**UNIVERZITA KONŠTANTÍNA FILOZOFA V NITRE**

FAKULTA PRÍRODNÝCH VIED

Ovládanie robota zo stavebnice merkur pomocou androidu

bakalárska práca

**2016 Darko Šajben**

**UNIVERZITA KONŠTANTÍNA FILOZOFA V NITRE**

FAKULTA PRÍRODNÝCH VIED

OVládanie robota zo stavebnice merkur pomocou androidu

bakalárska práca

Študijný odbor: 9.2.9 Aplikovaná informatika

Študijný program: Aplikovaná informatika

Školiace pracovisko: Katedra informatiky

Školiteľ: PaedDr. Martin Magdin, Ph.D.

Nitra 2016 Darko Šajben

# Zadanie bakalárskej práce

Zadanie bakalárskej práce je dokument, ktorým vysoká škola stanoví študentovi študijné povinnosti v súvislosti s vypracovaním bakalárskej práce.

Zadanie spravidla obsahuje:

* typ záverečnej práce (BP, DP, Rigorózna práca, Habilitačná práca a ďalšie),
* názov bakalárskej práce,
* meno, priezvisko a tituly študenta,
* meno, priezvisko a tituly školiteľa,
* v prípade externého školiteľa meno, priezvisko a tituly konzultanta,
* školiace pracovisko,
* meno, priezvisko a tituly vedúceho pracoviska,
* anotáciu bakalárskej práce,
* jazyk, v ktorom sa práca vypracuje,
* dátum schválenia zadania.

# Abstrakt

ŠAJBEN, Darko: Ovládanie robota zo stavebnice Merkur pomocou Androidu. [Bakalárska práca]. Univerzita Konštantína Filozofa v Nitre. Fakulta prírodných vied. Školiteľ: Titul. Meno Priezvisko, Titul. Stupeň odbornej kvalifikácie: Bakalár odboru Aplikovaná informatika. Nitra: FPV, 2016. .... s. (uviesť počet strán BP)

Robotické stavebnice v súčasnosti predstavujú platformu fungujúcu za pomoci mikrokontroléra a grafického vývojového prostredia. Takáto stavebnica môže byť použitá k vytváraniu samostatných interaktívnych zapojení alebo pripojená pomocou softvéru k počítaču. V bakalárskej práce sa zameriavame na konštrukciu a ovládanie robotického zariadenia, ktorého základ tvorí mikrokontrolér Arduino (zmena je zdôvodnená v bakalárskej práci), pričom komunikácia sa uskutočňuje pomocou Bluetooth. V úvode bakalárskej práce analyzujeme robotické stavebnice, ktoré sa používajú v súčasnosti. Tiež sú uvedené rôzne spôsoby vytvárania aplikácii pod OS Android, zostava a funkcionalita robota. Taktiež sme popísali postup programovania Arduina a aplikácii Android. Použité technológie sa často používajú a preto praktické využitie nepredstavuje problém. Môže sa používať ako na zábavu tak i na edukáciu.

Kľúčové slová: Robot. Arduino. open-source. Android. Bluetooth. Apache Cordova.

# Abstrakt

ŠAJBEN, Darko: Ovládanie robota zo stavebnice Merkur pomocou Androidu. [Bachelor Thesis]. Constantine the Philosopher University in Nitra. Faculty of Natural Sciences. Supervisor: Titul. Meno Priezvisko, Titul. Degree of Qualification: Bachelor of Applied Informatics. Nitra: FNS, 2016. .... p.

Arduino is an open source platform which functions with help of a micro controller and graphic development environment. It can be used in order to create standalone interactive connections or it can be connected to software on the computer. In the thesis we are focusing on construction and control of robotic equipment which base is Arduino while communication takes place via Bluetooth. The introduction contains analysis of robotic kits which are used in the present. It also includes description of various ways of creating Android applications, composition and functionality of the robot. There is also mentioned the method of programming the Arduino and Android application. The technology behind this project is used frequently, therefore we do not see any obstacles for implementation and practical use. It can be used for fun as well as for the educational purposes.

Keywords: Robot. Arduino. open-source. Android. Bluetooth. Apache Cordova.

# **obsah**

[Zoznam skratiek a značiek 6](#_Toc447032397)

[Slovník pojmov 8](#_Toc447032398)

[Úvod 9](#_Toc447032399)

[1 Bezdrôtové ovládanie zariadení 10](#_Toc447032400)

[1.1 Merkur stavebnice 10](#_Toc447032401)

[1. Arduino 10](#_Toc447032402)

[1.3 Raspberry Pi 11](#_Toc447032403)

[1.4 Lego mindstorm 12](#_Toc447032404)

[1.5 Využitie Elektronických stavebníc v praxe 13](#_Toc447032405)

[1.6 Bezdrôtové technológie 14](#_Toc447032406)

[1.6.1 Bluetooth 14](#_Toc447032407)

[1.6.2 WiFi 14](#_Toc447032408)

[1.7 Možnosti vývoja aplikácii 15](#_Toc447032409)

[1.7.1 Natívne aplikácie 15](#_Toc447032410)

[1.7.2 Hybridné aplikácie 15](#_Toc447032411)

[2 Ciele bakalárskej práce 16](#_Toc447032412)

[3 Realizácia Ardubota 16](#_Toc447032413)

[3.1 Zostavenie robotického zariadenia 17](#_Toc447032414)

[3.1.1 Arduino Nano 17](#_Toc447032415)

[3.1.2 Motorový ovládač HG7881 (L9110S) 18](#_Toc447032416)

[3.1.3 Bluetooth modul HC-06 18](#_Toc447032417)

[3.2 Vývoj aplikácie na ovládanie zariadenia 19](#_Toc447032418)

# **Zoznam skratiek a značiek**

|  |  |
| --- | --- |
| **HTML** | Komunikačný protokol na zabezpečenie prenosu HTML dokumentov medzi serverom a klientom. |
| **PC** | Personal Computer |
| **USB** | Universal Serial Bus |
| **IDE** |  |
| **PWM** | Pulse Width Modulation |
| **Ucc** |  |
| **GND** | Ground |
| **BD ADR** | Bluetooth Device Address |
| **PAN** | Personal Area Network |
| **LAN** | Local Area Network |
| **WAN** |  |
| **MAC** |  |
| **IEEE** |  |
| **DOM** | Document Object Model |
| **AJAX** |  |

# Úvod

Robotika sa v súčasnosti stáva populárnym vzdelávacím nástrojom najmä v oblastiach vedy a techniky pre všetky stupne škôl (základné, stredné i vysoké) ako metóda aktívneho učenia, ktorá umožňuje podporu a rozvoj rôznych predmetov, ako je matematika, informatika, mechanika, informačné technológie, elektronika, programovanie, umelá inteligencia (RUS, 2006). Zároveň jej zavedenie v podobe riešenia určitého problému, zvyšuje možnosti vytvárania tzv. projektových skupín, ich vzájomnú spoluprácu a v neposlednom rade rozvoj zručnosti a schopností, ktoré sú v tomto prípade založené na praktických postupoch (Haeuser, 2016). Možnosti implementácie a využívania robotiky vo vzdelávacom procese boli umožnené vďaka vývoju jednoduchých vzdelávacích robotických súprav. Edukačnú stavebnicu zvyčajne môžeme charakterizovať ako hračku rozvíjajúcu intelektuálne a kognitívne procesy jedinca, ktorá je zložená zo skupiny dielov. Tieto diely je možné zostavovať do trojrozmerných zostáv a vytvárať modely. Rozširovaním klasických stavebníc (napr. Merkur) o  elektronické súčiastky vznikla nová kategória stavebníc, tzv. elektronické stavebnice. Jadrom elektronickej stavebnice môže byť mikrokontrolér alebo mikroprocesor. Používajú sa v automaticky riadených systémoch, ako sú riadenie automobilového motora, lekárske prístroje, dochádzkový systém, elektronický zámok či automatické otváranie a zatváranie dverí. Práca je zameraná na stavanie robotického autíčka s možnosťou diaľkového ovládanie pomocou Aplikácie Android. V prvej kapitole sa zaoberáme analýzou a porovnávaním rôznych typov stavebníc, bezdrôtových technológii a vývojom aplikácii Android. Prehľad na trhu nám umožnil správne sa rozhodnúť pri výbere stavebnice pre náš projekt. V nasledovnej kapitole sme určili cieľ a jednotlivé podciele bakalárskej práce, ktoré sme použili ako smernice počas vývoja celého projektu. Tretia kapitola je praktická časť bakalárskej práce. V nej je zahrnuté postup zostavenie robotického zariadenie a vývoj aplikácie na ovládanie robota cez Bluetooth.

# 1 Bezdrôtové ovladanie zariadení

V  kapitole sú analyzované technológie ktoré sa používajú počas vývoja bezdrôtovo ovládateľných zariadení. Na začiatku sú porovnávané robotické stavebnice, ďalej sa nadväzujú bezdrôtové technológie používané v súčasnosti. V poslednej časti kapitole sa zaoberáme spôsobmi programovanie aplikácii pre Android.

## 1.1 Merkur stavebnice

Merkur je česká stavebnica, ktorá sa úspešne predávala od polovice 20. rokov 20. storočia. Jej predaj sa úplne zastavil po roku 1989, avšak v posledných 5 rokoch sa opäť pod novým majiteľom spoločnosti snaží stavebnica opätovne etablovať na trhu s didaktickými pomôckami. Avšak chýba jej dodatočný marketing a najmä trpí nízkou podporou z technického hľadiska.

Základom stavebnice sú pásy a profily rôznych tvarov a dĺžok z lakovaného oceľového plechu, so sieťou predvŕtaných otvorov. Tieto diely sa spájajú skrutkami a maticami. Je možné z nich postaviť rôzne modely, cestné a koľajové vozidlá, žeriavy a mnoho iných konštrukcií. Stavebnica obsahuje aj ďalšie súčiastky, rôzne hriadele, ozubené a neozubené kolesá. V niektorých sadoch stavebnice je možné nájsť aj kladky, buldozérové pásy a elektromotory. Umožňuje tak stavbu veľmi realistických konštrukcií rôznych strojov (merkurtoys.cz, 2012). Nevýhodou tejto stavebnice je nedostupnosť dokumentácie a súčiastok na rozširovanie stavebnice. Kvôli prekážkam na ktoré sme narazili so stavebnicou Merkur, rozhodli sme zmeniť platformu pre vývoj robota (po konzultácii s vedúcim práce). Na výber sme mali viacero stavebníc a k nášmu projektu sa najviac hodila platforma Arduino.

## 1.2 Arduino

Arduino je open-source platforma pre vytváranie elektronických prototypov, vyznačuje sa jednoduchým použitím hardvéru a softvéru. Platforma Arduino je prispôsobená pre začiatočníkov, nevyžaduje dlhoročne skúsenosti v elektronike či programovaní. Vďaka dostupnosti príkladov pre túto platformu sa môžeme naučiť rozsvietiť LED diódy, ovládať motor, či spracúvať údaje zo senzorov. K PC sa pripája pomocou USB portu, tak že dosku dokážeme naprogramovať z akéhokoľvek zariadenia s takýmto portom a nainštalovaným vývojovým prostredím.

Programy pre platformu Arduino sa bežne vytvárajú v grafickom programovacom prostredí Arduino IDE. Je to open-source softvér voľne stiahnuteľný na oficiálnej stránke projektu (arduino.cc, 2015). Popri samozrejmej podpore pre Windows, k dispozícii máme softvér pre MAC a Linux, tak že nie sme obmedzení na používanie komerčných OS. Pre projekty v tomto prostredí sa používa názov „sketch“, čo v preklade do slovenčiny znamená skica. Arduino ponúka vopred pripravené programy pre ukážkovú činnosť rôznych modulov, ktoré nájdeme v *File→Examples*. Ak používateľ má k dispozícii moduly kompatibilné s Arduinom, môže nahrať ukážkový program a rovno otestovať. Arduino softvér tiež umožňuje kompiláciu, nahrávanie programu, či sledovanie komunikácie sériovej linky. Ak už máme uložené niektoré svoje projekty, ľahko ich zobrazíme v menu *File→Sketchbook.* Záložka *Tools* ponúka okrem automatického formátovania textu tiež nastavenie programátora, ktorým bude program nahratý do zariadenia, samotný typ toho zariadenia a sériový port. Programovací jazyk v ktorom sú programy písané vychádza z projektu wiring a je založený na jazyku C++. Súbor s kódom pre Arduino je uložený s príponou .*ino* a možno ho editovať v ľubovoľnom textovom editore, však prostredie Arduino IDE ponúka zvýrazňovanie syntaxe aj pohodlnú správu vytvorených programov. Je tiež možné v každom projekte nastaviť kľúčové slova ktoré sa budú zvýrazňovať spolu s ostatnými kľúčovými slovami. (arduino.cc, 2013)

Aby sme program mohli nahrať a otestovať, potrebujeme dosku Arduino. Existuje viacero druhov a líšia sa počtom vstupno-výstupných vývodov, veľkosťou pamäte, rozmermi a využitím. Existujú aj špecificky zamerané typy dosiek ako napríklad Arduino Yun, ktorá uľahčuje prácu cez Wifi. Môžeme si zaobstarať oficiálne dosky, ale aj rad kompatibilných dosiek od iných výrobcov. Rôzne typy dosiek sú predstavené na **obrázku č.**  Pre každý druh Arduino dosky je potrebné v prostredí Arduino IDE nastaviť patričný typ, kvôli nastaveniu programovacích parametrov. Väčšina dosiek má osadený mikrokontrolér od firmy Atmel. Napájanie dosky je možné cez USB port, ktorý zároveň slúži ako spojenie s PC pre programovanie, prípadne využitie externej batérie. Väčšina typov pracuje s napätím 5V. Na doske sa nachádzajú niekoľko typov vývodov:

* Digitálne vývody slúžia na ovládanie zariadenia pomocou logických hodnôt 0 a 1. Týchto vývodov býva na doske najviac.
* PWM je označenie pre impulzovú šírkovú moduláciu, umožňuje meniť výstupné napätie na výstupe vďaka čomu môžeme kontrolovať rýchlosť otáčok motora.
* Analógové vývody majú svoje vlastné miesto na doske a slúžia na snímanie hodnôt napätia v rozmedzí 0V – Ucc, teda bežne 0 - 5V, v programe následne interpretovaná ako číselná hodnota v intervale 0 – 1023.
* Napájacie vývody poskytujú pracovné úrovne napätia (3.3V a 5V), uzemnenie (GND) a vstup pre napájanie doske cez batériu.

## 1.3 RASpberry pi

Raspberry Pi je open-source minipočítač založený na platforme ARM. Keď naň pripojíme klávesnicu, počítačovú myš a monitor, je to úplne funkčný počítač veľkosti platobnej karty, ktorý beží na OS Linux. (raspberrypi.org, 2014) Najnovšia verzia tohto mini počítaču pod názvom Raspberry Pi 3 bola predstavená 29. Februára 2016. Na doske obsahuje sieťovú kartu, WiFi a Bluetooth. Na rozšírenie modulmi máme k dispozícii 40 GPIO pinov. (raspberrypi.org, 2016) Ako základný jazyk na Raspberry Pi platforme bol zvolený skriptovací jazyk Python. Výhodou používania skriptovacích jazykov je, že nepotrebujeme kompilovať zdrojový kód, čo uľahčuje a urýchľuje proces programovania. Treba spomenúť, že už niekoľko rokov dozadu sa Python nachádzal medzi prvými piatimi najpoužívanejšími jazykmi na svete. Alternatívou na programovanie na platforme Raspberry Pi je Skretch ktorý umožňuje začiatočníkom programovať bez poznania syntaxe. Sketch je predinštalovaný v OS Raspbian, čo je verzia Debiana prispôsobená pre Raspberry Pi počítače (safaribookonline.com, 2013). Podľa tabuľky 1 je Raspberry Pi 3 z pohľadu parametrov jednoznačne víťazom, ale pre niektoré projekty je Arduino lepšia a lacnejšia voľba.

Tabuľka 1 Porovnanie základných parametrov medzi Arduino a Raspberry Pi

|  |  |  |
| --- | --- | --- |
|  | **Arduino Nano** | **Raspberry Pi 3** |
| **Základ** | Mikrokontrolér | Mikroprocesor |
| **Frekvencia** | 16MHz | 1.2Ghz |
| **Šírka registru** | 8-bit | 64-bit |
| **RAM** | 2KB | 1024MB |
| **GPIO** | 20 | 40 |
| **Maximálny prúd** | 40mA | 5-10mA |
| **Zdroj** | 175mW | 700mW |
| **Operačný systém** | N/A | Linux |

Ak máme za úlohu kontrolovať stav senzorov a zároveň meniť rýchlosť otáčania motora, Arduino bude tá správna voľba, v prípade že pracujeme v grafickom prostredí a spracúvame veľké množstvo údajov tak si zvolíme Raspberry Pi.

## 1.4 LEgo mindstorm

Keď k najznámejšej stavebnici na svete Lego, pridáme elektronické súčiastky, dostaneme edukačnú robotickú stavebnicu Lego Mindstorm NXT. Na jej používanie sa nemusíme vyznať v elektronike, dokonca nemusíme vedieť ani jeden programovací jazyk. To je dôvod prečo touto stavebnicu často stretáme na základných a stredných školách. Na pridanie nového modulu stačí pripojiť jeden kábel, no nevýhodou je, že musíme používať iba originálne moduly. Aj keď sa pomocou tejto stavebnice dajú zostrojiť veľké a zložité projekty, na profesionálne použitie sa neodporúča. Na programovanie tejto stavebnice máme viacero možnosti. Základné vývojové prostredie Lego Mindstorm EV3 umožňuje programovanie stavebnice tzv. blokovou štruktúrou. Bloková štruktúra je označenie pre zdrojový kód programu, ktorý je rozčlenený do samostatných blokov používaných v štruktúrovanom programovaní. Bloky rozčleňujú kód na súvislé logické časti (podmienku if, cykly for, do-while a podobné). V prípade práce na zložitejšom projekte existujú knižnice pre programovacie jazyky ako sú: Java, C/C++ a Python. (Brian, 2007)

## 1.5 využitie Elektronických stavebníc v praxI

Elektronická súprava je tvorená súborom elektronických súčiastok používaných na vytvorenie elektronického zariadenia. Obsahujú elektronické súčiastky, schému zapojenia a návod na montáž. K dispozícii sú dva odlišné typy súprav, tie ktoré sa použijú na stavanie jedného projektu, alebo tie ktoré umožňujú postaviť celý rad projektov. V súčasnej dobe klesá popularita stavebných zariadení od nuly, čiastočne aj kvôli zložitosti miniatúrnych elektronických obvodov, namiesto toho sa používajú platformy, ktoré uľahčujú návrh a programovanie (napr. Arduino). Edukačné elektronické súpravy ako napríklad Lego Mindstorm, sú prispôsobené deťom a umožňujú nespájkovaciu konštrukciu a jednoduché programovanie. Pomocou Lego Mindstorm NXT stavebnice môžeme postaviť rôzne zariadenia - napríklad autíčko, robotické rameno, či kráčajúceho robota ale aj mnoho zložitejšie inteligentné zariadenia ako je napríklad Cubestromer 3. Bol postavený z jedným účelom a to vyriešiť hlavolam pod názvom Rubikova kocka, ktorú dokáže poskladať za 3,25 sekund (lego.com, 2014).

Využitie minipočítaču Raspberri Pi v robotike má široké možnosti. Keď zostavíme autíčko, ktoré má na sebe mikrofón a k tomu pridáme Google technológiu na rozpoznávanie rečí dostaneme robota s možnosťou ovládania hlasom. Google Cloud Speech API umožňuje v reálnom čase prepisovať text z kamery, vykonávať hlasové príkazy a desiatky ďalších zaujímavých využití (Mach, 2016). Pre tých, ktorí majú radi Lego a pri tom chcú využívať všetky možnosti minipočítača Raspberry Pi existuje rozširujúca stavebnica BrickPi. Tento doplnok umožňuje pripájanie modulov určených pre Lego Mindstorm stavebnicu a ich programovanie pomocou Raspberri Pi (Cole, 2011). Ak máme poruke 3D tlačiareň, môžeme si vytlačiť súčiastky pre kosačku ktorú budeme ovládať pomocou mikrokontroléra Arduino Uno (Haeuser, 2016). Arduino môžeme použiť na rôzne projekty ako sú napríklad, automatické otváranie brány, dochádzkový systém, elektronický zámok s číselným heslom alebo automatizáciu domácnosti.

## 1.6 Bezdrôtové technológie

V roku 1898 Nikola Tesla predstavil diaľkové ovládanie, pomocou malej krabičky na vysielanie rádiových vĺn, dokázal manévrovať malú loď v bazéne s vodou a dokonca aj zapínať a vypínať svetielka na lodi, všetko bez akéhokoľvek káblového spojenia medzi loďou a ovládačom (Turi, 2014). Až od deväťdesiatych rokoch dvadsiateho storočia sa táto technológia stáva široko dostupná. Komponenty na diaľkové ovládanie pomocou rádiových vĺn sa stávajú cenovo dostupné a používajú sa na ovládanie menších modelov. Začiatkom 21. storočia na trh prichádzajú lacné hračky (autíčka, lode, vrtuľníky) s diaľkovým ovládaním. Postupom času sa vyvinuli ďalšie technológie na bezdrôtovú komunikáciu, ako sú WiFi a Bluetooth.

### 1.6.1 Bluetooth

Bluetooth je bezdrôtová technológia, štandard pre výmenu dát na krátke vzdialenosti. Bol vytvorený v roku 1994 spoločnosťou Ericsson ako bezdrôtová náhrada za sériové drôtové rozhranie RS232. Jednotlivé zariadenia sú identifikované pomocou svojej adresy (BD\_ADDR) podobne, ako je MAC adresa v sieti Ethernet. Technológia Bluetooth je definovaná štandardom IEEE 802.15.1. Spadá do kategórie osobných počítačových sietí, tzv. PAN. Vyskytuje sa v niekoľkých verziách, poslednou (December, 2014) je verzia 4.2, ktorou je vybavená väčšina predávaných zariadení ako sú napr. smartfóny, notebooky, tablety, ale aj televízia (diffen.com, 2013).

### 1.6.2 WiFi

WiFi je súbor štandardov pre bezdrôtové lokálne siete LAN (WLAN) v súčasnosti založených na špecifikácii IEEE 802.11. WiFi bolo navrhnuté na pripájanie bezdrôtových zariadení do lokálne siete, dnes sa najčastejšie používa na pripojenie k internetu. Wifi umožňuje osobe so zariadením s bezdrôtovým adaptérom (PC, notebook, smartfón) pripojenie k internetu v blízkosti prístupového bodu (Access point). Geografická oblasť pokrytá jedným alebo niekoľkými prístupovými bodmi sa nazýva Hotspot (diffen.com, 2013). Výhodou oproti Bluetooth je, že môžeme pripojiť viacero zariadení a taktiež vyššia prenosová rýchlosť. Nevýhodou je zložitejšie programovanie aplikácii, a väčšia spotreba v prípade napájania cez batérie.

## 1.7 Možnosti vývoja aplikácii

Mobilný operačný systém Android sa postupom času stáva stále viac populárnym medzi bežnými používateľmi po celom svete. Rovnako aj obchod s aplikáciami Google Play zaznamenáva progresívny rast počtu aplikácií a Android sa tak stáva výbornou platformou pre programovanie aplikácií. API (aplication programming interface) alebo rozhranie pre programovanie aplikácií, je v tomto prípade výborne zdokumentované a jednoduché pre použitie. Pre testovanie vytvorenej aplikácie nebudete dokonca zo začiatku potrebovať ani vlastný Android smartfón, nakoľko môžete využiť prítomnosť emulátora (Chovaňák, 2013). Máme na výber viacero možností, ako vyvíjať aplikácie pre Android (napr. Natívne a Hybridné aplikácie).

### 1.7.1 Natívne aplikácie

Vývoj v natívnom prostredí ponúka nesporné výhody pre vývojárov, ktorí sa rozhodnú zamerať na jednu platformu. Keďže zatiaľ neprišlo k zjednoteniu používateľských rozhraní medzi platformami, majú vývojári špecializujúci sa na jednu z nich výhodu pri vytváraní rozhraní, ktoré sú špecifické svojimi grafickými a ovládacími prvkami (Rudolph, 2014).

### 1.7.2 Hybridné aplikácie

Použitím jazyka HTML5 sa dajú rozšíriť možnosti webových prehliadačov. HTML5 napríklad umožňuje spracovanie precíznych vizuálnych návrhov, využívanie geolokačných funkcií, prístup k zoznamu kontaktov, senzorom a SQL databázam. Zároveň však musia vývojári bojovať s veľkou rozmanitosťou webových prehliadačov a rôznymi spôsobmi, ako tieto prehliadače implementujú špecifikácie HTML5. Výsledkom tejto fragmentácie je, že vývojári venujú neúmerne veľa času a prostriedkov na optimalizovanie svojich webových aplikácií pre rôzne platformy. Túto problematickú situáciu sa snažia riešiť tzv. hybridné aplikácie, ktoré kombinujú výhody jednoduchého vývoja webových aplikácií s jednoduchými možnosťami distribúcie a optimalizácie natívnych aplikácií. Jeden z nástrojov produkujúci hybridné aplikácie je Apache Cordova. Z pohľadu používateľa nie je na prvý pohľad rozdiel medzi hybridnými a natívnymi aplikáciami. Hybridné aplikácie sú takisto dostupné cez natívne trhoviská s mobilnými aplikáciami a inštalujú sa na zariadenia rovnako, ako natívne aplikácie. Po inštalácii sú dostupné z domovských obrazoviek (iOS, Android) a na ich spustenie nie je nutné pripojenie k internetu. Z pohľadu vývojára pripomína vývoj hybridných aplikácií proces vývoja natívnych aplikácií s jednou výnimkou, vývojári môžu použiť jazyky HTML, CSS a JavaScript na programovanie aplikácií. Model vývoja hybridných aplikácií je podporovaný všetkými veľkými platformami a umožňuje tak opakované použitie zdrojového kódu na viacerých platformách. skladajú z natívneho púzdra (tzv. wrapper), ktoré v sebe zapuzdruje aplikáciu napísanú pomocou HTML, CSS a JavaScript. Keď je takáto aplikácia spustená, wrapper vytvorí inštanciu webového prehliadača (WebView) a načíta samotnú webovú aplikáciu. Spustená inštancia nemá zobrazené typické ovládacie prvky webového prehliadača a tým umožňuje hybridnej aplikácii vyzerať podobne, ako keby bola natívna. (Rudolph, 2014)

# 2 Ciele bakalárskej práce

Cieľom bakalárskej práce je navrhnúť a skonštruovať robotické zariadenie, ktoré môžeme ovládať diaľkovo a následne naprogramovať aplikáciu pre smartfón ktorá umožní ovládanie robota cez Bluetooth.

**Medzi podciele sme zaradili:**

1. Konštrukcia robotického zariadenia.
2. Pripojenie Bluetoothu k robotickému zariadeniu.
3. Otestovať funkčnosť Bluetoothu.
4. Naprogramovať, aby robot zareagoval na určité príkazy.
5. Naprogramovať robota, aby mohol komunikovať cez Bluetoothom.
6. Vytvoriť aplikáciu pre Android, ktorá umožňuje odosielanie príkazov cez Bluetooth.
7. Prepojiť Android zariadenie s robotom cez Bluetooth a otestovať funkčnosť diaľkového ovládania.

# 3 Realizácia Ardubota

Realizácia Ardubota sa skladá z dvoch nezávislých častí, ktoré sa až na konci dávajú do hromady. V prvej časti je opísaný postup zostavenia robotického zariadenia a v druhej časti sa zaoberáme vývojom aplikácie pre Android, pomocou ktorej bude ovládané hore spomenuté robotické zariadenie.

## 3.1 Zostavenie robotického zariadenia

Na zostavenie robotického zariadenia sme použili nasledovné elektronické súčiastky:

* Arduino Nano 3.0 (ATmega 328),
* Adaptér Board for Arduino Nano V3.0,
* HC-06 Bluetooth modul,
* HG7881 Dual DC Motor Board,
* Podvozok stavebnice tanku a dva motory,
* 2 x 9V Batéria.

### 3.1.1 Arduino Nano

Arduino Nano je minimalizovaná vývojová doska s mikrokontrolérom AVR ATmega328. Funkčne je zhodná s verziou Arduino UNO, je ale prispôsobená na zasunutie do nespájkového kontaktného poľa. Piny vstupov a výstupov sú preto usadené na spodnú stranu dosky. Kvôli minimalizácii chýba konektor na externé napájanie a USB konektor je zmenšený na verziu mini USB. Samotná doska obsahuje 14 digitálnych vstupov/výstupov a 8 analógových. Niektoré z digitálnych výstupov majú možnosť generovania PWM signálu. Presné rozmiestnenie pinov na doske je znázornené na **obrázku číslo xyz**. Konektor na externé napájanie je nahradený dvojicou pinov na pripojenie 5 - 20 V. Rovnako je na doske pripojená jedna LED dióda, ktorú je možné ovládať. Ostatné voliteľné periférie je nutné pripojiť zvlášť. Výhodou zariadenia Arduino je jednoduché pripojenie k počítaču a programovanie. Arduino Nano je vybavené mini USB konektorom a po pripojení k PC sa hlási ako sériový port. Vlastné programovanie prebieha v jednoduchom prostredí Arduino IDE za pomoci jazyka odvodeného z wiring. Kód je tak veľmi prehľadný a oddeľuje programátora od zložitej konfigurácie hardvéru. (arduino.cc, 2009).

**Špecifikácie mikrokontroléra:**

* Čip: ATmega328
* Frekvencia: 16 MHz
* Pamäť: 32 kB, 2 kB použité bootloaderom
* SRAM: 2 kB
* EEPROM: 1 kB
* Rozmery: 4,32 x 1,85 cm

### 3.1.2 Motorový ovládač HG7881 (L9110S)

L9110S integrovaný obvod je kompatibilný s Arduino a Raspberry Pi. Tento obvod využíva aj HG7881. Je to kompaktný ovládač pre dva motory, ktorý podporuje napätie 2.5 až 12V na 800mA súvislého prúdu. Pretože nemá chladič, skôr sa odporúča pre menšie projekty. Výhodou tohto ovládaču je že má vstavané upínacie diódy, ktoré chránia citlivé mikrokontroléri (bananarobotics.com, 2012). Základná funkcia tohto zariadenia je otáčať motor v obidvoch smeroch a umožniť reguláciu otáčok pomocou PWM. Na jeho ovládanie sa používajú štyri piny. Schéma ovládača je znázornená na **obrázku** xx a reálny vzhľad na **obrázku xy**.

### 3.1.3 Bluetooth modul HC-06

Modul HC-06 má štyri popísané vývody vyvedené na kolíkové kontakty   
(Rx, Tx, GND, Vcc). **(obrázok**)Napájacie napätie modulu môže byt v rozmedzí 3.3 až 5V, úrovne napätia vyššie ako 7V modul zničia. Úroveň vstupného a výstupného signálu je 3.2V, pri spolupráci s mikrokontrolérom napájaným napätím 5V treba úrovne prispôsobiť. LED na module signalizuje stav pripojenia. Ak bliká, nie je nadviazané spojenie a modul je možne cez vývody Tx / Rx programovať. Ak LED svieti, je nadviazané spojenie a všetko zo vstupu sa považuje za prenášané dáta. Prenosová rýchlosť je defaultne nastavená na 9600 bit za sekundu, tú však môžeme zmeniť. (Černý, 2014)

### 3.1.4 Práca na robotickom zariadení

Ako základnú časť stavebnice sme použili podvozok s dvoma motormi. Je to časť stavebnice, ktorú je možné použiť ľubovoľne ako k Arduino tak i Raspberry Pi. Ako vidno na **obrázku** číslo X na podvozku na nachádzali iba dva motory. Začali sme pridaním Arduina na podvozok ktorí predstavuje základnú riadiacu jednotku. Arduino ponúka širokú škálu dosiek, rozhodli sme sa použiť Arduino Nano 3.0. Ten sme umiestnili do Adaptéra ktorí umožňuje jednoduchšie prepájanie Arduina s externými zariadeniami. (Na **obrázku číslo** x môžete vidieť Arduino Nano s Adaptérom.) Na ovládanie spomenutých motorov sme použili integrovaný obvod HG7881. Ďalším krokom bolo prepojenie mikrokontroléra s ovládačom, pridanie batérie a následné nahranie testovacieho programu. Testovali sme správnosť pripojenia obvodov a funkčnosť motorov. Ovládač HG7881 má na sebe dva napájacie a štyri ovládacie piny. Ovládacie piny sme pripojili na PWM vývody Arduina čo nám umožní kontrolovať rýchlosť otáčania motora. Na ovládanie každého z motorov sú určené dva piny, v závislosti od toho na ktorý pin privedieme napätie bude meniť smer otáčania motoru, ak v rovnaký čas privedieme napätie na oba piny motor bude zablokovaný. Pre vývoj softvéru sme použili grafické prostredia Arduino IDE. Po spustení aplikácie sa otvorí prázdny projekt, do ktorého môžeme písať vlastný program**. (obrázok).** Dosky Arduino sú známe po tom že nepotrebujú dodatočný programátor na ich programovanie, stačí prepojiť dosku z počítačom cez USB kábel a v počítači mať inštalovaný Arduino IDE ktorý sa postará o všetko ostatné. Na to aby sme úspešne naprogramovali dosku potrebné je aby sme vo vývojovom prostredí mali nastavené správne parametre. Zvolíme si dosku ktorú používame a port na ktorom je doska pripojená. Kompilovaním program prispôsobíme aby fungoval na zvolenej doske. To umožňuje rovnaký program nahrávať na rôzne typy Arduino dosiek. Pri testovaní sme zistili že ak zmenšíme výstupné napätie na ovládacích pinoch o polovicu motor sa prestane točiť, vyriešili sme to stabilizáciou PWM signálu pridaním kondenzátorov na motory. Ďalším krokom bolo navrhnúť a naprogramovať pohyb robota. Robot mal zareagovať na vstup zo sériového portu. Inštrukcie pre robota sú písmenká ktoré predstavujú smer pohybu.

* f - Dopredu,
* b - Dozadu,
* l - Odbočiť vľavo,
* r - Odbočiť vpravo,
* q - Otočiť sa vľavo,
* e - Otočiť sa vpravo,
* s - Zastaviť

Na správne fungovanie programu musíme mať pridané dve metódy a to *setup()* a *loop().* Metóda setup sa súšťa ako prvá a v nej sa definujú veci ako Parametre pripojenia na sérový port *napr. Serial.begin(9600)* a vstupné a výstupné piny. Napríklad ak chceme aby bol pin 10 výstupný použijeme metódu *pinMode(10,OUTPUT)*. Samotný program sa píše v metóde loop čo znamená opakovať. Namiesto písania konkrétneho čísla do metódy môžeme použiť premenné do ktorých uložíme čísla. Ak chceme použiť iní pin stačí na jednom mieste zmeniť číslo pinu. Sérovú komunikáciu umožňuje knižnica Serial, tá ponúka metódu *Serial.read()* ktorú sme použili na čítanie zo vstupu sériového portu. Údaje zo vstupu sme uchovali do premennej *command* aby sme ich neskôr mohli použiť. Obsah premennej *command* sme vložili do príkaz *switch*. Tento príkaz na základe vstupu vykoná rôzne príkaz, v našom prípade pohyb robota. Robota ovládame privádzaním napätia na výstupné piny Arduina. To môžeme spraviť dvoma spôsobmi a to metódami *analogWrite()* a *digitalWrite()*. Ako prvý parameter vnášame pin na korí chceme priviesť napätie a druhý parameter je hodnota. V prípade digitalWrite je to HIGH (logická jednotka) a LOW (logická nula) a na výstupe máme napätie buď 0 alebo 5V. Metóda *analogWrite()* sa môže používať iba na pinoch ktoré majú PWM rozhranie a ako jej druhý parameter vnášame hodnotu od 0 do 255 čo reprezentuje napätie na výstupe kde 0 predstavuje 0V a 255 maximálnych 5V. Pri pohybe robota dopredu a dozadu na výstupných pinoch bolo 5V, tak že robot šiel maximálnou rýchlosťou. Zatáčanie tanku je vyriešene tak že rýchlosť jedného z motorov sme zmenšili o 40%. Tank má možnosť sa otočiť na mieste otáčaním motorov v opačnom smere. Pri maximálnej hodnote sa tank príliš rýchlo otáčal, kvôli tomu bola rýchlosť motorov obmedzená na 60%. Keď že Arduino IDE ponúka monitorovanie sériovej linky mohli sme si to razom aj odskúšať. Počas testovania Arduino ako napájací zdroj používal USB kábel a motory 9V batériu. Na ovládanie robota bezdrôtovo bolo treba vyriešiť určité technické záležitosti a to zdroj napätia pre Arduino a pridanie bezdrôtového modulu. Prvá možnosť pripojiť Arduino a motory na 9V batériu sa ukázala ako nie celkom správna, motory spôsobili pád napätia a Arduino sa vypol. Pridaním druhej 9V batérie sa tento problém odstránil. Na bezdrôtové ovládanie sme mali na výber WiFi a Bluetooth. Rozhodli sme sa pre Bluetooth technológiu, preto že na komunikáciu s Arduinom používa sériovú linku tak že nebolo nutné meniť program a má aj nižší odber oproti WiFi čo je dôležité keď že nám je zdroj napätia batéria. Bluetooth modul ktorý sme použili je HC-06, má dosah 75 metrov čo je v tomto prípade viac ako postačujúce. HC-06 má na sebe 4 vývody, dva pre napájanie a dva pre dátovú komunikáciu. Na vytvaranie schéme pripojenia sme používali softvér pod nazvom fritzing. Je to open-source program ktorý umožnuje návh elektronických obvodov a zároveň vytvorý návrh plošného spoju dosky. Schéma pripojenia jednotlivých súčiastok robota sa nachádza na **obrázku** xx. Práca na robotickom zariadení bola dokončená a pristúpili sme k vývoju aplikácie Android na ovládanie robota.

## 3.2 Vývoj aplikácie na ovládanie zariadenia

Aplikáciu na ovládanie robotického zariadenia cez Bluetooth pre Android platformu sme vyvíjali pomocou frameworku Apache Cordova. Tento spôsob programovania umožňuje používať webové technológie na vývoj aplikácii pre smartfóny. Framework Apache Cordova sa skladá z webovej a natívnej časti. Natívna časť pre Android platformu je neprogramovaná v jazyku Java. Na vývoj webovej časti sa používajú HTML, CSS a JavaScript. K dispozícii máme knižnicu Cordova.js ktorá poskytuje funkcionalitu pre volanie natívnych častí frameworku. Jednotlivé funkcionality sú zapuzdrené do zásuvný modulov tzv. pluginov. Pluginy umožňujú webovej aplikácie pristupovať k hardvérovým častiam smartfónu ako napríklad Kamera, Bluetooth, WiFi, NFC, GPS alebo čítanie a zápis do pamäte. Na vývoj webových aplikácii sa často používajú knižnice ktoré urýchľujú a uľahčujú písanie kódu, zároveň je kód krajší a prehľadnejší.

### 3.2.1 Inštalácie a nastavenie Apache Cordove

Skôr ako začneme programovať aplikáciu musíme ju nainštalovať. Na oficiálnej stránke Apache Cordova projektu sa nachádza návod na inštaláciu. Treba pripomenúť že inštalácia neprebieha spôsobom na ktorý sú väčšina ľudí zvyknutý tak že na stránke nenájdete žiadne tlačidlo Download. Inštaluje sa cez príkazový riadok pomocou správcu balíčkov npm. Správca balíčkov je nástroj ktorým inštalujeme rôzne aplikácie cez príkazový riadok bez potrebe navštíviť stránku a stiahnuť konkrétnu aplikáciu, on to všetko urobí pre nás. Konkrétne správca balíčkov npm sa primárne používa na inštaláciu JavaScriptových technológii. Inštaláciou NodeJS do počítaču sa inštaluje aj správca balíčkov npm. Keď máme inštalovaný npm jednoduchým príkazom *npm install -g cordova* nainštalujeme Apache Cordovu do počítača. Príkazom *cordova create MyApp* vytvoríme projekt s ukážkovou aplikáciou. Ďalším krokom je pridať platformu Android do projektu príkazom *corodva platform add android*. V prípade že máte na svojom počítači Android SDK príkazom *cordova emulate android* sa aplikácia spustí v emulátory na vašom počítači.

### 3.2.2 Vývojové prestretie

Keďže vývoj hybridnej mobilnej aplikácie prebieha veľmi podobne vývoj webovej aplikácie, je možné použiť akékoľvek vývojové prostredie ktoré umožňuje tvorbu webových aplikácii. Medzi populárne editory patria: Sublime, Atom, WebStrom, NetBeans, VisualStudio. Sublime a Atom sú malé flexibilne a modulárne odporúčajú sa na menšie projekty. WebStrom, NetBeans a Visual Studio sú robustnejšie vývojové prostredia pre väčšie projekty a spoločné majú že podporujú kompilovanie Cordova aplikácii priamo v samotnom programe. Na vývoj aplikácie sme použili Atom ktorý prichádza od strane GitHubu, tak že od začiatku mal stavanú podporu verzionovacieho systému git. Okrem toho podporuje aj doplňovanie kódu zvýrazňovanie syntaxe a môžeme ho rozšíriť od ďalšie funkcionality pridávaním balíčkov. Náhľad vývojového prostredia je na **obrázku xx**.

### 3.2.3 Použité knižnice

* jQuery je ľahká cross-browser JavaScript knižnica, ktorá kladie dôraz na interakciu medzi JavaScriptom a HTML. jQuery syntax je navrhnutá pre jednoduchšiu navigáciu dokumentu, výber DOM elementov, vytváranie animácií, spracovanie udalostí, a vývoj Ajax aplikácií. jQuery tiež poskytuje možnosti pre vývojárov na vytváranie pluginov postavených na tejto JavaScript knižnici.
* jsRender je šablónovací nástroj ktorý nám pomáha pridávať údaje do šablóny na miesta ktoré sme určili. Ak máme zoznam produktov ktoré načítame z databáze, všetky ich zobrazíme na obrazovke použitím jednej šablóny. Renderovanie sa používa pri vývoji OnePage aplikácii, čiže nemusíme sa prekliknúť na ďaľšiu stránku aby sme videli iný obsah.
* Font Awesome nám prináša škálovateľné vektorové ikonky, ktoré môžu byť okamžite prispôsobené, zmeníme čokoľvek, čo môže byť vykonané s výkonom CSS. K dispozícii máme stovky ikoniek rôznych druhov ktoré postačia na väčšinu projektov ktoré vyvíjame.

### 3.2.4 Git a GitHub

Git verzionovací systém zaznamenáva zmeny v súboroch. Umožňuje vrátiť sa k ľubovoľnej verzie súboru alebo celého projektu. Ponúka nám možnosť porovnávania rôznych verzii projektu alebo obnoviť poškodené či stratené súbory. Môžeme pracovať offline, a online. Jednoduchá spolupráca na rovnakom projekte. Všetky zmeny ktoré spravíme môžeme prehľadávať a vrátiť späť. GitHub predstavuje online úložisko pre git projekty. Aby sme ho používali musíme mať vytvorený účet a v ňom repozitár ktorí cez git napojíme na nás projekt. V prípade že potrebujeme môžem pridať vetvu projektu ktorá sa bude vyvíjať nezávisle od hlavného projektu a následne ju pripojiť k hlavnej vetve. Git je dostupný pre Widnows, Linux a MAC OS X operačné systémy. Prevažne sa používa cez príkazový riadok ale sú k dispozícii aj verzie s grafickým rozhraním.

### 3.2.5 Tvorba aplikácie

Cez príkazový riadok sme vytvorili cordova projekt a pridali Android platformu. Otvorili sme projekt editorom Atom a začali so základnými úpravami. V súbore *config.xml* sa nachádzajú základné informácie o aplikácii: meno aplikácie, verzia, krátky opis, autor a iné. V priečinku www sa nachádzajú všetky HTML, CSS a JavaScript súbory ktoré potrebujeme na vývoj aplikácie. Ukážkovú aplikáciu odstránime z projektu a pridáme potrebné balíčky a knižnice. Ďalší správca balíčkov ktorý používame sa menuje bower. Používa sa prevažne na inštaláciu technológii ako sú angular, bootstrap, jquery a podobné. Cez bower sme inštalovali jquery, jsrender.js a fontawesome. Aby sme aplikácii umožnili komunikovať s Bluetoth pridali sme dva cordova pluginy. Pluginy sa pridávajú príkazom v komandovom riadku napr. *cordova plugin add cordova-plugin-bluetooth-serial.* Bluetooth Serial je plugin ktorý umožňuje aby sme sa pripojili k Bluetooth zariadeniu, posielali údaje, čítali údaje a po skončení komunikácie odpojili. Ďalší plugin Bluetooth Status je doplnkový, on sleduje stav Bluetoothu a v prípade že je vypnutý týmto pluginom ho môžeme zapnúť.

Zmeny počas programovania sme sledovali verzionovacím systém git. Na stránke GitHub som si vytvoril repozitár ktorý som následne prepojil s projektom v počítači. Školiteľ tak mohol sledovať v akom štádiu je vývoj aplikácie keď pristúpil na daný repozitár. Súbor ktorý sa ako spustí ako prvý je index.html, v ňom nalinkujeme všetky JavaScriptové a CSS súbory ktoré budeme používať, lebo ich program nebude vidieť. CSS štýly sa pridávajú iba do hlavičke HTML dokumentu. JavaScriptové súbory môžeme pridávať buď do hlavičky alebo na koniec tela HTML dokumentu. Samotný program aplikácie sa nachádza v súbore index.js. Keď spustíme aplikáciu potrebný je určitý čas pokiaľ sa načítajú všetky súbory a pluginy. Na začiatok index.js súboru pridáme príkaz *document.addEventListener("deviceready", onDeviceReady, false);* ktorý spustí metódu *onDeviceReady()* až keď bude zariadenie pripravené na spustenie aplikácie. Po spustení aplikácie sa spustia dva Event Listener-y, ktoré zabezpečia ako má aplikácia zareagovať ak ju necháme bežať na pozadí a keď ju vrátime do popredia. Napríklad ak hráme hru a niekto nám zavolá, aplikácia sa pozastaví a po skončení hovoru môžeme pokračovať. Pluginy ktoré sme pridávali musíme na nejakým spôsobom spustiť a to príkazom *cordova.plugins.BluetoothStatus.initPlugin().* Spustením Bluetooth Status pluginu načítame informácie o stave Bluetooth v smartfóne. Ďalším príkazom *cordova.plugins.BluetoothStatus.enableBT();* zapneme Bluetooth aby sme ho mohli používať v aplikácii. Spustenie Bluetoothu trvá nejaký čas, tak že musíme počkať pokým sa Bluetooth nezapne. Ako už bolo spomenuté keď čakáme na udalosť spustíme Event Listener príkazom *window.addEventListener('BluetoothStatus.enabled', function(){});* Keď sa Bloetooth zapne spustí sa vnorená anonymná metóda. Druhý pridaný plugin BluetoothSerial okrem iného má možnosť načítať zoznam Bluetooth zariadení uchovaných v smartfóne. Zoznam je štruktúrovaný ako JSON súbor, aby sme použili daný súbor existujú dve možnosti:

JQuery metódou konvertovať JSON na reťazec a následne prechádzaním cez reťazec načítavať údaje do premenných a pridať ich do súboru *index.html* pre zobrazenie na obrazovke.

Lepším spôsobom bolo použitie šablóny a do nej pridávať hodnoty z JSON súboru tzv. renderovanie. To je spôsob pridávania údajov do predpripravenej šablóny na zadefinované miesta. Existuje mnoho knižníc na renderovanie, najznámejšie sú Angular, React a JsRender. V programe som použil JsRender knižnicu, síce má menej možnosti ako Angular ale je postačujúca pre náš program. Pri tvorení šablóny najskôr napíšeme HTML kód a na miesta kde sa text bude meniť pridáme zápis v tvare *{{:id}}* kde id predstavuje udaj z JSON-u. Metóda bluetoothSerial.list(function(devices) {}) vracia zoznam zariadení v tvare: [{

"class": 276,

"id": "10:BF:48:CB:00:00",

"address": "10:BF:48:CB:00:00",

"name": "Nexus 7"

}, {

"class": 7936,

"id": "00:06:66:4D:00:00",

"address": "00:06:66:4D:00:00",

"name": "HC-06"

}]

Zariadenie je predstavené ako JSON súbor v ktorom sa údaje uchovávajú v tvare "nazov\_premennej":"hodnota" jednotlivé zápisy sú dodelené čiarkou. Šablóna pre výpis zariadení bola nasledovná:

*<li><a class="connectBT" href="{{:id}}">{{:name}} | {{:id}}</a></li>*

Príkazom *var tmpl = $.templates()* šablónu zmeníme na tvar ktorý JsRender môže používať. Keď máme šablónu a údaje prichystané môžeme spustiť renderovanie príkazom *var html= tmpl.render(devices) kde devices predstavuje retazec a vysledný kód sa zapisuje do premennej pod názvom html.* V *index.html* je pripravený prázdny zoznam ktorý ma id s hodnotou „ulBtList“. Na to miesto môžeme pridať kód generovaný JsRenderom a to pomocou jQuery. Príkazom *$('#ulBtList').html(html)* vnútro zoznamu nahradíme našim html kódom čo spôsobí že sa zoznam zariadení okamžite zobrazí na obrazovke. (**obrazok** ) Stlačením na odkaz sa má aplikácia pokúsiť o nadviazanie s daným zariadením. Problém bol v tom že dynamický pridávané elementy nemajú pridaný Event Listener, takže nie je možné odchytiť či sme stlačili na odkaz alebo nie. Tak že sme sledovali či sme stlačili na obrazovku, ak áno tak skontrolujeme či národov to nie je odkaz na pripojenie.

$(document).on('touchstart','.connectBT',function(){

event.preventDefault();

var bladr = $(this).attr('href');

bluetoothSerial.connect(bladr, function () {

$('#btControl').load("temp.html");

$('#btList').hide();

}, function () {

alert("Pripojenie zlyhalo");

window.location.reload(true);

var bladr="";

});

})

Po stlačení na odkaz sa spustí tento kus kódu, zistíme na ktoré konkrétne zariadenie je stlačené. Bluetooth adresu ktorá sa nachádza v href načítame do premennej. Metódou *connect()* spustíme pripájanie k zariadeniu. Ak sa po určitom čase aplikácii nepodarilo pripojiť k zariadeniu spustí sa oznámenie pre používateľa že sa nepodarilo pripojiť a zoznam zariadení sa opätovne načíta. Pri úspešnom pripojení k zariadeniu sa zoznam skryje a pridá sa nová stránka ktorá je uložená v súbore *temp.html.* Nová stránka obsahuje meno projektu, šesť ovládacích tlačidiel a tlačidlo na odpojenie. (**obrazok**).

*<button class="btn" id= "cw"><i class="fa fa-repeat"></i></button>*

Každé tlačidlo obsahovalo klas btn a jedinečné id po ktorom ho rozoznávame. Namiesto názvu tlačidla je pridaná ikonka z Font Awesome.

Tlačidla boli pridané ale ich funkcionalita ešte nebola definovaná v JavaSkripte.

$(document).on('touchstart', '#cw, function() {

bluetoothSerial.write("q", function(){}, function(){});

});

$(document).on('touchend', '.btn', function() {

bluetoothSerial.write("s", function(){}, function(){});

});

Keď sa dotkneme tlačidla spustí kód ako na príklade a začne s odosielaním inštrukcii cez Bluetooth. Inštrukcie sa posielajú pokiaľ je tlačidlo stlačené, ako náhle prestaneme držať tak sa spustí metóda ktorý pošle príkaz na zastavenie. Rozdiel je to tom že druhý príkaz je univerzálny a spustí sa pre každé tlačidlo, tak že ho nemusíme písať šesť krát ale stačí iba raz. Údaje cez Bluetooth sa odosielajú pomocou metódy write(). Ako prvá hodnota sa uvádza hodnota ktorú posielame, za ňou nasledujú dve vnorené metódy ak chceme zareagovať na úspešné alebo neúspešné odosielanie údajov. Keď skončíme s ovládaním alebo sa chceme pripojiť na iné Bluetooth zariadenie stačí keď sa odpojíme strašením tlačidla na spodku aplikácie.

$(document).on('touchend', '#disconnect', function() {

bluetoothSerial.disconnect(function () {

$('#btControl').html("");

$('#btList').show();

alert('Úspešne ste sa odpojili');

window.location.reload(true);

}, function () {});

});

Po ukončení spojenia ovládacie tlačidla zmiznú a načíta sa zoznam Bluetooth zariadení.

Aplikáciu môžeme používať rovnako, ako po prvom spustení.

# Zoznam bibliografických odkazov

arduino.cc, 2009. *Arduino Nano.* [Online]   
Available at: https://www.arduino.cc/en/Main/ArduinoBoardNano  
[Cit. 5 December 2015].

arduino.cc, 2013. *What is Arduino?.* [Online]   
Available at: https://www.arduino.cc/en/Guide/Introduction  
[Cit. 10 September 2015].

bananarobotics.com, 2012. *How to use the HG7881 (L9110) Dual Channel Motor Driver Module.* [Online]   
Available at: https://www.bananarobotics.com/shop/How-to-use-the-HG7881-(L9110)-Dual-Channel-Motor-Driver-Module  
[Cit. 29 November 2015].

Brian, B., 2007. Maximum LEGO NXT: Building Robots with Java Brains Variant Press.. In: s.l.:s.n.

Cole, J., 2011. *If you like LEGO, you'll love the BrickPi.* [Online]   
Available at: http://www.dexterindustries.com/brickpi/  
[Cit. 12 Február 2016].

Černý, M., 2014. *Ovládání pomocí Bluetooth – pro Arduino.* [Online]   
Available at: http://robodoupe.cz/2014/ovladani-pomoci-bluetooth-pro-arduino/  
[Cit. 14 Jánuár 2016].

diffen.com, 2013. *Bluetooth vs. Wifi.* [Online]   
Available at: http://www.diffen.com/difference/Bluetooth\_vs\_Wifi  
[Cit. 7 Február 2016].

Haeuser, A., 2016. *Ardumower Arduino Controlled 3D Printable Lawn Mover.* [Online]   
Available at: http://diy3dprinting.blogspot.sk/2016/02/ardumower-arduino-controlled-3d.html  
[Cit. 5 Marec 2016].

Chovaňák, M., 2013. *Programovanie Android aplikácií pre začiatočníkov.* [Online]   
Available at: https://www.mojandroid.sk/programovanie-android-aplikacii-pre-zaciatocnikov-1-cast/  
[Cit. 20 Marec 2016].

lego.com, 2014. *ROBOT SOLVES RUBIK’S CUBE IN 3 SECONDS.* [Online]   
Available at: http://www.lego.com/en-us/mindstorms/news/2014/march/cubic-stormer  
[Cit. 14 November 2015].

Mach, S., 2016. *Rozoznávanie reči od Googlu už môžeš využívať aj ty!.* [Online]   
Available at: http://www.startitup.sk/rozoznavanie-reci-googlu-uz-mozes-vyuzivat-aj/  
[Cit. 26 Marec 2016].

merkurtoys.cz, 2012. *MERKUR Robotika a mechatronika.* [Online]   
Available at: http://www.merkurtoys.cz/vyrobky/robotika-a-mechatronika  
[Cit. 17 Október 2015].

raspberrypi.org, 2014. *WHAT IS A RASPBERRY PI?.* [Online]   
Available at: https://www.raspberrypi.org/help/what-is-a-raspberry-pi/  
[Cit. 5 Október 2015].

raspberrypi.org, 2016. *Raspberry Pi 3 Model B.* [Online]   
Available at: https://www.raspberrypi.org/products/raspberry-pi-3-model-b/  
[Cit. 2 Marec 2016].

Rudolph, P., 2014. *Hybrid Mobile Apps: Providing A Native Experience With Web Technologies.* [Online]   
Available at: https://www.smashingmagazine.com/2014/10/providing-a-native-experience-with-web-technologies/  
[Cit. 11 Marec 2016].

RUS, D., 2006. Teaching robotics everywhere. IEEE Robotics and Automation Magazine.

Turi, J., 2014. *Tesla's toy boat: A drone begore it's time..* [Online]   
Available at: http://www.engadget.com/2014/01/19/nikola-teslas-remote-control-boat  
[Cit. 16 Jul 2015].

<http://i2.wp.com/learnelectronicsblog.com/wp-content/uploads/2014/08/Arduino-Boards.png> - Zdroj obrazka pridat