Park1t&Go

Készítette:

Kovács Zétény

Tóth Bence

Zsiros Máté

**Nyilatkozat:**

**Kijelentjük, hogy a programkód teljes szövege, valamint a dokumentáció teljes egésze saját szellemi termékünk, melyet önállóan készítettünk.**

# Feladatspecifikáció

A célunk az volt, hogy egy könnyen használható parkolási platformot hozzunk létre. Ez nemcsak egy új lehetőséget ad, hanem leegyszerűsíti a parkolóházak kezelését és a parkolást.

A Park1t&Go többféle felhasználói szinttel rendelkezik: parkolóház-üzemeltetők, adminisztrátorok és parkolókat használó emberek. A cél az, hogy könnyebben lehessen foglalni és kezelni a parkolóhelyeket, valamint egyszerűbbé teszi az ügyfelek és üzemeltetők együttműködését.

Ez a rendszer nem csupán egy újabb alkalmazás, hanem segít az adminisztrációban, leegyszerűsíti a parkolást, és támogatja a környezetbarát megoldásokat, hozzájárulva a városi közlekedés jobb működéséhez.

# Alkalmazott szoftverek és technológiák

* 1. Alkalmazott szoftverek:
     + Visual Studio 2022 v17.13
     + Visual Studio Code v1.81
     + Android Studio v3.1.14
     + XAMPP v3.3.0
     + GitHub v3.16.0
     + Firefox verzió: 136.0.4
     + Git v2.49.0
     + Trello
     + Figma
     + Node.js v22.14.0
  2. Programnyelvek, keretrendszerek és modulok:

**Frontend:**

* + - * **Nyelv: JavaScript(x)**
      * **Keretrendszer: React**
      * **UI Komponenskönyvtár:**
      * **Routing:**

**Backend:**

* **Nyelv: JavaScript**
* **Web Framework:**
* **Adatbázis kezelő:**
* **Authentikáció és Authorizáció: JsonWebToken**
* **HTTP kliens:**
* **Környezeti változók kezelése:**
* **Middleware:**

**Android alkalmazás:**

* **Nyelv: Kotlin**
* **Keretrendszer: LadyBug 2022**

**Asztali alkalmazás:**

* **Nyelv: C#**
* **Keretrendszer: .Net**

**Tesztelés:**

* 1. **Megvalósítás**

A parkolóház projektet otthoni környezetben, saját munkaeszközökön, engedélyezett tanórák keretében valósítottuk meg. Az asztali alkalmazást C# WinForm-ban fejlesztettük, az Androidos verziót Kotlin nyelven írtuk, míg a weboldalt React és JavaScript technológiával építettük fel. A három platform közötti adatáramlást egy központi RESTful API biztosítja, amely egységes kommunikációt tesz lehetővé a backend és a különböző felületek között.

A verziókövetést GitHub-on végeztük, a weboldal működését pedig a Google Chrome és a Mozilla Firefox legújabb verzióiban teszteltük. A projekt során külön kihívást jelentettek az új technológiák – például a React – gyors elsajátítása, valamint az egy főre jutó munkamennyiség, amely gondos tervezést és precíz kivitelezést igényelt a többplatformos megoldás kialakításában.

# A felület funkciói, felépítése és használata

# Adatbázis

# Forráskód és a projekt felépítése

# Tesztelési dokumentáció

# Források

Tartalomjegyzék

[I. Feladatspecifikáció 3](#_Toc195263141)

[II. Alkalmazott szoftverek és technológiák 4](#_Toc195263142)

[III. A felület funkciói, felépítése és használata 5](#_Toc195263143)

[IV. Adatbázis 5](#_Toc195263144)

[V. Forráskód és a projekt felépítése 5](#_Toc195263145)

[VI. Tesztelési dokumentáció 5](#_Toc195263146)

[VII. Források 5](#_Toc195263147)