

Házi feladat dokumentáció

Android alapú szoftverfejlesztés

Félév: 2018 őszi



RoboBlue

LUDVIG ANDRÁS LÁSZLÓ - (FMLVA8)

andras.ludvig@yahoo.de

Laborvezető: Gazdi László

Bemutató

Az alkalmazás célja Bluetooth kapcsolaton keresztül kommunikációs felület kialakítása egy önvezető járművel. Az alkalmazás ötlete a RobonAUT versenyen való indulással kapcsolódóan fogalmazódott meg, ugyanis a fejlesztés közben hasznos, ha az önjáró RC autóval lehet kommunikálni. Így figyelve az állapotát és esetleg beavatkozni.

Főbb funkciók

Az alkalmazással lehetőség van kiválasztani melyik Bluetooth eszközre csatlakozzon a telefon.

Fő funkciója egy Bluetooth soros terminál egy külön felületen. Illetve programozható gombok (távírányítószerű, illetve funkció gombok) egy másik felületen. A funkció gombokhoz tetszőleges üzenet rendelhető.

Az alkalmazás képes Bluetooth-on keresztül üzenet küldeni, illetve fogadni a RobonAUT-ra megépített önjáró járműtől.

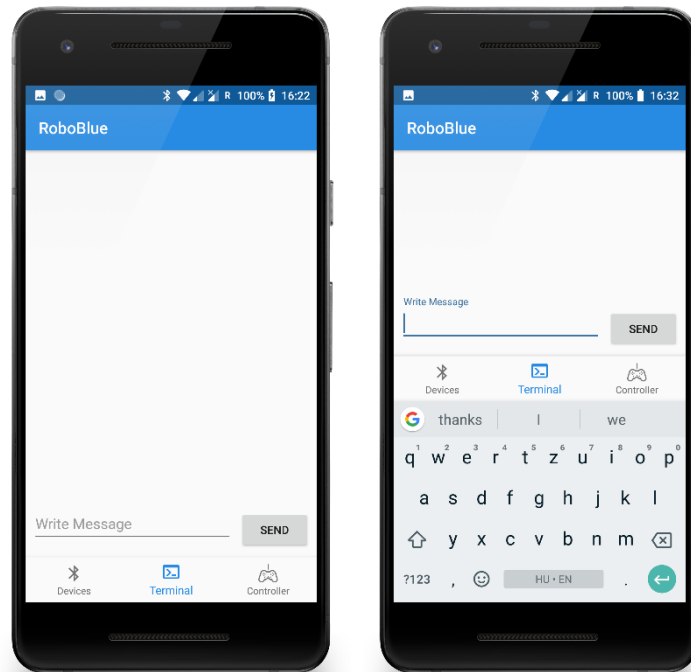
Felhasználói kézikönyv

Az alkalmazás kezdőképernyője (1.ábra) egy olyan felület, ahol kiválaszthatjuk melyik Bluetooth eszközre szeretnénk csatlakozni. Itt csak a már egyszer párosított eszközök láthatóak. Új eszközt a telefon beállítása közt kell először hozzáadni.



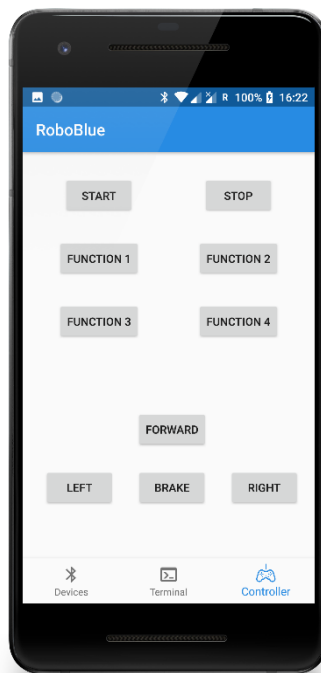
1. ábra: Bluetooth eszközök felülete

A következő felület (2.ábra), amire az alsó menü második elemére kattintva érünk el, egy soros Bluetooth terminál. Itt jelenik meg az üzenet, amit a már csatlakoztatott Bluetooth eszköztől fogadunk. Az alján található szöveg beviteli mezőbe, pedig a mi általunk elküldeni kívánt üzenetet írhatjuk be. A küldés a „SEND” gomb megnyomásával történik.



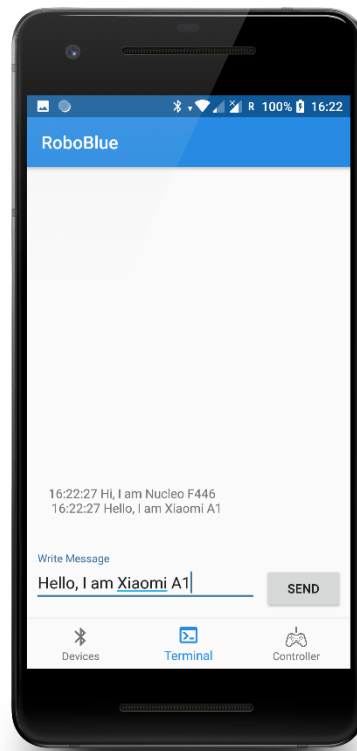
2. ábra: Soros Bluetooth terminál felülete

A legutolsó menü pont látható a 3.ábrán. Ez egy olyan felület, ahol különböző gombok helyezkednek el. Ezekhez tartozhat adott funkcionalitás, mint például a „START” gomb megnyomásával egy „START” üzenetet küldünk a másik Bluetooth eszköznek. Illetve van négy darab programozható funkció gomb. Ezeket hosszan nyomva beállíthatjuk a gomb nevét és az üzenetet, amit küldeni fog a telefonunk, ha rányomunk a gombra.



3. ábra: Távirányító és funkció gombok felülete

A 4.ábra egy üzenet küldését és fogadását szemlélteti. A kapott üzenet, az üzenet érkezéskor aktuális idővel kiegészül és úgy jelenik meg. A felületen lehet görgetni és így visszanézni a régebbi üzeneteket.



4. ábra: Üzenet küldése és fogadása példa

Felhasznált technológiák

Az alkalmazásomban a következő technológiákat használtam fel:

- **Fragment API** használata a különböző nézetekhez (Bluetooth eszközök, Terminál, Távirányító funkciók)
- **SQLite** alapú adattárolás a funkció gombok nevének és üzenetének megőrzéséhez
- **Interface** alapú kommunikáció a fő Activity és a különböző Fragment-ek közt
- Bluetooth eszközök megjelenítése **RecyclerView** segítségével
- Activity és szál közti kommunikáció **Handler** használatával
- **Bluetooth** kapcsolat létrehozása a telefon és egy másik Bluetooth eszköz közt
- **Bluetooth üzenet fogadása és küldése** saját protokoll alapján
- Bluetooth kommunikációhoz különböző **szálak (Thread)** indítása
- **BroadcastReceiver**-re való feliratkozás a Bluetooth modul kikapcsolásának az érzékeléséhez
- **Intent** indítása és az eredményének a feldolgozása a Bluetooth modul bekapcsolásához
- **Bottom Navigation Bar** használata a különböző felületek (mindegyiknek saját Fragment-je van) közti váltáshoz
- Funkció gombok beállításához **PopupWindow** feldobása és kezelése
- **Toast** üzenetek feldobása a folyamatok jelzésére. Például, ha sikerült bekapcsolni a Bluetooth modult.

Fontosabb technológiai megoldások

Számomra a legnehezebb a Bluetooth kommunikáció stabil működésének a megvalósítása volt. Az első nehézséget a két irányú kapcsolat megnyitása jelentette. A kétirányú kapcsolatnál fontos, hogy a küldő és fogadó metódusok külön szálban fussanak. Ezek után pedig az üzenet küldés és fogadás folyamatát kellett definiálni. Itt azt a módszert választottam, hogy minden üzenetet egy „\r” karakterrel kell zárni. Így az önjáró autón lévő mikrokontroller és a telefon is tudja, mikor fejeződött be az üzenet. Ehhez az Android API `InputStream.read` metódusát alkalmaztam és bájtanként ellenőriztem a fogadott adatot. Az üzenet fogadás és küldés a kapcsolatok megnyitásához hasonlóan egy külön szálát indít. A kommunikáció az üzeneteket kezelő szál és a fő Activity közt Handler-rel történik. A kapott üzenetet egy a TerminalFragment-ben lévő TextView-ban jelenik meg. Emellett ugyan ebben a Fragmentben található egy TextInput is, amibe a beírt szöveget a SEND gomb megnyomásával lehet elküldeni a Bluetooth eszköznek (a mi esetünkben az önjáró autónak).