MASARYKOVA UNIVERZITA
FAKULTA INFORMATIKY

Automatizovaný zavlažovací systém

DIPLOMOVÁ PRÁCA

Bc. Milan Strnad

Brno, jar 2019

This is where a copy of the official signed thesis assignment and a copy of the Statement of an Author is located in the printed version of the document.

Prehlásenie

Prehlasujem, že táto diplomová práca je mojím pôvodným autorským dielom, ktoré som vypracoval samostatne. Všetky zdroje, pramene a literatúru, ktoré som pri vypracovaní používal alebo z nich čerpal, v práci riadne citujem s uvedením úplného odkazu na príslušný zdroj.

Bc. Milan Strnad

Vedúci práce: RNDr. Zdeněk Matěj, Ph.D.

Poďakovanie

Rád by som poďakoval RNDr. Zdeňkovi Matějovi, Ph.D., za vedenie mojej záverečnej práce, užitočné rady a ústretový prístup. Ďakujem tiež svojej rodine a priateľom za podporu a ohľaduplnosť po celú dobu práce na tomto projekte.

Abstrakt

Táto diplomová práca sa zaoberá analýzou súčasných automatizovaných zavlažovacích systémov pre izbové rastliny, identifikovaním ich nedostatkov a návrhom a vypracovaním riešenia nového, ktoré tieto nedostatky rieši. Výsledkom je kompaktný prototyp, ovládateľný pomocou iOS mobilnej aplikácie, ktorá umožňuje správu viacerých takýchto zariadení súčasne. Prototyp je zasadený do púzdra v tvare kocky, ktoré bolo zhotovené technológiou 3D tlače. Projekt nesie pracovný názov WaterCubes.

Kľúčové slová

automatizácia, smarthome, zavlažovací systém, izbové rastliny, Raspberry Pi, Python, mobilná aplikácia, iOS, Swift, Xcode, 3D tlač, dizajn

Obsah

1 Úvod **1**
2 Analýza **3**
2.1 Súčasné riešenia 3
2.1.1 Systém 1: Gardena 3
2.1.2 Systém 2: Plantui 4
2.1.3 Systém 3: Sense 5
2.2 Zhrnutie 5
3 Návrh **7**
3.1 Návrh zariadenia 7
3.1.1 Funkcionalita zariadenia 7
3.1.2 Použité technológie 8
3.2 Návrh mobilnej aplikácie 11
3.2.1 Funkcionalita mobilnej aplikácie 11
3.2.2 Užívateľské rozhranie 13
3.3 Návrh púzdra 16
3.3.1 Modelovanie 17
4 Realizácia **20**
4.1 Zapojenie komponent 20
4.2 Vývoj zariadenia 23
4.2.1 Pripojenie na internet 23
4.2.2 Vzdialený prístup 23
4.2.3 Programovací jazyk Python 24
4.2.4 Implementácia modulov 25
4.2.5 Manipulácia s dátami 25
4.2.6 Server 26
4.2.7 Bootovanie 28
4.3 Vývoj mobilnej aplikácie 28
4.3.1 Programovací jazyk Swift 28
4.3.2 Architektúra projektu 29
4.3.3 Použité knižnice 29
4.3.4 Implementovanie harmonogramov 30
4.4 Zhotovenie púzdra 30
4.4.1 Technológia 3D tlače 31
4.4.2 Produkcia 31

4.5 Kalibrácia 32
5 Plány do budúcnosti 34
5.1 Vylepšenia zariadenia 34
5.2 Vylepšenia mobilnej aplikácie 34
5.3 Vylepšenia púzdra 35
6 Záver 36
Literatúra 37
A Príloha – zdrojové kódy 39

Zoznam obrázkov

2.1 Systém 1: Gardena 3
2.2 Systém 1: Gardena – detail 3
2.3 Systém 2: Plantui 4
2.4 Systém 2: Plantui – detail 4
2.5 Systém 3: Sense 5
3.1 Use case diagram – funkcionalita zariadenia 7
3.2 Raspberry Pi 3A+ 9
3.3 Mini čerpadlo 9
3.4 Nízkonapäťový relé modul 10
3.5 Ultrazvukový senzor 10
3.6 Rezistor – 330 ohmov 10
3.7 Rezistor – 470 ohmov 10
3.8 LED – modrá 11
3.9 LED – červená 11
3.10 Use case diagram – funkcionalita aplikácie 12
3.11 UI – zoznam zariadení (prázdny) 13
3.12 UI – zoznam zariadení 13
3.13 UI – pridať zariadenie 14
3.14 UI – inicializácia 14
3.15 UI – detail zariadenia 15
3.16 UI – editácia zariadenia 15
3.17 Modelovací nástroj Tinkercad 17
3.18 Konektor 18
3.19 3D model púzdra – vrchná časť 18
3.20 3D model púzdra – spodná časť 19
4.1 Schéma pinov Raspberry Pi B+ 20
4.2 Obvod: RPi + relé modul + čerpadlo 21
4.3 Obvod: RPi + ultrazvukový senzor 22
4.4 Obvod: RPi + 2x LED 22
4.5 Model MVVM+C architektúry 29
4.6 Výsledný prototyp 32
4.7 Kalibrácia 33
6.1 Zavlažovací systém WaterCubes 36

1 Úvod

Automatizácia je trendom súčasnosti. Ide o etapu rozvoja techniky, ktorá nástupom informačných technológií nadobudla modernejšiu formu už v 50. rokoch 20. storočia. [1] Vďaka neustálemu vedecko-technickému pokroku pokračuje dodnes a najväčší rozmach zažíva práve teraz.

Snahu o automatizáciu je možné vidieť takmer všade. Dobrým príkladom je priemysel, v ktorom sa automatické výrobné linky a roboty stali neodmysliteľnou súčasťou výrobných procesov, doprava, kde dnes nie sú netradičné už ani autonómne vozidlá ako napríklad Tesla, alebo automatizácia budov, u ktorých je možné automaticky riadiť činnosti ako vykurovanie, klimatizáciu alebo vetranie, a to vo veľkých merítkach.

Automatizácii sa však nevyhli ani bežné domácnosti a pojem smarthome sa dostáva do čoraz väčšieho povedomia verejnosti. Súčasný trh ponúka stovky, možno až tisícky najrôznejších smart zariadení, ktoré po ich iniciálnom nastavení nevyžadujú žiadnu, nanajvýš len občasnú interakciu s užívateľom a je možné ich vzdialene ovládať prostredníctvom diaľkových ovládačov a mobilných aplikácií.

Medzi tie najbežnejšie a najznámejšie prvky patria napríklad inteligentné elektrické zásuvky a spínače, ktoré aj obyčajný spotrebič dokážu v zápätí premeniť na smart, inteligentné osvetlenie, meteostanice, termostaty, detektory vlhkosti, oxidu uhličitého, dymu, vody a vibrácií, alarmy a senzory ako napríklad senzor pohybu, infračervený senzor prítomnosti alebo magnetické senzory otvorenia okien a dverí. Za zmienku stojí aj automatické otváranie brán a garážových dverí.

Niektoré z nich na svoje fungovanie potrebujú centrálnu jednotku, tzv. bránu, ku ktorej sa pripájajú a komunikujú s ňou prostredníctvom protokolu Zigbee, Z-Wave alebo Apple HomeKit. Vďaka tejto bráne je však zvyčajne možné zariadenia kombinovať a vytvárať tak rôzne scénáre, ktoré sa môžu automaticky aktivovať alebo deaktivovať v situáciách, ktoré užívateľ predom definuje.

Možný je potom napríklad aj nasledujúci scénár: užívateľ dorazí domov, automaticky sa zapne osvetlenie, zapne sa vykurovanie alebo klimatizácia, zapne sa televízia alebo rádio, a ak sa v niektorej z miestností nezdržuje, spotrebiče v nej sa vypnú. Keď ide spať, vypnú sa všetky nepotrebné zariadenia, vypnú sa svietidlá, zatiahnu sa žalúzie, zníži sa teplota a zapne sa alarm. Ráno, keď sa bude prebúdzať, sa postupne otvoria žalúzie, zvýši sa teplota alebo zapne klimatizácia, zapne sa rádio a vypne sa alarm. Podobných scénárov je možné vytvoriť neobmedzené množstvo a všetko závisí výhradne na preferenciách užívateľa.

Dnes už existuje celý rad spoločností, ktoré sa touto problematikou zaoberajú a je možné si u nich zakúpiť kompletné riešenie plne automatizovanej

domácnosti, prípadne do domácnosti integrovať len niektoré z vybraných funkcionalít.

Trh s inteligentnými zariadeniami otvára úplné nové možnosti a má nepochybne veľký potenciál do budúcnosti. V blízkej dobe budú takmer s určitosťou vznikať stále ďalšie, modernejšie a inteligentnejšie zariadenia, ktorých hlavnými cieľmi búdú vždy: zvýšenie bezpečnosti, úspora energie a užívateľský komfort.

Jednou z oblastí týkajucou sa pohodlia užívateľov, ktorá sa zdá sa byť nepokrytá úplne dostatočne, je starostlivosť o izbové rastliny, ktoré sa bežne v domácnostiach nachádzajú. Trh síce ponúka niekoľko riešení, ktoré s týmto problémom súvisia, ale nie všetky ho adresujú priamo a nedá sa povedať, že by šlo o univerzálne riešenia bez priestoru na vylepšenie.

A práve to je cieľom tejto diplomovej práce – vytvorenie automatického zavlažovacieho systému, ktorý bude univerzálne použiteľný pre akýkoľvek druh izbovej rastliny, a ovládateľný pomocou mobilnej aplikácie, vďaka ktorej bude možné riadiť a monitorovať viacero takýchto zariadení súčasne. Dôležitými atribútmi projektu pritom budú kompaktnosť, praktickosť a dizajn.

2 Analýza

Táto kapitola je venovaná analýze už existujúcich automatických zavlažovacích systémov pre izbové rastliny alebo zavlažovacích systémov, ktoré s touto problematikou aspoň okrajovo súvisia a je možné ich zhodnotiť, prípadne sa z nich v nejakom zmysle inšpirovať.

2.1 Súčasné riešenia

Nasledujúci text popisuje tri nazaujímavejšie systémy zaoberajúce sa starostlivosťou o rastliny, ktoré bolo možné nájsť pomocou týchto kľúčových slov a ich kombinácií: automatický, automatizovaný, inteligentný, zavlažovací systém, izbové rastliny, automated, smart, irrigation system, indoor plants, watering system.

2.1.1 Systém 1: Gardena

Prvým zaujímavým systémom je plne automatizovaný zavlažovací systém od spoločnosti Gardena, ktorá sa špecializuje na kompletnú starostlivosť o záhradu. Tento systém je možné použiť v exteriéri aj interiéri a vďaka patentovanej spojovacej technike je možné ním zalievať až 36 kvetináčov súčasne. Zavlažovanie sa aktivuje raz denne a to na 1 minútu, pričom intenzitu závlahy je možné nastaviť individuálne v každom vývode. Systém má zásobník vody o objeme 9 litrov a v ňom umiestnené nízkonapäťové čerpadlo s filtrom nečistôt. [2]

Obrázok 2.1: Systém 1: Gardena

Obrázok 2.2: Systém 1: Gardena – detail

Najväčšou výhodou tohoto systému je jeho modularita a schopnosť obsluhovať viacero rastlín naraz. V ideálnom prípade by však všetky rastliny, ktoré chcú byť súčasťou zálievky, mali byť blízko seba, čo výrazne komplikuje bežné inetriérové použitie v domácnosti, v ktorej sa kvetináče nachádzajú na rôznych miestach bytu alebo domu. Zariadenie taktiež nie je schopné užívateľa informovať o nízkej hladine vody v zásobníku. Po bližšom preskúmaní tohoto systému je jasné, že je určený najmä pre exteriér, balkóny a situácie dlhodobej neprítomnosti.

2.1.2 Systém 2: Plantui

Ďalším zaujímavým systémom je inteligentný kvetináč Plantui Smart Garden, ktorý je vybavený svetelným LED systémom a automatickým zavlažovacím čerpadlom, ktoré by malo zaistiť ideálne podmienky pre celoročné pestovanie byliniek a rastlín v interiéri. Najprv sa do kvetináča umiestnia špeciálne kapsule so sadeničkami (maximálne 6) a do zásobníka s kapacitou 3 litre sa naleje voda. Kvetináč zavlažuje 1 – 7x denne podľa stupňa rastu rastliniek a v noci prechádza do režimu spánku, v ktorom ostáva 8 hodín. Pri nízkej hladine vody sa spustí alarm, ktorý vyzýva na doplnenie. [3]

Obrázok 2.3: Systém 2: Plantui

Obrázok 2.4: Systém 2: Plantui – detail

Najväčšou výhodou tohoto systému je to, že rastlinám poskytuje ucelený inteligentný ekosystém, ktorý sa automaticky prispôsobuje ich potrebám. S tým sa však spája aj jeho najväčšia nevýhoda, ktorou je skutočnosť, že tento systém funguje len so špecifickým typom špeciálnych sadeničiek a nie je riešením pre už rastúce rastliny v domácnosti, ktoré sa nachádzajú vo vlastných kvetináčoch. Ďalšou veľkou nevýhodou je tiež jeho vysoká cena.

2.1.3 Systém 3: Sense

Medzi systémy, ktoré taktiež stoja za zmienku, patria určtite aj samozavlažovacie kvetináče ako napríklad Sense, ktorý neobsahuje žiadnu elektroniku. Kvetináč tvorí plniaca trubica s indikátorom množstva vody a oddeľovacia mriežka. Do plniacej trubice stačí naliať vodu, ktorá sa tak dostane na dno nádoby a rastlina si následne čerpá vlahu sama, vždy keď ju korene potrebujú. Interval dolievania závisí na veľkosti zásobníku, počasí, type a veľkosti rastliny. [4]

Najväčšou výhodou takýchto kvetináčov je ich nízka cena, avšak rastliny je do nich potrebné presadiť a nedisponujú absolútne žiadnymi inteligentnými funkciami. Vo väčšine prípadov môžu byť vyhovujúce, avšak v prípade náročnejších alebo exotickejších rastlín je nutné použiť iné riešenie.

Obrázok 2.5: Systém 3: Sense

2.2 Zhrnutie

Hlavným dôvodom pre vytvorenie nového systému je skutočnosť, že na trhu zrejme zatiaľ neexistuje žiadne interiérové riešenie, ktoré by bolo možné aplikovať na už rastúce rastliny, v ich vlastných kvetináčoch, bez potreby presádzania, a ktoré by bolo dostatočne kompaktné, aby sa s ľahkosťou vošlo napríklad aj na pracovný stôl, a zároveň dizajnované tak, aby príliš nevynikalo a nenarúšalo tak prostredie, do ktorého je umiestnené.

Ďalej zrejme neexistuje systém, ktorý by bolo možné ovládať prostredníctvom jednoduchej, spoľahlivej a užívateľsky prívetivej mobilnej aplikácie, ktorá by dovoľovala správu viacerých takýchto zariadení súčasne a umožňovala by nastavenie presného harmonogramu zálievky podľa konkrétneho druhu rastliny pre každé z nich.

Okrem toho by tento systém vďaka existencii sprievodnej aplikácie mohol implementovať funkcie ako zaslanie notifikácie na mobilný telefón užívateľa s výzvou na doplnenie vody vždy, keď hladina v zásobníku v niektorom zo zariadení klesne pod kritickú hranicu, a vedenie histórie udalostí, kde by bolo možné vidieť záznamy o všetkých zálievkach a situáciách, ktoré sa v systéme udiali.

Po prieskume súčasného trhu so zavlažovacími systémami pre izbové rastliny a uskutočnení analýzy najzaujímavejších súvisiacich riešení, je možné povedať, že zariadenie, ktoré by v sebe kombinovalo všetky vyššie uvedené atribúty, zatiaľ neexistuje a vzhľadom na jeho charakter a potenciál do budúcnosti ho dáva zmysel realizovať.

3 Návrh

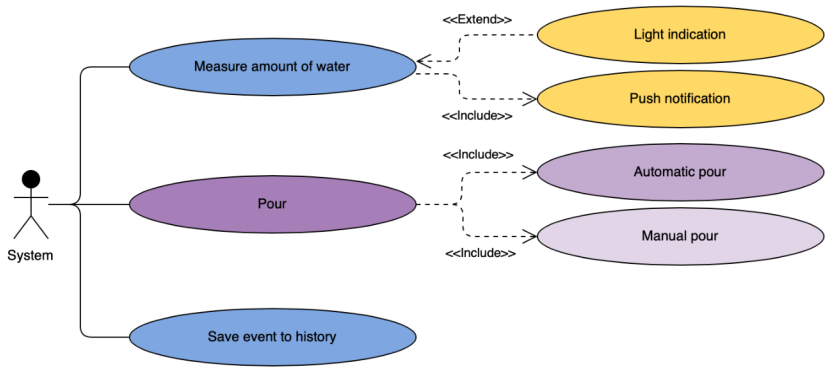
Táto kapitola sa venuje návrhu celého systému a je rozdelená na tri časti, ktorými sú: návrh samotného zariadenia, návrh mobilnej aplikácie ako ovládacieho rozhrania a 3D návrh púzdra, do ktorého bude zariadenie vo finále zasadené.

3.1 Návrh zariadenia

Návrh zariadenia sa zaoberá rozborom požadovaných funkcií, ktoré zariadenie musí implementovať, aby bol systém kompletný a zmysluplný, a technológií, ktoré budú pre ich realizáciu použité.

3.1.1 Funkcionalita zariadenia

Nasledujúci use case diagram reprezentuje základnú funkcionalitu systému. Kľúčovým slovom **Include** sú označené väzby, ktoré sú pre jeho fungovanie nevyhnutné, a kľúčovým slovom **Extend** sú naopak označené väzby, ktoré nevyhnutné nie sú a systém len vylepšujú.



Obrázok 3.1: Use case diagram – funkcionalita zariadenia

Hlavnou funkciou tohoto zavlažovacieho systému je samozrejme schopnosť uskutočniť zálievku cieľovej rastliny. Tento úkon by mal byť navyše systémom vykonávaný automaticky a to v pravidelných intervaloch podľa prednastaveného harmonogramu. Môžu však nastať situácie, kedy harmonogram nastavený nebude alebo bude požadované uskutočnenie dodatočnej, neplánovanej zálievky. Z tohoto dôvodu musí zariadenie implementovať aj funkciu manuálneho zalievania.

Ďalšou zásadnou funkciou, bez ktorej by bol systém takmer nepoužiteľný, je schopnosť zistiť, aké množstvo vody sa v zásobníku aktuálne nachádza. Tento

údaj je dôležitý najmä v situácii, kedy dochádza voda a o tejto skutočnosti je potrebné informovať užívateľa. Vďaka tomu, že ovládacím rozhraním je mobilná aplikácia, výzva na doplnenie vody mu môže byť doručená formou push notifikácie na jeho mobilný telefón.

Zariadenie by mohlo byť doplnené o svetelnú signalizáciu pomocou dvoch LED diód – modrej a červenej, ktorá by v čase, kedy by bolo v zásobníku vody dostatok, svietila na modro, a naopak v čase, kedy by hladina klesla pod stanovenú kritickú hranicu, svietila na červeno.

Poslednou a veľmi dôležitou funkciou je vedenie histórie udalostí, do ktorej by systém ukladal záznamy o úspešnej, prípadne (kvôli nedostatku vody alebo odstavenému zariadeniu) neúspešnej zálievke, a o dosiahnutí nedostatočného stavu zásobníka a zaslaní upozornenia. Užívateľ by tak vedel, či systém funguje správne a mal by podrobný prehľad o zavlažovaní cieľovej rastliny.

Po tomto rozbore je vhodné zhrnúť hlavné funkcie systému, ktorými sú: automatická zálievka, manuálna zálievka, meranie objemu vody, svetelná signalizácia, posielanie notifikácii a vedenie histórie udalostí. Z ich podstaty jasne vyplýva, že na ich realizáciu budú potrebné tieto komponenty: logická jednotka, zásobník vody, čerpadlo vody, trubica, technológia merania objemu vody a dve LED diódy – modrá a červená.

3.1.2 Použité technológie

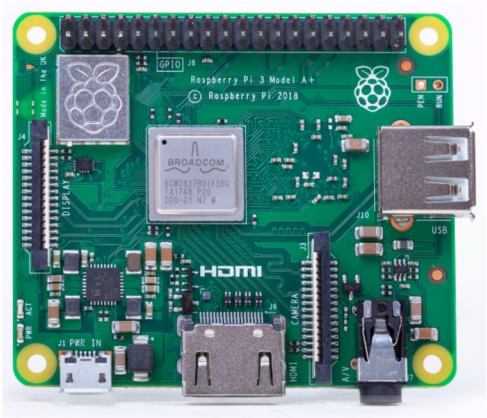
Aby zariadenie vôbec mohlo fungovať, potrebuje logickú jednotku, ktorá bude všetko riadiť a vyhodnocovať. Z bežne dostupných výpočtových modulov malých rozmerov, ktoré je možné priamo použiť, dominujú dve platformy: Raspberry Pi a Arduino.

Raspberry Pi je malý jednodoskový počítač s mikroprocesorom, ktorý sa v mnohom podobá na tradičný desktopový. Nie je na ňom problém rozbehnúť operačný systém (zvyčajne Unixový) a je schopný vykonávania viacerých úloh súčasne. Arduino je naopak mikrokontrolér, ktorý je schopný vykonávať len jeden program dookola. V porovnaní s Raspberry Pi má okrem toho niekoľkonásobne nižší výkon a zvyčajne nemá integrovaný WiFi modul. [5]

Keďže projekt si kvôli komunikácii s mobilnou aplikáciou vyžaduje, aby bolo zariadenie pripojené na internet a bežal na ňom server, čo znamená, že je potrebný WiFi modul (ideálne priamo integrovaný) a zariadenie musí okrem programu zalievania paralalene odbavovať dotazy prichádzajúce z aplikácie, bude na realizáciu použité Raspberry Pi.

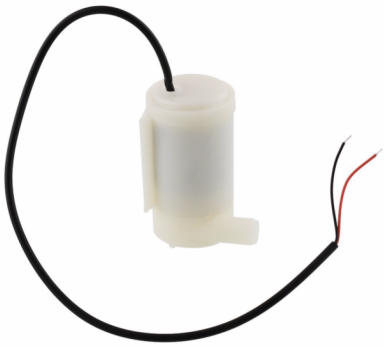
Po voľbe platformy je však potrebné rozhodnúť sa ešte pre konkrétny model, pričom kľúčovými atribútmi sú rozmery a či má daný model integrovaný WiFi modul kvôli už spomínanému bezdrôtovému pripojeniu na internet. Z celej rady modelov tak ostávajú modely Zero W o veľkosti 6,5 x 3 cm a 3A+ o veľkosti 6,5 x 5,6 cm. Rapsberry Pi 3A+ je síce o niečo väčšie, ale vzhľadom na to, že jedno vlastním a jeho rozmery sú pre projekt stále prijateľné, bude použité to.

Tento model disponuje 64-bitovým štvorjadrovým procesorom Cortex-A53 s taktom 1,4 Ghz a 512 MB operačnej pamäte. Neobsahuje však rozhranie pre pevný alebo SSD disk. Pre zavedenie operačného systému a uchovanie dát je určený slot na SD kartu. Na komunikáciu s okolím má k dispozícii 40 pinov. [6]



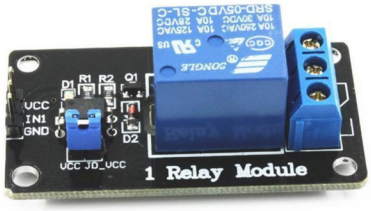
Obrázok 3.2: Raspberry Pi 3A+

Ďalšou zásadnou komponentou systému je čerpadlo. Najmenším bežne dostupným je ponorné mini čerpadlo o rozmeroch 4,5 x 3,3 cm s priemerom 2,4 cm a s prietokom 80 – 120 l za hodinu. Na napájanie potrebuje napätie od 2,5 do 6V, takže použiť ho v kombinácii s Raspberry Pi a jedným z jeho dvoch 5V pinov by nemal byť problém. V internetovom obchode, kde bolo zakúpené, je mimochodom udávaná životnosť 500 hodín, čo vzhľadom na to, že bude zapínané len na pár sekúnd a nanajvýš len raz za deň, môžeme považovať prakticky za nekonečno. [7]



Obrázok 3.3: Mini čerpadlo

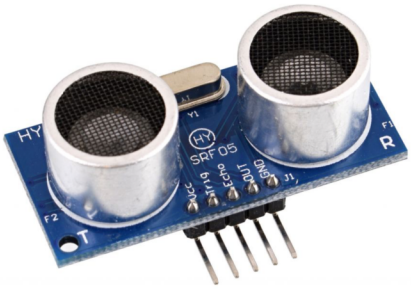
Ak by bolo toto čerpadlo pripojené k Raspberry Pi priamo, nepretržite by pracovalo a nedalo by sa vypnúť. Aby bolo možné jeho obvod zatvárať a otvárať podľa potreby, systém si vyžaduje ešte jednu elektronickú komponentu, ktorou je nízkonapäťový relé modul, ktorý sa spína už pri napätí 5V. Pre účely tohto projektu stačí jednokanálový.



Obrázok 3.4: Nízkonapäťový relé modul

Hlavným prvkom tohto modulu je elektromagnetické relé. Relé má v sebe cievku navinutú na jadre z mäkkého feromagnetického materiálu, ktorá sa po aplikovaní elektrického prúdu stáva elektromagnetom. Ten k sebe v daný moment pritiahne pohyblivú kotvu a tá uzavrie obvod. Pri prerušení dodávky prúdu sa kotva vráti do svojej pôvodnej polohy a obvod sa opäť otvorí. [8] Ostatné komponenty modulu umožňujú, aby sa relé zoplo až v momente, kedy sa pin z Raspberry Pi, ktorý je pripojený na pin modulu s označením IN1, nastaví na 1.

Ďalšou technológiou, u ktorej je potrebné rozhodnúť, ako bude realizovaná je technológia merania objemu vody v nádobe. Tento problém je možné riešiť viacerými spôsobmi, avšak dvoma najlepšími alternatívami sú: ultrazvukový senzor na bezkontaktné meranie vzdialenosti a váhový senzor, ktorého obdoba sa nachádza takmer v každej kuchynskej váhe. Keďže zadanie práce si explicitne vyžaduje, aby bolo množstvo vody v zásobníku zisťované bezkontaktne, na realizáciu v tomto projekte bol zvolený ultrazvukový senzor, konkrétne HY-SRF05. Tento ultrazvukový senzor potrebuje na svoje fungovanie napätie 5V, takže podobne ako čerpadlo dokáže fungovať v kombinácii s Raspberry Pi a bude pripojený k druhému z jeho dvoch 5V pinov. Pripojenie si však bude vyžadovať aj dva rezistory – jeden s odporom 330 a druhý s odporom 470 ohmov.



Obrázok 3.5: Ultrazvukový senzor

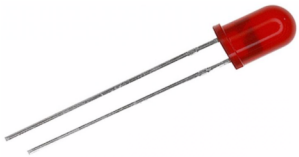
Obrázok 3.6: Rezistor – 330 ohmov

Obrázok 3.7: Rezistor – 470 ohmov

Na jeho plošnom spoji sa nachádzajú dve veľké oválne elementy, z ktorých jeden je ultrazvukový vysielač a druhý ultrazvukový prijímač. Meranie vzdialenosti funguje tak, že vysielač vyšle signál, ktorý putuje priestorom, a v momente, kedy narazí na prekážku, sa odrazí a vracia späť do prijímača. Doba oneskorenia tak udáva vzdialenosť prekážky na základe rýchlosti šírenia zvuku (340 m/s). [9] Akonáhle je výška hladiny vody známa, nie je problém dopočítať jej objem.

Ako posledné je potrebné realizovať svetelnú signalizáciu. Ide o jednoduchú úlohu, na ktorú postačia dve bežné LED diódy – modrá a červená. Aby sa dióda nepoškodila, pripojenie k Raspberry Pi si bude vyžadovať použitie rezistora, konkrétne s odporom 470 ohmov.

Obrázok 3.8: LED – modrá



Obrázok 3.9: LED – červená

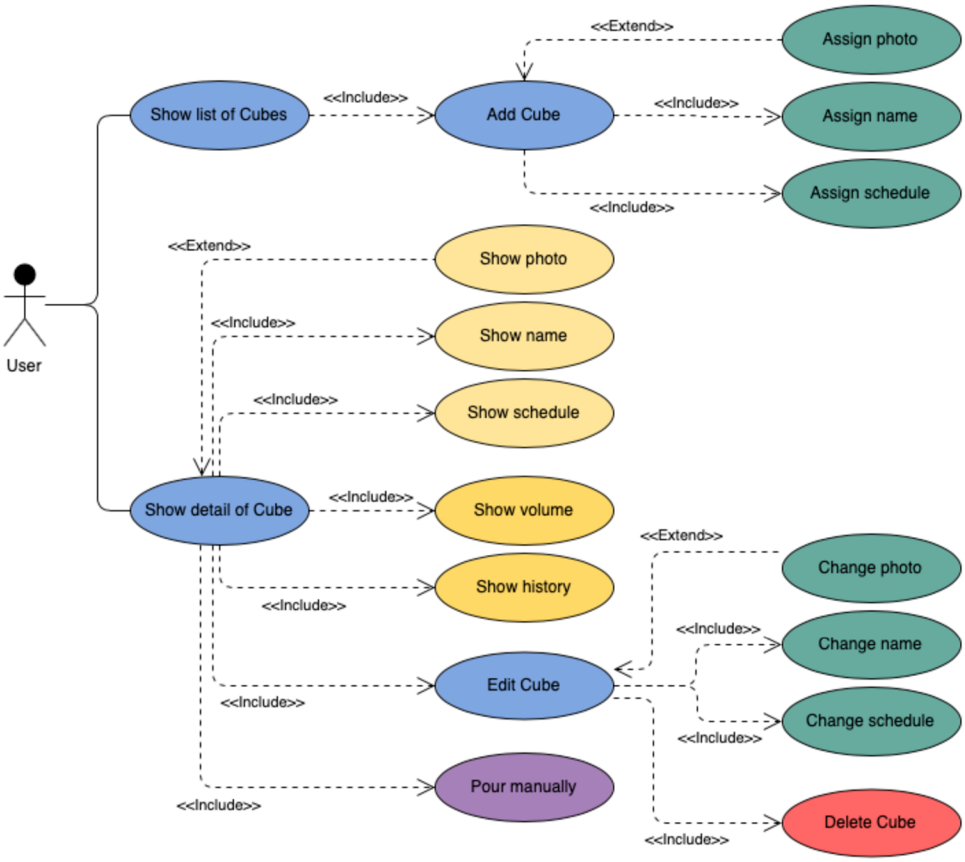
Na realizáciu projektu budú teda okrem elementov, ktoré boli výstupom z kapitoly 3.1.1 Funkcionalita zariadenia, potrebné aj nasledujúce elektronické komponenty: jednokanálový nízkonapäťový relé modul s 5V elektromagnetickým relé a tri rezistory s odporom 470 ohmov a jeden s odporom 330 ohmov. Logickou jednotkou bude Raspberry Pi 3A+ a na meranie objemu vody v zásobníku bude použitý ultrazvukový senzor HY-SRF05. Káble sú samozrejmosťou.

3.2 Návrh mobilnej aplikácie

Táto podkapitola je venovaná návrhu mobilnej aplikácie ako ovládacieho rozhrania, ktoré užívateľovi umožní nastavovať, sledovať a ovládať viacero zariadení súčasne. Aplikácia je určená pre mobilné telefóny s operačným systémom iOS od spoločnosti Apple.

3.2.1 Funkcionalita mobilnej aplikácie

Nasledujúci use case diagram reprezentuje všetky funkcie, ktoré by mobilná aplikácia mala v základe podporovať.



Obrázok 3.10: Use case diagram – funkcionalita aplikácie

Od mobilnej aplikácie sa v prvom rade požaduje, aby umožňovala správu viacerých zariadení súčase, takže jednou z funkcionalít musí byť nepochybne schopnosť zobraziť zoznam všetkých zariadení, ktoré užívateľ do systému pridal.

Z toho logicky vyplýva, že musí existovať spôsob ako do aplikácie zariadenia pridávať. Každé zariadenie bude po aktivovaní prístupné na svojej unikátnej URL, takže od užívateľa sa bude vyžadovať ho zadať.

V priebehu pridávania by malo byť možné uskutočniť iniciálne nastavenie, počas ktorého bude novému zariadeniu priradená fotografia, ktorá vznikne buď priamym použitím fotoaparátu, alebo výberom z galérie, priradený názov a nastavený harmonogram zalievania výberom z knižnice rastlín.

Mobilná aplikácia by ďalej mala umožňovať preklik z pridaného zariadenia v zozname zariadení na jeho detail, kde by boli uvedené už spomenuté zariadenie charakterizujúce atribúty ako fotografia, názov daného zariadenia a názov vybraného harmonogramu, ako aj všetky dôležité informácie ako stav zásobníku vody a história udalostí. Z tohoto miesta by malo byť v prípade potreby možné uskutočniť aj manuálnu zálievku rastliny.

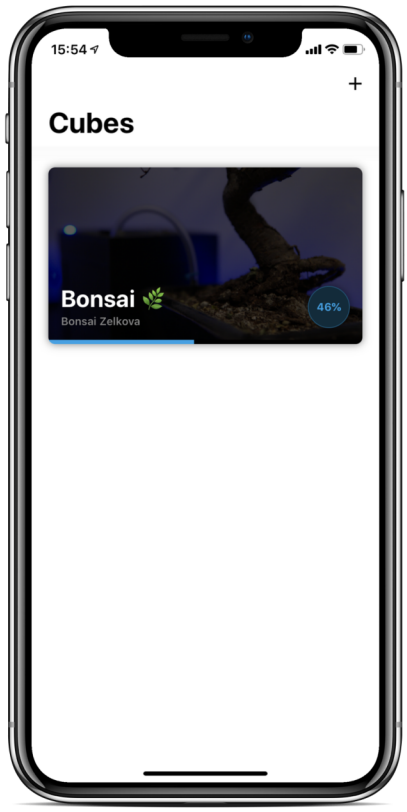
Okrem toho nesmie chýbať funkcionalita, ktorá by dovoľovala atribúty zariadenia meniť. Týmito atribútmi sú opäť fotografia, názov a výber harmonogramu, a ich uloženie by sa malo okamžite prejaviť na detaile zariadenia a fungovaní systému. No a napokon musí existovať aj spôsob ako dané zariadenie zo systému zmazať.

Z rozboru požadovanej funkcionality vyplýva, že aplikácia bude pozostávať celkom z 5 obrazoviek, ktorými sú: zoznam zariadení, pridanie zariadenia na základe URL, iniciálne nastavenie zariadenia, detail zariadenia a úprava zariadenia.

3.2.2 Užívateľské rozhranie

Pri dizajnovaní týchto obrazoviek sa vychádzalo z popisu funkcionality z predchádzajúcej kapitoly a dbalo sa hlavne na čistotu, jednoduchosť a intuitívnosť. Grafický návrh striktne rešpektuje jednotný štýl písma, farebnú paletu a 20 px odsadenie všetkých prvkov od oboch bočných krajov displeja.

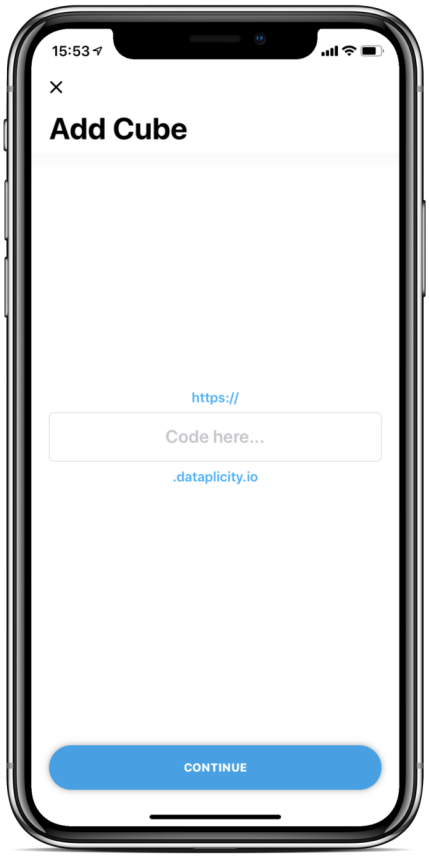
Obrázok 3.11: UI – zoznam zariadení (prázdny)



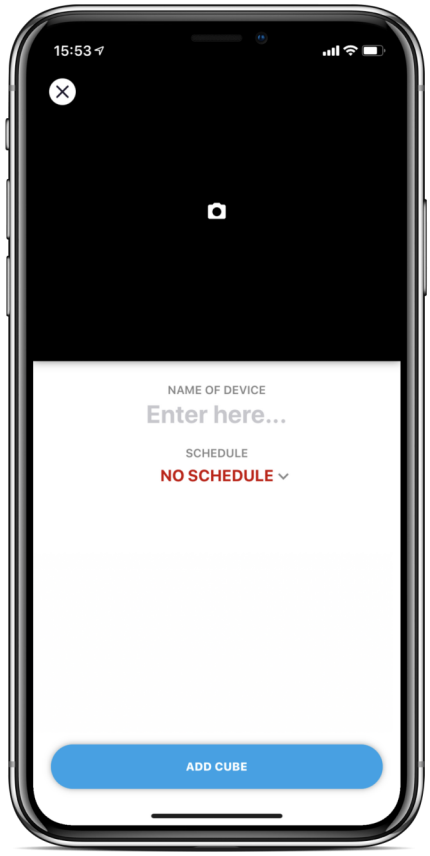
Obrázok 3.12: UI – zoznam zariadení

Prvou obrazovkou, ktorá sa zobrazí ihneď po spustení aplikácie, je zoznam zariadení s titulkou Cubes, ktorý je po prvom štarte logicky prázdny, v strede s nápisom NO CUBES, YET. Tlačítko na pridávanie zariadení je vyobrazené ako symbol plus a umiestnené v pravom hornom rohu navigačného panelu a teda aj v pravom hornom rohu obrazovky.

Pre každé nové zariadenie sa v tabuľke objaví nová bunka, ktorá ho reprezentuje. Celé jej pozadie vypĺňa fotografia zariadenia a v ľavom dolnom rohu je zobrazený jeho názov. Pod ním sa nachádza názov vybraného harmonogramu zalievania, pričom ak nastane situácia, že zariadenie ho nastavený nemá, objaví sa namiesto toho nápis NO SCHEDULE. V spodnej časti bunky je umiestnený modrý ukazovateľ aktuálneho stavu zásobníku vody, ktorý je v pravom dolnom rohu bunky doplnený o percentuálne vyjadrenie aktuálneho objemu k maximálnemu možnému.



Obrázok 3.13: UI – pridať zariadenie



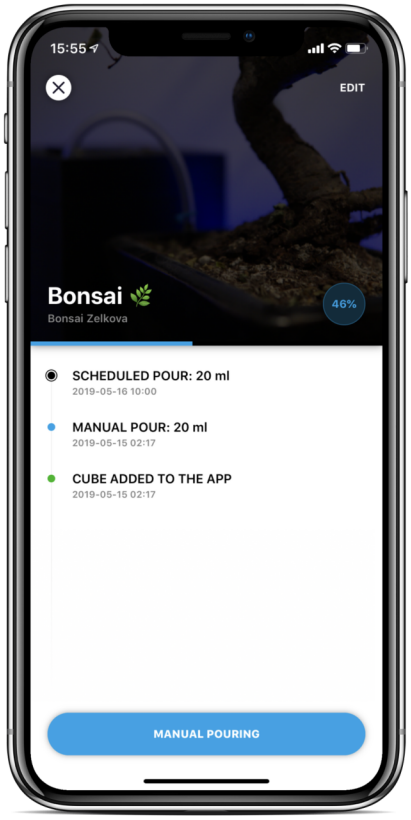
Obrázok 3.14: UI – inicializácia

Po kliknutí na už spomínané tlačítko pridania zariadenia sa užívateľ dostane na obrazovku s titulkou Add Cube. V jej strede sa nachádza textové pole na dopísanie unikátnej URL práve pridávaného zariadenia a v spodnej časti tlačítko CONTINUE, ktoré po kliknutí overí jeho dostupnosť a v prípade, že je

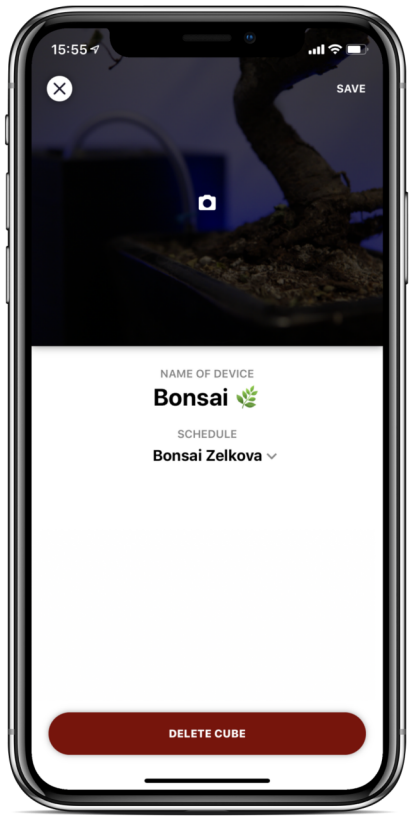
všetko v poriadku, užívateľa presmeruje na druhý a zároveň posledný krok procesu pridávania, ktorým je iniciálne nastavenie nového zariadenia.

Obrazovka iniciálneho nastavenia je rozdelená na dve časti, pričom horná, nazývaná hlavička, je na začiatku čierna, v strede s ikonou fotoaparátu. Ten napovedá, že po kliknutí do tejto oblasti je možné zariadeniu priradiť fotografiu. V ľavom hornom rohu je potom tlačítko na opustenie celého procesu, ktoré po kliknutí obe obrazovky súbežne zahodí smerom nadol a užívateľa dostane späť na zoznam zariadení.

V spodnej časti obrazovky, hneď pod hlavičkou, sa nachádza textové pole vyhradené na priradenie názvu zariadenia a pod ním dropdown tlačítko, defaultne s nápisom NO SCHEDULE, ktoré po kliknutí zobrazí tzv. picker so skrolovateľným zoznamom harmonogramov, ktorý sa vysunie zo spodnej časti displeja rovnakým spôsobom ako systémová klávesnica. Každý harmonogram nesie názov rastliny, pre ktorú je určený. Posledným prvkom tejto obrazovky je tlačítko ADD CUBE, umiestnené v spodnej časti obrazovky, na ktoré je po vyplnení všetkých potrebných atribútov možné kliknúť a celý proces pridávania tak dokončiť. Užívateľ sa následne ocitne späť na zozname zariadení, avšak doplnenom o nové zariadenie, ktoré práve pridal.



Obrázok 3.15: UI – detail zariadenia



Obrázok 3.16: UI – editácia zariadenia

Ak na obrazovke zoznamu zariadení užívateľ klikne na niektoré z nich, dostane sa na obrazovku zobrazujúcu jeho detail. Tá je rovnako ako obrazovka iniciálneho nastavenia rozdelená na dve časti, pričom hlavička je tentokrát takmer zhodná s dizajnom bunky zariadenia z tabuľky Cubes. V tomto prípade však fotografia vypĺňa displej až po okraj, v ľavom hornom rohu sa nachádza tlačítko, ktoré umožňuje detail zavrieť a dostať sa späť na zoznam, a v pravom hornom rohu sa nachádza tlačítko EDIT, ktoré bude popísané neskôr. Inak aj tu je zobrazený názov zariadenia, názov vybraného harmonogramu a indikátor a percentuálne vyjadrenie stavu zásobníka vody.

V druhej časti obrazovky detailu sa nachádza skrolovateľná história udalostí, v ktorej sú záznamy o všetkých dôležitých momentoch, ktoré sa na danom zariadení odohrali. Výnimkou je len prvá položka, nesúca informáciu o najbližšej zálievke, ktorá sa však nezobrazuje, ak nie je nastavený žiaden harmonogram. Okrem hlavičky a histórie je v spodnej časti tejto obrazovky umiestnené aj tlačítko s nápisom MANUAL POURING, ktoré po kliknutí vysunie picker, podobne ako tomu bolo u výberu harmonogramu na obrazovke iniciálneho nastavenia, tentokrát však so skrolovateľným zoznamom možných veľkostí zálievky v mililitroch, začínajúcich na hodnote 5, stúpajúcej vždy o 5 a končiacich na hodnote 100 alebo maximálnej možnej, v závislosti na stave zásobníka vody. Po potvrdení voľby sa spustí zalievanie a do histórie udalostí sa prída záznam MANUAL POUR s podtitulom IN PROGRESS, ktorý sa po dokončení zmení na klasický záznam MANUAL POUR v podtitule s dátumom uskutočnenia udalosti.

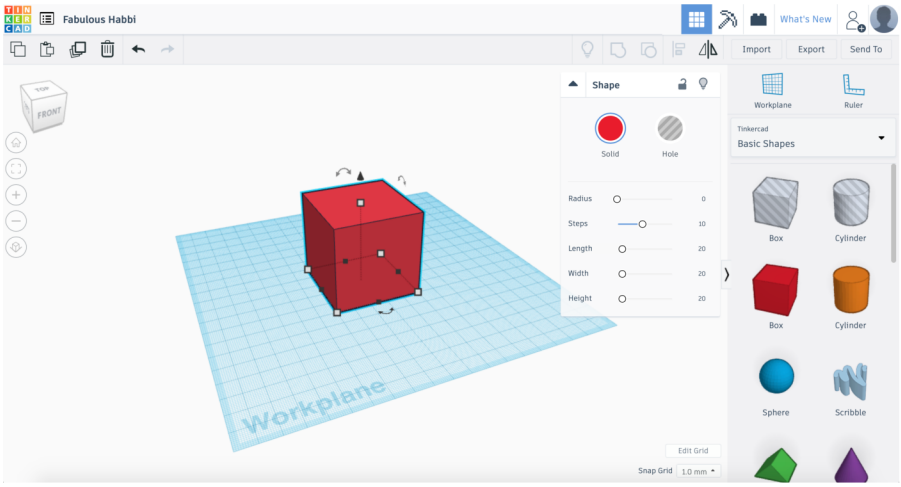
Poslednou obrazovkou je obrazovka editácie zariadenia, ktorú je možné vyvolať po kliknutí na už spomenuté tlačítko EDIT. Táto obrazovka je takmer zhodná s obrazovkou iniciálneho nastavenia, pretože umožňuje vykonať presne tie isté úkony. Rozdiel je len v tom, že tlačítko odchodu prepne režim späť na detail, v spodnej časti obrazovky je umiestnené červené tlačítko DELETE CUBE, ktoré umožňuje zariadenie vymazať, a v pravom hornom rohu obrazovky je umiestnené tlačítko SAVE, ktoré uskutočnené zmeny uloží.

3.3 Návrh púzdra

Po dokončení návrhu systému, z ktorého je jasné, aké elektronické komponenty budú na realizáciu prototypu použité, je vhodné zhotoviť mu reprezentatívne púzdro, do ktorého budú neskôr všetky tieto komponenty zasadené. Púzdro bude zhotovené technológiou 3D tlače a z estetických a priestorových dôvodov je zamýšľané v tvare kocky o rozmeroch 10x10x10 cm, čo je dostatočne malé na to, aby bol výsledok kompaktný, a dostatočne veľké na to, aby sa doň vošlo všetko potrebné a stále disponovalo dostatkom miesta na vodu.

3.3.1 Modelovanie

Digitálne modely všetkých čaastí púzdra boli vytvárané pomocou online nástroja Tinkercad, ktorý je zdarma dostupný na tomto odkaze [10] a tvorenie teda prebieha priamo v internetovom prehliadači. Použitie si vyžaduje vytvorenie účtu užívateľa, avšak po prihlásení nástroj umožňuje vytváranie a ukladanie neobmedzeného počtu modelov, čo bolo pre tento projekt v konečnom dôsledku veľmi dôležité.

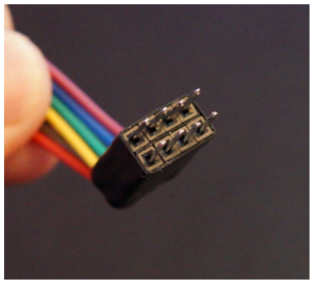


Obrázok 3.17: Modelovací nástroj Tinkercad

Proces modelovania je pomerne jednoduchý. Spočíva v použití základných geometrických objektov ako kocka a valec a ich presné škálovanie a pozicovanie v rámci 3D priestoru, ktorý je možné približovať alebo oddiaľovať a neobmedzene otáčať. Na tvorený model je tak môžné nahliadať aj zo spodu, čo bolo častokrát veľmi prínosné. Vložené objekty je možné otáčať, preklápať a vzájomne zarovnávať. Je na mieste podotknúť, že všetky funkcie, ktoré nástroj Tinkercad poskytuje, sa veľmi príjemne používajú.

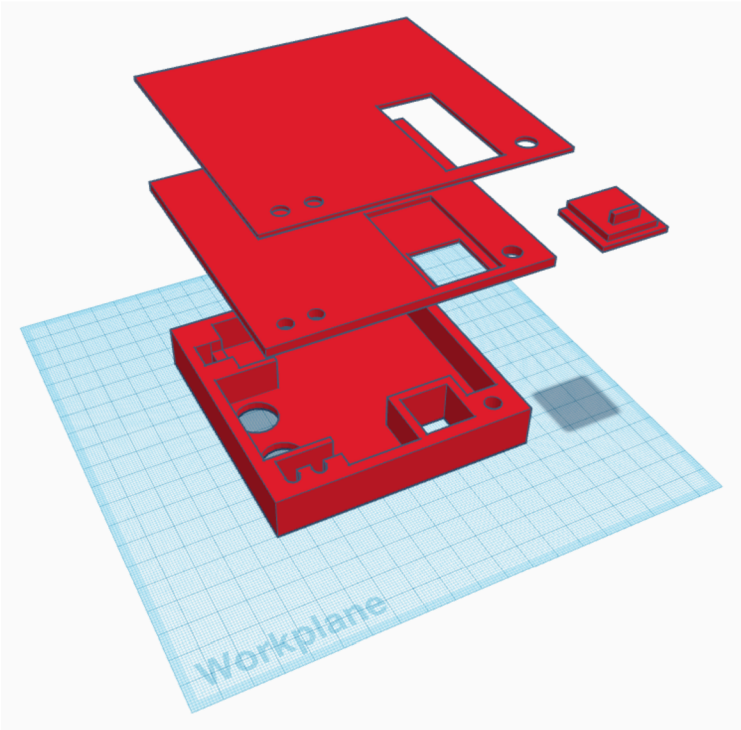
Najdôležitejšou funkciou je však zoskupovanie, tzv. groupping. Ten umožňuje ľubovoľný počet samostatných objektov spojiť do jedného a ďalej pracovať len s ním. Toto zoskupovanie dáva najväčší zmysel napríklad hlavne vtedy, ak je potrebné v nejakom objekte urobiť výrez. U každého objektu je možné určiť, že sa jedná o pevný materiál alebo reprezentuje prázdne miesto. Ak je objekt prázdneho miesta umiestnený tak, aby zasahoval do objektu pevného materiálu, tieto dva objekty sa označia a aplikuje sa na nich zoskupenie, výsledkom bude objekt, ktorý má výrez v objekte z pevného materiálu v oblasti, do ktorej pred tým zasahoval objekt prázdneho miesta. Týmto spôsobom je možné postupne vytvárať čoraz zložitejšie komponenty a vo výsledku veľmi sofistikované riešenia.

3D model púzdra zariadenia bol tvorený tak, aby spĺňal kritérium tvaru kocky a skladal sa z dvoch hlavných do seba zapadajúcich a zároveň ľahko odoberateľných častí. Vrchná je určená na zasadenie ultrazvukového senzora a LED diód, zatiaľ čo spodná je určená na zasadenie Raspberry Pi, relé modulu a čerpadla. Elektronika týchto dvoch objektov je zamýšľané prepojiť pomocou skupiny 8 káblov, ktoré budú spolu zlepené do tvaru konektora a to nasledovne:



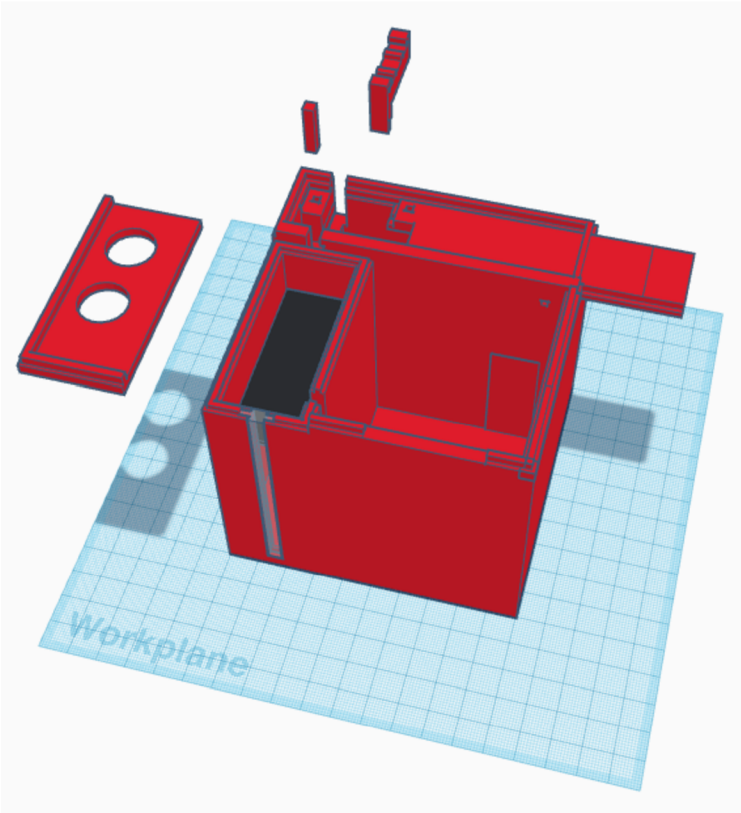
Obrázok 3.18: Konektor

Vrchná časť kocky sa skladá celkom zo 4 samostatných dielov, ktoré do seba presne zapadajú pomocou drážok. Jedným z dielov je základňa s výrezmi a priestorom pre všetky jej vyššie uvedené komponenty, tunelom na dopĺňanie vody a výrezom na trubicu čerpadla. Základňu potom uzatvoria dve ploché doštičky, ktoré spolu s menším posuvným dielom vytvoria mechanizmus poklopu na otváranie a zatváranie prístupu k tunelu.



Obrázok 3.19: 3D model púzdra – vrchná časť

Spodná časť kocky sa skladá celkom zo 6 samostatných dielov, pričom hlavným a najväčším je samotné telo kocky, ktoré je priečnou stenou rozdelené na 2 separátne priestory. Zadný je vyhradený pre Raspberry Pi a relé modul, a je možné ho posuvným mechanizmom zatvárať a po zatvorení zaistiť poistkou. Priestor v prednej časti plní úlohu zásobníka vody, ktorého súčasťou je čerpadlo. To bude prilepené na zadnú stenu zásobníka v mieste vymedzenom obdĺžnikovým výbežkom. Nad ním v hornej časti tejto steny je navyše malá diera na kábel, ktorý bude čerpadlo prepájať s relé modulom v zadnej časti. Meranie hladiny vody sa bude uskutočňovať v ultrazvukovej komore, ktorá vznikla pridaním ďalšej steny. Prietok vody do komory je zaistený plným vertikálnym otvorom v stene. Okrem toho je v prednej časti tela výrez pre nepriehľadný diel, ktorý sa do neho zasunie a malo by byť cez neho možné vidieť dovnútra a získať tak povedomie o výške hladiny vody aj voľným okom, čo je dôležité najmä pri dolievaní vody.



Obrázok 3.20: 3D model púzdra – spodná časť

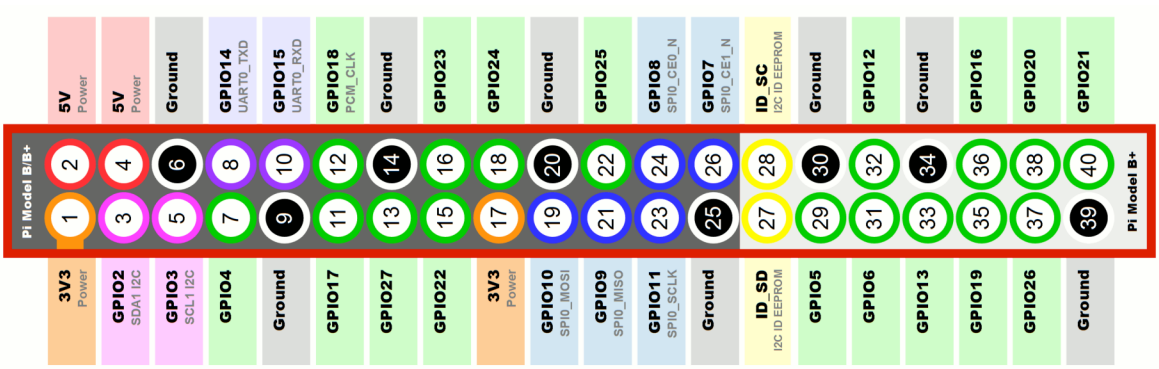
4 Realizácia

Po dokončení návrhu je čas na realizáciu a zhotovenie celého systému. Kapitola je rozdelená na 5 častí, ktoré na seba postupne nadväzujú. Sú nimi: zapojenie komponent, vývoj zariadenia, vývoj mobilnej aplikácie, zhotovenie púzdra a kalibrácia.

4.1 Zapojenie komponent

Prvou dôležitou úlohou je zapojenie všetkých elektronických komponent, z ktorých má výsledný systém pozostávať. Základom je Raspberry Pi, pretože všetky ďalšie komponenty budú pripojené práve do neho.

Raspberry Pi má na komunikáciu s inými elektronickými komponentami k dispozícii 40 pinov, z ktorých len 2 poskytujú napätie 5V na napájanie iných zariadení. [11] V tomto projekte sú takýmito komponentami len relé modul s čerpadlom a ultrazvukový senzor, takže množstvo je dostatočné a tieto 2 piny budú plne využité. Dôležitou informáciou je, že piny s označením GPIO je možné programovo prepínať medzi stavmi 0 a 1 a prinášať tak na nich prúd nízkeho napätia. Kompletná schéma pinov je zobrazená na nasledujúcom obrázku. Rozloženie pinov u modelu B+ je zhodné s rozložením pinov u modelu 3A+. [11]



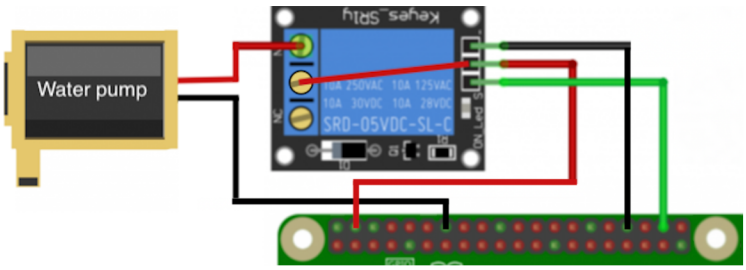
Obrázok 4.1: Schéma pinov Raspberry Pi B+

Prvou komponentou, ktorú je potrebné zapojiť, je relé modul, na ktorý bude následne pripojené čerpadlo. Modul má 3 piny s označením Vcc, IN1 a GND. Vcc je pin určený na napájanie, IN1 je skratka pre INput1 a ide v podstate o trigger a GND je skratka pre GrouND. Modul má tiež 3 výstupy, z ktorých 2 krajné sú označené ako NC a NO, pričom NC je skratka pre NormallyClosed a NO je skratka pre NormallyOpen.

Na stredný by mal byť aplikovaný elektrický prúd, ktorý je v úmysle poslať obvodom ďalej. Po jeho aplikácii je k dispozícii na výstupe NC, pretože obvod medzi stredným a NC výstupom je defaultne zatvorený. Naopak NO nie je so stredným výstupom nijak spojený a nedostáva tak žiaden prúd.

Po aplikovaní nízkeho napätia z Raspberry Pi na pin IN1 bude cievkou v relé prechádzať dostatočne veľký prúd na to, aby sa z nej stal elektromagnet. Ten si pritiahne pohyblivú kotvu, ktorá je súčasťou relé, a uzavrie tak obvod medzi stredným a NO výstupom, ktorý bol pred tým otvorený, a naopak otvorí tak obvod medzi stredným a NC výstupom, ktorý bol predtým zatvorený. Tým sa zmenil stav a prúd aplikovaný na stredný výstup je teraz k dispozícii na výstupe NO. Naopak NC teraz nedostáva žiaden prúd. Po ukončení napájania pinu IN1 cievka v relé prestane generovať elektromagnetické pole, kotva sa vráti na svoje miesto a obvody sa tak vrátia do svojho pôvodného stavu.

Týmto spôsobom je teda pomocou Raspberry Pi možné spínať obvod a na výstup NO, ktorý je defaultne otvorený, prinášať prúd len vtedy, keď je to žiadané, a zapínať a vypínať tak čerpadlo. Modul zapojíme podľa nasledujúcej schémy:



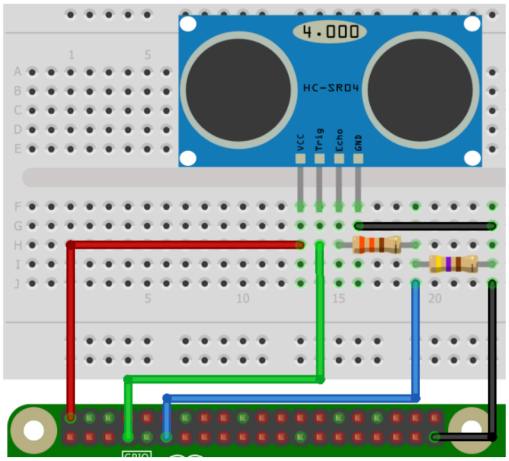
Obrázok 4.2: Obvod: RPi + relé modul + čerpadlo

Pin Vcc bude zapojený do 5V Raspberry Pi pinu s číslom 4, IN1 bude zapojený napríklad do pinu číslo 38 (GPIO 20) a GND napríklad do pinu Ground 34. Okrem toho je dôležité káblom preniesť elektrický prúd z pinu Vcc na stredný výstup, aby ho bolo možné v momente uzatvorenia obvodu poslať ďalej.

Po úspešnom zapojení relé modulu je možné plnohodnotne zapojiť aj čerpadlo. To disponuje dvoma káblami, z ktoých jeden je plus a druhý mínus. Plus je smozrejme určený na napájanie, takže sa zapojí do výstupu NO a mínus je potrebné zapojiť do niektorého z Ground pinov na Raspberry Pi, napríklad 14.

Ďalšou komponentou, ktorú je potrebné zapojiť, je ultrazvukový senzor. Ten má 5 pinov s označením Vcc, Trig, Echo, Out a GND. Vcc je pin určený na nápajanie, Trig je skratka pre Trigger a jeho úlohou je zabezpečenie vysielania signálu v momente aplikácie prúdu nízkeho napätia, Echo je pin, ktorý je určený na prijímanie vracajúceho sa signálu, Out je skratka pre Output (tento pin však

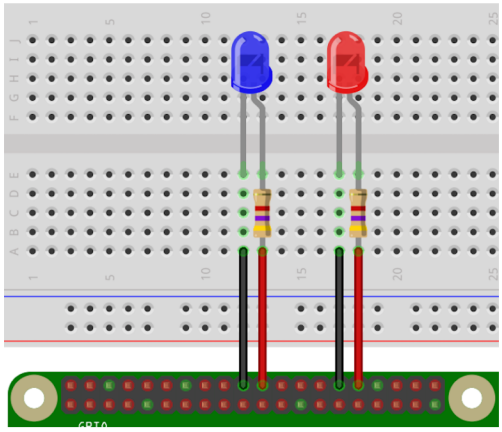
použitý nebude) a GND je skratka pre GrouND. Komponenta bude zapojená podľa nasledujúcej schémy [12]:



Obrázok 4.3: Obvod: RPi + ultrazvukový senzor

Pin Vcc bude zapojený do 5V Raspberry Pi pinu s číslom 2, Trig bude zapojený napríklad do pinu číslo 7 (GPIO 4), Echo bude zapojené napríklad do pinu číslo 11 (GPIO 17) a GND napríklad do pinu Ground 39. Pri zapájaní sa nesmie zabudnúť na rezistory.

Poslednými dvoma komponentami, ktoré je potrebné zapojiť, sú modrá a červená LED dióda. Obe budú zapojené rovnakým spôsobom, rozdiel bude len v tom, do akých pinov na Raspberry Pi. Každá LED dióda má dva póly plus a mínus, pričom plus bude zapojené do niektorého z GPIO pinov a mínus bude zapojené do niektorého z Ground pinov. Kvôli tomu, aby sa dióda z dôvodu pre ňu príliš vysokého napätia elektrického prúdu nepoškodila, je na cestu z plus do GPIO pinu nutné prostredníctvom rezistora aplikovať aj odpor o veľkosti 470 ohmov. LED diódy budú zapojené podľa nasledujúcej schémy:



Obrázok 4.4: Obvod: RPi + 2x LED

Plus modrej diódy bude zapojené napríklad do Raspberry Pi pinu číslo 22 (GPIO 25) a mínus modrej diódy napríklad do pinu Ground 20. Plus červenej diódy bude zapojené do Raspberry Pi pinu číslo 32 (GPIO 12) a mínus napríklad na Ground 30.

Po zapojení všetkých komponent by bolo vhodné otestovať ich správne fungovanie. Na to je však potrebné programovo nastavovať GPIO piny na Raspberry Pi, preto nasleduje kapitola, ktorá to umožní.

4.2 Vývoj zariadenia

Po zapojení všetkých elektronických komponent sa môže prejsť na samotné programovanie zariadenia. Nasledujúci text pojednáva o úkonoch, ktoré bolo potrebné vykonať, aby systém podporoval všetky funkcie, ktoré boli výstupom kapitoly 3.1.1 Funkcionalita zariadenia.

V tomto bode je nutné podotknúť, že Raspberry Pi 3A+, ktoré bolo na realizáciu použité, bolo dodané s už predinštalovaným operačným systémom Raspbian GNU/Linux 9 (stretch) [13], ktorý poskytuje všetko potrebné.

4.2.1 Pripojenie na internet

Aby bolo možné s Raspberry Pi pracovať, vstavaným USB portom sa k nemu pripojí USD hub s externou klávesnicou a myšou a prostredníctvom HDMI vstupu sa pripojí monitor.

Provou kľúčovou úlohou je pripojenie zariadenia na internet. To je však vďaka predchádzajúcemu kroku veľmi jednoduché. Na pripojenie stačí v hornej lište systému kliknúť na ikonu WiFi, vybrať sieť a zadať heslo. Ak všetko prebehne v poriadku, zariadenie je úspešne pripojené na internet a zároveň si túto sieť uloží pre budúce automatické pripojenie, čo je pre tento projekt v aktuálnej verzii veľmi dôležité.

4.2.2 Vzdialený prístup

Aby na vývoj nebolo nutné používať externú klávesnicu a myš, Raspberry Pi je možné po pripojení na internet sprístupniť vzdialene a programovať tak v komfortnejšom prostredí bežne používaného notebooku, ktoré so sebou prináša množstvo výhod.

Na tento úkon bola použitá služba Dataplicity [14], ktorá umožňuje spárované zariadenia spravovať a ovládať vždy, keď sú pripojené na internet. Funguje to tak, že na cieľovom zariadení sa najprv nainštaluje Dataplicity agent, ktorý následne naviaže klientom iniciované zabezpečené HTTPS spojenie medzi zariadením a Dataplicity IoT Routerom, pričom obíde bežné routre a firewally. [14]

Na nainštalovanie tohto agenta a pridanie nového zariadenia do služby Dataplicity sa stačí zaregistrovať a kliknúť na tlačítko ADD DEVICE. Tým sa vygeneruje príkaz podobný nasledujúcemu, pričom zdrojový kód tohto skriptu je voľne dostupný k nahliadnutiu na stránkach služby:

```
curl https://www.dataplicity.com/tyjmdidb.py | sudo python
```

Následne ho stačí skopírovať alebo vpísať do terminálu Raspberry Pi a spustiť tak inštaláciu. Po dokončení procesu je so spárovaným zariadením možné okamžite pracovať a to priamo v internetovom prehliadači prostredníctvom remote shellu, ktorý služba integrovala do ich webového rozhrania.

Vďaka tomu je teraz možné od Raspberry Pi všetky periférie odpojiť, programovať vo vlastnom prostredí a kód na zariadenie dostávať pomocou tejto služby v kombinácii s klasickým gitom.

4.2.3 Programovací jazyk Python

Na implementáciu logiky systému bol použitý programovací jazyk Python, ktorý je zvyčajne v Unixových operačných systémoch už predinštalovaný a výnimkou nebol ani operačný systém použitého Raspberry Pi. [15]

Python je považovaný za jazyk vedy a je už cez 20 rokov používaný najmä v sektoroch ako priemysel, služby a veda a technika, v ktorých pomáha naplniť širokú škálu najrôznejších požiadaviek. Používajú ho dokonca aj organizácie ako Google alebo NASA.

Je klasifikovaný ako skriptovací jayk, pretože na vygenerovanie strojového kódu nepotrebuje prekladač. Používa interpreter na vytvorenie byte-kódu, ktorý je multiplatformný. [16]

Programy napísané v tomto jazyku je možné spustiť pomocou kľúčového slova python a názvu súboru, ktorý musí mať príponu **.py**. Napríklad:

```
python CUBE.py
```

4.2.4 Implementácia modulov

Na implementáciu kompletnej logiky zariadenia bola na Raspberry Pi na ceste **/home/pi/** vytvorená dedikovaná zložka s názvom **CUBE**, ktorá bude obsahovať všetok kód a dáta, potrebné na plnohodnotné fungovanie celého systému.

V tejto zložke bolo vytvorených 12 samostatných modulov. Za modul je v rámci tejto práce považovaný každý súbor s príponou **.py** (napísaný v programovacom jazyku Python), ktorý sa nachádza v hlavnej zložke **CUBE**, a je ho vďaka tomu možné jednoducho importovať do ktoréhokoľvek iného pomocou kľúčového slova **import** a uvedenia názvu súboru (bez prípony). Napríklad:

```
import ULTRASOUND
```

Každý z nich má svoju úlohu, implementuje určitú časť logiky systému a v prípade potreby využíva služby ostatných modulov. Tento prístup bol zvolený najmä kvôli udržateľnosti a prehľadnosti kódu.

Moduly **CUBE** a **SERVER** sú hlavnými modulmi systému, ktoré sa po poslednej fáze vývoja zariadenia budú spúšťať vždy pri štarte Raspberry Pi.

Moduly **MEASUREMENTS**, **SCHEDULE** a **HISTORY** sú modulmi, ktoré zapúzdrujú logiku spojenú s vykonávaním meraní objemu vody v zásobníku, s vyhodnocovaním potreby automatickej zálievky a so správou histórie udalostí.

Moduly **PUMP**, **ULTRASOUND** a **LED** sú hardvérovými modulmi, z ktorých každý ovláda výhradne jednu z troch hardvérových komponent, ktoré sú súčasťou systému – tzn. čerpadlo, ultrazvukový senzor a LED diódy.

No a napokon moduly **TIMES**, **FILES** a **DATA** sú pomocnými modulmi na prácu s dátumami, súbormi a dátami.

Okrem toho existuje aj modul **APNS**, ktorý implementuje funkciu zaslania notifikácie s výzvou na doplnenie vody na mobilný telefón užívateľa.

Modulu **SERVER** sa bližšie venuje podkapitola 4.2.6 Server a modul **DATA** je bližšie popísaný už v nasledujúcej podkapitole 4.2.5 Manipulácia s dátami.

4.2.5 Manipulácia s dátami

Pre dáta bol vyhradený samostatný priečinok príznačne nazvaný DATA, v ktorom sa nachádza 5 súborov vo formáte **txt**, ktoré uchovávajú všetky dôležité

informácie, ktoré zariadenie pre svoje plnohodnotné fungovanie potrebuje. Ide o súbory: **cube.txt**, **now.txt**, **schedule.txt**, **history.txt** a **notification.txt**.

V súbore **cube.txt** sú uložené údaje ako napríklad maximálny objem zásobníku vody a maximálna a minimálna možná vzdialenosť hladiny vody.

V súbore **now.txt** sú uložené údaje o aktuálnom stave zásobníka vody ako napríklad aktuálna vzdialenosť hladiny vody, z nej dopočítaný aktuálny objem vody a jeho percentuálne vyjadrenie k objemu maximálnemu možnému.

V súbore **schedule.txt** sú uložené údaje vybraného harmonogramu zalievania, ktorými sú dátum a čas najbližšej zálievky, veľkosť zálievky a počet dní, ktoré sa majú pri vypočítavaní ďalšieho dátumu zálievky preskočiť.

Do súboru **history.txt** sa ukladajú všetky dôležité uskutočnené udalosti ako napríklad inicializácia zariadenia, manuálna zálievka, automatická zálievka, nemožnosť uskutočniť automatickú zálievku a dosiahnutie kritickej hladiny vody v zásobníku.

No a napokon v súbore **notification.txt** je uložený príznak o potrebe odoslania notifikácie s výzvou na doliatie a token z mobilného telefónu, ktorý sa pri jej odosielaní použije.

Aby sa so všetkými týmto súbormi a ich dátami pracovalo v moduloch jednoducho, existuje medzi nimi aj už spomínaný modul **DATA**, ktorý pre každú existujúcu premennú, ktorá je uložená v niektorom z vyššie uvedených súborov, implementuje funkciu načítania a v prípade, že to dáva zmysel, aj funkciu uloženia. Nasleduje ukážka časti kódu z modulu **DATA**:

```
import FILES

FILENAME_NOW = /home/pi/CUBE/DATA/now.txt
ROW_VOLUME = 11

def load_volume():
      return int(FILES.loadline(FILENAME_NOW, ROW_VOLUME))

def save_volume(volume):
      FILES.saveline(FILENAME_NOW, ROW_VOLUME, str(int(volume)))
```

4.2.6 Server

Jednou z najdôležitejších úloh tejto práce bolo zabezpečiť spoľahlivú komunikáciu medzi zariadením a mobilnou aplikáciou, pretože bez nej by nebolo možné zariadenie monitorovať ani ovládať pomocou mobilného telefónu. Aby vzdialený

prístup možný bol, na zariadení musí bežať server, ktorý bude odbavovať dotazy z mobilnej aplikácie.

Na zariadení bolo implementované REST (RESTful) API. Tento pojem je skratkou pojmu REpresentational State Transfer a ide architektúru rozhrania pre distribuované systémy, ktorá vznikla v roku 2000 a v značnej miere sa používa dodnes. Predstavuje spôsob ako jednoducho komunikovať so serverom pomocou jednoduchých HTTP volaní. [17]

Toto REST API je implementované v súbore API.py v jazyku Python za použitia knižnice Flask. [18] Pripravených je celkom 5 endpointov, ktorými sú: deviceInit, getDevice, manualPouring, updateSchedule a saveToken.

Na spustenie servera bol napísaný už spomínaný modul SERVER, ktorého jedinou úlohou je spustiť shell script s názvom **startServer.sh**, ktorý vykoná nasledujúce tri bash príkazy:

```
sudo fuser 80/tcp -k
export FLASK_APP=/home/pi/CUBE/API.py
sudo -E flask run --host=0.0.0.0 --port=80
```

Najprv je nutné uvoľniť port 80 ukončením všetkých procesov, ktoré ho používajú. Potom je potrebné aplikáciu implementovanú v súbore **API.py** vyexportovať do premennej **FLASK_APP**, aby ju nakoniec bolo možné použiť pri vykonaní tretieho príkazu, ktorý ju spustí na localhoste s využitím uvoľneného portu 80.

Aby bol server prístupný aj mimo lokálnej siete, tzn. ak užívateľ nie je doma v dosahu domáceho routeru, je nutné server sprístupniť aj pre vzdialenú komunikáciu. To je vďaka už spomínanej službe Dataplicity možné uskutočniť len jedným klikom myši. V internetovom prehliadači stačí na stránkach služby otvoriť remote shell daného zariadenia a v pravom postrannom paneli zapnúť wormhole, ktorá server doteraz bežiaci na localhoste sprístupní cez server Dataplicity na unikátnom URL, ktoré služba sama vygeneruje. URL je vždy v nasledujúcom tvare:

https://[unikátny kód zariadenia].dataplicity.io

Od tohto momentu je možné všetky dotazy na server posielať na túto URL.

4.2.7 Bootovanie

Po kompletnom dokončení celej logiky je nutné, aby sa pri vypnutí alebo výpadku elektrickej energie a opätovnom zapnutí zariadenia vždy znovu spustil hlavný modul **CUBE** s programom zalievania a hlavný modul **SERVER** s obsluhou endpointov.

Požadované chovanie je možné dosiahnúť tak, že na Raspberry Pi upravíme súbor **/etc/rc.local**. Tento skript sa spúšťa vždy po štarte systému, defaultne nevykonáva žiaden kód a môžeme do neho pridať nasledujúce príkazy na spustenie oboch požadovaných modulov:

```
sudo python /home/pi/CUBE/CUBE.py &
sudo python /home/pi/CUBE/SERVER.py &
```

Je dôležité podotknúť, že príkazy musia byť spustené s kľúčovým slovom **sudo** a teda ako superuser (resp. root) a na konci každého z nich musí byť pridaný symbol **&**, aby spustené programy bežali súbežne ako dva separátne procesy. [19]

Po dokončení tohto kroku je vhodné zariadenie reštartovať a overiť, že pri bootovaní sa tieto moduly skutočne spustia a fungujú správne. Ak všetko prebehlo v poriadku, zariadenie je možné týmto považovať za aktivované.

4.3 Vývoj mobilnej aplikácie

Mobilné aplikácie na operačný systém iOS je za normálnych okolností možné vyvíjať len na počítačoch s operačným systémom macOS od spoločnosti Apple. Tie umožňujú voľné stiahnutie a nainštalovanie vývojového prostredia Xcode, ktoré je spolu s programovacím jazykom Swift určené výhradne na vývoj aplikácií na zariadenia tohto ekosystému.

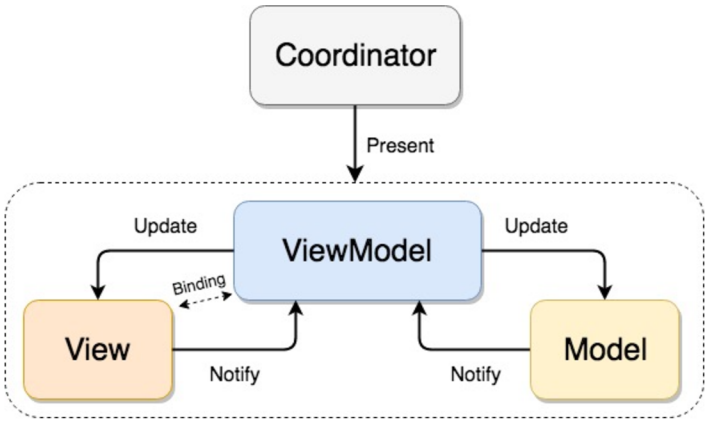
4.3.1 Programovací jazyk Swift

Swift je relatívne nový, intuitívny, kompilovaný, multi-paradigmatický, objektovo-orientovaný jazyk, ktorý v sebe spája to najlepšie z jazykov ako Objective-C, C#, Haskell, Ruby, Python, Java, JavaScript a mnohých ďalších [20]. Najnovšou verziou je Swift 5. [21]

4.3.2 Architektúra projektu

Voľba správnej architektúry je veľmi dôležitým aspektom každého projektu. Medzi hlavné vzroy patria MVC a MVVM. Pre tento projekt bola zvolená architektúra MVVM s rozšírením o Coordinator čoho výsledkom je MVVM+C.

MVVM+C je skratka pojmu Model-View-ViewModel+Coordinator. Tento typ architektúry spočíva v tom, že každá scéna (tzn. každá obrazovka aplikácie) obsahuje View, ktorým je typicky ViewController, ViewModel, ktorý obsahuje kompletnú biznis logiku scény, a Model, ktorý definuje štruktúru dát. Okrem týchto troch entít každá scéna navyše obsahuje Coordinator, ktorý je zodpovedný za navigáciu medzi jednotlivými scénami aplikácie. [22]



Obrázok 4.5: Model MVVM+C architektúry

Funguje to tak, že View a teda ViewController definuje a implementuje všetky viditeľné a interaktívne prvky scény ako Views, TextFields, Buttons a iné (v podstate celé UI). V prípade interakcie, napríklad ak užívateľ klikne na tlačítko, sa táto akcia vypropaguje do ViewModelu, kde je pre ňu definovaná konkrétna funkcia, v rámci ktorej sa vykonajú všetky potrebné logické úkony na dátach (Model) a následne sa prekreslí View, prípadne sa pomocou Coordinatora odnaviguje späť alebo na ďalšiu scénu.

4.3.3 Použité knižnice

V projekte boli použité len dve knižnice tretích strán, ktorými sú FuntastyKit a PromiseKit. Obe boli do projektu integrované použitím manažéra závislostí CocoaPods.

CocoaPods je manažér závislostí pre projekty napísané v programovacích jazykoch Swift a Objective-C, ktorý sprostredkúva cez 61 tisíc knižníc tretích strán,

a je použitý vo viac než troch miliónoch aplikácií. Okrem jednoduchej integrácie knižníc umožňuje elegantné škálovanie projektov. [23]

FuntastyKit je knižnica vytvorená spoločnosťou FUNTASTY Digital, s.r.o. obsahujúca zbierku užitočných Swiftových tried, funkcií a protokolov, ktoré vo svojich projektoch používajú. [24] Pre projekt WaterCubes boli použité najmä protokoly týkajúce sa architektúry a šablóny na vytváranie scén.

PromiseKit je knižnica, ktorá zjednodušuje programovanie asynchrónnych udalostí, čím dokáže programátora uvoľniť, aby sa sústredil na dôležitejšie veci. Knižnicu nie je príliš ťažké pochopiť, práca s ňou je pomerne jednoduchá a výsledkom je čistejší a čitateľnejší kód. Aktuálnou verziou je PromiseKit 6. [25] V projekte WaterCubes bola použitá najmä pri zaobchádzaní s API volaniami.

4.3.4 Implementovanie harmonogramov

V projekte mobilnej aplikácie WaterCubes v súbore **Library/Library.swift** je implementovaná knižnica harmonogramov, ktorá zohráva kľúčovú úlohu pri iniciálnom nastavení nového a pri upravovaní už pridaného zariadenia.

Každý harmonogram je reprezentovaný objektom **Schedule**, ktorý pozostáva z názvu, času zálievky a veľkosti zálievky v mililitroch. Tieto atribúty sa pri uložení posielajú na server zariadenia, ktorý ich v rámci endpointu **updateSchedule** spracuje a uloží do súboru **schedule.txt**, podľa ktorého sa potom vykonáva automatická zálievka. Bez údaju o čase a množstve vody by nebolo jasné kedy a na ako dlho sa má čerpadlo spustiť a automatizácia, ktorá je podstatou tohto zavlažovacieho systému, by nebola možná.

V prvej verzii má užívateľ na výber z harmonogramov 15 izbových rastlín zoradených podľa abecedy. Na nastavenie ich harmonogramov sa vychádzalo z knihy s názvom Veľká kniha izbových rastlín [26] a zo stránky umiestnenej na tomto odkaze [27]. Tento zoznam rastlín nie je problém kedykoľvek jednoducho a rýchlo rozšíriť o ďalšie druhy.

4.4 Zhotovenie púzdra

Púzdro zhotovíme na základe návrhu z kapitoly 3.3 Návrh púzdra. Púzdro sa vo výsledku skladá celkom z 10 samostatných častí, ktoré je potrebné vytlačiť na 3D tlačiarni jednu po druhej a následne ich do seba zložiť. Vo finále k sebe budú niektoré komponenty zlepené lepidlom.

4.4.1 Technológia 3D tlače

3D tlač je automatizovaný proces, pri ktorom sa z digitálnej predlohy (3D modelu) vytvára fyzický model. Funguje to tak, že objekt vzniká postupne, vrstva po vrstve, natavovaním tenkého prúžku plastového materiálu. Ide o tzv. aditívnu výrobu, kde sa materiál postupne pridáva, čo je presným protipólom k spôsobu subtraktívnej výroby, ktorá spočíva v obrábaní materiálu. [28]

4.4.2 Produkcia

Na tlač púzdra bola použitá 3D tlačiareň Creality Ender-3, ktorá je bežne dostupná online. Tento model je dodávaný ako stavebnica, takže po doručení je v prvom rade nutné ju postaviť.

Okrem tlačiarne je na produkciu potrebný aj filament – materiál v podobe lanka navinutého na kotúči, z ktorého sa bude výsledný objekt produkovať. Existuje mnoho typov, priemerov, farieb a dĺžok. Najbežnejšie používanými typmi filamentu sú však PLA a ABS. Na výrobu tohto púzdra bolo použité PLA.

Ďalším krokom je príprava 3D modelov. Všetky modely vzniknuté v nástroji Tinkercad je nutné vyexportovať ako súbory s príponou **.stl**. Tie však nie je možné použiť priamo a musia sa najprv spracovať programom, ktorý sa obecne nazýva slicer – napríklad Cura. Slicer je program, ktorý vezme digitálny model vo formáte **stl** a na základe vstupných nastavení, ako napríklad hrúbky jednej vrstvy alebo množstva vnútornej výplne, ho pretransformuje do mapy pohybu hlavice tlačiarne a uloží do súboru s príponou **.gcode**, s ktorým už tlačiareň dokáže pracovať. Všetky takto vygenerované súbory sa uložia na SD kartu, ktorá sa vloží do tlačiarne.

Po zložení tlačiarne, vložení filamentu, vložení SD karty s vygenerovanými súbormi už ostáva len predohriať tlačiareň a produkcia sa môže začať. Po postupnom vytlačení všetkých dielov púzdra a zasadení zariadenia je prototyp hotový a toto je výsledok:

Obrázok 4.6: Výsledný prototyp

4.5 Kalibrácia

Po realizácii všetkých vyššie spomenutých úkonov je pre správne fungovanie systému potrebné nastaviť ešte správne hodnoty pre premenné v súbore **cube.txt**, ktorými sú maximálny objem vody v zásobníku v ml a maximálnu a minimálnu vzdialenosť hladiny vody v cm s presnosťou na minimálne 1 destinné miesto.

Aby sme zistili, aké maximálne množstvo vody môžeme do zásobníku naliať, aby zariadenie fungovalo správne a čo najpresnejšie, do vyprázdneného zásobníku začneme liať vodu a prestaneme v momente, keď bude hladina od ultrazvukového senzora vzdialená 2,5 cm. Minimálna vzdialenosť pre správne fungovanie ultrazvukového senzora je totiž 2 cm a 0,5 cm volíme ako rezervu. Na priehľadnej časti púzdra fixkou sa zaznačí výška hladiny vody a prostredníctvom služby Dataplicity sa spustí pripravený modul **CALIBRATION**, ktorý uskutoční 10 samostatných meraní a po chvíľke vráti hodnotu, ktorá sa následne zapíše do súboru **cube.txt** na príslušný riadok **DISTANCE_FULL**.

Potom sa použitím funkcie manuálnej zálievky všetka voda odčerpá do nádoby s mierkou a opäť sa spustí modul **CALIBRATION** a výsledná hodnota sa zapíše na príslušný riadok **DISTANCE_EMPTY**.

Následne sa pozrieme na výšku hladiny odčerpanej vody v nádobe a odčítame z nej objem, ktorý predstavuje maximálne množstvo vody, ktoré je zariadenie schopné uchovávať. Tento údaj sa rovnako zapíše do súboru **cube.txt** na odpovedajúci riadok **VOLUME_MAX**. Tým je kalibrácia zásobníku vody hotová a systém je pripravený k použitiu.

```
VOLUME_MAX
250
DISTANCE_FULL
2.5
DISTANCE_EMPTY
6.7
```

Obrázok 4.7: Kalibrácia

5 Plány do budúcnosti

Táto kapitola popisuje ďalšie smerovanie projektu a je rozdelená na tri časti. Prvá sa venuje vylepšeniam, ktoré je možné uskutočniť priamo na zariadení, druhá sa zaoberá vylepšeniami, ktoré sa týkajú mobilnej aplikácie, a tretia popisuje vylepšenia, ktoré by mohli zdokonaliť púzdro.

5.1 Vylepšenia zariadenia

Zariadenie je možné vylepšiť mnohými spôsobmi, avšak tým najdôležitejším je nahradenie nepresného a nepraktického ultrazvukového senzoru iným, lepším riešením, ktorým by v tomto prípade mohol byť váhový senzor. Táto zmena by si vyžadovala pomerne veľký zásah do pôvodného modelu púzdra, ale pre projekt by bola tou najprínosnejšou vôbec.

Ďalším vylepšením by mohlo byť nahradenie logickej jednotky, ktorou je aktuálne Raspberry Pi 3A+, kopaktnejším a modelom Zero W, ktorý je menší o 2,6 cm, čím by sa uvoľnil priestor, ktorý by mohol byť využitý na zväčšenie zásobníku vody. Keďže má tento modu aj nižšiu cenu, jeho použitím sa taktiež znížia náklady na výrobu.

Po dokončení prototypu sa ďalej zistilo, že priehľadný filament použitý na tlač priehľadnej komponenty púzdra, ktorá mala zohrávať rolu sklíčka, cez ktoré by bolo vidno dovnútra a užívateľ by tak mohol voľným okom vidieť výšku hladiny vody, nie je v danej hrúbke až tak priehľadný ako sa predpokladalo. Vylepšenie by teda mohlo spočívať v tom, že do hornej odoberateľnej časti púzdra by bola zo spodu umiestnená biela LED dióda, ktorá by priestor komory osvetľovala a výšku hladiny vody by bolo možné vidieť aj cez túto polopriehľadnú komponentu.

Okrem týchto dvoch zásadných vylepšení by zariadenie mohlo byť doplnené o funkciu merania vlhkosti pôdy vlhkomerom. Senzor vlhkosti by mohol byť pripevnený na koniec trubice a meral by tak vlhkosť priamo v mieste zálievky. Dáta zo senzora by otvorili nové možnosti nastavení a umožnili tak komplexnejšie ovládanie systému.

5.2 Vylepšenia mobilnej aplikácie

Mobilná aplikácia aktuálne umožňuje výber harmonogramu zálievky len na základe názvu rastliny a to výberom prostredníctvom jednoduchého skrolova-

cieho pickeru, ktorý sa vysúva zo spodnej časti obrazovky, podobne ako klávesnica. Jedným z možných vylepšení by teda mohlo byť implementovanie samostatnej obrazovky s obrázkovou knižnicou rastlín, ich popisom a požiadavkami na starostlivosť.

Ďalším praktickým vylepšením, by mohlo byť zobrazovanie presného počtu dní do vyčerpania zásobníku vody na detaile u každého zo zariadení. Tento údaj je však možné získať len vtedy, keď je zariadeniu priradený harmonogram. V ňom je totiž uložený údaj o veľkosti automatickej zálievky a na jej základe a na základe aktuálneho objemu vody v zásobníku je možné vyčísliť počet zálievok, ktorý je s týmto množstvom vody možný a vďaka údaju o intervale zalievania, ktorý je vyjadrený premennou skipDays, je možné sa k žiadanému výsledku dopočítať.

5.3 Vylepšenia púzdra

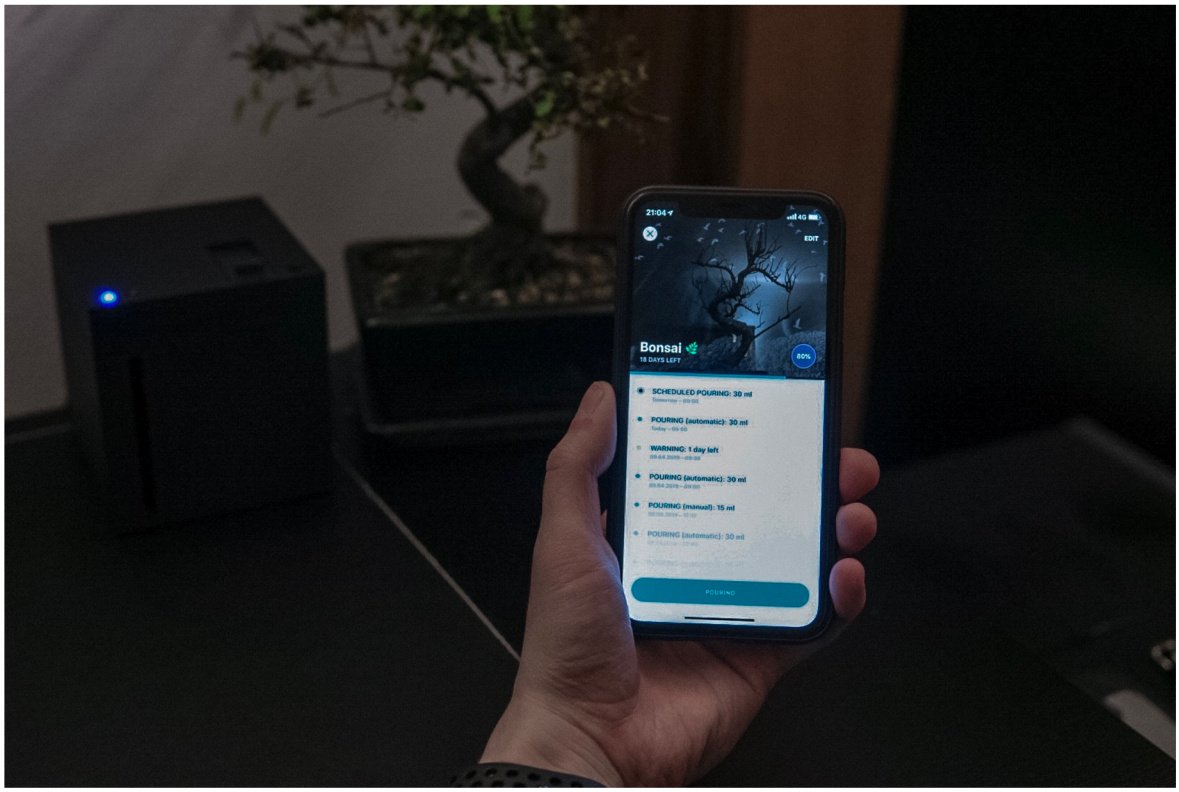
Aktuálny dizajn púzdra neumožňuje bezproblémové a rýchle odpojenie zariadenia od zdroja elektrickej energie, pretože Raspberry Pi je uložené vertikálne, konektor nie je v blízkosti steny a púzdro pre neho nemá vyhradený otvor, cez ktorý by sa kábel pripájal. Vylepšenie by mohlo spočívať v tom, že motherboard by bol umiestnený horizontálne a to v podstave púzdra, priamo pri stene s konektorom nasmerovaným von zadnou stranou kocky, a v tomto mieste by sa v púzdre nachádzal výrez, cez ktorý by bolo možné kábel odpájať a zapájať. Toto vylepšenie si síce vyžaduje reorganizáciu komôr, avšak bolo by obrovským prínosom. Užívateľ by tak mohol v momente, kedy je potrebné doplniť vodu, rýchlo a jednoducho zariadenie odpojiť, v ruke by mu ostala iba kocka, s tou by prešiel pod prúd vody, otvoril poklop a vodu by doplnil.

Ďalším praktickým vylepšením by mohlo byť zväčšenie otvoru na dolievanie vody, ktorý má aktuálne veľkosť 2x2 cm, čo je pomerne málo a pri dopĺňaní zásobníku sa môže stávať, že voda sa rozleje aj do okolia. Otvor by mohol byť bez väčších problémov rozšírený kľudne aj na 3x3 cm, čo by tento úkon výrazne spríjemnilo.

Do budúcna by bolo dobré zamyslieť sa aj nad tým, či je aktuálna veľkosť zariadenia skutočne dostatočná a nebolo by vhodné ju o 1 až 2 cm zväčšiť a to samozrejme vo všetkých troch smeroch, aby sa zachoval tvar kocky. Zväčšil by sa tak priestor zásobníka a tým aj maximálny objem vody, ktorý dokáže uchovávať.

6 Záver

Výstupom tejto diplomovej práce je automatizovaný zavlažovací systém pre izbové rastliny, zasadený do kompaktného púzdra s dôrazom na dizajn, spoľahlivo ovládateľný pomocou jednoduchej a užívateľsky prívetivej mobilnej aplikácie na platforme iOS, ktorá umožňuje pridávať, nastavovať, sledovať a ovládať viacero takýchto zariadení súčasne.



Obrázok 6.1: Zavlažovací systém WaterCubes

Vytvorenie tohoto systému si vyžadovalo osvieženie základných znalosti elektrotechniky, zoznámenie sa s platformou Raspberry Pi a programovacím jazykom Python, pochopenie základov programovania REST API, pokročilejšiu znalosť programovania aplikácií na platorme iOS v programovacom jazyku Swift vo vývojovom prostredí Xcode, naučenie sa modelovať 3D objekty a nadobudnúť zručnosť realizácie tlače na 3D tlačiarni.

Vo vývoji sa bude pokračovať a v blízkej dobe budú dokončené všetky spomenuté vylepšenia. WaterCube teraz čaká redizajn, nahradenie ultrazvukového senzoru váhovým a sprievodná mobilná aplikácia WaterCubes bude čoskoro vylepšená a zverejnená v obchode s mobilnými aplikáciami iTunes AppStore.

Literatúra

1. UNIVERZITA KOMENSKÉHO V BRATISLAVE. Automatizácia [online]. 2019 [cit. 2019-05-10]. Dostupné z: http://dai.fmph.uniba.sk/~filit/fva/automatizacia.html
2. GARDENA. Zavlažování o dovolené se zásobníkem na vodu [produkt online]. 2019 [cit. 2019-05-10]. Dostupné z: https://www.gardena.com/cz/produkty/zavlaha/zavlaha-o-dovolene/zavlazovani-o-dovolene---sada/900903201/
3. PLANTUI. Plantui 6 [produkt online]. 2019 [cit. 2019-05-10]. Dostupné z: https://plantui.com/products/plantui-6
4. Samozavlažovací kvetináč SENSE [produkt online]. 2019 [cit. 2019-05-10]. Dostupné z: http://www.tescoma.sk/samozavlacovaci-kvetinac-sense-pitny-rezim-vasich-kvetin-nechajte-na-nas-
5. CIRCUITO TEAM. Arduino vs. Raspberry Pi: Which board is best? [online]. 2018 [cit. 2019-05-10]. Dostupné z: https://www.circuito.io/blog/arduino-vs-raspberry-pi/
6. Raspberry Pi 3 Model A+ 64-bit 512 MB RAM [produkt online]. 2019 [cit. 2019-05-10]. Dostupné z: http://rpishop.cz/raspberry-pi-3a/1114-raspberry-pi-3-model-a.html
7. Eses mini čerpadlo [produkt online]. 2019 [cit. 2019-05-10]. Dostupné z: https://arduino-shop.cz/arduino/1271-cerpadlo-mini-ultra-tiche-dc-3-6v-120-l-h.html
8. Elektromagnetické relé [online]. 2019 [cit. 2019-05-10]. Dostupné z: https://cs.wikipedia.org/wiki/Elektromagnetické_relé
9. Měřič vzdálenosti ultrazvukový 5Pin HY-SRF05 pro Arduino [produkt online]. 2019 [cit. 2019-05-10]. Dostupné z: https://arduino-shop.cz/arduino/1735-meric-vzdalenosti-ultrazvukovy-5pin-hy-srf05-pro-arduino.html
10. AUTODESK INC. Tinkercad | Create 3D digital designs with online CAD [softvér]. 2019 [cit. 2019-05-10]. Dostupné z: https://www.tinkercad.com
11. RASPBERRY PI FOUNDATION. Raspberry Pi Documentation – GPIO [online]. 2019 [cit. 2019-05-10]. Dostupné z: https://www.raspberrypi.org/documentation/usage/gpio/
12. IOTGUIDER. Ultrasonic Distance Sensor in Raspberry Pi [online]. 2017-09-07 [cit. 2019-05-10]. Dostupné z: https://iotguider.in/raspberrypi/ultrasonic-distance-sensor-raspberry-pi/
13. Raspbian Documentation [softvér]. 2019 [cit. 2019-05-10]. Dostupné z: http://www.raspbian.org/RaspbianDocumentation
14. MACHINE FOREST LTD. Dataplicity [softvér]. 2019 [cit. 2019-05-10]. Dostupné z: https://docs.dataplicity.com
15. PYTHON SOFTWARE FOUNDATION. Python Documentation [softvér]. 2019 [cit. 2019-05-10]. Dostupné z: https://www.python.org/doc
16. JAKSON, Cody. Learn Programming in Python with Cody Jackson. 2018. 304 str. ISBN 13-9781789531947.

17. Representational State Transfer [online]. 2019 [cit. 2019-05-10]. Dostupné z: https://cs.wikipedia.org/wiki/Representational_State_Transfer
18. Flask Documentation [softvér]. 2019 [cit. 2019-05-10]. Dostupné z: http://flask.pocoo.org/docs/1.0/
19. DEXTER INDUSTRIES. Run a Program On Your Raspberry Pi At Startup [online]. 2019 [cit. 2019-05-10]. Dostupné z: https://www.dexterindustries.com/howto/run-a-program-on-your-raspberry-pi-at-startup/
20. LATTNER, Chris. Swift [online]. 2019 [cit. 2019-05-10]. Dostupné z: http://nondot.org/sabre/
21. APPLE INC. Swift [softvér]. 2019 [cit. 2019-05-10]. Dostupné z: https://developer.apple.com/swift/
22. VALDÉS, Lozano Daniel. iOS Architecture: MVVM-C [online]. 2017 [cit. 2019-05-10]. Dostupné z: https://medium.com/sudo-by-icalia-labs/ios-architecture-mvvm-c-introduction-1-6-815204248518
23. CocoaPods documentation [softvér]. 2019 [cit. 2019-05-10]. Dostupné z: https://cocoapods.org
24. FUNTASTY DIGITAL. FuntastyKit [softvér]. 2019 [cit. 2019-05-10]. Dostupné z: https://github.com/thefuntasty/FuntastyKit
25. PromiseKit [softvér]. 2019 [cit. 2019-05-10]. Dostupné z: https://github.com/mxcl/PromiseKit
26. AUGUSTYNOVÁ, Malgorzata. Veľká kniha izbových rastlín. Príroda, 2006. 191 str. ISBN 8007014306
27. Nenáročné izbové rastliny, ktorých pestovanie zvládne každý [online]. 2018 [cit. 2019-05-10]. Dostupné z: https://mojdom.zoznam.sk/cl/10075/1694533/Nenarocne-izbove-rastliny--ktorych-pestovanie-zvladne-kazdy
28. GONER, Jakub. Začíname s 3D tiskem. Computer Press, 2017. 216 str. ISBN 978-80-251-4876-1

A Príloha – zdrojové kódy

Zdrojové kódy zariadenia a mobilnej aplikácie sú k dispozícii na univerzitnom serveri v archíve tejto diplomovej práce.