
Software Anforderungsspezifikation

für

ICWHO – The Official ICU App



Version 2.1

Autor: Florian Ruosch

Juli 2016

Inhalt

1. Einleitung.....	1
1.1 Zweck	1
1.2 Zielpublikum	1
1.3 Produktumfang	1
1.4 Referenzen	1
2. Beschreibung	2
2.1 Produkt-Ausblick.....	2
2.2 Funktionalitäten.....	2
2.3 Benutzerklassen	2
2.4 Ausführungsumgebung.....	2
2.5 Einschränkungen in Design und Implementation.....	2
2.6 Benutzer Dokumentation.....	3
2.7 Annahmen und Abhängigkeiten	3
2.8 Benutzerschnittstellen	3
2.9 Hardwareschnittstellen	3
2.10 Softwareschnittstellen	3
2.11 Kommunikationsschnittstellen	3
3. System Features.....	4
3.1 ICU Events	4
3.2 ICU News	4
3.3 Gebäudeplan	5
3.4 Menüplan.....	5
3.5 Fahrplan.....	6
3.6 Fotos von Events	6
3.7 About ICU	7
4. Nichtfunktionale Anforderungen	9
4.1 Performance.....	9
4.2 Sicherheit.....	9
4.3 Qualität	9
4.4 Geschäftsregeln	9
5. Andere Anforderungen	9
6. Appendix A: Glossar.....	10

Versionen

Name	Datum	Änderungen	Version
Florian Ruosch	31.05.16	Erste veröffentlichte Version	1.0
Florian Ruosch	13.07.16	Neue Anforderungen hinzugefügt	2.0
Florian Ruosch	20.07.16	Anforderungen finalisiert	2.1

1. Einleitung

1.1 Zweck

Dieses Dokument bietet einen Überblick über die Anforderungen an die im Rahmen des Software-Projekt entwickelte Android-App «ICWHO – The Official ICU App». Auftraggeber stellt der ICU dar: Fachverein Informatik Universität Zürich (<https://www.icu.uzh.ch/>).

1.2 Zielpublikum

Verfasst wurde dieses Dokument, um den Umfang der App dem betreuenden Professor Lorenz Hilty aufzuzeigen. Auch dient es aber zur Dokumentation der Ergebnisse von Gesprächen zwischen Auftraggeber ICU (vertreten durch William Martini) und dem Entwickler (Florian Ruosch).

1.3 Produktumfang

Die App soll die Interaktion zwischen dem Studentenverein sowie den Studenten vereinfachen und den Umweg über die ICU-Homepage überflüssig machen.

1.4 Referenzen

Keine

2. Beschreibung

2.1 Produkt-Ausblick

Die App wird entwickelt im Rahmen des Software-Projekts von Florian Ruosch an der Universität Zürich im Bachelor-Studium der Informatik. Die Spezifikation und Umsetzung der Funktionalitäten erfolgt gemeinsam mit dem Auftraggeber (ICU).

2.2 Funktionalitäten

Die folgende Aufzählung ist weder aus- noch abschliessend und stetigem Wandel unterworfen, da während der Entwicklung auch noch Ideen aufgenommen werden sowie Umsetzbarkeits-Studien durchgeführt werden müssen.

Die App soll den Studierenden das Leben vereinfachen mit dem Anzeigen von Gebäude-, Menü- und Fahrplänen, Informationen über den ICU bieten sowie seine Events promoten.

2.3 Benutzerklassen

Die identifizierten Benutzerklassen lauten wie folgt:

- Administratoren
Der Entwickler sowie der Vorstand sollen erweiterte Möglichkeiten bezüglich Verwaltung der Daten haben.
- Studenten
 - o ICU-Mitglieder
Den ICU-Mitgliedern stehen alle Funktionalitäten zur Verfügung, insbesondere in Bezug auf Events.
 - o Andere
Da andere Studenten keinen ICU-Accounts besitzen, sind verschiedene Funktionen nicht zugänglich
- «Erstsemestler»
Studenten im ersten Semester sollen einfachen Zugriff auf nützliche Informationen erhalten, insbesondere Pläne.

2.4 Ausführungsumgebung

Die App soll kompatibel sein mit dem Handy-Betriebssystem Android, ab der Version 4.1.X, sprich API v16. Auf der Serverseite wird je nach Bedarf eine Heroku-Instanz zum Einsatz kommen sowie auch (falls möglich) ein Teil auf den Servern vom ICU laufen.

2.5 Einschränkungen in Design und Implementation

Bei der Implementation muss Rücksicht genommen werden auf die verschiedenen Android Versionen und APIs. Besondere Vorsicht ist bei der Kommunikation mit dem ICU-Server bezüglich Login-Daten geboten.

2.6 Benutzer Dokumentation

Die App soll soweit als möglich selbsterklärend sein, sodass keine Dokumentation nötig sein wird. Wo nötig, wird dies durch Anzeigen in der App abgedeckt.

2.7 Annahmen und Abhängigkeiten

Abhängig ist die Entwicklung der App vor allem von den Rückmeldungen aus dem Vorstand des ICU insbesondere des technischen Verantwortlichen (Benjamin Solenthaler).

Externe Schnittstellen-Anforderungen

2.8 Benutzerschnittstellen

Die App soll ein möglichst einfaches Interface haben. Die genaue Erarbeitung dessen und Erstellung zugehöriger Mockups geschehen in der ersten Phase der Entwicklung.

2.9 Hardwareschnittstellen

Keine

2.10 Softwareschnittstellen

Die App soll mit dem bereits vorhandenen Android-Standard-Kalender kommunizieren können, um die Einträge für Events zu importieren. In der API von Android sollten diese Befehle bereits vorhanden sein.

2.11 Kommunikationsschnittstellen

Die Kommunikation mit den Server-Teilen soll über eine REST-Schnittstelle mittels JSON-Objekten geschehen.

3. System Features

Wie bereits unter 2.2 erwähnt, sind die Funktionalitäten der App noch nicht endgültig bestimmt. Sollten während der Umsetzung neue, wichtigere hinzukommen, so werden diese berücksichtigt und bei unüberwindbaren Hürden auf andere verzichtet.

Zur Bewertung von Priorität, Nutzen, Aufwand und Risiko wurde folgende Skala verwendet: 1 (niedrig) bis 10 (hoch).

3.1 ICU Events

3.1.1 Beschreibung

Anzeigen der geplanten Events in einer Liste, Login und damit Möglichkeit zur Anmeldung zu Events, Export eines Events in einen Kalendereintrag

Priorität:	10
Nutzen:	10
Aufwand:	10
Risiko:	9

3.1.2 Verhalten

Wenn der Benutzer auf «Events» klickt, sollen die bevorstehenden Events angezeigt werden. Zusätzlich gibt es auch einen Login-Button, wo man seine Daten eingeben kann und dadurch die Anmeldung zu Events ermöglicht wird. Für jedes Event gibt es einen Button, wodurch ein Kalendereintrag im Android-Standard-Kalender erstellt wird als Erinnerung.

Für jeden Event wird – falls in den Einstellungen so gewünscht – am Durchführungstag eine Erinnerung per Pushnachricht gesendet.

3.1.3 Funktionale Anforderungen

REQ-1:	Anzeigen der Event-Liste (inkl. Abrufen vom Server)
REQ-2:	Login-Möglichkeit
REQ-3:	Anmelde-Möglichkeit
REQ-4:	Export des Events in einen Kalendereintrag
REQ-5:	Pushnachricht als Erinnerung

3.2 ICU News

3.2.1 Beschreibung

Anzeigen von durch die Administratoren veröffentlichte News

Priorität:	8
Nutzen:	8
Aufwand:	7
Risiko:	7

3.2.2 Verhalten

Beim Klicken auf «News» wird eine chronologisch absteigend geordnete Liste mit allen veröffentlichten News angezeigt.
Falls in den Einstellungen gewünscht, werden diese auch als Pushnachrichten angezeigt.

3.2.3 Funktionale Anforderungen

REQ-1: Anzeigen der News (inkl. Abrufen vom Server)
REQ-2: Pushnachrichten mit der News schicken

3.3 Gebäudeplan

3.3.1 Beschreibung

Anzeigen eines Gebäudeplans der Binzmühlestrasse

Priorität: 5
Nutzen: 4
Aufwand: 3
Risiko: 2

3.3.2 Verhalten

Beim Klicken auf «Gebäudeplan» wird jeweils ein Stockwerk des Gebäude Binzmühlestrasse angezeigt. Weitere Buttons machen es möglich, andere Stockwerke anzuzeigen.
Wie dies genau aussieht ist noch zu bestimmen.

3.3.3 Funktionale Anforderungen

REQ-1: Anzeigen eines Stockwerkplans
REQ-2: Möglichkeit zum Wechseln des angezeigten Stockwerks

3.4 Menüplan

3.4.1 Beschreibung

Anzeigen des Menüplans der Mensa in der Binzmühlestrasse

Priorität: 4
Nutzen: 3
Aufwand: 6
Risiko: 3

3.4.2 Verhalten

Beim Klicken auf «Essensplan» wird die Wochenübersicht mit den Menüs der Mensa an der Binzmühlestrasse angezeigt.

3.4.3 Funktionale Anforderungen

- REQ-1: Anzeigen des Menüplans
- REQ-2: Abrufen des Menüplans (woher ist noch zu klären)

3.5 ~~Fahrplan~~

3.5.1 ~~Beschreibung~~

- ~~_____ Anzeigen der nächsten Verbindungen zwischen BIN, Irchel und Zentrum~~
- ~~_____~~
- ~~_____ Priorität: 4~~
- ~~_____ Nutzen: 3~~
- ~~_____ Aufwand: 6~~
- ~~_____ Risiko: 4~~

3.5.2 ~~Verhalten~~

- ~~_____ Nach dem Klicken auf «Verbindungen» kann Abfahrts- und Zielort ausgewählt werden sowie die Ankunfts- oder Abfahrtszeit.~~

3.5.3 ~~Funktionale Anforderungen~~

- ~~_____ REQ-1: Anzeigen der nächsten Verbindungen~~
- ~~_____ REQ-2: Abrufen der nächsten Verbindungen gemäss Kriterien vom Server~~

3.6 ~~Fotos von Events~~

3.6.1 ~~Beschreibung~~

- ~~_____ Möglichkeit zum Hochladen von Fotos nach einem Event~~
- ~~_____~~
- ~~_____ Priorität: 4~~
- ~~_____ Nutzen: 6~~
- ~~_____ Aufwand: 7~~
- ~~_____ Risiko: 6~~

3.6.2 ~~Verhalten~~

- ~~_____ Nach dem Klicken auf «Foto hochladen» kann ein Foto sowie ein zugehöriges Event ausgewählt werden, welches dann auf den Server geladen wird und nach der Freischaltung durch einen Administrator auch auf der Homepage angezeigt wird.~~

3.6.3 ~~Funktionale Anforderungen~~

- ~~_____ REQ-1: Auswählen eines Fotos~~
- ~~_____ REQ-2: Hochladen eines Fotos~~

3.7 About ICU

3.7.1 Beschreibung

Anzeigen von Informationen zum ICU

Priorität: 7
Nutzen: 6
Aufwand: 2
Risiko: 4

3.7.2 Verhalten

Nach dem Klicken auf «About ICU» werden Informationen zum Fachverein angezeigt: Mitglieder des Vorstands, Ort des Büros, Kontaktmöglichkeiten

3.7.3 Funktionale Anforderungen

REQ-1: Anzeigen der Informationen
REQ-2: Einfache Bearbeitungsmöglichkeit durch Administratoren

~~3.8 «Tech News»~~

~~3.8.1 Beschreibung~~

~~—— Anzeigen eines noch zu bestimmenden News Feeds mit IT-News~~

~~—— Priorität: 3
—— Nutzen: 5
—— Aufwand: 5
—— Risiko: 4~~

~~3.8.2 Verhalten~~

~~—— Nach dem Klick auf Tech News werden die neusten Nachrichten chronologisch sortiert angezeigt~~

~~3.8.3 Funktionale Anforderungen~~

~~—— REQ-1: Abrufen und Anzeigen der News~~

3.9 Tür Öffner

3.9.1 Beschreibung

Möglichkeit zum Öffnen der ICU-Lounge-Tür

Priorität: 6
Nutzen: 8
Aufwand: 6
Risiko: 7

3.9.2 Verhalten

Verstecktes Feature

3.9.3 Funktionale Anforderungen

REQ-1: Weiterleitung auf die entsprechende Website

3.10 XKCD Comic of the Day

3.10.1 Beschreibung

Comic des Tages von der bekannten Website xkcd.com

Priorität: 2
Nutzen: 2
Aufwand: 4
Risiko: 2

3.10.2 Verhalten

Nach dem Klicken auf «XKCD CotD» wird der jeweils neuste Comic von der Website xkcd.com angezeigt.

3.10.3 Funktionale Anforderungen

REQ-1: Abrufen und Anzeigen des Comics

~~3.11 Feedbacks~~

~~3.11.1 Beschreibung~~

~~_____ Möglichkeit zum Abgeben von Feedbacks zu Events~~
~~_____~~
~~_____ Priorität: 6~~
~~_____ Nutzen: 8~~
~~_____ Aufwand: 7~~
~~_____ Risiko: 7~~

~~3.11.2 Verhalten~~

~~_____ tbd~~

~~3.11.3 Funktionale Anforderungen~~

~~_____ REQ-1: tbd~~

4. Nichtfunktionale Anforderungen

4.1 Performance

Die App soll nicht blockieren und innert nützlicher Frist Rückmeldung geben: sprich maximal 2 Sekunden Verzögerung. Bei Server-Anfragen ist dies auf 5 Sekunden festgelegt.

4.2 Sicherheit

Die Logindaten der Benutzer sollen auf sicherem Weg gesendet werden, der noch zu bestimmen ist. Das Speichern dieser auf dem Gerät soll auch verschlüsselt geschehen.

4.3 Qualität

Die App soll möglichst wenig Fehler aufweisen und nicht abstürzen. Festgelegt wird dies in der separaten Testvorschrift gegen Ende des Projekts.

Der Code soll auch wiederverwendbar und erweiterbar sein. Auf einen modularen und klaren Aufbau ist daher grosser Wert zu legen.

Die App soll möglichst einfach pflegbar sein, damit im Betrieb kein grosser zusätzlicher Aufwand entsteht.

4.4 Geschäftsregeln

Keine

5. Andere Anforderungen

Keine

6. Appendix A: Glossar

ICU	Fachverein Informatik Universität Zürich
API	Application Program Interface