1.Bevezető

2.Projekt célja

3.Követelmény specifikációk

    3.1. Felhasználói követelmények. Use case diagram.

    3.2. Rendszerkövetelmények

       3.2.1 Funkcionális

       3.2.2 Nem-funkcionális

4.Tervezés

    4.1 Architektúra. Komponens diagram

    4.2 Modulok leírása

Osztálydiagramok. Esetleg más diagramok

     UI terv

Adatbazis terv

    4.4 Managelés–GitHub-Kanban

    4.5 Schematic (ha van)

5. Alkalmazás működése

    5.1 UI –konkrét megvalósítás

    5.2 Hardver (ha van)  
alkatresek bemutatasa

6.Összegzés

    6.1 További fejlesztési lehetőségek

Szekvencia diagram kommunikaciok ui tol rendszerig kommunikaciok bemutatasa  
aktivitas diagram

Tartalom

[1. Bevezető 2](#_Toc184670778)

[2. Projekt célja 2](#_Toc184670779)

[3. Követelmény specifikációk 3](#_Toc184670780)

[3.1. Felhasználói követelmények. Use case diagram. 3](#_Toc184670781)

[3.2. Rendszerkövetelmények 4](#_Toc184670782)

[3.2.1. Funkcionális 5](#_Toc184670783)

[3.2.2. Nem-funkcionális 6](#_Toc184670784)

[4. Tervezés 7](#_Toc184670785)

[4.1. Architektúra. Komponens diagram 7](#_Toc184670786)

# Bevezető

Az okosotthon egy modern technológián alapuló rendszer, amely lehetővé teszi a háztartásban levő különféle funkciók központi vezérlését, akár távolról is az interneten keresztül (pl fűtés szabályozása , ajtók zárása, ablakredönyök vezérlése , mosógép ki-be kapcsolása stb.) Ezt a rendszert a szoftver, és az érzékelők közti kapcsolat teszi lehetővé, amit a “dolgok internetjének" (Internet of Things – IoT) neveznek

Ezek az otthonok nemcsak kényelmet nyújtanak, hanem alkalmazásukkal jelentősen lecsökkenthetjük az energiafelhasználásunkat, illetve nagyban befojásolják a biztonságunkat. Az okosotthon-technológia folyamatosan fejlődik, egyre fejlettebb a gépi tanulási algoritmusok és mesterséges intelligencia alkalmazásával. A legfejletebb rendszerek már mobiltelefonról is irányíthatok egy erre a célra fejlesztett applikáción keresztül.

# Projekt célja

A projekt célja egy olyan okos otthon rendszer megvalósítása, amely megkönnyíti a felhasználó mindennapi életét azáltal, hogy a háztartás különböző funkcióit automatizálja, illetve valós időben megjeleníti azok állapotát. Ezenfelül a felhasználónak legyen lehetősége különböző eszközök vezérlésére egy irányítópulton keresztül, vagy távollétében egy telefonos applikációt alkalmazva legyen hozzáférése a teljes rendszerhez.

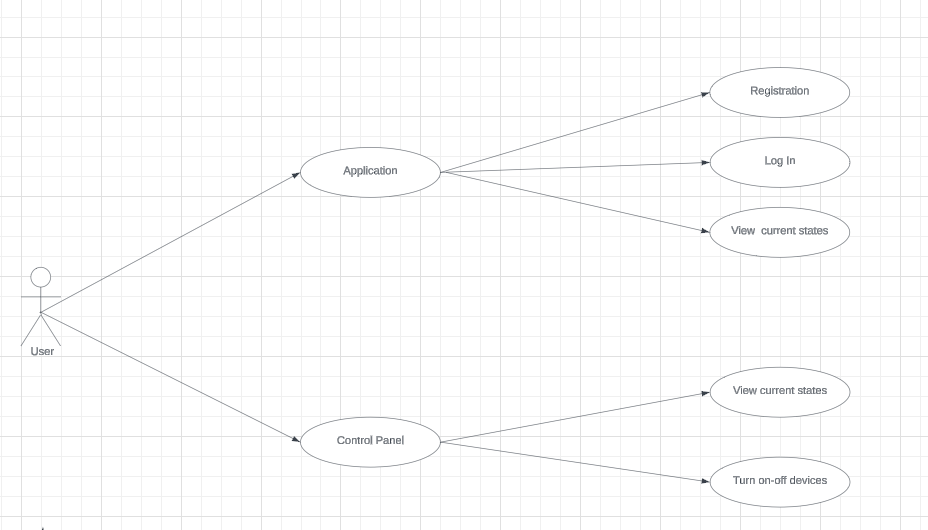
A megvalósítás során alkalmazott technológiák a modern okosotthon-rendszerek alapjait képezik, amelyek a jövőben továbbfejleszthetők, például mesterséges intelligencia segítségével.

# Követelmény specifikációk

## 3.1. Felhasználói követelmények. Use case diagram.

* A rendszernek lehetővé kell tennie, hogy a felhasználók valós időben megtekintsék az otthoni eszközök állapotát
* Az irányítópulton keresztül a felhasználóknak egyszerűen el kell tudniuk végezni a vezérlést , illetve az applikáción keresztül akár távolról is
* A rendszer felhasználói felületének (dashboard) könnyen áttekinthetőnek és logikus felépítésűnek kell lennie.
* A szoftvernek müködnie kell úgy automatikusan mint manuálisan
* A rendszernek értesítéseket kell küldenie kritikus helyzetekben
* A felhasználók számára meg kell jeleníteni múltbeli adatokat és statisztikákat diagramon
* A felhasználónak tudnia kell állítani a különböző időzítőket
* A rendszernek kompatibilisnek kell lennie a Google Home okos eszközzel.

Use case diagram

Use case diagram 

## Rendszerkövetelmények

Hardverkövetelmények:

* Az irányítópult futtatásához legalább 4 GB RAM, 2 GHz-es processzor, és 32 GB szabad tárhely szükséges.
* Az alkalmazás futtatásához minimum 2 GB RAM, és 500 MB szabad tárhely ajánlott.

Szoftverkövetelmények:

* Az irányítópult működéséhez Windows vagy Linux operációs rendszer szükséges.
* Az alkalmazás futtatásához Android 5.0 vagy újabb operációs rendszer szükséges.

Hálózati követelmények:

* A rendszernek internetkapcsolatra van szüksége a megfelelő működéshez, az OpenWeather API használata miatt.

### Funkcionális

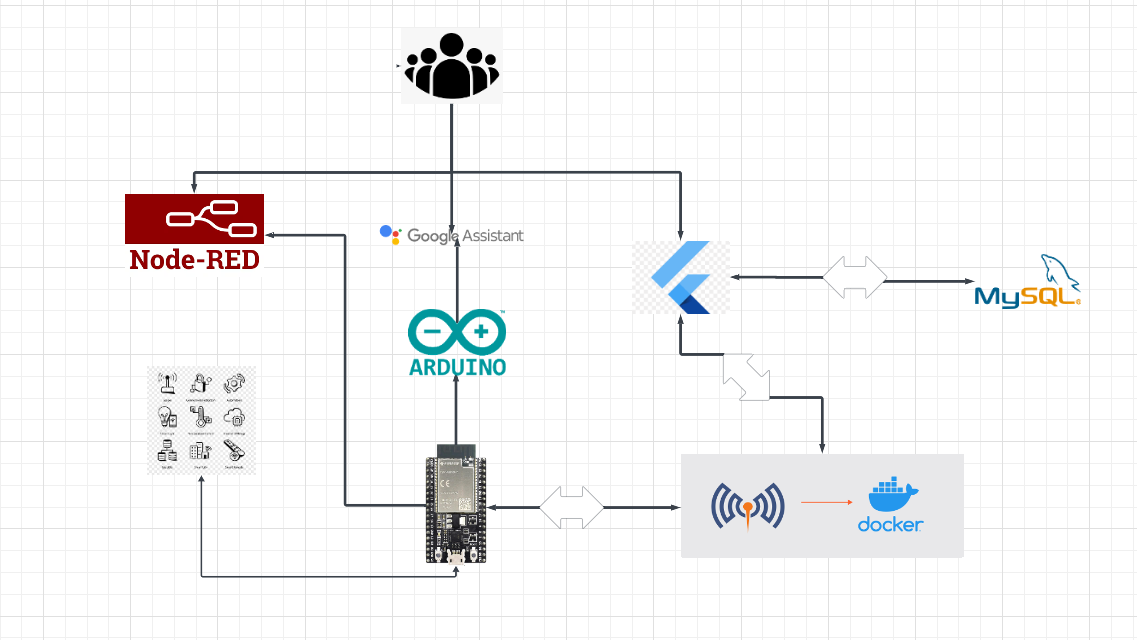
1. Felhasználói bejelentkezés: A rendszernek lehetővé kell tennie a felhasználók számára, hogy felhasználónév és jelszó megadásával bejelentkezzenek az applikáció
2. **Eszközvezérlés**: A rendszernek lehetővé kell tennie a felhasználók számára, hogy a háztartás eszközeit az irányítópulton keresztül vezérelhessék
3. **Távoli hozzáférés**: Az applikációval a felhasználóknak lehetőséget kell biztosítani a háztartási eszközök távoli vezérlésére internetkapcsolaton keresztül
4. **Állapotfigyelés** : A rendszernek valós idejű visszajelzést kell adnia a csatlakoztatott eszközök állapotáról (pl. világítás bekapcsolva)
5. **Automatizált funkciók**: A rendszernek támogatnia kell előre beállított időzítőket végrehajtania
6. **Biztonsági értesítések**: A rendszernek értesítéseket kell küldjön vészhelyzet esetén.
7. **Időjárás-integráció**: A rendszer kapcsolatban kell álljon az időjárás-szolgáltatással ( OpenWeather API), hogy az időjárási adatok alapján automatizálhassa a redőny leengedését eső esetén.

### Nem-funkcionális

1. **Teljesítmény**: Az irányítópult felületének legfeljebb 2 másodperc alatt kell betöltődnie egy átlagos internetkapcsolat mellett .
2. **Megbízhatóság**: A rendszernek a hét minden napján, napi 24 órában elérhetőnek kell Lennie
3. **Biztonság**: Az applikáción keresztül történő kommunikációnak titkosítottnak kell lennie
4. **Karbantarthatóság**: A rendszernek logikus felépítésünek kell Lennie, hogy az új funkciók hozzáadása vagy a hibajavítás könnyen elvégezhető legyen.
5. **Felhasználói élmény**: Az irányítópultnak könnyen kezelhető felületetnek kell lennie

# Tervezés

## 4.1. Architektúra. Komponens diagram



Alapvetoen a projektunk ket reszre bonthato : Iranyitopult , es egy applikacio.

Az applikacio lehetoseget nyújt valos idoben nyomonkovetni a renszdszert illetve lehetoseget nyújt a rendszer vezerlesre stb … ajnalott egy nagyobb kepernyon mukodtetni es akar falra szerelni a lakasban pl nappaliban . Az applikacio hasznos lehet ha nem tartozkodunk otthon vagy eppem stb stb ….. Leiras a node red rod es az aplikaciorol kulon kulon

## 4.1 Managelés–GitHub-Kanban

## 

A projekt menedzseléséhez a GitHub Kanban funkcióját használtuk, egyszerűsége miatt. A Kanban tábla lehetővé tette, hogy vizuálisan nyomon kövessük a feladatok állapotát, és segítségével hozzárendelhettük a feladatokat a csapattagokhoz, és valós időben követhettük a projekt fejlődését.