
Requirements and Design

PR Multimediale Systeme

SS 2016 – Gruppe [MZ7]

Document Control

Contributors

Person	Role	Company	Contribution
Bernhard Rieder	Developer	UNIVIE	Requirements and Design
Michael Pichler	Developer	UNIVIE	Requirements and Design

Revision History

Issue	Author	Date	Description
V0.1	Bernhard Rieder	15/03/2016	Created document draft
V0.2	Bernhard Rieder	04/04/2016	Application Overview and Objectives draft
V0.3	Bernhard Rieder	04/04/2016	Added Roles and Actors and Use Case draft
V0.3	Michael Pichler	04/04/2016	Added Component-Diagram
V1.0	Bernhard Rieder, Michael Pichler	05/04/2016	1 st Use Case 1 st Architecture 1 st Project Management
V1.1	Michael Pichler	05/04/2016	2 nd Project Management
V1.2	Michael Pichler	06/04/2016	2 nd New Use Case diagram and description
V1.3	Michael Pichler	06/04/2016	3 rd Use Case description
V1.4	Michael Pichler	06/04/2016	2 nd Architecture for Component [XY]
V1.5	Bernhard Rieder	06/04/2016	Updated UML class diagrams
V1.6	Bernhard Rieder	07/04/2016	Updated UML class diagrams
V1.7	Bernhard Rieder	07/04/2016	Created User Interface paragraphs
V2.0	Michael Pichler	08/04/2016	1 st Final Version – Corrections and improvements
V2.1	Bernhard Rieder	08/04/2016	2 nd Final Version – Corrections and improvements

Table of Contents

1	Application Overview and Objectives.....	3
1.1	Application Overview	3
1.2	Objectives	3
2	Use Cases.....	4
2.1	Roles and Actors.....	4
2.2	Use Case Diagrams.....	4
2.3	Textual Use Case Descriptions	4
3	Architecture	9
3.1	Overview	9
3.2	Architecture for Component <i>GazeTracker</i>	10
3.3	Architecture for Component <i>Application</i>	13
4	Project Management.....	16
4.1	Milestones and Schedules.....	16
4.2	Planned Effort per Person	16

1 Application Overview and Objectives

1.1 Application Overview

Die Verwendbarkeit einer Software und das damit verbundene Design des User Interface sind ein wesentlicher Bestandteil von erfolgreicher Software. Die Evaluierung der Benutzerfreundlichkeit ist meist ein schwieriges Unterfangen, bei der man sich mit TesterInnen über die Verwendung der Software austauschen muss. Um die Evaluierung zu erleichtern gibt es zudem teure Eye-Tracking Geräte, die sich jedoch nicht jede EntwicklerIn leisten kann.

Aus diesem Grund wird in diesem Projekt ein sogenannter "Gaze Tracker" entwickelt, welcher nur mit Hilfe einer Webcam funktioniert und somit eine kostengünstige Alternative zum teuren Gerät bietet. Der GazeTracker zeichnet Punkte der Blickrichtung in Bezug auf den Bildschirm auf. Mittels Aufzeichnung dieser Punkte kann die Evaluierung vorangetrieben werden.

Weiteres leitet sich damit das nächste mögliche Einsatzgebiet eines Gaze Trackers ab. Mithilfe der berechneten Punkte kann eine kontaktlose Steuerung von Software implementiert werden. Diese könnte zum Beispiel zur kontaktlosen Verwendung einer Bildschirmtastatur genutzt werden.

In diesem Projekt wird der Tacker jedoch nur als Analysetool für statische Elemente verwendet. Um den GazeTracker im weiteren Sinne für dynamische Elemente verwenden zu können und eine erweiterte Analyse von Daten zu schaffen, könnte man diesen zum Beispiel durch ein externes Screen Capturing Tool erweitern.

1.2 Objectives

Funktionale Anforderungen:

- Eye-Tracking mittels Webcam
- Blickrichtung der Augen in Bezug auf Bildschirm berechnen
- Aufzeichnung der Blickrichtung Auftrittspunkte

Nicht-Funktionale Anforderungen:

- Benutzerfreundlichkeit
- Wiederverwendbarkeit
- Performance

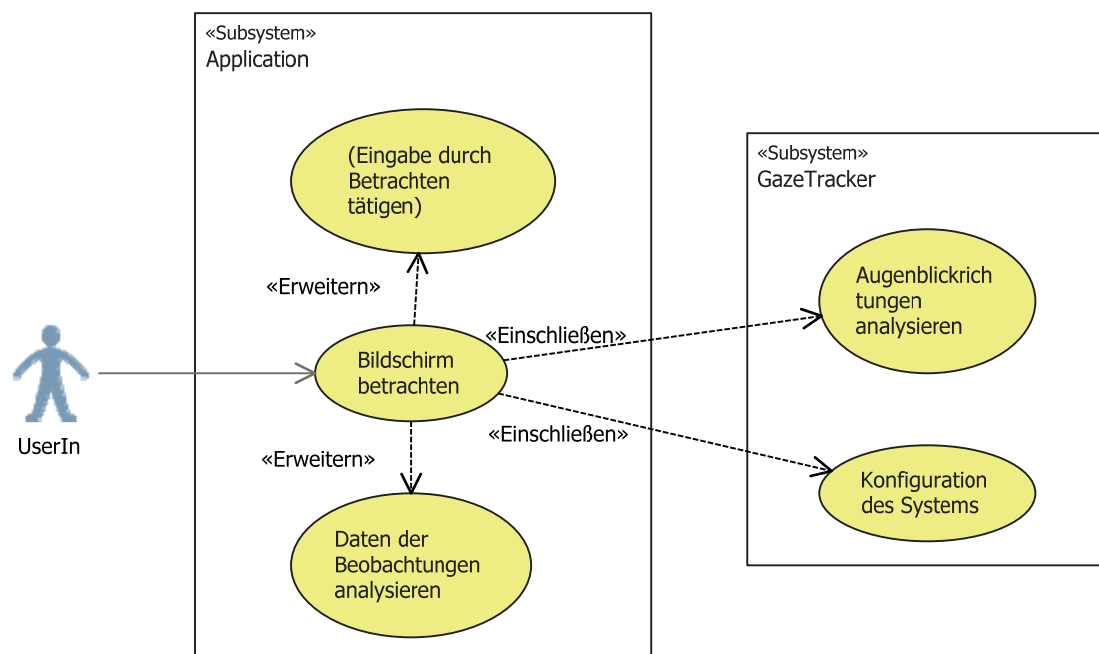
2 Use Cases

2.1 Roles and Actors

2.1.1 UserIn

Es gibt in diesem Projekt nur eine Akteurln, nämlich die UserIn. Diese muss lediglich das Programm starten und ihre Tätigkeiten durchführen, die sie möchte. Diese kann wie gewohnt durch physikalische Eingabegeräte erfolgen, aber auch einfach durch Betrachten des Bildschirms. In diesem Projekt rückt hauptsächlich die Analyse der UserIn in den Vordergrund. Dazu wird auf den GazeTracker zugegriffen, der die notwendigen Informationen bereitstellt. Geachtet werden muss, dass die UserIn eine Webcam besitzt und sie für ausreichende Lichtverhältnisse sorgt. Ansonsten kann der GazeTracker nicht verwendet oder das Gesicht/Auge kann nicht detektiert werden.

2.2 Use Case Diagrams



Der Punkt "Eingabe durch Betrachten tätigen" wurde aufgrund der Vollständigkeit hinzugefügt, ist aber für den weiteren Verlauf des Projektes weniger relevant.

2.3 Textual Use Case Descriptions

Category	Entry
Use Case ID	1
Title	Bildschirm betrachten
Scope	GazeTracker-System
Actors	UserIn
Preconditions	<ul style="list-style-type: none">UserIn hat eine Webcam angeschlossen

	<ul style="list-style-type: none"> <i>UserIn hat für ausreichende Lichtverhältnisse gesorgt</i> 	
Success Guarantees	<i>Das Programm kann die Augen analysieren und Daten für die weiteren Verarbeitungsschritte zur Verfügung stellen</i>	
Trigger	<i>Durch Bewegung der Augen</i>	
Main Success Scenario	<ol style="list-style-type: none"> <i>GazeTracker wird gestartet</i> <i>Konfigurationen werden von UserIn ausgeführt</i> <i>GazeTracker erkennt die Augen</i> <i>Software kann die Augenbewegungen analysieren</i> 	
Exceptions	1	Auf Webcam kann nicht zugegriffen werden <ul style="list-style-type: none"> <i>UserIn wird darüber informiert</i>
	2	Lichtverhältnis sind nicht ausreichend <ol style="list-style-type: none"> <i>Lichtanpassungen werden durchgeführt</i> <i>Fehlschlag -> UserIn wird informiert</i>
	3	Auge/Gesicht wird nicht erkannt <ul style="list-style-type: none"> <i>UserIn wird aufgefordert, die richtige Position einzunehmen</i>
Version	<i>0.2</i>	
Last updated	<i>06-April-2016</i>	
Issues/Comments	<i>UserIn kann mit den Daten die Betrachtungspunkte analysieren oder je nach Wunsch den Ablauf der Software mit den Augen steuern.</i>	

Category	Entry	
Use Case ID	2	
Title	<i>Daten der Beobachtungen analysieren</i>	
Scope	<i>GazeTracker-System</i>	
Actors	<i>UserIn</i>	
Preconditions	<ul style="list-style-type: none"> <i>UserIn hat Webcam eingeschalten</i> <i>System kann die Bewegungen der Augen erkennen</i> <i>GazeTracker wurde richtig konfiguriert</i> 	
Success Guarantees	<i>Daten werden aufbereitet und das Blickverhalten analysiert</i>	
Trigger	<i>Durch Bewegung der Augen und aufzeichnen der Daten</i>	
Main Success Scenario	<ol style="list-style-type: none"> <i>GazeTracker wird gestartet</i> <i>Konfigurationen werden von UserIn ausgeführt</i> <i>GazeTracker erkennt die Augen</i> <i>Auswerten der gespeicherten Daten</i> 	
Exceptions	1	Auf Webcam kann nicht zugegriffen werden <ul style="list-style-type: none"> <i>UserIn wird darüber informiert</i>

	2	Lichtverhältnisse sind nicht ausreichend. <ol style="list-style-type: none"> 1. <i>Lichtanpassungen werden durchgeführt</i> 2. <i>Fehlschlag -> UserIn wird informiert</i>
	3	Auge/Gesicht wird nicht erkannt <ul style="list-style-type: none"> • <i>UserIn wird aufgefordert, die richtige Position einzunehmen</i>
Version	0.2	
Last updated	06-April-2016	
Issues/Comments	Eine Option kann sein, dass das Ergebnis der Analyse gespeichert werden kann.	

Category	Entry	
Use Case ID	3	
Title	Konfiguration des Systems	
Scope	GazeTracker-System	
Actors	-	
Preconditions	<ul style="list-style-type: none"> • <i>UserIn hat eine Webcam angeschlossen</i> • <i>System kann auf den GazeTracker zugreifen</i> 	
Success Guarantees	<i>Es werden Einstellungen durchgeführt, für was der GazeTracker verwendet wird und es wird der Bildschirm kalibriert.</i>	
Trigger	<i>Applikation wird gestartet und der GazeTracker benutzt.</i>	
Main Success Scenario	<ol style="list-style-type: none"> 1. <i>Webcam funktioniert einwandfrei</i> 2. <i>Auf den GazeTracker wird zugegriffen</i> 3. <i>Einstellungen können durchgeführt werden</i> 	
Exceptions	1	Auf Webcam kann nicht zugegriffen werden <ul style="list-style-type: none"> • <i>UserIn wird darüber informiert</i>
	2	Konfiguration des Bildschirms schlägt fehl <ol style="list-style-type: none"> 1. <i>Error Code wird zur Verfügung gestellt</i>
Version	0.2	
Last updated	06-April-2016	
Issues/Comments	<i>UserIn muss bei der Bildschirmkonfiguration die verschiedenen Eckpunkte und auf den Mittelpunkt blicken, um die Bildschirmgröße und Augenbewegungen zu ermitteln.</i>	

Category	Entry	
Use Case ID	4	
Title	Augenblickrichtungen analysieren	
Scope	GazeTracker-System	

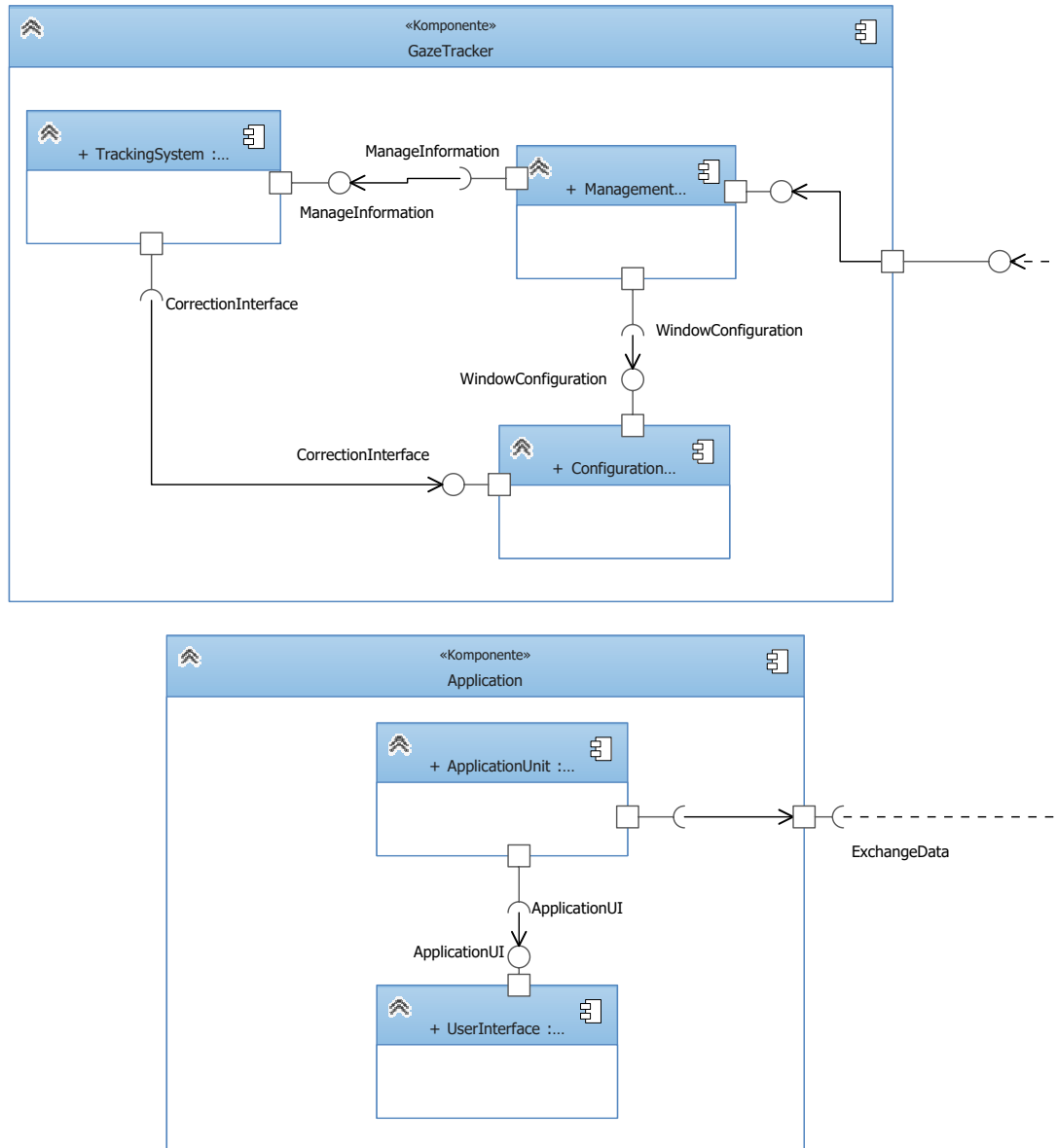
Actors	-
Preconditions	<ul style="list-style-type: none"> • <i>UserIn hat eine Webcam angeschlossen</i> • <i>System kann den GazeTracker benutzen</i> • <i>UserIn hat den GazeTracker konfiguriert</i>
Success Guarantees	<i>Die Berechnung der Daten der Blickrichtungen und Auftritt Punkt auf dem Bildschirm wird berechnet.</i>
Trigger	<i>Applikation wird gestartet und System wurde konfiguriert</i>
Main Success Scenario	<ol style="list-style-type: none"> 1. <i>Webcam funktioniert einwandfrei</i> 2. <i>Auf den GazeTracker wird zugegriffen</i> 3. <i>Einstellungen wurden durchgeführt</i> 4. <i>Webcam erkennt Auge/Gesicht und berechnet den Blickpunkt</i>
Exceptions	1 Auf Webcam kann nicht zugegriffen werden <ul style="list-style-type: none"> • <i>UserIn wird darüber informiert</i>
	2 Lichtverhältnis sind nicht ausreichend <ol style="list-style-type: none"> 1. <i>Lichtanpassungen werden durchgeführt</i> 2. <i>Fehlschlag -> UserIn wird informiert</i>
	3 Auge/Gesicht wird nicht erkannt <ul style="list-style-type: none"> • <i>UserIn wird aufgefordert, die richtige Position einzunehmen</i>
Version	0.2
Last updated	06-April-2016
Issues/Comments	<i>Die Position auf dem Bildschirm wird ermittelt und als Daten mit einem Zeitstempel zur Verfügung gestellt.</i>

Category	Entry
Use Case ID	5
Title	<i>Eingabe durch Betrachtungen tätigen</i>
Scope	<i>GazeTracker-System</i>
Actors	<i>UserIn</i>
Preconditions	<ul style="list-style-type: none"> • <i>UserIn hat Webcam eingeschalten</i> • <i>System kann die Bewegungen der Augen erkennen</i> • <i>GazeTracker wurde richtig konfiguriert</i>
Success Guarantees	<i>Daten werden aufbereitet und können für die Eingabe verwendet werden</i>
Trigger	<i>Durch Bewegung der Augen auf bestimmte Punkte</i>

Main Success Scenario	<ol style="list-style-type: none"> 1. <i>GazeTracker</i> wird gestartet 2. Konfigurationen werden von UserIn ausgeführt 3. <i>GazeTracker</i> erkennt die Augen 4. Auswerten der gespeicherten Daten 	
Exceptions	1	Auf Webcam kann nicht zugegriffen werden <ul style="list-style-type: none"> • UserIn wird darüber informiert
	2	Lichtverhältnis sind nicht ausreichend. <ol style="list-style-type: none"> 1. <i>Lichtanpassungen</i> werden durchgeführt 2. <i>Fehlschlag</i> -> UserIn wird informiert
	3	Auge/Gesicht wird nicht erkannt <ul style="list-style-type: none"> • UserIn wird aufgefordert, die richtige Position einzunehmen
Version	0.3	
Last updated	06-April-2016	
Issues/Comments	Durch die Eingabe durch Betrachten des Bildschirmes können zum Beispiel Objekte bewegt werden oder virtuelle Events ausgelöst werden	

3 Architecture

3.1 Overview



Dass System besitzt im Wesentlichen zwei Hauptkomponenten, welche sich zu mehreren Unterkomponenten unterteilen lassen. Eine wichtige Komponente stellt der GazeTracker dar und stellt somit das Herzstück dar. Diese Komponente stellt die wichtigen Informationen zur Verfügung. Jede Applikation, die den GazeTracker verwenden möchte, kann durch eine Schnittstelle darauf zugreifen und für seine Zwecke verwenden. Hier läuft das Programm, welches gesteuert oder analysiert werden soll.

3.2 Architecture for Component *GazeTracker*

3.2.1.1 Design Considerations

GazeTracker ist die erste Hauptkomponente des Projektes, die in diesem Projekt implementiert werden muss. Sie ist Verantwortlich dafür, dass im ersten Schritt das Gesicht/Auge erkannt wird und im folgenden Schritt die Erkennung des Blickes auf den Bildschirm. Diese Daten können mit Hilfe einer Kommunikationsschnittstelle an anderen Komponenten weitergeleitet werden. In diesem Projekt handelt es sich dabei um die eigene Applikation.

TrackingSystem

Sie ist die wichtigste Komponente im GazeTracker System. TrackingSystem hat die Aufgabe zu erkennen, wo die UserIn auf dem Bildschirm blickt. Dazu gehören das Erkennen des Auges und die Blickrichtung. Um später erkennen zu können, welche Stelle des Bildschirms die UserIn betrachtet, braucht dieser gewisse Kalibrierungsdaten, die beim Starten eingegeben werden müssen.

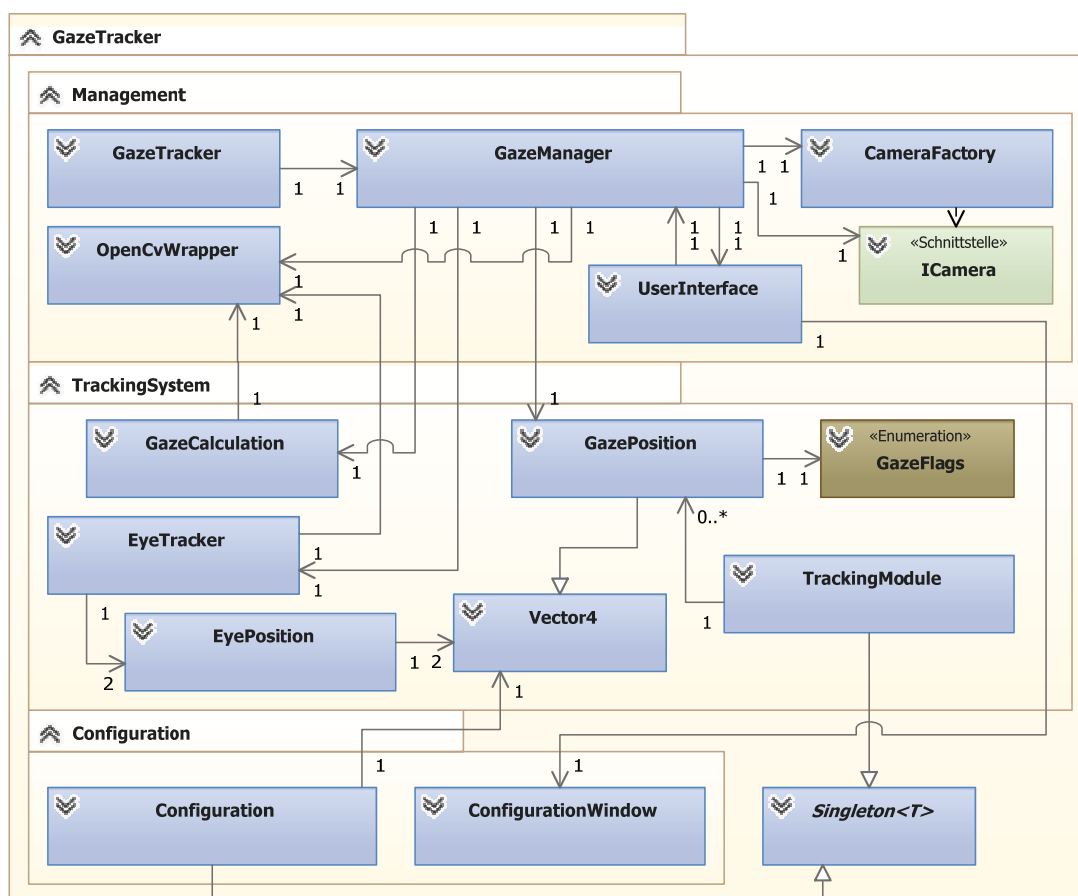
Configuration

Sie speichert alle Einstellungen vom ganzen System, die für den GazeTracker notwendig sind. Dazu gehören die Einstellungen, für was der GazeTracker verwendet wird (Analyse oder Steuerung) und die Bildschirmkalibrierung.

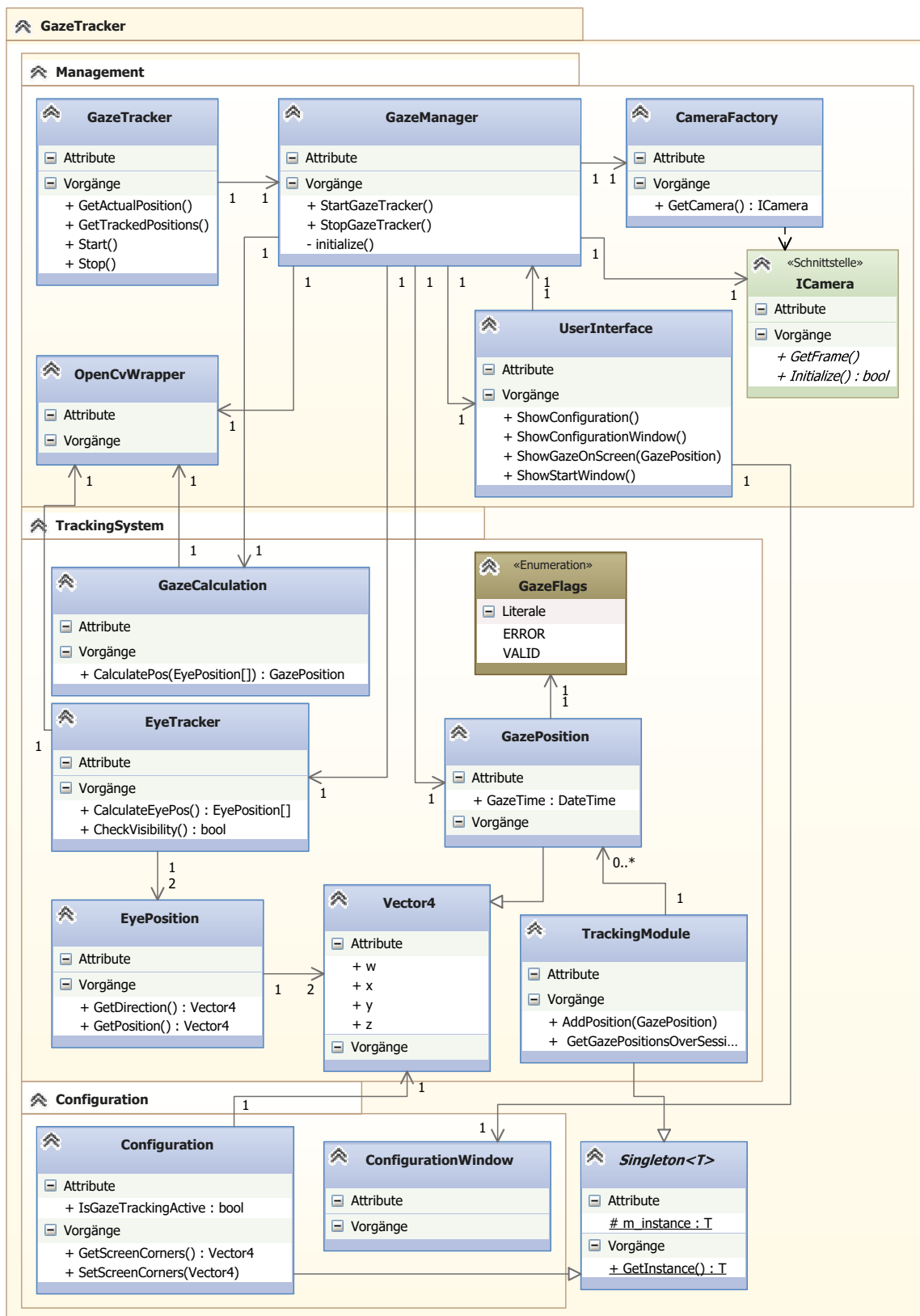
Management

Diese beinhaltet den Informationsaustausch zwischen der Applikation und dem GazeTracker System. Sie dient als Schnittstelle zwischen den einzelnen Komponenten.

3.2.1.2 Overview of data design



3.2.1.3 Interfaces to/from internal and external components



3.2.1.4 [User Interface](#)

Mock-Ups

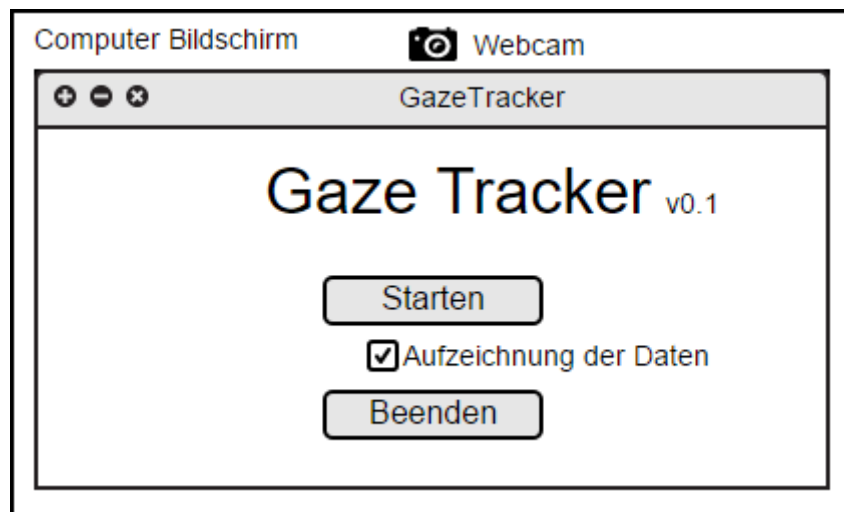


Figure 1: Startdialog

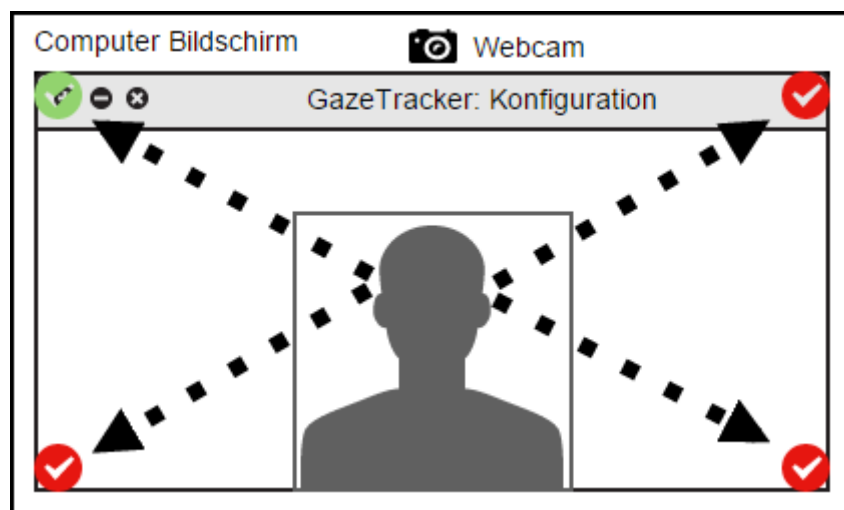


Figure 2: Konfigurationsdialog

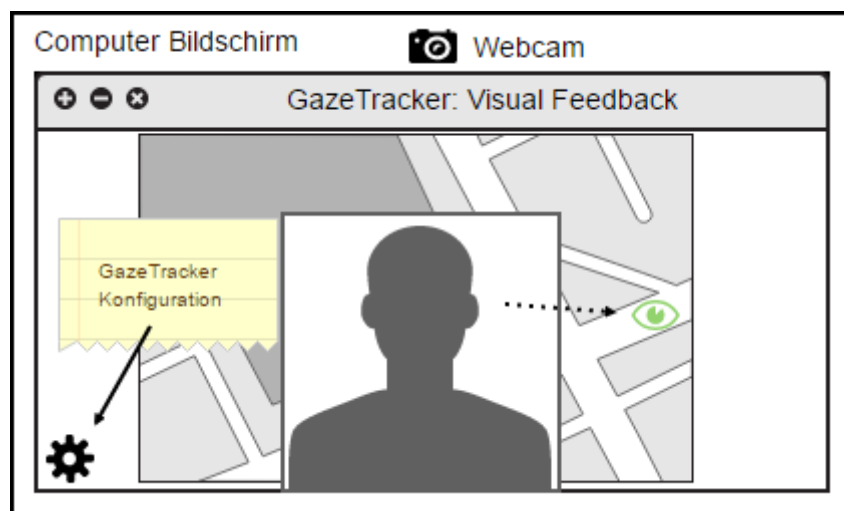


Figure 3: Anstarrpunkt

Benutzerinteraktionen

Zu Beginn steht der UserIn ein Startdialog zur Verfügung, mit dem der GazeTracker gestartet oder beendet werden kann. Zudem gibt er hier eine Auswahlbox, für die Speicherung der Daten, welche für die späteren Auswertungen verwendet werden kann.

Wenn die UserIn den GazeTracker startet, erscheint ein Konfigurationsdialog, indem die UserIn dazu aufgefordert wird, sich gerade vor der Webcam zu positionieren und in jede Ecke ihres Bildschirms zu starren. Sobald ein Punkt erfasst wurde, wechselt das Icon von der Farbe Rot zur Farbe Grün. Nachdem alle Ecke erfasst wurden wird der GazeTracker gestartet und kann verwendet werden.

Zur erleichterten Bedienung wird der UserIn ein visuelles Feedback des Anstarrpunktes auf dem Bildschirm angezeigt. Dieses wird in Form eines Auges in der Farbe Grün oder Rot angezeigt. Grün bedeutet hierbei, dass alles fehlerfrei funktioniert und die Farbe Rot, dass der Punkt nicht erfasst werden konnte. Zusätzlich gibt es in diesem Fenster für die UserIn noch frei wählbare Einstellungen, mit denen das visuelle Feedback abgeschaltet werden kann, die Eckpunkte des Bildschirms neu konfiguriert werden können oder die Speicherung der Daten unterbrochen werden kann.

3.2.1.5 Cross References

- Use-Case: *Augenblickrichtungen analysieren*
- Use-Case: *Konfiguration des Systems*

3.3 Architecture for Component *Application*

3.3.1.1 Design Considerations

Diese Komponente stellt die Anwendung dar, die den GazeTracker verwendet und ist die zweite Hauptkomponente des Projektes.

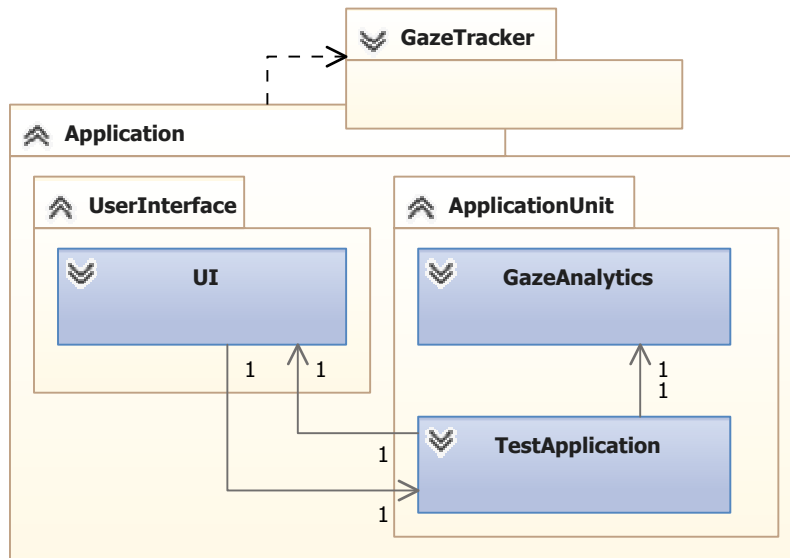
ApplicationUnit

Sie beinhaltet die Routinen und die Logik von der Anwendung. Es ist hier allgemein gehalten, da diese von einfacher Analyse bis hin zur Steuerung mit dem GazeTracker darstellen soll. Im unseren Projekt wird dieser ein Analysetool sein.

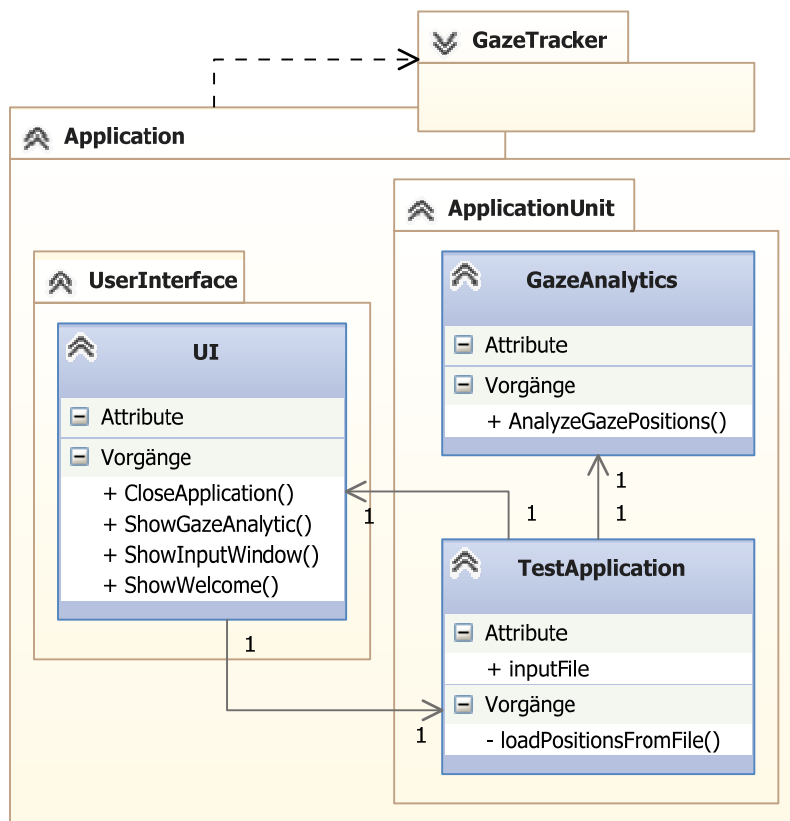
UserInterface

Ist die Komponente, die die UserIn als Fenster zur Gesicht bekommt und dadurch auch analysiert werden kann, welche Stelle sie gerade anstarrt.

3.3.1.2 [Overview of data design](#)



3.3.1.3 [Interfaces to/from internal and external components](#)



3.3.1.4 [User Interface](#)

Mock Ups

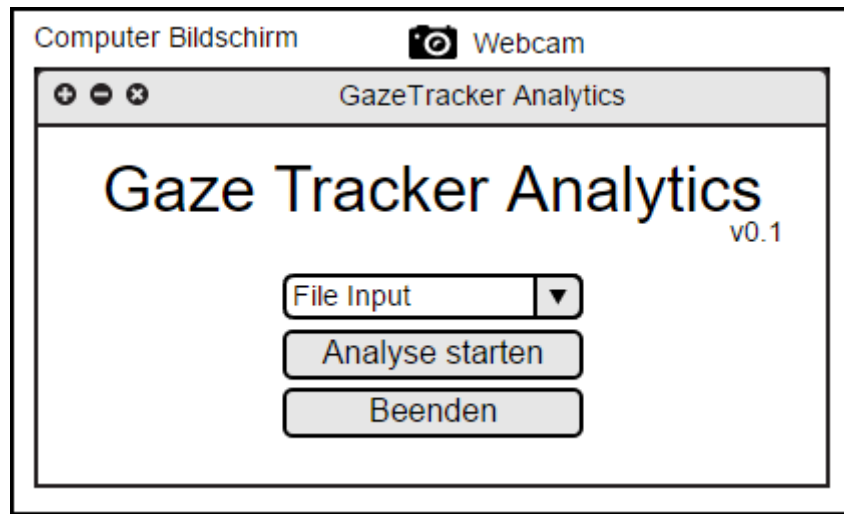


Figure 4: Startbildschirm

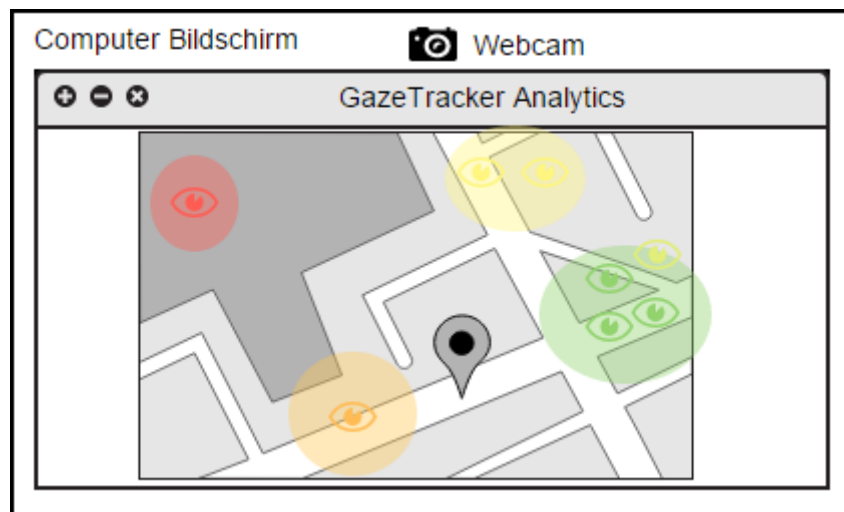


Figure 5: Beispiel einer Analyse

Benutzerinteraktionen

Der Startbildschirm der Test Applikation bietet der UserIn wenige Auswahlmöglichkeiten. Entweder wird die Analyse der Daten vom Gaze Tracker gestartet, indem eine Datei mit den aufgezeichneten Daten vorgelegt wird, oder die Applikation wird geschlossen.

Wird die Analyse gestartet, wird ein Bild erzeugt, welches die Augenkontakt Positionen mit dem Bildschirm anzeigt. Diese werden farblich markiert, um zu verdeutlichen, an welchen Positionen die UserIn oft geblickt hat. Grün bedeutet dabei am meisten Interesse und Rot am wenigsten. Die Farben dazwischen können interpoliert interpretiert werden.

3.3.1.5 [Cross References](#)

- Use-Case: Daten der Beobachtung analysieren

4 Project Management

4.1 Milestones and Schedules

1. Meilenstein

- Zugriff auf Webcam -> 18.04.2105
- Face/Eye Recognition -> 01.05.2015
- Ermitteln der Augen Position und Blickrichtung -> 13.05.2016

2. Meilenstein

- Berechnung des Auftrittpunktes auf dem Bildschirm -> 25.05.2016

3. Meilenstein

- Implementierung der Showcase für GazeTracker -> 17.06.2016

4. Meilenstein

- Fertigstellung des Projektes -> 01.07.2016

4.2 Planned Effort per Person

Um das Projekt erfolgreich zu beenden, wurden verschiedene Meilensteine gesetzt, die zu bestimmten Zeitpunkten beendet sein sollten. Zusätzlich ist eine Dauer angegeben, welche die Zeit für den jeweiligen Meilenstein schätzt, die benötigt wird. Diese Zeit ist in Arbeitsstunden pro Person angegeben. Basis dafür sind die 2*6 ECTS Punkte, die eine geschätzte Zeit von insgesamt 300h ergibt.

Bei diesem Projekt wird auf eine Einteilung verzichtet, die Vorschreibt, welche Person was machen muss. Es wird versucht, dass jedes Teammitglied in den einzelnen Meilensteinen (Schritte) miteinbezogen wird und die Aufgaben die anfallen auch umgesetzt werden.

Abschnitt	Beginn	Ende	Dauer in h
Anforderungsanalyse	12.03.2016	08.04.2016	20
Meilenstein 1	09.04.2016	13.05.2016	100
Meilenstein 2	14.05.2016	25.05.2016	80
Meilenstein 3	26.05.2016	17.06.2016	50
Meilenstein 4	18.06.2016	01.07.2016	50

Hier sehen sie dazu das GANTT Diagramm:

