von Vorteil ist.

3.2 Verbesserungsvorschläge

Zusammenhängendes Verkehrsnetz:

- Eine mögliche Erweiterung ist ein Verkehrsnetz mit mehreren Ein-/Ausfahrtpunkten.

3.3 Sonstiges

Seite	Objekt	Anmerkung
Seite 7 – Abbildung 3	Lollipop zwischen Traffic Simulation und Traffic Light Controller	Falsch herum gesetzt
Seite 8 – Abbildung 4	Verbindungen zwischen den einzelnen Komponenten	Um eine einheitliche Architektur zu erreichen, sollten hier auch die Lollipopnotation verwendet werden

Tabelle 3: Verbesserungsvorschläge

Anhang A - Funktionale Anforderungen an die Verkehrssimulation

Nr.	Anforderung	Beschreibung
M-1	Verteilt	Das System muss verteilt implementiert werden, d.h. die Verkehrssimulation an sich und die Lichtsteuerungsanlage werden jeweils als eigener Prozess implementiert.
M-2	Komponentenbasiert	Das System soll sich aus Komponenten zusammensetzen, die jeweils die einzelnen Funktionalitäten des Systems beinhalten.
M-3	Kreuzungen	Die Verkehrssimulation soll ein Straßennetz besitzen, das sowohl geregelte als auch ungeregelte Kreuzungen beinhaltet.
M-4	Verkehrsregeln	Die Verkehrssimulation muss sicherstellen, dass sich alle Verkehrsteilnehmer an die Straßenverkehrsordnung halten, d.h. an über Ampeln geregelten Kreuzungen folgen die Fahrzeuge den Anweisungen der Ampel, bei ungeregelten Kreuzungen beachten sie Rechts-Vor-Links.
M-5	Parametrisierbarkeit - Verkehrssimulation	Die Verkehrssimulation soll es dem Benutzer ermöglichen die Anzahl der in die Simulation einfahrenden Fahrzeuge zu steuern und die Eigenschaften der Verkehrsteilnehmer (max. Geschwindigkeit, LKW Anteil) einstellen zu können.
M-6	Parametrisierbarkeit - Lichtsteuerungsanlage	Die Lichtsteueranlage soll es dem Benutzer ermöglichen die Konfiguration der Ampelanlagen in der Verkehrssimulation zu ändern.
M-7	Hindernisse	Die Verkehrssimulation soll es dem Benutzer ermöglichen Hindernisse im Straßennetz zu platzieren, auf die die Fahrzeuge in der Simulation reagieren müssen.
M-8	Vernetzung mit weiterer Simulationen anderer Gruppen	Die Verkehrssimulation soll es dem Benutzer ermöglichen Fahrzeuge in die Verkehrssimulation einer anderen Projektgruppe zu schicken.
O-1	Optionale Regelungen	<ul style="list-style-type: none"> • Das Verhalten der Verkehrsteilnehmer sollte parametrierbar sein: aggressiv, schnell, langsam... • Die maximale Anzahl an Fahrzeugen die in die Simulation generiert werden, sollte einstellbar sein (ein Überschreiten durch Einfahren von Fahrzeugen anderer Simulationen ist möglich)

Anhang B - Qualitätsziele des Projekts

Nr.	Qualitätsziel	Beschreibung
Q-1	Bedienbarkeit	Die grafische Benutzeroberfläche ist benutzerfreundlich und kann intuitiv bedient werden.
Q-2	Verfügbarkeit	Die Simulation erfüllt ihre Funktion fehlerfrei.
Q-3	Verständlichkeit	Es werden ausschließlich aussagekräftige Variablen- und Funktionsnamen im Quellcode verwendet. Alle Funktionen werden kommentiert. Dabei gilt es zu beachten, dass die Kommentare Doxygenkonform sind, damit eine automatische Generierung der Dokumentation möglich ist.
Q-4	Modifizierbarkeit	Die Verkehrssimulation soll erweiterbar sein, sodass zusätzliche Aspekte des Straßenverkehrs berücksichtigt werden können (z.B. neue Straßentypen wie eine Einbahnstraße oder ein Kreisverkehr).