

STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

Obor č. 18: Informatika

Chytrý skleník

Smart greenhouse

Autoři: Petr Mikenda

Škola: Střední průmyslová škola strojní a elektrotechnická a Vyšší odborná škola, Liberec 1, Masarykova 3, příspěvková organizace;
Masarykova 3, 460 01 Liberec 1

Kraj: Liberecký kraj

Konzultant: Ing. Marek Pospíchal

Liberec 2022

Anotace

Práce se zabývá vytvoření chytrého skleníku s možností vzdáleného přístupu. Výsledkem bude funkční prototyp řídicí jednotky společně s potřebnými senzory a akčními členy. Práce pojednává převážně o odolnosti proti poruchám a zneužitím.

Summary

This work deals with the creation of a smart greenhouse with the possibility of remote access. The result will be a functional prototype of the control unit together with the necessary sensors and actuators. The work will discuss resistance to failures and abuses.

Prohlášení

Prohlašuji, že jsem svou práci SOČ vypracoval/a samostatně a použil/a jsem pouze prameny a literaturu uvedené v seznamu bibliografických záznamů.

Prohlašuji, že tištěná verze a elektronická verze soutěžní práce SOČ jsou shodné.

Nemám závažný důvod proti zpřístupňování této práce v souladu se zákonem č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších předpisů.

V Liberci dne 15. 3. 2022

.....
Petr Mikenda

Obsah

Úvod.....	1
1 Definice požadavků na klimatické prostředí	2
2 Návrh hardwarových částí skleníku	3
2.1 Řídící jednotka.....	3
2.1.1 Arduino	3
2.1.2 ESP32	4
2.1.3 Raspberry	4
2.1.4 Porovnání.....	5
2.1.5 Výběr	6
2.2 Senzory.....	6
2.2.1 Porovnání a výběr	6
2.3 Akční členy.....	9
2.3.1 Porovnání a výběr	9
3 Vzdálený přístup.....	12
3.1 Metody	12
3.1.1 ArduinoDroid	12
3.1.2 Blynk.....	12
3.1.3 Webový server	12
3.2 Výběr	12
4 Blynk.....	14
4.1 Platby v aplikaci	16
4.2 Šifrování a bezpečnost komunikace Blynk	16
4.2.1 Blynk.Egdet (hardwarová knihovna).....	17
4.2.2 Blynk.Apps pro iOS a Android.....	17
4.2.3 Blynk.Cloud	17
5 Zvýšení spolehlivosti	19
5.1 Metody.....	19
5.1.1 Výpadek napájení	19

5.1.2	Odpojení řídicí jednotky od internetu.....	19
5.2	Hodnocení.....	19
6	Odolnost proti zneužití.....	20
6.1	Metody.....	20
6.1.1	„Nabourání“ komunikace	20
6.1.2	Získání zdrojového kódu	20
6.1.3	Sdílení našeho projektu.....	20
6.2	Hodnocení.....	21
7	Konstrukce skleníku	22
7.1	Návrh konstrukce.....	22
7.2	Materiál	22
7.3	Bezpečnost.....	22
8	Rozšíření.....	24
8.1	Ovládání více skleníků	24
8.2	Akční členy.....	24
8.3	Senzory	25
8.4	Ovládání.....	25
9	Programování řídicí jednotky	26
	Závěr.....	27
	Seznam zkratk a odborných výrazů.....	28
	Seznam obrázků.....	29
	Použité zdroje	30
A.	Seznam příložených souborů	I

Úvod

Když jsem nastoupil na tuto střední školu, tak už jsem věděl, že v budoucnu, pokud se budu věnovat IT oboru, nechci pouze sedět za počítačem a něco programovat. Právě toto je jeden z důvodů, proč dělám tuto odbornou práci. Na tuto myšlenku, dělat chytrý skleník, jsem přišel díky panu Ing. Pospíchalovi, protože jsme dostali na konci třetího ročníku úkol, ve kterém jsme si měli vybrat nějaký *IoT* projekt a k němu napsat jaké bychom tam vybrali senzory a akční členy. Vybral jsem si Smart greenhouse, protože můj mimoškolní koníček je pěstování rostlin, takže mě to velice zaujalo a celé prázdniny jsem přemýšlel, jak to udělám a jak to vlastně pojmu. Vzniknul z toho greenhouse, který bude mít podporu vzdáleného přístupu a bude automatizovaný.

1 Definice požadavků na klimatické prostředí

Proto, aby mohla rostlina růst, je potřeba zajistit vhodné podmínky pro její vegetaci. Tyto podmínky zajistíme tím, že je budeme monitorovat, a to díky senzorům, které budou ve skleníku. Základní parametry ve skleníku jsou teplota, vlhkost vzduchu a vlhkost zeminy. Změny těchto parametrů budou zajišťovat akční členy.

2 Návrh hardwarových částí skleníku

Abychom mohli začít navrhovat, který hardware použijeme, musíme si určit nějaké parametry celého projektu, jako je například cena, složitost nebo velikost samotného skleníku. Můj projekt je pojat jako prototyp, takže budeme používat základní senzory a rozměry celého skleníku nebudou přesahovat 1 m². Návrh, nebo výběr komponent se dělí na tři oddíly. Jako první a úplně nejdůležitější část je hlavní (řídící) jednotka. Zbylé dvě části už nejsou tak zásadní, protože to jsou senzory a akční členy a ty na pozdější chování skleníku nemají takový vliv jako řídící jednotka.

2.1 Řídící jednotka

Řídící jednotka, jak už samotný název napoví, je „mozek“ celého projektu, a právě proto bychom si měli dobře rozmyslet její výběr. Vybíral jsem ze tří typů řídících jednotek, a to byly Arduino, ESP32 a Raspberry. Když se na to budeme koukat z hlediska výkonu, tak na tom je nejlépe Raspberry a nejhůř Arduino.

2.1.1 Arduino

Arduino je nejpoužívanější platforma, co se týče IoT projektů. Je nejlepší pro ty, kteří by si chtěli začít doma navrhovat vlastní IoT produkty. Od digitálního teploměru po ovládání topení v celém bytě či domě, Arduino má spoustu možností, jak ho použít. První model byl představen v roce 2005. Díky snadnému ovládání a programování Arduina, byl jeho původní smysl pro umělce, kteří chtěli vytvářet interaktivní projekty.

Arduino nabízí spoustu modelů s různými procesory, vstupy anebo rozšířeními. Nejpoužívanější z nich je Arduino UNO, ale zde budu porovnávat výkonnější a dražší model Arduino Zero.



Obrázek 1 Arduino Zero (1)

2.1.2 ESP32

ESP32, na rozdíl od Arduina, není už pro ty, kteří o IoT nebo programování nic nevědí, ale pro ty, kteří se už v tomto odvětví nějakou dobu pohybují. ESP32 bylo představeno v roce 2016, má výhodu v tom, že je Open-Source a byl navržen pro *DIY* projekty, které poté mohou sloužit pro komerční využití.



Obrázek 2 ESP32 (2)

2.1.3 Raspberry

Raspberry, jak už jsem zmínil, je z těchto tří jednotek nejvýkonnější a má výhodu v tom, že na něm můžeme spustit samotný operační systém a je už spíš brán jako počítač než jako řídicí jednotka pro IoT projekty.

Raspberry bylo poprvé uvedeno na trh v roce 2012 jako jednodeskový počítač. Bylo navrženo pro studenty, kteří se začínali učit základy v počítačové vědě, ale i přesto si našlo místo v IoT projektech.



Obrázek 3 Raspberry Pi 4 (3)

2.1.4 Porovnání

Hlavní výhodou ESP32 je to, že má v základní verzi v sobě zabudovaný wifi a Bluetooth modul, to Arduino nemá, pokud nedokoupíme přídatný modul.

	Procesor	Frekvence	Flash paměť	Vstupy/výstupy	Wi-Fi
Arduino Zero	SAMD21	48 MHz	256 kB	20 digitálních vstupních a výstupních pinů, 6 analog. vstupů a 1 analog. výstup	při dokoupení modulu
ESP32	Tensilica Xtensa LX6	160 nebo 240 MHz	4, 8 nebo 16 MB	38 až 77 vstupních a výstupních pinů	zabudovaná
Raspberry Pi 4	Broadcom BCM2711	1.5 GHz	maximální 8 GB		při dokoupení modulu

Už ze samotných parametrů je vidět to, že výkonově na tom je nejhůř Arduino a nejlépe Raspberry. Arduino je dobré pro real-time kontrolu vstupů a výstupů, ESP32, pokud potřebujeme Bluetooth nebo wifi komunikace, toto se většinou používá při ovládání jednotky telefonem nebo pro vzdálenou kontrolu. Raspberry je z těchto tří nejvýkonnější a zároveň nejuniverzálnější. Pokud potřebujete silný Linux počítač, tak Raspberry se k tomu perfektně hodí.

2.1.5 Výběr

Jako řídicí jednotku pro můj projekt jsem si vybral ESP32. Vybral jsem si ho hlavně díky třem parametrům, což jsou cena, výkon a komunikace. Když budeme cenově porovnávat Arduino Zero a ESP32, tak zjistíme, že ESP32 je přibližně 5krát levnější, přičemž ESP32 je výkonnější. Část mojí práce se zabývá vzdálenou správou skleníku, proto jsem bral v potaz to, že ESP32 má zabudovaný modul pro komunikaci přes Wifi a Bluetooth.

2.2 Senzory

Výběr senzorů už není tak složitý, protože není moc na výběr. Při definici požadavků na klimatické prostředí jsem uvedl, že potřebuji senzor na teplotu vzduchu, vlhkost vzduchu a vlhkost zeminy. Skleník bude mít automatické zavlažování, takže potřebujeme senzor hladiny v nádrži na vodu a senzor okolního světla, kvůli automatickému zapínání růstového světla.

2.2.1 Porovnání a výběr

U senzorů samotné porovnávání není tak důležité, pokud se budeme bavit o tom, že tento projekt je pouze prototyp a nebude používán v opravdových podmínkách. Senzory mají pouze dva parametry přesnost a napájecí napětí.

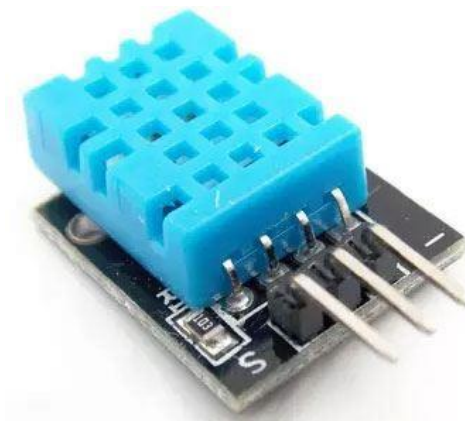
2.2.1.1 Vlhkost a teplota vzduchu

Když si do vyhledávače zadáte „senzor teploty a vlhkosti vzduchu“, tak jako první senzor uvidíte DHT11 nebo DHT22, a právě tyto dva senzory bych chtěl porovnat.

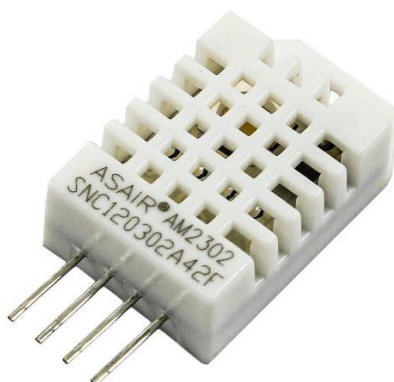
	Napájecí napětí	Teplotní rozsah	Rozsah vlhkosti	Knihovna
DHT11	3 až 5.5 V	0 až 50 °C	20 až 90 %	Adafruit DHT a Adafruit Unified Sensor
DHT22	3 až 6 V	-40 až 80 °C	0 až 100 %	Adafruit DHT a Adafruit Unified Sensor
LM35DZ	4 až 30 V	-55 až 150 °C	neměří vlhkost	analogRead()

BMP180	3,3 až 5 V	0 až 65	neměří vlhkost	Adafruit BME085 Adafruit Unified Sensor
--------	------------	---------	----------------	---

DHT11 a DHT22 jsou podle tabulky nejvhodnější, protože jsou oba senzory v jednom a tím pádem nemusí řešit senzor vlhkosti vzduchu. DHT22 má lepší rozsahy a je i dokonce přesnější, ale na běžné měření zcela stačí DHT11.



Obrázek 4 Senzor DHT11 (4)



Obrázek 5 Senzor DHT22 (5)

2.2.1.2 Vlhkost zeminy

U tohoto senzoru jsem se soustředil na samotnou odolnost měřící sondy, protože u základních modelů není sonda z antikoročního kovu a doporučuje se její výměna každý rok pro stejně přesné měření jako na začátku, právě proto jsem se soustředil na tento faktor. Když se podíváte na základní modely senzorů s antikorozní sondou, tak zjistíte, že je pouze jeden, takže výběr tohoto senzoru byl jednoduchý.

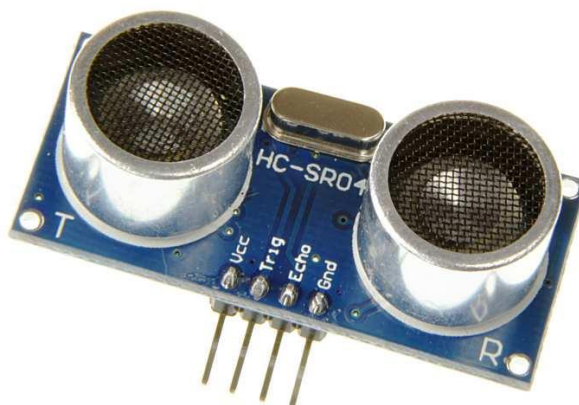


Obrázek 6 Senzor vlhkosti zeminy (6)

2.2.1.3 Hladina vody v nádrži

Tyto senzory se rozdělují do dvou skupin na odporové senzory a plovákové senzory. První ze zmíněných funguje tak, že posílá do řídicí jednotky odpor a my můžeme určovat hladinu. Plovákový senzor funguje jako spínač, tím pádem nemůžeme určit kolik je v nádrži vody, ale zjistíme, zda je voda v nádrži pod nebo nad určenou hladinou. Proto jsem měl v plánu použít první ze zmíněných senzorů, ale poté, co jsem ho objednal a zkusil, zjistil jsem, že tento senzor je nepřesný. Ing. pan Pospíchal, vedoucí mé odborné práce, mi poradil použít ultrazvukový senzor vzdálenosti, který dokáže měřit až do vzdálenosti 2 metrů.

Ultrazvukové senzory posílají zvukové vlny o frekvenci 40kHz a do řídicí jednotky pošlou čas, za který se tato vlna vrátí. Protože senzor nedokáže snímat povrch vody, je nutné vložit mezi hladinu vody a senzor polystyrén.



Obrázek 7 Ultrazvukový senzor vzdálenosti (7)

2.3 Akční členy

Akční členy převádějí informační část procesu na mechanickou například pokud má zemina moc nízkou vlhkost, vodní čerpadlo zajistí, aby se do zeminy dostala voda.

2.3.1 Porovnání a výběr

U akčních členů se budeme soustředit na výkon, neboť skleník o ploše 50 m², potřebuje jiný výkon, než skleník o ploše 1 m². Hlavním parametrem je tedy velikost skleníku a díky tomu, že se jedná o prototyp, jeho plocha nepřesáhne 300 cm².

2.3.1.1 Osvětlení skleníku

Nedílnou součástí růstu rostlin je sluneční svit, díky němu může v rostlinách probíhat fotosyntéza. V dnešní době už naštěstí existují náhrady slunečního svitu. Tato světla musí mít plné spektrum, což se docílí vysokou teplotou světla o hodnotě 1200 K, o vlnové délce 650nm. Použil jsem LED pásek, který má stupeň krytí IP65, kvůli vysoké vlhkosti. LED pásky mají vstupní napětí 12 V.



Obrázek 8 Plno spektrální LED pásek (8)

2.3.1.2 Zavlažování skleníku

Stejně jako světlo je pro rostliny důležitá voda. Zavlažování funguje tak, že přímo ve skleníku je umístěna nádrž s vodou a k ní je připojeno vodní čerpadlo. Z čerpadla poté vede hadička přímo do skleníku. U čerpadel jsou důležité dva parametry průtok a dosah sání. Mohli bychom ještě zvažovat vstupní napětí a stupeň krytí.

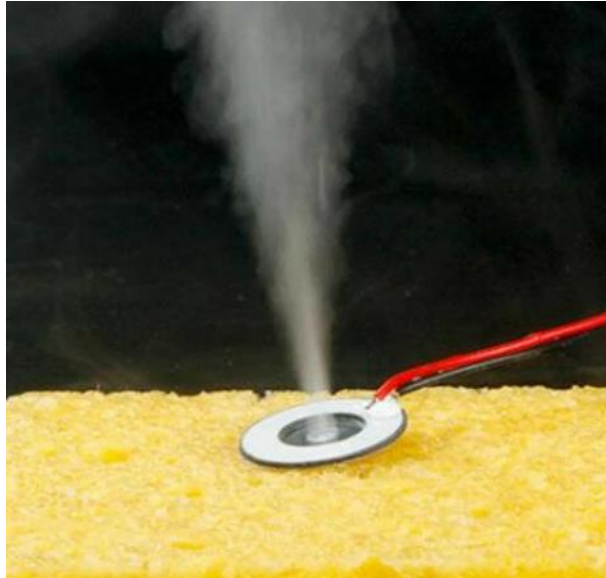
Vybral jsem si základní vodní čerpadlo, které se skládá z DC motoru. Jeho průtok je 1,5 - 2 l/min, což je dostačující pro tento skleník. Nádrž s vodou je umístěna přímo ve skleníku, čili dosah sání je plně dostačující.



Obrázek 9 Vodní čerpadlo (9)

2.3.1.3 Zvlhčovač vzduchu (difuzér)

Ve většině skleníků se pěstují zelené rostliny, které mají rády vysokou vlhkost vzduchu. Zvlhčovače vzduchu fungují na bázi piezoelektrického jevu. Ve zkratce, piezo kmitá vysokou frekvencí a pokud k jedné straně přiložíme zdroj vody z druhé strany se začne vytvářet vodní pára.



Obrázek 10 Piezo difuzér (10)

3 Vzdálený přístup

Při navrhování *IoT* řešení nesmí chybět vzdálená kontrola nebo ovládání, a to hlavně skrze mobilní telefon.

3.1 Metody

3.1.1 ArduinoDroid

ArduinoDroid funguje na principu USB On The Go (*OTG*). Arduino stačí připojit přímo k telefonu přes USB kabel. Tento princip má jednu velkou výhodu, a to že nepotřebujeme, aby jednotka měla modul Wifi nebo Bluetooth. Jelikož má ESP32 oba moduly zabudované přímo na desce, tak se tento způsob komunikace úplně nehodí.

3.1.2 Blynk

Blynk je služba. Jeho jednoduchost a flexibilita z něj dělá skvělý způsob na spouštění událostí na řídicí jednotce. Vyžaduje ke spuštění připojení k internetu, neboť používá svůj vlastní server. Velkou výhodou Blynk je to, že má podporu téměř každé vývojové desky. Díky intuitivnímu prostředí je opravdu jednoduché v Blynk cokoliv navrhnout.

3.1.3 Webový server

ESP32 také podporuje vytváření webového serveru stejně jako Arduino. Toto řešení se dnes už tolik nepoužívá, jelikož si musíte celé ovládací prostředí naprogramovat sami.

3.2 Výběr

Pro vzdálenou komunikaci jsem si vybral mobilní službu (aplikaci) Blynk, protože vytváření ovládacích panelů je velmi intuitivní. Prostor si sami navrhnete rozložením daných prvků jako jsou tlačítka, informační displeje, posuvníky a tak dále. Samotná řídicí jednotka komunikuje se vzdáleným (nebo lokálním) serverem, ze kterého potom samotná aplikace zobrazuje data na telefonu. Pokud budeme komunikovat přes vzdálený server, můžeme ovládat jednotku odkudkoliv, kde máme přístup k internetu. Má to jednu

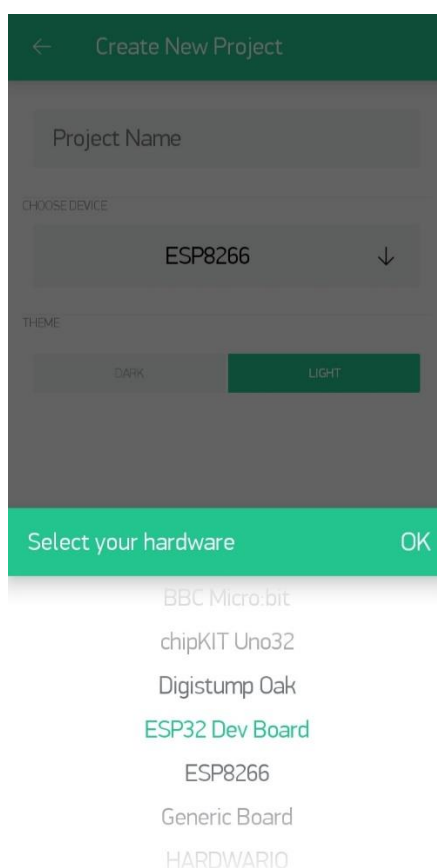
nevýhodu, kterou je zabezpečení. Pokud máme komunikaci se serverem, který není v lokální síti (tj. řídicí jednotka je připojena „ven“), tak je tu zde poměrně vysoké riziko toho, že se nám někdo připojí k naší jednotce, zcizí nám důležitá data a může je zneužít v náš neprospěch.

4 Blynk

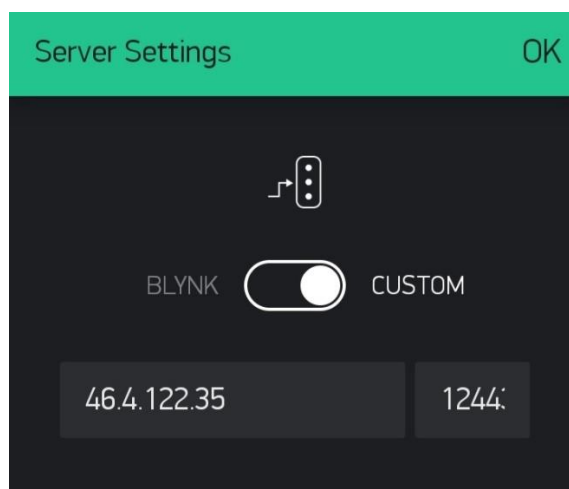
Protože jsem si Blynk vybral jako komunikační službu mezi mobilním telefonem a řídicí jednotkou, tak je určitě důležité si ji trochu více rozebrat.

Do aplikace se zaregistrujete a zde si zvolíte, zdali chcete svůj vlastní server nebo Blynk server. Pokud zvolíte vlastní server zadáte jeho IP adresu a port. Poté už budete přihlášení a můžete si začít vytvářet svoje vlastní projekty.

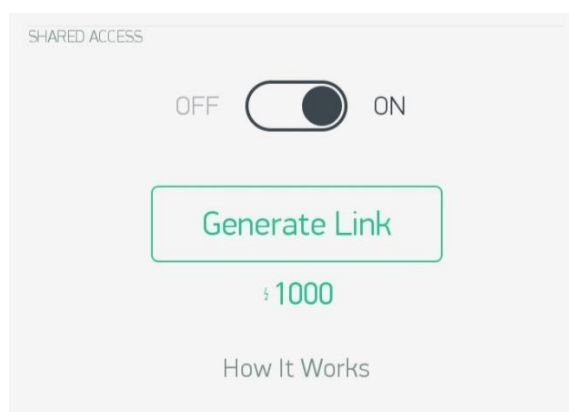
Při vytváření projektu zvolíte, jakou používáte řídicí jednotku, v mém případě je to ESP32. Do emailu vám poté přijde tzv. OAuth kód, který zadáte do zdrojového kódu řídicí jednotky. V nastavení si poté můžete zapnout tzv. „shared acces“, což vygeneruje QR kód a pomocí tohoto kódu se může kdokoli připojit na vaši jednotku a ovládat ji, pokud si stáhne aplikaci Blynk do mobilního telefonu.



Obrázek 11 Vytváření nového projektu v aplikaci Blynk



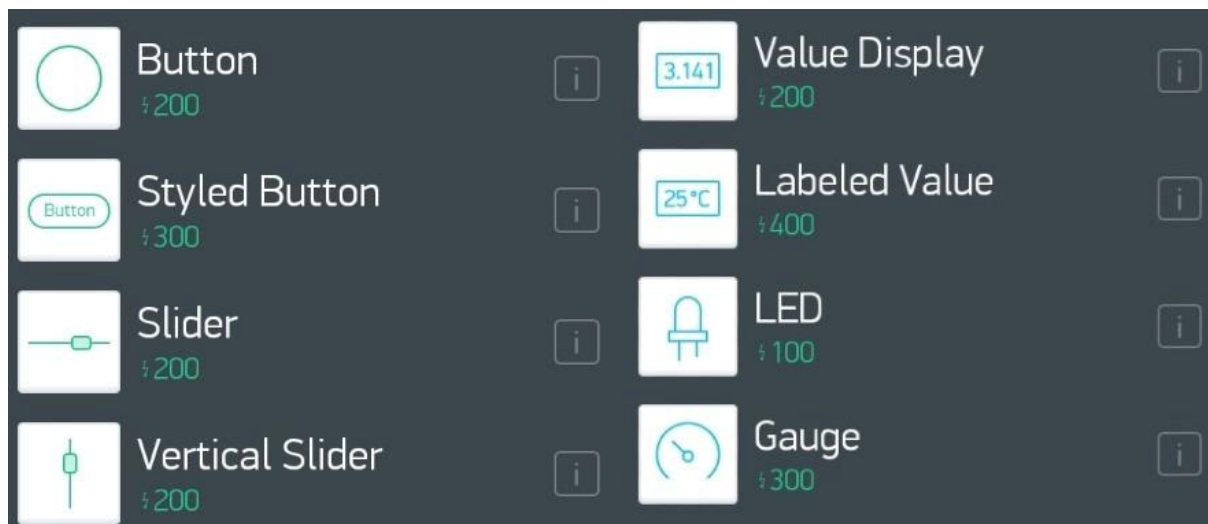
Obrázek 12 Nastavení vlastního serveru v aplikaci Blynk



Obrázek 13 Zapnutí sdílení našeho projektu s ostatními v aplikaci Blynk

Nyní už přichází hlavní část, a to je přidávání ovládacích prvků. Máte na výběr ze 49 různých zobrazovacích, ovládacích a řídicích komponent. Dělí se do 7 kategorií, což jsou kontroléry (tlačítka, displeje), notifikace, správce zařízení, vstupní (pro vkládání parametrů uživatelem), mobilní senzory a poslední kategorií jsou ostatní. Je zde i možnost zapisování a sledování hodnot pomocí grafů, které zaznamenávají informace a zapisují je do databáze, kterou poté můžete vyexportovat.

Komponenty nastavujeme přiřazením pinu, ze kterého chceme informace sledovat, nahrávat nebo pouze zapínat a vypínat. Zde jsou na výběr dva typy pinů fyzické a virtuální. Virtuální piny používáme pro zobrazování dat v mobilní aplikaci z *GPIO* pinů (fyzické) nebo proměnných, které jsou v programu. Fyzické piny používáme pro zapínání nebo vypínání těchto *GPIO* pinů.



Obrázek 14 Ukázka Blynk komponent

4.1 Platby v aplikaci

Jak jsem zmiňoval Blynk funguje, tak že komunikaci mezi jednotkou a mobilním telefonem zajišťuje server. Zde vzniká problém, protože k tomu, abyste měli neomezený režim v přidávání komponent, si musíte koupit tzv. energie. V základu dostanete 2000 energií, to stačí přibližně na 3 tlačítka, výpisový graf a 2 displeje. Tento problém jde vyřešit třemi způsoby. Přímo v Blynk aplikaci si dokoupíte energie za 28 000 energií za 20 \$ US. Pokud nechcete za Blynk platit, můžete si udělat lokální server, který je bezplatný, ale má nevýhodu, neboť musíte mít zapnutý počítač, kdykoliv chcete Blynk používat. Třetím způsobem je zajistit si server hosting. Tento způsob jsem si zvolil, protože jsem ho sehnal bezplatně.

4.2 Šifrování a bezpečnost komunikace Blynk

Rozebereme nejdříve obecnou bezpečnost Blynk. Veškeré zprávy, které jsou odeslané přes Blynk jsou zabezpečeny a zašifrovány v případě, kdy používaný hardware podporuje *TLS*, který zajišťuje bezpečnost a šifrování dat posílaných přes internet. Blynk používá nejnovější bezpečnostní protokol *TLSv1.3*, ale pokud není tato verze podporována používá starší *TLSv1.2*. Porty, které používá pro komunikaci, jsou 443 a 80. První port ze zmíněných používá pro *TLS* připojení a druhý k jednoduchému připojení v případě, že daný hardware nepodporuje *TLS*. Většina zařízení podporuje *TLS* a k nim patří i ESP32.

V aplikaci si můžete nastavit, kdo může zobrazit vaše zařízení a data z něj. Každý uživatel aplikace Blynk by měl mít platnou emailovou adresu. Aplikace má v sobě zabudovaný ověřovací proces. Každé zařízení musí mít svůj jedinečný OAuth token a ID produktu.

Při popisu zabezpečení Blynk cloudu se budeme bavit o těchto třech částech.

- Blynk.Edgent, který je spuštěný na řídicí jednotce
- Blynk.Console a Blynk.App pro iOS a Android zařízení
- Blynk.Cloud běžící na serveru

4.2.1 Blynk.Edgent (hardwarová knihovna)

Blynk je nastaven tak, aby používal ve výchozím nastavení *TLS*, pokud však používaný hardware nepodporuje *TLS*, tak komunikuje přes nešifrované připojení po portu 80, jak je zmíněno v předchozí kapitole.

Výchozí autentizací hardwaru je zabezpečený token OAuth. Skládá se z 24 bajtů nebo z 64 bajtů v základním kódování. vypadá následujícím způsobem *xw7ITVneg1DifRRQuPGcA7f*. Tento token lze měnit postupem opětovaného poskytování.

4.2.2 Blynk.Apps pro iOS a Android

Blynk komunikuje zabezpečenými webovými sokety (*TSLv1.3* nebo *TSLv1.2*) a používá pro základní autentizaci email a heslo v mobilní aplikaci. Před přenosem na cloudový server jsou všechna hesla zašifrována na straně klienta. Nepřenášejí se a ani neukládají v prostém formátu. Na straně serveru se používá šifrovaný hash hesla.

Poté, co se uživatel 5krát neúspěšně přihlásí ze stejné IP adresy musí počkat 10 minut, aby se mohl přihlásit znovu.

4.2.3 Blynk.Cloud

Všechna data, která jsou přenášena mezi cloudem a databází jsou vždy šifrována. V privátní síti je tato databáze plně izolována, nemá přístup k internetu a lze k ní pouze přistupovat z privátní sítě.

4.2.3.1 Certifikáty

Blynk používá certifikáty Let's Encrypt pro připojení *TLS*. Tyto certifikáty jsou obnovovány každé dva měsíce.

4.2.3.2 Porty

Blynk nevyužívá žádné speciální porty. Komunikuje pouze přes port 443 při použití *TLS* a port 80 při jednoduchém spojení bez šifrování.

5 Zvýšení spolehlivosti

Díky tomu, že se ve skleníku nachází elektronické součásti, je potřeba si dát pozor na poruchy, které by mohly nastat v různých situacích jako je například výpadek napájení.

5.1 Metody

5.1.1 Výpadek napájení

Výpadky napájení jsou běžná věc, a proto by se na ně mělo myslet při samotném programování řídicí jednotky. Když máme například proměnnou, do které uživatel zadá procento vlhkosti zeminy, při kterém se má spustit automatické zalévání a dojde k výpadku napájení, proměnná se automaticky smaže a uživatel ji musí zadat znovu.

5.1.2 Odpojení řídicí jednotky od internetu

Je potřeba zajistit to, aby si jednotka řídila všechno sama a nesouvisela s Blynk aplikací. Myslím to tak, že mobilním telefon by měl být pouze jakýsi ovladač, čili pokud vypadne připojení, neměly by v řídicí jednotce nastat žádné problémy.

5.2 Hodnocení

Výpadek napájení jsem vyřešil tím, že všechny parametry, které jsou dynamické (tj. zadány uživatelem) zapisuji do paměti *EEPROM*, ve které zadané hodnoty zůstanou i po odpojení napájení. Při výpadku připojení se řídicí jednotka přepne do auto režimu, tento režim si uživatel může sám také zvolit v aplikaci Blynk. Auto režim je funkční pro osvětlení a zalévání. Zajišťují to senzory intenzity osvětlení a vlhkosti zeminy.

6 Odolnost proti zneužití

Při nasazování vzdálené podpory projektu je důležité vědět to, že zde vzniká určité bezpečnostní riziko, pokud je komunikace vedená vzdáleně (tj. není ve stejné síti jako server).

6.1 Metody

6.1.1 „Nabourání“ komunikace

Uživatelé berou *IoT* zařízení jako běžné domácí spotřebiče a vůbec nehledí na jejich zabezpečení nebo bezpečnost komunikace na internetu. Většina dnešních *IoT* zařízení komunikuje z vaší privátní sítě ven, aby se dala ovládat odkudkoli, a to činí z těchto zařízení terč. Pokud máte nezabezpečené síťové připojení s přístupem k internetu, ohrožujete svá osobní data. Díky nim totiž mohou třetí strany získat vzdálenou kontrolu nad vašim zařízením. Uvádí se, že až 98 % procent *IoT* zařízení používá zcela otevřenou komunikaci přes internet. Firmy, které vyrábějí *IoT* zařízení se snaží nastavovat stejná hesla pro všechny příležitosti, chtějí tím zvýšit jednoduchost ovládání, ale za cenu zabezpečení. Pro snížení spotřeb energie dokonce i některá zařízení nemají šifrování hesel, nebo zde není možno vytvářet účty s různými právy (admin, uživatel).

6.1.2 Získání zdrojového kódu

Díky tomu, že používáme vzdálenou kontrolu, je potřeba mít uvedené ve zdrojovém kódu informace se jménem a heslem připojení k Wifi. Pokud se tedy někdo dostane přímo k řídicí jednotce, nemá problém tyto informace získat a zneužít je. V ukázce ze zdrojového kódu můžete vidět, jak jsou informace o Wifi připojení snadno získatelné.

```
char ssid[] = "2.4ghz"; //jméno wifi připojení (SSID)
char pass[] = "EgHLLjjkLuj"; // heslo k připojení
```

6.1.3 Sdílení našeho projektu

Ve čtvrté kapitole jsem informoval o tzv. „shared access“. Po spuštění této funkce se vygeneruje QR kód, který obsahuje následující informace:


```
blynk://token/projpub/cgxYhei5Jinipldk-  
E0lt6KQnTSSNru7?username=petrmikenda%40seznam.cz&server=46.4  
.122.35&port=12443&dashId=981832969&dashTitle=Greenhouse
```

Jak si můžete všimnout v QR kódu je uložený email, IP adresa a port serveru. Pokud tedy získá tento kód neoprávněná osoba, může ovládat vaši řídicí jednotku nebo útočit na daný server. Další nebezpečnou věcí je, že se ani nemusíte do aplikace Blynk registrovat stačí na úvodní stránce pouze kód naskenovat. Pokud tedy zapínáte tuto funkci a máte v plánu ji využít, tak si dávejte pozor na to, komu svěřujete tento kód.

6.2 Hodnocení

Získání zdrojového kódu jde vyřešit fyzickým zabezpečením obalu řídicí jednotky. Řídicí jednotka v tomto projektu je umístěna na místě, kde nevzniká riziko odcizení zdrojového kódu.

Problémy se zabezpečením můžete vyřešit tím, že si budete například často aktualizovat hesla a snažit se o to, je mít co nejsložitější. Dvoufázová autentizace se dnes často používá, v praxi to znamená, že je účet spojený s telefonním číslem nebo dalším emailem, po nastavení už není tak snadné nabourat se do vašeho účtu.

7 Konstrukce skleníku

Součástí mé práce je konstrukce skleníku a obalu na řídicí jednotku. Táto část zabrala největší množství času, proto bych ji chtěl zmínit i v teoretické části mé odborné práce. Konstrukce mi trvala asi 50 hodin.

7.1 Návrh konstrukce

Už dlouho předtím, než jsem začal skleník konstruovat, jsem měl přibližnou představu, jak bude vypadat. Jakmile nastal čas konstrukce, tak jsem vzal papír a přibližně jsem si všechno načrtl. Skleník se skládá ze tří oddělených částí nádrž na vodu, kryt na řídicí jednotku s ostatní elektronikou a skleník se zeminou. Samotná nádoba na zeminu má tvar krychle s rozměry 40×40×40 cm. Vzadu ve spodní části se nachází nádrž na vodu, která má přibližný objem 8 litrů a má tvar kvádrů s šířkou 40 cm, na výšku a hloubku má 15 cm. Ve skutečnosti je 20 cm hluboká, ale 5 cm zasahuje přímo skleníku (nádoby se zeminou), kvůli zvlhčovači vzduchu. Nad nádrží se nachází obal na řídicí jednotku a veškerou elektroniku. Skleník se otvírá směrem nahoru.

7.2 Materiál

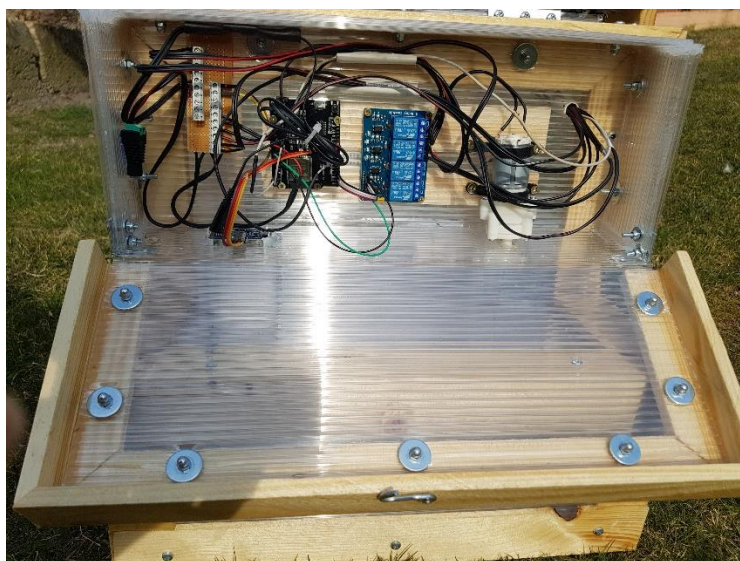
Skleníky se dnes vyrábějí převážně z polykarbonátových desek, takže jsem zvolil tento materiál, neboť je měkký a dobře se s ním pracuje. Pro spojování rohů jsem použil dřevěné L profily, které jsem vyřezal z hranolů. Stěny a profily jsou spojené dohromady šrouby a matkami. Tyto spoje jsem musel vyplnit silikonem, pro dosažení vodotěsnosti. Dřevěné profily jsem nalakoval kvůli zvýšení odolnosti.

7.3 Bezpečnost

Důležitou částí konstrukce je zajistit bezpečí pro uživatele. Proto je obal s řídicí jednotkou plně oddělen od skleníkové části i nádrže s vodou. Nejdříve jsem měl v plánu zabudovat zdroj napájení přímo do obalu řídicí jednotky. Došlo mi ale, že to není úplně dobrý nápad, pokud by nastala situace, kdyby se vyskytla voda v obalu, mohlo by dojít ke zkratu a následnému elektrickému úrazu 230 volty, případně vzplanutí. Raději jsem zvolil bezpečnější variantu a to 12 V. Samotný převod 230 V probíhá mimo skleník. V obalu se nachází pouze konektor, do kterého připojíme 12 V.



Obrázek 15 Hotová konstrukce skleníku



Obrázek 16 Ukázka obal s řídící jednotkou



Obrázek 17 Popis částí skleníku

8 Rozšíření

IoT projekty jsou dobré v tom, že to jde vždycky udělat lépe, něco přidat nebo lépe naprogramovat, proto bych chtěl v této kapitole zmínit pár zlepšení a rozšíření, která budou pouze teoretická.

8.1 Ovládání více skleníků

Toto rozšíření je vhodné, pokud bychom chtěli mít dva na sobě nezávislé sektory a ovládat je zvlášť. Pro dosažení je zapotřebí pouze zdvojit všechny senzory a akční členy. V aplikaci Blynk bychom potom také zdvojili všechny ovládací prvky. Toto rozšíření nemusí být pouze pro dva skleníky, ale můžeme rozdělit příčkou náš aktuální skleník, díky čemuž vzniknou dvě na sobě nezávislé komory. Samozřejmě to nemusí být pouze dvě komory, díky tomu, že ESP32 nabízí I2C sběrnici, je možné připojit na jeden pin až 127 sensorů, z toho tedy vyplývá, že možnosti tohoto rozšíření jsou téměř neomezené.

8.2 Akční členy

Pokud bychom chtěli dosáhnout naprosté nezávislosti skleníku na okolních podmínkách jeden z hlavních faktorů by bylo přidání topení. Zde si můžeme vybrat ze dvou způsobů zahřívání vzduchu. První z nich je infračervené zahřívání, což není úplně vhodné, protože je uzpůsobeno na přímé použití, a to rostliny nemají rády. Druhý typ je zahřívání proudem vzduchu, což je vhodnější, protože zde můžeme měnit intenzitu proudění a teplotu. Stačí pořídit ventilátor s chladičem a topné těleso.

Většina skleníků stojí na přímém slunci, kvůli dosažení co největšího dopadu světla, tím pádem, pokud je náš skleník malý, může vzniknout uvnitř velké dusno, což rozhodně rostlinám neprospívá, proto by bylo dobré zajistit odvětrávání. Zde můžeme využít způsob z minulého odstavce tím, že spustíme pouze samotný ventilátor, který zařídí proudění vzduchu a následné ochlazení vzduchu. Pokud nechceme zařizovat topení, tak je zde možnost pasivního chlazení. Stačilo by, aby se k víku skleníku připevnil píst, který by se následně připojil na relé a při překročení zadané teploty, by se víko pomocí pístu otevřelo.

V mém případě se nádrž na vodu nachází u skleníku, ale má „pouze“ 8 litrů, což není hodně, bylo by tedy vhodné zabudovat do skleníku čerpadlo, které bude spuštěno

poté, co dojde v nádrži voda a dopustí ji zpátky. Stačilo by stejné vodní čerpadlo jako už je ve skleníku. Pokud bychom měli stálý zdroj vody, mohli bychom použít elektrický uzávěr.

8.3 Senzory

Další z možností rozšíření je zajistit větší zabezpečení celého skleníku. Mohli bychom například přidat do skleníku senzor pohybu, který by po detekci pohybu zaktivoval kameru, která by začala nahrávat. Ve skleníku také může nastat požár, který by zaznamenal sensor kouře a upozornil by uživatele.

8.4 Ovládání

Ovládat tento skleník lze pouze mobilním telefonem, ale je zde možnost přidat k němu informační displej, kde by se například zobrazovaly nastavené parametry. Přidáním tlačítka už by bylo možné skleník ovládat, kdy by se muselo navrhnout jednoduché *UI*. Toto rozšíření je vhodné, pokud nechcete používat vzdálenou podporu, ale by bylo vhodné udělat kombinaci obou řešení.

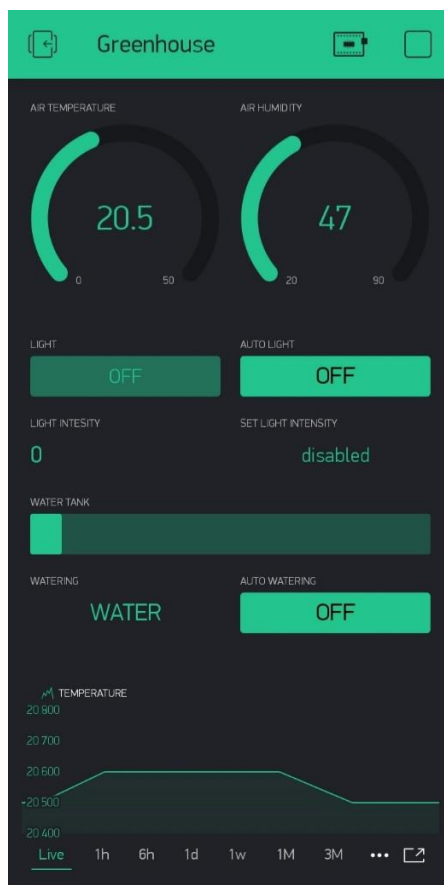
9 Programování řídicí jednotky

Po sestavení skleníku a zakomponování elektrických částí, jsem se přesunul na programování jednotky. První jsem začal s osvětlením. Automatický režim snímá hodnoty ze senzoru intenzity světla a poté je porovnává s hodnotou, kterou uživatel načetl přes Blynk. Manuální funguje jako tlačítko, které uživatel pouze zapíná nebo vypíná.

Hodnoty ze senzoru teploty a vlhkosti zeminy se zobrazují v aplikaci. Pomocí snímače vlhkosti zeminy, zde také může fungovat automatický režim. Snímá hodnoty a odesílá je do jednotky. Uživatel má na výběr ze 3 stupňů vlhkosti.

Posledním snímačem je ultrazvukový sensor, který má na starosti dostatek vody v nádrži. Jak už jsem zde informoval, musel jsem na vodní hladinu položit polystyrén, aby tento senzor fungoval. Poté pouze nasnímáme vzdálenost polystyrénu od čidla.

Pro programování jsem používal aplikaci Arduino IDE, základní aplikace, ve které píšete kód a poté ho nahráváte do řídicí jednotky. Neboť jsem si vybral programovací jazyk C/C++, tak je možnost využít i jiné programy jako je například Visual Studio Code.



Obrázek 18 Ukázka hotového projektu v aplikaci Blynk

Závěr

Práci jsem začal teoretickou přípravou senzorů, akčních členů a řídicí jednotky. Poté jsem vše objednal a vyzkoušel.

Zkoušení probíhalo připojením každého sensoru k řídicí jednotce a následného naprogramování tak, aby jednotka odesílala informace ze senzoru po sériové lince do počítače. Při zkoušení senzoru snímání hladiny vody jsem zjistil, že tento senzor je téměř nepoužitelný, takže jsem místo něj použil senzor vzdálenosti, který je mnohem spolehlivější. Když jsem si byl jistý, že je vše v pořádku, přesunul jsem se na testování a použití aplikace Blynk. Nejdříve jsem používal nativní server od Blynk a když vše fungovalo, začal jsem to samé zkoušet na mém vlastním Blynk serveru. Po odzkoušení jsem se přesunul na konstrukci skleníku.

Návrh skleníku jsem si načrtl na papír a začal jsem skládat postupně všechny části. Následně jsem skleník rozložil, abych mohl nalakovat dřevěné profily a utěsnit spoje. Poté jsem sestavoval elektronickou část a skleník dohromady.

Rozvrhl jsem si, jak elektronické komponenty rozmístit do obalu. Musel jsem brát v úvahu, že vodní čerpadlo musí být na místě, kde se k němu dají dobře připojit hadičky. Při vymýšlení zabudování difuzéru se objevil jeden zásadní problém. Když jsem tyto difuzéry objednával nevěděl jsem, že je k nim potřeba řídicí obvod, díky tomu nejsem schopný je do skleníku zakomponovat, v době odevzdání teoretické části.

Jak jsem již zmínil všechny senzory, akční členy a relé jsem otestoval, pomocí kódů, kterými jsem se inspiroval na internetu. Poté jsem k nim přidal kód, který zajišťuje komunikaci přes Blynk a začal jsem tyto části kódu dávat dohromady. Po naprogramování už mě čekalo samotné testování, které netrvalo dlouho. Zkusil jsem všechna tlačítka v aplikaci a všechny režimy, tak aby to fungovalo bez problémů.

Při programování automatického zavlažování nastal problém. Pokud jsem použil funkci `delay()` (tj. funkce, která pozastaví běh kódu a čeká), tak komunikace s Blynk byla nepoužitelná. Naštěstí jsem ale našel způsob, jak to vyřešit.

Díky této práci vím, že nechci být pouze programátor, který píše jen kód a pracuje se softwarem, neboť to chci mít spojené i s hardware. Chtěl bych se do budoucna více věnovat *IoT* oboru.

Seznam zkratk a odborných výrazů

IoT

Internet of Things – internet věcí, síť fyzických zařízení, které jsou vybaveny elektronikou a senzory

DIY

Do It Yourself – udělej si sám, používá se pokud mluvíme a projektech, které děláme doma nebo v neprofesionálních podmínkách

IP65

Ingress Protection – stupeň krytí, většinou se používá u elektroniku, a to hlavně u mobilních telefonů

TSL

Transport Layer Security – protokoly, které se využívají pro zabezpečenou komunikaci po internetu

EEPROM

Electrically Erasable Programmable Read-Only Memory – jedná se o typ paměti flash, která jde elektricky vymazat

I2C

Inter-Integrated Circuit – multi-masterová počítačová sběrnice, vyvinutá firmou Philips

UI

User Interface – uživatelské rozhraní, většinou grafické prostředí, pro lepší orientaci v ovládání

Seznam obrázků

Obrázek 1 Arduino Zero (1)	4
Obrázek 2 ESP32 (2)	4
Obrázek 3 Raspberry Pi 4 (3)	5
Obrázek 4 Senzor DHT11 (4)	7
Obrázek 5 Senzor DHT22 (5)	7
Obrázek 6 Senzor vlhkosti zeminy (5)	8
Obrázek 7 Ultrazvukový senzor vzdálenosti (6)	9
Obrázek 8 Plno spektrální LED pásek (7)	10
Obrázek 9 Vodní čerpadlo (8)	10
Obrázek 10 Piezo difuzér (9)	11
Obrázek 11 Vytváření nového projektu v aplikaci Blynk	14
Obrázek 12 Nastavení vlastního serveru v aplikaci Blynk	15
Obrázek 13 Zapnutí sdílení našeho projektu s ostatními v aplikaci Blynk	15
Obrázek 14 Ukázka Blynk komponent	16
Obrázek 15 Hotová konstrukce skleníku	23
Obrázek 16 Ukázka obal s řídící jednotkou	23
Obrázek 17 Popis částí skleníku	23
Obrázek 18 Ukázka hotového projektu v aplikaci Blynk	26

Použité zdroje

1. **reichelt elektronik GmbH & Co. KG.** Arduino Zero, SAMD21, USB Micro. *Reichelt*. [Online] [Citace: 26. Únor 2022.] <https://www.reichelt.com/de/en/arduino-zero-samd21-usb-micro-arduino-zero-p313442.html?r=1>.
2. **Mouser Electronics, Inc.** Espressif Systems ESP32-DevKitC-32UE. *Mouser Electronics*. [Online] [Citace: 26. Únor 2022.] <https://cz.mouser.com/ProductDetail/Espressif-Systems/ESP32-DevKitC-32UE?qs=GedFDFLaBXFguOYDKoZ3jA%3D%3D>.
3. —. Raspberry Pi RPI4-MODBP-8GB-BULK. *Mouser Electronics*. [Online] [Citace: 26. Únor 2022.] <https://cz.mouser.com/ProductDetail/Raspberry-Pi/RPI4-MODBP-8GB-BULK?qs=sPbYRqrBIVlrjhMid19vUA%3D%3D>.
4. **KONTAKT ELEKTRONIK spol. s r.o.** A-DHT11. *EMO*. [Online] [Citace: 26. Únor 2022.] <https://www.emo.cz/p/a-dht11>.
5. **ElectroPeak Inc.** ElectroPeak. *DHT22 Temperature Humidity Sensor*. [Online] [Citace: 26. Únor 2022.] <https://electropeak.com/sensor-dht22-1>.
6. **Netigo.** Capacitive Analog Soil Moisture Sensor. *Netigo*. [Online] [Citace: 26. Únor 2022.] <https://nettigo.eu/products/capacitive-analog-soil-moisture-sensor>.
7. **ECLIPSERA s.r.o.** Ultrazvukový měřič vzdálenosti. *Drátek*. [Online] [Citace: 26. Únor 2022.] <https://dratek.cz/arduino/846-eses-ultrazvukovy-meric-vzdalenosti-hc-04-pro-jednodeskove-pocitace.html>.
8. **Softima s.r.o.** LED páska GROW 12V. *Hadex*. [Online] [Citace: 23. Únor 2022.] https://www.hadex.cz/k039-100-led-pasek-grow-12v-10mm-plnospektralni-60x-led5050m-ip65-civka-5m/?gclid=CjwKCAiAg6yRBhBNEiwAeVyL0NmZFkhmSCY4yFnVwly1lBMQ1xxsy_r8WSjngA6LjelzytY4wpnsUgBoCCNcQAvD_BwE.
9. **ECLIPSERA s.r.o.** Vodní čerpadlo se silentbloky DC 6-12V R385. *Drátek*. [Online] [Citace: 26. Únor 2022.] https://dratek.cz/arduino/7521-vodni-cerpadlo-se-silentbloky-dc-6-12v-r385.html?gclid=CjwKCAiAg6yRBhBNEiwAeVyL0HV7gG9EKYTUXuWV-X4rqiUJPUOJk6WB6-u2xyf1aVgEH_Tg6UvohoCtGIQAvD_BwE.

-
10. **Fruugo.com Ltd.** Zvlhčovač vzduchu Driver Board Mist Maker Fogger. *Fruugo*. [Online] [Citace: 26. Únor 2022.] https://www.fruugo.cz/zvlhcovac-vzduchu-driver-board-mist-maker-fogger-ultrazvukove-atomizacni-disky/p-67824361-136174521?language=cs&ac=croud&gclid=CjwKCAiAg6yRBhBNEiwAeVyL00cCLuGhr4iPYAXNzCfCKBVHtWRtjwYOqNSppXd9fFvcA-BJjf1TCxoC4FQQAxD_BwE.
 11. **Ghosh, Abhishek.** ESP32 vs Arduino : How ESP32 is Different from Arduino. *The Customize Windows*. [Online] 16. Květen 2020. [Citace: 15. Leden 2022.] <https://thecustomizewindows.com/2020/05/esp32-vs-arduino-how-esp32-is-different-from-arduino/>.
 12. **Aufranc, Jean-Luc.** Know the Differences between Raspberry Pi, Arduino, and ESP8266/ESP32. *CNX software - Embedded systems news*. [Online] 24. Březen 2020. [Citace: 15. Leden 2022.] <https://www.cnx-software.com/2020/03/24/know-the-differences-between-raspberry-pi-arduino-and-esp8266-esp32/>.
 13. **Santos, R.** 9 Arduino Compatible Temperature Sensors. *RandomNerdTutorials*. [Online] 16. Červenec 2019. [Citace: 15. Leden 2022.] <https://randomnerdtutorials.com/9-arduino-compatible-temperature-sensors-for-your-electronics-projects/>.
 14. **Arduino.** Sensors&Environment. *Arduino*. [Online] [Citace: 13. Listopad 2021.] https://store.arduino.cc/collections/sensors-environment?page=3&grid_list=list-view.
 15. **Climate Control Systems Inc.** GREENHOUSE CONTROL SYSTEMS & TECHNOLOGY. *Climate Control System*. [Online] [Citace: 13. Listopad 2021.] <https://www.climatecontrol.com/blog/greenhouse-control-systems/>.
 16. **Autodesk, Inc.** Comparison Between Micro:bit and Arduino. *Instructables circuits*. [Online] [Citace: 13. Listopad 2021.] <https://www.instructables.com/Comparison-Between-Microbit-and-Arduino/>.
 17. **Buckley, Ian.** MUO. 6 Easy Ways to Connect Arduino to Android. [Online] 02. Únor 2017. [Citace: 16. Leden 2022.] <https://www.makeuseof.com/tag/6-easy-ways-connect-arduino-android/>.
 18. **Blynk Inc.** Security - Blynk Documentation. *Blynk*. [Online] [Citace: 24. Únor 2022.] <https://docs.blynk.io/en/blynk.cloud/security>.

19. **Techvice Sp. z o.o.** Are IoT Systems Too Easily Hacked, And What Would Be The Fix?
Techvice. [Online] 5. Říjen 2021. [Citace: 13. Březen 2022.]
<https://techvice.org/blog/popular/iot-systems-hacked/>.
20. **Arduino**. Language Reference. *Arduino*. [Online] [Citace: 13. Březen 2022.]
<https://www.arduino.cc/reference/en/>.

A. Seznam přiložených souborů

1. smart_GREENHOUSE.ino