

Sitzungsprotokolle

Marvin Janosch

10. November 2017

Inhaltsverzeichnis

1	Kundeninterview - 2017-10-18	3
2	Gruppentreffen - 2017-10-24	4
3	Treffen mit Auftraggeber - 2017-10-24	5
4	Treffen mit Auftraggeber - 2017-11-03	6
5	Treffen mit Auftraggeber - 2017-11-07	7

1 Kundeninterview - Beschreibung der Anforderungen

1. Datum: 2017-10-18

1.1. Beginn: 15:00 Uhr

1.2. Ende: 15:44 Uhr

2. Teilnehmer:

Jan Amann, Marvin Janosch, Dennis Szczepanski, Jan Philip Wahle

3. Protokollant: Jan Philip Wahle

Vorbesprechung zur Projektplanung und Informationssammlung zur Erstellung des Lastenhefts

- Codeverwaltungssystem einheitlich auswählen: git, github. (Khalid)
- Synchronisation aller Sitzungs- sowie Zeitprotokolle.
- Frontend soll folgendes beinhalten:
 - Map Editor:
 - * Graphische Oberfläche zum Erstellen von Strecken
 - * Speicher- und Ladefunktion
 - * Statische Objekte und Hindernisse
 - * Dynamische Objekte und Hindernisse
 - 2D Simulation
 - 3D Visualisierung
- Backend soll folgendes beinhalten:
 - Sensorik des Fahrzeugs
 - Physikalische Trägheit
 - Selbstlernendes Netzwerk (Tensorflow)

2 Gruppentreffen

1. Datum: 2017-10-24

1.1. Beginn: 14:00 Uhr

1.2. Ende: 14:45 Uhr

2. Teilnehmer:

Jan Amann, Khalid Bellouch, Marvin Janosch, Dennis Szczepanski,
Jan Philip Wahle

3. Protokollant: Dennis Szczepanski

- Anlegen einer Codefibel bis zum 01.11.2017
- Überarbeiten des Pflichtenhefts und Glossars bis zum 01.11.2017
- Diskussion über Grundgerüst der Strecke
 - Strecke als mehrdimensionales Grid
 - Statische Objekte → Umfahren
 - Dynamische Objekte → Bremsen
 - Aspekte für Fitness definieren: Zeit ohne Kollision, passierte Teilstrecken
 - Vorerst nur ein Fahrzeugtyp
 - Wie werden Strecken gespeichert? XML? JSON?

3 Treffen mit Auftraggeber

1. Datum: 2017-10-24

1.1. Beginn: 15:05 Uhr

1.2. Ende: 15:35 Uhr

2. Teilnehmer:

Jan Amann, Khalid Bellouch, Marvin Janosch, Dennis Szczepanski,
Jan Philip Wahle

3. Protokollant: Dennis Szczepanski

- Erster Einblick in Lasten- und Pflichtenheft sowie Klärung von Fragen zu:
 - Einsatzgebiet der Software
 - Produktübersicht
- Besprechung der Gedanken aus Gruppentreffen
 - Raster als Strecke → Hindernisse sollten als Fließkommawert gespeichert werden, nicht als weiteres Raster
 - Streckenspeicherung → JSON praktisch
 - Ladereihenfolge von Objekten
 - Beispielhafte Produktleistungen → 64x64 Map, mit 100 Objekten

4 Treffen mit Auftraggeber

1. Datum: 2017-11-03
 - 1.1. Beginn: 11:00 Uhr
 - 1.2. Ende: 11:30 Uhr
2. Teilnehmer:
Jan Amann, Khalid Bellouch, Dennis Szczepanski
3. Protokollant: Dennis Szczepanski

- Einsicht in Pflichtenheft
- Einsicht in Klassendiagramm
 - Polygone oder Splines als Begrenzung → Splines empfohlen, da diese das Arbeiten vereinfachen. Begrenzung als zwei mal stetig differenzierbare Funktion, bei einfachen Kurven nur testen, ob Fahrzeug sich zwischen den Funktionen befinden
 - * Problem → Kreuzungen
 - Editor und Simulation nutzen QWindow → QT als Paket assoziieren, und QWindow explizit nennen
- Besprechen von kommender Abgabe
 - Benutzerhandbuch
 - * Beschreibungen aller Menüs und Funktionen
 - Zusätzlich dynamische Diagramme abgeben
 - * Use-Case Diagramm
 - * Sequenzdiagramm

5 Treffen mit Auftraggeber

1. Datum: 2017-11-07

1.1. Beginn: 15:00 Uhr

1.2. Ende: 15:30 Uhr

2. Teilnehmer:

Jan Amann, Khalid Bellouch, Jan Philip Wahle

3. Protokollant: Jan Philip Wahle

- Einsicht in Benutzerhandbuch
- Einsicht in Use-Case-Diagramm/Sequenzdiagramm
 - Weiteres Sequenzdiagramm für das Laden einer Map in die Simulation und die darauffolgende Simulation
- Ziele für kommende Besprechung
 - Hauptaugenmerk auf Schnittstellendefinition
 - * Einheitlichen Standard für das Speichern von Maps erstellen (XML) und eine zu simulierende Map bauen (hardcoded erlaubt)
 - * Eine Simulation erstellen, die auch ohne Visualisierung läuft (wichtig für späteres Training)
 - Aufteilung der Gruppen in
 - * Visualisierung des Editors
 - * Visualisierung der Simulation
 - * Simulation (reine Simulation also nur backend)