

Diplomarbeit

Gesichtsregistrierung



AUTOR: ARON TERZETA

Inhaltsverzeichnis

1	Gesichtsregistrierung	1
1.1	Umsetzung	1
1.1.1	Allgemein	1
1.1.2	Verwendete Technologien	1
1.1.3	Technische Lösung	2
1.1.4	Herausforderungen	8
1.1.5	Qualitätssicherung und Controlling	10
1.2	Ergebnisse	12
1.2.1	Implementierung	12

Kapitel 1

Gesichtsregistrierung

In diesem Kapitel wird genauer beschrieben, wie der Gesichtsregistrierungsteil funktioniert.

1.1 Umsetzung

Hier wird erklärt, wie der Gesichtsregistrierungsteil implementiert wird.

1.1.1 Allgemein

Sicherheit ist heutzutage hoch interessant und relevant in vielen technischen und nicht technischen Bereichen. Das System, das entwickelt wird, hat mit Gesichter von Personen zu tun. Alle wissen, dass das Gesicht für jede Person anders ist. Jede Person wird mit ihrem Gesicht authentifiziert, weil es einzigartig ist. Das Gesicht hat Merkmale, die von verschiedenen Algorithmen bestimmt werden können, um diese für die Authentifizierung der Personen zu verwenden

Das System ist in zwei Teile geteilt. Es gibt den Registrierungsteil und den Erkennungsteil. Bei dem Registrierungsteil wird die komplette Registrierung der Schüler und Schülerinnen, der Lehrer und Lehrerinnen durchgeführt. Bei dem Erkennungsteil wird die Authentifizierung der Schüler und Schülerinnen gemacht.

1.1.2 Verwendete Technologien

Hier werden die Technologien beschrieben, die verwendet wurden. Das andere Paket namens "git" wird auch verwendet. Git ist ein Versionsverwaltungssystem, das verschiedene Versionen einer Software auf einem Server speichert. Auf dem Server gibt es dann verschiedene Versionen der Software und wir können zu den alten Versionen zurückkehren. Es wird meistens bei der Implementierungsphase verwendet, um die Veränderungen des Source-Codes zu verwalten.

Ein anderes Paket, das wir verwenden, heißt cmake¹ ist ein Paket, das gebraucht wird,

¹ein erweiterbares Open-Source-System, das den Erstellungsprozess in einem Betriebssystem und auf compilerunabhängiger Weise verwaltet.

wenn man mit OpenCV² arbeitet. Es muss das System so konfiguriert sein, damit das OpenCV-Framework in einem C++ Programm verwendet werden kann. Deshalb brauchen wir spezielle cmake Befehle, die dies ermöglichen.

”CMake wird verwendet, um den Softwarekompilierungsprozess mithilfe einfacher plattform- und compilerunabhängiger Konfigurationsdateien zu steuern und native Makefiles und Arbeitsbereiche zu generieren, die in einer Compilerumgebung Ihrer Wahl verwendet werden können.” [1]

Das gleiche passiert auch, wenn z.B. Python statt C++ verwendet wird. Die anderen Pakete wie z.B. libgtk2.0-dev pkg-config, libavcodec-dev, libavformat-dev, libswscale-dev, sind benötigte Paketen, damit man das OpenCV-Framework verwenden kann.

Linux als Betriebssystem [6]

Linux ist das meist verwendete Betriebssystem der Welt. Es ist eine open-source Software. Linux ist flexibel, man kann die einzelnen Modulen wegnehmen, ohne dass das Betriebssystem abstürzt. Der Benutzer kann auch die Kernkomponenten wählen wie z.B. welcher System-Grafiken angezeigt werden, bzw. die ganzen Komponenten der Benutzeroberfläche können ausgewählt werden. Ich habe Linux aus verschiedene Gründe gewählt. Linux ist für eingebettete Systeme sehr geeignet. Es ist sicher gegen Schadprogrammen, Viren und Trojanern. Linux ist einfacher. Vorher war es ein kompliziertes System, seit den Bemühungen der Ubuntu-Fundationen und der Ubuntu-Distribution ist es sehr einfach verwendbar.

Python als Programmiersprache

”Python ist eine Programmiersprache, die 1991 veröffentlicht wurde. Python besitzt eine einfache Lesbarkeit und eine eindeutige Syntax. Python lässt sich leicht erlernen und unter UNIX, Linux, Windows und Mac OS verwenden.” [2] Warum Python gewählt wurde, hat verschiedene Gründe. Python hat weniger Schlüsselwörter, reduziert die Syntax auf das Wesentliche und optimiert die Sprache. Ein Programm, das in Python geschrieben ist, ist vom Betriebssystem unabhängig. Das bedeutet, es kann Plattform unabhängig interpretiert werden.

1.1.3 Technische Lösung

In diesem Unterabschnitt wird alles was mit Technik zu tun hat erklärt. Es wird eine detaillierte Beschreibung im Bezug auf die technischen Lösungen gemacht.

Das System besteht aus verschiedenen sogenannten Terminatoren. Ein Terminator befindet sich außerhalb des zu definierenden Systems. Es kann eine andere Person, System oder eine Organisation sein. ”Terminatoren können von unserem System Informationen, Nachrichten, Materialien oder Energie erhalten oder das System empfängt diese.” [7]

Ein Terminator ist der ”Register-Schalter”. Er initialisiert das ganze System. Schalter in der Technik ist nichts anderes, als ein Gerät zum Ein- und Ausschalten des Stroms

²eine Bibliothek von Programmierfunktionen, die hauptsächlich auf Computer Vision in Echtzeit abzielen.

oder zum Leiten des Stromflusses. Wenn der Schalter gedrückt wird, bekommt das System einen Input, transformiert ihn und gibt dann einen Output. Das System ist sehr einfach verwendbar.

Die Register-LED³ dient als eine Anzeige. Wenn mit dem System etwas nicht stimmt, dann wird mit rot geleuchtet. Wenn alles passt, dann wird mit grün geleuchtet. Eigentlich hat die normale LED nur eine Farbe, aber es wird eine spezielle LED verwendet, namens RGB LED⁴. Eine RGB LED hat drei Grundfarben, rot, grün und blau. Mit diesen drei Farben kann man alle Farben darstellen. Es könnte auch zwei LEDs geben, rot und grün, aber es ist effektiver, eine RGB LED zu verwenden.

Eine spezielle Eigenschaft des Systems ist die Verwendung einer Tastatur. Sie wird verwendet, weil die einzelnen Personen ihren Namen, bzw. ihr Email eingeben müssen. Die andere spezielle Eigenschaft ist die Verwendung eines LCD-Bildschirms⁵. Da werden z.B. Errors oder die Login-Daten angezeigt. Es ist für den Benutzer einfach zu erkennen, wenn es ein Problem mit dem System gibt. Eigentlich ist die Hauptfunktion des LCD-Bildschirms, den Input des Benutzers anzuzeigen. Warum es so geplant ist? Das Problem besteht darin, dass, wenn der Benutzer seine Email-Adresse schreibt, Fehler passieren können, weil er nicht sieht, was er schreibt. Um das zu vermeiden, wird der LCD-Bildschirm verwendet. Der Benutzer kann sehen, was er schreibt.

Um registrierte Personen mit ihren Gesichtsdaten zu speichern, braucht das System einen Server. Auf diesem Server läuft ein Datenbank Management System. Die Datenbank ist so konfiguriert, dass die Person mit ihren Informationen gespeichert werden können.

Das System hat auch einen Backup-Server. Die Daten werden parallel bei dem Hauptserver sowie bei dem Backup-Server gespeichert, damit die Daten noch verfügbar sind, wenn der Hauptserver ein Problem hat. Die Verwendung des Backup-Servers ist zustande gekommen, weil das System 24/7 arbeiten muss. Wenn der Hauptserver Wartungen oder Probleme hat, kann der Backup-Server die Arbeit übernehmen. Auf Abb. 1.2 wird gezeigt, wie der Gesichtsregistrierung-Teil arbeitet.

³Light-emitting Diode, fließt Strom durch, strahlt sie Licht aus

⁴rot, grün,blau LED

⁵Liquid Crystal Display, präsentiert die elektrischen Signale in Form von visuellen Bildern.

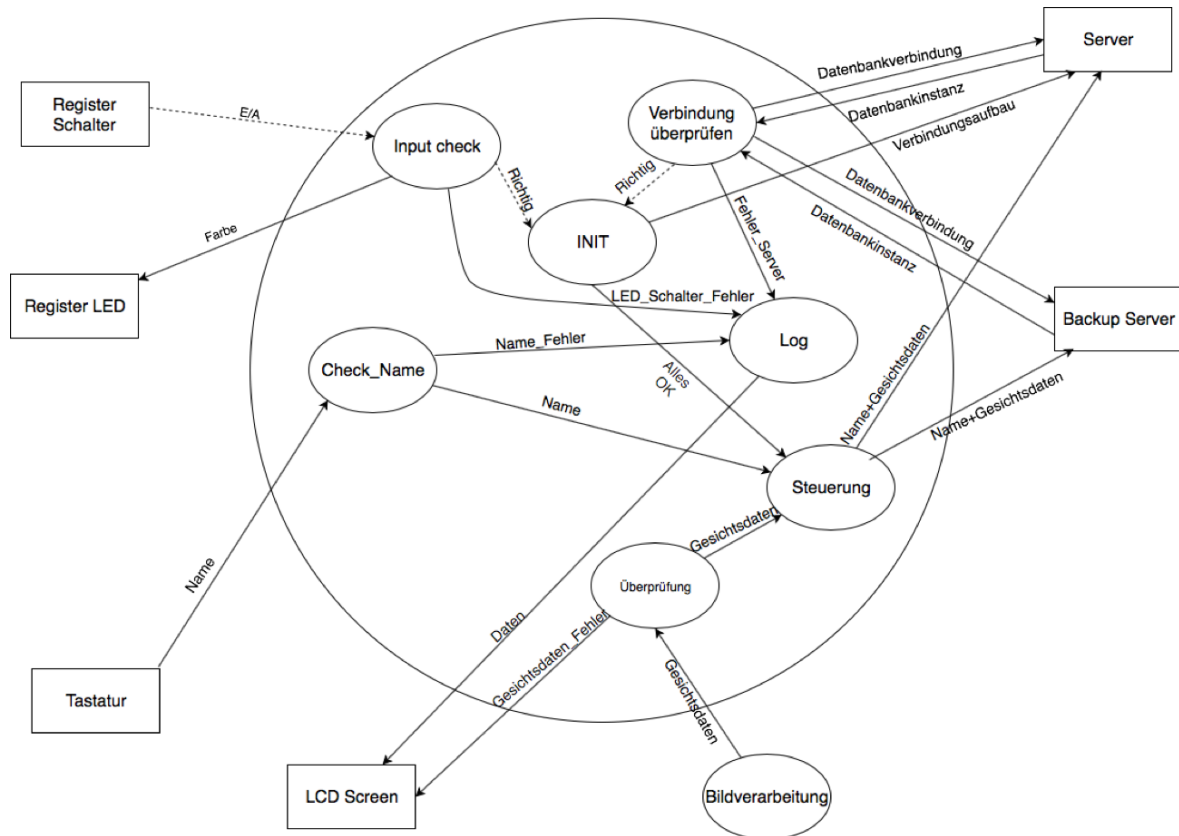


Abbildung 1.1: Structed Software Design bzw. erste Ebene

Legende:

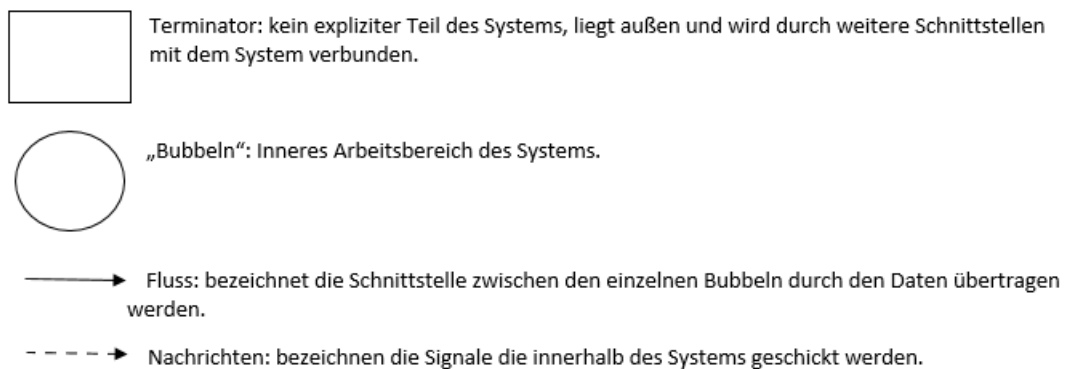


Abbildung 1.2: Structed Software Design Erklärung

Das System hat verschiedene Terminatoren. Die Terminatoren sind im Kapitel 1.1.3 erklärt. Auf Abbildung 1.2 können 6 Terminatoren identifiziert werden:

- Register Schalter - Externer Trigger Schalter zum Starten des Registrierungsteils
- Register LED - Anzeiger
- Tastatur - Input Hardware für die Datenerfassung(Name,Email,Rolle)
- LCD Screen - Anzeiger
- Server bzw. Backup Server - eigener Server für die Speicherung der Daten

Auf der erste Ebene können 7 Hauptprozesse identifiziert werden:

Um Schalter und LEDs im System verwenden zu können, brauchen wir ein spezielles Paket namens "RPi.GPIO". Dieses Paket macht es möglich, den Raspberry PI mit Hardware (LED und Schalter) verbinden zu können. Dafür werden GPIOs verwendet. Der Schalter hat 3 Anschlüsse. Einer wird mit 5V verbunden, der andere mit Ground und der dritte ist für Daten. Dies wird dann mit einem GPIO-Port in Raspberry PI verbunden. Das gleiche gilt auch für die LED, damit sie vom Raspberry PI kontrolliert werden kann, wird sie mit einem GPIO-Port verbunden. Mithilfe dieser GPIO-Ports bekommt das System eine Information, wenn der Schalter gedrückt wird. Wert "1" bedeutet, dass der Schalter gedrückt wird und die verschiedenen Skripten aufgerufen werden.

Schritte:

1. Am Beginn des Skripts wird diese Zeile schreiben: "#!/usr/bin/python". Es gibt zwei Gründe, warum diese Zeile geschrieben wird. Der erste Grund ist, dass dieses Program mit einem Python-Interpreter ausgeführt wird, der zweite ist, der Inhalt der Datei wird von der Python-Binärdatei unter /usr/bin/python interpretiert
2. Alle Paketen importieren. Sehen Sie Listing. 1.1

Listing 1.1: Packages zu importieren

```
import RPi.GPIO as GPIO
import time
import os
import subprocess
import sys
```

- RPi.GPIO ist ein Paket, das verwendet wird, um Zugriff auf die sogenannten GPIO-Ports⁶ zu haben. Vorher habe ich erwähnt, wenn wir Zugriff auf die HW-Komponenten haben wollen, die mit Raspberry PI verbunden sind, brauchen wir die GPIO-Ports. Um diese GPIO-Ports in Python zu verwenden, brauchen wir das sogenannte Paket "RPi.GPIO". Es gibt verschiedene Pakete, die einen Zugriff zwischen GPIO-Ports und Python ermöglichen, wie z.B. rpi.gpio, GPIOZero usw. Es wird das rpi.gpio Paket verwendet, weil es leicht verständlich, programmierfreundlich und einfach zu verwenden ist. [8]
- time ist ein Paket in Python. Von diesem Paket wird nur die Funktion "sleep" verwendet. Diese Funktion pausiert das python-Programm. [4]
- os ist ein wichtiges Paket in unserem Skript. Es erlaubt uns, dass wir in einem Python-Skript andere Skripte aufrufen können. Es ist egal, in welcher Programmiersprache diese Skripten geschrieben sind. Es gibt auch verschiedene Methoden, wie man verschiedene Skripte in einem Python-Skript

⁶Eingehende und Ausgehende digitale Signale, am Eingangsport können verwendet werden, um digitalen Messwerte mitzuteilen, die von Sensoren empfangen werden

aufrufen kann. Man kann mit dem subprocess Paket, eine Main-Funktion in dem Skript starten und die verschiedenen Funktionen des anderen Skripts aufrufen.

- Das “subprocess” Paket dient zur Verbindung zwischen verschiedenen Prozessen, in meinem Fall, den Aufruf eines Skriptes.
- Das sys Paket wird verwendet, um Console Parameter zu bekommen. Das bedeutet, wenn das Skript aufgerufen wird, wie z.B. login.py dann kann man login.py einen Parameter mitgeben: login.py <parameter >

3. GPIO-Ports Datenrichtung einrichten. Datenrichtung für LED ist “out”, weil das LED als ein Output für unseres System dient. Datenrichtung für Schalter ist “in”, weil der Schalter als ein Input für unseres System dient. Auf Listing. 1.2 ist der Python-Code.

Listing 1.2: GPIO-Ports Konfiguration

```
GPIO.setmode(GPIO.BCM)
GPIO.setup(23,GPIO.OUT)
GPIO.setup(18,GPIO.IN)
GPIO.setup(17,GPIO.IN)
GPIO.setup(27,GPIO.OUT)
```

Es gibt verschiedene Betriebsarten für GPIO wie BCM und Board. Ich verwende BCM⁷, weil ich das Paket RPi.GPIO verwende. Mit diesem Paket darf nur die Betriebsart ”BCM ”verwendet werden. [5] Für die Registrierung der Schüler und Schülerinnen bzw. Lehrer und Lehrerinnen ist es nötig, dass der Administrator sich bei dem System einloggt. Die Überprüfung, ob der Administrator da ist oder nicht, wird mit einem Vergleich von zwei Bildern gemacht. Ein Bild von Administrator ist gespeichert, das andere wird gemacht. Dazu wird das Skript, das Bilder macht, aufgerufen, und dann vergleicht man es mit einem Skript mit dem anderen Bildern. Es gibt “matched” zurück, wenn die Gesichter bei den beiden Bildern übereinstimmen und ”not matched” wenn die Gesichter nicht übereinstimmen.

4. Dann kommt der Teil “Input check”. Hier verwende ich dann die Methode “input”. Die Methode befindet sich im Paket “RPi.GPIO” und gibt entweder true oder false zurück. Im Verzeichnis /sys/class/gpio/gpio<GPIO-PORT>” gibt es zwei Dateien, value and direction. Direction für den Port des Schalters ist IN und

⁷Broadcom Pin Number

für den Port des LEDs ist OUT. Mit der Methode “input” hole ich den Wert (value) der Schalter-Port. Wenn der Schalter gedrückt wird, wird der Wert “1” herausgekommen, 1 repräsentiert “true”. Das bedeutet, dass die Input-methode “true” zurückliefert und das Programm läuft weiter.

Nachdem der Schalter gedrückt wird, wird ein Skript aufgerufen. Dieses Skript dient zur Registrierung der Person in der Datenbank. Für die Registrierung der Schüler und Schülerinnen bzw. Lehrer und Lehrerinnen ist es nötig, dass der Administrator sich bei dem System einloggt. Ist der Administrator da, können die Personen mit der Registrierung beginnen. Diese Person wird nach ihrem Vornamen, Nachnamen, Email und Rolle gefragt. Mit der Rolle ist gemeint, die Funktion bzw. die Stelle dieser Person in der Schule, ob die Person ein Schüler, Lehrer oder ein Administrator ist. Die Rolle wird vom Administrator eingegeben. 1 steht für Administrator, 2 für Schüler und 3 für Lehrer und Lehrerinnen. Sie wird eigentlich nur für die Verwaltung des Systems verwendet. Wenn der Administrator wissen möchte, wie viele Lehrer oder Schüler schon registriert sind, oder ob es noch nicht registrierte Personen gibt, dann schaut er in der Datenbank nach. Die E-Mail speichern wir dann in einer Variable und diese Variable übergeben wir dann an ein anderes Skript. Dieses Skript erstellt mit der Kamera eine Verbindung und macht ein Bild. Der Name des Bildes ist gleich mit der Email der Person. Es ist so gewählt, weil es für das Einfügen der Daten in der Datenbank und bei der Speicherung des Paths des Bildes in der Datenbank mit dem gleichen Namen wie die E-Mail einfacher ist.

Es wird die E-Mail verwendet, weil es eine performantere Suche in der Datenbank ermöglicht. Die E-Mail ist einzigartig, nur einmal für jede Person. Von 1000 Datensätze wird nur der Datensatz ausgegeben, in dem die E-Mail mit der eingetippten E-Mail vom Benutzer übereinstimmt.

In der Datenbank gibt es nur eine Tabelle. In dieser Tabelle wird alles gespeichert: Name, E-Mail, Rolle, Path des Bildes und alle Gesichtspunkte. Für die Registrierung der Personen in der Datenbank werden zwei Schritte durchgeführt. Der erste Schritt ist das Einfügen der neuen Person mit ihrem Vornamen, Nachnamen, E-Mail und Rolle in der Datenbank. Der zweite Schritt ist das Update der Gesichtspunkte. Es wird ein Bild gemacht, von diesem Bild werden die Gesichtspunkte extrahiert. Danach werden diese Gesichtspunkte in den zuständigen Spalten der Datenbank gespeichert. Dieser Schritt wird durch eine Update-Anweisung gemacht. Nachdem die Update-Anweisung erfolgreich ist und die Person mit ihrem Vornamen, Nachnamen, E-Mail, Rolle, und ihren Gesichtspunkten gespeichert ist, wird die LED grün angesteuert. Wenn ein Fehler auftritt, wird die LED rot angesteuert. Es gibt einige Fehler, die bei der Registrierung auftreten können:

- (a) Auf dem Bild gibt es kein Gesicht oder mehr als ein Gesicht.
- (b) Das Gesicht ist schlecht positioniert und es ist nicht möglich, die Gesichtspunkte zu extrahieren.
- (c) Die Person ist schon registriert.

Wenn ein von den oben erwähnten Fehler auftritt, wird das auf dem Bildschirm angezeigt und die LED wird rot angesteuert.

1.1.4 Herausforderungen

Das Projekt hatte für mich viele Herausforderungen. Das lag daran, dass es ein ziemlich großes Projekt war, und ich neue Technologien verwendet habe, die ich vorher nie verwendet hatte. Ich habe z.B. keine Erfahrung mit OpenCV, Python und andere Dinge, die ich später erwähnen werde. Ich habe von diesen Herausforderungen und Problemen viel gelernt. Einerseits bin ich froh, andererseits bin ich frustriert, weil dies das Enddatum des Projekts verzögert hat. Die Herausforderungen waren:

- **opencv zu installieren.** Es hat 3 Woche gedauert, bis ich es installiert hatte.
- **Beginn des Projektes.** Der Beginn eines Projektes ist immer schwierig. Die Koordination im Team war sehr schwierig. Ich, als Projektleiter, musste allen sagen, wie sie arbeiten sollen, wo sie die Dateien finden können usw. Das war die größte Herausforderung.
- **Git repository, Einrichtung von git.** Manche Teammitglieder wussten sehr wenig von git und ich musste es ihnen erklären. Manchmal gab es merge conflicts, weil sie pull gemacht haben, ohne dass Sie die Änderungen committed haben. Ich sollte dies lösen, weil ich mehr Erfahrung mit git hatte.
- **Python als Programmiersprache.** Wir wollten vorher mit C++ arbeiten, aber es war sehr schwierig, OpenCV in Visual Studio zu installieren. Manche von uns wollten in Windows arbeiten und der einzige Weg war, mit Visual Studio zu arbeiten. Es hat nicht funktioniert, deshalb sind wir zu Python gewechselt. Wir haben Python gewählt, weil opencv in Python sehr einfach installierbar ist. Mit Python hatten wir keine große Erfahrung. Alles, was ich mit Python gemacht habe, ist eine Verbindung mit der Datenbank und Aufbau der SQL-Abfragen (Insert, Select, Update, Delete). Alle andere Wissen sollte ich selbst von Bücher, Internet, Tutorials lernen. Die große Herausforderung hier war, die richtigen Quellen zu finden.
- **Verwendung der Kamera und verbinden mit Python.** Ich wusste nicht, welche Funktionen man verwendet, um die Verbindung mit der Kamera zu erstellen.
- Bei der älteren Version von **Raspbian** heißt das Paket, das python mit dem Datenbank Managment System(MySQL) verbindet, 'python-mysqldb' und jetzt heißt es 'python-mariadb'. Ich wusste das nicht und dies hat mir etwas Zeit gekostet.
- **Abhängigkeiten zwischen einzelnen Arbeitsteilen.** Die Aufgaben waren so geteilt, dass es Abhängigkeiten zwischen ihnen gab. Das hat dann zu einer Verspätung der Projektabgabe geführt, weil die Teammitglieder aufeinander warten mussten.
- Die großen Teile meiner **Planung** haben gepasst, nur wenige Kleinigkeiten musste ich ändern. Sie sind erst in der Implementierungsphase angezeigt.

- Problem mit dem Zugriff auf die Elementen eines **Numpy-Arrays**.
- **Verbindung der Skripte miteinander** und Verknüpfung der Variablen, die sich in verschiedenen Skripten befinden.
- **Aufteilen der Gesichtspunkte in zwei Arrays**. In einem Array nur die X-Werte und in dem anderen Array nur die Y-Werte.
- **Die Aufgabenteilung** zwischen den Teammitgliedern war unausgewogen.

Ich habe diese Lösungen für die Herausforderungen gefunden:

1. Ich habe viel Tutorials geschaut, Websites gelesen, wie opencv in Raspberry PI installiert werden kann. Ich habe viele verschiedene Methoden ausprobiert, aber keine guten Ergebnisse erzielt. Nach vielen Versuchen wurde es erfolgreich installiert. Dannach habe ich verschiedene Skripte in Python gemacht, um es zu testen. Manche der Skripte haben funktioniert, manche nicht. Eine kleine Herausforderung war für mich, dass ich die Skripte, die nicht richtig funktioniert haben verbessert. Anschließend habe ich herausgefunden, dass das Problem bei dem Kompilieren von opencv aufgetreten sind (cmake). Ich habe es noch einmal vom Beginn kompiliert. Das hat das Problem gelöst, alle Skripte funktionieren, es gibt keine Fehler mehr, die mit dem opencv Paket zu tun haben.
2. Ein Treffen mit meiner Gruppe vor dem Beginn des Projektes war notwendig. Ich habe Ihnen gesagt und erklärt, in welchen Verzeichnisse sie arbeiten sollten, die Struktur der Dokumentation, welcher Kommunikationskanal verwenden wir, um Probleme, Herausforderungen usw. zu besprechen. Jede Person hatte ihre Vorschläge, um Aufgaben zu lösen, und dieses Treffen hat sehr lange gedauert, bis alle verstanden hatten, wie, wo, was, wann gemacht werden soll. Aber auch nach dem Treffen gab es zwischendurch Missverständnisse bzw. Probleme mit der Kommunikation, z.B. wurde nicht im richtigen Verzeichnis gearbeitet usw.
3. Ein Git-Repository zu erstellen und einzurichten war einfach. Ich hab es online in github.com erstellt, einen Name eingegeben und dann die anderen Teammitglieder als Collaborators hinzugefügt. Um strukturierter zu arbeiten, habe ich verschiedene Branches angelegt. Es gab mit dem Befehl "push" und "pull" Probleme. Das habe ich gelöst, indem ich allen Teammitgliedern gesagt habe, dass, wenn sie in einem Github-Repository arbeiten möchten vor dem Beginn der Arbeit einen "pull" machen müssen, damit die Änderungen, die von anderen in dem Repository gemacht wurden, mit der Version am Computer synchronisiert wird. Sie wussten nie, was die anderen in diesem Repository gemacht haben. Sie machen "push", ohne zu sagen, dass sie einen "push" gemacht haben. Das führt dann zu merge-Probleme usw.
4. Ich habe jedem Teammitglied gesagt, er muss mindestens zwei Wochen mit dem Lernen von Python verbringen. Tutorials ansehen, Beispiele selbst probieren, die Quellen dafür selbst finden.

5. Für die Verbindung der Kamera mit OpenCV, gibt es einen Skript in der offiziellen Website-Dokumentation von OpenCV. Da habe ich alle Funktionen gesucht und gefunden, die ich brauchte, um die Kamera in Python verwenden zu können.
6. Um die Abhängigkeiten zu minimieren, hat jedes Teammitglied andere Aufgaben bekommen, als die, die in der Dokumentation stehen. Ich war gezwungen, diese Änderung zu machen, sonst hätte das Projekt viel länger gedauert.
7. Bei der Implementierung sind Kleinigkeiten herausgekommen, die bei der Planung nicht berücksichtigt waren. Die habe ich direkt in der Implementierung verbessert, ohne dass ich noch einmal die Planung machte. Aber ich habe diese Kleinigkeiten zur Kenntnis genommen, damit ich in der Zukunft keinen solchen Fehler (Kleinigkeiten) mehr in der Planungsphase machen werde.
8. Ich habe die Dokumentation von Numpy Array nicht gut gelesen. Ich bat meinem Betreuer um Hilfe, weil ein Problem mir viel Zeit genommen hat. Der Betreuer hat mir dann geholfen, die Lösung zu finden.
9. Alle Teammitglieder haben ihre Skripts erstellt. Es war meine Aufgabe, alle diese Skripts mit einem Prototyp-Skript zu verbinden. Es war keine wissenschaftliche Arbeit, man musste viel testen, während man den Prototyp erstellte. Nach vielen Tests, habe ich erfolgreich einen Prototyp erstellt.
10. Eigentlich sollte Egli die Gesichtspunkte in ein Array speichern. Sie konnte dieses Ziel nicht erfolgreich abschließen und ich sollte ihr helfen. Zuerst wollte ich alle Gesichtspunkte in ein 2D Array speichern. Nach vielen Versuchen kam ich zu dem Schluss, dass es schwierig war und viel Zeit kosten würde. Stattdessen habe ich die Gesichtspunkte in zwei Arrays gespeichert. In einem Array habe ich nur die x-Werte gespeichert und in dem anderen Array habe ich nur die y-Werte gespeichert. Das war eigentlich die einzige Möglichkeit, wie ich dieses Problem lösen konnte.
11. Am Beginn der Diplomarbeit hat das Team eine Aufgabenteilung gemacht. Mit der Zeit haben die Teammitglieder bemerkt, dass nicht jedes Teammitglied die gleiche Menge an Aufgaben hat. Dannach haben wir uns dazu entschlossen, eine neue Aufgabenteilung zu machen. Zuerst hatte ich auch den 2D vs 3D Unterschied zu realisieren. Mit der neuen Aufgabenteilung ist es ein Ziel von Jordi und nicht mehr ein Ziel von mir.

1.1.5 Qualitätssicherung und Controlling

Risiko ist meistens eine Einschätzung, was es einem Unternehmen kostet, wenn die Projektziele nicht erreicht werden. Ich, als Projektleiter, musste das machen. Eine Risikoanalyse zu erstellen ist sehr schwierig, weil es mit der Zukunft zu tun hat. Zuerst musste ich an die Zukunft denken, bei welchen Bauteilen können Fehler auftreten, welche Programme können ausfallen. Das bedeutet, einen Überblick über die Zukunft zu haben und einzuschätzen, was für Fehler und Risiken es geben kann. Dann schätzte ich die Wahrscheinlichkeit ihres Eintretens ab und am Ende die Maßnahmen. Dahinter

versteckt sich eine große Arbeit.[3] Auf Abb. 1.3 wird die Risikoanalyse dargestellt. Wahrscheinlichkeit, Kosten usw. sollen beachtet werden.

Risikoanalyse: Gesichtsregistrierung und Gesichtserkennung								
Risikotyp	Nr.	Wahr- sch.	Aus- wirk.	Ampel	Manager	Beschreibung	Behandlung und Kontrolle	Termin / Nächster Schritt
Standardrisiken								
Ressourcen	1	1	8	8	Rei Hoxha	Ausfall von Ressourcen	Ressourcen im Voraus sichern (Reserven an Mitarbeitern, an HW, an Zeit)	Projekt gleich abschließen
Planung	2	4	8	32	Aron Terzeta	Schlechte Planung (verschiedene Eintrittsfälle nicht berücksichtigt)	Nocheinmal mit der Planung beginnen	Den Projektantrag ablehnen
Technik	3	3	3	9	Egli Hasmegej	Nicht eindeutig Gesichtspunkte-Extraktion	Fertige Skripts aus dem Internet holen	Die Implementierung zu anderen Firmen zulassen
Staat	4	8	7	56	Aron Terzeta	Rechtliche Aspekte (Persönliche Informationen) + Datenschutz	Mit dem Staat vor der Implementierung sprechen	Geld zu Staat bezahlen, damit der Staat nicht in Schwierigkeiten uns bringt
Planung	5	2	4	8	Jordi Zmiani	Mehr Benutzer als geplant (Datenbankdesign)	ein skalierbares Design der Datenbank	nocheinmal Datenbank erstellen
Zeit	6	3	5	15	Alle	Spät mit Arbeit begonnen, z.B. die Installation von OpenCV sollte 4 Stunden aber ist 2 Tage	Schneller dann arbeiten oder vorher dem Kunden sagen, dass es ein bisschen spät das Produkt fertig ist.	Kunden sagen, dass entweder wartet bis zum Ende(eine große Verspätung) oder nimmt das nicht fertige Produkt zurück
Staat	7	4	0	0	Niemand	Wächter wird seinen Job verlieren, weil System viel Know-How braucht	-	-
Kommunikation	8	4	8	32	Alle	Wenn das Team keine gute Beziehung zwischen den Mitgliedern hat	Versuchen, einen gemeinsamen Weg und Sprache zu finden, wenn nicht, neue Teammitglieder	Team wechseln
Technik	9	3	3	9	Rei Hoxha	HW nicht genug, keine Know-How, wie man die speziellen Bauteile verwenden kann	Reserven, Bedienungsanleitungen lesen(auch in Internet suchen)	Experten fragen, Hilfe von Experten bekommen
Technik	10	5	5	25	Alle	Mangelnde Einführung	Tutorials sehen	Teile von Algorithmen und Skripts von Internet holen
Technik	11	3	3	9	Aron Terzeta	Nachträgliche Änderungswünsche des Systems	Vorher planen (Skalierbarkeit und Erweiterung)	Nocheinmal Planung
Technik	12	2	3	6	Aron Terzeta	Veränderung am kritischen Weg	Schnell den neuen kritischen Weg finden	Projekt ohne kritischen Weg(sehr gefährlich)
Technik	13	1	3	3	Aron Terzeta	fehlende Terminüberwachung	-	-
Zeit	14	2	2	4	Alle	Zeitprognose unterschätzen	Sagen, dass es eine Verspätung gibt. Der Kunde und der Chef muss es wissen	-
Zeit	15	3	4	12	Jordi Zmiani	Mangelnde Puffer in der Kalkulation	Mehr Stunde daran bis zum Ende arbeiten	Hilfe von außen bekommen
Ressourcen	16	1	1	1	Rei Hoxha	Mangelhafte Kontrolle der Projektkosten	Selbst dann den Mangel bezahlen	-
Ressourcen	17	2	4	8	Rei Hoxha	fehlende Ausrüstung	Direkt mit dem Projektleiter sprechen, und dann er entscheidet, ob es gekauft, ausgeliehen oder ... wird	-
Planung	18	0	1	0	Aron Terzeta	Geringe Personalkapazitäten	Entweder bleiben wir mit diesen Personalkapazität und das Produkt später fertig machen oder neue Personal einstellen, damit das Produkt in time fertig zu machen	-
Ressourcen	19	1	4	4	Alle	Ausfall einzelner Projektglieder	Ein anderer Projektglieder diese Arbeit machen	Einem neuen Team das Projekt einrichten

Abbildung 1.3: Risikoanalyse in Excel

Es gibt eine Priorität bei den Maßnahmen. Je höher die Zahl in der Spalte „Ampel“ ist, desto wichtiger die Maßnahmen und die Risiken sind. Die, in denen die rote Farbe ist, werden zuerst behandelt. Wenn es grün ist, bedeutet es, dass das Risiko eine sehr kleine Rolle auf das System spielt. Sie werden am letztens behandelt, wenn alle andere Maßnahmen für die roten und gelben Risiken getroffen werden. Spalte „Ampel“:

- 0 : das Risiko spielt keine Rolle oder eine sehr kleine Rolle. Die Maßnahmen für diese Risiken können getroffen werden.
- 1-3 : das Risiko hat eine Bedeutung. Die Maßnahmen für diese Risiken sollen getroffen werden.
- 3-100 : das Risiko spielt eine große Rolle auf das System. Die Maßnahmen für diese Risiken müssen getroffen werden.

1.2 Ergebnisse

Die Zeit für den Abschluss der Diplomarbeit ist vorbei. Jetzt sollen alle Ziele erfolgreich abgeschlossen werden. Meine Ziele sind eigentlich erfolgreich abgeschlossen. Eine Person kann sich mit ihrem Vornamen, Nachnamen, E-Mail und Rolle registrieren lassen. Die Gesichtspunkte werden von dem Gesicht dieser Person extrahiert und in der Datenbank zusammen mit Vornamen, Nachnamen, E-Mail und Rolle gespeichert. Das System hat zwei Administratoren. Die Administratoren loggen sich mit einem Passwort ein. Ohne das Einloggen des Administrators kann keine Person registriert werden. Am Beginn war geplant, dass das System ein LCD-Screen enthält. Jetzt geht es sich mit der Zeit nicht aus, dieses LCD-Screen zu programmieren und im System zu implementieren. Statt des LCD-Screens wird einen Bildschirm verwendet. Es wird nun alles auf dem Bildschirm angezeigt. Diese Veränderung hat eigentlich keine Bedeutung für das System.

1.2.1 Implementierung

Nachdem einer umfangreichen und guten Arbeit meinerseits, kann ich einen Prototyp mit einigen Funktionen vorweisen. Die Funktionen, die in diesem Prototyp integriert sind, gehören zu den Zielen, die ich realisieren sollte. Diese Funktionen sind:

- Administrator-Konto. Die Registrierung einer neuen Person geht nicht ohne ein Administrator-Konto. Der Administrator muss bei dem System eingeloggt sein, damit der Registrierungsteil funktioniert. Der Administrator loggt sich mit einem Passwort bei dem System ein.
- Eine Person wird mit ihrem Vornamen, Nachnamen, E-Mail und Rolle in der Datenbank gespeichert
- Ein Bild von einer Person wird gemacht
- Von diesem Bild werden die Gesichtspunkte extrahiert
- Die Gesichtspunkte werden in den Arrays gespeichert.
- Ich hole dann die Werte von diesen Arrays und speichere sie in der Datenbank.

Abbildungsverzeichnis

1.1	Structed Software Design bzw. erste Ebene	4
1.2	Structed Software Design Erklärung	4
1.3	Risikoanalyse in Excel	11

Tabellenverzeichnis

Literatur

Aus dem Netz

- [1] Kitware inc. *CMake*. 2000. URL: <https://cmake.org/overview/>.

Der ganze Rest

- [2] Stephan Augsten. *Was ist Python?* <https://www.dev-insider.de/was-ist-python-a-843060/>. [Online; accessed 2019]. 12/7/2019.
- [3] MA BSc Bekim Alibali. “Projektmanagement Teil5”.
- [4] Jackson Cooper. *Python’s time.sleep() – Pause, Stop, Wait or Sleep your Python Code*. <https://www.pythoncentral.io/pythons-time-sleep-pause-wait-sleep-stop-your-code/>. [Online; accessed Tuesday 23rd July 2013]. 2013.
- [5] Ben Croston. *RPi.GPIO Python Module*. <https://sourceforge.net/p/raspberry-gpio-python/wiki/Home/>. [Online; accessed 2014]. 2014.
- [6] CHIP Digital GmbH Niels Held. *Linux-Umstieg: So einfach gelingt der Windows-Wechsel*. https://www.chip.de/artikel/Linux-Umstieg-So-einfach-gelingt-der-Windows-Wechsel-2_140047889.html. [Online; accessed 2007]. 2007.
- [7] Dominik Stocklasser. “Architektur”.
- [8] Jeff Tranter. *Control Raspberry Pi GPIO Pins from Python*. <https://www.ics.com/blog/control-raspberry-pi-gpio-pins-python>. [Online; accessed Wednesday, July 31, 2019]. 2019.