

reSign

Pflichtenheft



Projektbezeichnung	reSign
Projektleiter	Benjamin Imsirovic
Erstellt am	13.10.2021
Letzte Änderung am	04.11.2021
Status	in Bearbeitung
Aktuelle Version	1.0

Änderungsverlauf

Nr.	Datum	Version	Geänderte Kapitel	Art der Änderung	Autor
1	13.10.2021	1.0	Alle	Erstellung	Simon RS
2	29.10.2021	1.0	Ausgangssituation, Team	Bearbeitung	Simon RS
3	02.11.2021	1.0	Konzept	Erstellung	Simon RS
4	04.11.2021	1.0	Abnahmekriterien, Mengengerüst, Administratives	Erstellung	Simon RS, Arsham E, Maxi S, Benjamin I
5	06.11.2021	1.0	Abnahmekriterien <ul style="list-style-type: none"> - Musskriterien - Wunschkriterien Anforderungen <ul style="list-style-type: none"> - Nicht Funktionale Anforderungen Glossar Konzept <ul style="list-style-type: none"> - Benutzeroberfläche Verwendete Technologien <ul style="list-style-type: none"> - Entwicklungsumgebungen - Betriebssysteme Qualitätsbestimmung	Erstellung & Bearbeitung	Arsham E
6	07.11.2021	1.0	Benutzeroberflächen, Verwendete Technologien, Anforderungen, Qualitätsbestimmungen	Bearbeitung	Simon RS, Maxi S

Inhalt

1	Allgemeines	4
1.1	Ausgangssituation	4
1.2	Team.....	4
2	Konzept	5
2.1	Use Cases	5
2.2	Systemarchitektur	6
2.3	Benutzeroberfläche	7
3	Verwendete Technologien.....	9
3.1	Programmiersprachen	9
3.2	Hardware.....	9
3.3	Entwicklungsumgebungen	9
3.4	Betriebssysteme	9
4	Anforderungen.....	9
4.1	Funktionale Anforderungen.....	9
4.2	Nichtfunktionale Anforderungen	9
5	Abnahmekriterien.....	10
5.1	Musskriterien.....	10
5.2	Wunschkriterien.....	10
6	Qualitätsbestimmung	10
7	Glossar	10

1 Allgemeines

Die folgenden Informationen sind auf die HTL Leonding bezogen und können daher nicht unbedingt auf andere Schulen und Institutionen angewendet werden.

1.1 Ausgangssituation

Jede Klasse hat ein Türschild. Zurzeit enthalten diese lediglich die Raumnummer, die Klassenbezeichnung und den Klassenvorstand. Die Informationen auf diesem Türschild werden selten geändert, da jede Änderung einen großen Aufwand bedeutet. Aus diesem Grund werden auch keine zusätzlichen Informationen angezeigt. Gut wäre es, wenn man mit einfachen Mitteln zusätzliche Informationen auf einem Touch-Display zur Anzeige bringen könnte, und damit die alten Türschilder digitalisiert.

1.2 Team

Rolle	Name	Telefon	E-Mail
Projektleiter	Benjamin Imsirovic	+43 664 4178523	benjamin.imsirovic@hotmail.com
Member	Arsham Edalatkhah	+43 676 5878382	arsham.edalatkhah@gmail.com
Member	Simon Rausch-Schott	+43 676 6693393	simon.rausch-schott@24speed.at
Member	Maximilian Siegl	+43 677 62154815	siegl.maxi@gmail.com

2 Konzept

2.1 Use Cases

- **Rauminformationen abrufen**
Die wichtigsten Informationen über den Raum werden auf der „Startseite“ angezeigt und können nach einmaligem (täglich) Beschaffen der anzuzeigenden Daten ohne Interaktion mit der Hardware abgelesen werden.
- **Stundenplan abrufen**
Um den Stundenplan ablesen zu können, muss auf die dafür vorhandene Schaltfläche tippen, um die Ansicht zu wechseln. Die angezeigten Daten werden vom Server angefragt.
- **Als anwesend melden/stempeln**
Um sich als anwesend zu melden, muss auch die Ansicht des Displays gewechselt werden. Auf der dafür vorgesehenen Seite wird dann ein QR-Code für die jeweilige Klasse und den jeweiligen Tag angezeigt. Mit diesem Code und dem Schulaccount kann dann eingestempelt werden.
- **Als abwesend melden/stempeln**
Um sich am Ende des Tages wieder auszustempeln, gilt dieselbe Vorgehensweise wie beim Einstempeln.

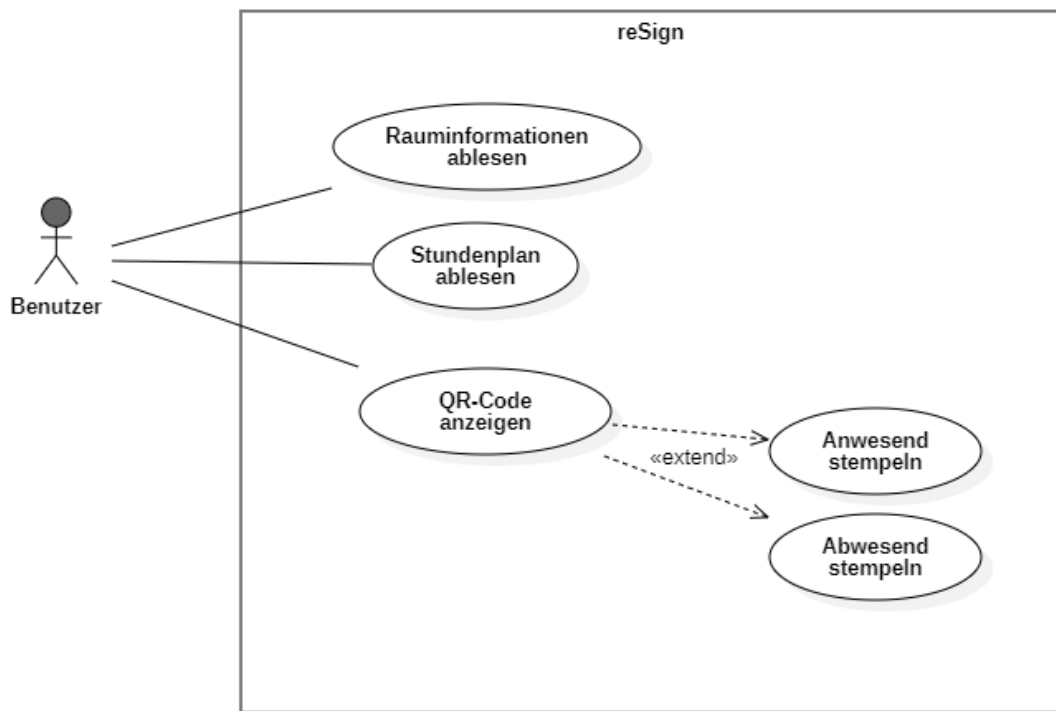


Abbildung 1: UML-Diagramm der Use-Cases

2.2 Systemarchitektur

In der folgenden Grafik wird grob auf die Systemarchitektur eingegangen. Die wichtigsten Informationen wie mögliche bzw. verwendete Sprachen und Betriebssysteme sind darin vorhanden.

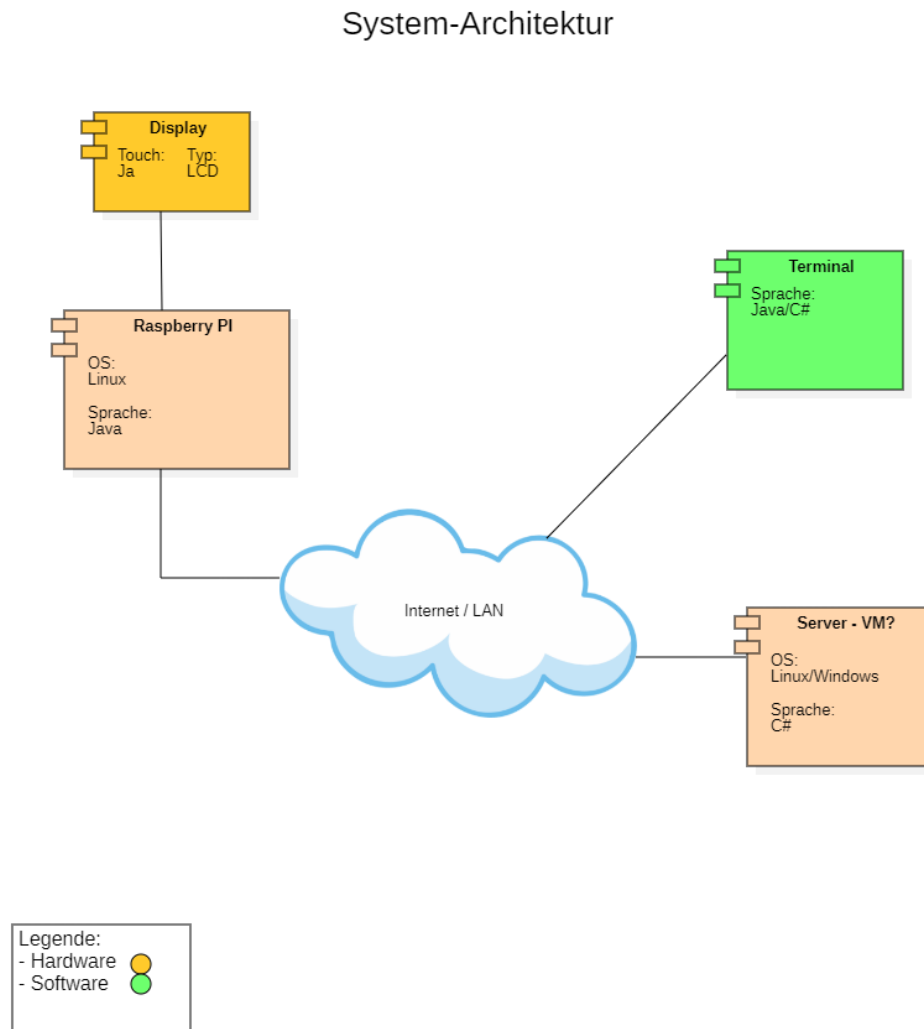


Abbildung 2: Visualisierung der Systemarchitektur

2.3 Benutzeroberfläche

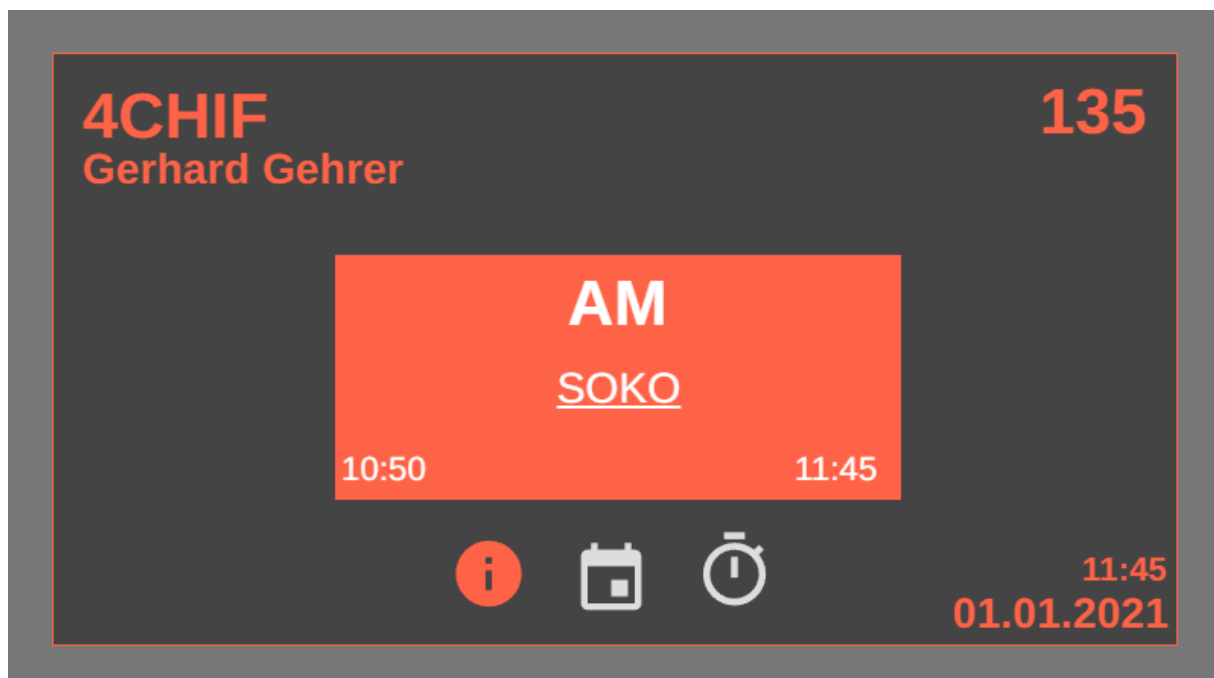


Abbildung 3: UI des Informationstabs (Standardansicht)

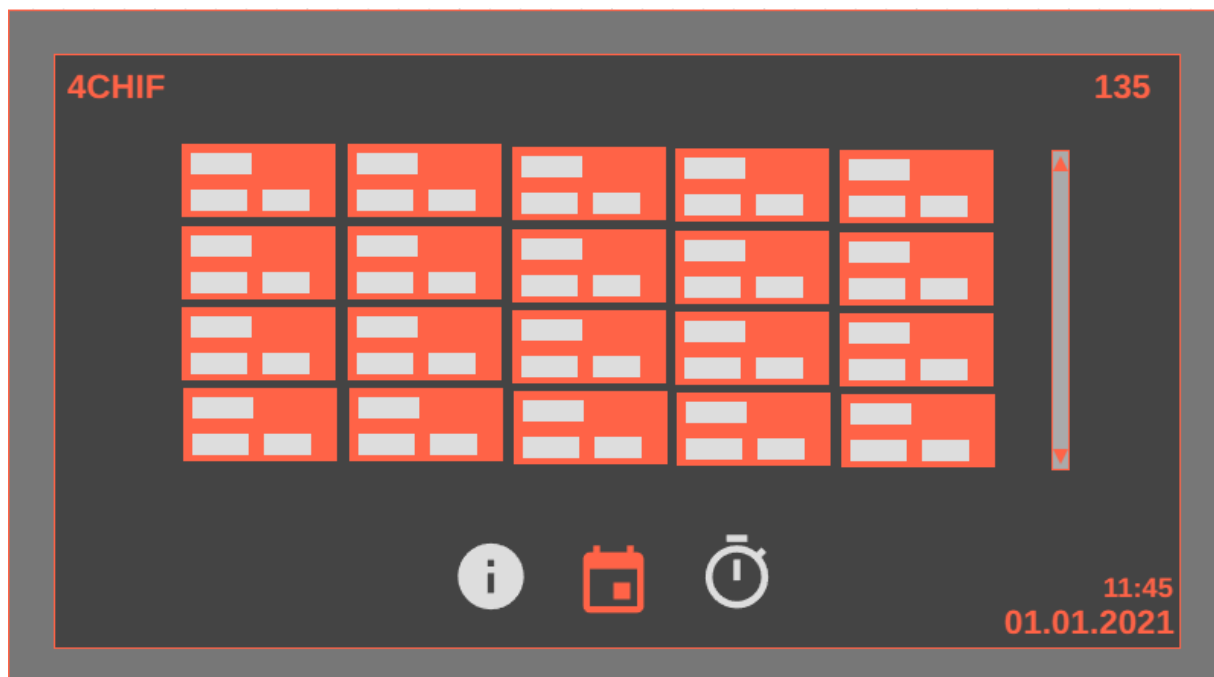


Abbildung 4: UI der Stundenplanansicht

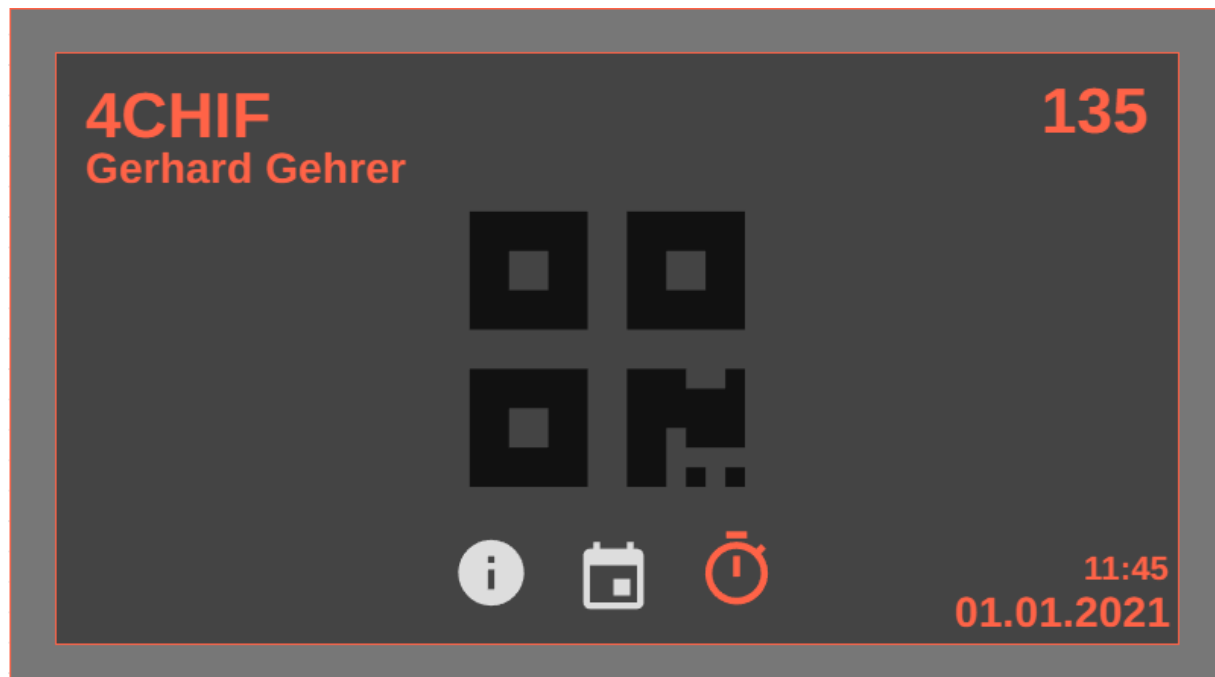


Abbildung 5: UI der Stempelansicht

3 Verwendete Technologien

3.1 Programmiersprachen

- Java
- C# .NET Core

3.2 Hardware

- LCD-Display
- Raspberry PI

3.3 Entwicklungsumgebungen

- IntelliJ IDEA
- Visual Studio 2019
- Visual Studio Code

3.4 Betriebssysteme

- Windows 10
- Linux Ubuntu

4 Anforderungen

4.1 Funktionale Anforderungen

- Anzeige von Rauminformationen
- Anzeige von Stundenplan
- Anzeige von QR-Code zur An- und Abwesenheitskontrolle

4.2 Nichtfunktionale Anforderungen

- Benutzerfreundlich
- Intuitives Design
- Wartbarkeit und Stabilität

5 Abnahmekriterien

5.1 Musskriterien

- Anzeige von Rauminformationen
- Anzeige vom Stundenplan

5.2 Wunschkriterien

- Anwesenheit Stempelsystem
- Zentrales Verwaltungssystem als Desktop Applikation

6 Qualitätsbestimmung

	sehr wichtig	wichtig	weniger wichtig
Korrektheit		 reSign	
Zuverlässigkeit	 reSign		
Robustheit			 reSign
Effizienz		 reSign	
Benutzerfreundlichkeit	 reSign		
Vertrauenswürdigkeit		 reSign	

7 Glossar

QR-Code	Zweidimensionaler, aus Punkten zu einem Quadrat zusammengesetzter, elektronisch lesbarer Code
Server	Rechner, der zentral Dienste zur Verfügung stellt
Client	Anzeigedisplays
Applikation	Softwareanwendung
UI	User Interface bzw. Benutzeroberfläche
API	Eine API ist eine Anwendungsprogrammierschnittstelle, die eine Schnittstelle zwischen verschiedenen Bibliotheken, Frameworks, Anwendungen und Diensten darstellt.
Adapter	Ein Adapter ist ein Entwurfsmuster, mit dem die Schnittstelle einer vorhandenen Klasse als eine andere Schnittstelle verwendet werden kann.