PROJEKTHANDBUCH

**Projekt:**

***Innovative Optimierungen an der TAU***

Lehrveranstaltung: Mensch-Maschine Kommunikation

Semester:  4 bzw. 8 Semester

Hochschullehrer/in: Dr. Burcu Yildiz

Projektmanager/in: Dr. Zeynep Tuncer

Arbeitspaket-Head:

Arbeitspaket-Head:

Version: März 2019

**Inhaltsverzeichnis**

[**Änderungsverzeichnis**](#_30j0zll) **6**

[**Verzeichnis der Abkürzungen**](#_1fob9te) **8**

[**Verzeichnis der Abbildungen**](#_3znysh7) **10**

[**Verzeichnis der Tabellen**](#_2et92p0) **12**

[**Projektplanung**](#_tyjcwt) **13**

[Ansprechpartner](#_3dy6vkm) 13

[Projektauftrag](#_17dp8vu) 16

[Projektzieleplan](#_1ci93xb) 18

[Projektkontext](#_1pxezwc) 21

[Projektorganisation](#_49x2ik5) 23

[Projektkommunikationsplan](#_ihv636) 26

[Projektstrukturplan (PSP)](#_1hmsyys) 27

[Arbeitspaketspezifikation](#_o6d6iixcmfzx) 27

[Projektmeilensteinplan](#_t8zn8y2f7lxr) 38

[Projektterminplan](#_2u6wntf) 40

[Projektkostenplan](#_19c6y18) 42

[Projektrisiken](#_3tbugp1) 42

[Projektfortschrittsbericht](#_28h4qwu) 43

[**Projektumsetzung**](#_z90lecu4ki8k) **44**

[Einleitung](#_cd8b6g88xa8) 45

[Problemstellung](#_2zbgiuw) 45

[Stand der Technik](#_1egqt2p) 46

[METU App für Studenten - Anwendungsbeispiele](#_1baon6m) 46

Stanford University Mobile App - Anwendungsbeispiele47

[Zielsetzung](#_caoi9ktizb6z) 49

[Zielgruppen](#_67z48x1hw07x) 50

[Aufbau des Projekthandbuches](#_jsrnystow5et) 50

[**Android**](#_jy0bqi84ygkd) **51**

[Methodische Ermittlung](#_4dwdj1tz5wze) 51

[Methoden und Werkzeuge](#_odc9jc) 51

[Anforderungsanalyse und Konzeption](#_lklgx6qmuul8) 51

[Design (Entwicklung) und Implementierung](#_1rvwp1q) 54

[Anwendungsfälle](#_3q073x4tu0gg) 54

[EER und UML Domänenmodell](#_4bvk7pj) 56

[Funktionale und nichtfunktionale Anforderungen](#_ixf745daycn1) 61

[Systemarchitektur](#_2r0uhxc) 64

[Design-Layouts](#_1664s55) 68

[Evaluation](#_25b2l0r) 73

[Versuchsaufbau und Versuchsdurchführung](#_kgcv8k) 76

[Ergebnisse](#_xhjmyrnqd87) 76

[Interpretation der Ergebnisse](#_vkg6x92ok20q) 77

[Zusammenfassung und Ausblick](#_dvsjm499zpth) 78

[**IOS**](#_jvwcyqmfhezy) **79**

[Methodische Ermittlung](#_xkq5t9bosfyy) 79

[Methoden und Werkzeuge](#_1egqt2p) 79

[Anforderungsanalyse und Konzeption](#_k8g0wfkeldch) 80

[Design (Entwicklung) und Implementierung](#_1rvwp1q) 81

[Anwendungsfälle](#_kr6cxin30o92) 81

[EER und UML Domänenmodell](#_4bvk7pj) 82

[Funktionale und nichtfunktionale Anforderungen](#_c50cofatcolw) 87

[Systemarchitektur](#_2r0uhxc) 89

[Design-Layouts](#_1664s55) 89

[Evaluation](#_25b2l0r) 94

[Versuchsaufbau und Versuchsdurchführung](#_kgcv8k) 94

[Ergebnisse](#_tjwtx2ckqi2x) 95

[Interpretation der Ergebnisse](#_34g0dwd) 98

[Zusammenfassung und Ausblick](#_ln0yz2tp9c2s) 98

[**Webseite**](#_61k4w4tgjcxr) **98**

[Methodische Ermittlung](#_7cq58gn8mqtq) 98

[Anforderungsanalyse und Konzeption](#_4i0d7um3jxsa) 99

[Design (Entwicklung) und Implementierung](#_1rvwp1q) 99

[Anwendungsfälle](#_c2adn4tgoojw) 99

[EER und UML Domänenmodell](#_4bvk7pj) 102

[Funktionale und nichtfunktionale Anforderungen](#_6kech3dean0g) 105

[Systemarchitektur](#_2r0uhxc) 109

[Design-Layouts](#_1664s55) 109

[Evaluation](#_25b2l0r) 113

[Versuchsaufbau und Versuchsdurchführung](#_kgcv8k) 113

[Ergebnisse](#_96edfayzissj) 114

[Interpretation der Ergebnisse](#_34g0dwd) 115

[Zusammenfassung und Ausblick](#_94dr23wzxe0d) 116

[**Backend**](#_tdl71cfoe6qf) **118**

[Methodische Ermittlung](#_k5yy5sbaa021) 118

[Anforderungsanalyse und Konzeption](#_rf73heczh3dn) 120

[Design (Entwicklung) und Implementierung](#_1rvwp1q) 121

[Anwendungsfälle](#_l5lq6rrohmiw) 130

[EER und UML Domänenmodell](#_4bvk7pj) 133

[Funktionale und nichtfunktionale Anforderungen](#_d2j4cxdjosyv) 135

[Systemarchitektur](#_2r0uhxc) 137

[Zusammenfassung und Ausblick](#_out8yt2jw82p) 141

[**NextSteps**](#_e6wndo1ku8a) **141**

[Projektumsetzung](#_4f1mdlm) 141

[Konzept 1: Die Bestellung von der Kantine mittels mobiler Applikation](#_n58jtv6wk0zl) 141

[Einleitung](#_19c6y18) 141

[Problemstellung](#_3tbugp1) 141

[Stand der Technik](#_28h4qwu) 142

[Zielsetzung](#_nmf14n) 142

[Aufbau des Projekthandbuches](#_37m2jsg) 142

[Anforderungsanalyse und Konzeption](#_1mrcu09) 142

[Design (Entwicklung) und Implementierung](#_46r0co2) 142

[Anwendungsfälle](#_2lwamvv) 142

[EER und UML Domänenmodell](#_111kx3o) 143

[Funktionale und nichtfunktionale Anforderungen](#_4k668n3) 144

[Systemarchitektur](#_2zbgiuw) 145

[Design-Layouts](#_3ygebqi) 145

[Evaluation](#_sqyw64) 145

[Zusammenfassung und Ausblick](#_3cqmetx) 146

[Konzept 2: Der Eintritt zur Universität mittels mobiler Applikation (Batuhan Özsoy)](#_1rvwp1q) 147

[Einleitung](#_4bvk7pj) 147

[Problemstellung](#_l23ul47j7dfy) 147

[Zielsetzung](#_n9jfcvs7kork) 147

[Aufbau des Projekthandbuches](#_vevirj3ya81) 147

[Methodische Ermittlung](#_2r0uhxc) 147

[Methoden und Werkzeuge](#_c616e9d1wo04) 147

[Anforderungsanalyse und Konzeption](#_1664s55) 147

[Design (Entwicklung) und Implementierung](#_3q5sasy) 148

[Anwendungsfälle](#_y2o9z95kpoch) 148

[EER und UML Domänenmodell](#_r8s9cwilmxd5) 149

[Funktionale und nichtfunktionale Anforderungen](#_ezahxs6xjefz) 150

[Systemarchitektur](#_gcj87quug4ux) 150

[Design-Layouts](#_c0k138xavssa) 151

[Evaluation](#_kgcv8k) 151

[Zusammenfassung und Ausblick](#_34g0dwd) 152

[Konzept 3: Der Eintritt zu den Laboren mittels mobiler Applikation (Sebiha Gökçem Hakbilen)](#_1jlao46) 152

[Einleitung](#_43ky6rz) 152

[Problemstellung](#_yyus6bnvtv36) 152

[Stand der Technik](#_156r6p9stvqb) 153

[Zielsetzung](#_6bk10aogbljw) 153

[Methodische Ermittlung](#_xvir7l) 154

[Methoden und Werkzeuge](#_eemzybrjs0kv) 154

[Anforderungsanalyse und Konzeption](#_3hv69ve) 154

[Design (Entwicklung) und Implementierung](#_1x0gk37) 154

[Anwendungsfälle](#_pt1xsy7el12v) 154

[EER und UML Domänenmodell](#_nv74nuecs9ou) 156

[Funktionale und nichtfunktionale Anforderungen](#_xjfhpj42k0ae) 156

[Systemarchitektur](#_ifjuvlqqcq79) 157

[Design-Layouts](#_mxb0k8ey2l10) 157

[Evaluation](#_1302m92) 162

[Zusammenfassung und Ausblick](#_3mzq4wv) 162

[Das Ausleihen von Büchern aus der Bibliothek  (Muhammed GÖK)](#_2250f4o) 162

[Einleitung](#_haapch) 162

[Problemstellung](#_cy188ajz8e3g) 162

[Stand der Technik](#_kih15zlgloiu) 163

[Zielsetzung](#_trdhnqq62dxo) 164

[Anforderungsanalyse und Konzeption](#_319y80a) 164

[Design (Entwicklung) und Implementierung](#_1gf8i83) 165

[Anwendungsfälle](#_42d24ucky1m) 165

[EER und UML Domänenmodell](#_v0wyplg39dn9) 166

[Funktionale und nichtfunktionale Anforderungen](#_dfn0z951fqvj) 167

[Systemarchitektur](#_h7vry9z7bsym) 167

[Design-Layouts](#_g62aa4s210ak) 168

[Evaluation](#_40ew0vw) 169

[Zusammenfassung und Ausblick](#_2fk6b3p) 169

[Konzept 5: TDU Zielführung- und Informationssystem (Yusufhan Yılmaz)](#_upglbi) 170

[Einleitung](#_3ep43zb) 170

[Problemstellung](#_hr735kwme5yi) 170

[Stand der Technik](#_3wousxbr2xxx) 170

[Zielsetzung](#_vlsxjmt4zlx) 170

[Methodische Ermittlung](#_1tuee74) 170

[Methoden und Werkzeuge](#_k4la8ehgunts) 170

[Anforderungsanalyse und Konzeption](#_4du1wux) 170

[**Design (Entwicklung) und Implementierung**](#_2szc72q) **171**

[Anwendungsfälle](#_hpove2y2pvcz) 171

[EER und UML Domänenmodell](#_2mw9nka58fa3) 172

[Funktionale und nichtfunktionale Anforderungen](#_ymnjkqz9bbgb) 172

[Systemarchitektur](#_g8libvfznhfx) 173

[Design-Layouts](#_5919i23yjr96) 173

| **Änderungsverzeichnis** | | | |
| --- | --- | --- | --- |
| **Versions-Nummer[[1]](#footnote-0)** | **Datum** | **Änderung** | **Ersteller** |
| 1.0 | 02.03.2019 | Anfertigung des Projekthandbuches | ZT |
| 1.1 | 05.03.2019 | Füllung der Projekthandbuches | BK,GH,MG,YY,BÖ,CA, |
| 1.2 | 15.03.2019 | Füllung der Projekthandbuches | BK,GH,MG,YY,BÖ |
| 1.3 | 17.03.2019 | 1.1 - 1.5 | MMA |
| 1.4 | 24.03.2019 | 1.6 | SZB |
| 1.5 | 15.04.2019 | Füllung der Projekthandbuches | BK,GH,MG,YY,BÖ |
| 1.6 | 20.04.2019 | 1.7 - 1.11 | SZB, AA |
| 1.7 | 04.05.2019 | Anfertigung der Teilen (1.1 - 1.13) | AB |
| 1.8 | 29.04.2019 | Füllung der Projekthandbuches | BK,GH,MG,YY,BÖ |
| 1.9 | 05.05.2019 | Anfetigung der Teilen 1.5, 1.7 | ÖCT |
| 1.10 | 14.05.2019 | Fertigstellung des Projekthandbuches | Android-und  IOS Gruppe |
| 1.11 | 14.05.2019 | Anfertigung des Teils 2.5.4 | AB |
| 1.12 | 15.052019 | Füllung der Projekthandbuches | BK,GH,MG,YY,BÖ |
| 1.13 | 15.05.2019 | End Dokumentation | SZB, AA, OK |
| 1.14 | 15.05.2019 | Anfertigung der Teilen 2.5.3, 2.5.4 | OBK |
| 1.15 | 15.05.2019 | Anfertigung der Teilen 2.5.1.3, 2.5.4 | KCÇ |
| 1.16 | 15.05.2019 | Anfertigung der Teilen 2.5.1.2, 2.5.1, 2.5.4.1 | OMD |
| 1.17 | 15.05.2019 | Anfertigung der Teilen 2.5.2, (2.5.4.2 – 2.5.4.4) | ÖCT |
| 1.18 | 15.05.2019 | Anfertigung der Teilen 2.5.4, 2.5.5 | ARÖ |
| 1.19 | 15.05.2019 | Anfertigung der Teilen 2.5.1.1 | Backend Team |
| 2.0 | 17.05.2019 | Vereinigung der Projekthandbücher | DMT, Mİ |

# Tabelle 1: ÄnderungsverzeichnistabelleVerzeichnis der Abkürzungen

| **Abkürzung** | **Bedeutung** |
| --- | --- |
| MMK | Mensch-Maschine Kommunikation |
| SUS | System Usability Scale |
| AP | Arbeitspaket |
| AG | Arbeitsgruppe |
| PSP | Projektstrukturplan |
| EER | Enhanced Entity Relationship |
| UML | Unified Modeling Language |
| TDU | Türkisch-Deutsche Universität |
| TAÜ | Türk-Alman Üniversitesi |
| TMS | Technik Mechatronischer Systeme |
| INF | Informatik |
| SWE | Software Engineering |
| ZT | Zeynep Tuncer |
| MMI | Mensch-Maschine Interaktion |
| SQL | Structured Query Language |
| App | Applikation |
| AP | Arbeitspaket |
| AG | Arbeitsgruppe |
| DB | Datenbank |
| SSL | Secure Socket Layer |
| JPA | Java Persistence API |
| API | Application Programing Interface |
| FTP | File Transfer Protocol |
| SSH | Secure Shell |
| HTTP | Hyper Text Transfer Protocol |
| CSS | Cascading Style Sheet |
| HTML | Hyper Text Markup Language |
| PHP | Hypertext Preprocessor |
| DMT | Deniz Mert Tecimer |
| Mİ | Melih İlerialkan |
| MMA | Muhammed Mahi Aydın |
| SZB | Sema Zeynep Bulut |
| AA | Alperen Asan |
| AB | Aybüke Bayramiç |
| ÖCT | Ömer Cem Turan |
| OK | Orkun Kuyucu |
| OBK | Onur Burak Kılıç |
| KCÇ | Kartal Can Çelikkanat |
| OMD | Onur Mert Doğan |
| ARÖ | Ayşe Rabia Özbek |
| BK | Berk KARAAĞAÇ |
| GH | Gökçem HAKBİLEN |
| MG | Muhammed GÖK |
| YY | Yusufhan YILMAZ |
| BÖ | Batuhan ÖZSOY |
| CA | Cengiz ASLAN |

# Tabelle 2: Verzeichniss der Abkürzungen

# **Verzeichnis der Abbildungen**

Wird automatisch angelegt, wenn Sie mit der Funktion „Abbildung“ arbeiten

| List Of Images   * [Abbildung 1.](#cqwamp92wn8m) * [Abbildung 2.](#vwdmvv1ts4m0) * [Abbildung 3.](#2dcl2mriqzr) * [Abbildung 4.](#3gl1501mmeda) * [Abbildung 5.](#9q82txa032sh) * [Abbildung 6.](#4mctsrcw0a3) * [Abbildung 7.](#f73vrngc6ajb) * [Abbildung 8.](#vk0z08ocont3) * [Abbildung 9.](#icx7mgxny4zd) * [Abbildung 10.](#otocfecy110a) * [Abbildung 11.](#obhe78yqg7o5) * [Abbildung 12.](#y4lk1tjofsp6) * [Abbildung 13.](#qykjfbeffti4) * [Abbildung 14.](#f7by5y2sdruk) * [Abbildung 15.](#g8wazfrg0puk) * [Abbildung 16.](#3xxqo1vl3l4u) * [Abbildung 17.](#1787q6s7wpr) * [Abbildung 18.](#2ri1n4c0w8sr) * [Abbildung 19.](#p08bzpoe0oqd) * [Abbildung 20.](#62cwk7z5psz8) * [Abbildung 21.](#4b3zrpedx0gz) * [Abbildung 28.](#stfnidnv52lh) * [Abbildung 28.](#stfnidnv52lh) * [Abbildung 28.](#stfnidnv52lh) * [Abbildung 28.](#stfnidnv52lh) * [Abbildung 28.](#stfnidnv52lh) * [Abbildung 28.](#stfnidnv52lh) * [Abbildung 29.](#sezt2qe6gtgo) * [Abbildung 30.](#o3i6zmyx4fti) * [Abbildung 31.](#nnhh5coiat93) * [Abbildung 32.](#ah3exuc1978a) * [Abbildung 33.](#ohzjgv37p29i) * [Abbildung 35.](#alco6q274kju) * [Abbildung 36.](#j1axdlw47vkg) * [Abbildung 37.](#66brs6tajhr) * [Abbildung 38.](#govkk6bcsakm) * [Abbildung 39.](#bs5t37m1wxuf) * [Abbildung 40.](#xqpudtra3tit) * [Abbildung 41.](#ftad4b8cnh65) * [Abbildung 42.](#d9m5tq40qew) * [Abbildung 43.](#pad4oydlso9j) * [Abbildung 44.](#e78sp9lbxud9) * [Abbildung 46.](#utfkyr18dn5h) * [Abbildung 46.](#utfkyr18dn5h) * [Abbildung 47.](#kz9v36o4ms6k) * [Abbildung 48.](#9ex2m4c3nh5j) * [Abbildung 50.](#7sjrlxm5bokq) * [Abbildung 50.](#7sjrlxm5bokq) * [Abbildung 53.](#7jrj39fpellu) * [Abbildung 53.](#7jrj39fpellu) * [Abbildung 53.](#7jrj39fpellu) * [Abbildung 56.](#ouqxs5up0un5) * [Abbildung 56.](#ouqxs5up0un5) * [Abbildung 56.](#ouqxs5up0un5) * [Abbildung 59.](#fyyd7fupszy9) * [Abbildung 59.](#fyyd7fupszy9) * [Abbildung 59.](#fyyd7fupszy9) * [Abbildung 61.](#glym3k9lgxhm) * [Abbildung 61.](#glym3k9lgxhm) * [Abbildung 62.](#ye7dapolng17) * [Abbildung 64.](#dizooz6o6g0q) * [Abbildung 64.](#dizooz6o6g0q) * [Abbildung 65.](#9h2sa7gg5eju) * [Abbildung 73.](#rk01y9lz8ncr) * [Abbildung 73.](#rk01y9lz8ncr) * [Abbildung 73.](#rk01y9lz8ncr) * [Abbildung 73.](#rk01y9lz8ncr) * [Abbildung 73.](#rk01y9lz8ncr) * [Abbildung 73.](#rk01y9lz8ncr) * [Abbildung 73.](#rk01y9lz8ncr) * [Abbildung 73.](#rk01y9lz8ncr) * [Abbildung 75.](#1uxdgffzd1jb) * [Abbildung 75.](#1uxdgffzd1jb) * [Abbildung 76.](#v137ihbft3ht) * [Abbildung 77.](#thb8skn01qxn) * [Abbildung 78.](#nx5giqsgjszy) * [Abbildung 79.](#xijsno8gtpu2) * [Abbildung 80.](#opxioootjwcc) * [Abbildung 81.](#ln2pg46cypaw) * [Abbildung 82.](#ofuif1ccpgdp) * [Abbildung 83.](#dzfdvgypfpfp) * [Abbildung 84.](#h1hxyjg5q9rc) * [Abbildung 85.](#xplvtdmp3bzq) * [Abbildung 86.](#r7ggwekl7hyd) * [Abbildung 87.](#h17z0p8ohu54) * [Abbildung 88.](#etb8uymlnqq1) * [Abbildung 89.](#nflpkwqoy7wi) * [Abbildung 90.](#ns5lwzqgt6kr) * [Abbildung 91.](#lnh77zfgdxmt) * [Abbildung 92.](#2ul2voo8r9co) * [Abbildung 93.](#qbpnq7hdds69) * [Abbildung 94.](#cou9y5dapv69) * [Abbildung 95.](#97co9librnev) * [Abbildung 96.](#ke9ec9cdg5hx) * [Abbildung 97.](#btcfdy9llpyk) * [Abbildung 98.](#f2jrdo2xnpe) * [Abbildung 99.](#ijniadnwrzn6) * [Abbildung 100.](#8bdt00smrb9m) * [Abbildung 101.](#2n9n42cbrmvs) * [Abbildung 102.](#yvol64lrh3z0) * [Abbildung 103.](#88no8ixgbc07) * [Abbildung 104.](#rl3qp3ia4yny) * [Abbildung 105.](#kiv80izho4wl) * [Abbildung 106.](#3ptj7jytvli8) * [Abbildung 107.](#14ht24hjgskn) * [Abbildung 108.](#tewe20ol2ti0) * [Abbildung 109.](#3cc6u3nzghc7) * [Abbildung 110.](#lg738ojnqdw4) * [Abbildung 111.](#wpn6pk2d3mys) * [Abbildung 112.](#9mab25w3xt5v) * [Abbildung 113.](#ez5d4ci7u6w) * [Abbildung 114.](#2u0duc85z1gf) * [Abbildung 115.](#bn2tgof2nam2) * [Abbildung 116.](#mph1xro0qj9v) * [Abbildung 117.](#aq4ktw1heg5o) * [Abbildung 118.](#oj6tyh3zoa03) * [Abbildung 119.](#jeeb7daec935) * [Abbildung 120.](#e90m1cb0w2h7) * [Abbildung 121.](#9xtiqbcsazuh) * [Abbildung 122.](#avqchcvlymdv) * [Abbildung 123.](#wdeym0o40tf6) * [Abbildung 124.](#r4gvaiapw9bf) * [Abbildung 125.](#9wf887mlje3d) * [Abbildung 126.](#2rdieer2exmf) * [Abbildung 127.](#eu8oiprw00gz) * [Abbildung 128.](#bgqv2hgtw14b) * [Abbildung 129.](#ycf1812spd8w) * [Abbildung 130.](#cn6wsvjryoyi) * [Abbildung 131.](#nnzeaw1p2ec9) * [Abbildung 132.](#2j57u4abob8p) * [Abbildung 133.](#xw4umoi4ityy) * [Abbildung 134.](#fzv2qqrpq5ft) * [Abbildung 135.](#yg72qstatdk7) * [Abbildung 136.](#kauiu3b24cnq) * [Abbildung 137.](#ns1w1ffw6jhg) * [Abbildung 138.](#aa8rejf3ls26) * [Abbildung 139.](#9o7qjlp4odcq) * [Abbildung 140.](#ec9kuw6yvtrt) * [Abbildung 141.](#fuzt7h5stksc) * [Abbildung 142.](#ud6p8djzczqp) * [Abbildung 143.](#tvb6o6y49uy5) * [Abbildung 144.](#5tu4csytio0v) * [Abbildung 145.](#tx84z1lwoadw) * [Abbildung 146.](#14t56w8q9m4g) * [Abbildung 147.](#cx3b3uao9ibc) * [Abbildung 148.](#nlvinw3q59oj) * [Abbildung 149.](#50h4oz5jkdli) * [Abbildung 150.](#uwb1zd5kbakw) |
| --- |

## Verzeichnis der Tabellen

Wird automatisch angelegt, wenn Sie mit der Funktion „Tabellen“ arbeiten

Tabelle 1: ÄnderungsverzeichnistabelleVerzeichnis der Abkürzungen

Tabelle 2: Verzeichniss der Abkürzungen

Tabelle 3: Ansprechpartner

Tabelle 4: Projektauftrag

Tabelle 5: Projektzielplan

Tabelle 6: Zeitlicher Projektkontext

Tabelle 7: Sachlicher Projektkontext

Tabelle 8: Sozialer Kontext

Tabelle 9: Projektorganisation

Tabelle 10: Projektkommunikationsplan

Tabelle 11: Arbeitspezifikation von Android-Gruppe

Tabelle 12: Arbeitspezifikation von IOS-Gruppe

Tabelle 13: Arbeitspezifikation von Backend-Gruppe

Tabelle 14: Arbeitspezifikation von Webseite-Gruppe

Tabelle 15: Arbeitspezifikation von Webseite-Gruppe

Tabelle 16: Arbeitspezifikation von Nextsteps-Gruppe

Tabelle 17: Meilensteine

Tabelle 18: Projektmilensteine

Tabelle 19: Projektterminplan

Tabelle 20: Projektkostenplan

Tabelle 21: Projektrisiken

Tabelle 22: Projektfortschrittsbericht

Tabelle 23: Projektumsetzung

Tabelle 24: Funktionale Anforderungen

Tabelle 25: Nichtfunktionale Anforderungen

Tabelle 26: Projektabschlussbericht (Android)

Tabelle 27: Projektabschlussbericht (Backend)

Tabelle 28: Projektabschlussbericht (IOS)

Tabelle 29:

# Projektplanung

## Ansprechpartner

|  | **Ansprechpartner** | | | |
| --- | --- | --- | --- | --- |
| **Nr.** | **Arbeitspaket** | **Projektrolle** | **Vorname, Nachname** | **Kontaktdaten** |
| 1 | AP1 | Lead | Ayşe Rabia Özbek | ozbekra@gmail.com  +905537233481 |
| 2 | AP1 | Member | Onur Mert Doğan | onurmertdogan@gmail.com  +90532557504 |
| 3 | AP1 | Member | Kartal Can Çelikkanat | kartal\_can\_94@hotmail.com  +905348423939 |
| 4 | AP2 | Lead | Ali Egin | eginali@outlook.com +90 537 831 55 62 |
| 5 | AP2 | Member | İsmail Çağlar Bayram | caglarmeka@gmail.com +90 534 953 52 00 |
| 6 | AP2 | Member | Kaan Göktaş | kaagoktas@gmail.com  +90 534 465 62 88 |
| 7 | AP3 | Lead | Mustafa Burak Çakmak | mustafaburak95@yahoo.com +905426278071 |
| 8 | AP3 | Member | Emirhan Özkan | em-ir-han@windowslive.com +905531464720 |
| 9 | AP3 | Member | Ömer Özgür Oyman | ozgroyman@gmail.com +905349493363 |
| 10 | AP4 | Lead | Berk Karaağaç | berkkaraagac25@gmail.com |
| 11 | AP4 | Member | Cengiz Furkan Aslan |  |
| 12 | AP4 | Member | Batuhan Özsoy | batozsoy@gmail.com |
| 13 | AP5 | Lead | Sebiha Gökçem Hakbilen | hakbilengokcem@gmail.com |
| 14 | AP5 | Member | Yusufhan Yılmaz | yyılmaz96@gmail.com |
| 15 | AP5 | Member | Muhammed Gök | muhammed.gok1703@gmail.com |
| 16 | AP6 | Lead | Mert Özgür |  |
| 17 | AP6 | Member | Mert Cenk Sabancı |  |
| 18 | AP6 | Member | Yiğit Mesci |  |
| 19 | AP7 | Lead | Muhammed Mahi Aydın | mahiaydin96@gmail.com |
| 20 | AP7 | Member | Alperen Asan | alperenasan@gmail.com |
| 21 | AP7 | Member | Sema Zeynep Bulut | semazeynepbulutt@hotmail.com |
| 22 | AP7 | Member | Orkun Kuyucu | orkunkuyucu@hotmail.com |
| 23 | AP8 | Lead | Uğur Güneş | ugurguenes@web.de +90 505 122 43 92 |
| 24 | AP8 | Member | Minel Küpçü | minelkupcu@gmail.com +90 553 738 99 82 |
| 25 | AP8 | Member | Beste Şanlı | bestesanli39@hotmail.com +90 541 484 60 39 |
| 26 | AP9 | Lead | Tayyip Ensar Özkaya | te.ozkaya@gmail.com |
| 27 | AP9 | Member | Melih İlerialkan | e160503104@stud.tau.edu.tr |
| 28 | AP9 | Member | Ulaş Algül | ulasalgul@gmail.com |
| 29 | AP10 | Lead | Sevcan Avcı | sevcan\_avci@outlook.de  +90 551 981  6267 |
| 30 | AP10 | Member | Rumeysa Çeşmeci | rumeysa.cesme  ci@gmail.com  +90 541 549 56  43 |
| 31 | AP10 | Member | İhsan Oğuz Merdan | iomerdan@gmai  l.com  +90 536 432 0368 |
| 32 | AP11 | Lead | Deniz Mert Tecimer | deniz.mert.tecimer@gmail.com |
| 33 | AP11 | Member | Zeynep Sena Karabacak | e160503110@stud.tau.edu.tr |
| 34 | AP11 | Member | Fatih Özoğlu | fatihozoglu61@gmail.com |
| 35 | AP12 | Lead | Ömer Cem Turan | omercemturan@gmail.com  +905317447010 |
| 36 | AP12 | Member | Aybüke Bayramiç | aybuke-52@hotmail.com  +905382736350 |
| 37 | AP12 | Member | Onur Burak Kılıç | burakkilic95@windowslive.com  +905514023935 |
| 38 | AP13 | Lead | Alp Akyüz | bongutcha@gm  ail.com +90 545 351  6008 |
| 39 | AP13 | Member | Gamze Fıçı | e160503117@st  ud.tau.edu.tr +90 535 838  7885 |
| 40 | AP13 | Member | Merve Barıtcı | dominotasi@hot  mail.com +90 506 585  1531 |
| 41 | AP13 | Member | Beyza Patlican | beyza786@outl  ook.com +90 554 279  8463 |
| 42 | AP14 | Lead | Sedat Hatip | hatipsedat@gmail.com |
| 43 | AP14 | Member | Merve Sarı | mervesarinin@gmail.com |
| 44 | AP14 | Member | Eda Atmaca | e140501130@stud.tau.edu.tr |
| 45 | AP15 | Lead | Muhammed Zahid Bozkuş | mzbozkus@hotmail.com |
| 46 | AP15 | Member | Hakan Öznur | e160503125@stud.tau.edu.tr |
| 47 | AP15 | Member | Ahmet Çakırer | e130501123@stud.tau.edu.tr |
| 48 | AP16 | Lead | Nusret Özateş | nozatespc@gm  ail.com  +90 546 219 23  32 |
| 49 | AP16 | Member | Erinç Kula | erinckula95@g  mail.com +90 506 334 65  65 |
| 50 | AP16 | Member | Deha Derviş Bayram | deha388@gmail  .com +90 541 234  2934 |

Tabelle 3: Ansprechpartner

## Projektauftrag

| **Projektauftrag** | |
| --- | --- |
| Projektziel (Output):  Erstellung eines effektiven und innovativen Online-Zahlungsystem | Nicht-Ziel:  Gästkonto Monatliches Abonnments Online-Banking / Geld aufladen |
| Projektnutzen (Outcome):  Eine Anwendung(App) für Android und IOS  Eine Webseite für Admins Ein Datenbanksystem | |
| Projektauftraggeber/in: Burcu Yıldız | Projektleiter/in: Zeynep Tuncer |
| Projektteammitglieder:   * Ayşe Rabia Özbek * Kartal Can Çelikkanat * Ali Egin * İsmail Çağlar Bayram * Kaan Göktaş * Mustafa Burak Çakmak * Emirhan Özkan * Ömer Özgür Oyman * Mert Özgür * Mert Cenk Sabancı * Yiğit Mesci * Muhammed Mahi Aydın * Alperen Asan * Sema Zeynep Bulut * Orkun Kuyucu * Uğur Güneş * Minel Küpçü * Beste Şanlı * Tayyip Ensar Özkaya * Melih İlerialkan * Ulaş Algül * Sevcan Avcı * Rumeysa Çeşmeci * İhsan Oğuz Merdan * Deniz Mert Tecimer * Zeynep Sena Karabacak * Fatih Özoğlu * Ömer Cem Turan * Aybüke Bayramiç * Onur Burak Kılıç * Alp Akyüz * Gamze Fıçı * Merve Barıtcı * Beyza Patlican * Sedat Hatip * Merve Sarı * Eda Atmaca * Muhammed Zahid Bozkuş * Hakan Öznur * Ahmet Çakırer * Nusret Özateş * Erinç Kula * Deha Derviş Bayram | Sonstige Beteiligte:  - |
| Hauptaufgaben:   * einfach bedienbare Benutzeroberfläche auf Android mit den zugehörigen Funktionen implementieren * Ein Benutzeroberfläche für Barcode-Scanner * Backend und Server aufsetzen * Eine Webseite für Admins erstellen | Meilensteine:   * 04.03.2019: Projektstart * 19.03.2019: Anforderungsanalyse * 09.04.2019: Systementwurf, Design und Prototyp * 30.04.2019: Implementierung * 10.05.2019: Tests, Evaluierungen, Anpassungen * 17.05.2019: Dokumentation |
| Projektstartereignis: Brainstorming | Projektstarttermin: 04.03.2019 |
| Projektendereignis: Evalierung | Projektendtermin: 17.05.2019 |
| Projektkosten:  102h Arbeitsaufwand  Serverkosten monatlich ca. 10$ | Projektrisiken:  Kompabilität  Leitung  Technische Störungen  Cybersicherheit  Benutzerfreundlichkeit  Qualitätsrisiken  Fehler bei der Kommunikation |
| ……………………………………………..  <Datum>, Unterschrift Projektleiter/in | ……………………………………………..  <Datum>, Unterschrift Auftraggeber |

Tabelle 4: Projektauftrag

## Projektzieleplan

| **Projektzieleplan** | | |
| --- | --- | --- |
| **Zielart** | **Projektziele** | **Adaptierte Ziele per** **<Datum>** |
| **Hauptziel (Output):** | Zusammenführen mehrerer Funktionen in einer mobilen Anwendung | 17.05.2019 - Ende des Projekts |
| **Teilziele:** | Anmelden:  - Anmelden  - Passwort vergessen  - First time login  Bezahlen:  - Bezahlung durch QR-Code  Geldübertragung:  - Geld an eine Person schicken  Spenden:  - Mahlzeit oder Shuttlefahrt spenden    Empfehlung:  - Eine Hilfsbedürftige Person empfehlen  Profil:  - Guthaben  - Feedback  - Einstellungen  Feedback:  - Feedback an Mensa oder Shuttle  geben    Einstellungen:  - Passwort verändern  - Sprache  - Logout  Erstellung des Http-Servers:  - Programmierung durch Java Spring  - Integration von Rest API  -Sicherheitsgewährleistung bei der Kommunikation  DB modellieren inkl. Logging:  - Erstellung eine Datenbank durch MySQL  - Darstellung der Diagramm von -Relationen von Tabellen in der Datenbank  -Speicherung von benötigte Daten in Server  -Sicherung, dass die Daten immer erreichbar sind  Überprüfung von Gültigkeit von Anforderungen des folgenden Clients:  -Mobile-Geräte (IOS, Android)  -Webseiten (PHP)  -Barcode-Scanner  Implementierung einer Benutzeroberfläche für Barcode-Scanner  Integration mit der Kantine  Eintritt zu TDU und Laboren  Integration mit der Bibliothek  Zielführung- und Informationssystem  Koordination der Informationen der Administratoren:   * Administrator hinzufügen * Bearbeiten / Änderungen der Informationen der vorhandenen Administratoren * Administratoren löschen * Neue Administratorgruppe erstellen   Webseite Design:   * CSS   Webseite Implementierung:   * PHP * HTML * JavaScript   Verbindung mit dem Backend:   * http-Protokoll * JSON   Person-Informationen bearbeiten:   * Hinzufügen * Löschen * Bearbeiten   Webseite-Erstellung:   * Eine visuelle und benutzerfreundliche Benutzeroberfläche | 05.05.2019  01.04.2019  05.04.2019  01.05.2019 |
| **Nicht-Ziel:** | Gästkonto Monatliches Abonnments Online-Banking / Geld aufladen |  |
| **Projektnutzen (Outcome):** | Optimierung und Vereinfachung der Mensa des Shuttles und dem Bezahlsystem und Helfen von Hilfsbedürftigen.  Kommunikation zwischen Web-Client/Mobile Devices/Barcodescanner und Web-Server(DB).  Hinzufügen, bearbeiten und löschen der Information der TDU-Studenten und Mitarbeitern in das System.  erwaltung und Kontrolle spezifischer (Shuttle, Mensa, Bibliothek usw.) Administratorengruppe | 05.05.2019 - Ende des Projekts |

Tabelle 5: Projektzielplan

## Projektkontext

| **Ausgangssituation und Problembeschreibung** |
| --- |
| Bisher musste man in der Mensa und bei Shuttle-Fahrten mit zwei unterschiedlichen Karten bezahlen. Zumal man sie häufig zu Hause vergessen hat, ist es auch sehr lästig, ständig zwei Karten mit sich zu tragen. Außerdem konnten Studenten keine finanzielle Hilfe anfordern ohne eventuell unter sozialem Druck zu leiden. |

| **Zeitlicher Projektkontext** | |
| --- | --- |
| **Vorprojektphase** | **Nachprojektphase** |
| * Es gibt keine Möglichkeit sein Geld für die Mensa aufzuladen, außer die Automaten. Diese sind jedoch weit entfernt und es gibt wenige im gesamten Haus, weshalb es weniger benutzt wird. * Für das Shuttle gibt es nur eine Stelle innerhalb des Campuses * Das Geld wird in der Mensa an einem Automaten per Karte oder mit Bargeld bezahlt * Im Shuttle wird mit einer extra Shuttle Karte das Geld abgebucht * Es ist ungewiss, wie viel Guthaben einem zur Verfügung stehen. * Es gibt keine Möglichkeit an Hilfsbedürftige anonym zu spenden * Es kann kein Feedback abgegeben werden * Sowohl die Mensa als auch das Shuttlesystem läuft über voneinander abhängigen Karten. * Keine Lagerung für Mensakartebearbeitung * Keine kommunikationsnetz * Kein Sicherheit * Keine Identifikation * Es gibt keine Webseite für die Kontrolle und Verwaltung der Administratorengruppe | * Guthaben per iOS-Android App von überall aus aufladen, sowohl für die Mensa als auch für das Shuttle * Durch eine Applikation in der Mensa und im Shuttle bezahlen * Guthaben kann jederzeit abgefragt werden * Anonyme Spende Möglichkeit an eine Person * Feedback Abgabe für Mensa und Shuttle * Personen können anonym empfohlen werden * Alles kompakt innerhalb einer Applikation * Lagerung durch Mysql DB * Effektive Kommunikation * Dienst für viele Personen * Versicherung von den Login * Es gibt das System, das die Administratorengruppe kontrolliert und verwaltet. Darüber hinaus werden die Schulbehörden die Kontrolle und Verwaltung übernehmen. |

Tabelle 6: Zeitlicher Projektkontext

| **Sachlicher Projektkontext** | |
| --- | --- |
| **Kontext** | **Erforderliche Maßnahmen** |
| Erstellung einer Webseite | * PHP lernen * HTML lernen * JavaScript lernen |
| Design der Webseite | * CSS lernen |
| Einheitlichkeit | * Mit anderen Arbeitspaketen der Webseite Programmierung Sprache absprechen. |
| Verwendbarkeit der Webseite | * Entwurf Umfrage mit einer Benutzergruppe |
| Android-Studio | * Java lernen * Android Studio lernen |
| Anbindung an Datenbank | * MySQL lernen * mit den anderen Arbeitsgruppen absprechen |
| Einheitlichkeit | * mit anderen Arbeitsgruppen der Android. und iOS Programmierung absprechen |
| Projektgruppen:  Android Gruppe  IOS Gruppe  Barcode Scanner Gruppe | Einheitlich Kommunikationsnetz zwischen Client und Server (Json) |
| Projektgruppe:  Webseite Gruppe | Benutzen von Bestimmtes QR-Code per Personen für Barcode Programmierung |

Tabelle 7: Sachlicher Projektkontext

| **Sozialer Kontext (Projektumweltanalyse)** | | | |
| --- | --- | --- | --- |
| **Anspruchsgruppe** | **Potenziale / Chancen** | **Konflikte / Risiken** | **Maßnahmen** |
| Studenten | Vereinfachte Nutzung von der Mensa und dem Shuttle, außerdem Erstellung einer Plattform für finanizielle Unterstützung | -Kein Akku  -Server-/Applikationsprobleme  -Stromausfall  -Cybersicherheit | -Powerbank  -Tägliche Kontrollen der Server und der Datenbanken  -Elektrogenerator |
| Dozenten | Vereinfachte Nutzung von der Mensa und dem Shuttle, außerdem Erstellung einer Plattform für finanizielle Unterstützung | -Kein Akku  -Server-/Applikationsprobleme  -Stromausfall  -Cybersicherheit | -Powerbank  -Tägliche Kontrollen der Server und der Datenbanken  -Elektrogenerator |
| Mitarbeiter | Vereinfachte Nutzung von der Mensa und dem Shuttle, außerdem Erstellung einer Plattform für finanizielle Unterstützung | -Kein genügendes wissen über die Nutzung von Technologien (inkompatibles Design)  -Kein Akku  -Server-/ Applikationsprobleme  -Stromausfall  -Cybersicherheit | -Crashkurs der Bedienung  -Hilfe am Stand  -Powerbank  -Tägliche Kontrollen der Server und der Datenbanken  -Elektrogenerator |
| Barcode Scanner Gruppe | Qr-Code Scanner | Server absturz | Mehr testen |
| Android Gruppe  IOS Gruppe | Mobile Bezahlung und Geld laden  Empfehlen  Feedback | Server absturz  Sicherheitsproblem  Design Probleme | Mehr testen und Simulationen  Gute Kommunikation mit alle Arbeitsgruppen |
| Mensa Administratoren | * Verwaltung der Informationen der Administratoren,die in der Mensa arbeiten | * Die Kontaktinformationen können verwirrend sein | * Wenn Sie Personen nach bestimmten Merkmalen trennen, bsw Benutzername ist für alle Personen privat. |

Tabelle 8: Sozialer Kontext

## Projektorganisation

| **Projektorganisation** | | |
| --- | --- | --- |
| **Projektrolle** | **Rollenbeschreibung[[2]](#footnote-1)** | **Name** |
| Projektauftraggeber | * Dozent an der Universität | Burcu Yıldız |
| Projektmanager | * Koordination | Zeynep Tuncer |
| Projektteammitglieder | * Lead - Koordination unterhalb der Backend-Gruppe(AG1 und AG12) | Ömer Cem Turan |
| Projektmitarbeiter | * Mitarbeiter von Backend-Gruppe | Ayşe Rabia Özbek |
| Projektmitarbeiter | * Mitarbeiter von Backend-Gruppe | Aybüke Bayramiç |
| Projektmitarbeiter | * Mitarbeiter von Backend-Gruppe | Onur Burak Kılıç |
| Projektmitarbeiter | * Mitarbeiter von Backend-Gruppe | Onur Mert Doğan |
| Projektmitarbeiter | * Mitarbeiter von Backend-Gruppe | Kartal Can Çelikkanat |
| Projektteammitglieder | * Lead- Koordination unterhalb der Andorid-Gruppe(AG14) * Implementierung der Backendanbindung | Sedat Hatip |
| Projektmitarbeiter | * Gestaltung, Implementierung, Dokumentation von Teilprojekt-Android | Merve Sarı |
| Projektmitarbeiter | * Gestaltung, Implementierung, Dokumentation von Teilprojekt-Android | Eda Atmaca |
| Projektteammitglieder | * Lead- Koordination unterhalb der Andorid-Gruppe(AG9) * Implementierung von Freemeal und Freereide | Tayyip Ensar Özkaya |
| Projektmitarbeiter | * Implementierung von Freemeal und Freeride | Ulaş Algül |
| Projektmitarbeiter | * Implementierung von Freemeal und Freeride | Melih İlerialkan |
| Projektteammitglieder | * Lead- Koordination unterhalb der Andorid-Gruppe(AG11) * Implementierung des Sidebars und des Profils | Deniz Mert Tecimer |
| Projektmitarbeiter | * Design und Dokumentation | Fatih Özoğlu |
| Projektmitarbeiter | * Design und Dokumentation | Zeynep Sena Karabacak |
| Projektteammitglieder | * Lead - Koordination unterhalb der IOS-Gruppe(AG13) * Gibt Aufgaben an Projektmitarbeiter innerhalb der AG’s | Alp Akyüz |
| Projektmitarbeiter | * Gegebene Aufgaben qualitativ hochwertig durchzuführen | Gamze Fıçı |
| Projektmitarbeiter | * Gegebene Aufgaben qualitativ hochwertig durchzuführen | Merve Batırcı |
| Projektmitarbeiter | * Gegebene Aufgaben qualitativ hochwertig durchzuführen | Beyza Patlican |
| Projektteammitglieder | * Lead - Koordination unterhalb der IOS-Gruppe(AG10) * Gibt Aufgaben an Projektmitarbeiter innerhalb der AG’s | Sevcan Avcı |
| Projektmitarbeiter | * Gegebene Aufgaben qualitativ hochwertig durchzuführen | Rumeysa Çeşmeci |
| Projektmitarbeiter | * Gegebene Aufgaben qualitativ hochwertig durchzuführen | İhsan Oğuz Merdan |
| Projektteammitglieder | * Lead - Koordination unterhalb der IOS-Gruppe(AG16) * Gibt Aufgaben an Projektmitarbeiter innerhalb der AG’s | Nusret Özateş |
| Projektmitarbeiter | * Gegebene Aufgaben qualitativ hochwertig durchzuführen | Erinç Kula |
| Projektmitarbeiter | * Gegebene Aufgaben qualitativ hochwertig durchzuführen | Deha Derviş Bayram |
| Projektteammitglieder | * Lead - Koordination unterhabl der Webseite-Gruppe(AG8) * Technischer Architekt | Uğur Güneş |
| Projektmitarbeiter | * Dokumentation * Technischer Architekt | Minel Küpçü |
| Projektmitarbeiter | * Technischer Architekt | Beste Sanli |
| Projektteammitglieder | * Lead - Koordination unterhabl der Webseite-Gruppe(AG2) * Technischer Architekt | Ali Egin |
| Projektmitarbeiter | * Dokumentation | İsmail Çağlar Bayram |
| Projektmitarbeiter | * Dokumentation | Kaan Göktaş |
| Projektteammitglieder | * Lead - Koordination unterhalb der Nextsteps-Gruppe(AG4) | Berk Karaağaç |
| Projektmitarbeiter | * Arbeitet, um das Ziel zu erreichen | Cengiz Furkan Aslan |
| Projektmitarbeiter | * Arbeitet, um das Ziel zu erreichen | Batuhan Özsoy |
| Projektteammitglieder | * Lead - Koordination unterhabl der Webseite-Gruppe(AG5) | Sebiha Gökçem Hakbilen |
| Projektmitarbeiter | * Arbeitet, um das Ziel zu erreichen | Yusufhan Yılmaz |
| Projektmitarbeiter | * Arbeitet, um das Ziel zu erreichen | Muhammed Gök |
| Projektteammitglieder | * Lead - Koordination unterhalb der Webseite-Gruppe(AG7) * Programmierung | M. Mahi Aydın |
| Projektmitarbeiter | * Fragebogen und Dokumentation | Orkun Kuyucu |
| Projektmitarbeiter | * Dokumentation | S. Zeynep Bulut |
| Projektmitarbeiter | * Dokumentation | Alperen Asan |

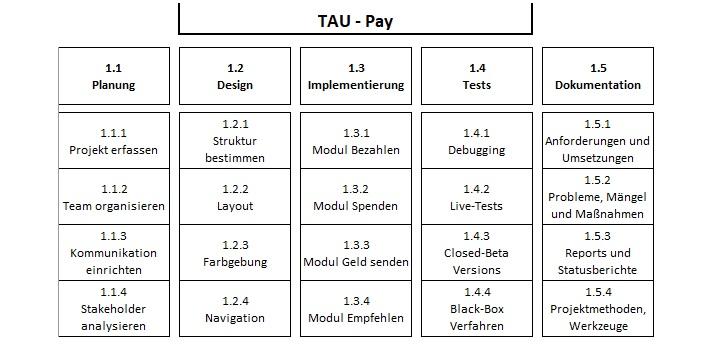
Tabelle 9: Projektorganisation

## Projektkommunikationsplan

| **Projektkommunikationsplan** | | | |
| --- | --- | --- | --- |
| **Bezeichnung der Sitzung / Medium** | **Inhalte** | **Teilnehmerinnen / Zielgruppe(n)** | **Häufigkeit / Dauer** |
| Projektsitzung / Intern meeting | Am Projekt arbeiten | AP10, AP13, AP16 | 1 mal in der Woche |
| Projektstatusbericht | Wie weit wir sind und ob wir noch in dem Zeitplan sind | AP10, AP13, AP16 | 1 mal in der Woche |
| Gruppensitzung | Feedback | Dr. Zeynep Tuncer | Jede 2. Woche |
| Gruppensitzung | Zusammenarbeit und Informationsaustausch | AP10, AP13, AP16 | 1 mal in der Woche |
| Projektsitzung / Intern meeting | Am Projekt arbeiten | AP10, AP13, AP16 | 1 mal in der Woche |
| Workshop 1 | Grobe Besprechung des Projektes | TMS 4 und INF 2 | 4h |
| Projektsitzung / Intern meeting | Zielbestimmungen | AG9, AG11, AG14, AG15 | 1h |
| Wöchentliche Absprache | Fortschritt | Android-Gruppenmitglieder | 1h/Woche |
| WhatsApp Gruppe für Teilprojekt Android | Aktuelle Angelegenheiten | Android-Gruppenmitglieder | permanent |
| WhatsApp Gruppe für Leiter der AG’s | Aktuelle Angelegenheiten | Leiter der AG’s | permanent |
| Projektsitzung | Kommunikation mit AP-Gruppen zu erstellen, für Lagerungsdaten in DB | AP1 | 2 Woche |
| Erste Projektsitzung | Projektbeschreibung | AP2 und AP8 | 1 Stunde |
| Zweite Projektsitzung | Gruppentreffung für Organisation, Design und Implementierung | AP2 und AP8 | 7 Stunden |
| Dritte Projektsitzung | Gruppentreffung für Design und Implementierung | AP2 und AP8 | 6 Stunden |
| Vierte Projektsitzung | Gruppentreffung für Design und Implementierung | AP2 und AP8 | 7 Stunden |
| Projektsitzung | Bestimmen von Kommunikationsnetz zwischen Barcodescanner und Server | AP12 | 1 Woche |
| Projektsitzung | Implementierung/Integration | Alle AP-Gruppen | 1 Monat |
| Projektsitzung | Test und Evaluierung | Alle AP-Gruppen | 2 Woche |

Tabelle 10: Projektkommunikationsplan

## Projektstrukturplan (PSP)[[3]](#footnote-2)



*Abbildung 1.*

*Abbildung 1.1. Projektstrukturplan von Android*

## Arbeitspaketspezifikation

| **Arbeitspaketspezifikation** | |
| --- | --- |
| **Arbeitspaket:**Free-Meal, Free-Ride, Empfehlen (Android) | |
| AP Inhalte / Ergebnisse: | * Implementierung der Free-Meal Funktion * Implementierung der Free-Ride Funktion * Implementierung der Empfehlen Funktion |
| Verantwortung: | Tayyip Ensar Özkaya |
| Mitarbeit: | Merve Sari, Ulas Algül, Melih Ilerialkan |
| Information: | - |
| Abgenommen: | Tayyip Ensar Özkaya, 14.03.2019 |
| **Arbeitspaket:** Login-Seite, Passwort vergessen, Geld Transfer, Session Control, Backend-Anbindung(Android) | |
| AP Inhalte / Ergebnisse: | * Implementierung der Login Seite * Implementierung der Passwort vergessen Seite * Implementierung der Geld Transfer Seite * Implementierung des Session Controls * Backend-Anbindung mit zugehörigen Funktionen |
| Verantwortung: | Sedat Hatip |
| Mitarbeit: | Merve Sari, Hakan Öznur |
| Information: | - |
| Abgenommen: | Sedat Hatip, 14.03.2019 |
| **Arbeitspaket:** Bezahlen, Passwort ändern, Dual Language, Feedback, Einstellungen (Android) | |
| AP Inhalte / Ergebnisse: | * Implementierung der Bezahlen Seite * Implementierung der Passwort ändern Seite * Implementierung der Dual Language Seite * Implementierung der Feedback Seite * Implementierung der Einstellungen Seite |
| Verantwortung: | Muhammed Zahid Bozkus |
| Mitarbeit: | Hakan Öznur, Ahmet Cakirer |
| Information: | - |
| Abgenommen: | Muhammed Zahid Bozkus, 14.03.2019 |
| **Arbeitspaket:** Sidebar, Kontostand, Profil (Android) | |
| AP Inhalte / Ergebnisse: | * Implementierung des Sidebars mit Kontostand und Profilanzeige |
| Verantwortung: | Deniz Mert Tecimer |
| Mitarbeit: | Zeynep Sena Karabacak |
| Information: | - |
| Abgenommen: | Deniz Mert Tecimer, 14.03.2019 |
| **Arbeitspaket:** Design, Umfragen, Dokumentation (Android) | |
| AP Inhalte / Ergebnisse: | * Designs, Umfragenerstellung und -verbreitung, Dokumentation |
| Verantwortung: | Merve Sari, Eda Atmaca |
| Mitarbeit: | Ulas Algül, Hakan Öznur, Ahmet Cakirer, Zeynep Sena Karabacak, Fatih Özoglu, Melih Ilerialkan |
| Information: | - |
| Abgenommen: | Merve Sari, 14.03.2019 |

Tabelle 11: Arbeitspezifikation von Android-Gruppe



*Abbildung 2.*

*Abbildung 2.*

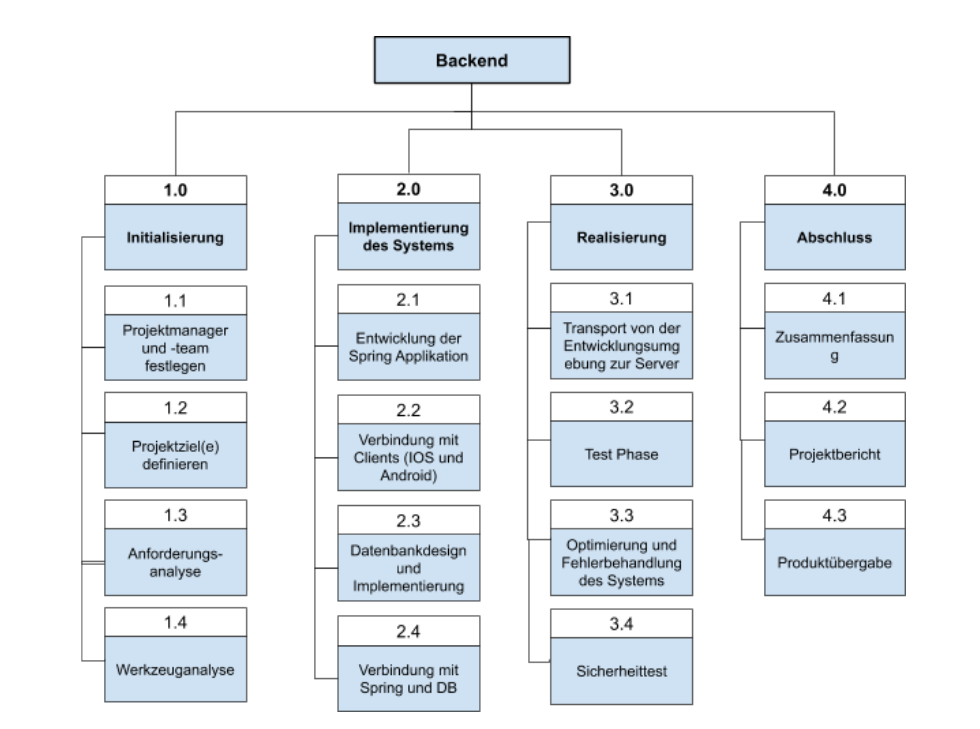
*Abbildung 2.*

*Abbildung 1.2. Projektstrukturplan von IOS*

| **Arbeitspaketspezifikation** |
| --- |

| **Arbeitspaket:** <1.1: Anmelden> | |
| --- | --- |
| AP Inhalte / Ergebnisse: | * Direkte Anmeldung durch Matrikelnummer und Passwort |
| Verantwortung: | Front- und Backend: Sevcan Avci |
| Mitarbeit: | AP 1.2, AP 1.3, AP 3.4, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Sevcan Avci |
| **Arbeitspaket:** <1.2: Passwort vergessen> | |
| AP Inhalte / Ergebnisse: | * Wenn jemand sein Passwort vergessen hat, kann es durch diesen Button durch eine E-Mail erneuert werden |
| Verantwortung: | Frontend: Deha Beyram, Backend: Alp Akyüz |
| Mitarbeit: | AP 1.1, AP 1.3, AP 3.4, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Deha Beyram und Alp Akyüz |
| **Arbeitspaket:** <1.3: First time login screen> | |
| AP Inhalte / Ergebnisse: | * Bei der ersten Anmeldung default Passwort Veränderung |
| Verantwortung: | Front- und Backend: Alp Akyüz |
| Mitarbeit: | AP 1.1, AP 1.2, AP 3.4, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Alp Akyüz |
| **Arbeitspaket:** <2.1: Bezahlen> | |
| AP Inhalte / Ergebnisse: | * Essen in der Mensa oder die Shuttlefahrt bezahlen |
| Verantwortung: | Front- und Backend: Rumeysa Cesmeci |
| Mitarbeit: | AP 1.1, AP 3.1, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Rumeysa Cesmeci |
| **Arbeitspaket:** <2.2: Geldübertragung> | |
| AP Inhalte / Ergebnisse: | * Geld an jemanden schicken, der kein ausreichendes Guthaben hat |
| Verantwortung: | Front- und Backend: Nusret Özates |
| Mitarbeit: | AP 1.1, AP 3.1, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Nusret Özates |
| **Arbeitspaket:** <2.3: Spenden> | |
| AP Inhalte / Ergebnisse: | * Hier kann Geld an hilfsbedürftige Personen geschickt werden |
| Verantwortung: | Frontend: Merve Baritci, Backend: Alp Akyüz |
| Mitarbeit: | AP 1.1, AP 3.1, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Merve Baritci und Alp Akyüz |
| **Arbeitspaket:** <2.4: Empfehlen> | |
| AP Inhalte / Ergebnisse: | * Eine hilfsbedürftige Person die noch nicht auf dieser Liste ist, kann hier von seiner Umgebung auf diese Liste empfohlen werden |
| Verantwortung: | Front- und Backend: Gamze Fici |
| Mitarbeit: | AP 1.1, AP 3.1, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Gamze Fici |
| **Arbeitspaket:** <2.5: Pop Ups> | |
| AP Inhalte / Ergebnisse: | * Warnungen und Mitteilungen innerhalb der App |
| Verantwortung: | Front- und Backend: Alp Akyüz |
| Mitarbeit: | AP 3.3, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Alp Akyüz |
| **Arbeitspaket:** <3.1: Profil> | |
| AP Inhalte / Ergebnisse: | * <möglichst messbare Beschreibung der Inhalte und Ergebnisse des jeweiligen Arbeitspakets>> |
| Verantwortung: | AP 1.1, AP 2.1, AP 4.0 |
| Mitarbeit: | Front- und Backend: Alp Akyüz |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Alp Akyüz |
| **Arbeitspaket:** <3.2: Feedback> | |
| AP Inhalte / Ergebnisse: | * Das Essen und/oder die Shuttlefahrt kann hier benotet werden und man kann es bewerten |
| Verantwortung: | Front- und Backend: Beyza Patlican |
| Mitarbeit: | AP 2.0, AP 3.3, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Beyza Patlican |
| **Arbeitspaket:** <3.3: Einstellungen> | |
| AP Inhalte / Ergebnisse: | * Hier kann die Sprache verändert und Feedback gegeben werden |
| Verantwortung: | Front- und Backend: Alp Akyüz |
| Mitarbeit: | AP 1.1, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Alp Akyüz |
| **Arbeitspaket:** <3.4: Passwort verändern> | |
| AP Inhalte / Ergebnisse: | * Falls jemand sein Passwort verändern möchte |
| Verantwortung: | Front- und Backend: Nusret Özates |
| Mitarbeit: | AP 1.1, AP 3.3, AP 4.2, AP 4.3 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Nusret Özates |
| **Arbeitspaket:** <4.1: Lokalisierung> | |
| AP Inhalte / Ergebnisse: | * Die Veränderung der Sprache (Deutsch zu Türkisch und umgekehrt) |
| Verantwortung: | Nusret Özates |
| Mitarbeit: | AP 1.0, AP 2.0, AP 3.0, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Nusret Özates |
| **Arbeitspaket:** <4.2: Server Verbindung> | |
| AP Inhalte / Ergebnisse: | * Verbindungen zum Backend |
| Verantwortung: | Alp Akyüz |
| Mitarbeit: | AP 1.0, AP 2.0, AP 3.0, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Alp Akyüz |
| **Arbeitspaket:** <4.3: Kontodetails> | |
| AP Inhalte / Ergebnisse: | * Die Speicherung von den verwendeten Daten |
| Verantwortung: | Sevcan Avci |
| Mitarbeit: | AP 1.0, AP 2.0, AP 3.0, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Sevcan Avci |
| **Arbeitspaket:** <4.4: Verschönerung> | |
| AP Inhalte / Ergebnisse: | * Design Verschönerungen |
| Verantwortung: | Beyza Patlican |
| Mitarbeit: | AP 1.0, AP 2.0, AP 3.0, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Beyza Patlican |
| **Arbeitspaket:** <4.5: Code Verbesserung> | |
| AP Inhalte / Ergebnisse: | * Kleine Fehler im Code |
| Verantwortung: | Nusret Özates und Alp Akyüz |
| Mitarbeit: | AP 1.0, AP 2.0, AP 3.0, AP 4.0 |
| Information: | Dr. Zeynep Tuncer |
| Abgenommen: | 10.03.2019, Nusret Özates und Alp Akyüz |

Tabelle 12: Arbeitspezifikation von IOS-Gruppe



*Abbildung 3.*

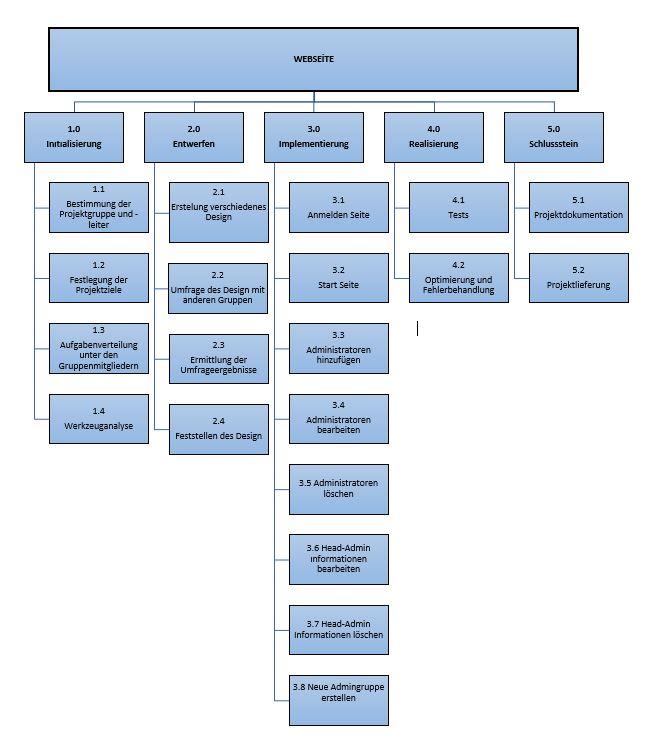
*Abbildung 3.*

*Abbildung 3.*

*Abbildung 1.3 Projektstrukturplan von Backend*

| **Arbeitspaketspezifikation** | |
| --- | --- |
| **Arbeitspaket: AP12** Server Einstellung | |
| AP Inhalte / Ergebnisse: | * Akzeptierung der Anforderungen von Clients * Programmierung durch Java Spring * Integration von Rest API * Integration von Programm(Java Spring) zu Server * Implementierung zwischen Web-Clients/Mobile Devices/BarcodeScanner und Server |
| Verantwortung: | Ömer Cem Turan |
| Mitarbeit: | Aybüke Bayramiç  Onur Burak Kılıç |
| Abgenommen: | 04.03.2019, Ömer Cem Turan |
| **Arbeitspaket: AP1** DB modellieren (SQL Team) | |
| AP Inhalte / Ergebnisse: | * Erstellen eine Database in Mysql für Mensakarte System * Darstellen der Diagramm von Database * Lagerung von einige Daten zur Database * Testprüfung und Evaluierung nach der Implementierung |
| Verantwortung: | Ayşe Rabia Özbek |
| Mitarbeit: | Onur Mert Doğan  Kartal Can Çelikkanat |
| Abgenommen: | 04.03.2019 Ayşe Rabia Özbek |

Tabelle 13: Arbeitspezifikation von Backend-Gruppe

  
  
Image 4:

*Abbildung 4.*

*Abbildung 4.*

*Abbildung 4.*

*Abbildung 1.4 Projektstrukturplan von Webseite*

| **Arbeitspaketspezifikation** | |
| --- | --- |
| **Arbeitspaket:** 2.1: AP8, AP7 | |
| AP Inhalte / Ergebnisse: | * Entwerfen verschiedener Designs |
| Verantwortung: | Mahi Aydin |
| Mitarbeit: | Minel Küpcü |
| Information: | Entwerfen - Minel Küpcü, Mahi Aydin |
| Abgenommen: | 03.04.2019, Mahi Aydin |
| **Arbeitspaket:** 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8: AP2, AP8 | |
| AP Inhalte / Ergebnisse: | * Aufbau der Anmelden Seite mit Benutzer- und Passwortname. Dann Erstellung einer Start Seite mit Tabelle. Erstellen der erforderlichen Funktionen zum Ausführen von Schaltflächen. Darüber hinaus muss eine weitere Seite erstellt werden, die zum Bearbeiten der Head-Admin-Informationen und zum Einfügen der erforderlichen Funktionen erstellt wurde. |
| Verantwortung: | Ali Egin, Ugur Günes |
| Mitarbeit: | Minel Küpcü, Beste Sanli, Ismail Caglar Bayram |
| Information: | Minel Kupcu, Ali Egin, Ugur Günes |
| Abgenommen: | 12.04.2019, Dr. Zeynep Tuncer |
| **Arbeitspaket:** 4.1, 4.2: AP2, AP8 | |
| AP Inhalte / Ergebnisse: | * Die Webseite wurde von Anfang bis Ende aller Versuche überprüft. Danach wurden die notwendigen Vorkehrungen und Verbesserungen entsprechend den Mängeln und Rückmeldungen getroffen. |
| Verantwortung: | Ali Egin, Ugur Günes |
| Mitarbeit: | Minel Küpcü |
| Information: | - |
| Abgenommen: | 05.05.2019 , Ali Egin |

Tabelle 14: Arbeitspezifikation von Webseite-Gruppe



*Abbildung 5.*

*Abbildung 5.*

*Abbildung 5.*

*Abbildung 1.5. Projektstrukturplan von Webseite*

| **Arbeitspaketspezifikation** | |
| --- | --- |
| **Arbeitspaket:**AP7:Admin2 Shuttle | |
| AP Inhalte / Ergebnisse: | Erstellung der einen Webseite für Shuttle Admin um Personen zu löschen, zu ändern, hinzuzufügen |
| Verantwortung: | Koordination der Shuttle-Adminseite |
| Mitarbeit: | Sema Zeynep Bulut,Alperen Asan,Mahi Aydın,Orkun Kuyucu |
| Information: | <Name> |
| Abgenommen: | <Datum, Name> |
| **Arbeitspaket:** AP2:Head Admin | |
| AP Inhalte / Ergebnisse: | Koordination und Überprüfung der einen Webseite für Admin um Personen zu löschen, zu ändern, hinzuzufügen |
| Verantwortung: | Koordination der Adminseiten |
| Mitarbeit: | Ali Engin,İsmail Çağlar Bayram,Kaan Göktaş |
| Information: | <Name> |
| Abgenommen: | <Datum, Name> |
| **Arbeitspaket:** AP8:Head Admin | |
| AP Inhalte / Ergebnisse: | Koordination und Überprüfung der einen Webseite für Admin um Personen zu löschen, zu ändern, hinzuzufügen |
| Verantwortung: | Koordination der Adminseiten |
| Mitarbeit: | Uğur Güneş,Minel Küpçü,Beste Şanlı |
| Information: | <Name> |
| Abgenommen: | <Datum, Name> |
| **Arbeitspaket:** AP6:Admin1 Mensa | |
| AP Inhalte / Ergebnisse: | Erstellung der einen Webseite für Mensa Admin um Personen zu löschen, zu ändern,hinzuzufügen |
| Verantwortung: | Koordination der Mensa-Adminseite |
| Mitarbeit: | Yiğit Mesci,Mert Özgür,Mert Cenk Sabancı |
| Information: | <Name> |
| Abgenommen: | <Datum, Name> |

Tabelle 15: Arbeitspezifikation von Webseite-Gruppe



*Abbildung 6.*

*Abbildung 6.*

*Abbildung 6.*

*Abbildung 1.6. Projektstrukturplan von Nextsteps*

| **Arbeitspaketspezifikation** | |
| --- | --- |
| **Arbeitspaket:** Die Bestellung von der Kantine mittels mobiler Applikation | |
| AP Inhalte / Ergebnisse: | * Das Smartsystem für Mensa soll auch für die Kantine entwickelt werden. Diese AP ermöglicht es, mittels der zu entwickelnder mobilen Applikation, Essen zu bestellen und bezahlen, ohne in der Kantine anwesend zu sein. * Ergebnisse: Ein online Essenbestellungssystem |
| Verantwortung: | Berk Karaağaç |
| Mitarbeit: | Sebiha Gökçem Hakbilen |
| Information: | [Berkkaraagac25@gmail.com](mailto:Berkkaraagac25@gmail.com), [hakbilengokcem@gmail.com](mailto:hakbilengokcem@gmail.com) |
| Abgenommen: | 04.03.2019 |
| **Arbeitspaket:** Der Eintritt zur Universität mittels mobiler Applikation | |
| AP Inhalte / Ergebnisse: | * Jeder Student hat sein/ihr Studentenausweis schon einmal zu Hause vergessen, um dieses Problem zu erheben, muss ein System entwickelt werden, mittels um zu entwickelnder mobiler Applikation in die Universität einzutreten. |
| Verantwortung: | Batuhan Özsoy |
| Mitarbeit: | Berk Karaağaç |
| Information: | [batozsoy@gmail.com](mailto:batozsoy@gmail.com), [berkkaraagac25@gmail.com](mailto:berkkaraagac25@gmail.com) |
| Abgenommen: | 04.03.2019 |
| **Arbeitspaket:** Der Eintritt zu den Laboren mittels mobiler Applikation | |
| AP Inhalte / Ergebnisse: | * Kommunikation mit anderen Gruppen zu erstellen, um eine systematische Struktur zu erhalten und Möglichkeiten zu erfahren |
| Verantwortung: | Sebiha Gökçem Hakbilen |
| Mitarbeit: | Yusufhan Yılmaz |
| Information: | [yusufhanylmaz@gmail.com](mailto:yusufhanylmaz@gmail.com), hakbilengokcem@gmail.com |
| Abgenommen: | 04.03.2019 |

Tabelle 16: Arbeitspezifikation von Nextsteps-Gruppe

## Projektmeilensteinplan

| Milestones | Beginn – Ende |
| --- | --- |
| Milestone 1: *Projektbeginn* | 04.03. – 04.03.2019 |
| Milestone 2: *Projektbeschreibung* | 05.03. – 05.03.2019 |
| Milestone 3 : *Prototypenbau und Umsetzung, inkl. Evaluierung und Test* | 05.03. – 05.05.2019 |
| Milestone n : *Fertigstellung des Projekts (Dokument und Technik)* | 17.05.2019 |

Tabelle 17: Meilensteine

| **Projektmeilensteinplan** | | | | |
| --- | --- | --- | --- | --- |
| **PSP-Code** | **Meilenstein-Bezeichnung** | **Termin PLAN** | **Adaptierte Plantermine per** **<Datum>** | **Termin IST** |
| AP1/12.1 | Datenbank Erstellung und Spring | 19.03.2019 | 26.03.2019 | 26.03.2019 |
| AP1/12.2 | Überprüfung durch Postman | 26.03.2019 | 08.04.2019 | 30.03.2019 |
| AP1/12.3 | Verbindung mit anderen AG‘s | 13.04.2019 | 15.04.2019 | 20.04.2019 |
| AP1/12.4 | Prüfung mit Client‘s | 29.04.2019 | 30.04.2019 | 26.04.2019 |
| AP1/12.5 | Abschlusspräsentation und Simulation | 17.05.2019 | 18.05.2019 | 20.05.2019 |
| Android/IOS | Projektplanung - Beschreibung | 04.03.2019 | 11.03.2019 | 04.03.2019 |
| Android/IOS | Erstellung der Oberfläche | 26.03.2019 | 03.04.2019 | 26.03.2019 |
| Android/IOS | Implementierung | 30.04.2019 | 06.04.2019 | 30.04.2019 |
| Android/IOS | Tests durchführen | 10.05.2019 | - | 10.05.2019 |
| Android/IOS | Dokumentation | 17.05.2019 | - | 17.05.2019 |
| Webseite 1.1 | 1-Die Anfangszeit des Projekts | 04.03.2019 | - | 04.03.2019 |
| Webseite 1.2 | 2-Festlegung der Ziele, die am Ende des Projekt erreicht werden wollen | 05.03.2019  11.03.2019 | 05.03.2019 | 05.03.2019 |
| Webseite 1.3 | 3-Bestimmung der  Anforderungen und Erwartungen von Kunden | 13.03.2019 | 13.03.2019 | 13.03.2019 |
| Webseite 1.4 | 4-Besttimung die Tools, die alle Arbeitspaketten gemeinsam benutzt werden soll | 05.03.2019  19.05.2019 | 05.03.2019  19.05.2019 | 05.03.2019 |
| Webseite  2.1  2.2 | 5- Erstellen der Entwürfe des Systems.  Erstellung des ersten Prototyps gemäß den Entwürfen. | 11.03 – 08.04.2019 | - | 08.04.2019 |
| Webseite  2.3  2.4 | 6-Entwerfen der Benutzeroberfläche | 20.03.2019 | - | 27.03 -  03.04.2019 |
| Webseite  3.1  3.2  3.3  3.4  3.5  3.6  3.7  3.8 | 7- Anwendung des erstellten Produkts | 11.03 – 30.04.2019 | 12.04.2019 | 12.04.2019 |
| Webseite  4.1  4.2 | 8- Feststellen, ob das Produkt die Anforderungen erfüllt. Beobachtung fehlender Punkte | 08.04 – 10.05.2019 | 12.04.2019 | 12.04-  17.05.2019 |
| Webseite  5.1 | 9- Erstellung von Dokumentationen zum Produktionsprozess | 04.03 – 10.05.2019 | - | 04.03 -  15.05.2019 |
| Webseite  5.2 | 10-Bewertung des Produkts | 17.05.2019 | - | 17.05.2019 |

Tabelle 18: Projektmilensteine

## **Projektterminplan**

Bei komplexen Projekten ist es in der Regel sinnvoll, den Terminplan in Form eines **Balkenplanes[[4]](#footnote-3)** darzustellen und an dieser Stelle einzufügen.

Bei Projekten mittlerer oder geringer Komplexität kann es aber auch schon ausreichen, eine Terminübersicht nach folgendem Muster zu führen und kontinuierlich zu aktualisieren:

| **Projektterminplan** | | | |
| --- | --- | --- | --- |
| **Arbeitspaket / Meilenstein** | **Verantwortlich für die termingerechte Fertigstellung** | **Termin PLAN** | **Termin IST** |
| AP1-Ladung Database auf Server | Ayşe Rabia Özbek | 25.04.2019 | 20.04.2019 |
| AP1-Datenbank Erstellen | Onur Mert Doğan | 20.04.2019 | 19.04.2019 |
| AP1-Datenbank Modellierung | Kartal Can Çelikkanat | 30.04.2019 | 29.04.2019 |
| AP12-Erstellen Java Spring Boot | Ömer Cem Turan | 06.04.2019 | 02.04.2019 |
| AP12-Java Spring Classes für Qr-Code Erstellung | Aybüke Bayramiç | 05.04.2019 | 01.04.2019 |
| AP12-Verbindung zwischen Java Spring Boot und Database | Onur Burak Kılıç | 26.04.2019 | 26.04.2019 |
| Verteilung der Aufgaben | Projektsitzung iOS /Android Team,  Leiter | 10.03.2019, 11. Woche |  |
| *4.0: Prototypenbau und Umsetzung* | Datenbankinformationen an AG1 und AG12 geben,  Leiter | 15.03.2019, 12. Woche |  |
| *1.0, 2.0, 3.0:* Prototypenbau und Umsetzung | Implementierung einer GUI ,  Leiter | 15.03.2019, 12. Woche |  |
| *1.0, 2.0, 3.0: Prototypenbau und Umsetzung* | Entwurf der allgemeinen App ,  Leiter | 17.03.2019, 12. Woche |  |
| *1.0, 2.5, Prototypenbau und Umsetzung* | Beginn von Bestätigungs- und Sicherheitsverfahren,  Leiter | 20.03.2019, 13. Woche |  |
| *1.0, 2.0, 3.0: Prototypenbau und Umsetzung* | Fertigstellung der Schnittstellen,  Leiter | 21.03.2019, 13. Woche |  |
| *4.2: Prototypenbau und Umsetzung* | Erste Verbindung mit Backend,  Leiter | 24.03.2019, 13. Woche |  |
| *4.0: Prototypenbau und Umsetzung* | Informations- und Datenaustausch mit AG 1 und AG 12,  Leiter | 26.03.2019, 14. Woche |  |
| *1.0, 2.0, 3.0, 4.0: Prototypenbau und Umsetzung* | Gemeinsamer Click-Dummy Entwurf mit AG15 und AG14(Android&iOS), Leiter | 28.03.2019, 14. Woche |  |
| *4.0: Prototypenbau und Umsetzung* | Backend von AG1 & AG12,  Leiter | 28.03.2019, 14. Woche |  |
| *4.0: Prototypenbau und Umsetzung* | Backend Implementierung,  Leiter | 30.03.2019, 14. Woche |  |
| *1.0, 2.0, 3.0, 4.0:* Testphase | Fertigstellung von demPrototyp ,  Leiter | 08.04.2019, 16. Woche |  |
| Letzte verfeinerungen | Implementierung und Verbesserung,  Leiter | 11.05.2019, 20. Woche | Letzte verfeinerungen |
| Webseite 1.1, 1.2: Bestimmung der Projektgruppe- und leiter, Festlegung der Projektziele | Dr Zeynep tuncer | 04.03.2019, Woche 5 | 04.03.2019, Woche 5 |
| Webseite 1.3, 1.4: Aufgabenverteilung unter den Gruppenmitgliedern, Werkzeuganalyse | AP2, AP8 | 05.03.2019, Woche 5 | 18.03.2019, Woche 7 |
| Webseite2.1, 2.2, 2.3, 2.4: Entwerfen | AP2, AP8, AP6, AP7 | Woche 7 | Woche 8-9 |
| Webseite 3.1, 3,2 , 3.3, 3.4, 3.5, 3.6, 3.7, 3.8: Implementierung | Ali Egin, Ugur Gunes, Minel Küpcü, Beste Sanli | 11.03. - 30.04.2019, Woche 6 - 12 | 11.03. - 30.04.2019, Woche 6 - 12 |
| Webseite 4.1, 4.2 : Realisierung | Ali Egin, Ugur Gunes, Minel Küpcü | 08.04 – 10.05.2019, Woche 9 -13 | 12.04-  17.05.2019, Woche 9 - 14 |
| Webseite 5.1 | Minel Kupcu, Ismail Caglar Bayram, Beste Sanli, Kaan Goktas | 04.03.- 15.03.2019, Woche 1 - 14 | 04.03.- 15.03.2019, Woche 1 - 14 |
| Webseite 5.2 | Dr Zeynep Tuncer | 17.05.2019 | 17.05.2019 |

Tabelle 19: Projektterminplan

## Projektkostenplan

| **Projektkostenplan** | | | | |
| --- | --- | --- | --- | --- |
| **Arbeitspaket / Meilenstein** | **Kostenart** | **Menge** | **Preis / Einheit** | **Kosten [€]** |
| Backend | Server Dienst | 1 | 20 Dollar/Monat | 20 Dollar/Monat |
| Backend | Domain Name | 1 | 10 Dollar/Jahr | 10 Dollar/Monat |
| Backend | SSL Zertifikat | 1 | 10 Dollar/Jahr | 10 Dollar/Monat |
| Vollendung des Projekts | Programmieraufwand | 102 Stunden | € 0 | € 0 |
| Webseite | Domain Name | für einen Jahr | kostenlos | €0 |
| Webseite | Web Hosting Service |  | kostenlos | €0 |

Tabelle 20: Projektkostenplan

## **Projektrisiken**

| **Projektrisikoanalyse** | | | |
| --- | --- | --- | --- |
| **Risiko** | **Eintritts-Wahrscheinlichkeit** | **Schadensausmaß / Auswirkungen** | **Maßnahmen** |
| Unverlässlichkeit anderer Gruppen | 90% | Arbeit der unverlässlichen Gruppe muss von jemand Anderem gemacht werden | Die Gruppen dazu drängen, ihre Arbeit zu machen |
| Nicht Einhaltung der Termine | 30% | Stress bei der Fertigstellung | Termine einhalten, regelmäßige Kontrollen und Besprechungen |
| Inkompatibilität der Applikation | 10% | Applikation würde nicht funktionieren. | Ältere API Version in Android Studio verwenden |
| Benutzerfreundlichkeit | 10% | Design stimmt nicht auf allen Seiten überein | Gute Kommunikation mit den anderen AGs |
| Cybersicherheit | 30% | Manipulation der Datenbank | Sicherheitsmaßnahmen, regelmäßige Kontrollen |
| Risikofelder im Projektumfeld | %10 | auf Qualität und Bedienbarkeit | Mehr tests und Simulationen |
| Risiken aus  Umweltveränderungen | %5 | auf Termin | Webbasierte Speicherdienste verwenden z.B. Github |
| Risiken aus Arbeitspaketen | %6 | auf Qualität und Termin | mehr Kommunikation und testen |
| Risiken aus internen Veränderungen | %5 | auf Qualität und Effektivität | alle Daten und Veränderungen müssen kopieren und speichern |
| Risiken aus fehlerhafter Planung | %5 | auf alle Projekt und also Termin | Mehr kontrollieren und testen |
| falsche Dateneingabe | %50 | regelmäßige Überprüfung | AP1-AP12-AP2-AP8 |
| falsche Datenveränderung | %12,5 | regelmäßige Überprüfung | AP1-AP12-AP2-AP8 |

Tabelle 21: Projektrisiken

## **Projektfortschrittsbericht**

| **Projektfortschrittsbericht** | |
| --- | --- |
| Berichtszeitraum: | Von 26.03.2019 bis 17.05.2019 |
| **☐ Projektkrise**  **☐ Projekt in Schwierigkeiten**  **☐ Projekt im Plan** | **Gesamtstatus:**   * Das Projekt ist vollendet, es werden lediglich kleinere Bugfixes und Usability-Maßnahmen vorgenommen * Fertig geplantes und Implementierungsfähiges Produkt, mit festgelegten Ressourcen. |
| **Status Leistungsziele**:   * Planung abgeschlossen * Implementierung und Programmierung * Kommunikationskonzept fertig | Maßnahmen:   * keine * delegieren und beobachten damit es in der Zeit bleibt * Kommunikationskonzept fertig, in der implementierungsphase die Umsetzung beobachten gegebenenfalls lenkend eingreifen. |
| **Status Terminziele**:   * Planungsphase abgeschlossen * Implementierung des Programms * Einführung des Kommunikationssystems begonnen | Maßnahmen:   * keine * im Blick haben * überwachen und steuern |
| **Status Kostenziele**:   * Monatlich 22 Dollar | Maßnahmen:   * keine |
| **Status Teamarbeit:**   * keine mehr nötig | Maßnahmen:  - |
| **Notwendige Entscheidungen:**   * - | **Nächste Schritte:**   * weitere Tests und Echtversuche |

Tabelle 22: Projektfortschrittsbericht

# Projektumsetzung

| Projektdefinition | 1. Ziele genau definieren |
| --- | --- |
| Projektplanung | 1. Meilensteine finden und festlegen. 2. Projekt strukturieren 3. Zeitrahmen feststellen 4. Einzelaufgaben planen |
| Projektdurchführung | 1. Arbeit organisieren 2. Hardware/Software festlegen 3. Informationen sammeln und auswerten 4. Integrationen durchführen 5. Dokumentieren 6. Präsentieren |
| Projektabschluss | 1. Dokumentationen bewerten lassen 2. Präsentation bewerten lassen 3. Den Arbeitsprozess bewerten lassen |

Tabelle 23: Projektumsetzung

## Einleitung

Da wir in einer Zeit leben, wo fast alle technologischen Geräte über das Internet verbunden sind und miteinander kommunizieren können, kann man sich ein Leben ohne das Internet nicht mehr vorstellen.

Obwohl sich fast alle Geräte mit dem Internet verbinden können, gibt es an unserer Universität immer noch das Problem, dass jeder Student 2 Karten bei sich tragen muss. Eine für das Shuttle und die andere für die Mensa/den Universitäts-Eintritt. Unsere Aufgabe als TMS-4 und INF-2 Studenten war das Minimieren der Anzahl der Karten

Am Projektende wird von unserem Team verlangt, dass jemand, sei es Student, Dozent oder wissenschaftlicher Mitarbeiter, einem hilfsbedürftigen Studenten ein Free Meal schenken darf. Außerdem wird auch erwartet, dass man auf Wunsch einer Person Geld durch das System senden kann.

Als Zusammenfassung kann man also sagen, dass das Hauptziel von unserem Projekt die Reduktion der Anzahl der Karten und die Reduktion des sozialen Drucks auf hilfsbedürftigen Studenten ist.

## 

## Problemstellung

Wie bereits in der Einleitung beschrieben, ist das Hauptproblem an unserer Universität, dass jeder 2 Karten bei sich tragen muss und dass es nur 2 Automaten gibt, bei der man auf die Mensa-Karte Guthaben aufladen kann und lediglich ein Standpunkt, wo man die Shuttle-Karte aufladen kann

Bei fehlendem Guthaben auf der Shuttle-Karte ist die Benutzung des Shuttles unter keinen Umständen möglich. Falls man in den Shuttle an einem anderen Standpunkt einsteigt (wo sich die Aufladenstelle nicht befindet), ist man bei ungenügendem Guthaben gezwungen, andere Fahrgäste zu fragen.

Ein anderes Problem ist das Bezahlen mit der Mensakarte: Denn es enthält einen großen Aufwand. Erst muss man das Guthaben aufladen. Da die Aufladestellen nicht in der Mensa stehen muss man erst zur Aufladestelle gehen und dann wieder zurück zur Mensa, welches zum Zeitverlust führt.

Als Letztes muss man noch berücksichtigen, dass Visitenkarten einen geringen bis zu keinem materiellen Wert tragen. Somit gehört das Vergessen oder gar das Verlieren mittlerweile zum Alltag. Da das Smartphone einen hohen materiellen Wert hat und bereits in vielen anderen Bereichen des Alltags benötigt wird, ist die Wahrscheinlichkeit, dass eine Person sein Smartphone vergisst/verliert, niedriger als das, dass sie eine Visitenkarte vergisst/verliert.

## Stand der Technik

Das Benutzen von einer einzigen Karte wird an sehr vielen Universitäten zurzeit angewendet. Als Beispiel kann man den „Yıldız Technik Universität“ geben. Denn kürzlich gab es in dieser Universität auch eine Änderung der Ringkarten: Die Studenten können den Studentenausweis als Shuttle-Karte und mit der Eigenschaft, dass man darauf Geld laden kann, als elektronische Geldbörse verwenden. Leider wird diese Karte dort in der Mensa nicht angewendet. **[YEK\_Tanıtım\_Kilavuzu 2019, S.1.]**

Als ein Anwende Beispiel für das System “Free Meal”, kann man das Projekt nehmen, welches in der „Düzce Fatih İlkokulu“ angewendet wird. Dieses Projekt wurde durch einen Familienverein gestartet und dient dafür, dass jeder Hilfsbedürftige aus der Kantine kostenlos Obst oder ein Sandwich Menü bekommen kann. Die festgestellten hilfsbedürftigen Studenten bekommen einen Beleg, welche die Eltern von anderen Studenten bezahlt haben.

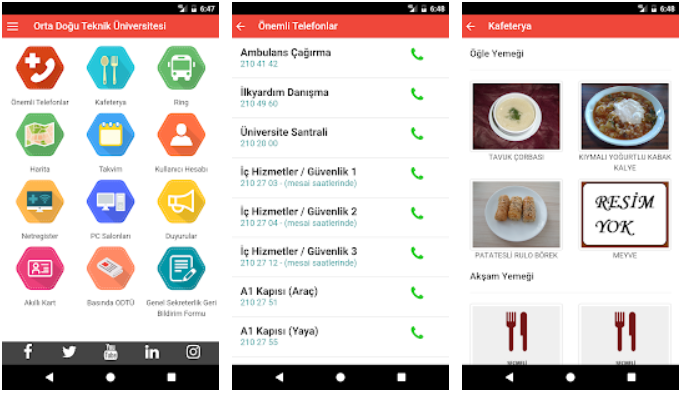
Wir haben das “Ein-Karten”-System noch weiterentwickelt und bringen all diese Kartenfunktionen in einer einzigen mobilen Anwendung unter.

## 2.1.2.1 METU App für Studenten - Anwendungsbeispiele

Die Middle East Technical University METU App ist die offizielle Anwendung von dieser Universität. App werden die Interaktionen , die Kartenoptionen , die Bestimmung von Ort der Klasse oder die Cafetaria auf dem Campus verwendet. Studenten können die Neuigkeiten auf dem Campus über diese App erfahren .

Verfügbare Dienste:

* Wichtige Telefonnummer
* Cafeteriaort
* Ringpositionen an der Universität
* Karte , die an der Universität benutzt werden .
* METU E-Mail
* Akademischer Kalender
* Der Ort der Klasse auf dem Campus
* Der Ort von PC-Räume
* Feedback-Formular des Generalsekretrias
* ODTÜ Karte
* ODTU Pass
* ODTÜ in der Presse
* Kompatibel mit Android & IOS



*Abbildung 7.*

*Abbildung 7.*

*Abbildung 7.*

*Abbildung 2.1: Zusätzliche Informationen von ODTÜ APP*

**Größe :** 3,5 M

**Installationen :** 20,000 +

**Erforderliche Android Version :** 4,4 oder höher

**Altersfreigabe :** PEGI 3

**Entwickler :** Die Studenten von Informatikingenieur

## 2.1.2.2 Stanford University Mobile App - Anwendungsbeispiele

Stanford Mobile ist die offizielle mobile App der Stanford University. Sie haben jederzeit und überall auf Ihrem Mobilgerät Zugriff auf wichtige Stanford-Informationen und -Dienste.



*Abbildung 8.*

*Abbildung 8.*

*Abbildung 8.*

*Abbildung 2.2: Logo von Stanford University Mobile App*

Mit Stanford Mobile können Sie:

\* Suchen Sie die Campus-Karte

\* Suche im Campusverzeichnis

\* Finden Sie gastronomische Einrichtungen und Öffnungszeiten

\* Überprüfen Sie die Routen und Fahrpläne der Shuttlebusse

\* Universitätsnachrichten lesen

\* Stöbern Sie in unserem Veranstaltungskalender

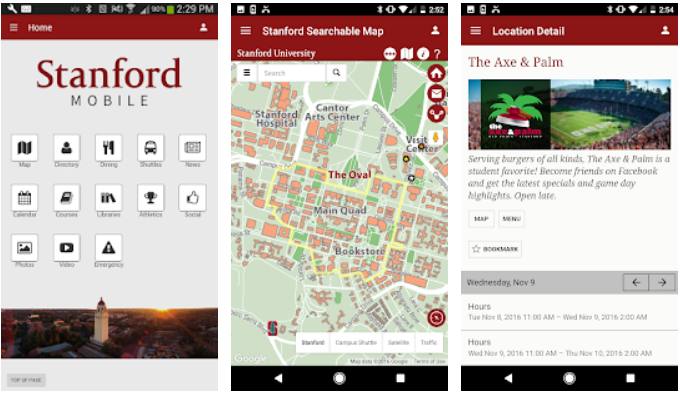
\* Kurse erkunden

\* Bibliotheksbestände durchsuchen

\* Erhalten Sie Neuigkeiten und Termine für Kardinalsportarten

\* Bleiben Sie in sozialen Netzwerken in Verbindung

\* Erhalten Sie Stanford-Info-Benachrichtigungen, einschließlich Push-Benachrichtigungen



*Abbildung 9.*

*Abbildung 9.*

*Abbildung 9.*

*Abbildung 2.3. Zusätzliche Informationen von Stanford Mobile App.*

## 2.1.2.3 Zielsetzung

Damit ein System entsteht, was mit allem benötigten Elementen funktioniert, d.h. damit die vorbestimmte Ziele erreicht werden, muss man mit den anderen Arbeitsgruppen ständig in Kontakt sein.

Bei der Erstellung von unserem System, wurde mit der Android- und IOS-Gruppen für ein einheitliches Design für Android & iOS Endgeräte und mit der Backend-Gruppe für die Datenbankanbindung zusammengearbeitet. Wenn die Integration vollständig abgeschlossen und das System betriebsbereit ist, wird das erstellte Produkt an der Türkisch-Deutsche Universität angewendet. Es wurde aber vorher eine Umfrage durchgeführt, damit jeder die Benutzeroberfläche sieht und wir somit auch Meinungen zur Benutzbarkeit erhalten und anschließend eventuelle Verbesserungen am Design vornehmen können.

Es ist gewünscht, dass das System kompatible mit zukünftige Entwicklungen seien muss.Zum Beispiel, es muss ermöglicht werden, ein zukünftiges Projekt für TDU-Bibliothek in diesem Projekt zu integrieren. Dazu wird eine Webseite von unserem Webseite-Gruppe erstellt, da man administrieren kann.

Bei der Erwartung von dem aktualisierbaren System und Nachhaltigkeit, werden zukünftige Aspekte von einer Nextsteps-Gruppe untersucht

Das von uns erstellte System wurde getestet und es wurde festgestellt, dass die gewünschten Funktionen auch genau so funktionieren, wie es sein sollte. Beispielsweise wurde erwartet, dass man einen hilfsbedürftigen Studenten empfehlen kann: Dies funktioniert indem man die Matrikelnummer der Person in das zugehörige Textfeld eingibt. Durch das Testen wurde auch festgestellt, dass das erstellte System in gewünschter Qualität und Geschwindigkeit ist.

## Zielgruppen

An der Universität haben wir zwei Zielgruppen: Diese sind Studenten und Mitarbeiter.

**Studenten**

* Fast 2000 Studenten gibt es an der Universität
* Sie haben alle eine Studentenausweis und sie bezahlen in einem Mensa mit dieser Karte für Essen
* Die Studenten , die mit dem Shuttle-Bus fahren , verwendet für die Fahrt eine Shuttle-Karte und die Geld vom Kassierer nur an der Universität aufladen .

**Mitarbeiter**

* Fast 150 Mitarbeiter gibt es an der Universität
* Sie essen auch in der Mensa
* Diese Mitarbeiter müssen auch ein Account erstellen , um das System genau zu verwenden

## Aufbau des Projekthandbuches

In Kapitel 1 wurde die Planungsphase in 13 Teilen detalliert erklärt. Die Themen wie Ziele, Kontext, Organisation und Kommunikationsplan; werden mit Diagrammen und Tabellen unterstützt. Arbeitspakete und Aufgabenverteilung wurde erläutert. Kostenplanung und Risikoanalyse wurde gemacht.

In den Kapitalen 2.1.1, 2.2.1, 2.3.1, 2.4.1, 2.5.1 wurde die allgemeine Erläuterung des Projekts, die Problembeschreibung und die Ergebnisse der Literaturrecherchen als ähnliche Anwendungen gegeben.

In den Kapitalen 2.1.2, 2.2.2, 2.3.2, 2.4.2, 2.5.2 wurde erklärt, wie welche Methoden und Werkzeuge im Projekt verwendet werden.

In den Kapitalen 2.1.3, 2.2.3, 2.3.3, 2.4.3, 2.5.3 wurde die Anforderungsanalyse des im Projekt erstellten Systems detailliert durchgeführt.

In den Kapitalen 2.1.4, 2.2.4, 2.3.4, 2.4.4, 2.5.4 wurde die Design Funktionsweise mit Diagrammen erklärt.

In den Kapitalen 2.1.5, 2.2.5, 2.3.5, 2.4.5, 2.5.5 wurde detallierte Information über Fragebögen, Ergebnisse und Interpretation der Anwendung gegeben um die App zu testen.

In den Kapitalen 2.1.6, 2.2.6, 2.3.6, 2.4.6, 2.5.6 wurde eine Zusammenfassung des Projektes gegeben.

In Kapitel 2.6 wurden Nextsteps detaliert erklärt.

# Android

## **Methodische Ermittlung**

Die methodische Nachforschung hat mit der Festlegung von genauen Zielen angefangen und es wurden dafür Meilensteine erstellt. Danach folgte die Auswahl von Software und Hardware, die verwendet werden soll. Somit wurde festgelegt, dass als Software Android Studio, dessen Programmiersprache Java ist und GitHub für den Datenaustausch bestimmt. Für die Kommunikation wurden die Applikationen “Discord” und “WhatsApp” angewendet. Als Hardware wurde ein Android Smartphone verwendet.

Als die Ziele bis zur Evaluierungsphase erreicht wurden, wurde ein Umfragebogen erstellt, damit eine Ideenaustausch mit den Kunden gemacht werden kann. Dieser war sehr wichtig, um zu sehen, wo Verbesserungen oder Entwicklungen durchgeführt werden benötigt werden.

//Usability Scaling miteinbeziehen

## 

## Methoden und Werkzeuge

Android Studio ist ein offizielles Programmiertool für Android-Anwendungen mit allen essenziellen Funktionen. Aus diesem Grund wurde für die Programmierung, dem Erstellen der Benutzeroberfläche und der Anbindung ans Backend Android Studio verwendet. Um die Daten ans Backend anzubinden, wurden unterschiedliche Java Bibliotheken angewendet wie etwa: “Gson” oder “Json”.

Damit jeder Zugriff zum Quellcode hat wurde GitHub verwendet, denn GitHub ist ein Speicherdienst für Softwareprojekte. Damit jedes Projektmitglied simultanen Zugriff auf die Dokumente, wie etwa den Präsentationen, hat wurde Google Docs bzw. Slides verwendet.

Die Kommunikation unter unserem Arbeitsgruppe und zwischen anderen Arbeitsgruppen verlief über “WhatsApp” und “Discord”. Es wurde in Discord für unterschiedliche Aufgaben verschiedene Chaträume eingerichtet.

## Anforderungsanalyse und Konzeption

Das Problem ist, wie in Kapitel 2.1.1 bereits angedeutet, dass leider hilfsbedürftige unter sozialem Druck leiden, indem sie gezwungen sind finanzielle Hilfe bei der Universität selbst anzufordern.

Um den Druck aufzuheben und um hilfsbedürftige Menschen eine bessere Lebensbedingung innerhalb des Kampuses zu gewähren wurde diese Anwendung gestaltet.

Wer ein IT- oder Software-Projekt in Auftrag geben und erfolgreich umsetzen will, braucht eine klare Vorstellung der Anforderungen an das System. Die systematische Anforderungsanalyse führt diesen Prozess und legt die Basis für eine erfolgreiche Konzeption und Umsetzung.

In der Anforderungsanalyse werden in Zusammenarbeit mit den Entscheidungsträgern und den zukünftigen Anwendern folgende Fragen beantwortet:

* Wie ist die Arbeitsweise/Philosophie der Firma und der Anwender?
* Welchen Nutzen soll die Lösung bringen?
* Welche Schlüssel-Eigenschaften und -Funktionen muss die Lösung haben? Und was muss unbedingt vermieden werden?
* Wer soll mit dem System was machen und wie?
* Wie sollen die Arbeitsabläufe (Workflows) aussehen?

**Wie ist die Philosophie der Anwender ? :** Die Arbeitsweise von Anwender ist eine innovative Campus für die Universitäten. Mit einigen praktischen Innovationen können die Lebensbedingungen von Studenten auf dem Campus besser sein. Studenten verbringen eine großes Teil von ihren Zeit auf dem Campus und deshalb haben sie viele Bedarf zu decken wie Essen , Fahren oder für soziale Aktivitäten. In diesem Projekt möchten wir für unsere Universität ein online Zahlungssytem und Aufladenssytem für diese Optionen machen. Mit diesem System können dies Studenten auf ihren Shutle-Karte oder Mensaakrte mit einer Android-Applikationen Geld aufladen und mit dieser Karte können sie mit dem Shuttle-Bus fahren.

**Welche Nutzen soll die Lösung bringen ? :** Durch die Android App werden einige Probleme für Studenten und für unseres Campus gelöst . Diese Lösungen unterscheiden sich zwei Klassen . Eine von dieser Lösungen für unsere Mensa und die andere Lösung ist für unsere Shuttle-Bus. Für dieser Prozess vorher an der Universität werden die Automaten verwendet und diese Automaten sind nicht effizient wegen ihrer Wartungszeit und ihrer begrenzter Funktionen.

**Lösungen für Mensa der Universität**

* Die Wartungszeit wird durch Android App fürs Aufladen von Geld für die Bezahlen der Essen und für Shuttle-Bus verkürzt.
* Geld Aufladen Interaktionen werden überall und kostenlose über Smartphones durchgeführt.
* Studenten und Mitarbeiter können ihre Karte Geld aufladen , wie viel Geld sie aufladen möchten .
* Die Android App wird effizienter und schneller als Automaten für Geld Aufladen und Geld Bezahlen.
* Mit Free-Meal Optionen können die bedürftigen Studenten mit dem Spendengeld ihre Essen bezahlen.

**Lösungen für Shuttlebus der Universität**

* Die Shuttle Karte werden durch die Android App mit Geld aufladen
* Durch die App können die Studenten und Mitarbeiter die Fahrt über ihre Smartphone bezahlen.

**Wer soll mit dem System machen und wie ?**

Dieser System wird für die Universität und ihre Mitglieder erstellt. Um diese App zu verwenden , soll man zunächst die Android App über Google Play Store herunterladen und sollen sie für eine eigene Account erstellen . Dann können sie ihre Karte für Mensa und Shuttle ohne Automaten zu brauchen mit Geld aufladen und ihre Bezahlung für Shuttle und Mensa durchführen. App haben keine Nutzungsgebühr deshalb ist sie auch kostenlos und erreichbare für alle, die eine Smartphone haben. Für die Erstellung der App haben die Studenten der TDU gearbeitet und die Benutzeroberfläche und die Optionen für Bezahlung und Aufladen erstellt.

**Schlüsseleigenschaften des Projektes**

* Die Aufladen und Bezahlungssystem für Mensa und Shuttlekarte
* Die Studenten und Mitarbeiter müssen eine Account für die Verwendung der App
* Die Geld werden durch die Android App aufladen

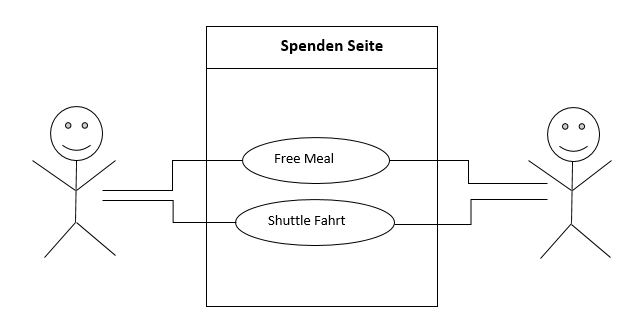
**Wie sollen die Arbeitsabläufe aussehen ?**

Die Arbeitsabläufe des Projektes können sie im **Kapitel 1.8 Arbeitsspezifikationen** detaillierter sehen.

## Design (Entwicklung) und Implementierung

## Anwendungsfälle

*Abbildung 3.1 Android Login Seite*

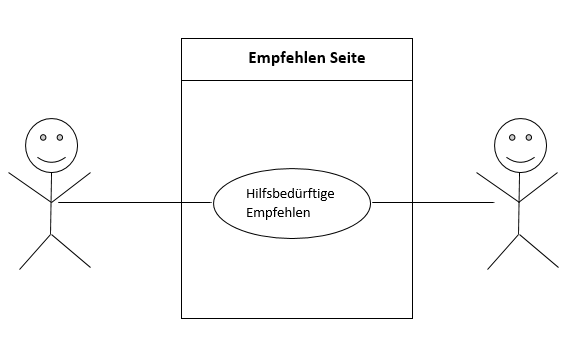
****

*Abbildung 10.*

*Abbildung 10.*

*Abbildung 10.*

*Abbildung 3.1 Android Spenden Seite.*

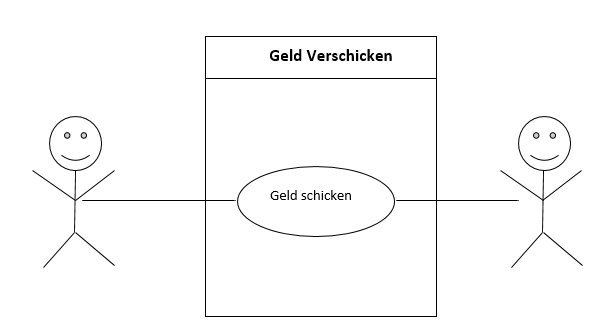
****

*Abbildung 11.*

*Abbildung 11.*

*Abbildung 11.*

*Abbildung 3.2 Android Empfehlen Seite.*

****

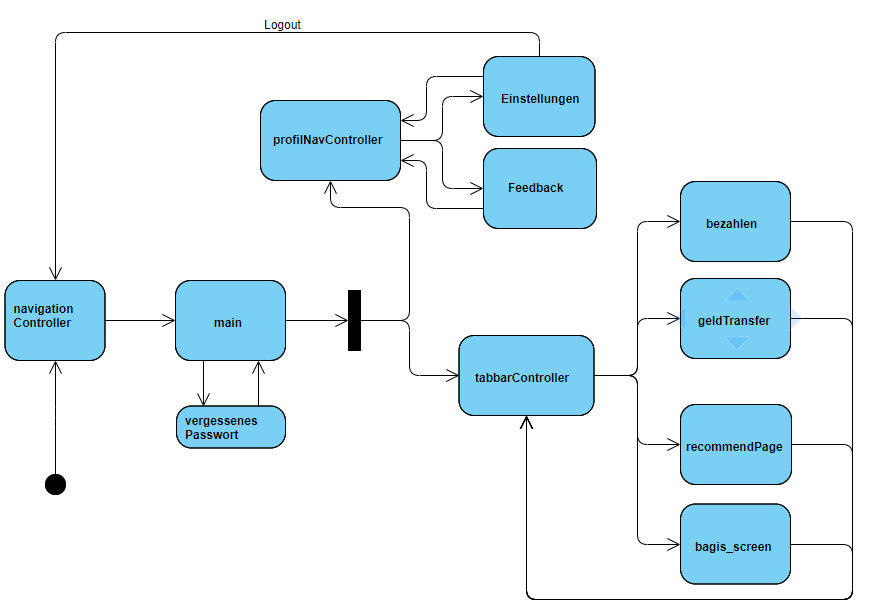
*Abbildung 12.*

*Abbildung 12.*

*Abbildung 12.*

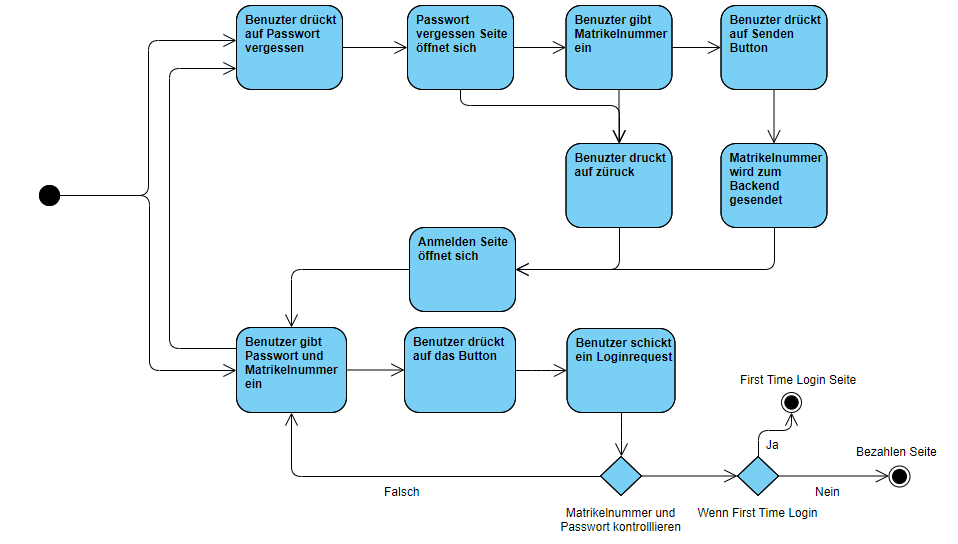
*Abbildung 3.3 Android Geld verschiken*

## EER und UML Domänenmodell

1. **Allgemein**

*Abbildung 4.1 Android Allgemeiner Aktivitätsdiagramm*

1. **Login-Seite**

****

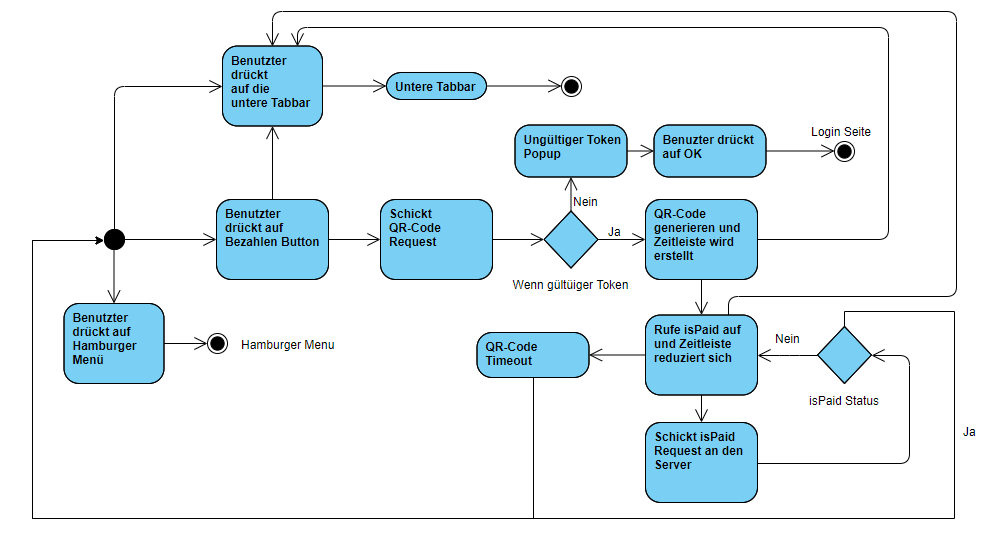
*Abbildung 13.*

*Abbildung 13.*

*Abbildung 13.*

*Abbildung 4.1 Android Login Seite Aktivitätsdiagramme*

1. **Bezahlen**

****

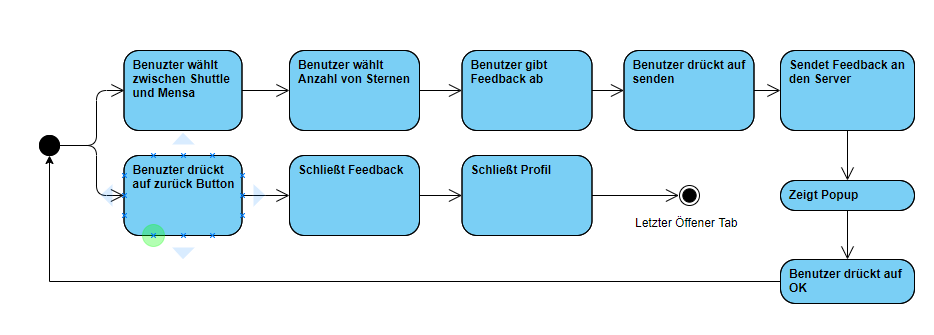
*Abbildung 14.*

*Abbildung 14.*

*Abbildung 14.*

*Abbildung 4.2 Android Bezahlen Aktivitätsdiagramme.*

1. **Feedback**

****

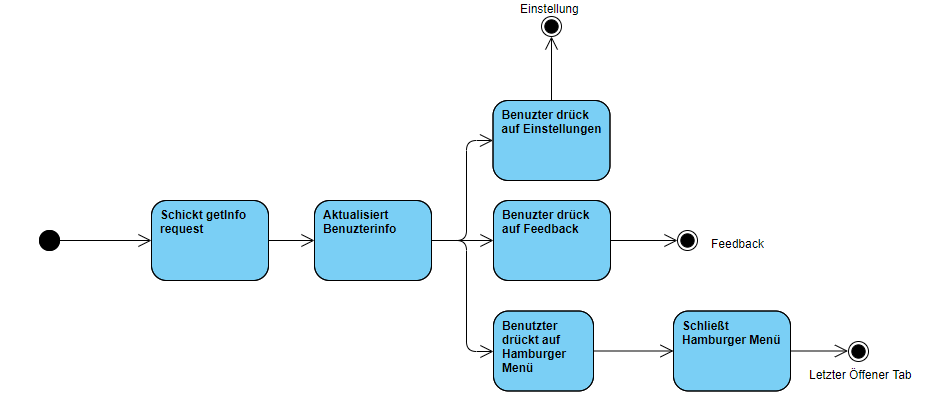
*Abbildung 15.*

*Abbildung 15.*

*Abbildung 15.*

*Abbildung 4.3 Android Feedback Aktivitätsdiagramm*

1. **Profil**

****

*Abbildung 16.*

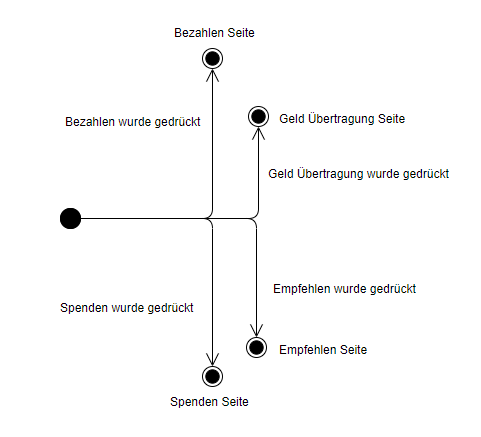
*Abbildung 16.*

*Abbildung 16.*

*Abbildung 4.4 Android Profil Aktivitätsdiagramm.*

*.*

1. **Tabbar**

****

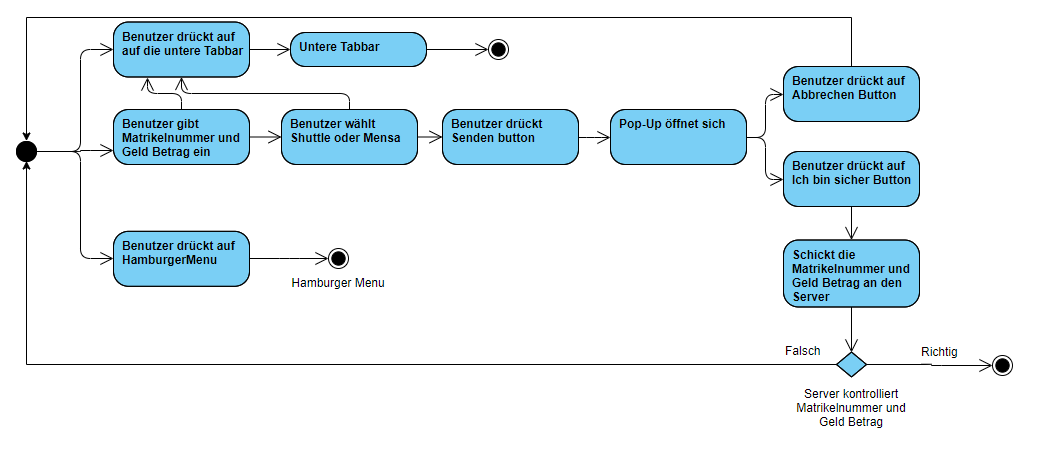
*Abbildung 17.*

*Abbildung 17.*

*Abbildung 17.*

*Abbildung 4.5 Android Tabbat Aktivitätsdiagramm*

1. **Geld Transfer**

****

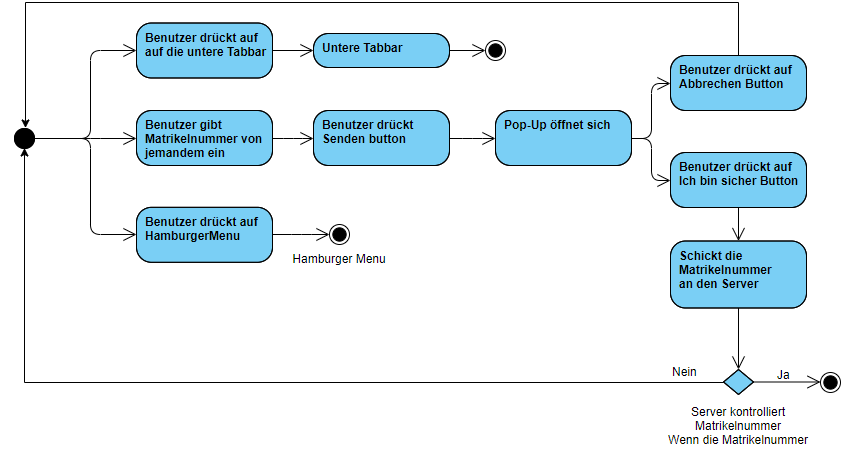
*Abbildung 18.*

*Abbildung 18.*

*Abbildung 18.*

*Abbildung 4.5 Android Geld Transfer Aktivitätsdiagramm.*

1. **Empfehlen**

****

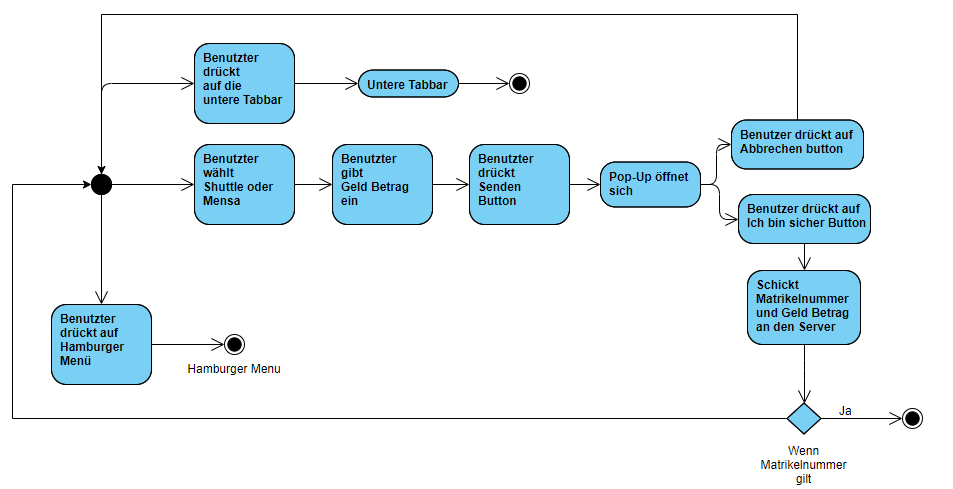
*Abbildung 19.*

*Abbildung 19.*

*Abbildung 19.*

*Abbildung 4.6 Android Empfehlen Aktivitätsdiagramm*

1. **Spenden**

****

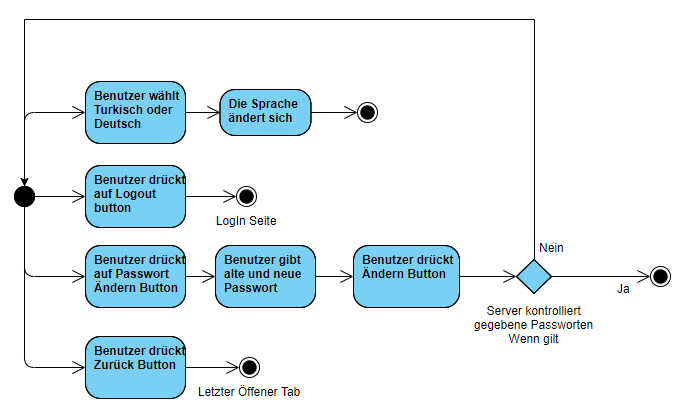
*Abbildung 20.*

*Abbildung 20.*

*Abbildung 20.*

*Abbildung 4.7 Android Spenden Aktivitätsdiagramm*

1. **Profil**

****

*Abbildung 21.*

*Abbildung 21.*

*Abbildung 21.*

*Abbildung 4.8 Android Profil Aktivitätsdiagramm*

## Funktionale und nichtfunktionale Anforderungen

| Funktionale Anforderung | Umsetzung |
| --- | --- |
| Der Nutzer kann einer unbekannten Person ein “Free Ride” ausgeben. Der Gehalt wird dabei vom eigenen Konto abgezogen. | Ein Button wird dazu dienen, die Transaktion als Button-Event auszuführen |
| Der Nutzer kann durch die Matrikelnummer dem System eine bestimmte Person empfehlen. | Die Matrikelnummer wird in die Datenbank eingetragen. Bei genug Empfehlungen wird der Schüler zur Liste der Hilfsbedürftigen hinzugefügt. Dies wird als Button-Event implementiert |
| Der Benutzer kann mit der Applikation in der Mensa/Shuttle bezahlen | Durch das Einlesen eines QR-Codes im “Bezahlen” Bereich kann die Bezahlung durchgeführt werden |
| Der Benutzer kann einer unbekannten Person ein “Free Meal” ausgeben. Der Gehalt wird dabei vom eigenen Konto abgezogen | Ein Button wird dazu dienen Transaktion als Button-Event auszuführen. |
| Dem Nutzer soll es ermöglicht werden, Geld des Shuttle- und Mensa Kontos an andere Nutzer zu versenden | Das Geldtransfer-Fragment: Durch das Klicken auf das Senden Buttons auf der Geldtransfer Seite wird der angegebene Betrag gesendet |
| Die Kunden können ihr Kontostand kontrollieren. | Eine Side-Bar wird dazu dienen,die Informationen über den Kontostand auszugeben..In diesem Side-Bar sind vorhanden:  Name,Vorname  Matrikelnummer  Mensa-Kontostand  Shuttle-Kontostand |
| Die Studierenden möchten möglicherweise ihre Wünsche, Meinungen und Vorschläge bezüglich des Essens und der Shuttlefahrten mitteilen. | Mit dem Feedback-Button können die Studierenden ihre Verbesserungsvorschläge für das Essen oder der Shuttlefahrten mitteilen. |
| Die Funktionalität der App muss hoch sein. Um dem Benutzer Freiheit, Dynamik und Anpassbarkeit über die Benutzungsoberfläche zu gewähren, wird ein Bereich zum Einstellen der Präferenzen benötigt | Mit dem ‘’Einstellungs-Button’’ können die Studierenden:  Ihr Passwort ändern,  Die Sprache auswählen,  sich abmelden |

Tabelle 24: Funktionale Anforderungen

| Nichtfunktionale Anforderung | Anforderung | Umsetzung |
| --- | --- | --- |
| Funktionale Eignung | Jegliche Funktionen müssen bereitgestellt sein | Es werden nur Funktionen im Rahmen der Anforderungsanalyse hinzugefügt |
| Zeitverhalten | Die Reaktionszeit der App darf nicht zu lang sein | Es werden Speicher- und Laufzeiteffiziente Algorithmen verwendet und Gebrauch vom Cache des Smartphones gemacht |
| Kompatibilität | Die Applikation ist mit allen weit verbreiteten Android Versionen kompatibel | Es wird eine geeignete Android API verwendet, die die meist verwendeten Systeme abdeckt |
| Attraktivität | Der im Meeting am 12.03.19 mündlich vereinbarte Style-Guide wird eingehalten | Die Design Funktionen von Android Studio werden dem Style-Guide nach benutzt |
| Benutzereingabe Sicherheit | Bestätigungsfragen werden gestellt bei Aktionen, die nicht rückgängig gemacht werden können | Programmiertechnische Methoden werden Implementiert, die diese Funktion ausführen |
| Bedienbarkeit | Schritte können jederzeit zurückgesetzt werden (außer nach Bestätigung) | Programmiertechnische Methoden werden Implementiert, die diese Funktion ausführen |
| Fehlertoleranz | Fehleingaben müssen korrekt behandelt werden | Bei Fehleingaben werden Warnungen seitens der Applikation gegeben |

Tabelle 25: Nichtfunktionale Anforderungen

## **Systemarchitektur***Abbildung 5.1 Systemarchitektur von Sidebar**Abbildung 5.2 Systemarchitektur von Freeride & Freemeal, Empfehlen*

*Abbildung 22.*

*Abbildung 22.*

## 

## 

## 

*Abbildung 27.*

*Abbildung 26.*

*Abbildung 25.*

*Abbildung 24.*

*Abbildung 23.*

*Abbildung 22.*

*Abbildung 28.*

*Abbildung 27.*

*Abbildung 26.*

*Abbildung 25.*

*Abbildung 24.*

*Abbildung 23.*

*Abbildung 28.*

*Abbildung 27.*

*Abbildung 26.*

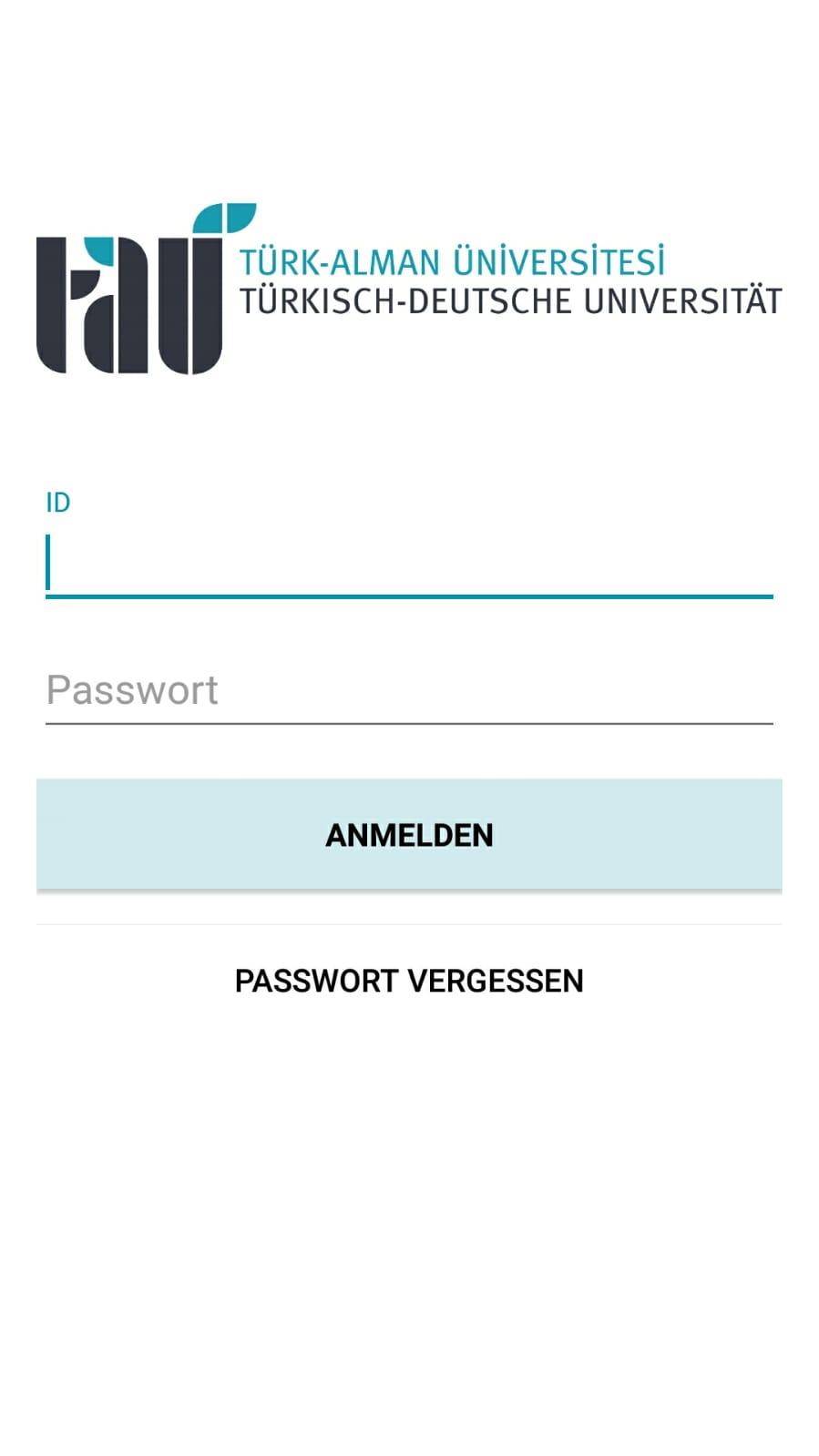
*Abbildung 25.*

*Abbildung 24.*

*Abbildung 23.*

## Design-Layouts

In Abb 6.1, 6.2 ,6.3 und 6.4 sehen sie die Ergebnisse von der Erstellten Benutzeroberfläche. Da das erstellte System zweisprachig ist kann man die Sprache auch verändern. Die Abb. wurde in Turkish angezeigt.



*Abbildung 28.*

*Abbildung 29.*

*Abbildung 29.*

*Abbildung 6.1 Android Design-Layouts. (Anmelden)*

Um sich einzuloggen, müssen sie den ID nummer und den Passwort, welche Sie von der Universität bekommen haben einfüllen. Wenn alle Daten stimmen, wird die Verbindung aufgebaut und Sie werden zur Hauptseite weitergeleitet.

****

*Abbildung 29.*

*Abbildung 30.*

*Abbildung 30.*

*Abbildung 6.2 Android Design-Layouts. (Spenden)*

Abb.6.2 ist der Benutzeroberfläche um einen hilfsbedürftigen eine Mahlzeit oder kostenlosen Fahrt mit dem Shuttle zu spenden.

Durch den ToggleButton kann der Spender sich auswählen ob Sie eine Mahlzeit oder eine Fahrt spenden möchte.

Über den Button “Bestätigen” wird der Aktion durchgeführt.



*Abbildung 30.*

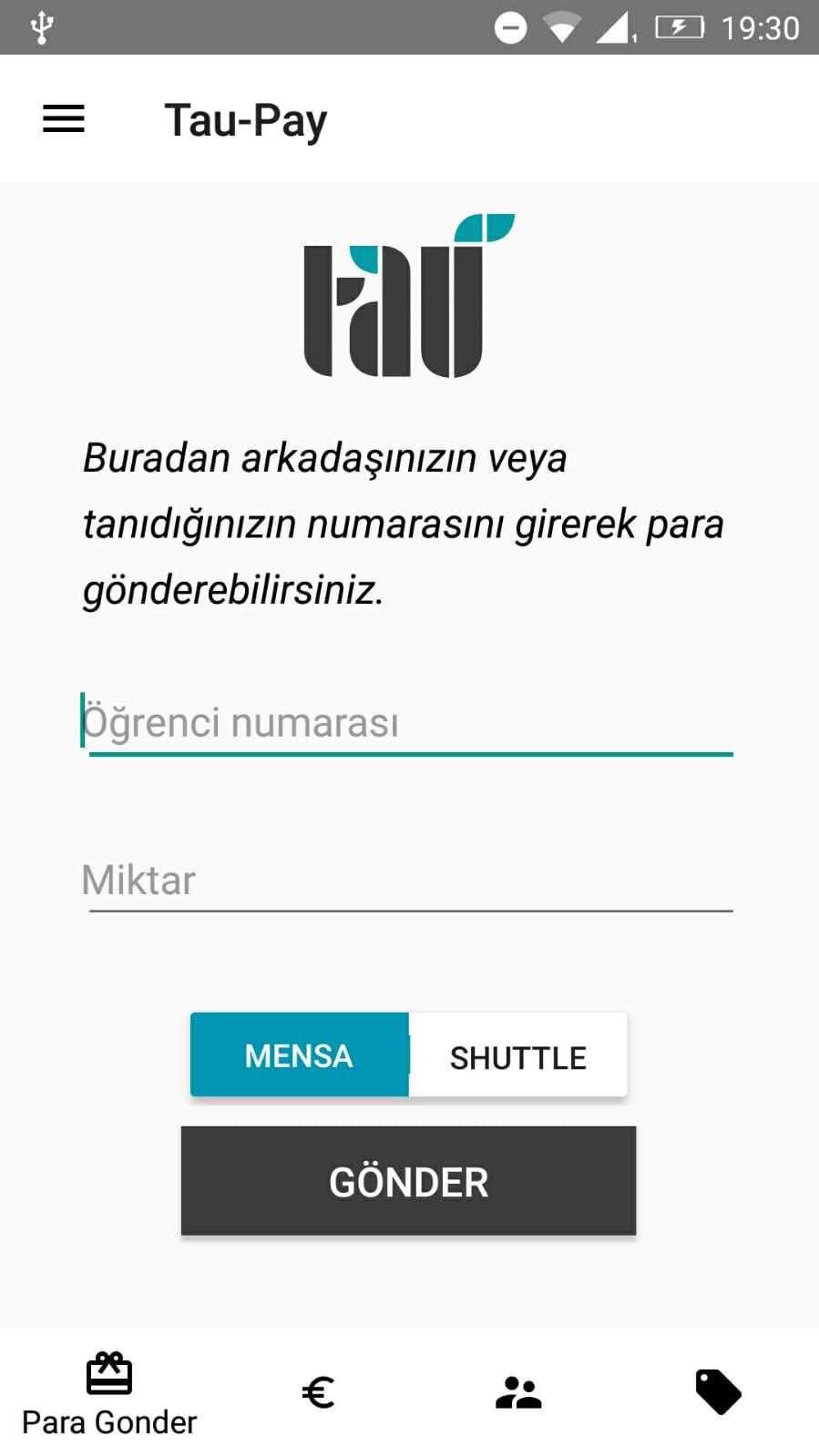
*Abbildung 31.*

*Abbildung 31.*

*Abbildung 6.3 Android Design-Layouts. (Empfehlen)*

In Abb. 6.3 sehen Sie die Benutzeroberfläche um einen Hilfsbedürftigen empfehlen zu können. Hierzu muss man die Matrikelnummer des Studenten, der zu empfehlen ist, in das Textfeld schreiben.

Durch den “Bestätigen” Button wird diese Information an die Datenbank übertragen.



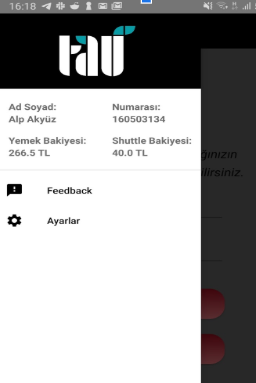
*Abbildung 31.*

*Abbildung 32.*

*Abbildung 32.*

*Abbildung 6.4 Android Design-Layouts. (Geld senden).*

Das System erhaltet auch ein “Geld senden” Bereich,welche Sie in Abb. 6.4 sehen. Dort können irgendein Person, dessen Matrikelnummer Sie wissen, Geld senden. Diese Aktion folgt, indem Sie die Nummer des Studenten und die Anzahl der zu spendenden Entität ein. Durch den Button “Gönder” wird die Aktion durchgeführt.



*Abbildung 32.*

*Abbildung 33.*

*Abbildung 33.*

*Abbildung 6.5 Android Design-Layouts. (Sidebar)*

Das System erhaltet noch einen Teil, in dem Studenten und Mitarbeiter an der TDU ihre Kontoinformationen sehen können . Die Kontozustand und einige Informationen wie Vorname,Name,Matrikelnummer,Budget des Kontos für Shuttle-Karte und Mensakarte werden in einem Side-Bar dargestellt und die Daten werden mit Datenbank verbunden. Dadurch Studenten und Mitarbeiter immer die aktualisierte Informationen in Side-Bar sehen , wenn sie z.B. für ihre Karte Geld aufladen oder mit der Karte bezahlen. Dieser Informationen können sie in der Abbildung 6.5 sehen.

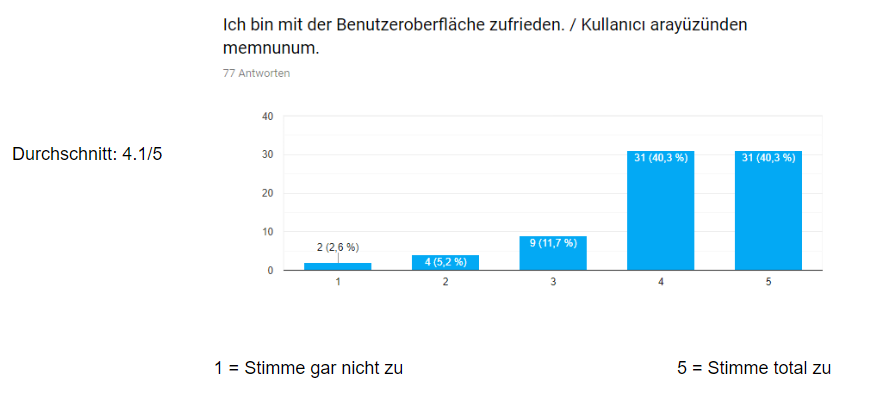
Funktionen der Side-Bar;

* Einstellungen Button
* Feedback Button
* Die Zustand des Kontos
* Persönliche Informationen von Benutzern

## Evaluation

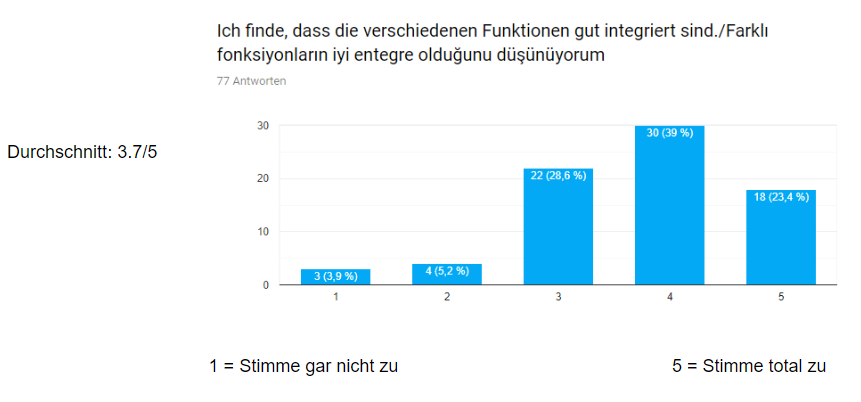
Da, dass Erstellte-System an unserer Universität verwendet wird, war für uns der Beurteilung von den Studenten, Lehrpersonale und Sicherheits-oder Reinigung Offizier eine sehr wichtige Thema.Darauf haben wir einen Umfragebogen erstellt, indem sich auch ledigliche Screenshots von der Anwendung befinden.

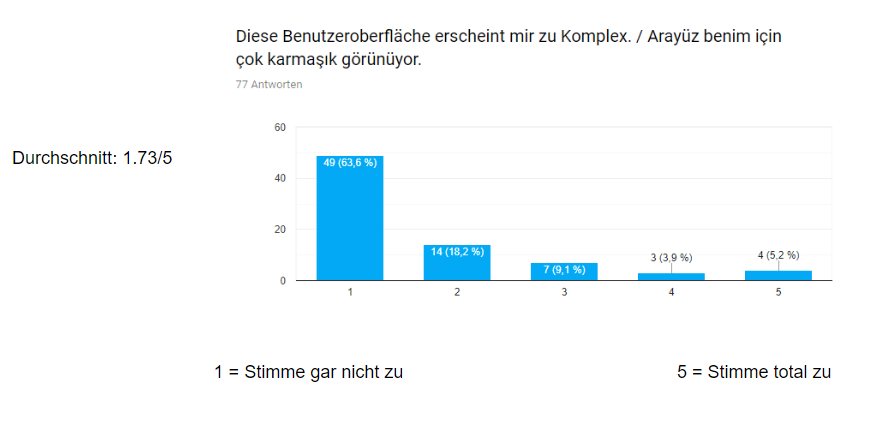
Der Umfragebogen wurde anhand des “SUS-Fragebogens” durchgeführt und wurde in Google-Formular erstellt.Den Umfragebogen haben 72 Personen ausgefüllt.

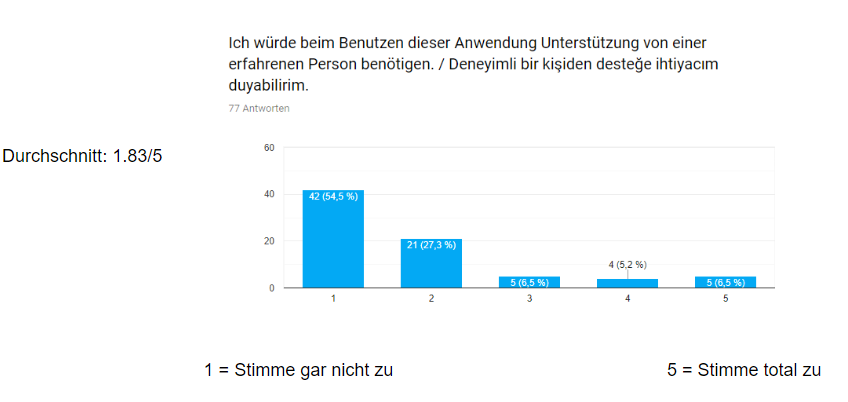
****

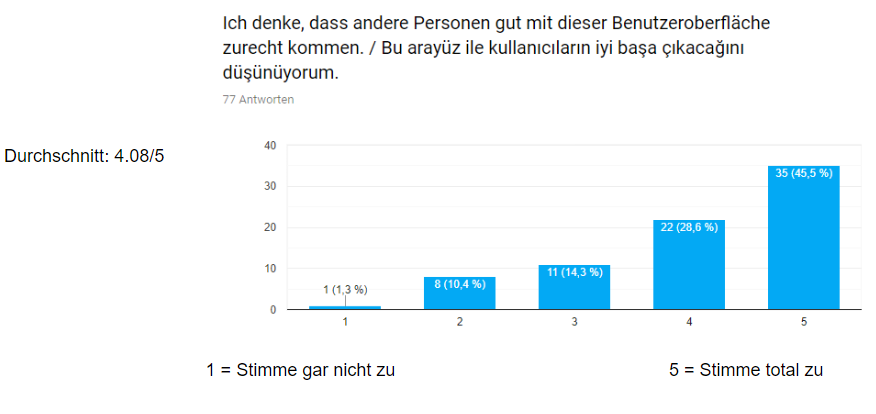
*Abbildung 33.*

*Abbildung 7.1 Ergebnis von der 1. Frage*





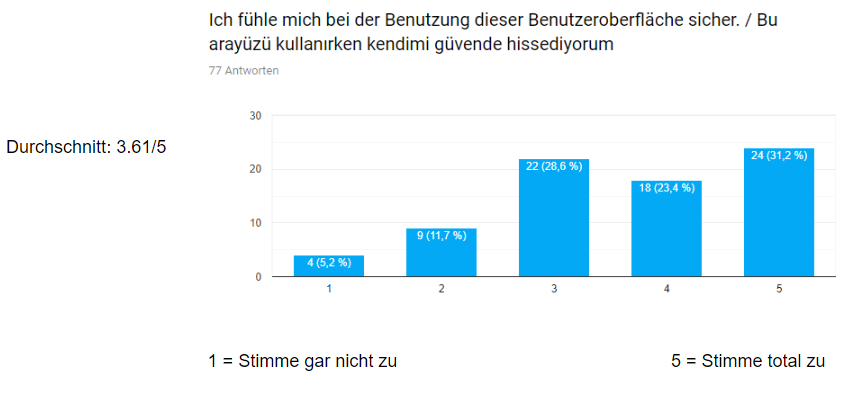
*Abbildung 7.2 Ergebnis von der 2. Frage* 

****

*Abbildung 34.*

*Abbildung 35.*

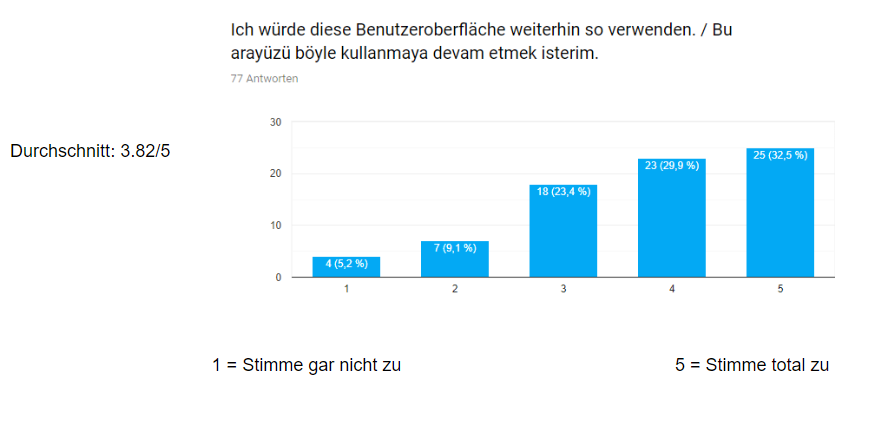
*Abbildung 7.3 Ergebnis von der 3. Frage*

**

*Abbildung 35.*

*Abbildung 36.*

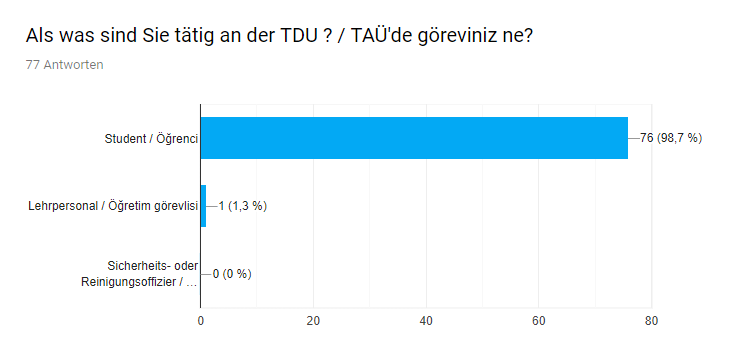
*Abbildung 7.4 Ergebnis von der 4. Frage*

****

*Abbildung 36.*

*Abbildung 37.*

*Abbildung 7.5 Ergebnis von der 5. Frage*

*Abbildung 7.6 Ergebnis von der 6. Frage* 

## Versuchsaufbau und Versuchsdurchführung

Um den Versuch bzw. das System zu realisieren benötigten wir einen Laptop mit Android Studio. Weil Android Studio in der Programmiersprache Java programmiert wird, benötigten wir Kenntnisse über Java. Um die erstellte Anwendung zu testen, brauchten wir ein Android Smartphone. Damit jedes Mitglied Zugriff auf den erstellten Code hat, musste jeder einen Github Konto besitzen.

Und um die Anwendung zu evaluieren und ein Usability Scaling durchzuführen, haben wir durch Google-Forms ein Umfrageformular erstellt, welches dann über Online WhatsApp Gruppen verteilt wurde

Die Versuchsdurchführung geschah indem in Android Studio zuerst eine Benutzeroberfläche mit den geeigneten Komponenten, wie Button, Textfeld usw., gestaltet wurde. Damit die Komponenten auch eine Funktion haben, also damit Sie auf die Eingaben reagieren können und einen Ausgabe geben können, wurde der geeignete Code geschrieben.

Es folgte somit die Testphase bei der jegliche Eingaben getestet und geschaut wurde, ob es eine Ausgabe gibt, und ob sie auch an den Backend gesendet werden.

→ Referenz Video

## Ergebnisse

Als Ergebnis wurde eine Anwendung dargestellt, in welcher auch alle Funktionen implementiert wurden. Es wurden alle Anforderungen, die unsere Ziele waren, der Reihe nach getestet und es wurde klar das jegliche Funktionen erwartungsgemäß funktionierten. Somit wurde festgestellt, dass alle Anforderungen erfüllt sind.

Im Zuge von der Richtlinie ISO 9241-110, wurde die Selbstbeschreibungsfähigkeit eingehalten und es wurde auf die Lernförderlichkeit geachtet. Außerdem wurden auch die Transaktionen zwischen den Schnittstellen erfolgreich durchgeführt. Und um Fehler zu vermeiden, werden die Log-Dateien in der Datenbank gespeichert und der Steuerbarkeit für die Benutzer ist begrenzt. Da die Dateien in der Datenbank abgespeichert werden, wurde gesehen, dass der Backend Teil richtig funktioniert.

## Interpretation der Ergebnisse

Die erstellte Applikation hat sehr viele Vorteile z.B. ist es sehr dynamisch und kann noch weiter erweitert werden. Man kann nach Wunsch oder Bedürfnis noch andere Anwendungen/ Funktionen erstellen und Implementieren.

Es war sehr wichtig, dass das System, auf die Eingaben in Echtzeit reagiert, denn somit kann eine komfortable Benutzung ermöglicht werden.

Auf die Sicherheit im Frontend wurde sehr geachtet: Jegliche Aktionen müssen ein zweites Mal vom Nutzer bestätigt werden, sodass die Anzahl der Fehleingaben möglichst minimiert wird.

## Zusammenfassung und Ausblick

Das Ziel war ein System zu entwickeln, welches das Leben auf der Universität einfacher machen sollte. Es war offensichtlich, dass ein solches System sehr nützlich sein konnte und auch realisierbar war.

Nach der Festlegung des Designs mit den anderen Gruppen haben wir mit der Implementierung angefangen. Die Umsetzung des Projekts lief nach Plan und wir konnten alle Funktionalitäten, die im Plan gestanden haben, auch ohne Probleme realisieren.

Die Resultate im Kapitel Evaluierung zeigen, dass ein solches System viele Vorteile hat und auch durch Anwender gut empfangen wird.

Natürlich hat das Projekt auch sehr viel Potential zur Erweiterung. Was eigentlich als eine Idee für die mobile Bezahlung in der Mensa angefangen hat, könnte erweitert und für viele andere Zwecke innerhalb der Universität genutzt werden, unter anderem für die Bibliothek und für die Kantine.

# IOS

## **Methodische Ermittlung**

IOS-unterstützte App-Programme wurden untersucht, da das gewünschte System eine Anwendung für iOS-basierte Geräte ist. Es wurde entschieden, dass XCode das am besten geeigneten Programm für das gewünschte System ist, und dann wurde XCode untersucht. Xcode ist eine integrierte Entwicklungsumgebung für MacOS, die von Apple für die Entwicklung von Software für MacOS, iOS, watchOS und TVOS entwickelt wurde. Beinhaltet Software-Entwicklungstools. Das XCode-Programm verwendet die Programmiersprache Swift. Swift ist eine neue Programmiersprache, die von Apple auch entwickelt wurde, um iOS- und Mac-Anwendungen zu entwickeln. Dann wurde methodisch untersucht, wie diese Programmiersprache ist. Darüber hinaus wurde entschieden, welche Art von Codesystem und Teamkommunikation erstellt werden soll. Die Verknüpfungen der Codes sollten über GitHub und die Teamkommunikation über Discord erfolgen. Die Präsentationen werden über Google Slides vorbereitet, damit sich jeder leicht vorbereiten kann.

## Methoden und Werkzeuge

Der Code wurde im XCode-Programm mit der Programmiersprache Swift geschrieben, die für das Schreiben von Code festgelegt wurde. In diesem Programm wurde die Anwendung entworfen. Das Design wurde in Anlehnung an das Android App Design entwickelt und es wurde eine Zufriedenheitsumfrage zum Design durchgeführt. Die zu schreibenden und zu erstellenden Abschnitte wurden unter den Teammitgliedern aufgeteilt. Das Pull-Commit-Push-System in GitHub wurde verwendet, um diese Abschnitte auf Teammitglieder aufzuteilen. Wenn dieses System kurz erklärt wird, wird der letzte Code mit dem Befehl "Pull" in den lokalen Code integriert, bevor eine Änderung am allgemeinen Code vorgenommen wird. Nachfolgende Änderungen werden in "Commit" geschrieben und auf GitHub übertragen. Zuletzt wurde der aktuelle Code mit dem Befehl "Push" integriert und mit dem Hauptcode in GitHub aktualisiert.

Die Treffen für den Fortschritt der Studie und die getroffenen Entscheidungen wurden durch Discord getroffen. Mit dem Discord können Sprach- und Videoanrufe getätigt und der Bildschirm gemeinsam genutzt werden. Auf diese Weise konnten sie schnell kommunizieren und Entscheidungen treffen und wurden auf andere Mitglieder aufmerksam.

Darüber hinaus wurde mithilfe von Google Slides, mit den mehrere Personen gleichzeitig Änderungen online vornehmen können, die Präsentationsfunktion so eingerichtet, dass sie nicht auf mehrere Personen geladen werden kann, und anschließend wurde eine allgemeine Kontrolle durchgeführt.

Diese Werkzeuge wurden allgemeine beim Erstellen und Entwickeln des Systems verwendet:

• Programmierung: XCode und Swift

• Koordination: GitHub und Google Slides

• Kommunikation: Discord

## Anforderungsanalyse und Konzeption

Die zu erstellende Anwendung ist ein von iOS unterstütztes Universitäts-Mensa- und Shuttlesystem. Mit dieser App kann Geld für die Mensa oder den Shuttle der Universität aufgeladen und bezahlt werden. Darüber hinaus kann der Benutzerbilanz auch aktuell angezeigt werden. Darüber hinaus kann die Mahlzeit- oder Shuttle-Optionen verwendet werden, um anderen Benutzern oder Benutzern, die Hilfe benötigen, zu helfen. In diesem System, das von der Benutzersicherheit unterstützt wird, kann über die Mensa oder das Shuttlesystem abgestimmt und eine Rückmeldung hinterlassen werden.

Auf diese Weise können die Mensa und das Shuttle-System der Universität praktisch genutzt werden. Darüber hinaus wird der Geldladekiosk oder die Geldladestelle für den Mangel an Geld von der Suche entfernt und kann Geld von überall aufladen.

Außerdem Kunden haben das genaue System mit Substanz, das sie erstellen möchten, erklärt. Es wurde festgelegt, was mit dem vom Kunden angeforderten System gemacht werden kann und was nicht. Diejenigen, die im gewünschten System arbeiten wollen und diejenigen, die nicht im System vorhanden sind, werden wie folgt geschrieben:

## Design (Entwicklung) und Implementierung

## Anwendungsfälle

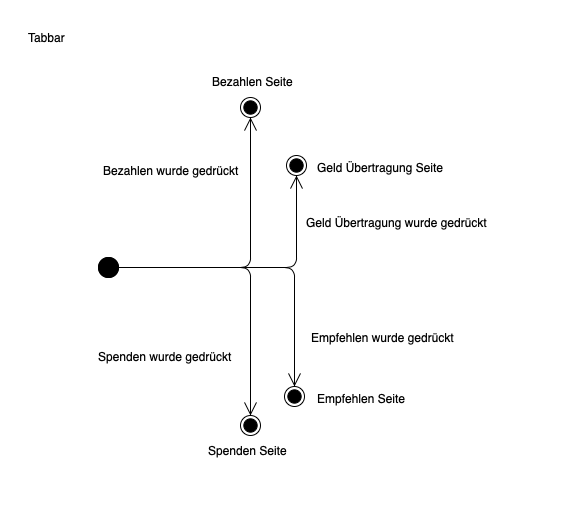
Studenten, Mitarbeitern und Gasten in TDU wird mit ihre Apple Handy auf diese App in ihrer Konto einloggen. Für Mensa wird Man maximal 2.5 Minuten früher für Mittagessen bezahlen. QR Code wird generiert und sie können es auf Barcode scanner zeigen.

Studenten und Mitarbeitern von TDU wird von Internet auf ihre Konto in der App Geld aufladen. Wenn sie auf den Shuttle einsteigen, werden sie QR Code auf der Barcode Scanner zeigen. Jeder Fahrt wird 2.20 TL aus ihre Konto bezahlen. Es wird mit dieser Methode keine spezifische Karte für Shuttle sein.

Wenn eine Student seine Freund oder eine Student die wenig Möglichkeit hat, helfen möchten, wird er den Option FreeMeal oder Free Ride wählen und eine Meal oder eine Shuttle Fahrt tzur diese Person schicken. Diese Person, die Free Meal nehmen, wird via eine Nachricht von informiert.

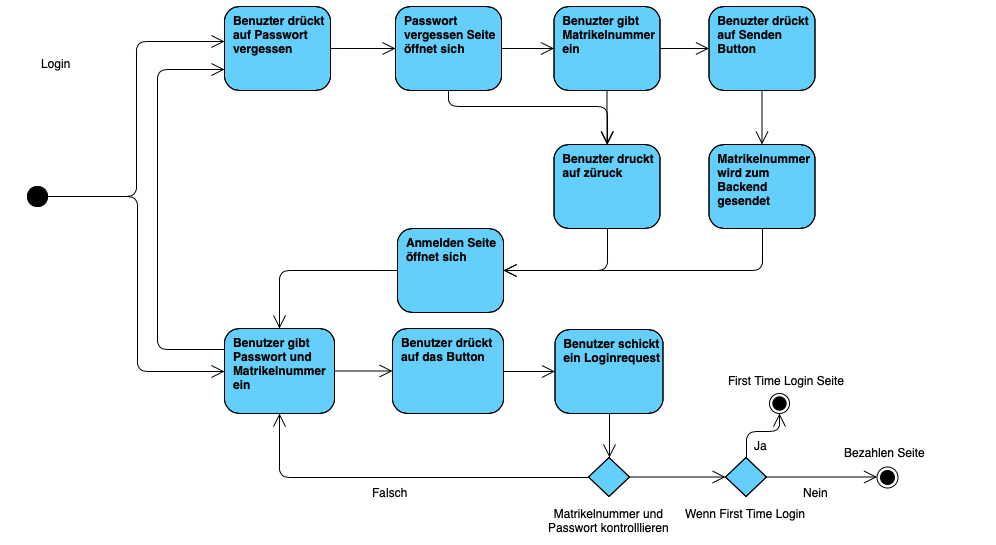
## EER und UML Domänenmodell

*Abbildung8.1 Klassendiagramm*

*****Abbildung 8.2 IOS Tabbar*

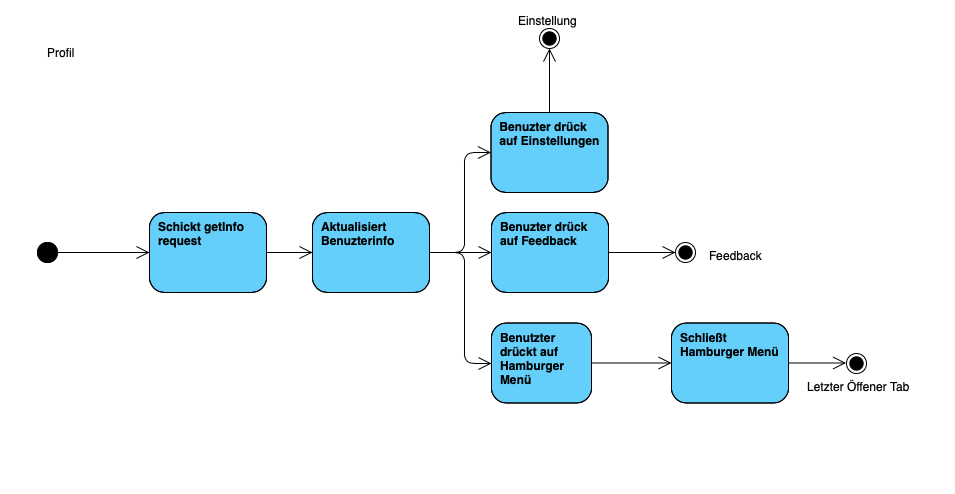
*Abbildung 37.*

*Abbildung 38.*

*****Abbildung 8.3 IOS Login Aktivitätsdiagramm*

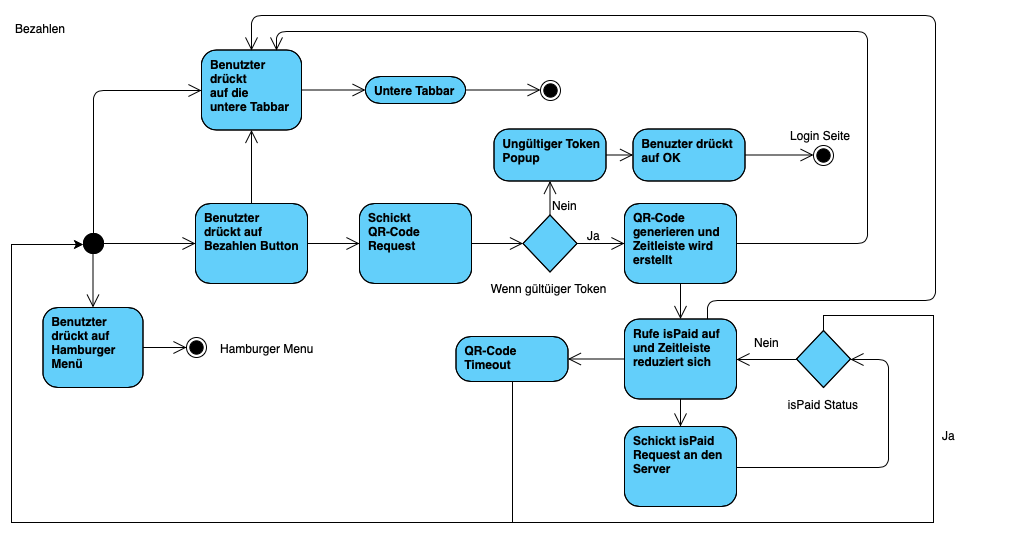
*Abbildung 38.*

*Abbildung 39.*

*****Abbildung 8.4 IOS Proifl Aktivitätsdiagramm*

*Abbildung 39.*

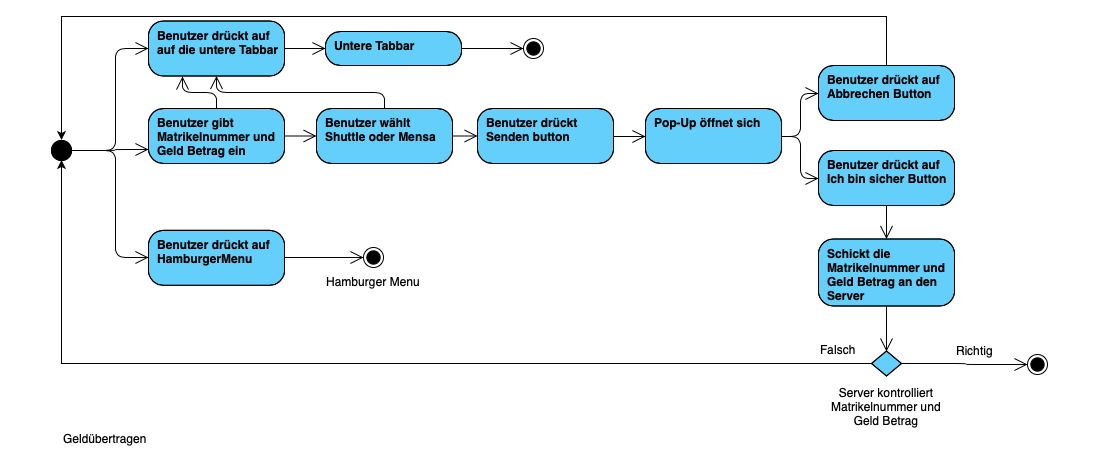
*Abbildung 40.*

****

*Abbildung 40.*

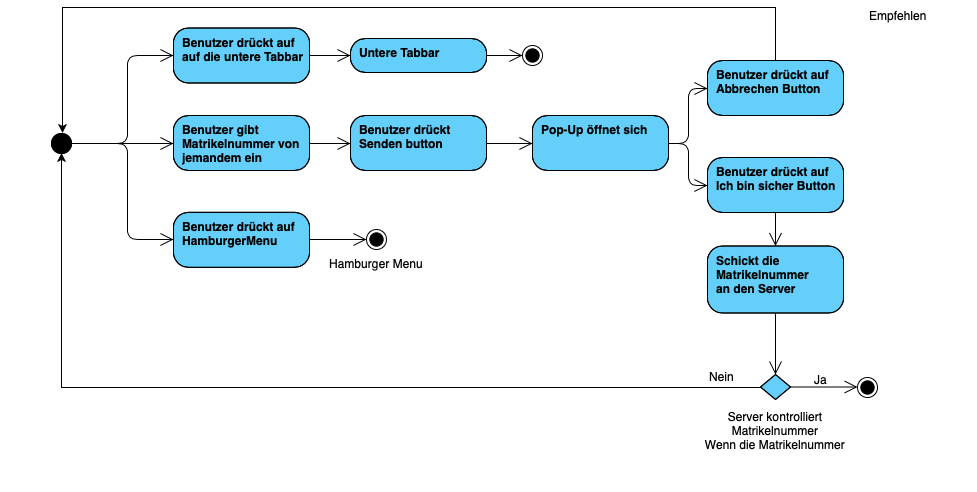
*Abbildung 41.*

*Abbildung 8.5 IOS Bezahlen Aktivitätsdiagramm*

*****Abbildung 8.6 IOS Geldübertragen Aktivitätsdiagramm*

*Abbildung 41.*

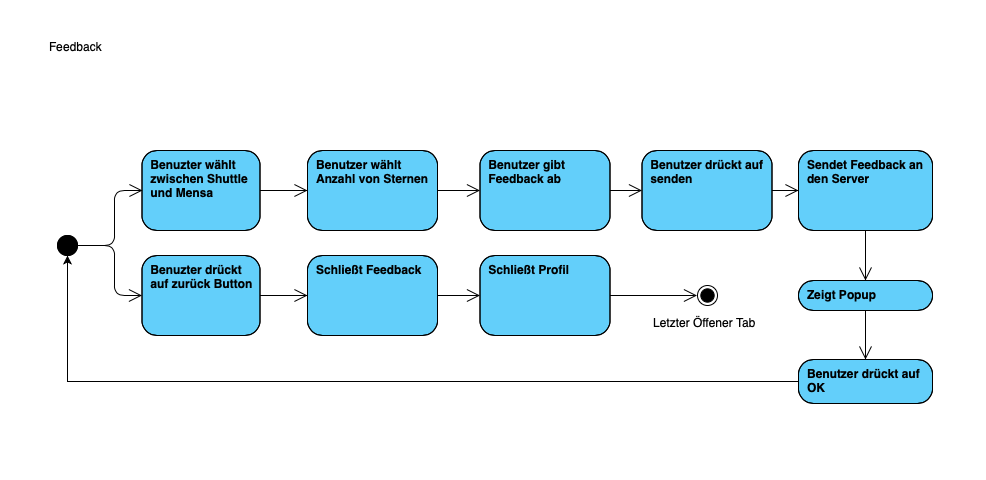
*Abbildung 42.*

****

*Abbildung 42.*

*Abbildung 43.*

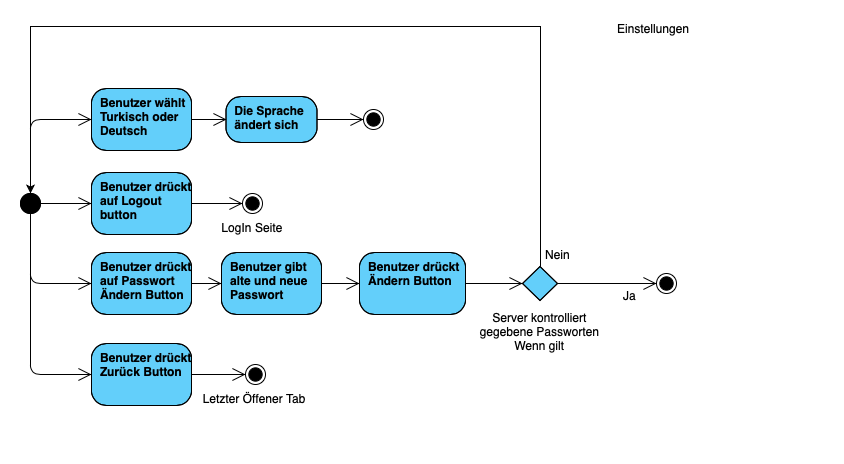
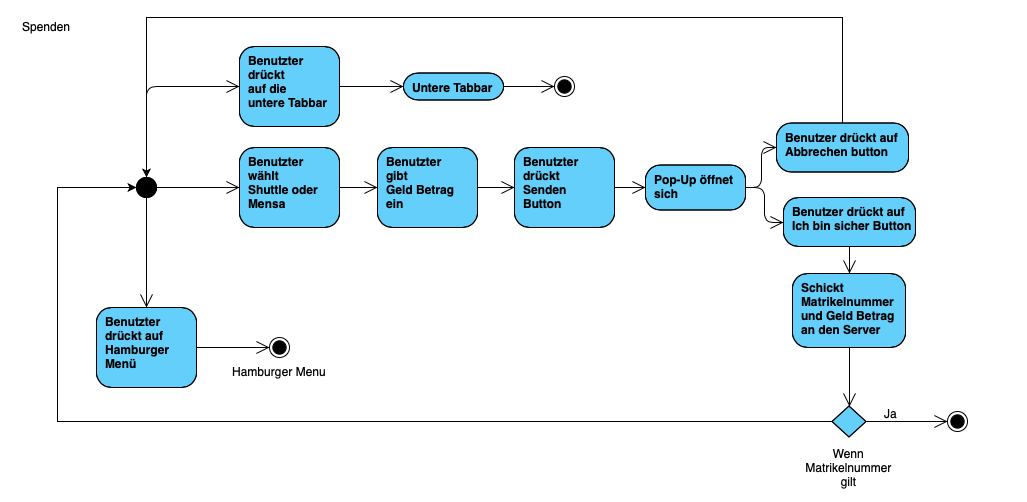
*Abbildung 8.6 IOS Empehlen Aktivitätsdiagramm*

****

*Abbildung 43.*

*Abbildung 44.*

*Abbildung 8.7 IOS Feedback Aktivitätsdiagramm*

*****Abbildung 8.8 IOS Einstellungen Aktivitätsdiagramm*****

*Abbildung 45.*

*Abbildung 44.*

*Abbildung 46.*

*Abbildung 45.*

*Abbildung 8.9 IOS Spenden Aktivitätsdiagramm*

## Funktionale und nichtfunktionale Anforderungen

**Funktionale Anforderungen**

* System muss unabhängig vom Automaten eine Zahlungsmethode anbieten.
* System soll auf einem iPhone oder iPad verschiedene Funktionen bieten. Diese Funktionen sind:
  + Das Profil zeigen,
    - ID, Vor- und Nachname, Konto
  + Anderen Studenten Free Meal spenden
  + Andere Studenten als Hilfsbedürftige empfehlen
  + Geldüberweisung zu anderen Studenten
  + Feedback zum Essen und Shuttle geben
  + Geldübertragung von einem Studenten zu einem anderen Studenten
  + Passwort ändern und Passwort wiederherstellen

**Nicht Funktionale Anforderungen**

1. Funktionale Angemessenheit

Es werden nur benötigte Funktionen bereitgestellt.

- Redundanz wird durch Kommunikation auf Minimum reduziert.

2. Performance/Effizienz

Benutzer soll nicht lange auf eine Antwort warten.

- Soll nicht länger als 2 Sekunden warten.

3. Kompatibilität

Jedes Modul muss miteinander funktionieren.

- Es werden entsprechend der mit anderen Arbeitsgruppen vereinbarten Architektur vorbereitet und jedes Modul muss im Rahmen diesem Entwurf umgesetzt werden.

4.Benutzbarkeit

Alle Benutzer müssen einfach und gleich verstehen, was sie im App machen können.

- Attraktivität wird durch eine Umfrage gesichert werden.

- Bedienbarkeit wird durch Dialoge vor jeder wichtigen Entscheidung umgesetzt.

- Erkennbarkeit wird dadurch gesichert werden, dass der Entwurf Eindeutig und gewohnt ist.

5. Zuverlässigkeit

Im Fehlersituationen muss das App sich selbst diese Fehler erkennen und lösen.

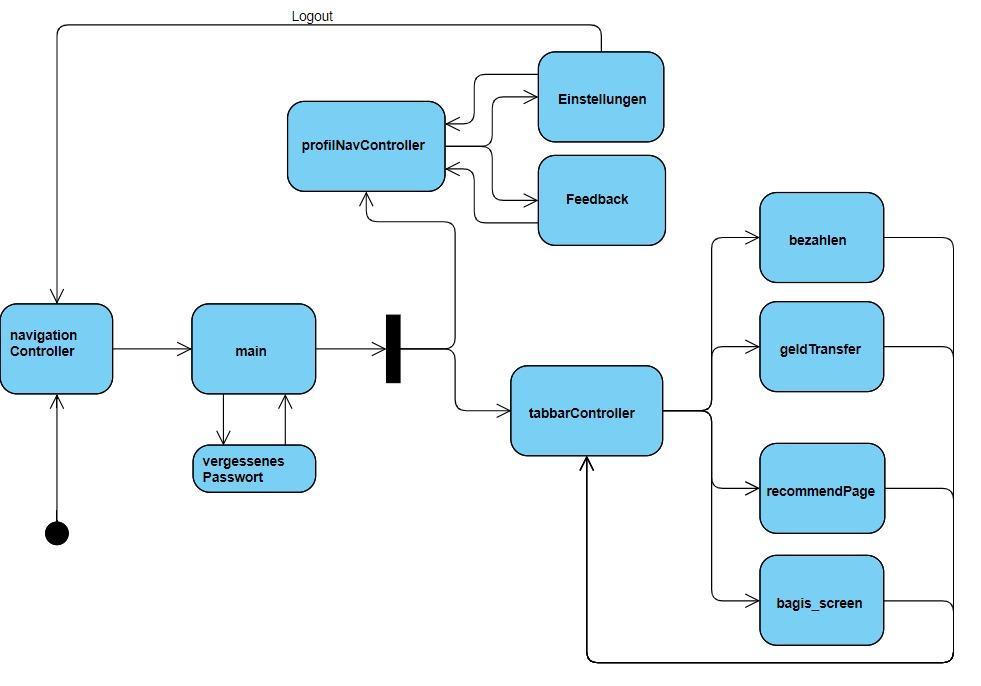
- Softwarezuverlässigkeit: Muss der Kode für alle mögliche Situationen bereit sein. Falsche Eingaben sollten die Software nicht abstürzen können.

6. Wartbarkeit

Dieses App muss in der Zukunft leicht und schnell veränderbar und verbesserbar sein.

- Die Teile des Apps werden modular programmiert, damit wenn es in einem Teil ein Problem entsteht, nur dieser Teil getroffen wird.

## Systemarchitektur

****

*Abbildung 46.*

*Abbildung 47.*

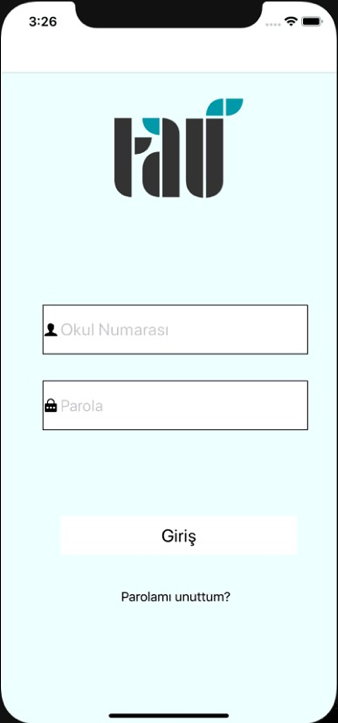
*Abbildung 44.*

*Abbildung 37.*

*Abbildung 37.*

## Design-Layouts

**Login Seite**



*Abbildung 47.*

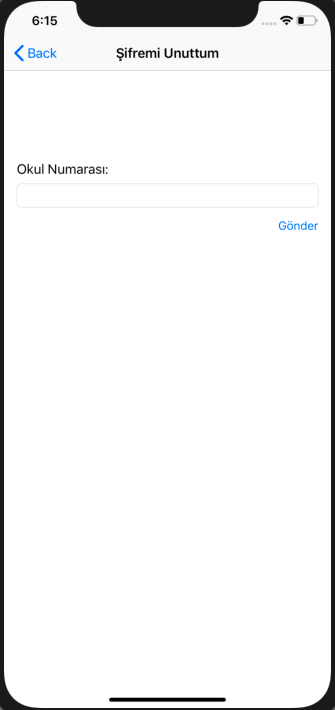
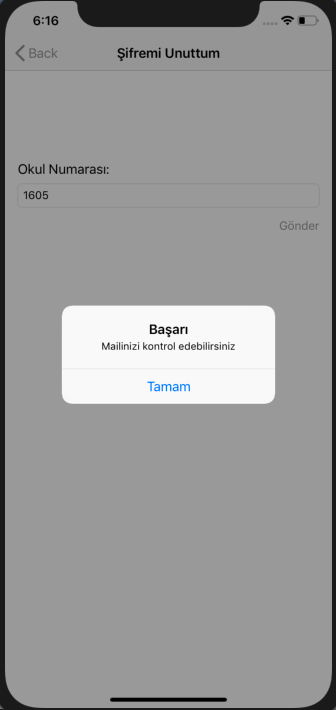
*Abbildung 48.*

*Abbildung 45.*

*Abbildung 38.*

*Abbildung 38.*

**Passwort Vergessen**

**** ****

*Abbildung 49.*

*Abbildung 48.*

*Abbildung 50.*

*Abbildung 49.*

*Abbildung 47.*

*Abbildung 46.*

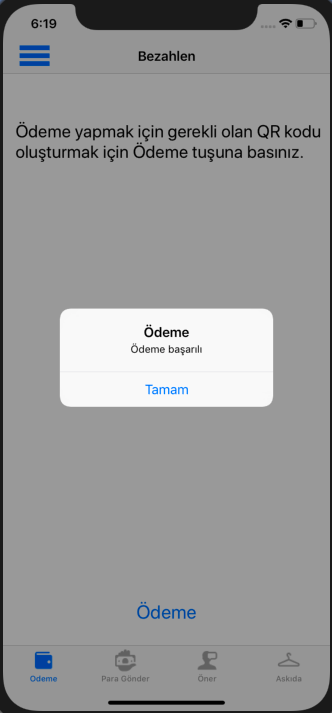
*Abbildung 40.*

*Abbildung 39.*

*Abbildung 40.*

*Abbildung 39.*

**Bezahlen**

**** **** 

*Abbildung 52.*

*Abbildung 51.*

*Abbildung 50.*

*Abbildung 53.*

*Abbildung 52.*

*Abbildung 51.*

*Abbildung 50.*

*Abbildung 49.*

*Abbildung 48.*

*Abbildung 43.*

*Abbildung 42.*

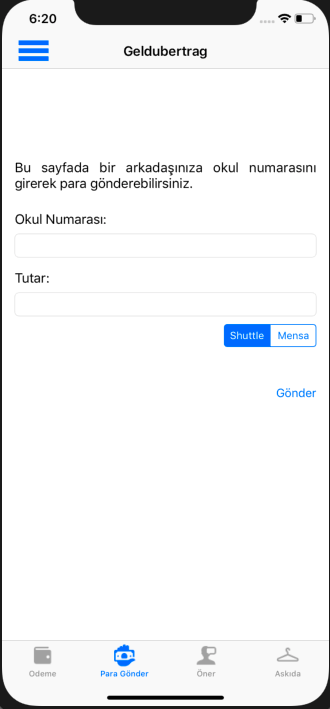
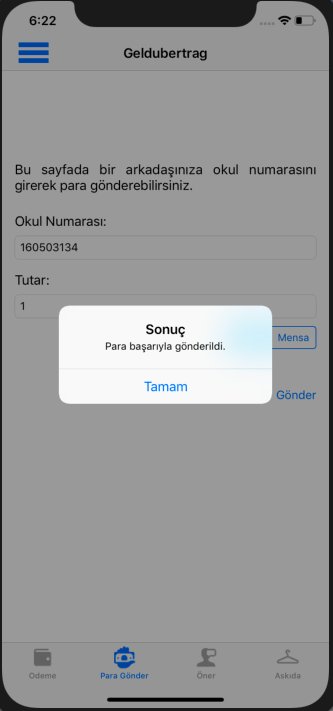
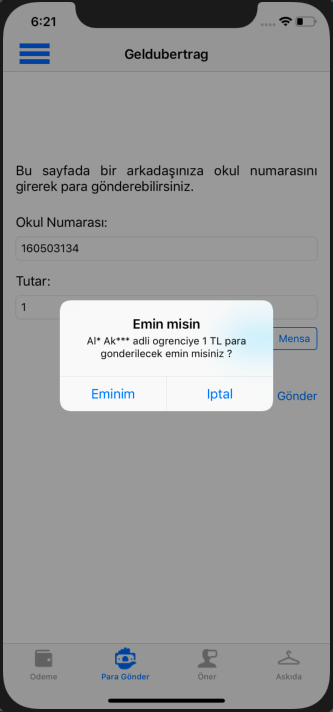
*Abbildung 41.*

*Abbildung 43.*

*Abbildung 42.*

*Abbildung 41.*

**Geldübertragung**

**** ****

*Abbildung 55.*

*Abbildung 54.*

*Abbildung 53.*

*Abbildung 56.*

*Abbildung 55.*

*Abbildung 54.*

*Abbildung 53.*

*Abbildung 52.*

*Abbildung 51.*

*Abbildung 46.*

*Abbildung 45.*

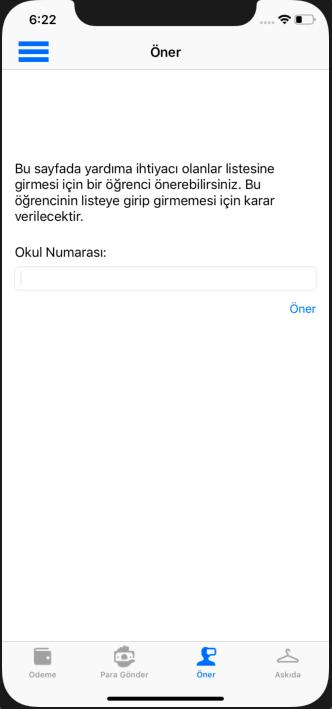
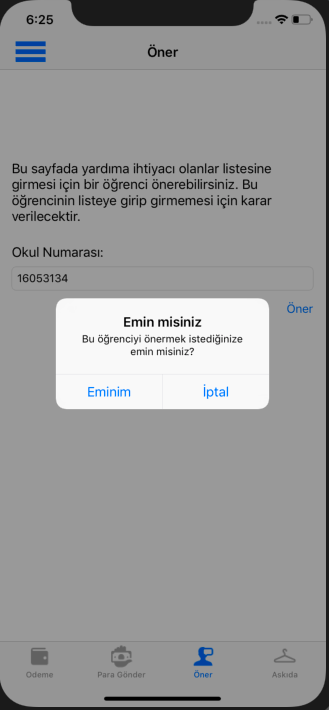
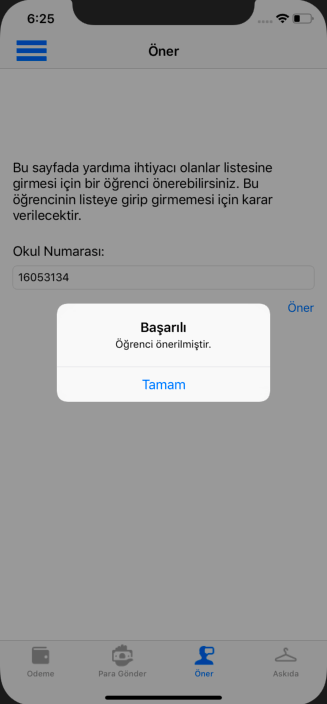
*Abbildung 44.*

*Abbildung 46.*

*Abbildung 45.*

*Abbildung 44.*

**Empfehlen**

**** **** 

*Abbildung 58.*

*Abbildung 57.*

*Abbildung 56.*

*Abbildung 59.*

*Abbildung 58.*

*Abbildung 57.*

*Abbildung 56.*

*Abbildung 55.*

*Abbildung 54.*

*Abbildung 49.*

*Abbildung 48.*

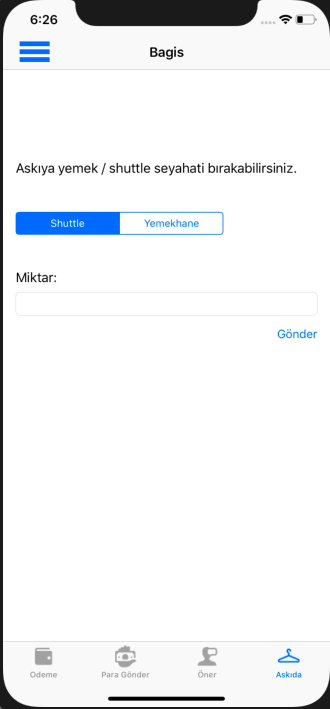
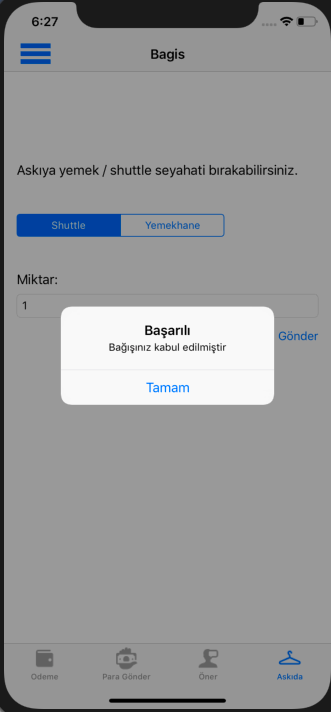
*Abbildung 47.*

*Abbildung 49.*

*Abbildung 48.*

*Abbildung 47.*

**Spenden**

**** ****

*Abbildung 60.*

*Abbildung 59.*

*Abbildung 61.*

*Abbildung 60.*

*Abbildung 58.*

*Abbildung 57.*

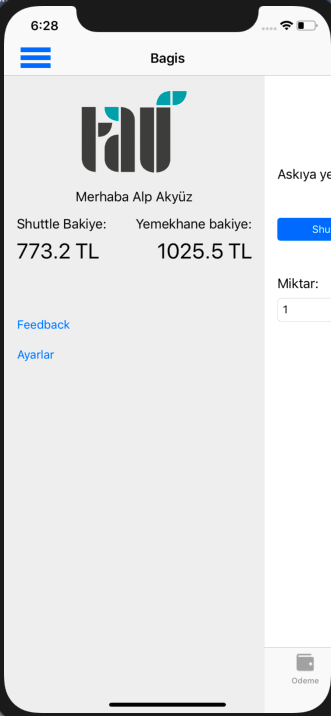
*Abbildung 51.*

*Abbildung 50.*

*Abbildung 51.*

*Abbildung 50.*

**Profil Seite**

****

*Abbildung 61.*

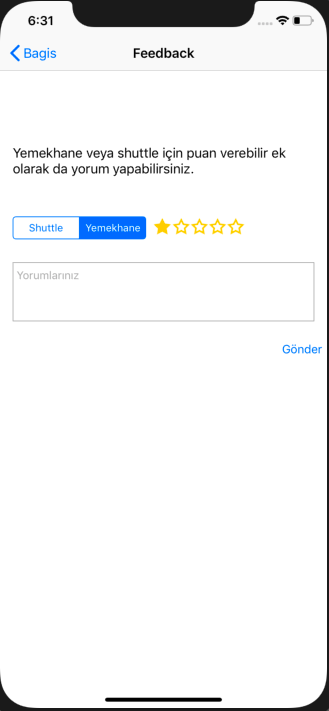
*Abbildung 62.*

*Abbildung 59.*

*Abbildung 52.*

*Abbildung 52.*

**Feedback**

**** ****

*Abbildung 63.*

*Abbildung 62.*

*Abbildung 64.*

*Abbildung 63.*

*Abbildung 61.*

*Abbildung 60.*

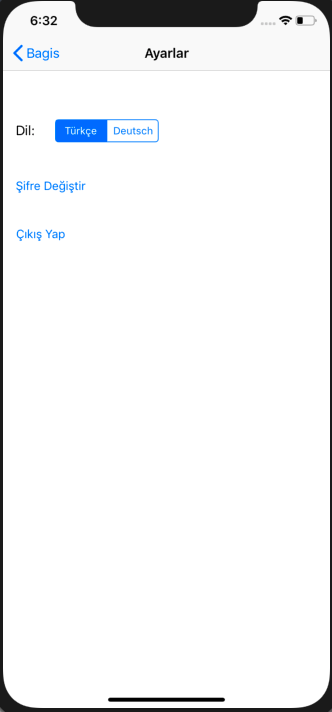
*Abbildung 54.*

*Abbildung 53.*

*Abbildung 54.*

*Abbildung 53.*

**Einstellungen**

****

*Abbildung 64.*

*Abbildung 65.*

*Abbildung 62.*

*Abbildung 55.*

*Abbildung 55.*

## Evaluation

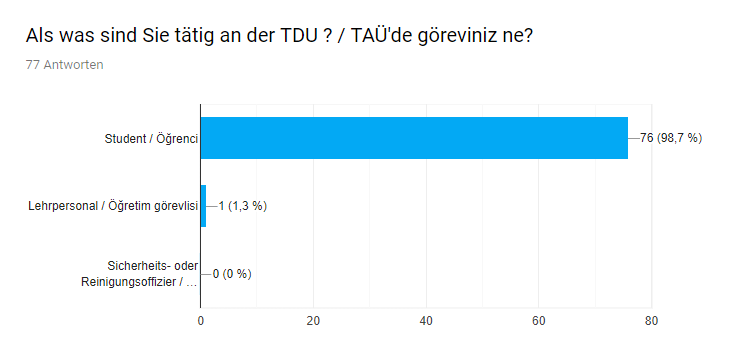
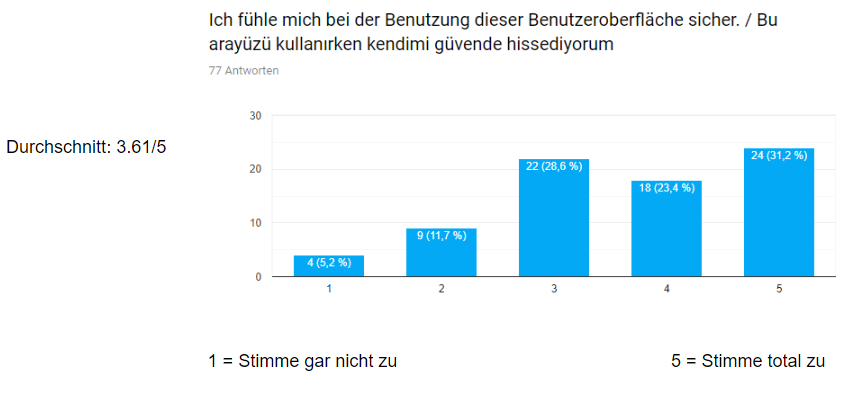
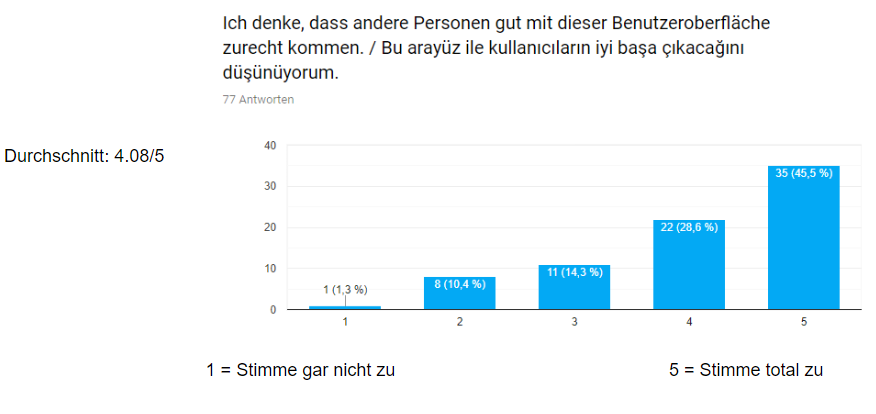
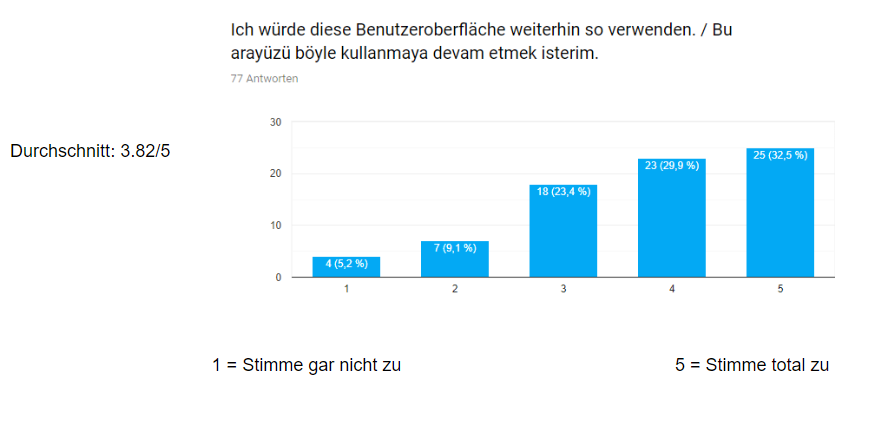
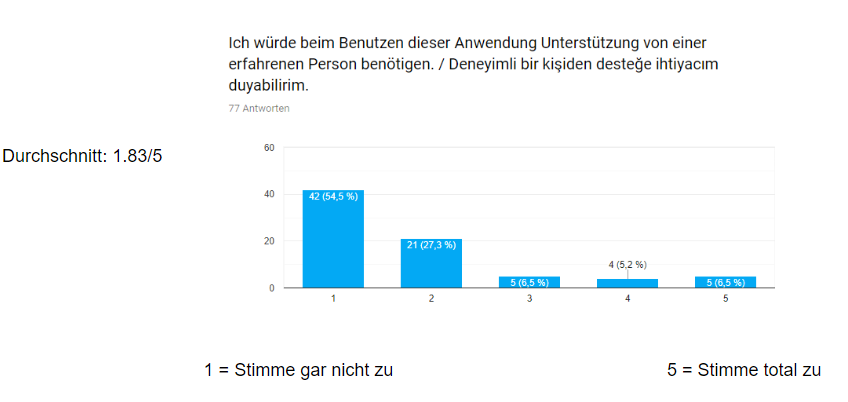
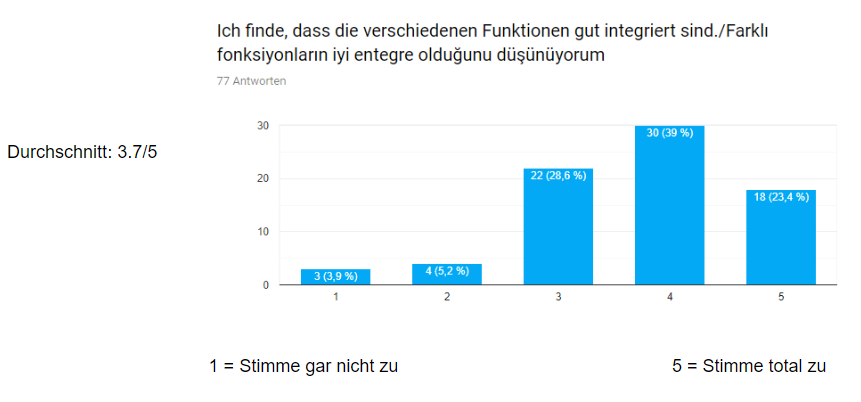
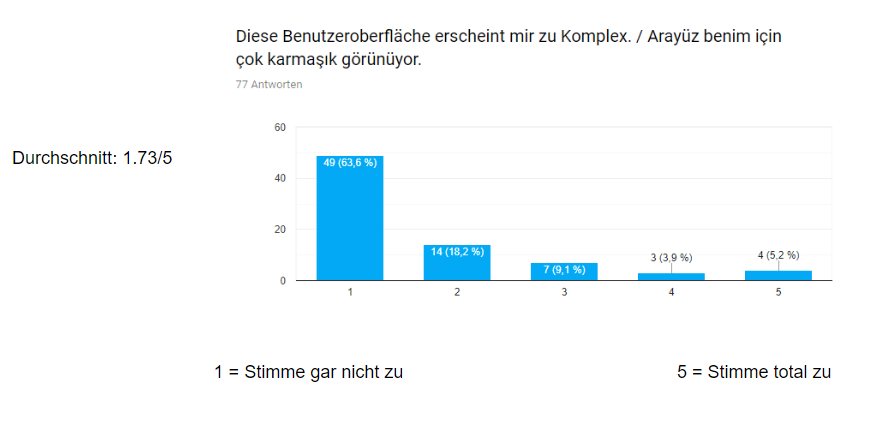
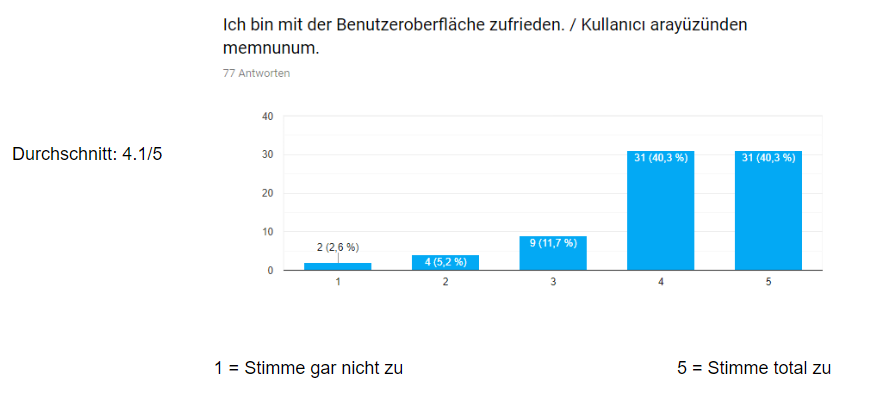
In diesem Teil wurde weitere Information über Versuche, Ergebnisse und Interpretation der Ergebnisse dieser Versuche gegeben.

## Versuchsaufbau und Versuchsdurchführung

Zum Testen der Anwendung wurde die SUS-Fragebogen-Methode verwendet. Insgesamt wurden 7 Fragen an eine Gruppe von 73 Personen aus der TDU (meistens Studenten) gestellt, die sich hauptsächlich aus Kundenzufriedenheit, Systemkomplexität, Funktionsvielfalt, Benutzerfreundlichkeit und Sicherheitsgefühl zusammensetzten. Diese Fragen wurden von den Teilnehmern mit 1 (Stimme gar nicht zu ) bis 5 (Stimme total zu) bewertet und wurden die Durchschnittswerte als Ergebnis genommen. Diese Frage waren;

* Ich bin mit der Benutzeroberfläche zufrieden. (Kundenzufriedenheit)
* Diese Benutzeroberfläche erscheit mir zu Komplex. (Systemkomplexität)
* Ich würde beim Benutzen dieser Anwendung Unterstützung von einer erfahrenen Person benötigen. (Benutzerfreundlichkeit)
* Ich finde, dass die verschiedenen Funktionen gut integriert sind. (Funktionsvielhaft)
* Ich würde diese Benutzeroberfläche weiterhin so verwenden. (Kundenzufriedenheit, Benutzerfreundlichkeit)
* Ich denke dass andere Personen gut mit dieser Benutzeroberfläche zurecht kommen. (Benutzerfreundlichkeit)
* Ich fühle mich bei der Benutzung dieser Benutzeroberfläche sicher. (Sicherheitsgefühl)

## Ergebnisse



*Abbildung 72.*

*Abbildung 71.*

*Abbildung 70.*

*Abbildung 69.*

*Abbildung 68.*

*Abbildung 67.*

*Abbildung 66.*

*Abbildung 65.*

*Abbildung 73.*

*Abbildung 72.*

*Abbildung 71.*

*Abbildung 70.*

*Abbildung 69.*

*Abbildung 68.*

*Abbildung 67.*

*Abbildung 66.*

*Abbildung 70.*

*Abbildung 69.*

*Abbildung 68.*

*Abbildung 67.*

*Abbildung 66.*

*Abbildung 65.*

*Abbildung 64.*

*Abbildung 63.*

*Abbildung 63.*

*Abbildung 62.*

*Abbildung 61.*

*Abbildung 60.*

*Abbildung 59.*

*Abbildung 58.*

*Abbildung 57.*

*Abbildung 56.*

*Abbildung 63.*

*Abbildung 62.*

*Abbildung 61.*

*Abbildung 60.*

*Abbildung 59.*

*Abbildung 58.*

*Abbildung 57.*

*Abbildung 56.*

## Interpretation der Ergebnisse

Diese Ergebnisse zeigen, dass die IOS-App relativ erfolgreich ist. Man kann sagen, dass es benutzerfreundlich ist. Die Benutzer sind mit der Benutzeroberfläche zufrieden, aber Änderungen können das Design noch benutzerfreundlicher machen. Es kann durch Hinzufügen weiterer Funktionen weiterentwickelt werden, erfüllt aber definitiv die ersten Erwartungen.

## Zusammenfassung und Ausblick

Das Ziel war ein System zu entwickeln, welches das Leben auf der Universität einfacher machen sollte. Es war offensichtlich, dass ein solches System sehr nützlich sein konnte und auch realisierbar war.

Nach der Festlegung des Designs mit den anderen Gruppen haben wir mit der Implementierung angefangen. Die Umsetzung des Projekts lief nach Plan und wir konnten alle Funktionalitäten, die im Plan gestanden haben, auch ohne Probleme realisieren.

Die Resultate im Kapitel Evaluierung zeigen, dass ein solches System viele Vorteile hat und auch durch Anwender gut empfangen wird.

Natürlich hat das Projekt auch sehr viel Potential zur Erweiterung. Was eigentlich als eine Idee für die mobile Bezahlung in der Mensa angefangen hat, könnte erweitert und für viele andere Zwecke innerhalb der Universität benutzt werden, unter anderem für die Bibliothek und für die Kantine.

# Webseite

## **Methodische Ermittlung**

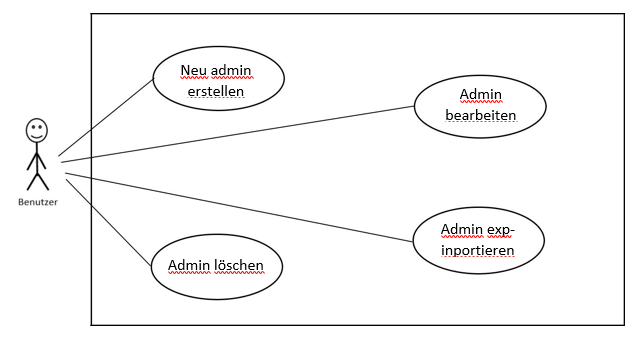
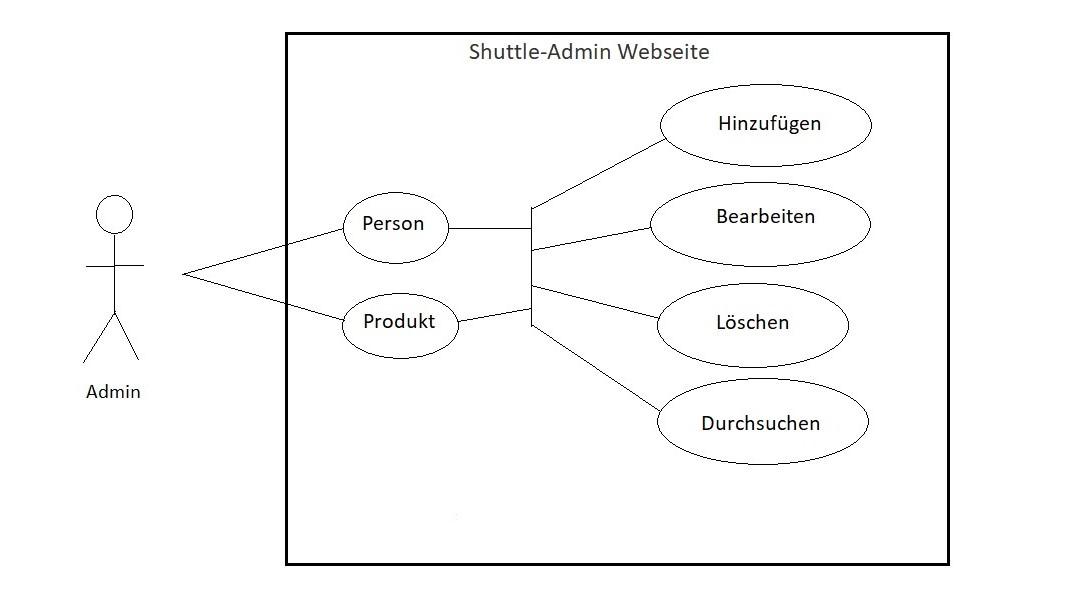
Die Erstellung einer Website erfordert den Einsatz verschiedener Technologien und Werkzeugen. Einige von ihnen bilden das Grundgerüst der Website. Einige Werkzeugen ermöglichen dem Benutzer den Zugriff auf die Site. Einige Tools verbessern die visuelle Darstellung der Website, andere beziehen sich auf die Verwendung von Datenbanken, in denen Daten gespeichert werden können.

## Anforderungsanalyse und Konzeption

## 

## Design (Entwicklung) und Implementierung

## Anwendungsfälle

**Anwendungsfalldiagramm**

*Abbildung 74.*

*Abbildung 73.*

*Abbildung 75.*

*Abbildung 74.*

*Abbildung 72.*

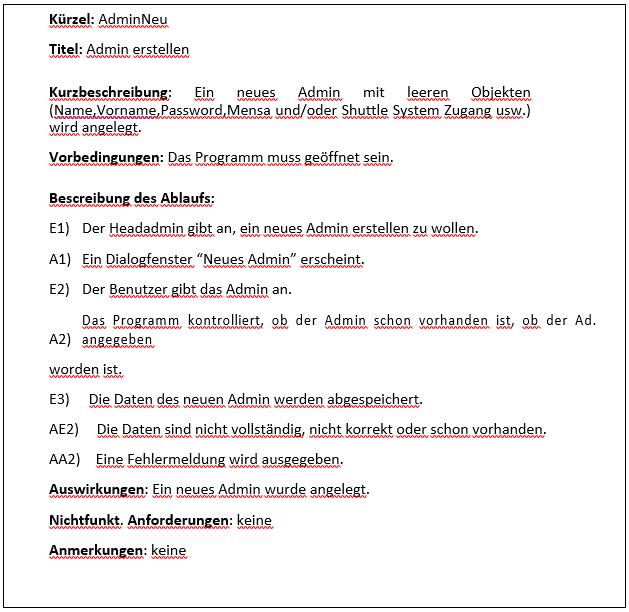
*Abbildung 71.*

*Abbildung 65.*

*Abbildung 64.*

*Abbildung 65.*

*Abbildung 64.*

****

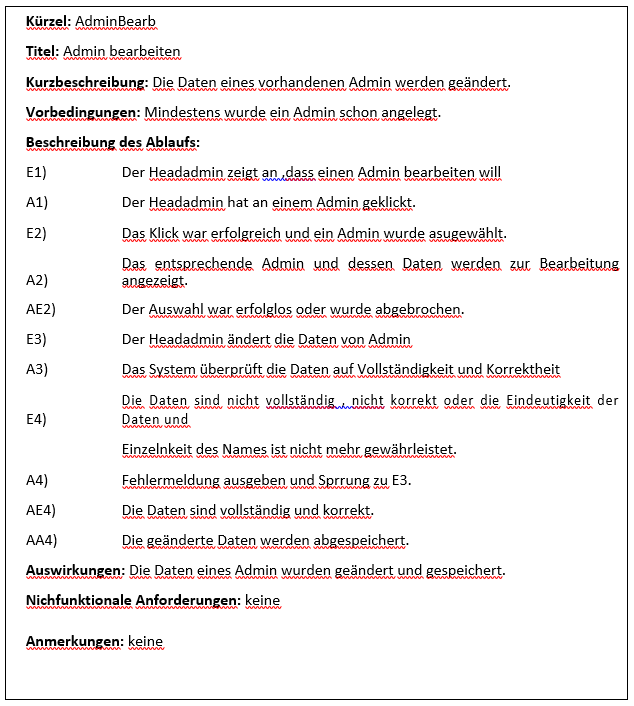
*Abbildung 75.*

*Abbildung 76.*

*Abbildung 73.*

*Abbildung 66.*

*Abbildung 66.*

****

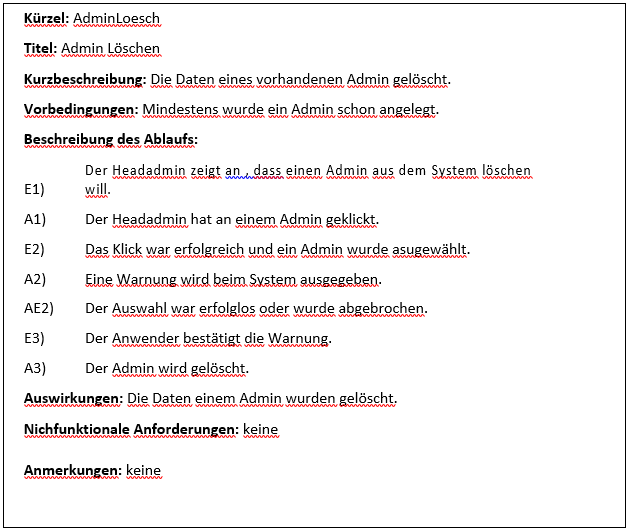
*Abbildung 76.*

*Abbildung 77.*

*Abbildung 74.*

*Abbildung 67.*

*Abbildung 67.*

****

*Abbildung 77.*

*Abbildung 78.*

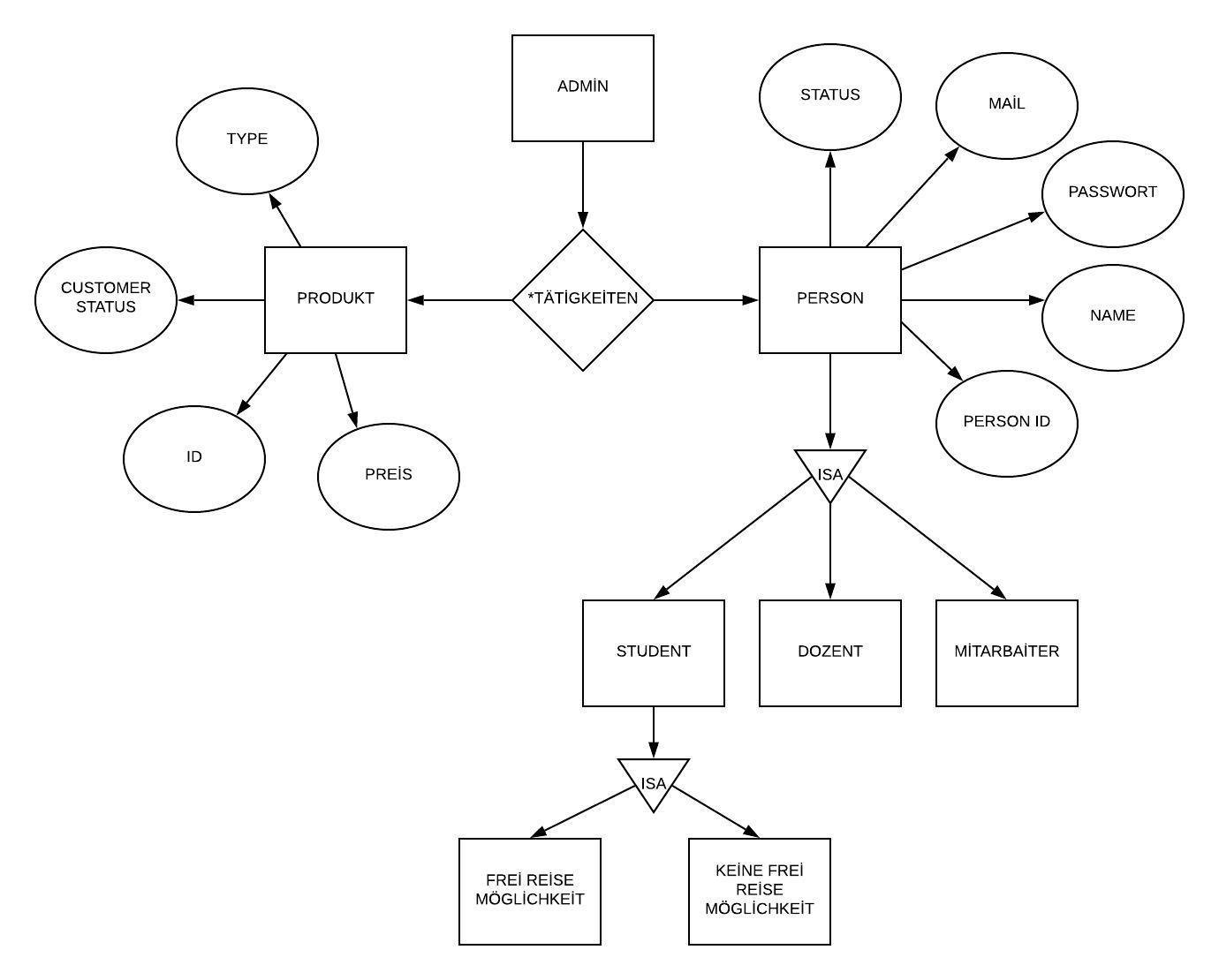
*Abbildung 75.*

*Abbildung 68.*

*Abbildung 68.*

## EER und UML Domänenmodell

(Macht OK)



*Abbildung 78.*

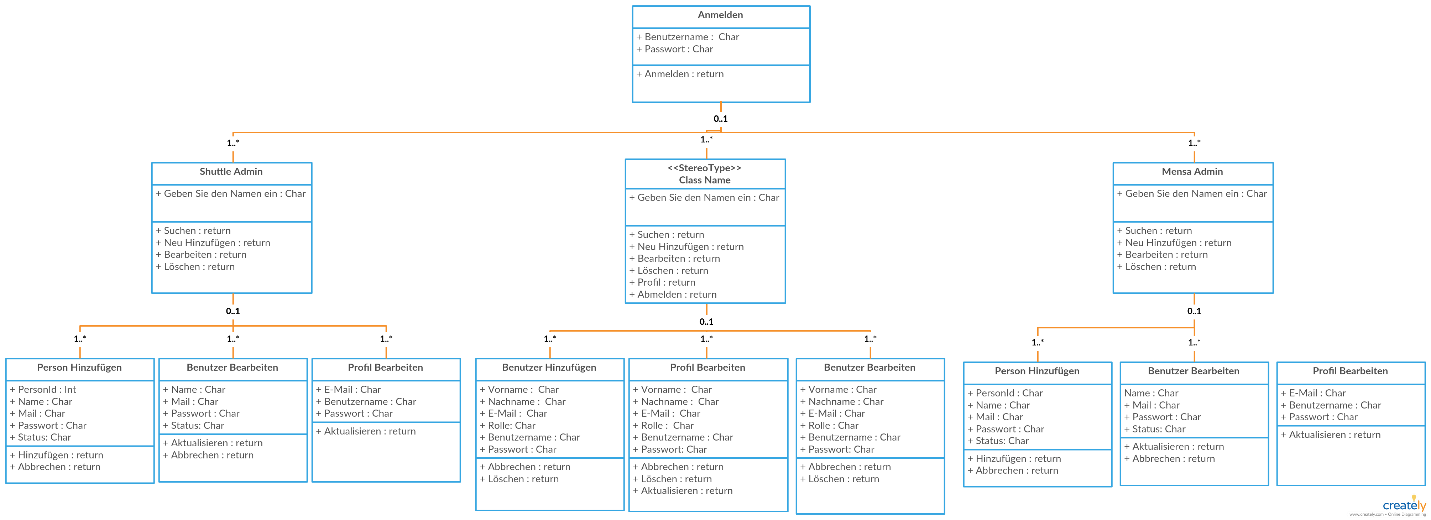
*Abbildung 79.*

*Abbildung 76.*

*Abbildung 69.*

*Abbildung 69.*

\*TÄTİGKEİTEN=Hinzufügen, Bearbeiten, Löschen Durchsuchen Prozesse

****

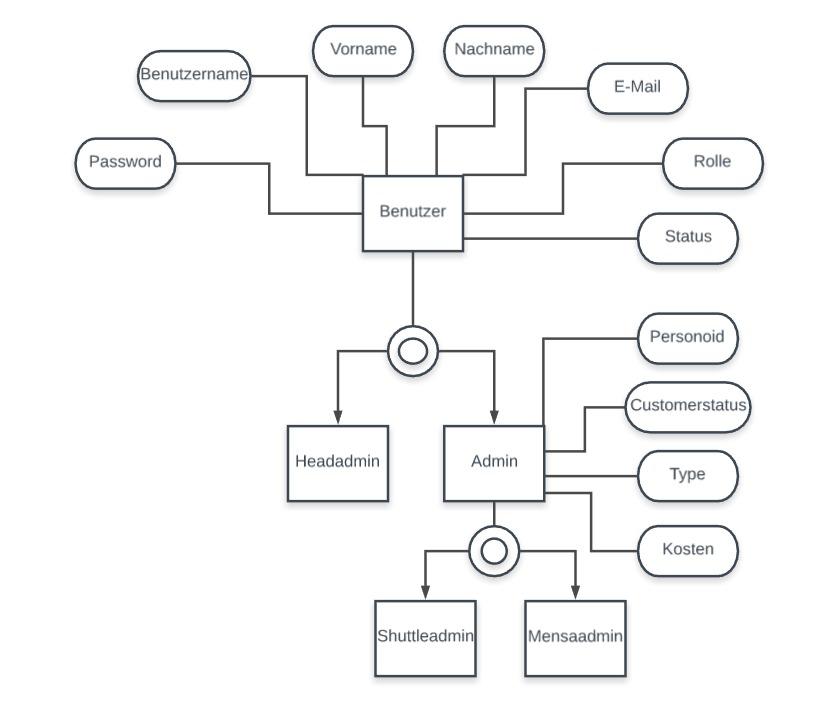
*Abbildung 79.*

*Abbildung 80.*

*Abbildung 77.*

*Abbildung 70.*

*Abbildung 70.*



*Abbildung 80.*

*Abbildung 81.*

*Abbildung 78.*

*Abbildung 71.*

*Abbildung 71.*

## Funktionale und nichtfunktionale Anforderungen

**Technische Anforderungen**

(Macht SZB)

• Allgemeine Beschreibung des Kunden

• Das System muss für einige Funktionen geeignet sein, die das Hinzufügen und Löschen der Student, die Änderung der Datei sind.

• Das System sollte aus einer Datenbank auf die zu bearbeitenden Daten der Personen zugreifen.

• Datenbank sollte einige Datei der Student, die Status, Preis, Produkt, ID und Name beinhalten.

• Benutzer sollten die Möglichkeit haben, den Studenten nach diesen Attributen durchzusuchen.

• Rahmenbedingung

• Das System soll auf eine WEB-Server funktionieren.

• Servercomputer sollte höheres RAM haben und schnell und ständig mit dem Internet verbunden werden.

• Die Webseite wird in PHP, HTML und CSS geschrieben. SQL Abfragen werden durch PHP kontrolliert.

**Systembeschreibung**

• Die Registrierung des Domainname sollte nach einem Jahr erneuert werden. Wenn der Domainname nicht erneuert wird, können andere Benutzer diese Domainname kaufen.

• Servercomputer muss immer mit Internet verbunden sein.

• Die Benutzer müssen eine Person ID und ein Passwort haben.

• Im System wird die Datei (ID, Name, Passwort, E-Mail, Balance) von den Personen gespeichert.

• Es gibt 3 verschiedene Status (Student, Dozent, Mitarbeiter) für die Personen.

• Die Benutzer können die Personen nach beliebigen Informationen durchsuchen.

****

*Abbildung 81.*

*Abbildung 82.*

*Abbildung 79.*

*Abbildung 72.*

*Abbildung 72.*

**Funktionale Anforderungen**

**Hardware:**

Bei der Programmierung des Systems wurden verschiedene Softwaresprachen verwendet. Grundsätzlich ist PHP die am weitesten verbreitete Software-Sprache. Einige der Codierungen verwenden JavaScript, eine fortgeschrittener Software-Sprache.

Der aktuelle Status des Systems ist offline, wurde jedoch von Designern mit Wamp-Server getestet und getestet.

**Software:**

Der Inhalt der Website besteht aus 6 verschiedenen Schnittstellen.

**1. Anmelden:**

· Zwei Textfelder: obligatorische Informationen

· Zeichenbox: So zeigen Sie das Passwort an

· Anmelden Button: Um sich am System anzumelden

**2. Start-Seite:**

· Profilname: Zeigt Head-Admin-Vorgänge an.

· Suchleiste und Schaltfläche “Suchen“: Suchen Sie nach dem Benutzernamen nach Admin.

· Liste: Alle in der Datenbank registrierten Admins werden angezeigt.

· Schaltfläche "Neu hinzufügen": leitet auf die Hinzufügen Seite weiter, um einen neuen Administrator hinzuzufügen.

· Schaltfläche Bearbeiten: Navigiert zu der Seite, auf der die Admininformationen bearbeitet werden.

· Schaltfläche "Löschen": Löscht den ausgewählten Namen aus der Datenbank.

**3. Profile Bearbeiten:**

· Drei Textfelder: obligatorische Informationen

· Eine Combobox: Hier können Sie die Qualifikation einer Person auswählen.

· Abbrechen: Leitet Sie an der Startseite die Änderungen, ohne zu speichern.

· Löschen: Löscht den angezeigten Head-Admin aus dem System.

· Aktualisieren: Aktualisieren Sie die Head-Admin-Informationen.

**4. Benutzer Hinzufügen:**

· Fünf Textfelder: obligatorische Informationen

· Eine Combobox: Hier können Sie die Qualifikation einer Person auswählen.

· Abbrechen: Leitet Sie an der Startseite die Änderungen, ohne zu speichern.

· Schaltfläche “Hinzufügen": Leitet auf die Hinzufügen Seite weiter, um einen neuen Administrator hinzuzufügen.

**5. Benutzer Bearbeiten:**

· Fünf Textfelder: obligatorische Informationen

· Eine Combobox: Hier können Sie die Qualifikation einer Person auswählen.

· Abbrechen: Leitet Sie an der Startseite die Änderungen, ohne zu speichern.

· Aktualisieren: Aktualisieren Sie die Admin-Informationen.

**Orgware:**

Das Hinzufügen eines neuen Administrators, das Löschen vorhandener, das Ändern vorhandener Informationen, das Erstellen einer neuen Administratoren-Gruppe und das Bearbeiten / Löschen der Head-Admin-Informationen sind die Funktionen, die das System bieten sollte. Gleichzeitig variieren die Berechtigungen von Personen auf dieser Webseite in Abhängigkeit von den Stellenbeschreibungen der Personen.

## Systemarchitektur



*Abbildung 82.*

*Abbildung 83.*

*Abbildung 80.*

*Abbildung 73.*

*Abbildung 73.*

## Design-Layouts

**Anmeldenseite**

****

*Abbildung 83.*

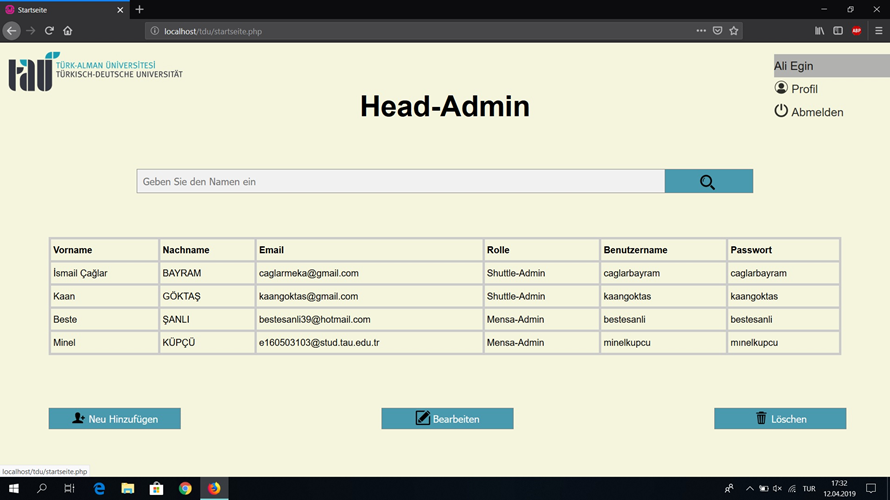
*Abbildung 84.*

*Abbildung 81.*

*Abbildung 74.*

*Abbildung 74.*

**Head Admin Eintrittsseite**

****

*Abbildung 84.*

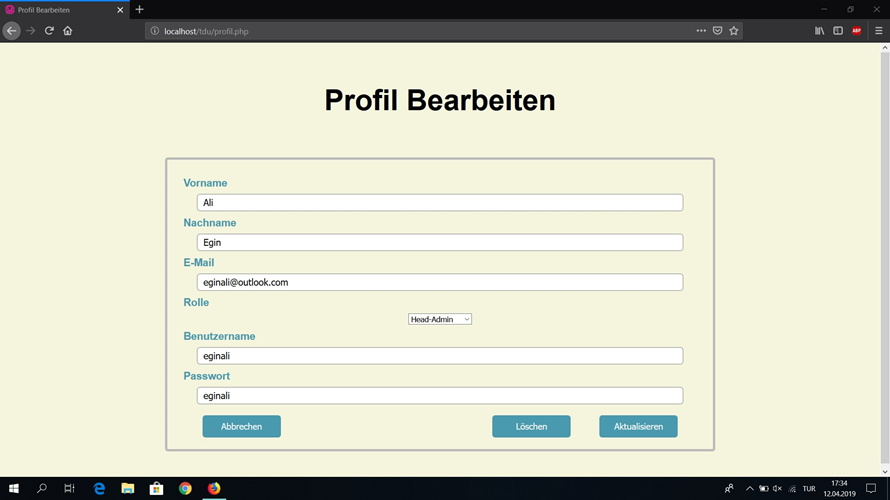
*Abbildung 85.*

*Abbildung 82.*

*Abbildung 75.*

*Abbildung 75.*

**Head Admin Profil Bearbeiten**

****

*Abbildung 85.*

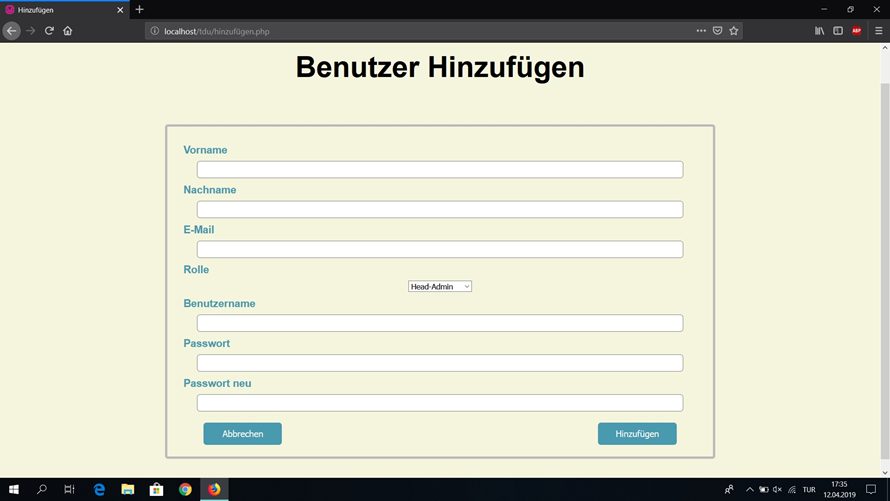
*Abbildung 86.*

*Abbildung 83.*

*Abbildung 76.*

*Abbildung 76.*

**Benutzer Hinzufügen**

****

*Abbildung 86.*

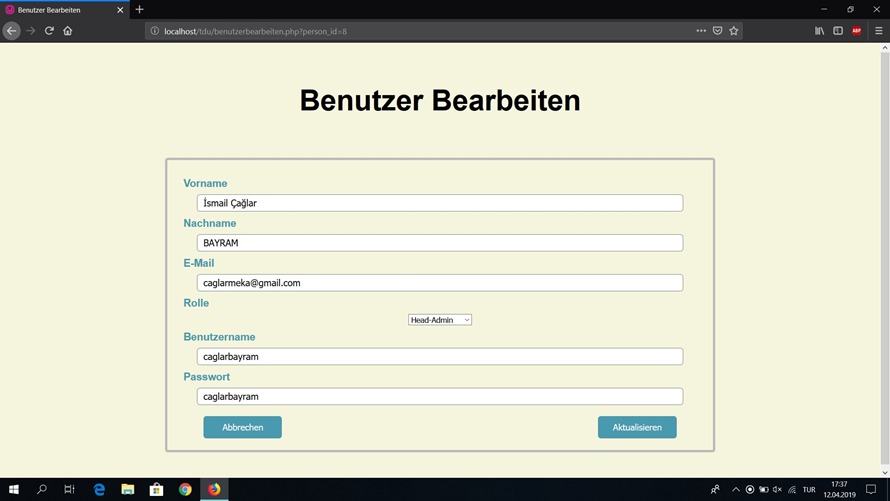
*Abbildung 87.*

*Abbildung 84.*

*Abbildung 77.*

*Abbildung 77.*

**Benutzer Bearbeitenseite von Head Admins**

****

*Abbildung 87.*

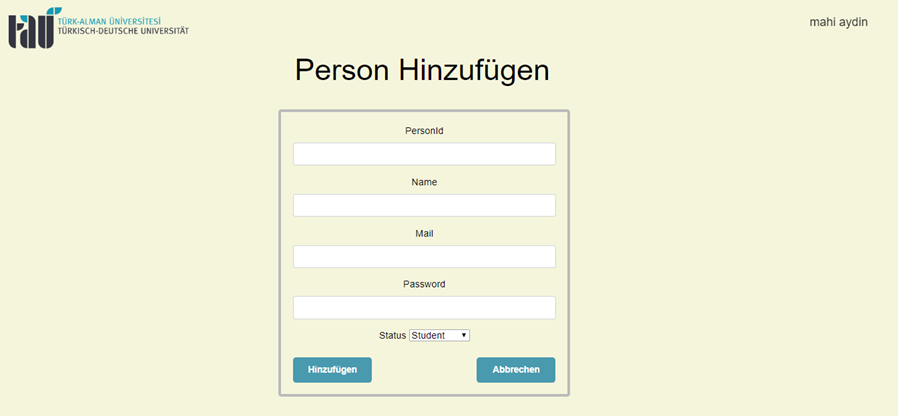
*Abbildung 88.*

*Abbildung 85.*

*Abbildung 78.*

*Abbildung 78.*

**Person Hinzufügen**

****

*Abbildung 88.*

*Abbildung 89.*

*Abbildung 86.*

*Abbildung 79.*

*Abbildung 79.*

**Benutzer Bearbeitenseite von Shuttle oder Mensa Admins**

****

*Abbildung 89.*

*Abbildung 90.*

*Abbildung 87.*

*Abbildung 80.*

*Abbildung 80.*

**Profil Bearbeitenseite von Shuttle oder Mensa Admins**

****

*Abbildung 90.*

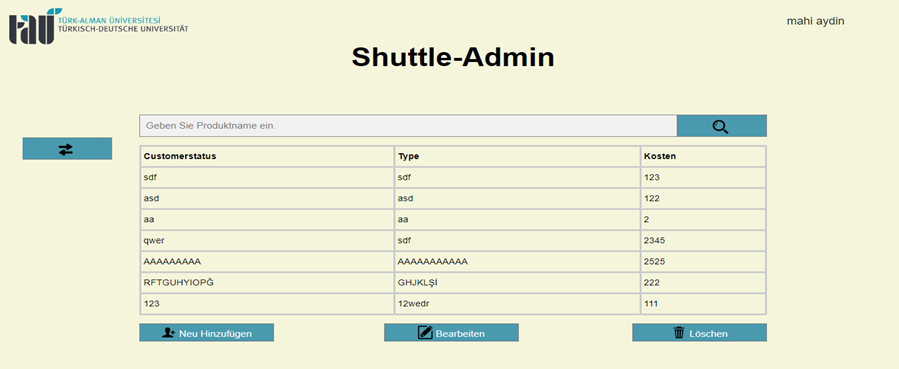
*Abbildung 91.*

*Abbildung 88.*

*Abbildung 81.*

*Abbildung 81.*

**Shuttle Admin Eintrittsseite**

****

*Abbildung 91.*

*Abbildung 92.*

*Abbildung 89.*

*Abbildung 82.*

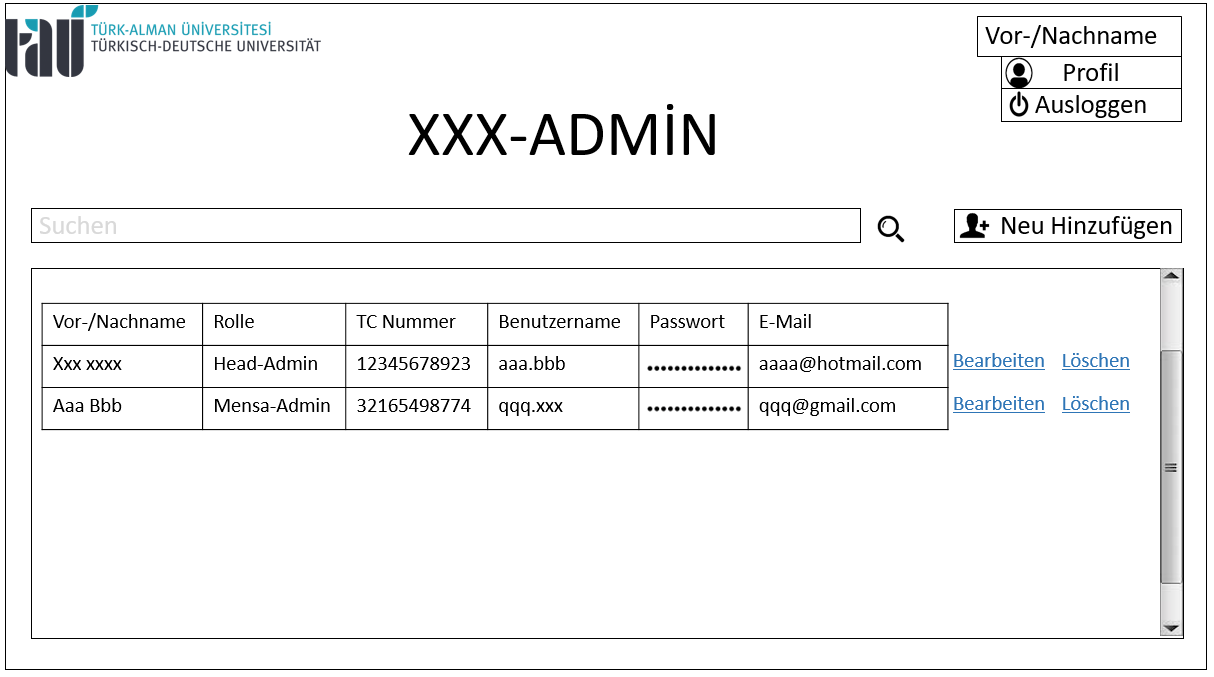
*Abbildung 82.*

## Evaluation

In diesem Projekt, das den Studenten viele Annehmlichkeiten bieten soll, soll eine Webseite entworfen werden, die den Studenten gefällt und von ihnen genutzt werden kann, indem ihre Wünsche und Vorschläge berücksichtigt werden. In dieser Richtung wurde eine Umfrage erstellt, um systematisch zu recherchieren und die Wünsche und Anregungen der Studierenden zu analysieren. In diesem Abschnitt werden auch die Struktur und die Ergebnisse dieser Umfrage erläutert.

## Versuchsaufbau und Versuchsdurchführung

Der Zweck dieser Umfrage ist es, ein Design auf der Webseite der Infrastruktur zu bestimmen. Die Zielgruppen wurden vor Durchführung dieser Umfrage identifiziert. Diese Zielgruppen sind insbesondere die Studierenden und die Personen, die den Service- und Speisesaal nutzen. Dann wurden drei verschiedene Seitendesigns erstellt, und dem Benutzer wurde die Frage gestellt, welches der folgenden Website-Designs Ihnen gefallen hat.

Die "andere" Option wurde zu den Antworten hinzugefügt, und es sollte herausgefunden werden, warum sie es nicht mochten, wenn die Designs nicht gemocht wurden. Diese Designs sind wie unten gezeigt. ****

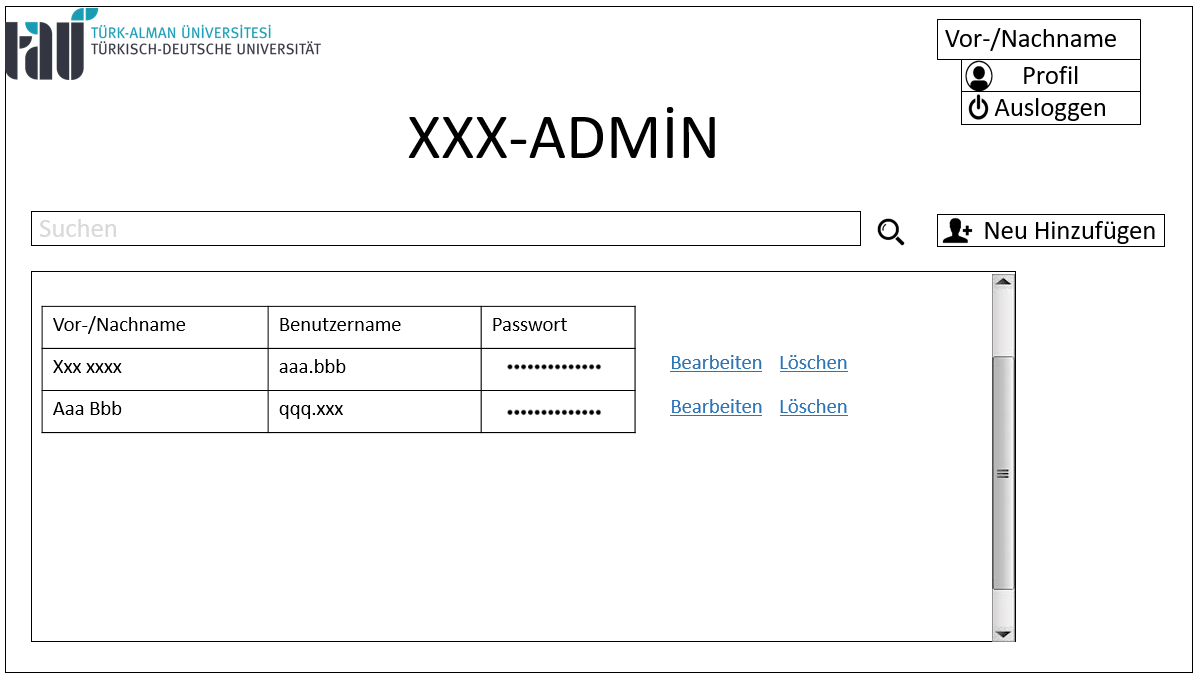
*Abbildung 92.*

*Abbildung 93.*

*Abbildung 90.*

*Abbildung 83.*

*Abbildung 83.*

****

*Abbildung 93.*

*Abbildung 94.*

*Abbildung 91.*

*Abbildung 84.*

*Abbildung 84.*

****

*Abbildung 94.*

*Abbildung 95.*

*Abbildung 92.*

*Abbildung 85.*

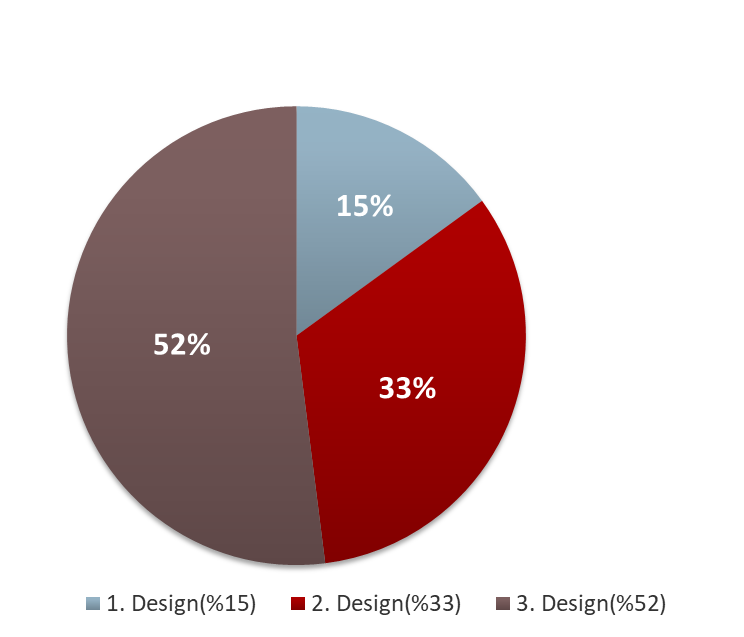
*Abbildung 85.*

Um die Stärken der Webseite zu enthüllen und diese Funktion dem zu wählenden Webseiten-Design hinzuzufügen, wurde dem Benutzer die Frage „Was hat Ihnen an diesen Designs am besten gefallen?“ Gestellt. Ebenso der Benutzer die Eigenschaften zu bestimmen, die Benutzer sicher auf der Webseite verwendet werden, nicht mögen „Was die Dinge sind Sie dachten, Sie haben nicht wie das Design oder Design zu ändern?“ Die Frage wurde gestellt. Die vorbereitete Umfrage wurde mit den Studentengruppen geteilt.

## Ergebnisse

Die Umfrage wurde von 32 Benutzern beantwortet und alle Befragten waren Studenten. Das beliebteste Webdesign war das dritte Webdesign. Dann kam das zweite Design, das am wenigsten gefeierte Design war das erste Webdesign. Das Ergebnisdiagramm der Umfrage sieht wie folgt aus.

Im Allgemeinen sind die Designs einfach und praktisch. Es scheinen zu viele Funktionen geschätzt zu werden. Einfach und elegant. Auf der anderen Seite wollten die Nutzer nicht farblos sein, sondern Farben, die es sein wollten. Einige der Befragten empfanden die Anzahl der Optionen als ausreichend, andere als unzureichend.



*Abbildung 95.*

*Abbildung 96.*

*Abbildung 93.*

*Abbildung 86.*

*Abbildung 86.*

## Interpretation der Ergebnisse

Als Ergebnis der Umfrage wurde entschieden, dass das Design der Webseite das dritte Design ist und dass dieses Design gemäß den Wünschen der Benutzer entwickelt wird. Die positiven Aspekte des gewählten Designs sind, dass dieses Design eine benutzerfreundliche Oberfläche hat und klar und verständlich ist. Die negativen Aspekte dieses Designs sind, dass sie farblos und nicht detailliert sind. In Übereinstimmung mit den Wünschen der Benutzer wird die Webseite klar dargestellt und die Funktionen, die klar und unklar sind, werden klar dargestellt.

## Zusammenfassung und Ausblick

Der Zweck ist, dass mit dem neuen System die Datenbank der Mensa- und Shuttlefirmen durch das Universitätspersonal zu kontrollieren. Somit werden mögliche Probleme verhindert werden. Die Head-Admin, die von der Universität ausgewählt werden, können die Informationen einfacher verwalten. Sie sind in der Lage, dass neue Administratoren (für Mensa oder Shuttle) hinzufügen, vorhandene Administratoren aus dem System löschen oder ihre Informationen bearbeiten.

Für die Realisierung des Projekts wurde die Website in PHP und HTML codiert. Anschließend wurde das Design der Website mit Hilfe von CSS entsprechend den Umfrageergebnissen vervollständigt. Dann wurde die Webseite mit SQL verbunden. Schließlich wurde diese Arbeit erfolgreich eingerichtet.

Allgemeiner Zweck des Projekts war nur eine Karte für verschiedene Tätigkeiten benutzen. An der TDU benutzen wir jetzt 2 Karten. Diese Karten sind Shuttle-Karte und Mensa-Karte. Aber mussten wir wirklich zwei verschiedene Karten benutzen? Für diese Probleme zu löschen TDU-Studenten ein Projekt entwickelt und das Projekt wurde in Untergruppen unterteilt.

Unser Teil ist AP7 Admin-Webseite Shuttle. Zuerst bestimmten wir unsere Projektplan und Zielsetzung. Danach erstellten wir Arbeitspakete nach unserem Ziel.

Erreichte Ziele:

* Wir erstellten eine Webseite für die Admin.
* Admin kann in Server über Person-Informationen bearbeiten kann. Diese Bearbeitung ist Hinzufügen, Löschen, Bearbeiten und Durchsuchen.
* Visuelle und benutzerfreundliche Benutzeroberfläche erstellen. Für es zu schaffen haben wir eine Umfrage vorbereitet. Wir stellten 3 alternative Designs. Wir haben uns das beliebteste Design entschieden und entwickelt es nach den Kommentaren.

# Backend

## **Methodische Ermittlung**

**1.** Spring Boot

**2.** Tomcat Server

**3.**  Apache Server

**4.** JWT

**5**. Postman

**6.** Ubuntu OS

**7.** Putty

**8.** MySQL Database

**9.** JPA

**10.** FileZilla

**11.** Github

**Spring Boot**

Spring Boot ist der Kernteil von dem ganzen System, es beinhaltet ein Spring Framework, das aus Java Klassen besteht. Dieses Framework nutzt REST API, das auf HTTP Protokoll lauft und Datenaustausch ermöglicht. Mit diesen Fähigkeiten kann das System die Anforderungen (Http Requests) bekommen, dann kann die nötige Antworte zurückgeben. Hauptziel von Spring Boot ist, eine Schnittstelle zwischen Clients und Datenbank zu sein.

**Tomcat Server**

Tomcat Server ist ein Web-Server-Dienst befindet sich in der Spring Boot. Mit diesem Dienst kann das System, die Datenpakete die aus Internet kommen, nach Spring Boot weiterleiten, hiermit kann Spring Boot sein Job machen.

**Apache Server**

Apache Server ist ein Web-Server-Dienst ähnlich Tomcat Server. Der Unterschied ist, dass Tomcat läuft für Spring Boot aber Apache Server läuft für den ganze Server. In diesem Projekt benutzt man zwei Web-Servers, um die maximale Sicherheit zu erreichen. Ein SSL (Secure Socket Layer) wurde integriert, damit die Kommunikation des Servers mit dem Client verschlüsselt ist.

**JWT (JSON Web Token)**

In diesem Projekt hat jeder User sein eigenes Konto, deswegen braucht das System ein Profilverwaltung. Man braucht eine Registrierungs- und Anmeldungssystem zu entwickeln. Wenn ein Benutzer von einem Client anmeldet, muss der Server ein eindeutiges, geheimes Token schicken, das zeigt, wer eingeloggt hat. JWT ist eine gute Lösung das sehr schwer zu hacken ist.

**Postman**

Postman ist ein Werkzeug, das benutzt, um das System zu testen. Bei der Entwicklung des Backendes gibt es kein Android oder IOS Applikation aber man braucht ein Weg, um die Funktionen des System zu testen. Postman hat eine wichtige Rolle, sodass keine Konflikte bei der Integration von Clients vorkommen.

**Ubuntu OS**

Ubuntu ist ein Linux Betriebssystem, das mit UNIX läuft. Es ist eine gute, schnelle und leichte Alternativ zu Windows. In diesem Projekt ist das Betriebssystem des Servers ist Ubuntu. Es kann Java Applikationen ausführen deswegen kann Spring Boot Applikation ausgeführt werden.

**PuTTy**

PuTTy ist ein Remote Access Programm. Es wird benutzt, um das Terminal von Ubuntu fernzusteuern. Um das ganze System zu starten oder stoppen, muss man Befehle in der PuTTy ausführen. Ubuntu OS hat keine Meinung ohne PuTTy.

**MySQL Database**

In diesem System braucht man ein Database Server, das mit dem Daten schnell und effizient arbeiten. MySQL ist ein guter Kandidat, weil es sehr berühmt und kompatibel mit Hunderten von Programmiersprachen.

**JPA (Java Persistence API)**

JPA ist eine Brücke zwischen Spring Boot Applikation und MySQL Database. Sein Job ist, die Daten zwischen Spring und MySQL zu übertragen, die notwendige SQL Query Code automatisch zu erstellen und die Übertragung zu optimieren. Diese API verringert die Programmierung Zeit und erhöht die Effizienz von Entwicklung.

**FileZilla**

FileZilla ist ein FTP (File Transfer Protocol) Client. Sein Job ist nicht so viel aber sehr wichtig. Für die Übertragung von dem Code, das in der lokalen Umgebung entwickelt wurde, zu den Server wurde FileZilla benutzt.

**GitHub**

GitHub ist eine gemeinsame Plattform, der von viele Entwickler benutzt wurde. In der Gruppe-Projekte hat GitHub eine sehr wichtige Rolle. Jeder Entwickler kann sein Code hier hochladen und den Code, die anderen Entwickler geschrieben hat, unterladen. In diesem Projekt ist GitHub immer benutzt und aller Code sind als Open-Source erreichbar.

## Anforderungsanalyse und Konzeption

Ein Backend für dieses Projekt wird aus diesem Teilprojekt anfordert. Backend beschäftigt mit Hintergrund von ganzem System. Das besteht aus 3 Hauptteilen;

* Server
* Datenbank
* Integration

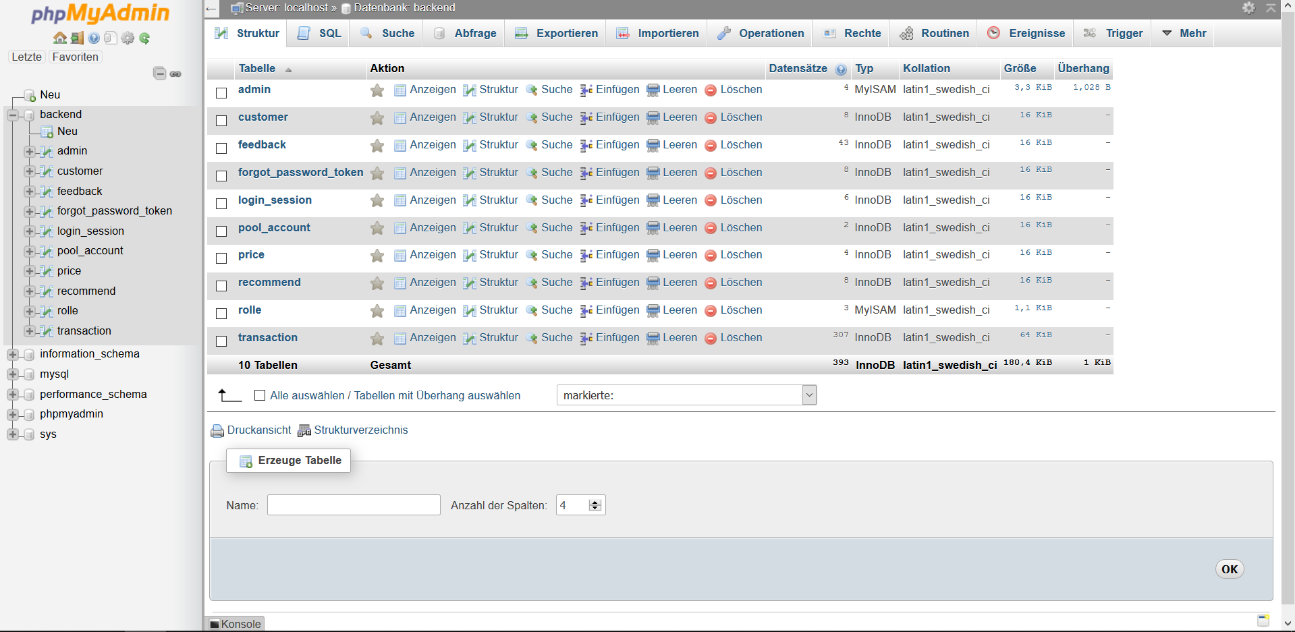
Über diese ansprechenden Hauptteile ist das Ziel ein Backend System zu erstellen. Das ganze System geht es um ein Zahlungssystem für Mensa und Shuttle herzustellen. Bezahlen kann per iOS, Android und Webseite ausgeführt werden. Dafür braucht das System ein Hauptserver und Datenbank. Server ist nötig um die alle Geräte zu dem Server verbinden zu können. Alles System führt über einen Server durch. Server funktioniert auf Linux Betriebssystem. Um die Personen addieren zu können und das Geldtransfer besorgen zu können ist noch eine Datenbank benötigt. Datenbank wurde per MySQL erstellt. In die Datenbank werden die Personen, die dieses System ausgenutzt, gespeichert. Kunden ID, Passwort, Shuttle-Balance, Mensa-Balance, Email und Status werden in die Datenbank gespeichert. Diese Tabelle ist über nur die Kunden-Infos. Reste Systeminformationen auch werden in die gleiche Datenbank gespeichert. Jeder Nutzer wird ein eigenes Passwort haben und dieses Passwort wird privat für die andere Kunden. Server wurde als schon bereitgestellter gekauft. Das private Passwort und Nutzername wurden eingestellt. Alle anderen Geräte sollen zuerst zu diesem Server verbinden. Andererseits funktioniert nicht das System. Mit privatem Passwort ist der Server auch sicher. Datenbank und Server wurden auseinander erstellt und haben miteinander keine Verbindung. Dafür wurde JPA benutzt. Nach der Verbindung wird das Backendssystem bereit. Für die Zahlung ist es noch ein QR-Code benötigt. Der Server erstellt einen QR-Code durch Java Spring für jeder Person nach Person ID. Danach scannt die Kunden den QR-Code per Smartphone ein und die Zahlung wird fertiggestellt.

## Design (Entwicklung) und Implementierung

**Datenbank**

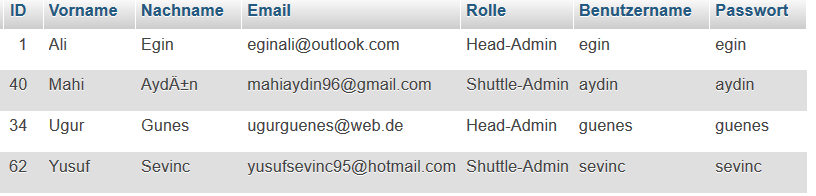
Die Datenbank ist das Herz des Systems.Jede Informationen werden dort gespeichert und wenn jemand eine Informarion fordert, die Datenbank muss jederzeit zur Verfügung stellen.

Die Datenbank bestehen aus diesen Tabellen;

* Admin
* Customer
* Feedback
* Forgot\_Password\_Token
* Login\_Session
* Pool\_Account
* Price
* Recommend
* Rolle
* Transaction

**Admin;**

Diese Tabelle steht für die Admins der Web Seite.



*Abbildung 96.*

*Abbildung 97.*

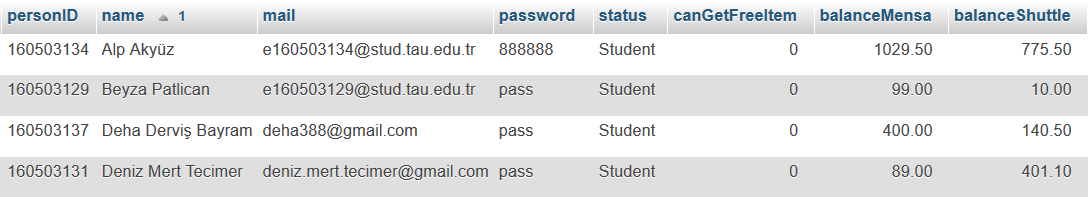
*Abbildung 94.*

*Abbildung 87.*

*Abbildung 87.*

**Customer;**

Die Tabelle Customer beinhaltet die Kundendaten mit ihrem/seinem Guthaben.



*Abbildung 97.*

*Abbildung 98.*

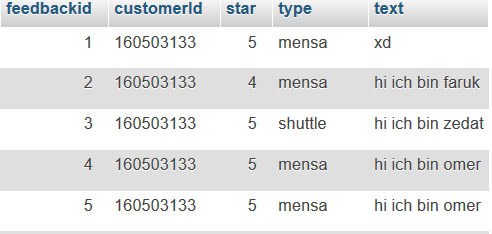
*Abbildung 95.*

*Abbildung 88.*

*Abbildung 88.*

**Feedback;**

Die Tabelle Feedback hat die Studierende, die von jemandem schon vorgeschlagen hat. Die Studierende bekommen die Sterne als Grad. Damit kann man an die Studierende Free-Meal und Free-Ride bieten.



*Abbildung 98.*

*Abbildung 99.*

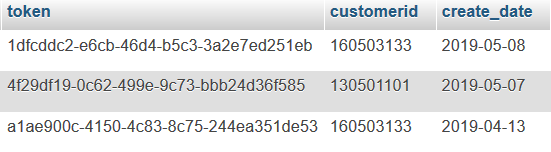
*Abbildung 96.*

*Abbildung 89.*

*Abbildung 89.*

**Forgot\_Password\_Token;**

Diese Tabelle beinhaltet die Tokens um das Konto wiederherzustellen. Die Tokens werden in dieser Tabelle speichert solange der Kunde sein Passwort geändert hat.



*Abbildung 99.*

*Abbildung 100.*

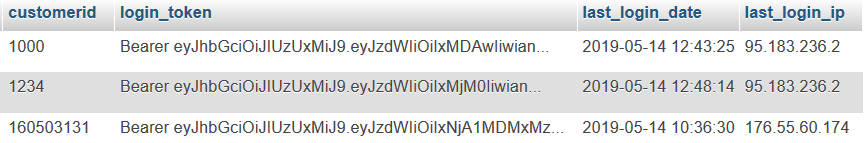
*Abbildung 97.*

*Abbildung 90.*

*Abbildung 90.*

**Login\_Session;**

Die Tabelle Login\_Session hat die Informationen die zuletzt angemeldete Gerätinfos mit deren IP’s.



*Abbildung 100.*

*Abbildung 101.*

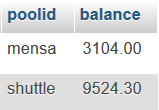
*Abbildung 98.*

*Abbildung 91.*

*Abbildung 91.*

**Pool\_Account;**

Die Tabelle Pool\_Account beinhaltet die gesamte Mensa und Shuttle Kasse.



*Abbildung 101.*

*Abbildung 102.*

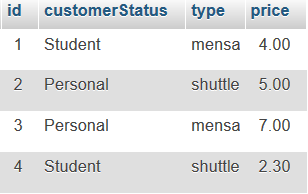
*Abbildung 99.*

*Abbildung 92.*

*Abbildung 92.*

**Price;**

Die Tabelle Price hat die Preis Informationen mit verschiedenen Preisklassen.



*Abbildung 102.*

*Abbildung 103.*

*Abbildung 100.*

*Abbildung 93.*

*Abbildung 93.*

**Recommend;**

Die Recommend Tabelle speichert die Anzahl von Vorschlägen.



*Abbildung 103.*

*Abbildung 104.*

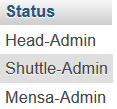
*Abbildung 101.*

*Abbildung 94.*

*Abbildung 94.*

**Rolle;**

Die Tabelle Rolle beinhaltet die Rollen, die in admin Tabelle verwendet.



*Abbildung 104.*

*Abbildung 105.*

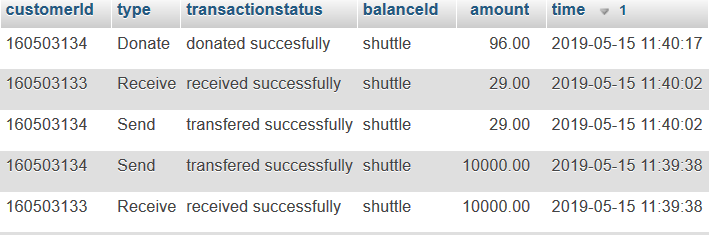
*Abbildung 102.*

*Abbildung 95.*

*Abbildung 95.*

**Transaction;**

Die Tabelle Transaction speichert die Transaktionen des Systems bis jetzt ausgeführt wurde.



*Abbildung 105.*

*Abbildung 106.*

*Abbildung 103.*

*Abbildung 96.*

*Abbildung 96.*

**Java Code**

**1) Entity**

Entity Paket beinhaltet Java Klassen für Entitäten, die im System sich befinden.

i.Customer.java: Diese Klasse enthaltet die Informationen für Kunden. Gespeicherte Informationen:

a. String personId: Matrikel Nummer

b. String name: Name von Kunde

c. String mail: Email von Kunde

d. String password: Passwort, um im System einzuloggen

e. String status: Status Info zb. Student, Personal, Dozent usw.

f. Boolean canGetFreeItem: Ein Info, die zeigt, ob diese Kunde frei Shuttle oder Essen bekommen kann.

g. Double balanceMensa: Balance für Mensa

h. Double balanceShuttle: Balance für Shuttle

ii.FeedbackInfo.java: Diese Klasse enthaltet die Informationen für die Feedbacks:

a. Int feedbackId: ID von Feedback

b. String id: ID von Customer, der diese Feedback gemacht hat.

c. Int star: Anzahl von Sterne

d. String type: Mensa/Shuttle

e. String feedbackText: Text von Feedback

iii.ForgotPasswordToken.java: Diese Klasse enthaltet die Informationen für einmal benutztes Token:

a. String token: automatische generierter UUID Code. Es ist geheim und eindeutig für jede Passwort-Reset Operation.

b. String customerId: Matrikelnummer von Kunde, wer sein Passwort neu stellen möchte.

c. Date createDate: Datum für dieses Token.

iiv.LoginSession.java: Die Klasse, um zu kontrollieren, dass jeder Kunde sich nur bei 1 Gerät anmelden kann und um die Login-Info zu protokollieren.

a. String customerId: Matrikelnummer von Kunde

b. String loginToken: Token für Anmeldung, es ist eindeutig für jede Anmeldung Operation.

c. Date lastLoginDate: Datum, um zu sehen, wann der Kunde zuletzt eingeloggt.

d. String lastLoginIp: Ip von Kunde

v. PoolAccount.java: Die Klasse für gemeinsames Konto für die Speicherung des gespendeten Geldes:

a. String poolid: Id von gemeinsames Konto

b. Double Balance: Balance von Konto

vi.Price.java: Die Klasse für Preisen für jedes Personstatus

a. Int id: Id von Preis

b. String Type: Shuttle/Mensa

c. String customerStatus: status von Kunde zB. “Student”

d. Double Price: Preis dieser Dienst

vii.Recommen.java: Die Klasse für Emfehlungen.

a. String customerId: Matrikelnummer von Kunde

b. Int recommendTimes: Anzahl davon, wie viel diese Kunde empfohlen wurde.

Transaction.java: Vergangene Logs von Transaktionen.

a. Int trasactionId: Id von Transaktion

b. String customerId: Matrikelnummer von Kunde

c. String type: Shuttle/Mensa

d. String transactionStatus: Das Ergebnis von dieser Transaktion. Zb. “paid successfully”, “insufficient balance” usw.

e. Double amount: Menge von Geld für dieser Transaktion

f. Date time: Datum von dieser Transaktion

**2) ForgotPasswordController**

ForgotPasswordController beschäftigt sich mit Passwort-Reset Operationen.

i. ForgotPasswordController

a. newPassHtml: Das ist eine Funktion, die ein HTML seite zurück gibt. Man kann sein Passwort da neu stellen.

b. confirmForgotPass: Das ist eine Funktion, die mit der HTML Seite automatisch kommuniziert und benötige Operationen macht.

c. NewPassInfo: JSON Modell für Passwort Reset

**3) Mail**

Mail Paket beschäftigt sich mit der automatisierten Sendung von der Mails.

i. MailService

a. sendMail: Eine Funktion, um Mail zu schicken. Diese Funktion braucht Parameter wie: Sender Email, Empfänger Mail, Subjekt, Inhalt von Email.

**4) QrCode**

Es ist eine automatisierte Steuerungssystem für die Bezahlung mit der Qr Code.

i. CustomerPaymentToken

a. getPaymentToken: Diese Funktion erstellt ein eindeutiges langes Token für QrCode-Bezahlung mit der UUID-Bibliotheken. Dann wird diese Token mit einer Id von Kunde verbindet. Dann speichert die Funktion diese Paar (Token, customerId) in die Tabelle und gibt sie zurück. Diese Token muss in der gegebenen Zeitraum benutzt werden, andernfalls wird diese Token gelöscht wird.

b. isPaid: Um zu kontrollieren, ob die Bezahlung mit der Token fertig gestellt ist.

c. getCustomer: Diese Methode sucht das gegebene Token und findet, wem gehört dieses Token in der Tabelle.

d. confirmPaymentToken: Diese Funktion bestätigt der Bezahlungsprozess. Dann macht das System die Kontotransaktionen automatisch und stellt der Prozess fertig.

e. deletePaymentToken: (Nicht Benutzt) Es ist eine Funktion für manuelle Löschung von Token.

**5) Repository**

Dieses Paket beschäftigt mit JPA Verbindung. Die Klassen generieren die SQL Befehle automatisch und ermöglicht die Verbindung mit dem MySQL Server.

**6) Security**

Dieses Paket ist verantwortlich von der Sicherheit des ganzen Systems. Hier wird JWT Tokens benutzt, um alle Befehle zu authentifizieren. Die Befehle, die kein JWT Token beinhalten, wird automatisch verhindert, somit kann das System von Manipulationen bei der Befehle vermeiden. Mit JWT Tokens kann System verstehen, wer eingeloggt hat.

**7) ServerInfo**

i.ServerInfo

Diese statische Klasse beinhaltet die wichtige Informationen des Servers.

a. String mainUrl: URL des Servers, wenn der Server ein unterschiedliche Domainname hat, muss man hier ändern.

**8) Startup**

Die Prozesse, die ausgeführt werden, wenn der Server neustartet.

i.ConnectionControl.java : Diese Klasse gibt ein HTML seite zurück, es hat kein funktionalität.

ii.StartMessage.java : Diese Klasse beinhaltet ein Textkunst oder ein Logo, wenn der Server neugestartet wird, kann man sehen dieses Logo.

iii.TomcatConfig.java : Es ist ein Einstellungklasse für Tomcat Web Sever. Sein Hauptziel ist die Übertragung von Http zu Https (Secure Socket Layer). Es ist wichtig für die Sicherheit.

**9) Swagger2**

Dieses Paket ein Bibliothek für Online-Dokumentation. Man kann diese Dokumentation von hier erreichen: [Https://workshop-omercem.com:8443/swagger-ui.html](https://workshop-omercem.com:8443/swagger-ui.html)

In dieser Online Dokumentation befindet sich die Informationen, wie kann man die REST API Funktionen rufen. Alle Funktionen können mit ihrer Modelle hier gefunden werden.

**10) CustomerController.java**

Diese Klasse ist einfach der Kern des Systems. Alle Clients (Android, IOS) schicken ihre Anforderungen hier und rufen die Funktionen auf. Die Funktionen dieser Klasse sind:

a.Sign-up: Es ist eine Funktion für Registrierung.

b.Get-Info: Wenn man eingelogt hat, wird diese Funktion geruft und das gibt Informationen der Benutzer zurück. (zB. Name, Mail, Konto Balance usw.)

c.Get-Name: Das Ziel dieser Funktion ist, zu finden, wem die als Parameter gegebene Matrikelnummer gehört.

d.Pay: (Nicht Benutzt) Das ist eine Funktion für Bezahlung.

e.Deposit: (Nicht Benutzt) Das ist eine Funktion für die Aufladung des Geldes.

f.Transfer: Die Funktion für Übertragung des Geldes zwischen Kunden.

g.Change-Password: Die Funktion für die Änderung des Passwortes.

h.Request-Qr-Code: Die Funktion, die einen neue Qr-Code anfordert und zurück gibt.

i.Send-Qr-Code: Die Funktion, die als Parameter einen Qr-Code erwartet. Wenn diese Funktion aufgeruft, wird die Bezahlungsprozess fertig gestellt. Diese Funktion kann von Barcode-Scanners aufgeruft werden.

j.Is-Paid: Die Funktion, um zu kontrollieren, ob die Bezahlungsprozess fertig gestellt ist.

k. Feedback: Die Funktion, um Feedback zum Server zu schicken.

l.Recommend: Die Funktion, um andere Kunde zu empfehlen, damit sie frei Shuttle oder Mahlzeit bekommen.

m. Forgot-Password: Die Funktion, die automatisch eine Mail schickt und Passwort neu stellt.

n.Donate-Item: Die Funktion, um Geld für Shuttle und Mahlzeiten zu spenden.

Die folgende Funktionen können nicht nicht von Clients aufgeruft werden:

a. Price-Check: Die Funktion, um der Preis einer Dienst für ein bestimmtes Status zu lernen.

b. CheckFreeItem: Die Funktion, die kontrolliert, ob jemand Free-Item nehmen kann.

c. TokenToCustomerIdParser: Diese Funktion sucht, wem die gegebene JWT Token gehört. Dann gibt die Matrikelnummer von die Kunde zurück.

d. GetSaltString: (Nicht Benutzt) Diese Funktion generiert ein neues zufälliges Passwort.

e. LogTransaction: Diese Funktion zeichnet alle Transaktionen in der Log-Tabelle auf.

Der Rest der Klassen sind JSON Modelle, um oben stehenden Funktionen aufzurufen. Wenn eine Funktion aufgeruft wurde, muss man ein JSON schicken, um benötigte Informationen zu geben.

**11) TauProjektApplikation.java**

Diese Klasse ist die Main Klasse von der Spring Applikation, deswegen wird die ganze Applikation hier neugestartet.

**Passwort Vergessen HTML Seite**



*Abbildung 106.*

*Abbildung 107.*

*Abbildung 104.*

*Abbildung 97.*

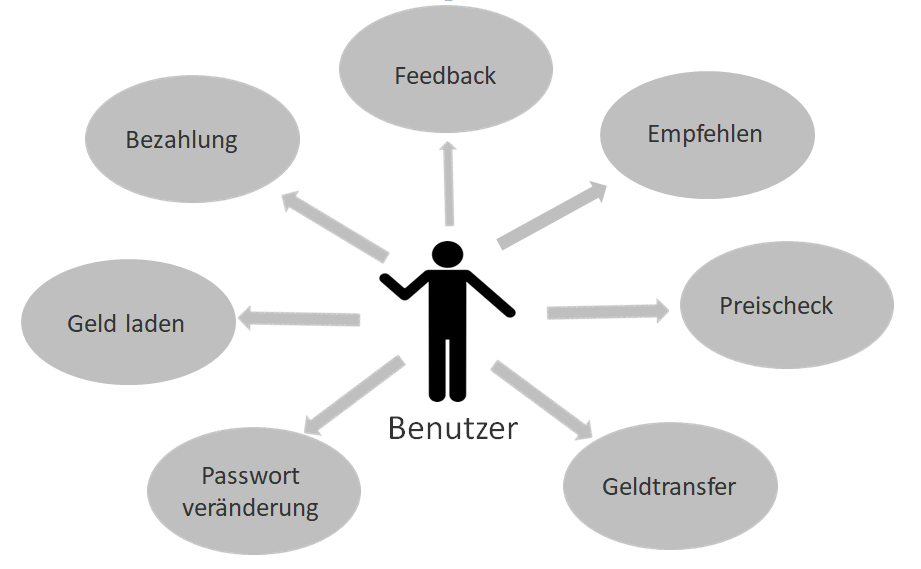
*Abbildung 97.*

In der Passwort-Wiederherstellung Seite muss man sein neues Passwort zweimal eingeben. Das neue Passwort muss länger als 5 Zeichen sein. Wenn man “Şifreyi Değiştir” klickt, die HTML Seite kommuniziert mit dem Server, dann stellt Passwort-WiederHerstellung Prozess fertig. Der Source-Code kann in dem Projekt gefunden werden.

**SSL (Secure Socket Layer)**

In diesem Projekt kommunizieren Clients mit dem Server, aber muss man sicher sein, dass sein Daten verschlüsselt sind. HTTPS-Protokoll ist eine gute Lösung für diese Situation. Um SSL Zertifikat mit dem Server zu integrieren, muss man Zertifikät in dem richtigen Format bringen. In der Linux Systems kann man OpenSSL als Werkzeug nutzen. Das benötigte Format ist. P12 für Spring Applikation.

## Anwendungsfälle



*Abbildung 107.*

*Abbildung 108.*

*Abbildung 105.*

*Abbildung 98.*

*Abbildung 98.*

Hier können Sie im Detail sehen, welche Operationen der Benutzer ausführen kann und welche Methoden für diese Operationen verwendet werden.

**Customer-Controller**: Geldtransaktionen, Kundeninformationen etc. Operationen werden hier ausgeführt.

1.[**/customers/change-password**](https://workshop-omercem.com:8443/swagger-ui.html#/operations/customer-controller/changePasswordUsingPOST): Funktion zum Ändern von Passwörtern aus den Profileinstellungen. Erforderliche Informationen sind neues und altes Passwort von dem Benutzer.

**json Model**: { “new pass”:”string”,

“old pass”: “string” }

Mögliche Antworten:

Code 200: (wrong password): Altes Passwort ist falsch.

(password changed successfully): Das Passwort wurde erfolgreich geändert.

2.[**/customers/deposit**](https://workshop-omercem.com:8443/swagger-ui.html#/operations/customer-controller/deopsitUsingPOST) : Die Funktion für Geld laden. Parameters also erforderliche Informationen sind Geldbetrag und id von Konto.

**json Model**: { "amount": 0,

"balanceId": "string" }

Mögliche Antworten:

Code 200: (balance not found) = Kontostand wurde nicht gefunden.

(deposited successfully) = erfolgreich hinterlegt

3.[**/customers/donate-item**](https://workshop-omercem.com:8443/swagger-ui.html#/operations/customer-controller/donateItemUsingPOST): Dies ist die Funktion, mit der der Benutzer spenden kann. Erforderliche Informationen sind der Geldbetrag und id von Preis.

**json Model**: { "amount": 0,

"priceId": "string" }

Mögliche Antworten:

Code 200: (price not found) = Preis wurde nicht gefunden. Die Variable “priceId” ist falsch.

(insufficient balance) = mangelhaftes Gleichgewicht

(donated successfully) = Spendenprozess wurde erfolgreich abgeschlossen.

4. [**/customers/forgot-password**](https://workshop-omercem.com:8443/swagger-ui.html#/operations/customer-controller/forgotPasswordUsingPOST): Diese Funktion wurde für Situationen entwickelt, in denen der Kunde sein Passwort vergessen kann. Erforderliche Information ist id von der Kunde.

**jsonModel** : { "id": "string"}

Mögliche Antworten:

Code 200: (user not found): Benutzer wurde nicht gefunden.

(ein Mail-String): erforderliche E-Mail an die angegebene Adresse

5.[**/customers/feedback**](https://workshop-omercem.com:8443/swagger-ui.html#/operations/customer-controller/feedbackUsingPOST): Diese Funktion wurde erstellt, um die Meinungen der Benutzer zu erhalten. Erforderliche Informationen sind Anzahl der Sterne (wenn du magst 5, sonst 4,3,2 oder 1), der Feedback-text und type von Feedback.

**json Model** : { "star": 0,

"text": "string"

"type": "string" }

Mögliche Antworten:

Code 200: (feedback successfully sent) = Feedback wurde erfolgreich hinzugefügt.

6.[**/customers/price-check**](https://workshop-omercem.com:8443/swagger-ui.html#/operations/customer-controller/priceCheckUsingPOST)**:** Funktion für Benutzer erstellt, um Produktpreise zu lernen. Erforderliche Informationen sind der Geldbetrag, id von dem Preis und Status von dem Benutzer.

**json Model**: { "amount": 0,

"priceId": "string",

"status": "string" }

Mögliche Antworten:

Code 200: (price not found) = priceId Variable ist falsch.

7.[**/customers/recommend**](https://workshop-omercem.com:8443/swagger-ui.html#/operations/customer-controller/recommendUsingPOST)**:** Eine Funktion, bei der einer Person empfohlen wird, ein hängendes Produkt zu erhalten. Erforderliche Information ist id von dem Benutzer, der Sie empfehlen möchte.

**json Model**: { "id": "string" }

Mögliche Antworten:

Code 200: (user not found) = Benutzer wurde nicht gefunden.

(recommended successfully) = erfolgreich empfohlen

8.**/customers/transfer:** Dies ist die Funktion, mit der Kunden Geld überweisen können.

**json Model:** { "amount": 0,

"balanceId": "string",

"receiverId": "string" }

Mögliche Antworten:

Code 200: (balance not found) = Der Kontostand ist falsch.

(user not found) = Der Benutzer wurde nicht gefunden.

(insufficient balance) = mangelhaftes Gleichgewicht

(transferred successfully) = Geldtransfer erfolgreich abgeschlossen.

9.[**/customers/request-qr-code**](https://workshop-omercem.com:8443/swagger-ui.html#/operations/customer-controller/getQrCodeUsingPOST)**:** Funktion, die für den Benutzer erstellt wurde, um den QR-Code anzufordern. Notwendige Information ist eine zufällige String.

10.[**/customers/is-paid**](https://workshop-omercem.com:8443/swagger-ui.html#/operations/customer-controller/isQrPaidUsingPOST)**:** Diese Funktion wurde erstellt, um eine Kontostandsabfrage in der mobilen Anwendung durchzuführen.

**json Model**: { "qrCode": "string" }

Mögliche Antworten:

Code 200: (not paid) = noch nicht bezahlt.

(qr code not found) = Die Qr-Code wurde nicht gefunden.

(insufficient balance) = mangelhaftes Gleichgewicht

(paid successfully) = Zahlungsvorgang erfolgreich abgeschlossen.

\*Code 500: “Internal Server Error”. Aufgrund eines Problems mit Ihrem Server oder Dateisystem kann dieser Fehler auftreten. Es ist mögliche Antwort für alle Funktionen.

## EER und UML Domänenmodell

UML ist eine Möglichkeit, ein Softwareprogramm mithilfe einer Sammlung von Diagrammen zu visualisieren. Die Notation wurde aus den Arbeiten von Grady Booch, James Rumbaugh, Ivar Jacobson und der Rational Software Corporation entwickelt, um für das objektorientierte Design verwendet zu werden. Seitdem wurde sie jedoch auf eine größere Anzahl von Softwareentwicklungsprojekten ausgedehnt. Heutzutage wird UML von der Objekt Management Group (OMG) als Standard für die Entwicklung von Modellierungssoftware akzeptiert.

UML steht für Unified Modeling Language. Mit UML 2.0 wurde die ursprüngliche UML-Spezifikation erweitert, um einen größeren Teil der Softwareentwicklungsbemühungen abzudecken, einschließlich agiler Praktiken.

Verbesserte Integration zwischen Strukturmodellen wie Klassendiagrammen und Verhaltensmodellen wie Aktivitätsdiagrammen.

Es wurde die Möglichkeit hinzugefügt, eine Hierarchie zu definieren und ein Softwaresystem in Komponenten und Unterkomponenten zu zerlegen.

Die ursprüngliche UML spezifizierte neun Diagramme; UML 2.x erhöht diese Zahl auf 13. Die vier neuen Diagramme heißen: Kommunikationsdiagramm, Verbundstrukturdiagramm, Interaktionsübersichtsdiagramm und Zeitdiagramm. Sie hat außerdem Zustandsdiagramme in Zustandsdiagramme umbenannt, die auch als Zustandsdiagramme bezeichnet werden.

Diese Diagramme sind in zwei Gruppen unterteilt: Strukturdiagramme und Verhaltens- oder Interaktionsdiagramme.

**Strukturelle UML-Diagramme**

Klassen Diagramm

Paketdiagramm

Objektdiagramm

Komponentendiagramm

Zusammengesetztes Strukturdiagramm

Bereitstellungsdiagramm

**Verhaltens-UML-Diagramme**

Aktivitätsdiagramm

Sequenzdiagramm

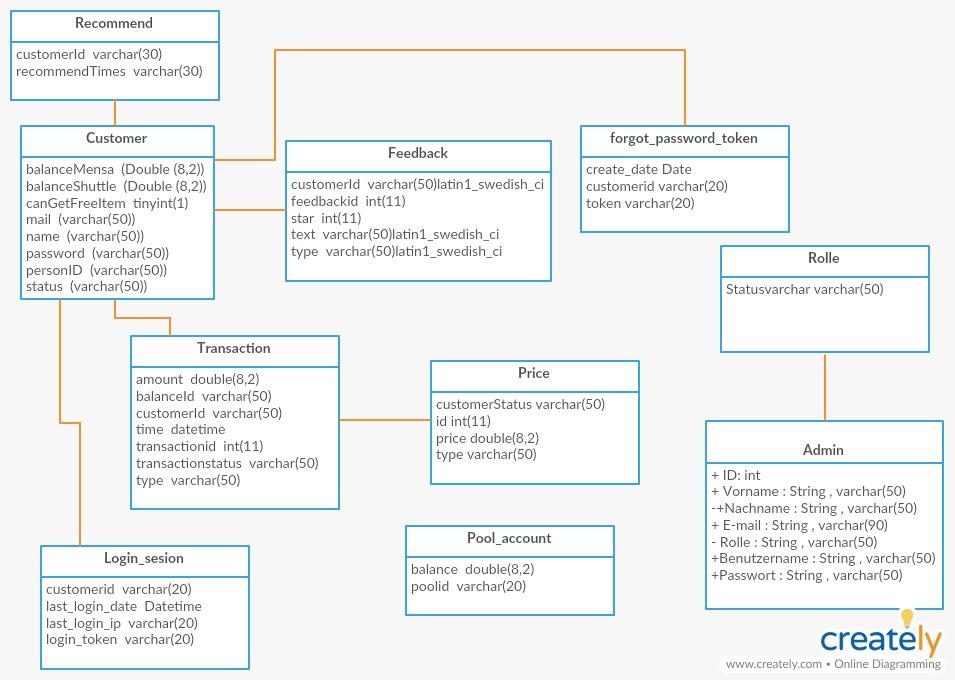
Anwendungsfalldiagramm

Zustandsdiagramm

Kommunikationsdiagramm

Interaktionsübersichtsdiagramm

Zeitdiagramm



*Abbildung 108.*

*Abbildung 109.*

*Abbildung 106.*

*Abbildung 99.*

*Abbildung 99.*

## Funktionale und nichtfunktionale Anforderungen

**Funktionale Eigenschaften**

Dieses Teilprojekt beschäftigt sich mit Backend. Das Hauptziel ist dieses Teilprojektes ein Server und Datenbank zu erstellen. Prozess läuft rechtzeitig bei dem Hintergrund und Vordergrund. Um dieses ganze System zu verwirklichen braucht jedes ein Background. Backend besorgt die Verknüpfungen, Befehlen usw vom Hintergrund für ganzes System zu durchlaufen.

* Die Abfragen in der Datenbank der vordefinierten Anfragen aus iOS und Android
* Die Überprüfung des Clients ob es den Benutzer in Datenbank gibt
* Nach der Überprüfung wird die Erlaubnis zum Eintritt gegeben
* Ein einzigartiger Code für jeden QR Code
* Bestimmung des ankommenden QR Code mit dem vorhandenen Code in der Datenbank
* Änderung zu vornehmen beim Datenbank nach den Abfragen

**Nichtfunktionale Eigenschaften**

* **Effektivität**

(System funktioniert erfolgreich. Die Daten werden aus QR-Code, IOS-Android, Webseite bekommt und durch Datenbank gespeichert, geändert, gelöscht.)

* **Zuverlässigkeit des Systems**

(Bis das Server und Datenbank werden geknackt, ist der Backend-Prozess Sicherheit, nachhaltig und erfolgreich zu funktionieren.)

* **Effizienz**

(Hilfe von den benutzten Algorithmen werden die Tasks ohne Zeit- und Kostenverlust erledigt. System reagiert schnell und stabil. )

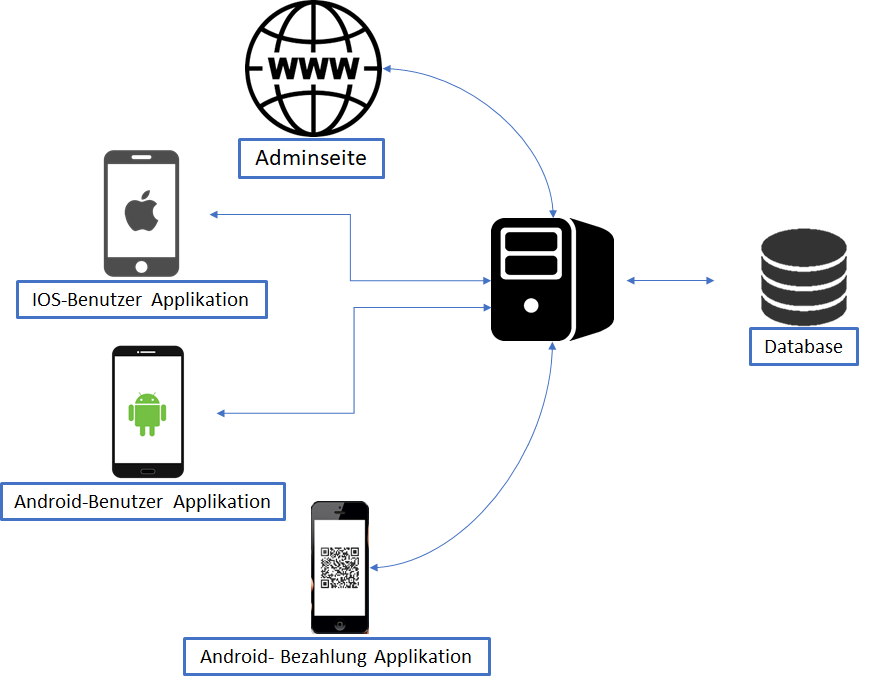
* **Bedienbarkeit**

(Die jeder TDU Studenten und Lehrpersonen kann einfach dieses System einen Zugriff haben und die Innovationen von diesem System ausnutzen.)

* **Systemsicherheit**

(Server und Database sind mit bestimmten und wechselbaren Kennzahl greifbar. Außerdem kann man dieses System aus Smartphone und Webseite nach eigenen Kundenpasswort greifen werden. Es ist unmöglich, dass die ein Person von anderen Passwort zu greifen.)

## Systemarchitektur



*Abbildung 109.*

*Abbildung 110.*

*Abbildung 107.*

*Abbildung 100.*

*Abbildung 100.*

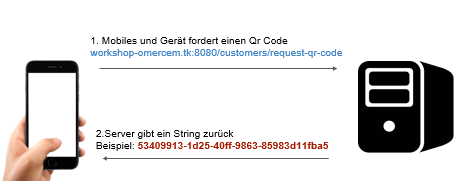
Das Backend-System führt alle Funktionen des Systems im Hintergrund aus. Beispiel: Abfragen der in der Datenbank registrierten Benutzerinformationen nach der Anmeldung, Erstellen eines QR-Codes zum Ausführen des Zahlungsprozesses, Senden des generierten QR-Codes an die verwendeten Applikationen und Überprüfen des Zahlungsprozesses.

Bevor diese Vorgänge ausgeführt werden, werden der im Backend System erstellte Datenbank Dump- und die Spring .jar Datei auf den Server geladen. In Spring .jar Datei die von den Applikationen zu verwendenden Funktionen definiert sind. Dazu müssen zuerst einige Schritte ausgeführt werden:

* Da der Server auf Linux basiert, ist vsftpd (FTP-Server) installiert.
* Das HTTPS-Protokoll gewährleistet die Sicherheit von Transaktionen zwischen Servern.
* PuTTY stellt über das Terminal eine Verbindung zum Linux-Betriebssystem her.
* Die Datenbank Dump und Spring Jar-Dateien mit FileZilla werden auf dem Server übertragen.

Zusätzlich werden die Teile von Systemarchitekturen im Backend-System sowie in Abb. X, Y, Z angezeigt.

**Wie läuft der Befehl?**



*Abbildung 110.*

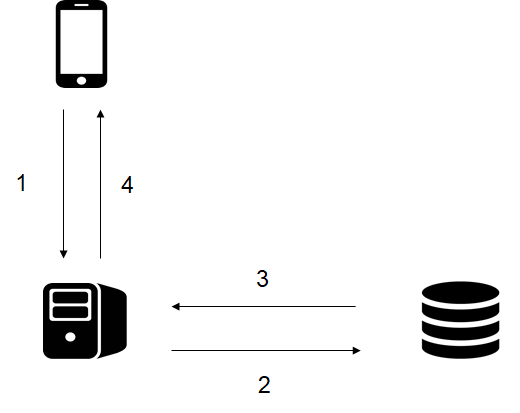
*Abbildung 111.*

*Abbildung 108.*

*Abbildung 101.*

*Abbildung 101.*

**Informationsanfrage von Benutzern**



*Abbildung 111.*

*Abbildung 112.*

*Abbildung 109.*

*Abbildung 102.*

*Abbildung 102.*

Wenn der Benutzer die Kontoinformationen nach erfolgreicher Anmeldung anfordert, sind die Hintergrund Transaktionen wie folgt:

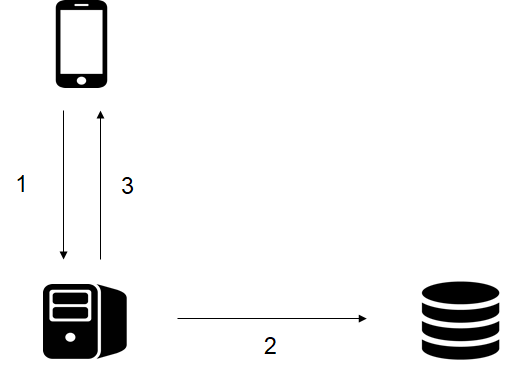
1. Die von Benutzer verwendete Applikation fragt den Befehl '/ get-info' ab.

2. Der Server sucht die Benutzerinformation mit ‘Customer ID’ in der ‘customer’ Tabelle in DB.

3. DB gibt Benutzerinformationen über den Server zurück.

4. Die Informationen werden an den Benutzer übertragen.

**Generierung der QR Code**



*Abbildung 112.*

*Abbildung 113.*

*Abbildung 110.*

*Abbildung 103.*

*Abbildung 103.*

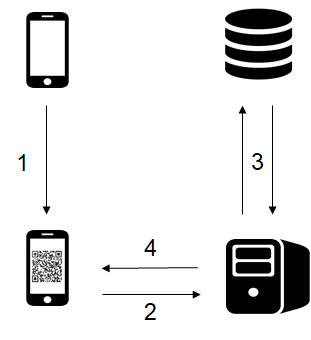
Der QR-Code wird erstellt, damit der Benutzer ihn während der Zahlung verwenden kann. Dieser QR-Code wird anhand der Kartennummer erstellt, die der Benutzer an der Türkisch-Deutsche Universität verwendet. Zum Beispiel Studentennummer, Personalnummer, Gäste Kartennummer und so weiter. Nachdem der QR-Code erstellt wurde, werden die folgenden Schritte ausgeführt:

1. Der Benutzer fordert einen QR-Code an, um die Zahlung durchzuführen.

2. Der Server generiert einen QR-Code und speichert diesen in der DB.

3. Der QR-Code wird an den Benutzer gesendet für Zahlungsoperation.

**Senden der QR Code**



*Abbildung 113.*

*Abbildung 114.*

*Abbildung 111.*

*Abbildung 104.*

*Abbildung 104.*

Nachdem der QR-Code eingegangen ist, wird der QR-Code auf einem anderen Smartphone über die Applikation von QR-Code Leser erstellt. Während des Bezahlvorgangs laufen im Backend-System folgende Schritte ab:

1. QR-Code wird von der Applikation der QR-Code-Leser gescannt

2. Applikation der QR-Code Leser sendet QR-Code an den Server

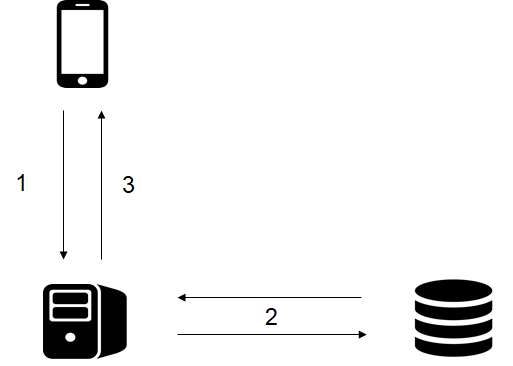
3. Prüft man, ob der QR-Code noch existiert ist.

3.1 Wenn ja; Fahrt mit der Meldung Akzeptiert mit Schritt 4 fort.

3.2 Wenn nein; Fahrt mit der Meldung Abgelehnt mit Schritt 4 fort.

4. Die Antwort vom Server wird an die Applikation der QR-Code Leser übertragen.

**Zahlungskontrolle**



*Abbildung 114.*

*Abbildung 115.*

*Abbildung 112.*

*Abbildung 105.*

*Abbildung 105.*

Das System kontrolliert kontinuierlich die Zahlung während Bezahlungsprozesses. Dieser Vorgang läuft im Backend-System folgendermaßen ab:

1. Die Benutzerapplikation fragt, ob der Transaktion erfolgreich ist.

2. Der Server fragt den DB, ob der Transaktion fertig ist.

3. Das Ergebnis aus dem DB wird mit dem Server an die Benutzerapplikation gesendet.

## Zusammenfassung und Ausblick

In diesem Projekt wurde ein großer Mehrwert für die Türkisch-Deutsche Universität geschaffen. Verbesserungssystem wird für aktuell verwendete Systeme vorgeschlagen. Das Backend System des Projektes sind für Applikationen von Mensa und Shuttle an Türkisch-Deutschen Universitäten funktionsfähig.

Dieses System, zum Beispiel als System der Bezahlung in der Mensa und in den Shuttles, werden den Studenten, Dozenten und andere Personen in der TDU das Leben erleichtern. Das Backend-System wird zwischen dem entwickelten Smartphone und Website Applikationen effektiv verbunden. Durch die erstellte Datenbank kann der Benutzer ihre Informationen über die Applikationen speichern, abfragen und ändern. Im Backend System des Projekts wurde eine Benutzeroberfläche entgegen des Vergessens von den Benutzern ihre Passwörter erstellt. Mit dieser Benutzeroberfläche können Benutzer das Passwort leichter ändern.

Schließlich werden die viele Operationen in diesem Teil des Projekts im Hintergrund des Systems mithilfe von Funktionen, Protokollen und Verbindungen ausgeführt.

# NextSteps

## Projektumsetzung

Türkisch-Deutsche Universität ist ziemlich eine sehr neue Universität im Verhältnis zu anderen Universitäten in der Türkei. Zu der Zeit studieren fast 1500 Studierende an unserer Universität, aber die Infrastruktur der Universität wird immer noch gebaut, deshalb fehlt unsere Universität einige Eigenschaften (z.B Shuttle mit Studentenkarte einsteigen). Natürlich nicht jede Universität hat diese Eigenschaften, aber Probleme entstehen von Feedbacks der TDU-Studenten. Unsere Universität ist neu, und fast alle TDU-Studenten haben mindesten einmal gesagt, dass unsere Universität schlecht, mangelhaft und nicht attraktiv ist, besonders wegen ihrem Standort in Istanbul. Als Ingenieurstudenten der Türkisch-Deutschen Universität ist es dann unsere Aufgabe, unsere Universität attraktiv, technologisch fortgeschritten und ein Bespiel für andere Universitäten zu machen. Deswegen helfen wir die Universität bei der Entwicklung unserer Universität, indem wir die Dienste und die Systeme mit einem App verbinden.

## Konzept 1: Die Bestellung von der Kantine mittels mobiler Applikation (Berk Karaağaç)

### Einleitung

Türkisch-Deutsche Universität ist eine neue Universität, ihre Gebäude werden noch gebaut und natürlich hat Mangeln dazu. Der Studentenanzahl steigt exponentiell jedes Jahr, aber der Infrastruktur kann den Studentenanzahl noch nicht unterstützen. Diese Situation führt zu vielen Problemen. Eine davon ist, dass die Dienstgeschwindigkeit der Kantinenpersonal nicht ausreichend ist, um fast alle Studenten der Universität zu dienen. Während wir auf den Bau der Universität warten, Systeme können entwickelt werden als ein temporäre/permanente als Lösungen dazu.

#### Problemstellung

Eines der Probleme an der TDU ist weniger Zeit zum Essen zu haben. Manchmal können die Studierenden wegen der Intensität des Unterrichtsprogramms keine Zeit zum Essen finden. Wenn sie kleine Zeit zum Essen finden, können sie diesmal aufgrund des Gedränges in der Mensa etwas nicht essen. Deswegen haben wir ein Konzept dafür gedacht. Das Konzept ist eine bereits bestehende Bestellung. Vor der Pause bestellen wir das Essen durch eine Mobilapplikation. Das System gibt uns ein QR Code für die Bestellung. Das Essen wird vorbereitet bevor wir es abholen. Und dann, wenn wir das Essen abholen wollen, zeigen wir einfach die QR Code und holen die Essen ab.

#### Stand der Technik

#### Zielsetzung

Das Ziel ist, ein System zu entwickeln, das die Essenbestellung vereinfacht. Das Essen soll von der mobilen Applikation bestellt und bezahlt werden, dann mit der anscheinenden QR Code, in der Kantine abgeholt werden, ohne zu warten für eine Sekunde.

#### Aufbau des Projekthandbuches

Im Kapitel „Design“ wurde die Anwendungsfälle, UML Diagramme, Funktionale- und nicht Funktionale Anforderungen, allgemeine Systemarchitektur und Auslegung der Software mit simpel Design erläutert.

Im Kapitel „Evaluation“ wurde Grafiken von Antworten zur zwei Fragen dargestellt.

Im Kapitel „Zusammenfassung und Ausblick“ wurde ein Fazit von dem ganzen Projekt erläutert.

### Anforderungsanalyse und Konzeption

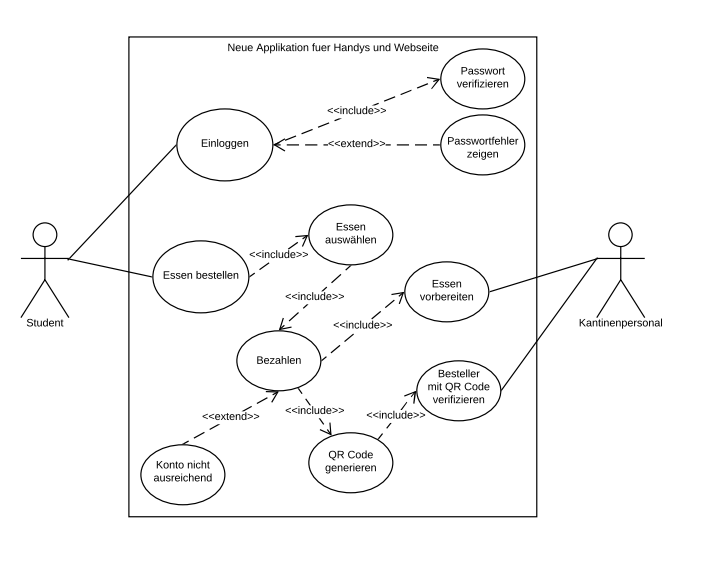
Die Aufgabe ist, das Bestellungssystem in der Kantine zu verbessern. Um eine Innovation in diesem Bereich zu machen, ist es nötig, mit dem zu entwickelnden Applikation ein onlines Essenbestellungssystem zu implementieren.

### Design (Entwicklung) und Implementierung

#### Anwendungsfälle

1. Ein Student verliert viel Zeit beim Warten für die Essenvorbereitung. Mit dem Vorbestellen von Essen, die Studenten können dann 15 Minuten vor der Pause ihre Essen schon bestellen.
2. Manchmal ist die Kantine sehr beschäftigt und die Kantinenpersonal können nicht immer auf die speziellen Anforderungen von Studenten achten. Das führt zu Studentenunzufriedenheit. Mit einer schriftlichen Bestellung ist die Bestellung noch deutlicher.

#### EER und UML Domänenmodell

****

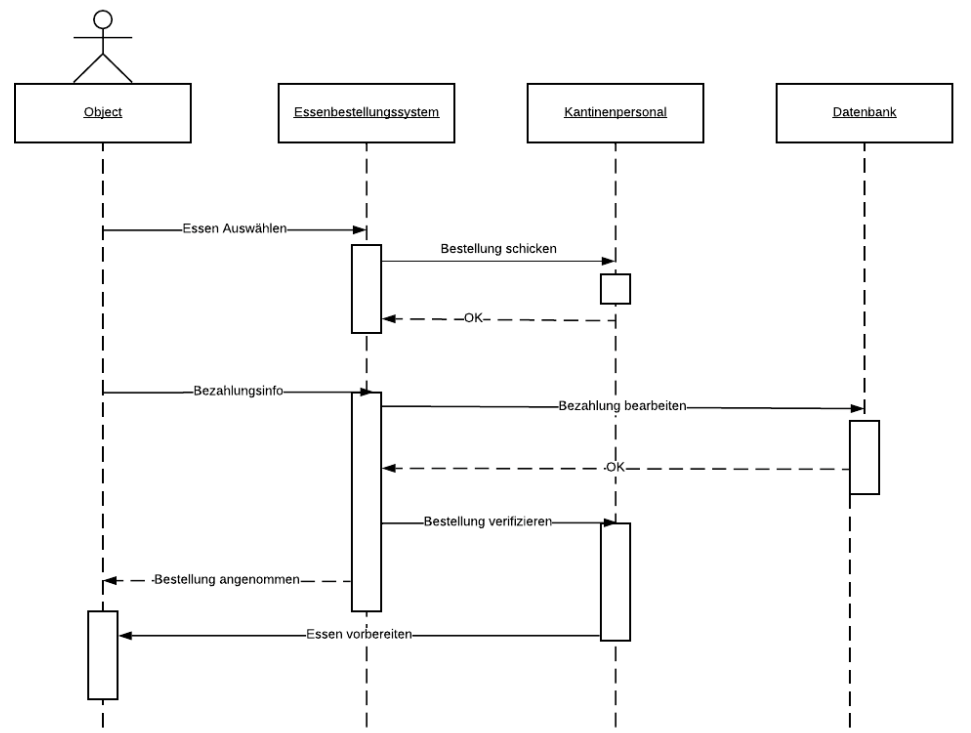
*Abbildung 115.*

*Abbildung 116.*

*Abbildung 113.*

*Abbildung 106.*

*Abbildung 106.*



*Abbildung 116.*

*Abbildung 117.*

*Abbildung 114.*

*Abbildung 107.*

*Abbildung 107.*

#### Funktionale und nichtfunktionale Anforderungen

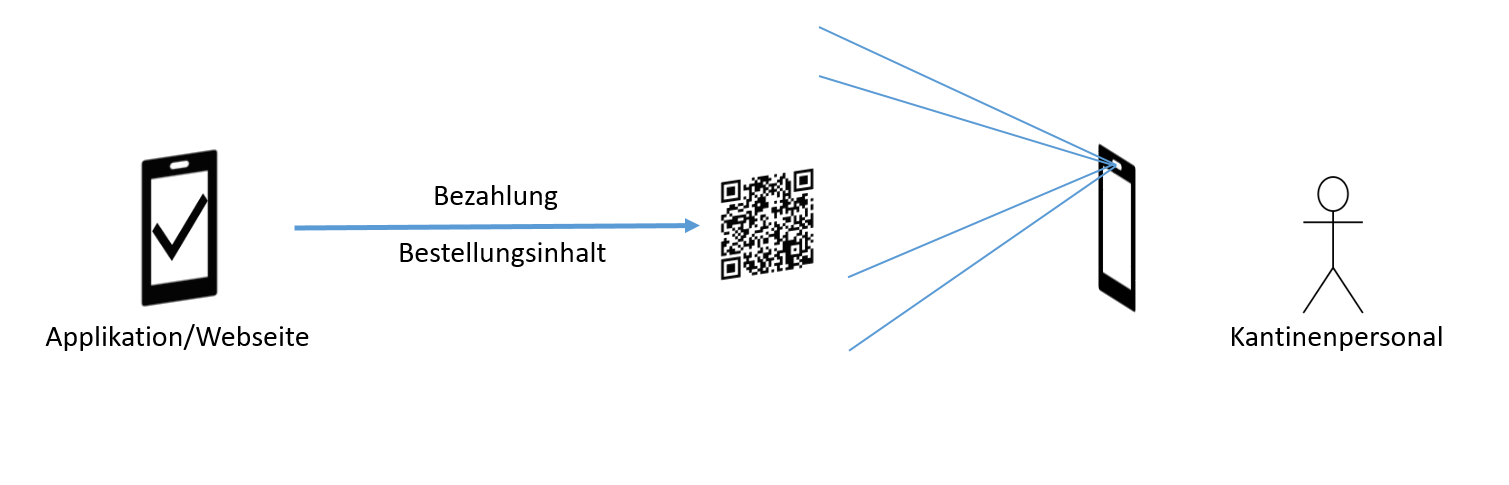
Funktionale Anforderungen:

* Die Studenten sollen in das System mit ihrem ID und Passwort einloggen können
* Die Studenten sollen aus der digitalen Speisekarte Essen auswählen
* In der Speisekarte soll eine Liste von verfügbaren Essen gelistet werden und die Liste soll jede Woche aktualisiert werden
* Nach der Auswahl, können die Studenten mit ihrem Studentenkonto für das Essen bezahlen können
* Nach der Bezahlung soll einen QR-Code anscheinen
* In QR-Code soll der Studentennummer, Bestellungsinhalt und ein temporärer einzigartiger Code stehen, damit die Sicherheit hoch ist
* Der QR-Code soll mit der mobilen Applikation von TDU gelesen werden können
* Nach Lesung, Kantinenpersonal soll die QR-Inhalt verifizieren und die Bestellung beenden.

Nicht-Funktionale Anforderungen:

* Übertragbarkeit: Das System soll auf Basis des Ziels, übertragbar sein. Online Essenbestellungen sollen von der Klasse oder von der Bibliothek gemacht werden können.
* Sicherheit: Da es Geldtransaktionen im System gibt, die Sicherheit des Systems muss erst hergestellt werden, um eine sichere Umgebung zu erhalten.
* Zuverlässigkeit: Die Bestellungen sollen richtig an die Kantinenpersonal geleitet werden, um Zeit zu sparen.
* Benutzbarkeit: Das System soll einfach benutzt werden können ohne vorherige Kenntnisse.
* Kompatibilität: Das System soll auf Android/iOS/Webseite/Mobile Webseite arbeiten.
* Effizienz: Das Ziel des gesamten Systems ist es, Zeit zu sparen. Das heißt, das System soll schnell genug arbeiten. Kantinenpersonal sollen über die Nutzung des Systems trainiert werden.

#### Systemarchitektur



*Abbildung 117.*

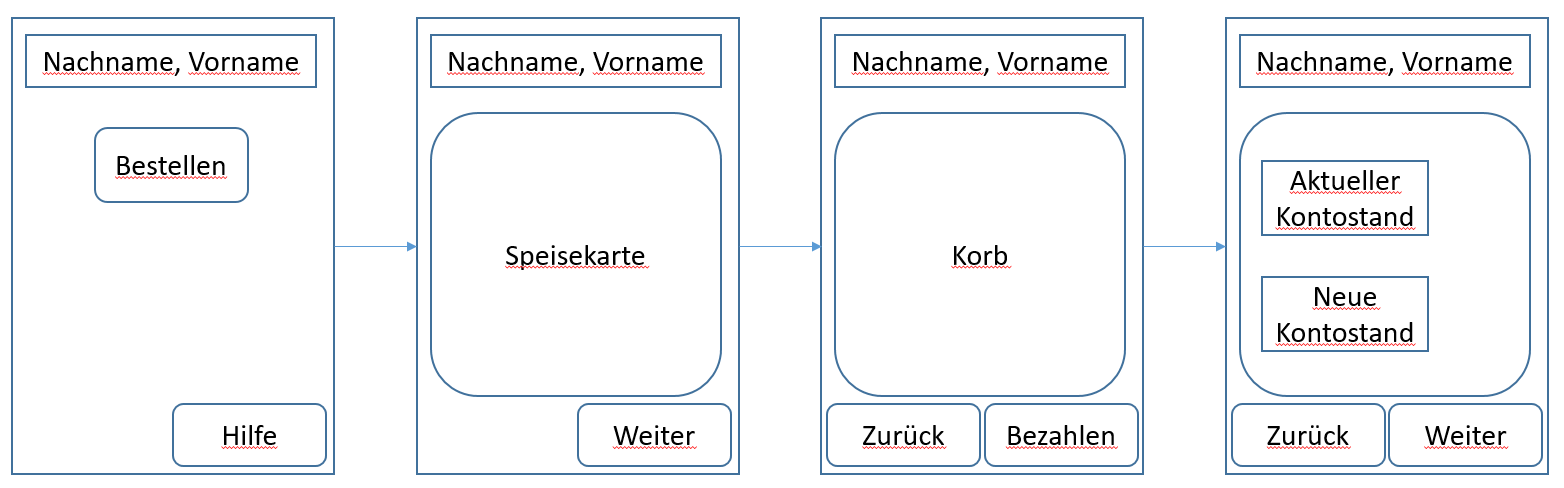
*Abbildung 118.*

*Abbildung 115.*

*Abbildung 108.*

*Abbildung 108.*

#### Design-Layouts



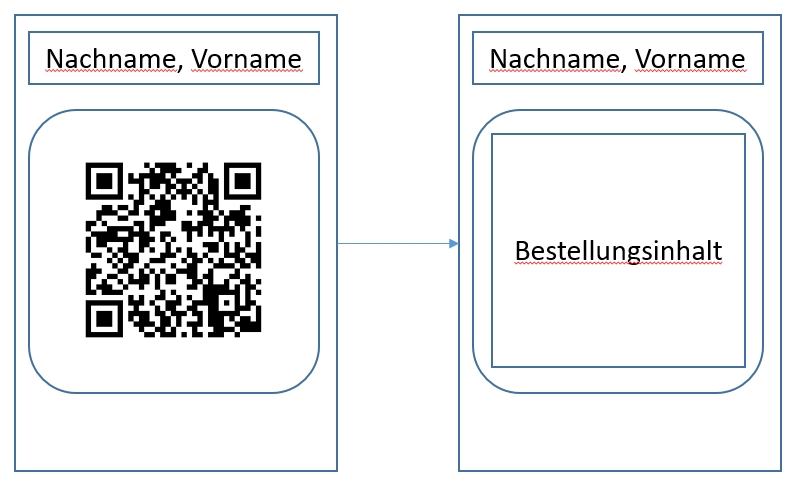
*Abbildung 118.*

*Abbildung 119.*

*Abbildung 116.*

*Abbildung 109.*

*Abbildung 109.*



*Abbildung 119.*

*Abbildung 120.*

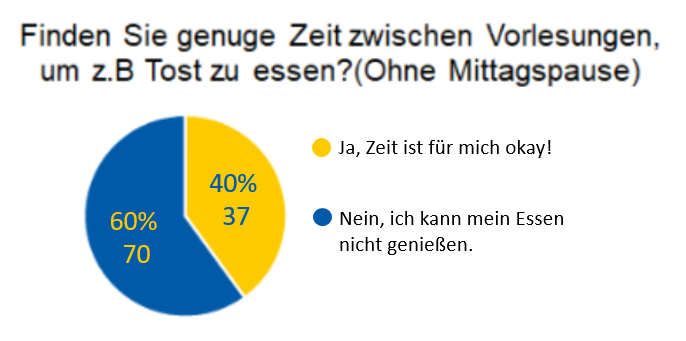
*Abbildung 117.*

*Abbildung 110.*

*Abbildung 110.*

### Evaluation

Für die Evaluation des Konzepts wurde eine Umfrage via Google Forms erstellt. 117 Studenten haben die Umfrage beantwortet. Das Ziel von dieser Umfrage war, um zu sehen, ob die Studenten wirklich so ein System brauchen und/oder möchten.



*Abbildung 120.*

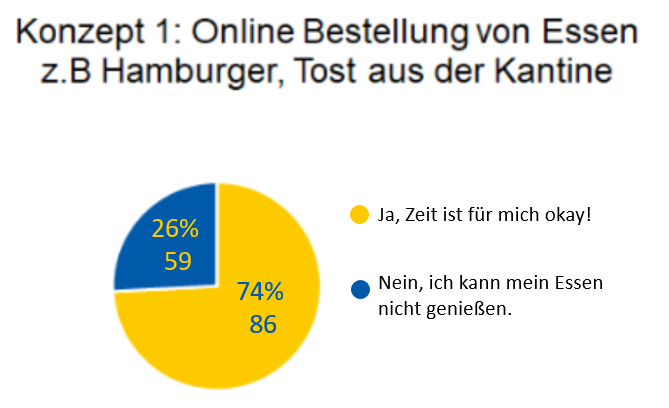
*Abbildung 121.*

*Abbildung 118.*

*Abbildung 111.*

*Abbildung 111.*

In dieser Grafik sieht man ob die Studenten Schwierigkeiten haben beim Essen zwischen Vorlesungen. Die Pausen sind meistens zehn Minuten, also die Studenten müssen mindestens die Halbe ihrer Pausen auf ihre Essen warten.



*Abbildung 121.*

*Abbildung 122.*

*Abbildung 119.*

*Abbildung 112.*

*Abbildung 112.*

In dieser Frage, wurde die Studenten gefragt, ob sie so ein System möchten. Für etwas Neues zu entwickeln, muss es nicht ein Problem geben. Manchmal wenn eine Idee interessant genug ist, es kann auch attraktiv sein für ihre potentiellen Benutzer.

### Zusammenfassung und Ausblick

Dieses Konzept kann eine Innovation sein im Bereich von Universitätskantinen. Da in dem Stand der Technik keine solche Anwendungen gibt, ist es attraktiv für die Studenten Essen aus der Kantine zu bestellen. Die Prozesse an den Universitäten sind schon zu schwierig. Man kann von einem einfachen Punkt anfangen, die Universitäten zu verbessern im Rahmen der Innovation

## Konzept 2: Der Eintritt zur Universität mittels mobiler Applikation (Batuhan Özsoy)

### Einleitung

Die Studierenden treten in die TDU wie in jeder Universität mit ihrem Studentenausweis ein**.** Mit der Entwicklung von der Technologie sollen wir als Ingenieur Studierenden neue Lösungen zum Eintritt finden**.**

#### Problemstellung

Studierenden an der TDU treten in die TDU mit ihrem Studentenausweis ein. Wenn sie ihren Studentenausweis vergessen, können sie nur in die TDU eintreten, indem sie ihren Ausweis beim Sicherheitskabine bleiben lassen. Beim Ausgang nehmen sie ihren Ausweis zurück. Manchmal vergessen sie auch ihren Ausweis. Das kann ärgerlich sein.

#### Zielsetzung

Das Hauptziel dieses Konzepts besteht darin, den Studierenden die Möglichkeit zu geben, mit ihrem eigenen QR-Code in die TDU einzutreten.

Jede Studierende an der TDU hat eigene QR-Code wie ein Studentenausweis in mobiler Applikation. Beim Eingang zeigen sie ihren eigenen QR-Code dem QR-Codeleser statt ihrem Ausweis. Sie können damit in die TDU einfach eintreten

Was ist nicht unser Ziel?

Wir planen dieses Konzept und führen die notwendigen Analysen durch. Wir machen jedoch keine Softwareentwicklung, um das Konzept zu realisieren. (Entwicklung einer Applikationsseite für die Lehrkräfte, um den Antrag zu genehmigen, usw. sind nicht in unsere Verantwortung.)

Gleichzeitig sind wir nicht für die Analyse, Auswahl und Beschaffung der für das System erforderlichen Hardware verantwortlich. (QR-Code-Leser beim Eingang, Vorbereitung des Verriegelungsmechanismus ist nicht unsere Aufgabe)

#### Aufbau des Projekthandbuches

Im Kapitel „Design“ wurde die Anwendungsfälle, UML Diagramme, Funktionale- und nicht Funktionale Anforderungen, allgemeine Systemarchitektur und Auslegung der Software mit simpel Design erläutert.

Im Kapitel „Evaluation“ wurde Grafiken von Antworten zur zwei Fragen dargestellt.

Im Kapitel „Zusammenfassung und Ausblick“ wurde ein Fazit von dem ganzen Projekt erläutert.

### Methodische Ermittlung

#### Methoden und Werkzeuge

### Anforderungsanalyse und Konzeption

Funktionale Anforderungen:

* Das System sollte es die Studierenden an der TDU ermöglichen, in die TDU mit QR-Code einzutreten.
* Im System können Verwaltung von TDU den QR-Code von Studenten aktivieren.

Nicht- funktionale Anforderungen:

* Benutzerfreundlichkeit, gut aussehend, Korrektheit
* Das System soll den Studierenden innerhalb von einer Sekunde antworten.

Für Benutzer:

* Smartphone
* Eintritt in die TDU mit QR-Code

Für System:

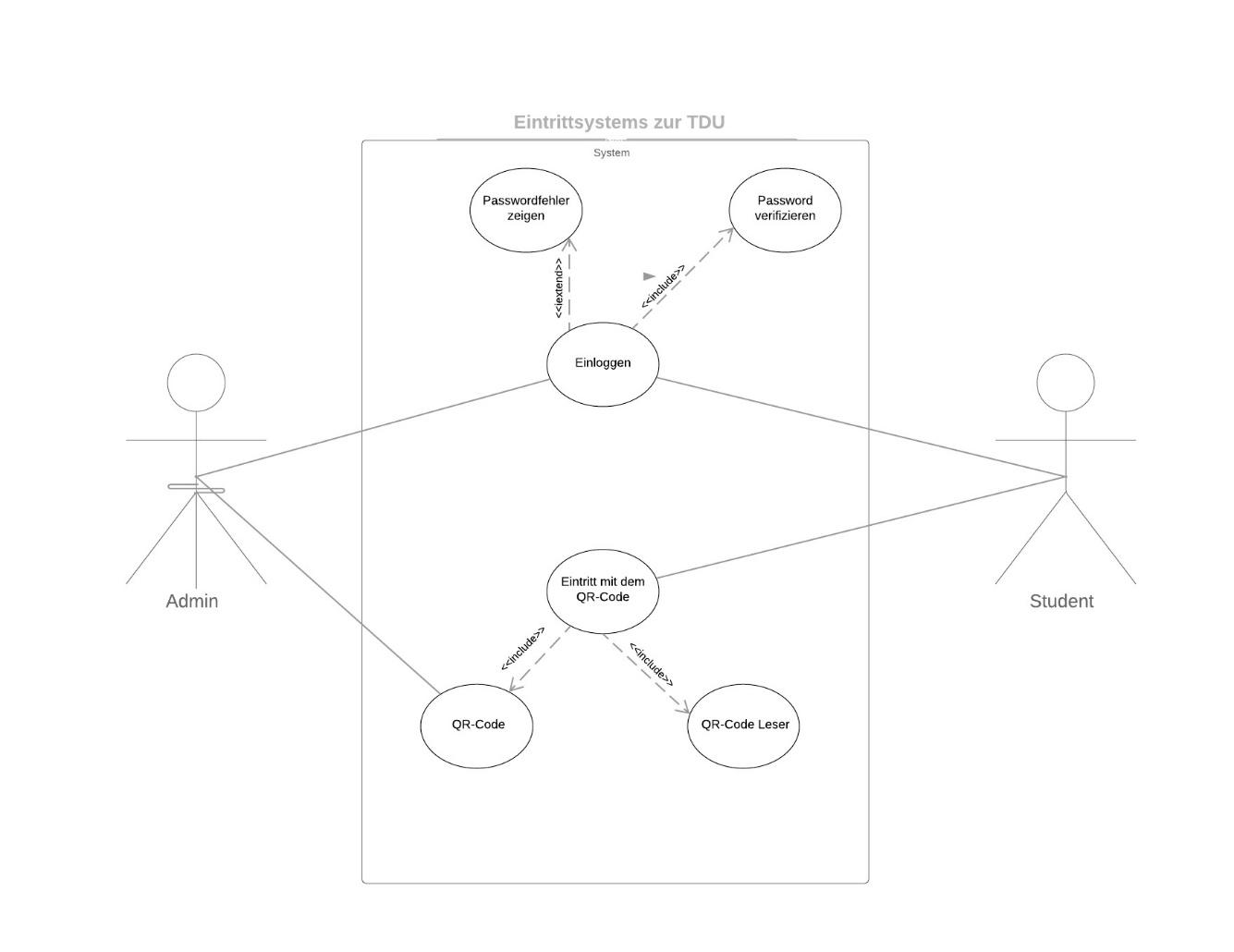
* Android Studio und Swift
* Windows und MacOS Betriebssystem
* QR-Codeleser

### Design (Entwicklung) und Implementierung

#### Anwendungsfälle

Es wurde in einem Use-Case Diagramm gezeichnet, um die Verwendung des Systems zu verstehen. Es gibt 2 Aktor. Sie sind Admin und Student. Student möchte in die TDU eintreten und Admin soll QR-Code für jede Studenten erstellen.

#### EER und UML Domänenmodell



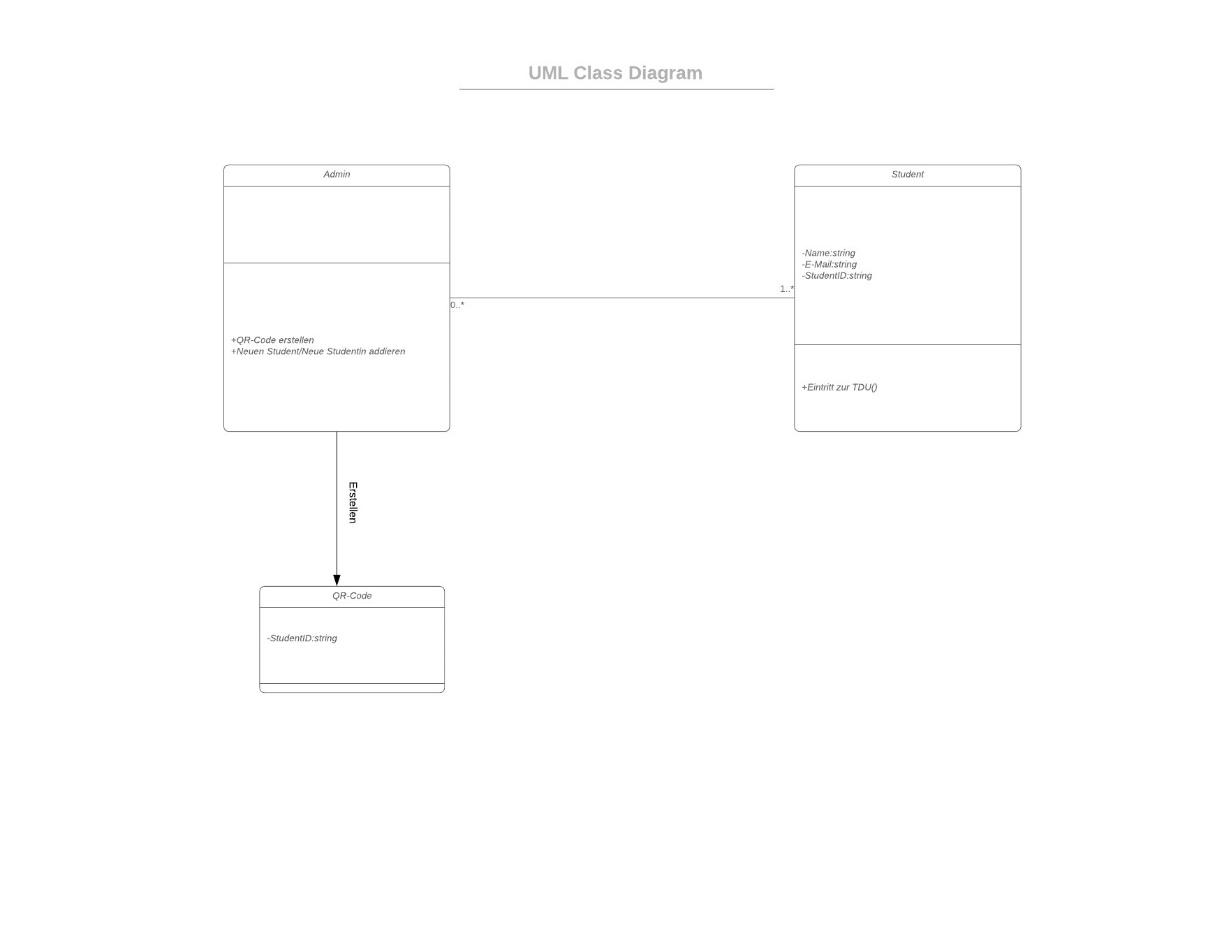
*Abbildung 122.*

*Abbildung 123.*

*Abbildung 120.*

*Abbildung 113.*

*Abbildung 113.*



#### Funktionale und nichtfunktionale Anforderungen

* Das System sollte es die Studierenden an der TDU ermöglichen, in die TDU mit QR-Code einzutreten.
* Im System können die Verwaltung von TDU den QR-Code von Studenten aktivieren.
* Die Studenten sollen in das System mit ihrem ID und Passwort einloggen können.
* Beim Eintritt soll sich ein QR-Code Leser befinden.
* In QR-Code sollen die Studenteninformationen stehen, damit die Sicherheit hoch ist.
* Der QR-Code soll mit der mobilen Applikation von TDU gelesen werden können.

Nicht-Funktionale Anforderungen:

* Sicherheit: QR-Code wird von der Verwaltung der TDU den Studierenden gegeben.
* Zuverlässigkeit: Jede Studierende hat seinen eigenen QR-Code wie Studentenausweis zum Eintritt.
* Benutzbarkeit: Das System soll einfach benutzt werden können ohne vorherige Kenntnisse.
* Kompatibilität: Das System soll auf Android/iOS/Webseite/Mobile Webseite arbeiten.
* Effizienz: Das Ziel des gesamten Systems ist es, Zeit zu sparen. Das heißt, das System soll schnell genug arbeiten.
* Reaktionszeit: Das System soll den Studierenden innerhalb von einer Sekunde antworten

#### Systemarchitektur

* Das System muss im Android und Ios System funktioniert.
* QR-Code mit nötigen Informationen soll erstellt werden.
* QR-Codeleser soll sich beim Eingang befinden.



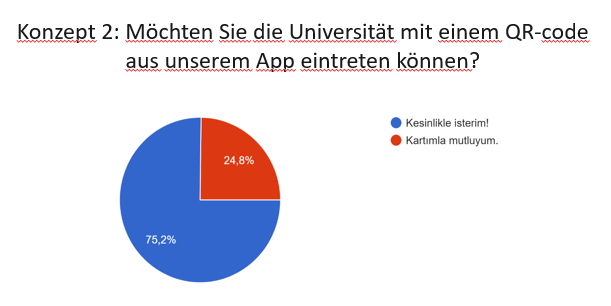


#### Design-Layouts



### Evaluation

Für die Evaluation des Konzepts wurde eine Umfrage via Google Forms erstellt. 117 Studenten haben die Umfrage beantwortet. Das Ziel von dieser Umfrage war, um zu sehen, ob die Studenten wirklich so ein System brauchen und/oder möchten.



In dieser Frage, wurde die Studenten gefragt, ob sie so ein System möchten. Für etwas Neues zu entwickeln, muss es nicht ein Problem geben. Manchmal wenn eine Idee interessant genug ist, es kann auch attraktiv sein für ihre potentiellen Benutzer.

### Zusammenfassung und Ausblick

Dieses Konzept kann eine Innovation sein im Bereich von Universitätseintritt. Da in dem Stand der Technik keine solche Anwendungen in der Türkei gibt, ist es attraktiv für die Studenten in die Universität einzutreten. Die Prozesse an den Universitäten sind nicht zu schwierig. Man kann von einem einfachen Punkt anfangen, die Universitäten zu verbessern im Rahmen der Innovation.

## Konzept 3: Der Eintritt zu den Laboren mittels mobiler Applikation (Sebiha Gökçem Hakbilen)

### Einleitung

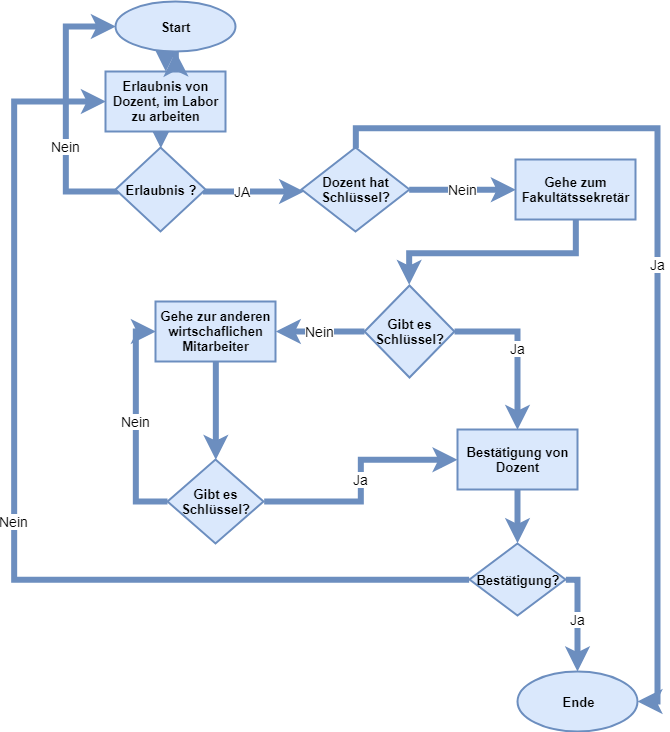
Die Türkisch-Deutsche Universität ist eine sich entwickelnde und wachsende Universität. Die Anzahl der Laboratorien an der Universität nimmt von Tag zu Tag zu. Diese Labore bieten den Studierenden die Möglichkeit, unterschiedliche Forschungs- und Entwicklungsarbeiten durchzuführen. Es ist jedoch nicht möglich, dass diese Laboratorien zu jeder Zeit Schüler einsetzen. In diesem Konzept werden wir dieses Thema diskutieren.

#### Problemstellung

In unserer Universität gibt es verschiedene Laboratorien, die verschiedenen Fakultäten angehören. Zusätzlich zu den Vorlesungen im Labor müssen die Studierenden im Labor anwesend sein, um Projekte und Praktiken zu entwickeln.

Die Labore sind aus bestimmten- und aus Sicherheitsgründen gesperrt. Die Schüler müssen die erforderlichen Berechtigungen für die Arbeit erhalten. Auch wenn wir die Erlaubnis erhalten, suchen wir stundenlang nach Schlüsseln, da der Schlüssel nicht immer in bestimmten Personen vorhanden ist. Obwohl wir Erlaubnis von dem Dozenten erhalten, leben wir Probleme mit Person, die wir Schlüssel bekommen können.

Daher wollen wir unsere Projekte in Laboren weiterentwickeln und arbeiten. Aber wir haben immer Probleme, die Schlüssel zu finden.



*Abbildung 123.*

*Abbildung 124.*

*Abbildung 121.*

*Abbildung 114.*

*Abbildung 114.*

#### Stand der Technik

#### Zielsetzung

Das Hauptziel dieses Konzepts besteht darin, den Studierenden die Möglichkeit zu geben, die Laboratorien zu nutzen, wann sie möchten, und eine Genehmigung zu erhalten.

Ein QR-Codeleser befindet sich an den Labortüren, und nur autorisierte Personen können die QR-Codes lesen lassen und das Labor betreten. Studenten, zum Beispiel, nachdem sie die Erlaubnis von einem Dozenten erhalten haben, autorisiert der Dozent den Studenten von der mobilen Applikation zum Labor. Danach wird der QR-Code des Studenten aktiviert, um das Labor zu betreten, und der Student kann den QR-Code lesen lassen. So können Studenten Labor nutzen.

Kurz gesagt, wird der Schüler von der Schlüsselsuche entlastet. Student können das Labor in dem Moment, in dem er vom Dozenten die Erlaubnis erhält, benutzen.

Was ist nicht unser Ziel?

Wir planen dieses Konzept und führen die notwendigen Analysen durch. Wir machen jedoch keine Softwareentwicklung, um das Konzept zu realisieren. (Entwicklung einer Applikation Seite für die Lehrkräfte, um den Antrag zu genehmigen, usw. sind nicht in unsere Verantwortung.)

Gleichzeitig sind wir nicht für die Analyse, Auswahl und Beschaffung der für das System erforderlichen Hardware verantwortlich. (QR-Code-Leser in Labortüren, Vorbereitung des Verriegelungsmechanismus ist nicht unsere Aufgabe.

### Methodische Ermittlung

#### Methoden und Werkzeuge

### Anforderungsanalyse und Konzeption

Funktionale Anforderungen:

* Das System sollte es Benutzern ermöglichen, das Labor mit dem QR-Code einzugeben
* Im System können Dozenten den QR-Code von Studenten aktivieren.
* Nur die autorisierten Benutzer können an Labortür eintreten

Nicht- funktionale Anforderungen:

* Benutzerfreundlichkeit, gutaussehend, Korrektheit
* Das System soll dem Anwender innerhalb von einer Sekunde antworten.
* Der QR-Code muss dem Studenten unmittelbar nach seiner Aktivierung durch den Dozenten mitgeteilt werden

Für Benutzer:

* Smartphone
* Eintritt an der Labor durch QR-Coden

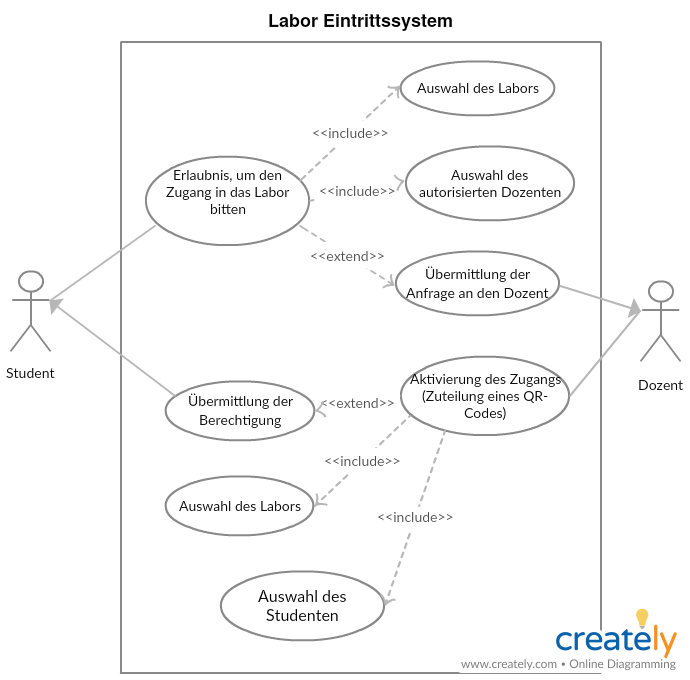
Für System:

* Android Studio und Swift
* Windows und MacOS Betriebssystem
* QR-Codeleser
* Geeignete Türschloss

### Design (Entwicklung) und Implementierung

#### Anwendungsfälle

Es wurde ein Use-Case Diagramm gezeichnet, um die Verwendung des Systems zu verstehen. Es gibt 2 Aktor. Sie sind Dozent und Student. Student möchte in das Labor eintreten und dafür möchte Erlaubnis aus Dozent bekommen. Es gibt auch an, für welches Labor die Zulassung beantragt wird. Der QR-Code des Studenten wird dann aktiviert, wenn er dies genehmigt.



*Abbildung 124.*

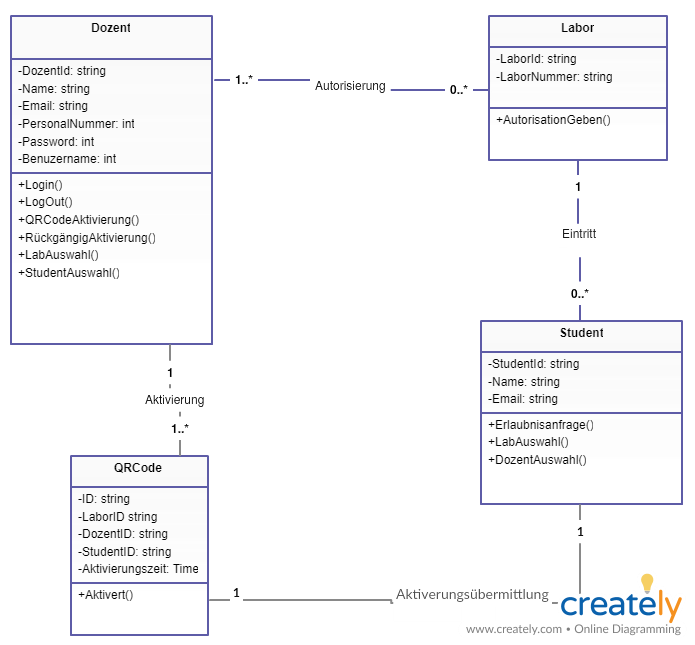
*Abbildung 125.*

*Abbildung 122.*

*Abbildung 115.*

*Abbildung 115.*

#### EER und UML Domänenmodell



*Abbildung 125.*

*Abbildung 126.*

*Abbildung 123.*

*Abbildung 116.*

*Abbildung 116.*

#### Funktionale und nichtfunktionale Anforderungen

Funktionale Anforungen:

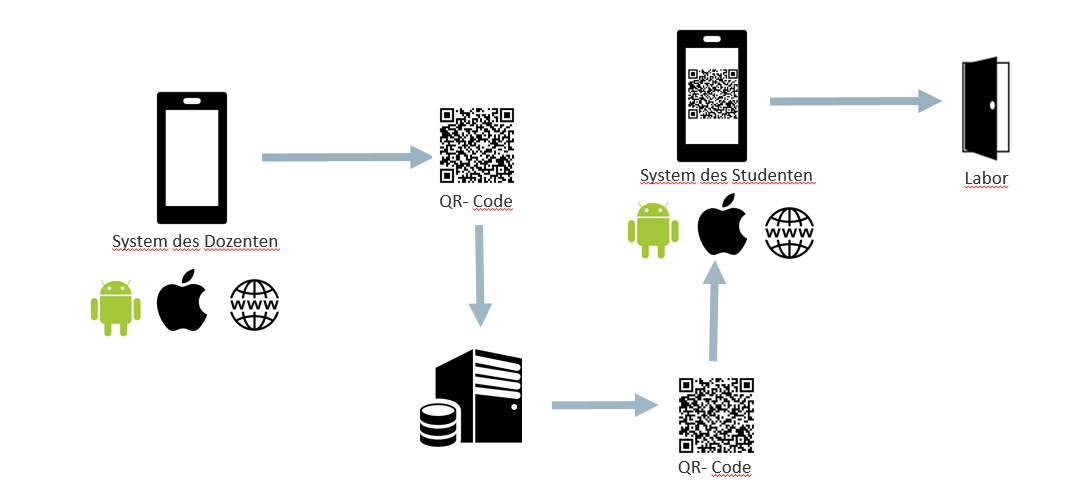
* Das System sollte sowohl für Android als auch für iOS und das Web entwickelt werden.
* Dozenten sollten das System als Admin benutzen.
* Dozent meldet sich ins System mit einem Benutzernamen und einem Passwort ein.
* Es sollte eine separate Seite für den Dozenten geben, um einen Studenten zu autorisieren.
* Auf dieser Seite sollte der Dozent die Möglichkeit haben, den Studenten, das Labor und die Zeitzone auszuwählen.
* Nach dieser Auswahl sollte ein QR-Code mit diesen Informationen erstellt werden.
* Dozent sollte den erstellten QR-Code mit Hilfe eines Buttons senden können.
* Eine Benachrichtigung muss an den Studenten gesendet werden, wenn der qr-Code übertragen wird.
* Studenten können den Code mit Hilfe eines Buttons nach der Eingabe des Systems anzeigen.
* Am unteren Rand der Seite, auf der der Qr-Code geöffnet ist, sollten Informationen zum Labor und zur Nutzungszeit enthalten sein.
* Student kann bekommt den Eintritt, indem er den Code in den QR-Code-Leser einliest.
* Nach Ablauf der Zeit sollte dem Student eine Benachrichtigung gesendet werden.
* Der Schüler verlässt das Labor, indem er den QR-Code erneut einliest und somit die den Weggang bestätigt.
* Nach dem Verlassen bekommt der Dozent eine Benachrichtigung.
* Wenn die Labornutzungszeit abgelaufen ist, geht der Dozent zur Deklaration.
* Der Student sollte in der Lage sein, über eine Seite eine Benachrichtigung an den Dozent zu senden, um die Erlaubnis zu erhalten, das Labor zu betreten.
* Die Benachrichtigung öffnet eine neue Seite und diese Seite enthält die Button zum Annehmen und Ablehnen.
* Dozent kann die Labornutzungszeit auf dieser Seite ändern.

Nicht funktionale Anforderungen:

* Zuverläsisgkeit: Nur autorisierte Benutzer können Zugang anderen Benutzern freischalten.
* Benutzerbarkeit: Alle Studenten und Dozenten können das System einfach benutzen.
* Reaktionszeit: Die Autorisierung sollte den Studenten innerhalb einer Minute erreichen.
* Kompatibilität: Die Website und die App müssen miteinander kompatibel sein.

#### Systemarchitektur

* Das System muss über Android, ios und web korrekt funktioniert.
* Mit Hilfe der geeigneten Server und geeigneten Kommunikation Protokoll System miteinander erreichen.
* QR-Code mit nötigen Informationen soll erstellt werden.
* QR-Codeleser ist nötig vor Labortür.



*Abbildung 126.*

*Abbildung 127.*

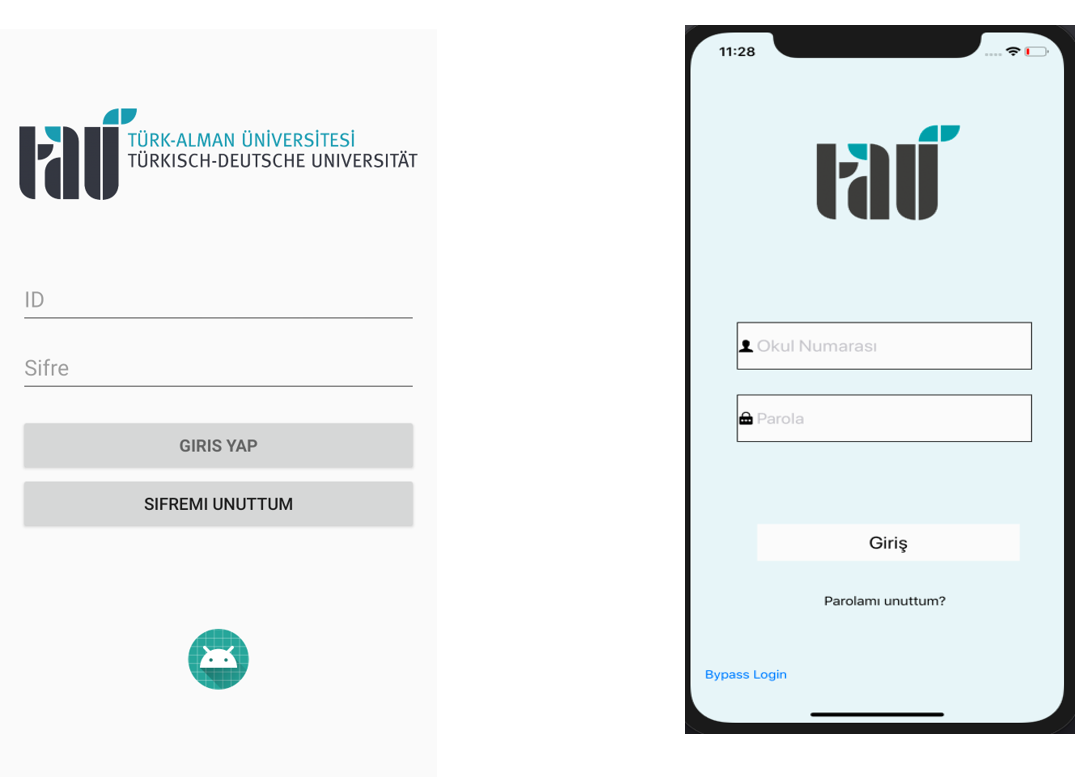
*Abbildung 124.*

*Abbildung 117.*

*Abbildung 117.*

#### Design-Layouts

* Alle Funktionen sind in Anforderungen aufgeführt.



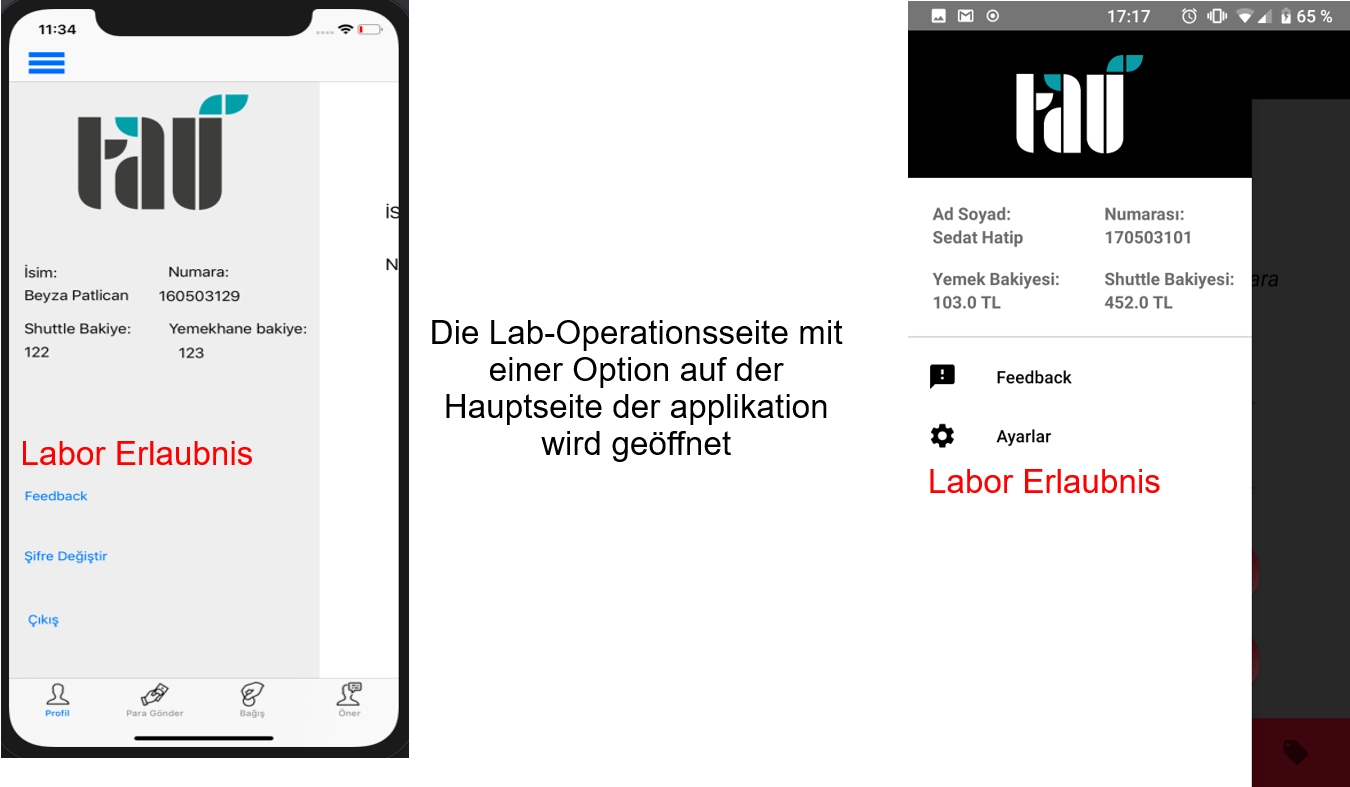
*Abbildung 127.*

*Abbildung 128.*

*Abbildung 125.*

*Abbildung 118.*

*Abbildung 118.*



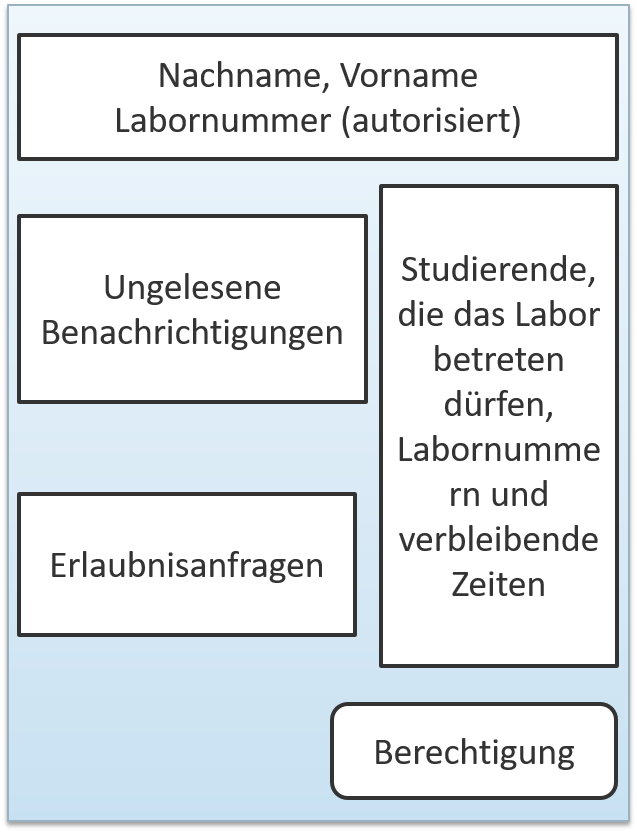
*Abbildung 128.*

*Abbildung 129.*

*Abbildung 126.*

*Abbildung 119.*

*Abbildung 119.*



*Abbildung 129.*

*Abbildung 130.*

*Abbildung 127.*

*Abbildung 120.*

*Abbildung 120.*



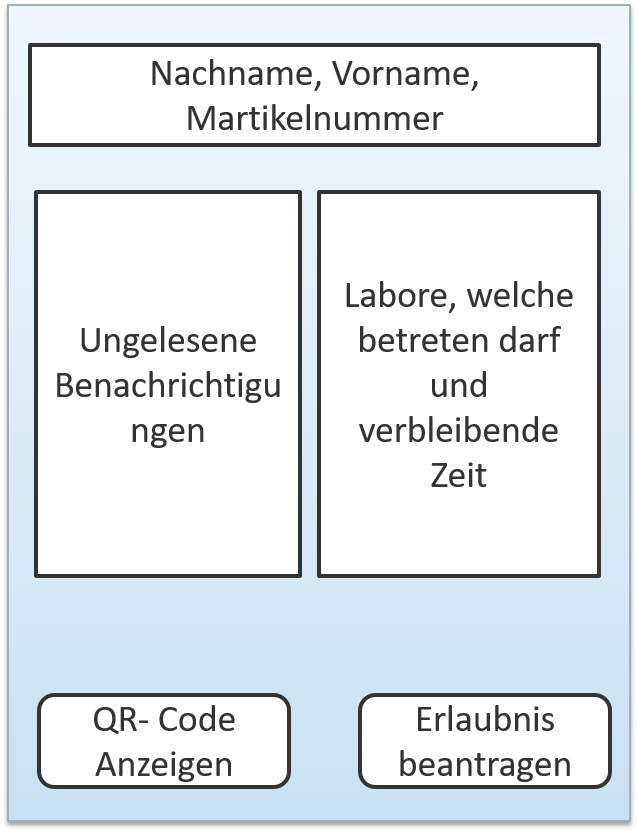
*Abbildung 130.*

*Abbildung 131.*

*Abbildung 128.*

*Abbildung 121.*

*Abbildung 121.*



*Abbildung 131.*

*Abbildung 132.*

*Abbildung 129.*

*Abbildung 122.*

*Abbildung 122.*



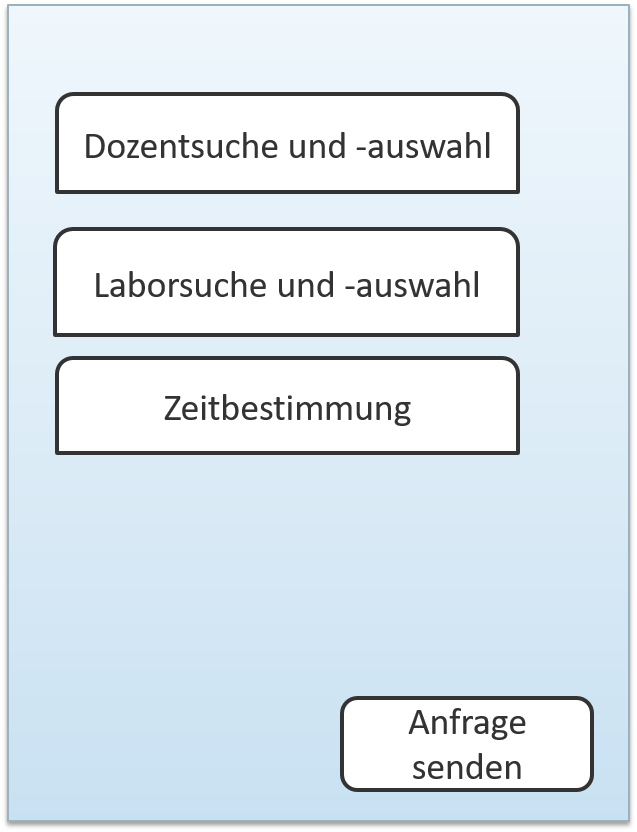
*Abbildung 132.*

*Abbildung 133.*

*Abbildung 130.*

*Abbildung 123.*

*Abbildung 123.*



*Abbildung 133.*

*Abbildung 134.*

*Abbildung 131.*

*Abbildung 124.*

*Abbildung 124.*

### Evaluation

Für die Evaluation des Konzepts wurde eine Umfrage via Google Forms erstellt. 117 Studenten haben die Umfrage beantwortet. Das Ziel von dieser Umfrage war, um zu sehen, ob die Studenten wirklich so ein System brauchen und/oder möchten.



*Abbildung 134.*

*Abbildung 135.*

*Abbildung 132.*

*Abbildung 125.*

*Abbildung 125.*

### Zusammenfassung und Ausblick

Es ist offensichtlich, dass ein solches Konzept die Arbeit von Studenten und Akademikern in der Universität erleichtern wird. Alle Anforderungen, Design und Diagramme zur Entwicklung des Konzepts wurden erläutert. Der nächste Schritt ist die Implementation des Konzepts mit das System.

## Das Ausleihen von Büchern aus der Bibliothek  (Muhammed GÖK)

### Einleitung

İn sich verändernden und sich entwickelnden Welt ändert sich auch die Art und Weise, wie man Informationen erhält. Eine verbesserte und technologischere Bibliothek ist also ein wichtiger Teil dieser Änderungen. Wenn Technologie in etwas involviert ist, ist es immer besser. Die Motivation dieses Konzepts ist es, eine Bibliothek zu schaffen, die für das technologische Zeitalter praktisch und angemessen ist.

#### Problemstellung

Wie das System in der Bibliothek jetzt funktioniert?

* Sie erhalten ein Passwort für studentische Angelegenheiten über Ihre E-Mail-Adresse, die die Universität angegeben hat.
* In der Bibliothek befindet sich ein Gerät, mit dem Sie Bücher erkennen können, die Sie damit scannen.
* Danach geben Sie Ihr Kennwort für dieses System ein und Sie sehen einen Bildschirm, auf dem Sie die Bücher, die Sie scannen, Ihrem Konto hinzufügen können.

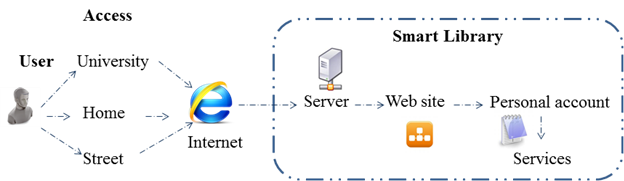
Nach den Recherchen, die von mir gemacht wurden. Die Probleme sind:

* Es gibt eine begrenzte Anzahl von Maschinen in der Bibliothek, daher müssen die Schüler in Massenzeiten eine Weile warten.
* Normalerweise muss, wenn jemand die Bücher nicht rechtzeitig übergibt, eine Gebühr entrichtet werden. Aber gemäß der Information, die ich versammle, folgen viele Menschen diesem Verfahren nicht. Diese Situation ermutigt die Menschen, die ihre Lieferung verschieben.
* Wenn ein Buch von jemandem aus der Bibliothek genommen wird, wird das Buch nicht rechtzeitig geliefert. Die Bibliothek verfügt nicht über ein anständiges System, um die Bücher zu finden, die genommen und nicht zurückgegeben wurden. In der gegenwärtigen Situation fehlen viele Bücher.Aufgrund dessen kann der Student nicht wissen, wann das von ihm gesuchte Buch der Bibliothek übergeben wird.

#### Stand der Technik

**Intelligente Bibliothekslösung:**

* + Es ist ein Bibliotheksmanagementsystem der neuen Generation, das auf der RFID-Technologie (Radio Frequency Identification) basiert. Es wurde auf der ganzen Welt mit rascher Expansion eingesetzt.
  + In China blüht die intelligente Bibliothekslösung mit der Ermutigung der Regierung in intelligenter Bildung. Viele andere Länder und Gebiete haben außerdem intelligente Bibliothekslösungen bereitgestellt, um die Dienste und das Management ihrer Bibliotheken zu verbessern



*Abbildung 135.*

*Abbildung 136.*

*Abbildung 133.*

*Abbildung 126.*

*Abbildung 126.*

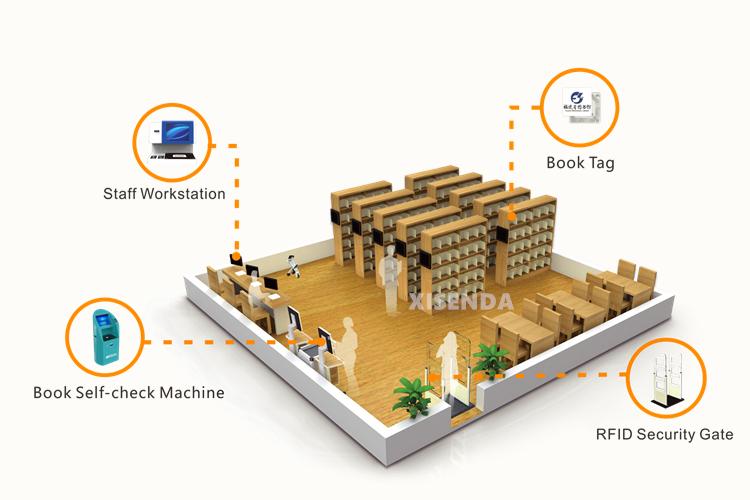
Derzeit gibt es verschiedene intelligente Bibliothekslösungen, und unsere Universität verwendet die

einfachste Version der Minimalversion.

**Vorteilen von Smart-Library**

* + Das System ist eine Selbstfunktion.
  + Automatische Sicherheitserkennung mit Ton- und Lichtalarm.
  + Automatisierte Materialhandhabung mit Rückgabe- und Sortiersystem
  + Materialverfolgung mit OPAC, 3D-Navigation und

dem RFID-Netzwerk.



*Abbildung 136.*

*Abbildung 137.*

*Abbildung 134.*

*Abbildung 127.*

*Abbildung 127.*

#### Zielsetzung

Unser Ziel ist es, eine praktischere Bibliotheksumgebung für Kunden zu schaffen. Im Erfolgsfall können die Kunden die Bücher aufspüren und erfahren, wann sie verfügbar sind, und sie können die Bücher ohne Passwort oder Nummer problemlos ausleihen.

Mehr Bücher in die Bibliothek zu bringen oder eine bessere Lernumgebung zu schaffen, ist nicht unser Ziel.

Wir werden mehr Geräte benötigen, die Barcodes scannen können, und einen Server, um alle Schüler- und Buchinformationen zu speichern.

### Anforderungsanalyse und Konzeption

**Für den Benutzer:**

* Ein Benutzer braucht nur ein Smartphone, das Barcodes scannen kann und natürlich Mitglied bei TDU ist.

**Für das System:**

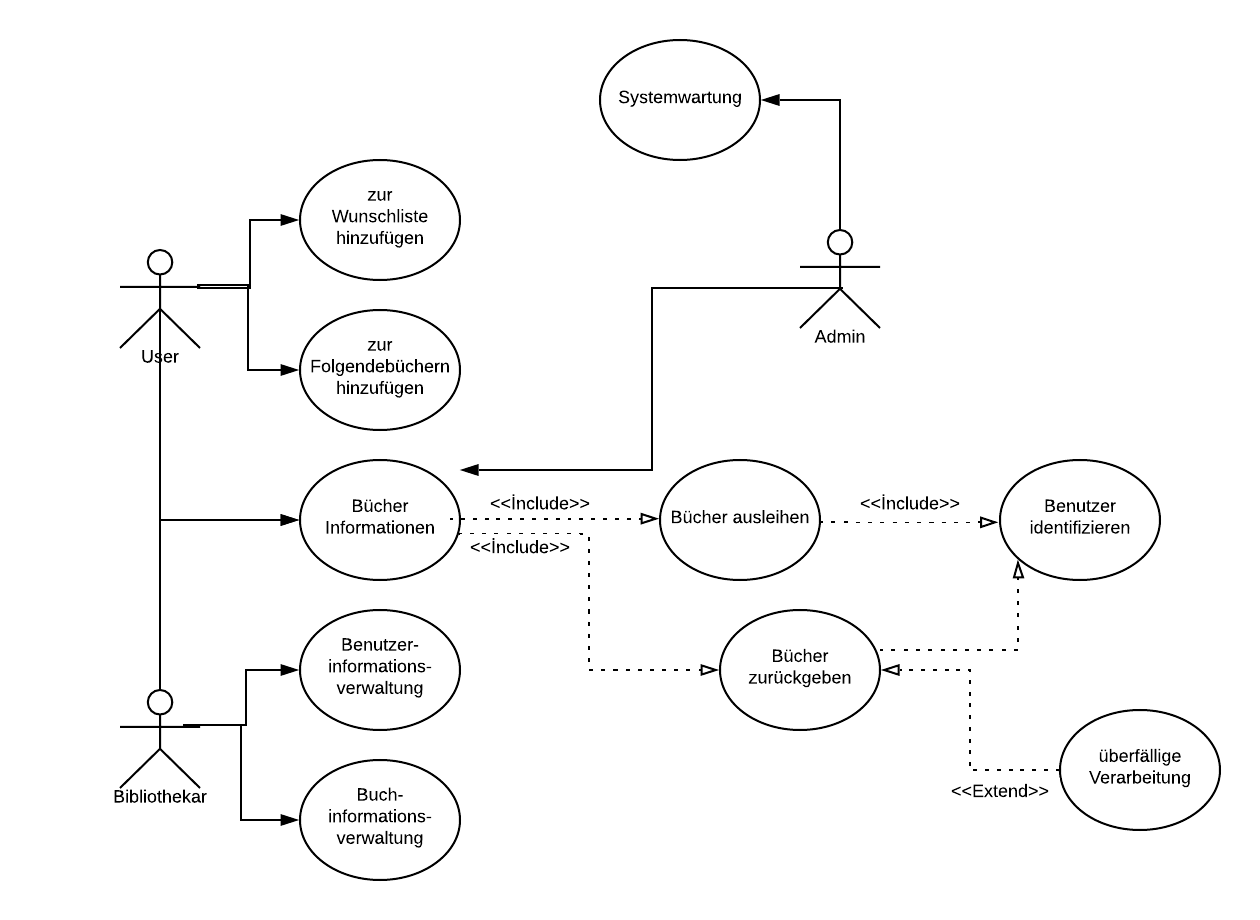
* Für das System eine Datenbank, die alle Informationen über Bücher in der Bibliothek und alle Studenten, Professoren usw. Enthält.
* Android Studio und Swift
* Barcodeleser Geräten.

### Design (Entwicklung) und Implementierung

#### Anwendungsfälle

Es wurde ein Use-Case Diagramm gezeichnet, um die Verwendung des Systems zu verstehen. Es gibt 3 Aktor. Sie sind Benutzer, Bibliothekar und Admin.

Use-Case Diagramm:



*Abbildung 137.*

*Abbildung 138.*

*Abbildung 135.*

*Abbildung 128.*

*Abbildung 128.*

#### EER und UML Domänenmodell



*Abbildung 138.*

*Abbildung 139.*

*Abbildung 136.*

*Abbildung 129.*

*Abbildung 129.*

#### Funktionale und nichtfunktionale Anforderungen

Funktionale Anforderungen:

* Das System sollte sowohl für Android als auch für iOS entwickelt werden.
* Das System ermöglicht dem Benutzer das Anzeigen aller Bücher in der Bibliothek.
* Alle Bücher können mit einem persönlichen Barcode auf einem Smartphone ausgeliehen werden. Dieser Barcode wird von der Bibliothek ausgegeben.
* Ein Buch kann ausgeliehen, einer Wunschliste hinzugefügt und folgenden Büchern hinzugefügt werden.
* Bei den folgenden Büchern handelt es sich um Bücher, die von anderen Benutzern geliehen werden. Mit diesem Benutzer können Sie sehen, wann das Buch zurückgegeben wird. Wenn es zurückgegeben wird, wird eine Benachrichtigung an den Benutzer gesendet.
* Wunschliste ist eine Liste, die vom Benutzer erstellt wird, um Bücher hinzuzufügen, die in der Bibliothek angezeigt werden sollen.

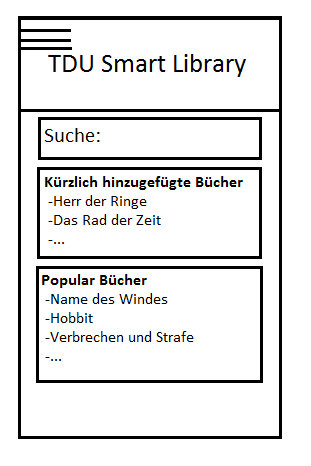
Nicht- funktionale Anforderungen:

* Benutzerfreundlichkeit, gutaussehend, Korrektheit.
* Das System soll dem Anwender schnell antworten.
* Alle Barcodes sind persönlich.

#### Systemarchitektur

* Das System muss über Android, İos korrekt funktioniert.
* Mit Hilfe der geeigneten Server und geeigneten Kommunikation Protokoll System miteinander erreichen.
* Barcode mit Benutzer Informationen soll erstellt werden.
* Barcode wird als ein ID in Bibliothek anwenden

#### Design-Layouts



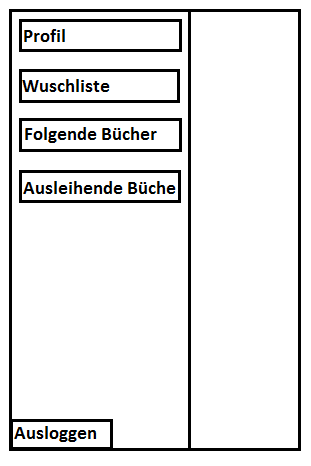
*Abbildung 139.*

*Abbildung 140.*

*Abbildung 137.*

*Abbildung 130.*

*Abbildung 130.*



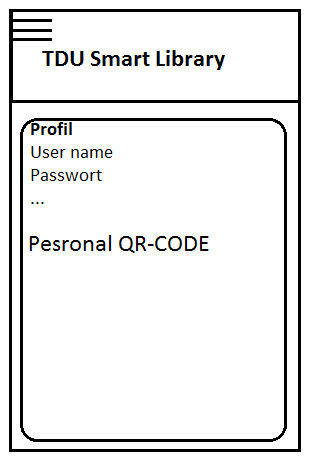
*Abbildung 140.*

*Abbildung 141.*

*Abbildung 138.*

*Abbildung 131.*

*Abbildung 131.*



*Abbildung 141.*

*Abbildung 142.*

*Abbildung 139.*

*Abbildung 132.*

*Abbildung 132.*

### Evaluation

Als Ergebnis wurde es verstanden, dass da so viele Neuigkeit man in der TDU warte aus Implantation gibt. Wenn die İdee von uns realisieret hat, die Leben von Studenten einfacher sein wird.

### Zusammenfassung und Ausblick

Dieses Konzept kann eine Innovation sein im Bereich von Universitätsbibliothek. Da in dem Stand der Technik Anwendungen gibt, ist es attraktiv für die Studenten. Die Prozesse an den Universitäten sind schon zu schwierig. Man kann von einem einfachen Punkt anfangen, die Universitäten zu verbessern im Rahmen der Innovation.

## Konzept 5: TDU Zielführung- und Informationssystem (Yusufhan Yılmaz)

### Einleitung

Der Lehrkörper von TDU hat sehr viele Dozenten und unter anderem sind mehrere deutsche Dozenten. Im Allgemein zwei Dozenten nehmen einen Raum teil. Aber es gibt kein augenfälliger Raum für deutsche Dozenten. Als Studenten, wenn wir ein/e Dozent/in uns für Besprechung umsehen, anstrengen wir uns. Außerdem neue Studenten können nicht Hörsaal und Laboren auffinden. Dieses Problem wird durch eine Applikation durchgeschlagen.

#### Problemstellung

Im Allgemein sind die Studenten in der Suche. Die Frage wird gehört, wo ist dieser Hörsaal oder Klasse? Außerdem Dozenten, die Deutschen im Großen sind, werden durch Studenten umgesehen. Deutsche Dozenten werden mehrere Studenten umgesehen, weil sie kein Raum haben, also sie unbestimmte Raumnummer haben. Neue Studenten suchen allgemein das Fakultätssekretariat oder das Dechanat. Also dieses Problem gilt nicht nur TDU.

#### Stand der Technik

#### Zielsetzung

Für die Problemlösung wird einfache Navigationsapplikation entwickelt. Mit dieser Applikation wird ein/e Student/in gewünschte Platz gehen. Dafür wird eine Zielführung - und Informationssystem aufgebaut. Das System wird funktioniert, dass eine QR Code, die in dem augenfälligen Punkt ist, wird durch Studenten gescannt. QR Coden können neben Aufzuge oder in der Kantine sein. Ein Student wird QR Code scannen, damit die Position des Students kann bestimmt werden. Wenn gewünschte Adresse(Klasse oder Raum) beigetreten wird, wird Zielführung anfangen. Äußerstenfalls wird Informationen abgehandelt.

### Methodische Ermittlung

#### Methoden und Werkzeuge

Man braucht eine Navigationsalgorithmen, um die Absicht mit weniger Zeit zu erreichen. Die Werkzeuge sollen QR-Code und das Handy, das QR-Code scannen kann, sein.

### Anforderungsanalyse und Konzeption

Funktionale Anforderungen:

* Das System soll mit QR Code funktioniert werden.
* Die Applikation kann für alle Menschen verwendet werden. Also Applikation kann auch durch eine Person, die nicht an der TDU studiert, funktioniert werden.

Nicht- funktionale Anforderungen:

* Benutzerfreundlichkeit, gutaussehend, Korrektheit
* Das System soll dem Anwender innerhalb von einer Sekunde antworten.

Für Benutzer:

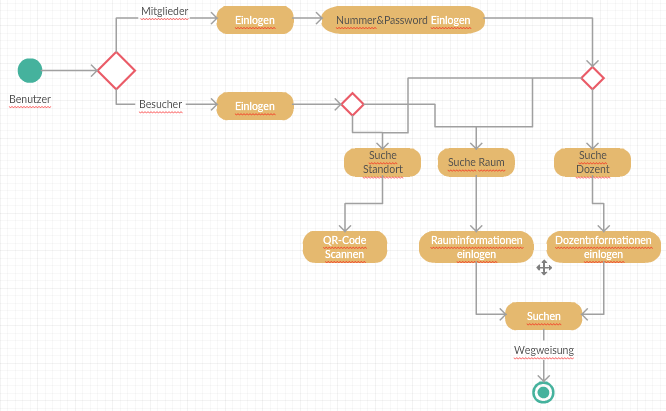
* Smartphone
* QR Coden in dem bestimmten Punkt

Für System:

* Android Studio und Swift
* Windows und MacOS Betriebssystem
* QR-Codeleser

### Design (Entwicklung) und Implementierung

#### Anwendungsfälle



*Abbildung 142.*

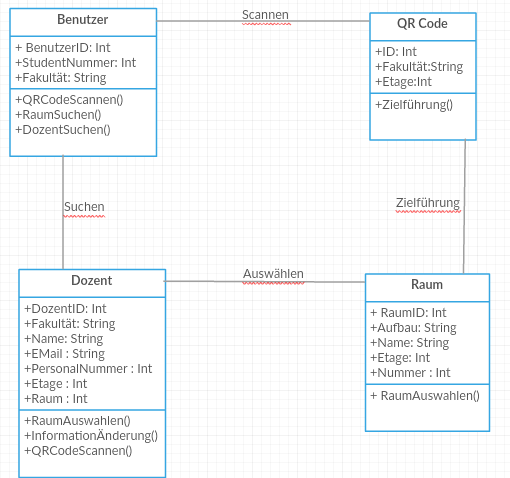
*Abbildung 143.*

*Abbildung 140.*

*Abbildung 133.*

*Abbildung 133.*

#### EER und UML Domänenmodell



*Abbildung 143.*

*Abbildung 144.*

*Abbildung 141.*

*Abbildung 134.*

*Abbildung 134.*

#### Funktionale und nichtfunktionale Anforderungen

Funktionale Anforderungen:

* Das System soll mit QR Code funktioniert werden.
* Die Applikation kann für alle Menschen verwendet werden. Also Applikation kann auch durch eine Person, die nicht an der TDU studiert, funktioniert werden.

Nicht- funktionale Anforderungen:

* Benutzerfreundlichkeit, gutaussehend, Korrektheit
* Das System soll dem Anwender innerhalb von einer Sekunde antworten.

#### Systemarchitektur

* QR-Code mit nötigen Informationen soll erstellt werden.
* Das System muss über Android, Ios und web korrekt funktioniert.
* Database



*Abbildung 144.*

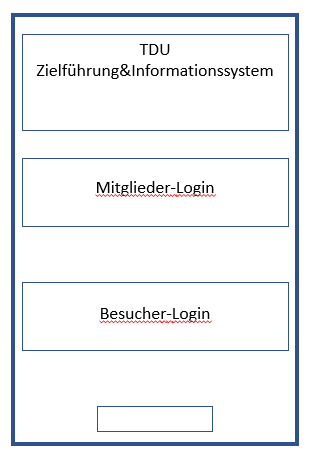
*Abbildung 145.*

*Abbildung 142.*

*Abbildung 135.*

*Abbildung 135.*

#### Design-Layouts



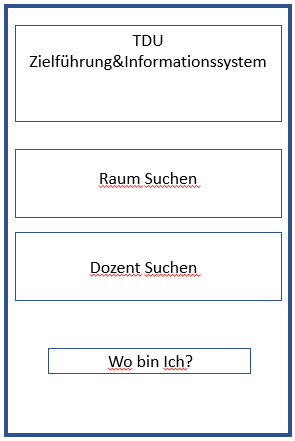
*Abbildung 145.*

*Abbildung 146.*

*Abbildung 143.*

*Abbildung 136.*

*Abbildung 136.*



*Abbildung 146.*

*Abbildung 147.*

*Abbildung 144.*

*Abbildung 137.*

*Abbildung 137.*



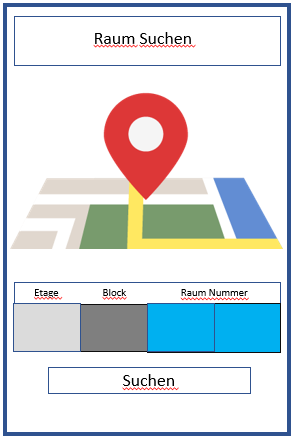
*Abbildung 147.*

*Abbildung 148.*

*Abbildung 145.*

*Abbildung 138.*

*Abbildung 138.*



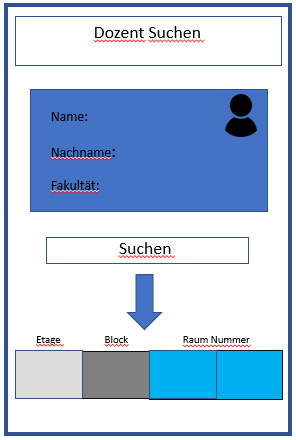
*Abbildung 148.*

*Abbildung 149.*

*Abbildung 146.*

*Abbildung 139.*

*Abbildung 139.*



*Abbildung 149.*

*Abbildung 150.*

*Abbildung 147.*

*Abbildung 140.*

*Abbildung 140.*

### 

**3. Projektabschlussbericht**

Hinweis: Für jeweils alle Arbeitsgruppen!

| **Projektabschlussbericht (Android)** | | |
| --- | --- | --- |
| **Erfolgsmessung** | | |
| **Zielerreichung Leistungen****:**  Das Ergebnis wurde wie angefordert erreicht. | | |
| **Zielerreichung Termine:**  23.04.2019 | | |
| **Zielerreichung Kosten:**  Bei der Erstellung von den Anforderungen wurde kein Geld ausgegeben. | | |
| **Reflexion / Lessons Learned** | | |
| **Team / Zusammenarbeit:**  Es wurde von dem Teamleiter darauf geachtet, dass jeder Mitarbeiter , den gegebenen Auftrag sorgfältig macht.  Bei der Durchführung des Projekts waren alle Arbeiter sehr motiviert. | | |
| **Projektmanagement:**  Das Projekt wurde von dem Teamleiter sehr gut verwaltet. Den der Teamleiter var selbst auch sehr motiviert, wusste sehr gut, wer, welche Aufgabe, bis wann erledigen kann und hat darauf geachtet,dass niemand unter Stress ist.  Der Teamleiter legte Wert auf eine gute Kommunikation, welche zwischen den Teammitgliedern und auch mit anderen Gruppenmitglieder geschah. | | |
| **Sonstige “Learnings”:**  Während des Projekts haben wir uns sehr viel hinzugefügt.   * Das wichtigste, wir haben gelernt zusammenzuarbeiten, auch wenn wir nicht in der gleichen Umgebung sind * Gelernt wie man einen Mobilen App herstellt * Ein neues Programm, mit dem effizientesten Weg, so schnell wie möglich, zu lernen | | |
| **Planung Nachprojektphase / Restaufgaben** | | |
| **Was?** | **Wer?** | **Bis wann?** |
|  |  |  |
| **Projektabnahme** | | |
| ☐ Das Projekt abgenommen.  ☐ Das Projekt wird unter der Einhaltung folgender Auflagen abgenommen:      ……………………………………………..  <Datum>, Unterschrift Auftraggeber | | |

Tabelle 26: **Projektabschlussbericht (Android)**

| **Projektabschlussbericht (Backend)** | | |
| --- | --- | --- |
| **Erfolgsmessung** | | |
| **Zielerreichung Leistungen****:**  Erreicht | | |
| **Zielerreichung Termine:**  Erreicht 15.05.2019 | | |
| **Zielerreichung Kosten:**  Monatlich 20$ | | |
| **Reflexion / Lessons Learned** | | |
| **Team / Zusammenarbeit:**  Backend Team / AP1 und AP12 | | |
| **Projektmanagement:**   * Projektplanung * Kommunikation * Organisation * Aufgabenverteilung * Kostenplanung * Terminplanung | | |
| **Sonstige “Learnings”:**  Datenbank zu erstellen / Die benötigte Funktionen mit JAVA Spring erstellt (Tomcat, Postman) / Verknüpfung zwischen Datenbank und Server(JPA, SSL) / Übertragung von Datenbank und Spring.jar Dateien | | |
| **Planung Nachprojektphase / Restaufgaben** | | |
| **Was?** | **Wer?** | **Bis wann?** |
| Java Spring / Implementierung | Ömer Cem Turan | 12.04.2019 |
| Übertragung von Datenbank und Java Spring.jar Dateien | Ayşe Rabia Özbek | 12.04.2019 |
| Verknüpfung zwischen Datenbank und Server | Onur Burak Kılıç | 12.04.2019 |
| Datenbank erstellen | Onur Mert Doğan | 12.04.2019 |
| Erstellen für QR-Code in den Java Script | Aybüke Bayramiç | 12.04.2019 |
| Datenbank Modellierung (ER und UML Diagramm) | Kartal Can Çelikkanat | 12.04.2019 |
| **Projektabnahme** | | |
| ☐ Das Projekt abgenommen.  ☐ Das Projekt wird unter der Einhaltung folgender Auflagen abgenommen:      ……………………………………………..  <Datum>, Unterschrift Auftraggeber | | |

## Tabelle 27: Projektabschlussbericht (Backend)

| **Projektabschlussbericht (IOS)** | | |
| --- | --- | --- |
| **Erfolgsmessung** | | |
| **Zielerreichung Leistungen:**  Das Ergebnis wurde wie angefordert erreicht. | | |
| **Zielerreichung Termine:**  23.04.2019 | | |
| **Zielerreichung Kosten:**  Bei der Erstellung von den Anforderungen wurde kein Geld ausgegeben. | | |
| **Reflexion / Lessons Learned** | | |
| **Team / Zusammenarbeit:**  Es wurde von dem Teamleiter darauf geachtet, dass jeder Mitarbeiter , den gegebenen Auftrag sorgfältig macht.  Bei der Durchführung des Projekts waren alle Arbeiter sehr motiviert. | | |
| **Projektmanagement:**  Das Projekt wurde von dem Teamleiter sehr gut verwaltet. Den der Teamleiter var selbst auch sehr motiviert, wusste sehr gut, wer, welche Aufgabe, bis wann erledigen kann und hat darauf geachtet,dass niemand unter Stress ist.  Der Teamleiter legte Wert auf eine gute Kommunikation, welche zwischen den Teammitgliedern und auch mit anderen Gruppenmitglieder geschah. | | |
| **Sonstige “Learnings”:**  Während des Projekts haben wir uns sehr viel hinzugefügt.  Das wichtigste, wir haben gelernt zusammenzuarbeiten, auch wenn wir nicht in der gleichen Umgebung sind  Gelernt wie man einen Mobilen App herstellt  Ein neues Programm, mit dem effizientesten Weg, so schnell wie möglich, zu lernen | | |
| **Planung Nachprojektphase / Restaufgaben** | | |
| **Was?** | **Wer?** | **Bis wann?** |
|  |  |  |
| **Projektabnahme** | | |
| ☐ Das Projekt abgenommen.  ☐ Das Projekt wird unter der Einhaltung folgender Auflagen abgenommen:      ……………………………………………..  <Datum>, Unterschrift Auftraggeber | | |

Tabelle 28: **Projektabschlussbericht (IOS)**

| **Projektabschlussbericht (Webseite AP2-AP8)** | | |
| --- | --- | --- |
| **Erfolgsmessung** | | |
| **Zielerreichung Leistungen:**  Die Head-Admin Webseite, die das Hinzufügen, Löschen und Bearbeiten Funktionen von Admin durchführen kann, wurde erfolgreich entwickelt. | | |
| **Zielerreichung Termine:**  Alle Termine wurden pünktlich und problemlos durchgeführt. Geplante aufgaben wurden ausgeführt. Projekt wird in der geplante Zeit fertiggestellt. | | |
| **Zielerreichung Kosten:**  Projekt wird kostenlos bereitgestellt. | | |
| **Reflexion / Lessons Learned** | | |
| **Team / Zusammenarbeit:**  Die Webseite wurde mit Arbeitspaket 6 ( Mensa-Admin) und Arbeitspaket 7 (Shuttle-Admin) zusammen fertiggestellt.  Unsere Head-Admin Team hätte 6 Mitarbeitern; Uğur Güneş, Beste Şanlı, Minel Küpçü, Ali Egin, İsmail Çağlar Bayram, Kaan Göktaş. | | |
| **Projektmanagement:**  Dr.-Ing. Zeynep Tuncer | | |
| **Sonstige “Learnings”:**  Erstellung einer Webseite mit Verwendung von PHP, HTML und CSS. | | |
| **Planung Nachprojektphase / Restaufgaben** | | |
| **Was?** | **Wer?** | **Bis wann?** |
|  |  |  |
| **Projektabnahme** | | |
| ☐ Das Projekt abgenommen.  ☐ Das Projekt wird unter der Einhaltung folgender Auflagen abgenommen:  ●        ……………………………………………..  <Datum>, Unterschrift Auftraggeber | | |

Tabelle 29: **Projektabschlussbericht (Webseite AP2-AP8)**

**Anhang**

**Glossar**

**Literaturverzeichnis**

Hinweis: Bitte Alphabetisch ordnen!

Beispiel:

1. **(Tuncer 2017, S.5.): Tuncer, Z.** (2017): "Usability in der Landtechnik - Empirische Studien zur Gestaltung von Benutzeroberflächen eines Maschinendisplays am Beispiel eines Feldhäckslers", Shaker Verlag, ISBN: 978-3-8440-5263-3

**Besprechungsprotokolle**

* Datum
* Ort
* Beginn
* Dauer
* Anwesenheitsliste
* Themen
* ToDo Liste

1. Beispiel Versionierungs-Systematik:   
   Arbeitsversionen 0.1, 0.2 etc. | Hauptversionen 1.0, 2.0 etc. [↑](#footnote-ref-0)
2. Eine Rolle besteht immer aus Aufgaben, Kompetenzen und Verantwortung (AKV-Prinzip) [↑](#footnote-ref-1)
3. Mögliche Strukturierungsvarianten: phasenorientiert, funktionsorientiert, objektorientiert, gemischtorientiert  
   Mögliche Tools zur Darstellung: MindMapping (z.B. MindManager, FreeMind etc.) bzw. MS PowerPoint [↑](#footnote-ref-2)
4. z.B. MS Project, MS Excel , Gantt-Project (Open-Source-Tool) etc. [↑](#footnote-ref-3)