

Technische Spezifikation

Baby-Blooz Baby-Überwachung

Autor:

Abdallah Tahboub, Mohamed Ben Amar

,Skander Baraket, Aziz Khelladi

Letzte Änderung: 05.Juni 2019

Dateiname:TechSpech_Baby-Blooz

Inhaltsverzeichnis	1
1. Einleitung	4
1.1 Überblick	4
1.2 Definitionen und Abkürzungen	4
1.3 Vorhandene Dokumente	4
2. Prozessüberblick	5
2.1 Realisierungsprozess	5
2.2 Technischer Workflow	6
3. System Architektur und Infrastruktur	8
3.1 System Architektur	8
3.2 System Infrastruktur	9
4. Spezifikationen Software	10
4.1 Überblick Komponenten	10
4.2 Schnittstellen zwischen den Komponenten	11
4.3 Beschreibung der Implementierung	11
4.3.1 F1: Log In	11
4.3.2 F2: WiFi Konfiguration	12

Abbildungsverzeichnis

Abbildung 1: Diagramm des Realisierungsprozesses	5
Abbildung 3: Technischer Workflow	6
Abbildung 4: Systemarchitektur	8
Abbildung 5: Spezialisierung Sensoren	9
Abbildung 7: Komponentendiagramm	10
Abbildung 8: Log In Seite	12

Tabellenverzeichnis

Tabelle 1: Versionshistorie	3
Tabelle 2: Vorhandene Dokumente	4
Tabelle 3: Zugriffsdaten	9
Tabelle 4: Beschreibung der Softwarekomponenten und deren Funktionen	10
Tabelle 5: Komponenten Log In	11

© 2019 **Baby-Blooze**

Die Weitergabe, Vervielfältigung oder anderweitige Nutzung dieses Dokumentes oder Teile davon ist unabhängig vom Zweck oder in welcher Form untersagt, es sei denn, die Rechteinhaber/In hat ihre ausdrückliche schriftliche Genehmigung erteilt.

Versionshistorie:

Version	Datum	Verantwortlich	Änderung
1.0	23.05.19	Skander,Mohamed,Abdallah, Aziz	Initiale Dokumentenerstellung
1.0	11.05.19	Mohamed,Aziz,Abdallah,Skander	Diagramme und Schaltpläne
1.0	19.05.19	Abdallah,Skander,Aziz,Mohamed	Texte
1.0	19.05.19	Aziz,Mohamed,Abdallah,Skander	Bilder, Diagramme und Texte
1.0	25.05.19	Abdallah,Skander,Aziz,Mohamed	Bilder, Text und Diagramme
1.0	05.06.19	Skander,Mohamed,Abdallah, Aziz	Diagramme und Texte
2.0		Skander,Mohamed,Abdallah, Aziz	
2.0		Skander,Mohamed,Abdallah, Aziz	
2.0		Skander,Mohamed,Abdallah, Aziz	
2.0		Skander,Mohamed,Abdallah, Aziz	
2.0		Skander,Mohamed,Abdallah, Aziz	
2.0		Skander,Mohamed,Abdallah, Aziz	
2.0		Skander,Mohamed,Abdallah, Aziz	

Tabelle 1: Versionshistorie

1. Einleitung

1.1 Überblick

Damit ein Baby remote überwacht werden kann, wird einen Baby Anzug und eine Mobile App, die miteinander durch API kommunizieren werden, gemacht. Der Hardware Teil (Baby Kleidung) besteht aus einem kleinem passendem Arduino heißt Lilypad, der für unseres Projekt sehr gut geeignet ist, und aus verschiedenen Sensoren, deren Aufgaben es sind, die Messwerte des Babys, aufzunehmen und auf der Oberfläche der App anzuzeigen bzw. zu vergleichen. Die App besteht aus WebAPI, einer Datenbank zum speichern der Daten und einer Benutzeroberfläche(Frontend). Mit der App kann sich der Benutzer mit Hilfe eines QR codes anmelden und Daten abrufen.

Außerdem ist der Benutzer in der Lage die WLAN des Lilypads einzustellen. Notwendige und kritische Infos werden dem Benutzer als Push-Benachrichtigungen angezeigt .

1.2 Definitionen und Abkürzungen

API – Application programing interface

QR Code – Quick response code

1.3 Vorhandene Dokumente

Dokument	Autor	Datum
Lastenheft_Baby-Blooz	Abdallah,Aziz,Skandar,Mohamed	26.05.19
Pflichtenheft_Baby-Blooz	Skandar,Aziz,Abdallah,Mohamed	17.05.19

Tabelle 2: Vorhandene Dokumente

2. Prozessüberblick

2.1 Realisierungsprozess

Um dieses Projekt zu realisieren ist zuallererst die Strukturierung des gesamten Systems erforderlich. Nachdem das Modell der Baby-blooz strukturiert wurde, müssen alle Bauteile des Projektes besorgt werden und gleichzeitig kann mit der Entwicklung begonnen werden. Diese besteht aus zwei Teilen: Der Mobile App und dem Mikrocontroller.

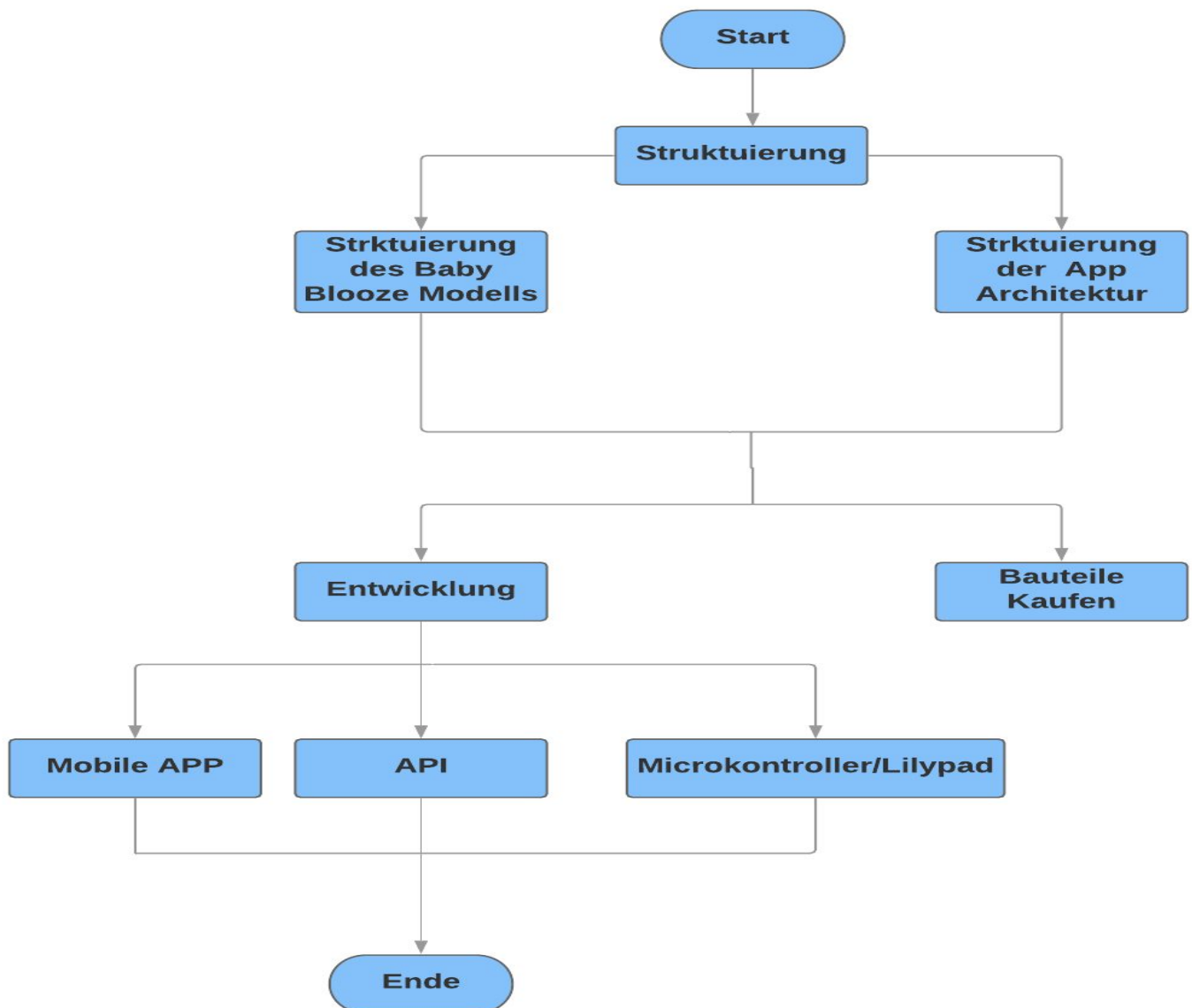


Abbildung 1: Diagramm des Realisierungsprozesses

2.2 Technischer Workflow

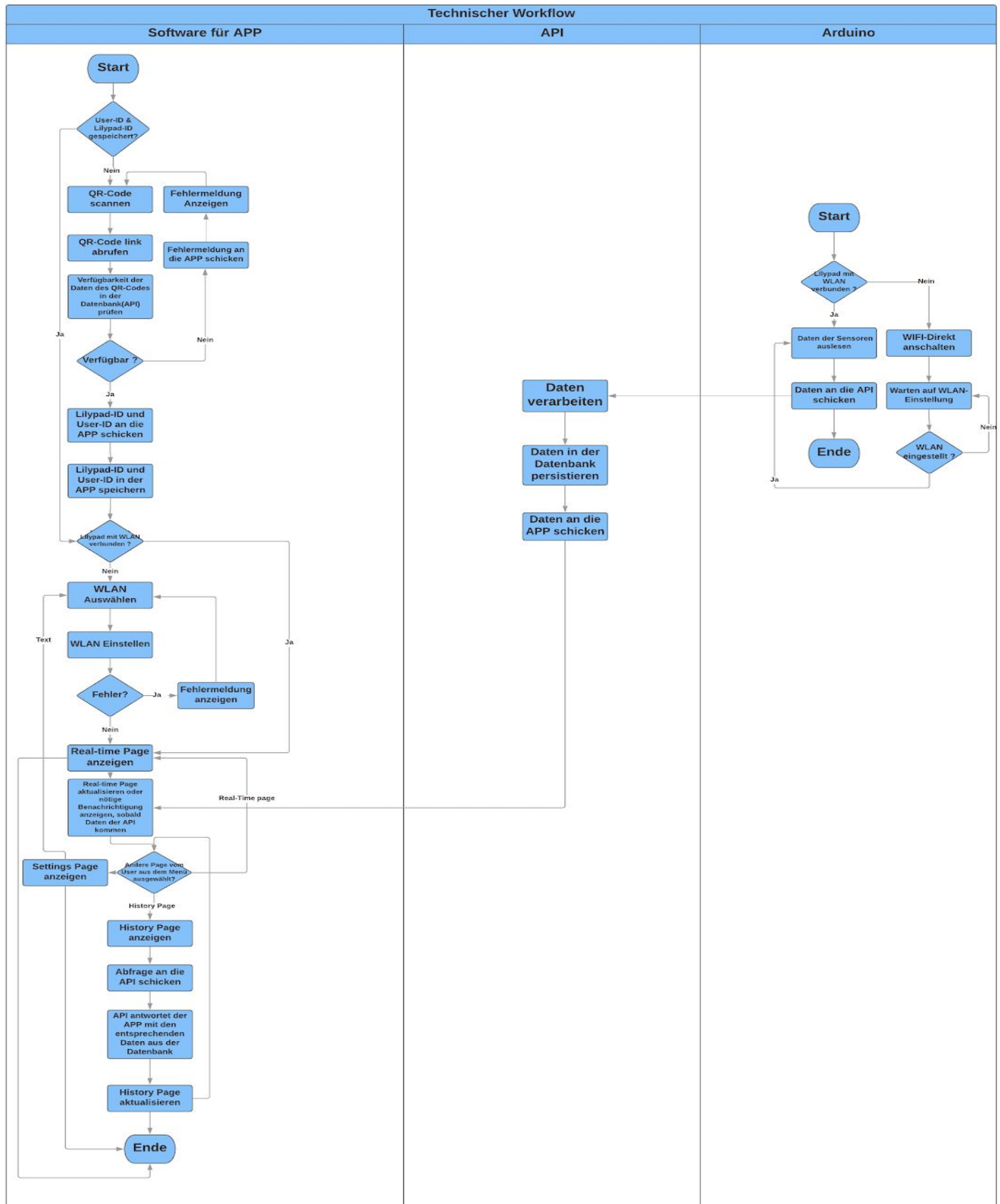


Abbildung 3: Technischer Workflow

3. System Architektur und Infrastruktur

3.1 System Architektur

Die folgende Abbildung zeigt unsere System Architektur in Verbindung mit dem Anwender.

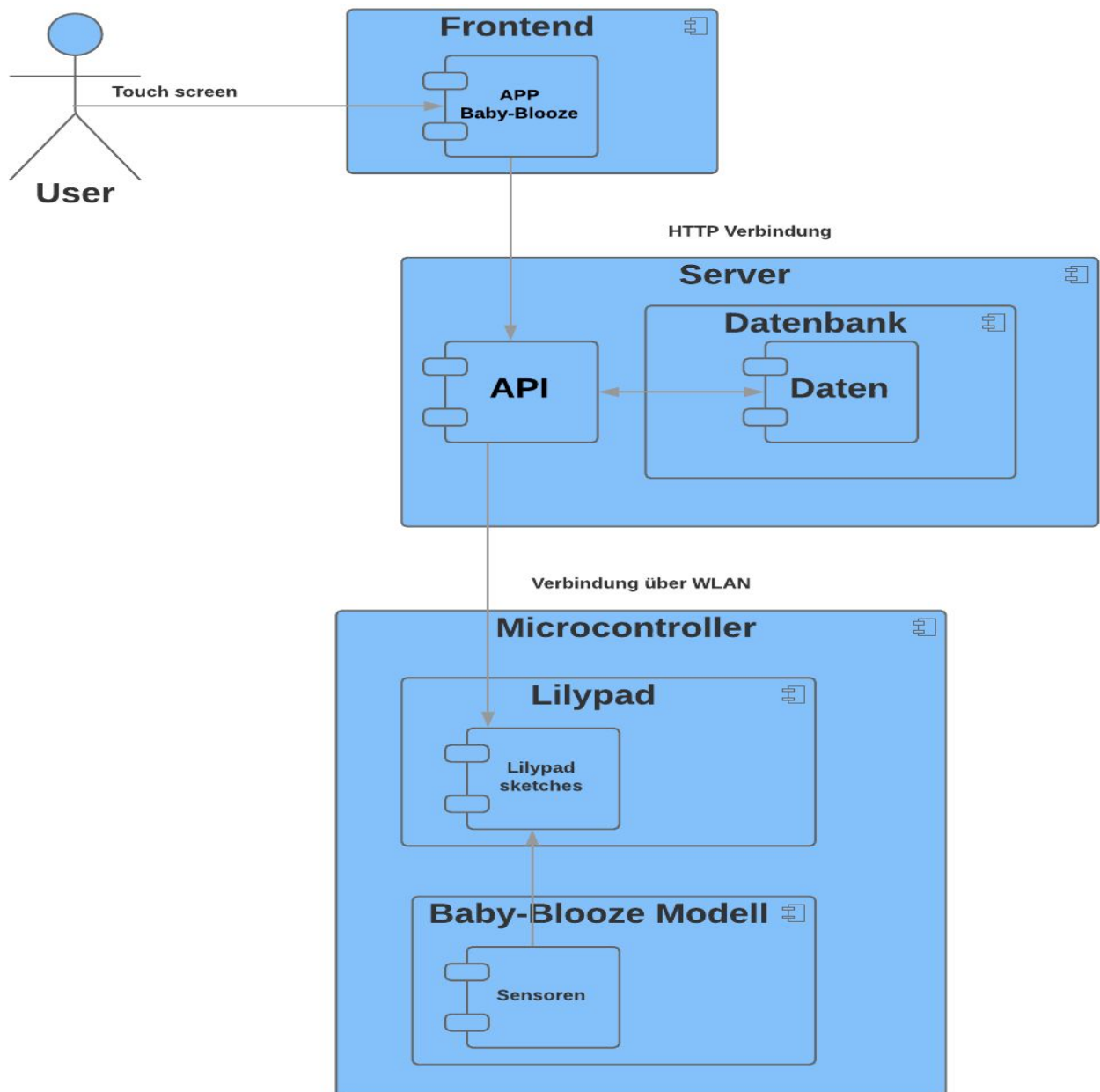


Abbildung 4: Systemarchitektur

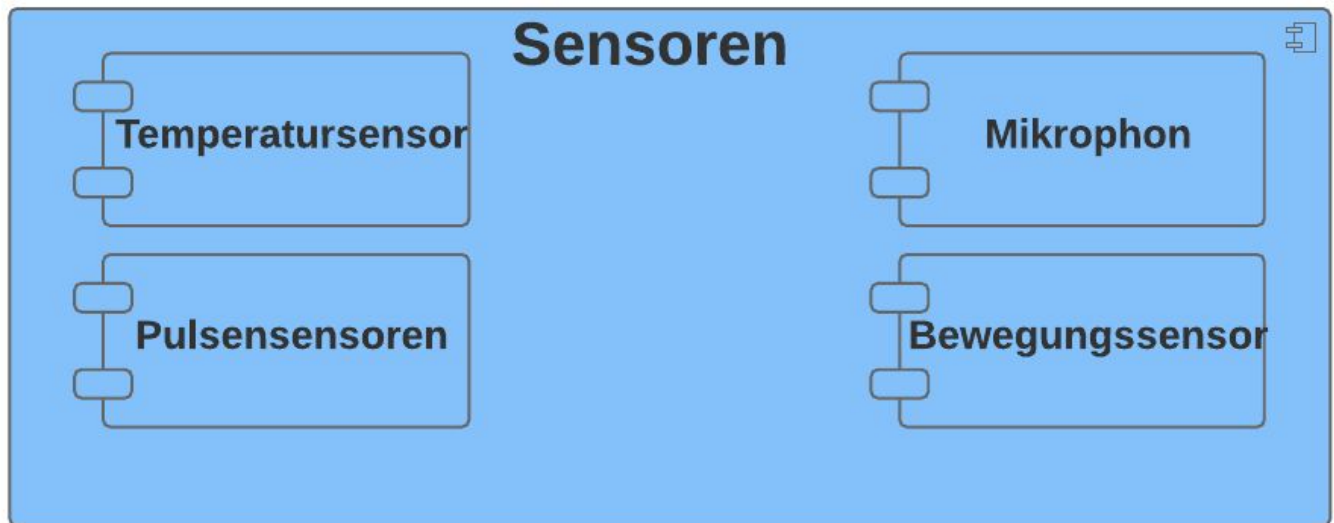


Abbildung 5: Spezialisierung Sensoren

3.2 System Infrastruktur

Daten die benötigt werden um Zugriff auf den Server zu erhalten, worauf die Daten zu dem Webserver, sowie der Datenbank zu finden sind.

IP	141.45.92.216
Benutzer	User ID
QR Code	Generierter QR Code

Tabelle 3: Zugriffsdaten

4. Spezifikationen Software

4.1 Überblick Komponenten

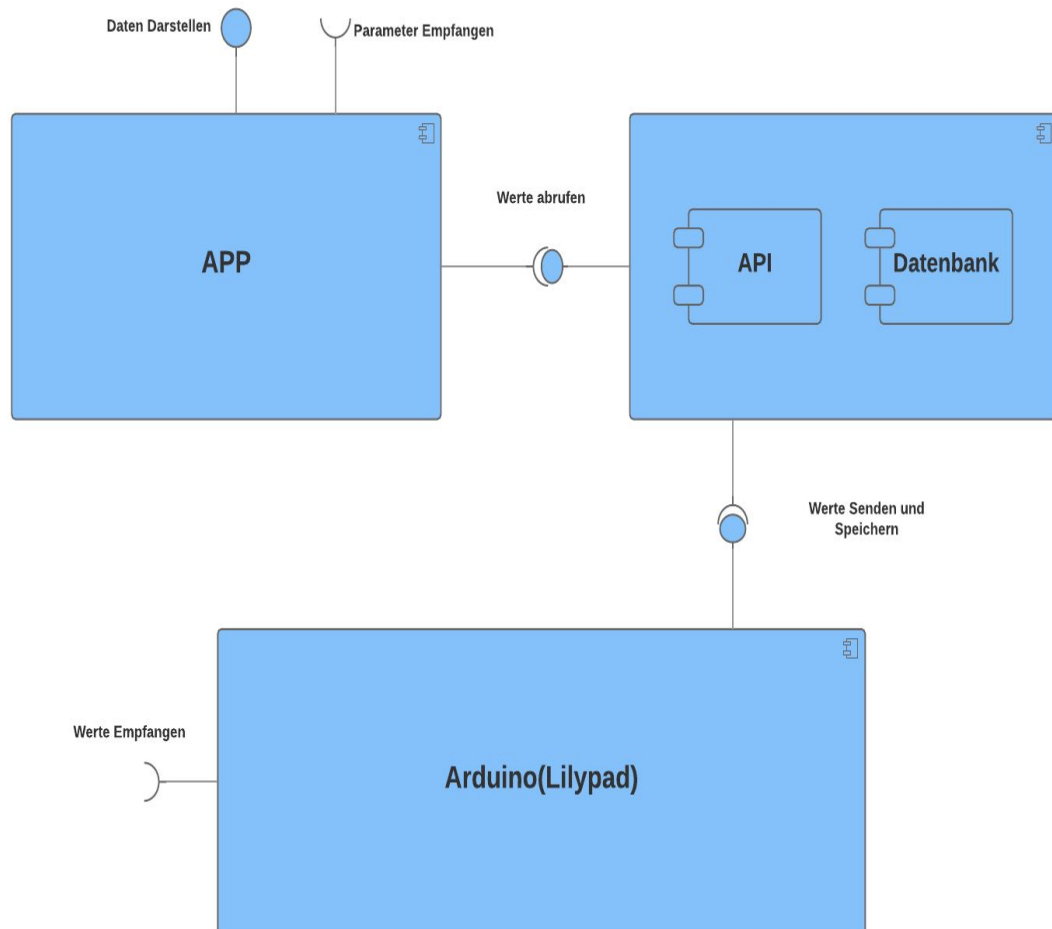


Abbildung 6: Komponentendiagramm

SW Komponente	Funktionen	Sprache / Typ	Ort
App	F1: Login F2: WiFi Konfigurieren F3: Real-Time Daten anzeigen F4: History Page anzeigen F5: Push-Benachrichtigungen anzeigen	HTML, CSS, js, xml typescript	App
API Datenbank	F6: Daten in DB speichern F7: Daten verarbeiten	C#,MySQL,Ty pescrypt	Server
Arduino (LilyPad)	F8: Sensordaten lesen F9: Sensordaten senden.	Arduino	LilyPad

Tabelle 4: Beschreibung der Softwarekomponenten und deren Funktionen

4.1 Schnittstellen zwischen den Komponenten

Die Übertragung der Sensordaten zwischen dem Arduino und dem Server funktioniert über das WLAN mit Hilfe des WLAN Moduls(ESP8266), welches an dem Arduino angeschlossen ist. Die Datenbank befindet sich auf demselben Server weshalb diesbezüglich keine weiteren Schnittstellen erforderlich sind. Der Server bekommt die Sensordaten vom Arduino. Der Server schickt in der echten Zeit die Sensordaten durch WLAN zur APP. Die APP selbst ist eine grafische Schnittstelle, die es dem Nutzer ermöglicht die Daten aus dem System auszulesen und die Anlage fernzusteuern. Die App kann in jeder Zeit die Sensordaten von der Datenbank, die im Server befindet, erhalten.

4.2 Beschreibung der Implementierung

4.2.1 F1: Log In

Um den Zugriff auf alle Daten und Funktionen zu erhalten, muss man sich als erstes im Log In Fenster anmelden. Dazu gibt es ein Feld für den QR-Code zu scannen. Mit dem "Scan QR-Code"-Button wird der Login Versuch gestartet. Bei erfolgreicher Anmeldung wird der Nutzer auf die Startseite, das *Dashboard* weitergeleitet. Bei erfolgloser Anmeldung wird der Nutzer zu einer Neuen QR-code zum scannen gebetet. Ein Account muss dafür in der Datenbank erstellt worden sein. Wenn eine Bluse schon erstellt wurde, wird der angepasste QR-code in der Bank gespeichert.

#	Komponente	Erforderliche Arbeiten
T1	APP	Vergleichen der QR-code mit den QR-code, die aus der Datenbank gespeicherten Blusen.
T4	Datenbank	Bereitstellen des Benutzer mithilfe des QR-codes.

Tabelle 5: Komponenten Log In

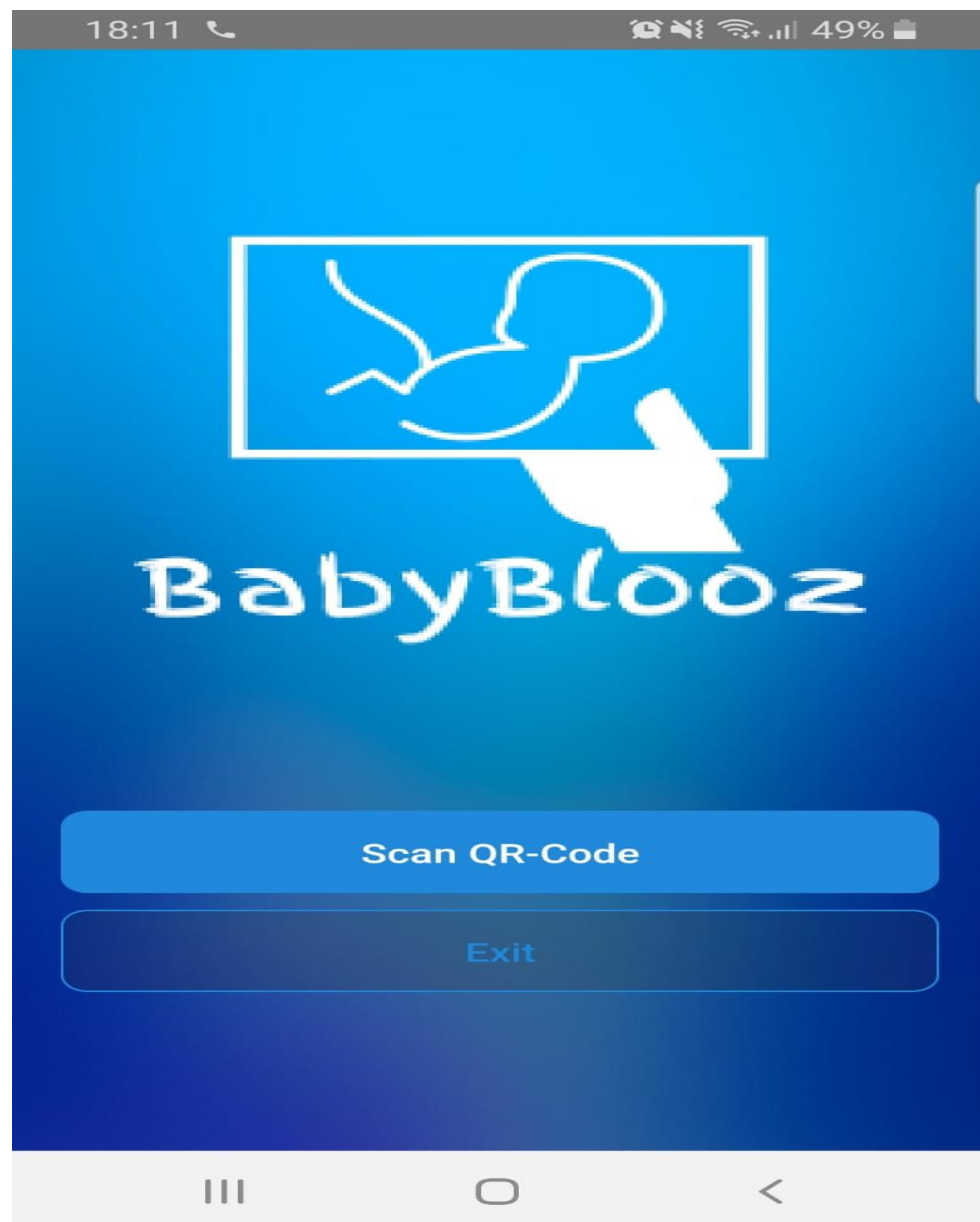


Abbildung 7: Log In Seite

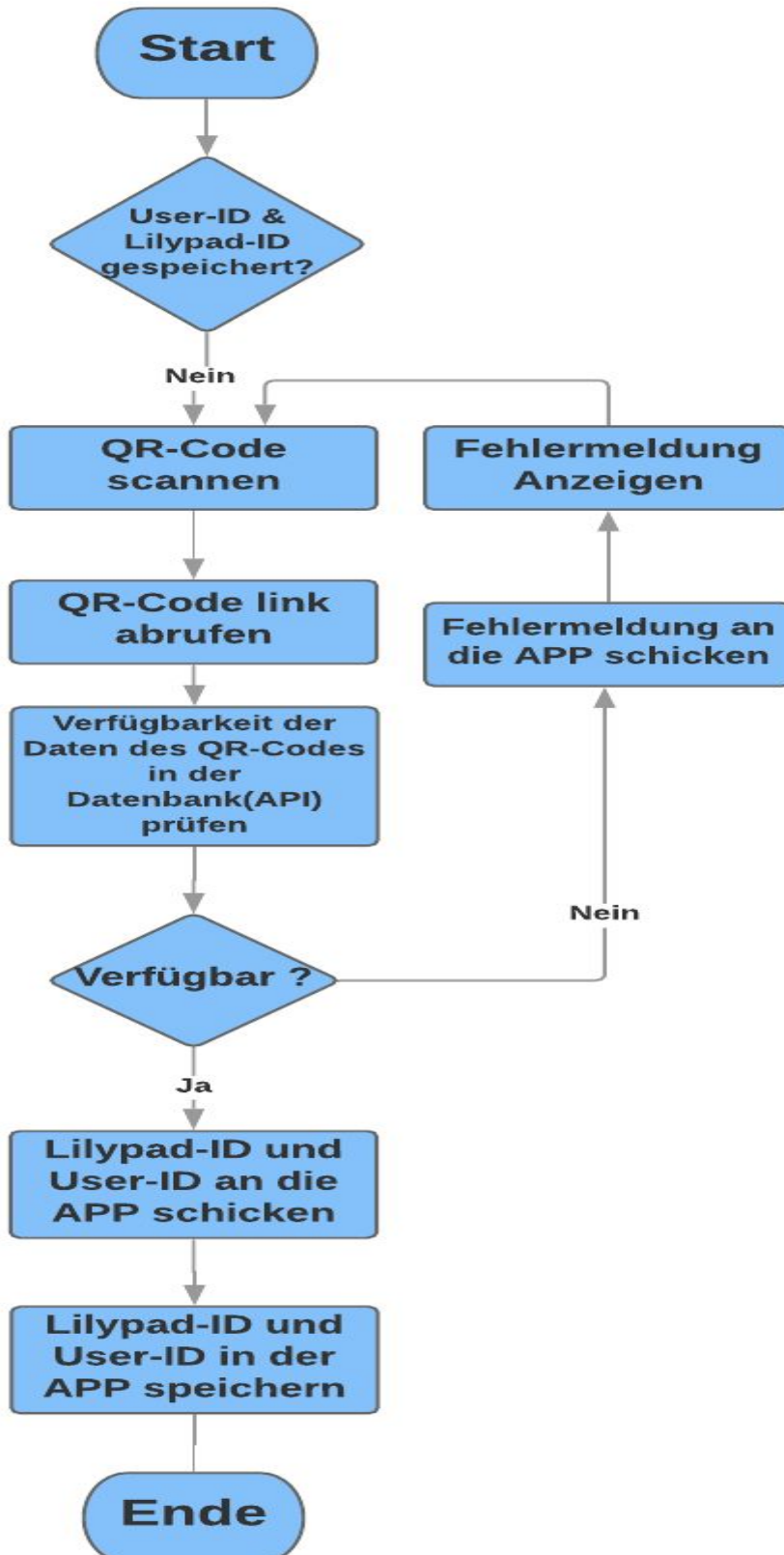


Abbildung 8: Ablauf des Anmeldung

4.2.2 F2: WiFi Konfiguration

Nachdem sich der Benutzer mit dem QR-Code angemeldet hat, landet er auf die WiFi Konfigurationsseite, wo man mit dem Internet verbinden kann. Dort darf man sich aus Sicherheitsgründen ausschließlich mit geschützten Netzen anmelden

#	Komponente	Erforderliche Arbeiten
	APP	<ul style="list-style-type: none">Mit dem Finger kann man das Netz auswählen und das Passwort eingeben.
	API	<ul style="list-style-type: none">Das Passwort wird in der API gespeichert

Tabelle 6: Komponenten Log In

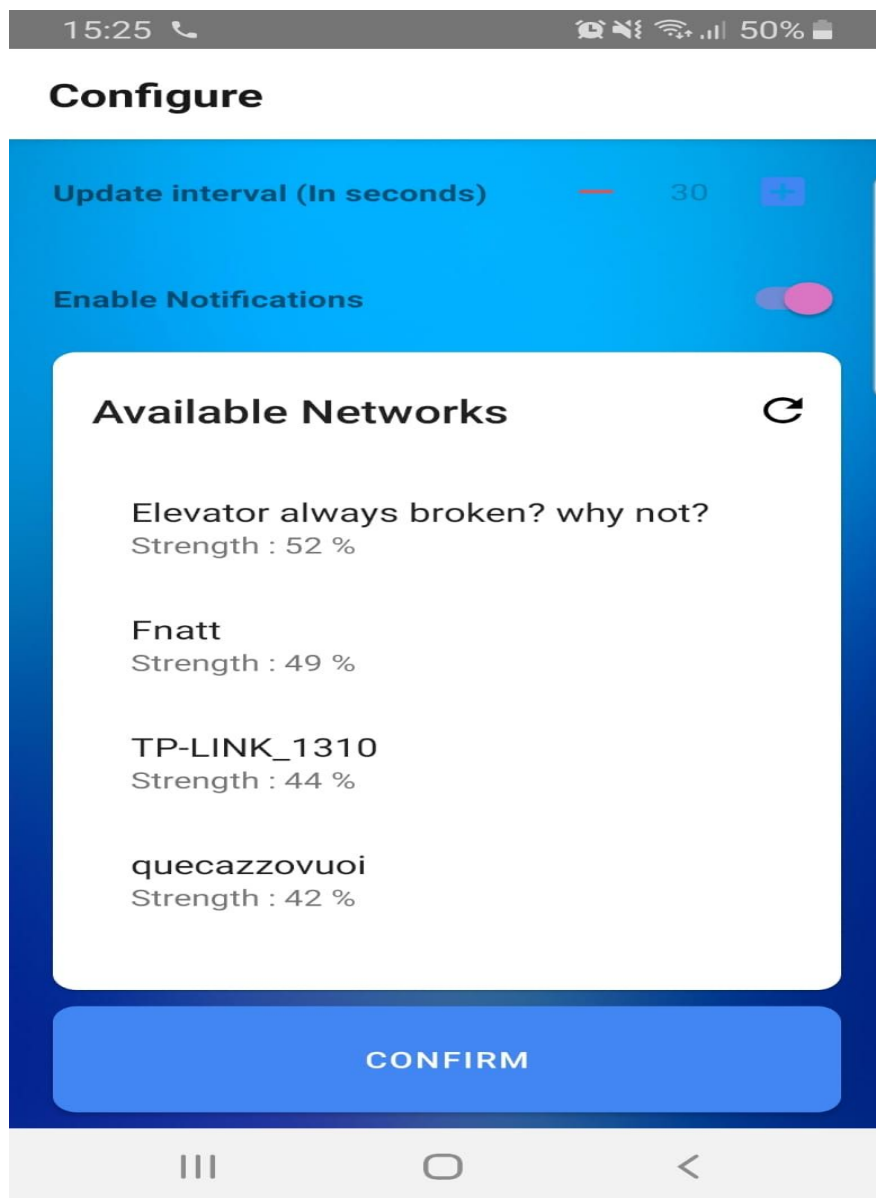


Abbildung 9: Konfigurationsseite

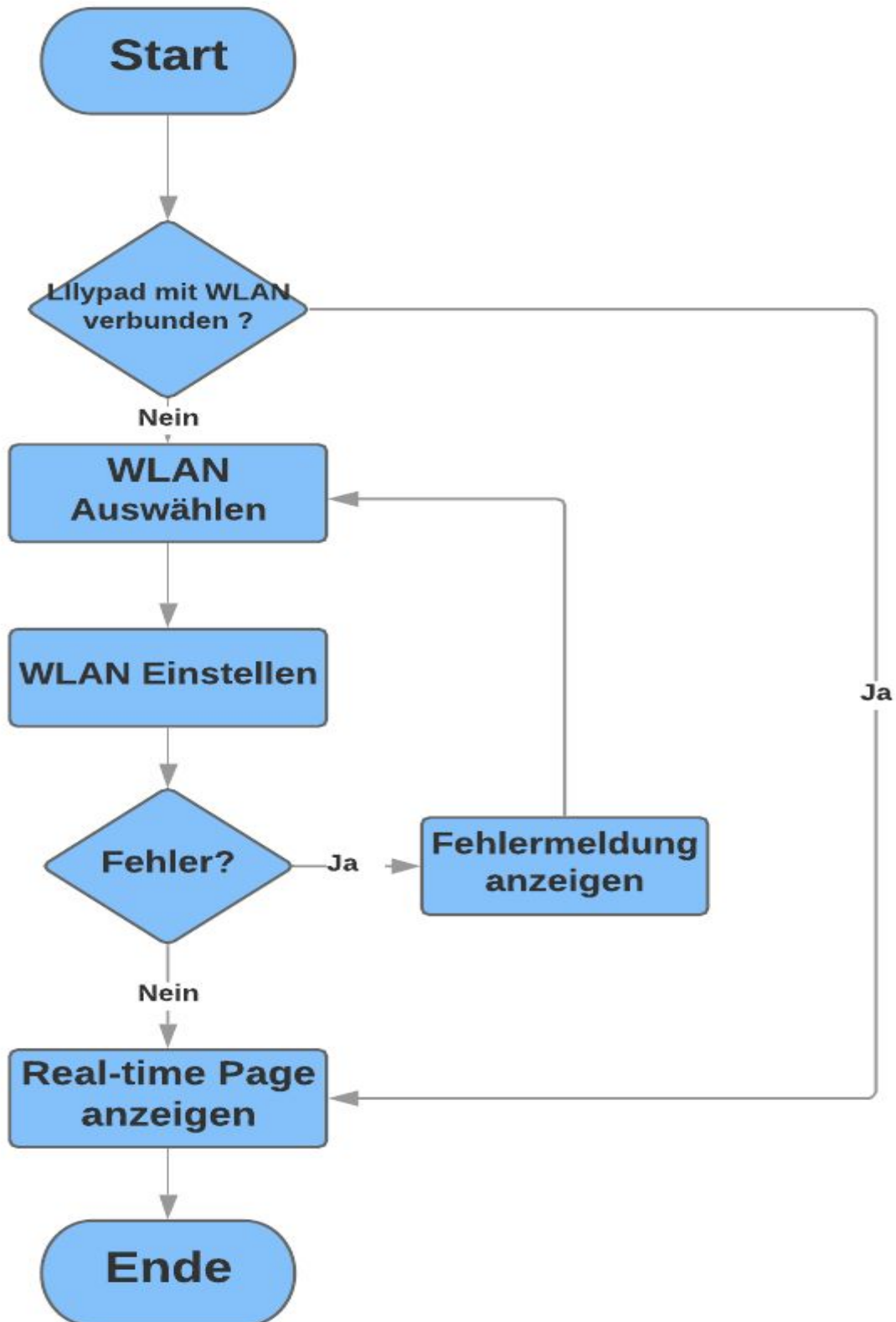


Abbildung 10: Ablauf der WLAN-Einstellung

4.2.3 F3:Real-Time Daten Anzeigen

Sobald die App mit dem Internet erfolgreich verbunden ist und das Netz mit "Confirm" bestätigt wurde, lassen sich Real-Time Daten der Sensoren darstellen. Dort hat man ein Einblick in die Sensordaten, die in der Datenbank gespeichert sind.

#	Komponente	Erforderliche Arbeiten
	APP	<ul style="list-style-type: none"> Es muss eine fehlerfreie Anmeldung des Benutzers bestehen. Die App muss mit dem Internet verbunden sein. Der Benutzer kann mit dem Finger auf der Echtzeit-Seite hoch und runterscrollen oder rein und raus zoomen.
	API	<ul style="list-style-type: none"> Verarbeitung der Sensordaten. Die Daten werden durch die API geschickt.
	Datenbank	<ul style="list-style-type: none"> Datenbank stellt die Sensordaten bereit.

Tabelle 7: Echtzeitdaten

4.2.4 F4:History-Seite

Der Benutzer wird in der Lage sein, die History in der Vergangenheit von 90 Tagen anzusehen. Außerdem hat der Benutzer die Möglichkeit, die 90 Tagen der History Seite in Teilsegmenten anzeigen zu lassen. Eine Woche, einen Monat und drei Monate.

Die History Seite beinhaltet Daten von Herzschlägen, Schlafrhythmus, und die Temperatur.

#	Komponente	Erforderliche Arbeiten
	APP	<ul style="list-style-type: none"> Es muss eine fehlerfreie Anmeldung des Benutzers bestehen (User meldet sich an). Die App muss mit dem Internet verbunden sein (User stellt die WILAN ein) Der Benutzer kann mit dem Finger auf der Echtzeit-Seite hoch und runterscrollen oder rein und raus zoomen.
	API	<ul style="list-style-type: none"> Verarbeitung der Sensordaten. Die Daten werden durch die API geschickt.
	Datenbank	<ul style="list-style-type: none"> Datenbank stellt die Sensordaten bereit.

4.2.5 F5: Push-Benachrichtigungen