## Putztrupp Software

Frau Sauber kommt zu dir und möchte einen Prototypen für folgende Software:

Ihre Firma – Mighty Putztrupp – ist für die Reinigung des größten Einkaufszentrums der Stadt zuständig. Der dortige Gastronomiebetrieb, die Toiletten und Mitarbeiter-Umkleidekabinen gehören in regelmäßigen Abständen gereinigt und desinfiziert. Bis dato hängen einfach A4-Zettel an der Wand und jede/r der zehn Mitarbeiter:innen muss – nach erfolgter Reinigung – seinen Namen eintragen um die Reinigungen zeitlich zu dokumentieren.

Der Zettel hat derzeit drei Pflichtfelder.

Name der Reinigungskraft, Uhrzeit, optional Notiz an den Hausmeister-Service (falls eine Lampe defekt ist, etc…).

Die Modernisierung dieses Arbeitsablaufes stellt sie sich wie folgt vor:

Jede Reinigungskraft scannt, nach erfolgter Reinigung, mit dem Diensthandy den NFC-Chip in den Räumen. Damit bestätigt die Reinigungskraft, dass sie diesen Raum gereinigt hat. Optional kann eine Notiz mit maximal 255 Zeichen mitgesendet werden (für den Hausmeister-Service).

Sie benötigt von dir diese App für Android-Smartphone. Folgende Funktionalitäten müssen gegeben sein:

* Jede Reinigungskraft muss sich einloggen können
* Nach erfolgtem Login stehen der Reinigungskraft folgende Optionen zur Verfügung:
  + Reinigung bestätigen
    - Nach Klick auf Reinigung bestätigen, muss man den NFC-Chip scannen und die Reinigung wird in der Datenbank gespeichert
  + Reinigungshistorie einsehen
    - Hier erhält die Reinigungskraft Einblick in die komplette Historie ihrer getätigten Reinigungen
  + RaumInformation
    - Es muss die Möglichkeit geben, dass man den Raum-NFC-Code scannen kann, um daraufhin zu sehen, wer die letzte Reinigung dort durchgeführt hat. Dieses Feature soll auch ohne Login möglich sein.

Für Frau Sauber ist es vorerst ausreichend, dass der Prototyp derzeit nur für zwei Räume funktioniert (Frauen und Herren-Toilette).

Frau Sauber erwähnt weiters, dass sie das Backend bereits selbst entwickelt hat. Du bist also lediglich für die Entwicklung der App zuständig. Nähere Informationen, zu den NFC-Chips und dem bereits bestehenden Backend, findest du auf den nächsten Seiten.

## NFC-CHIP

Du findest heute im WIFI-Raum zwei NFC-Chips die dem Prototypen dienen um eine Herren und eine Damentoilette zu simulieren.

Die Identifikation findet über den unique NFC-Tag (ID) der NFC-Chips statt.

Wenn man einen NFC-Chip scannt, erhält man – neben selbst definierten Tags – einen 14stelligen NFC-ID im HEX-Format. Beispiel: E1:87:D5:92:5F:66:01.

Diese Codes sind (ohne den Doppelpunkt) in der DB hinterlegt und dadurch ist der Raum identifizierbar.

## Datenbank

Es gibt folgende Entitäten in der Datenbank:

* Employee
  + Id : int PK
  + Email : string(100)
  + Password: string(255)
* Room
  + Id : int PK
  + Name : string(100)
  + NFC\_Id : string(100) (Hier ist der NFC-Code ohne die HEX-Doppelpunkte hinterlegt)

Die beiden Tabellen verbindet eine N:M Tabelle namens ‚Cleaning‘ mit zwei weiteren Properties:

* Id : int PK
* TimeOfCleaning : DateTime
* Note : string? (optionale Message für den Hausmeister-Service)
* EmployeeId : int FK
* RoomId : int FK

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

Die Entitäten der Datenbank benötigst du jedoch nicht wirklich, denn die SQL-DB wird über eine API angesprochen die auf der nächsten Seite im Detail beschrieben ist.

## API

Du erreichst die API über folgende BaseUrl:

[https://wifinfcbackend.upperdevelopment.at](https://wifinfcbackend.upperdevelopment.at/)

Es gibt folgende Endpunkte:

# /Login

GET

* Für den Login der Reinigungskraft

Notwendige Params im Header:

* email : string (email-adresse der Reinigungskraft)
* pwd : string (password der Reinigungskraft)

Returns:

* JTW-Token zur Authentifizierung für alle weiteren Requests an die API

Mögliche Statuscodes:

* 200; OK
* 500; internal Server Error
* 400; Username | Passwort nicht korrekt

# /Cleaning/GetCleaningHistory

GET

* Liefert die CleaningHistory der Reinigungskraft

Benötigt:

* JWT-Token

Returns:

* Folgendes Array:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Mögliche Statuscodes:

* 200; OK
* 401; invalider Token
* 500; internal Server Error
* 501; Token-Error

# /Cleaning/CreateCleaning

POST

* Hiermit wird deine Reinigung des Raums in der DB gespeichert

Benötigt:

* JWT-Token
* Folgendes Objekt im Body als Json

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Mögliche Statuscodes:

* 200; OK
* 400; gesendete NFCID gehört zu keinem Raum der Firma
* 401; invalider Token
* 500; internal Server Error
* 501; Token-Error

# /Room/LastCleanerOfRoom

GET

* Hier bekommst du den letzten Reiniger eines Raums geliefert

Benötigt:

* roomNfc : string im Header

Returns:

* Benutzername der Reinigungskraft als string im Body

Mögliche Statuscodes:

* 200; OK
* 204; Raum wurde bis dato nie gereinigt
* 400; gesendete NFCID gehört zu keinem Raum der Firma
* 401; invalider Token
* 500; internal Server Error
* 501; Token-Error

## Swagger

<https://wifinfcbackend.upperdevelopment.at/swagger/index.html>

## TestDaten

Für deinen Prototyp befinden sich derzeit folgende Testdaten bereits in der DB:

KEINE Einträge in der Cleaning-Table

Einträge in der Employees-Table:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

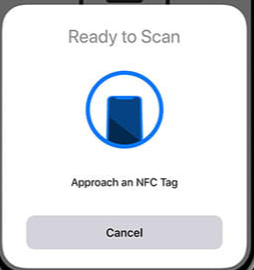
Einträge in der Room-Table:

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

# Hilfe für die APP

* Schreib dir zusammen welche Funktionen die App bieten muss und überleg dir ein passendes Layout / Aufbau der Navigation. Optik spielt im Prototyp natürlich gar keine Rolle. Kann ruhig alles schwarz auf weiss sein.
* Der Prototyp muss nur auf Android funktionieren
* Achtung bei Android:
  + Geht ein Android in den NFC-Listening-Modus erscheint am Android– im Gegensatz zu den iPhones – kein nativer Infoscreen!



Bitte bastele diesen, zwecks UI/UX Erfahrung, selbst (Modalpage?). Sonst würden Android-Anwender gar nicht registrieren, dass ihr Phone nun im NFC-Listening Modus ist.

* Frau Sauber hat gehört, dass sich angeblich das Nuget-Paket Plugin.NFC von Franckbour – auch CrossNFC genannt – hervorragend im .NET Umfeld für NFC-reading und writing eignet. Versuche den Prototypen damit um umzusetzen. Dieses ist auch hervorragend auf Github dokumentiert.

Viel Spaß und Erfolg.