

STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

Obor č. 18: Informatika

Chytrá domácnost

Tadeáš Fryčák

Litovel 2022

STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

Obor č. 18: Informatika

CHYTRÁ DOMÁCNOST

SMART HOME

Autor: Tadeáš Fryčák
Škola: Gymnázium Jana Opletala, Litovel,
Opletalova 189
Kraj: Olomoucký
Konzultant: Filip Szkandera

Litovel 2022

Prohlášení

Prohlašuji, že jsem svou práci SOČ vypracoval samostatně a použil 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 Litovli dne: _____

Tadeáš Fryčák

Poděkování

Děkuji svému kolegovi Filipu Szkanderovi za obětavou pomoc při navrhování designu, za podnětné připomínky a za nekonečnou trpělivost při konzultacích nápadů pro implementaci.

Anotace

Tato práce popisuje zhotovení softwaru Chytrá domácnost pro domácí použití. Výsledkem práce je program umožňující uživateli plnohodnotně ovládat svůj vlastní připojený HW a jiný HW třetích stran s otevřeným API a zobrazovat přehledně informace o vybraných zařízeních. Pro tyto účely program využívá grafické webové prostředí. Cílem bylo co nejvíce zjednodušit ovládání a přístup k informacím o zařízeních uživateli. Chytrá domácnost také nabízí několik funkcionalit z obrazové analýzy pro zařízení zachycující obrázky či videa – zejména pro kamerové systémy či domácí videotelefony.

Klíčová slova

Chytrá domácnost; IoT zařízení; AI; Umělá inteligence; Obrazová analýza

Annotation

This work describes development of Smart Home software for home use. The result of the work is a program that allows the user to fully control their own connected hardware and other hardware with an open API and display information about selected devices. For these purposes, the program uses a graphical web environment. The goal was to simplify the user's access to device information as much as possible. The Smart Home also offers computer image analysis functionalities for devices that capture images or videos – especially for CCTV or smart doorbells.

Keywords

Smart Home; IoT devices; Artificial Intelligence; Image analysis

Obsah

Úvod	10
1 Definice pojmů	11
1.1 Server, klient	11
1.1.1 Server	11
1.1.2 Klient	11
1.2 Frontend a backend	12
1.2.1 Frontend	12
1.2.2 Backend	12
1.2.3 Whitelist	13
1.2.4 Blacklist	13
1.3 Programovací jazyky	13
1.3.1 HTML	13
1.3.2 CSS	13
1.3.3 JavaScript	13
1.3.4 Python	14

1.3.5	C++	14
2	Frontend	15
2.1	Použité technologie	15
2.2	Grafické prvky	17
2.2.1	Pozadí	17
2.2.2	Ikony	18
2.3	Normální mód	19
2.3.1	Plocha	20
2.3.2	Tile	21
2.3.3	Modal	26
2.3.4	Item	27
2.3.5	Systémová nabídka	38
2.3.6	Nastavení	39
2.3.7	Uživatelský seznam	42
2.3.8	Klientský seznam	43
2.4	Upravovací mód	44
2.4.1	Plocha	45
2.4.2	Tile	46
2.4.3	Modal (upravovací)	46
2.4.4	Systémová nabídka	49
2.5	Synchronizace	51

2.6	Načítání	52
2.7	Vícejazyčnost	53
2.8	Aplikace	54
3	Backend	56
3.1	Použité technologie	56
3.2	Autentizace	57
3.2.1	Blacklist a whitelist	57
3.3	Registrace	58
3.4	Přihlášení	59
3.5	Role a práva	60
3.6	Protokoly	61
3.6.1	MQTT	62
3.6.2	Magic Packet	63
3.6.3	RTSP	64
3.6.4	Alarm	64
3.6.5	Průša	65
3.6.6	Ostatní	65
3.7	Terminál	66
3.8	Struktura kódu	67
3.9	Konfigurace	73
4	Server a jeho zabezpečení	74

4.1	Server	74
4.2	Zabezpečení	75
4.2.1	Uživatelské vstupy	75
4.2.2	Nginx	75
4.2.3	Firewall	75
5	Zařízení	77
5.1	Kamerový systém	77
5.2	Domácí videotelefon	78
5.3	Měření ve skleníku	79
5.4	Ovládání LED	83
	Závěr	86
	Seznam zdrojů	88
	Seznam zkratk	91
	Seznam obrázků	94
	Seznam příloh	95

Úvod

Tato práce popisuje vývoj projektu s názvem *Chytrá domácnost* pomocí programovacích technologií. Cílem bylo vytvořit intuitivní systém běžící na jakémkoli počítači, který centralizuje ovládání hardwarových zařízení komunikujících na různých protokolech s možností automatizace. V průběhu vývoje jsem obecně kladl důraz na to, aby Chytrá domácnost byla především:

- jednoduše ovladatelná a přehledná
- uživatelsky přívětivá
- univerzální, modulární a rozšiřitelná
- softwarově bezpečná a internetově nezávislá

Kapitola 1

Definice pojmů

1.1 Server, klient

Chytrá domácnost se dělí na tyto hlavní části: server, klienti a hardwarová IoT zařízení.

1.1.1 Server

Má dva významy, může se jednat o:

- označení HW zařízení připojeného do sítě počítačů, které je optimalizováno pro nepřetržitý stabilní provoz
- označení programu běžícím na hardwarovém serveru, který poskytuje klientům své služby [1]

1.1.2 Klient

Je označení zařízení připojeného do sítě počítačů, který žádá o poskytnutí služeb serverový počítač. V komunikaci se serverem Chytré domácnosti to

může být například jiný počítač, mobilní telefon, či tablet.

1.2 Frontend a backend

1.2.1 Frontend

Jedná se o uskupení dat webové aplikace často strukturalizované do souborů obsahující veškeré prvky a informace, se kterými uživatel (klient) může interagovat a zobrazit si je. Tato část dat je obdržena od webového serveru a vizualizována v prohlížeči uživatele, který zadal příkaz si tato data zobrazit. [2] Pomineme-li například cookies uchovávající stavy jednotlivých proměnných pro zachování paměti, přenášená data jsou zejména dělena na tyto soubory:

- **HTML soubory** – struktura (neboli kostra) webové stránky určující vztah a rozvržení jednotlivých prvků v okně prohlížeče
- **CSS soubory** – vzhled uživatelského prostředí
- **JavaScriptové soubory** – skripty umožňující uživateli interagovat s jednotlivými prvky na stránce

Jedná se o opak backendu.

1.2.2 Backend

Jedná se o uskupení dat webové aplikace strukturalizované do souborů obsahující kód běžící čistě na serveru, který říká počítači, co má při jakých úkonech dělat. Slouží k administraci a ke zpracování dat. Jedná se o opak frontendu. [3]

1.2.3 Whitelist

Neboli seznam povolených či bílá listina je seznam zařízení, které jsou akceptovány při žádání o přístup do Chytré domácnosti. [4] Seznam může být tvořen například pomocí IP nebo MAC adres zařízení.

1.2.4 Blacklist

Neboli černá listina je seznam zařízení, které nejsou akceptovány (tedy jsou blokovány) při žádání o přístup do Chytré domácnosti. [5] Seznam může být tvořen například pomocí IP nebo MAC adres zařízení.

1.3 Programovací jazyky

1.3.1 HTML

HTML je programovací jazyk, který se používá k naprogramování základní obsahové struktury webových stránek. [6]

1.3.2 CSS

Jedná se o kolekci metod umožňující navrhnout grafickou část webové stránky a tím zvýraznit či jinak graficky upravit HTML prvky. [7]

1.3.3 JavaScript

JavaScript je programovací jazyk, který se používá zejména ve vývoji webových stránek. [8] Často se používá pro vývoj frontendové části. JavaScript

jsem využil pro klasickou požadavkově založenou, tak moderní událostně založenou komunikaci mezi serverem a klientem.

1.3.4 Python

Python je jeden z moderních programovacích jazyků, který je:

- Univerzální – široce použitelný, je možné ho využít například pro backendový vývoj webových stránek, vývoj počítačových programů, či pro matematické výpočty a predikce
- Rychle naučitelný, skvělý pro výuku, jednoduchý
- Široce podporovaný – k dispozici je nespočetné množství knihoven
- Velmi rozšířený [9]

Jinja

Jinja je šablonový programovací jazyk, který dovoluje pohodlně, jednoduše a rychle vytvářet dynamické stránky pomocí statických HTML souborů v syntaxi podobné Pythonu. [10]

1.3.5 C++

C++ je nativní programovací jazyk, který se používá často pro vývoj aplikací s požadavkem na vysokou rychlost a optimalizovanost, ale také jako jednoduchý programovací jazyk pro mikrokontroléry. V Chytré domácnosti byl použit například pro programování čipů ESP, ATmega a ATtiny.

Kapitola 2

Frontend

Frontend Chytré domácnosti byl naprogramován v jazycích HTML, CSS (a jeho preprocesoru SCSS) a JavaScriptu.

2.1 Použité technologie

Pro zjednodušení práce s designem a oživením HTML prvků byly použity následující knihovny:

- **JQuery** – knihovna zjednodušující práci s JavaScriptem
- **Bootstrap** – CSS knihovna zjednodušující navrhování designu, zejména rozvržení prvků
- **Swiper JS** – JS knihovna vytvářející systém ploch; pro potřeby Chytré domácnosti byla trochu upravena
- **Fotorama** – CSS+JS knihovna vytvářející galerii obrázků pro jednoduchý výběr pozadí Chytré domácnosti
- **Socket IO** – JS knihovna pro asynchronní online komunikaci mezi serverem a klientem s možností odeslat zprávu jednoduše více zařízením

- **Sortable JS** – JS knihovna pro vytvoření systému pro změnění pozice tilů na ploše, či itemů v modalu
- **Chart JS** – JS knihovna pro vytváření interaktivních a rychlých grafů závislosti veličin

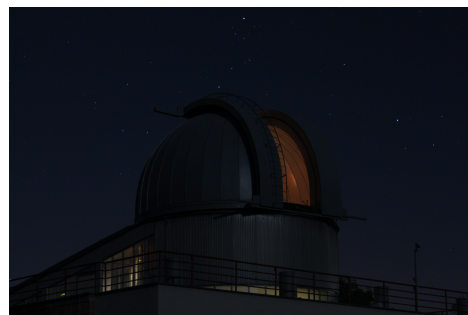
2.2 Grafické prvky

2.2.1 Pozadí

Pro prostředí Chytré domácnosti jsem použil přes 150 vlastních fotografií, které si sama Chytrá domácnost rozdělí na dvě části. První část fotografií je světlá a je použita jako pozadí ve světlém motivu (viz dále) a druhá tmavá část je použita jako pozadí v tmavém motivu. Několik pozadí jsem také vygeneroval v programu [Cinema 4D](#).



(a) Světlé pozadí



(b) Tmavé pozadí

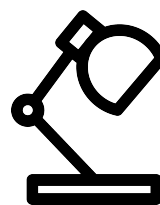
Obrázek 2.1: Příklady vyfocených pozadí

2.2.2 Ikony

Stejně jako fotografie pro pozadí bylo vytvořeno také přes 20 vlastních originálních ikon pro účely Chytré domácnosti. Tyto ikony jsou použité jak v systémové části Chytré domácnosti, tak i v její dynamické části kontrolované uživatelem.



(a) Ikona žárovky



(b) Ikona stolní lampy

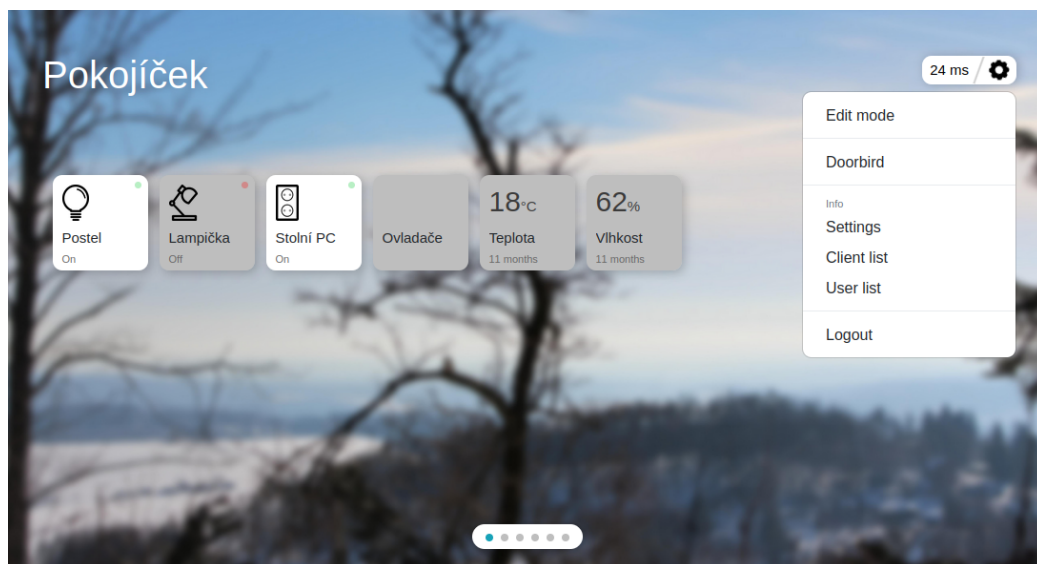
Obrázek 2.2: Ukázky nakreslených ikon

2.3 Normální mód

Po zadání domény/IP adresy serveru (například *home.cz* nebo *172.16.0.7*) do prohlížeče se uživateli otevře přihlašovací stránka (viz další kapitoly). Po úspěšném přihlášení se mu zobrazí výchozí stránka Chytré domácnosti v normálním módu, který slouží k vyčtení základních informací o zařízeních a k provedení jednoduchých akcí. V Chytré domácnosti lze zařízení přehledně strukturovat do hierarchie.

2.3.1 Plocha

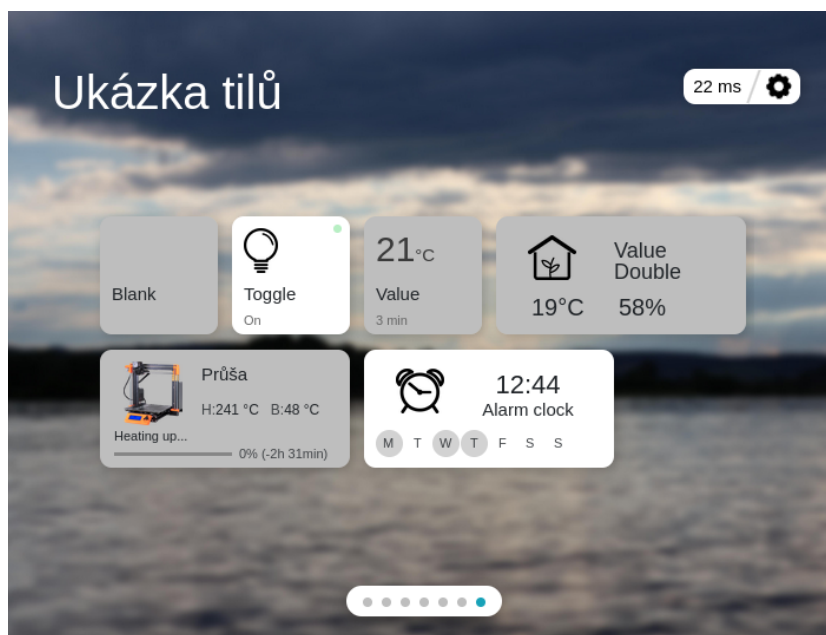
Nejvyšší složka v hierarchii. Jednotlivá zařízení je možné shlukovat do ploch (například dle umístění – pokojíček, kuchyň, garáž, ...), mezi kterými lze přepínat jednoduše jako na mobilním telefonu mezi plochami – přejetím prstu z levé strany doprava pro zobrazení předchozí plochy, či z pravé strany doleva pro zobrazení následující plochy. Mezi plochami lze také přepínat kliknutím na tečku v dolní části obrazovky (aktuální plocha je barevně zvýrazněna), nebo také pomocí posunutí kolečka myši v dolní části obrazovky.



Obrázek 2.3: Náhled na první plochu Chytré domácnosti v normálním módu

2.3.2 Tile

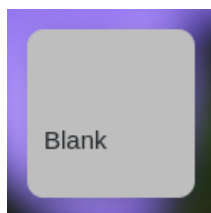
Tile (neboli dlaždice) je druhá složka v hierarchii vizualizace informací. Re-representuje jedno konkrétní zařízení. Může dle svého typu buď zobrazovat nějakou informaci (například tile *Value*), anebo po krátkém kliknutí na něj vyvolat uživatelem definovanou akci (například tile *Toggle*). V Chytré domácnosti je několik typů tilů. Po dlouhém přidržení prstu/levého tlačítka myši na tilu se otevře tzv. modal s itemy (viz další sekce), které zobrazují podrobnější/přesnější informace a umožňují rozšířené akce daného zařízení.



Obrázek 2.4: Ukázka několika typů tilů na poslední ploše

Blank

Výchozí tile o velikosti 1x1 slouží jen pro uskupení itemů. V upravovacím režimu může uživatel upravit název (text) tilu.

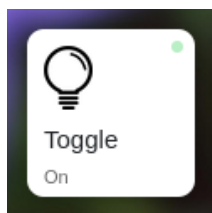


Obrázek 2.5: Ukázka tilu typu blank

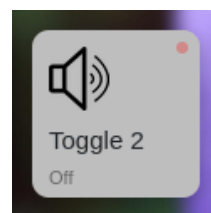
Toggle

Přepínací tile o velikosti 1x1 sloužící například pro zapínání světel, zásuvek, atp. V upravovacím režimu může uživatel upravit název (text) tilu a ikonu. Po kliknutí může tile nabývat dvou stavů:

- **aktivovaný** – pozadí je zvýrazněné a signální tečka v pravém horním rohu je světle zelená
- **deaktivovaný** – pozadí je zašedlé a signální tečka v pravém horním rohu je světle červená



(a) Aktivovaný toggle

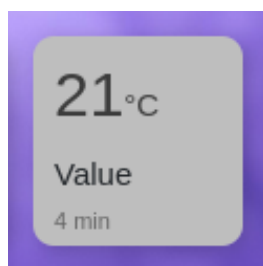


(b) Deaktivovaný toggle

Obrázek 2.6: Ukázky stavů tilu typu Toggle

Value

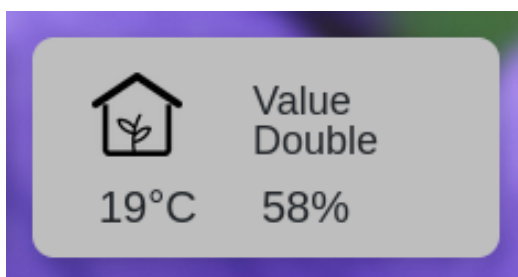
Tile o velikosti 1x1 je určen pro zobrazení jedné hodnoty zařízení s vlastní jednotkou, může zobrazovat například teplotu, vlhkost, nebo tlak. V upravitelském režimu může uživatel upravit název (text) tilu a jednotku. Tile zobrazuje svou hodnotu s předem definovanou jednotkou.



Obrázek 2.7: Ukázka tilu typu Value

Value Double

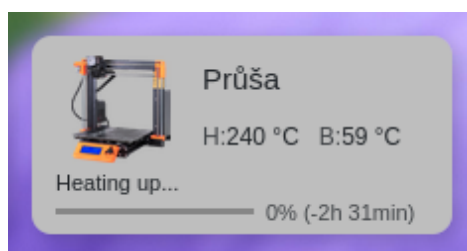
Tile o velikosti 2x1 je určen pro zobrazení více hodnot s vlastními jednotkami daného zařízení, může například zobrazit teplotu a vlhkost najednou. V upravitelském režimu může uživatel upravit název (text) tilu, ikonu a jednotky. Tile zobrazuje svou hodnotu s předem definovanou jednotkou.



Obrázek 2.8: Ukázka tilu typu Value Double

Průša

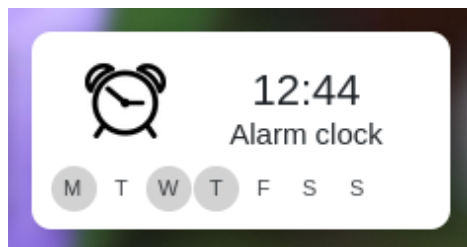
Tile o velikosti 2x1 je určen pro zobrazení stavu 3D tiskárny od výrobce [Průša](#). V upravovacím režimu může uživatel upravit název tiskárny. Tile zobrazuje teplotu heatbedu, teplotu trysky, stav tisknutí (kolik času zbývá do konce tisku, v jaké procentuální části tisku se zrovna tisk nachází, jaký GCode aktuálně tiskne, atp.)



Obrázek 2.9: Ukázka tilu typu Průša

Alarm Clock

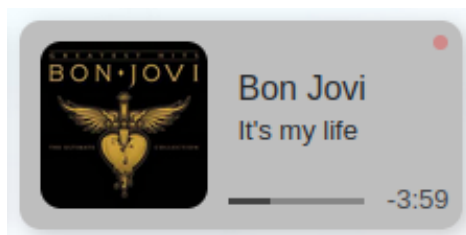
Tile o velikosti 2x1 je určen pro jednoduchou automatizaci díky možnosti nastavení vyvolání události na základě nastaveného času a dne v týdnu. V upravovacím režimu může uživatel upravit název tilu, ikonu, čas aktivace, den/dny aktivace. Tile zobrazuje čas a dny, ve kterých bude aktivován. Po krátkém kliknutí na tile se aktivuje časovač, který se spustí v daný čas a den/dny v týdnu.



Obrázek 2.10: Ukázka tilu typu Alarm Clock

Player

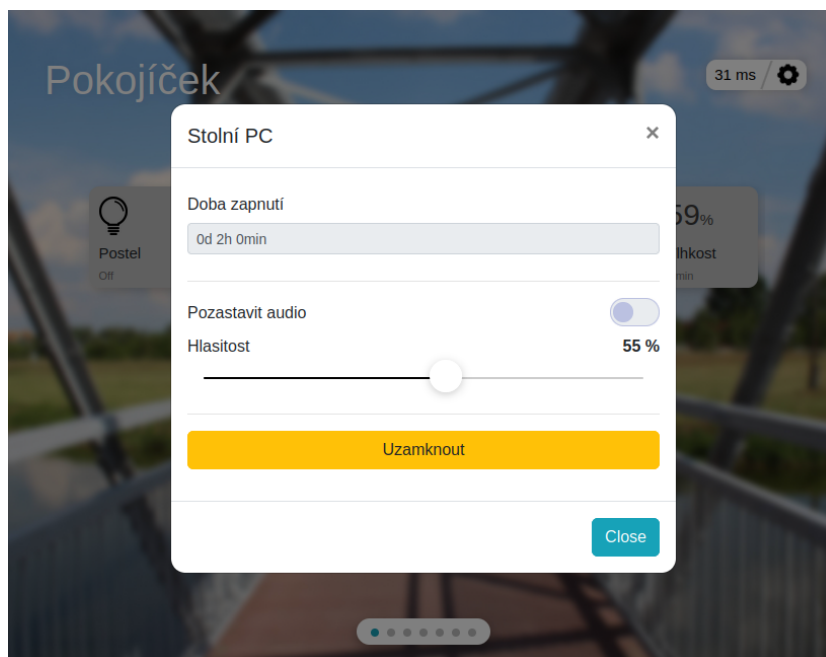
Přehrávač je tile o velikosti 2x1, který je aktuálně ve fázi vývoje. V budoucnu bude určen pro jednoduché přehrávání na koncových zařízeních i na zvukovém výstupu z Raspberry Pi.



Obrázek 2.11: Ukázka tilu typu Player

2.3.3 Modal

Modal je grafické okno, které se otevře po dlouhém stisku levého tlačítka myši na tilu, nebo v případě mobilního telefonu po dlouhém přidržení prstu na tilu. V hierarchii vizualizace se jedná o třetí složku – nachází se zde uživatelem nakonfigurované itemy, které slouží pro zobrazování přesnějších informací o zařízení, či pro podrobnější ovládání různých parametrů zařízení (viz další sekce). Modal lze zavřít stisknutím klávesy s označením *Esc*, kliknutím vně modalu, či kliknutím na tlačítko "Zavřít".



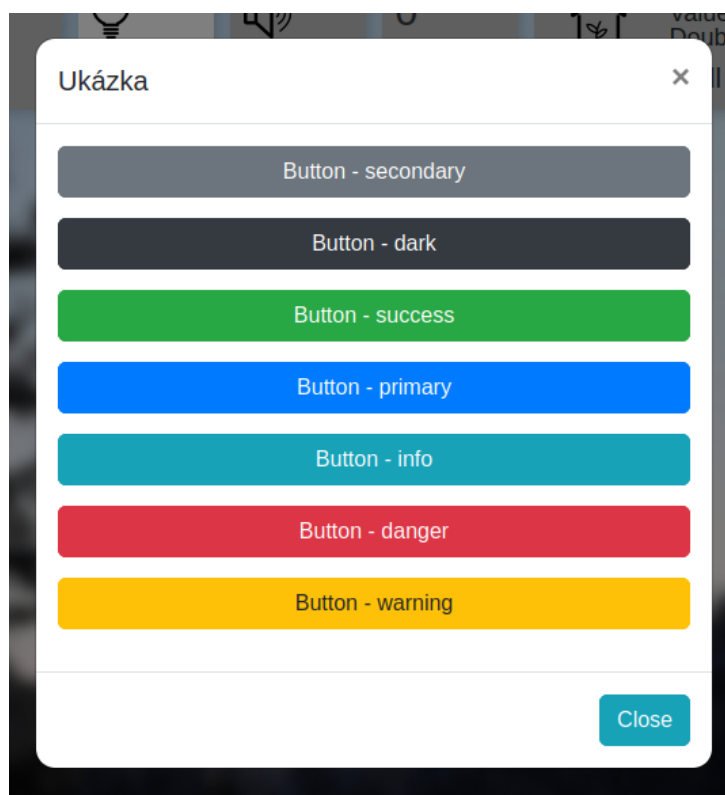
Obrázek 2.12: Ukázka modalu pro tile z první plochy

2.3.4 Item

Item je v hierarchii vizualizace poslední stupeň. Item může sloužit jako grafický prvek v modalu, ale také hlavně jako podrobnější ukazatel hodnot, či ovládací prvek pro přesnější nastavení daného zařízení. Pro Chytrou domácnost jsem vytvořil tyto typy itemů:

Tlačítko

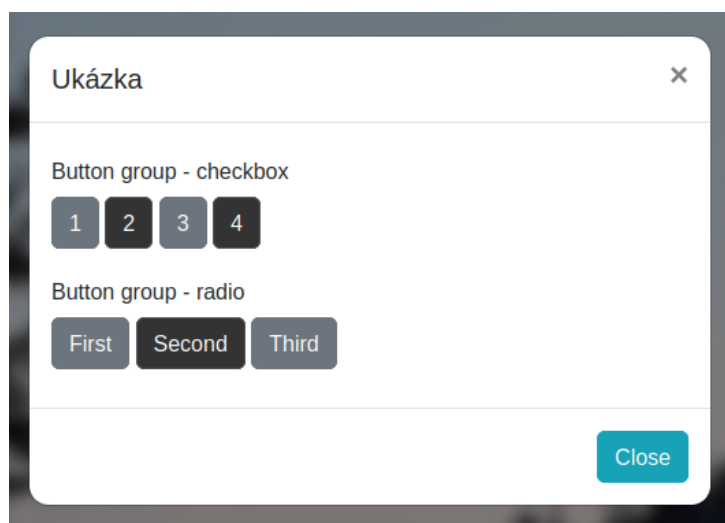
Neboli "Button" je čistě vstupní grafický element, který po kliku ihned pošle uživatelem zadanou hodnotu. V upravovacím režimu je možné nastavit název, barvu a hodnotu pro odeslání po stisku tlačítka. Tlačítko se hodí například pro spuštění úloh, od kterých není vyžadována zpětná vazba, či neumožňují zpětnou akci – například tlačítko, které po stisknutí vzdáleně uzamkne počítač.



Obrázek 2.13: Ukázka několika konfigurací itemu tlačítko

Skupina tlačítek

Neboli "Button group" je vstup, kde uživatel může zvolit jednu z více variant (tzv. radio button), nebo žádné nebo více variant (tzv. checkbox). V upravovacím režimu si uživatel dynamicky konfiguruje přidání a odebrání variant, určení odesílacích hodnot po stisku tlačítek a finálně také i název (text) před tlačítky.



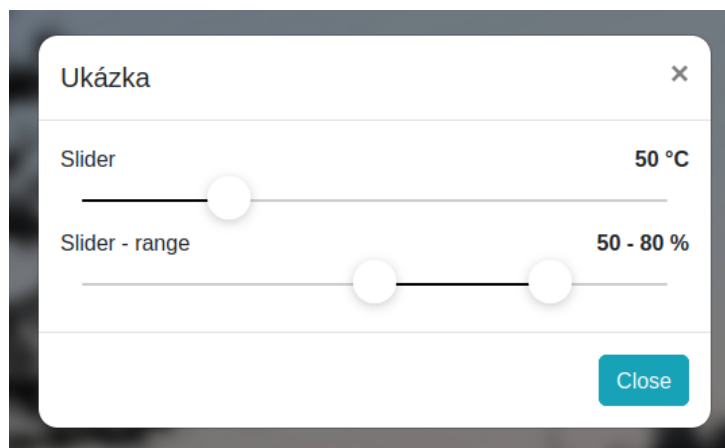
Obrázek 2.14: Ukázka konfigurací itemu skupina tlačítek

Posuvník

Neboli "Slider" je vstupně výstupní graficky animovaný posuvník, který mění svou hodnotu po uchopení prstem/kurzorem a přesunutí na novou hodnotu. V upravovacím režimu může uživatel nastavit název (text) posuvníku, informace o dané hodnotě – minimální a maximální hodnota, jednotka hodnoty a velikost kroku. Dále je možné nastavit:

- Parametrem "Plynulý" ("Smooth") plynulé odesílání hodnoty (hodnota je odeslána ihned po posunutí posuvníku do nové polohy, nikoli až po ukončení tahu prstem/levým tlačítkem myši). Signalizace, zda byla hodnota již propsána do systému a odeslána, je značena barvou textu – pokud je šedá, hodnota se ještě neodeslala, pokud je černá, hodnota se již odeslala.
- Parametrem "Rozsah" ("Range") rozsah hodnot – místo nastavení jedné hodnoty na jednom posuvníku bude možné nastavit rozsah hodnot dvěma kolečky.
- Parametrem "Vypnutý" ("Disabled") vypnutí daného posuvníku. Posuvník bude sloužit tedy jen jako ukazatel nastavené hodnoty (podobně jako item "ukazatel průběhu").

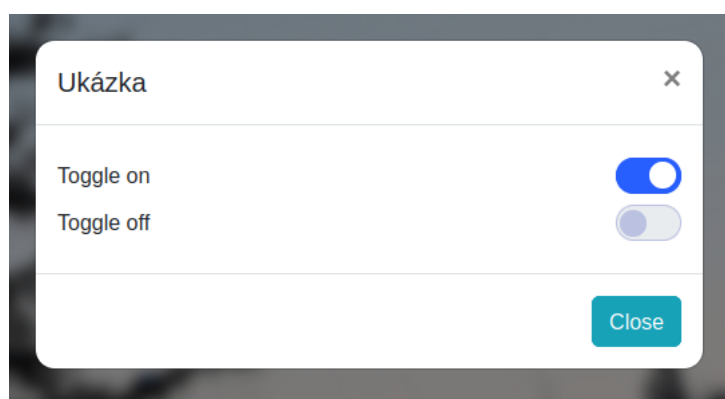
Posuvník se dá využít například pro nastavování úrovně osvětlení, nebo teploty termostatu.



Obrázek 2.15: Ukázka dvou konfigurací itemu posuvník

Přepínač

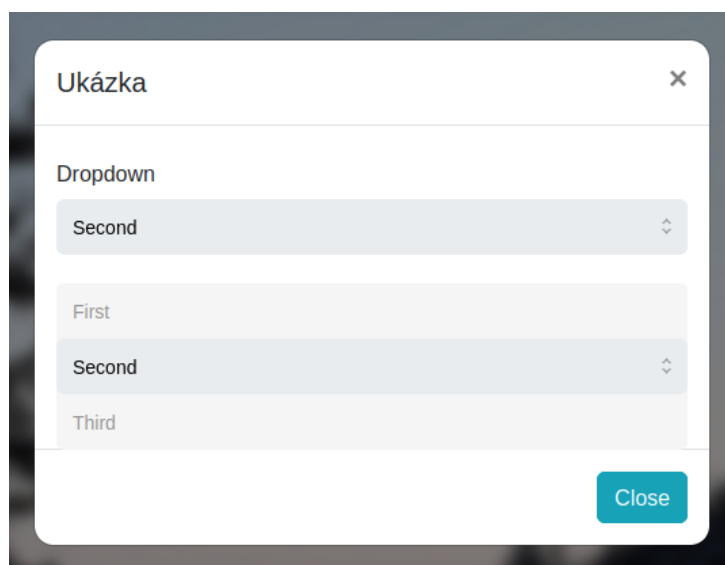
Neboli "Toggle", či switch je vstupně výstupní přepínač, který umožňuje uživateli přepínat mezi dvěma nakonfigurovanými stavy. V upravovacím režimu se nastavuje název (text) u přepínače a vypnutou a zapnutou hodnotu. Signalizace zapnutí a vypnutí je indikována pomocí zvýrazněné barvy a posunutí kolečka – obdobně jako na operačních systémech mobilních telefonů.



Obrázek 2.16: Ukázka dvou konfigurací itemu přepínač

Rozbalovací nabídka

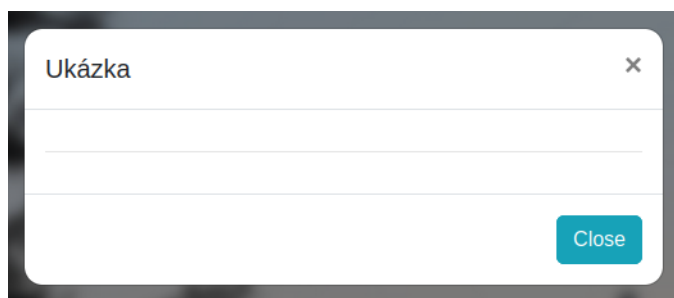
Neboli "Dropdown" je vstupní i výstupní nabídka, která se po kliku kurzorem myši animovaně rozbalí a umožní uživateli vybrat si jednu z možností. Oproti skupině tlačítek je dobrá pro využití nabídek s více variantami. V upravovacím režimu má uživatel možnost nastavit název rozbalovací nabídky a poté dynamicky přidávat nebo odebírat varianty.



Obrázek 2.17: Ukázka konfigurací itemu rozbalovací nabídka

Oddělovač

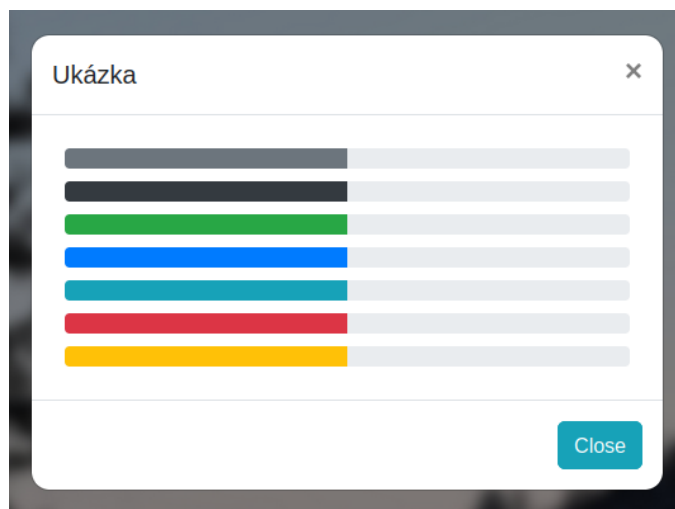
Neboli "Separator" je čistě grafický oddělovač pro přehlednost itemů v modalu.



Obrázek 2.18: Ukázka itemu oddělovač

Ukazatel průběhu

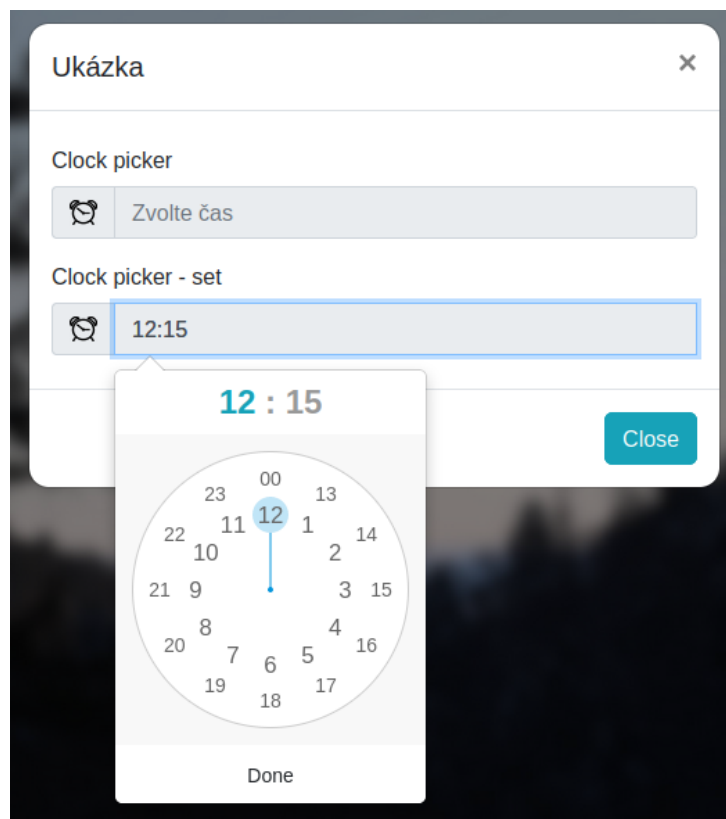
Neboli "Progress bar" je výstup, který graficky znázorňuje nabývání nebo pokles konkrétní hodnoty například v procentech. Uživatel má možnost v upravovacím režimu nastavit název (text) před ukazatelem průběhu, dále jeho minimální a maximální hodnotu a barvu. V praxi se dá použít například ke kontrole výšky hladiny v nádrži, nebo při kontrole vlhkosti půdy v květináči.



Obrázek 2.19: Ukázka konfigurací itemu ukazatel průběhu

Časový vstup

Neboli "Clock picker" je vstupní pole, pod kterým se po kliku na něj zobrazí malé okno s grafickou nabídkou pro výběr času. Výběr času probíhá stejným způsobem jako nastavování času v operačních systémech mobilních telefonů. Po stisku tlačítka "Hotovo" (neboli "Done") se zadaný čas uloží. V upravitelském režimu si uživatel může upravit název (text) vstupu a nahrazující text ve vstupu, který se objeví pouze pokud je vstup prázdný. Časový vstup je možné uplatnit například při jednoduché automatizaci.



Obrázek 2.20: Ukázka konfigurací itemu časový vstup

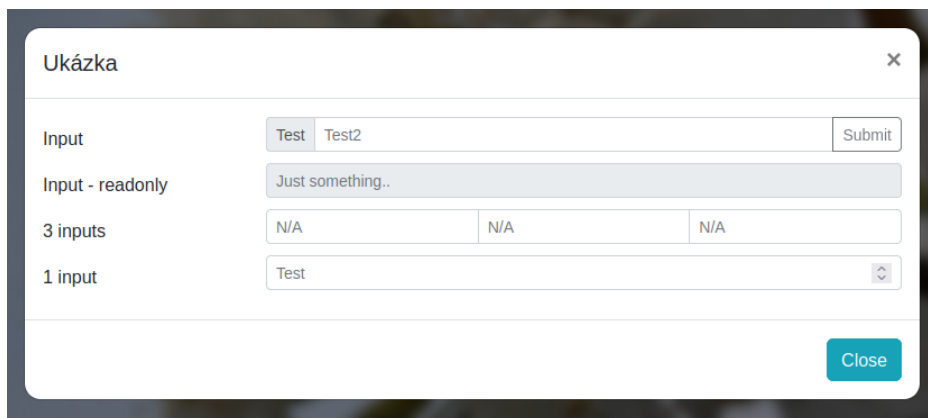
Vstupní pole

Neboli "Input" je vstupně výstupní pole pro zadání/zobrazení hodnoty v podobě textu nebo čísla. V upravovacím režimu lze nastavit název (text) vstupního pole a text, který je zobrazen v případě, že není zadána žádná hodnota (tzv. "Placeholder"). Dále je možné nastavit tyto parametry:

- Počet vstupů ("Input count") – používá se například pro zadání IP/MAC adresy
- Pouze pro čtení ("Readonly") – pokud je tento parametr aktivovaný, hodnotu vstupu nemůže měnit uživatel, ale jen připojené HW zařízení (zařízení pak může do tohoto pole odesílat například textové zprávy)

ohledně stavu zařízení)

- Číselný vstup ("Number") – toto nastavení umožňuje nastavit vstup hodnoty pouze pro číselný formát, vedle vstupu se zobrazí navíc šipky umožňující inkrementovat číslo ze vstupu
- Seznam hodnot ("Value list") – pokud je tento parametr aktivovaný, uživatel má schopnost přidávat a odebírat dynamicky vstupní pole
- Potvrzovací tlačítko ("Confirm button") – pokud je tento parametr aktivovaný, vstup nebude posílat svou hodnotu ihned po jakékoli změně, ale až po potvrzení klávesou "Enter", nebo po stisku nově vytvořeného tlačítka "Potvrdit"("Submit"). Toto nastavení uživatel použije zejména, když je potřeba změnit hodnotu na nastavovanou až ve finální podobě - například při nastavování teploty tiskárny z 249 °C na 240 °C by se bez tohoto nastavení odeslala nejdříve hodnota 24 °C a až poté 240 °C.



Obrázek 2.21: Ukázka konfigurací itemu vstupní pole

Graf

Neboli "Graph" je čistě výstupní grafický prvek, který se používá pro zobrazení a uchování časové řady. V upravitelném režimu má uživatel možnost

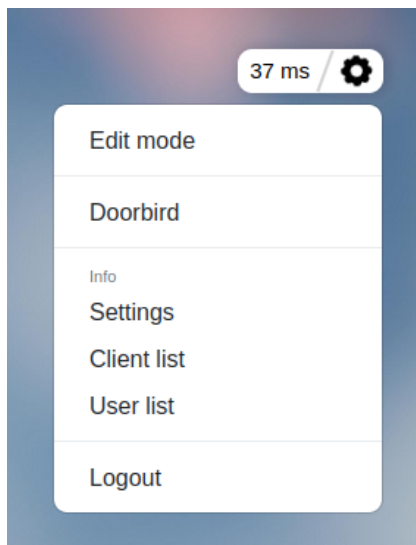
nastavit název grafu.

Obrázek

Neboli "Image" je čistě výstupní grafický prvek zobrazující jakýkoli obrázek či sekvenci obrázků (tedy video). V upravovacím režimu je možné nastavit jeho název a URL adresu, ze které si má stáhnout obrazová data.

2.3.5 Systémová nabídka

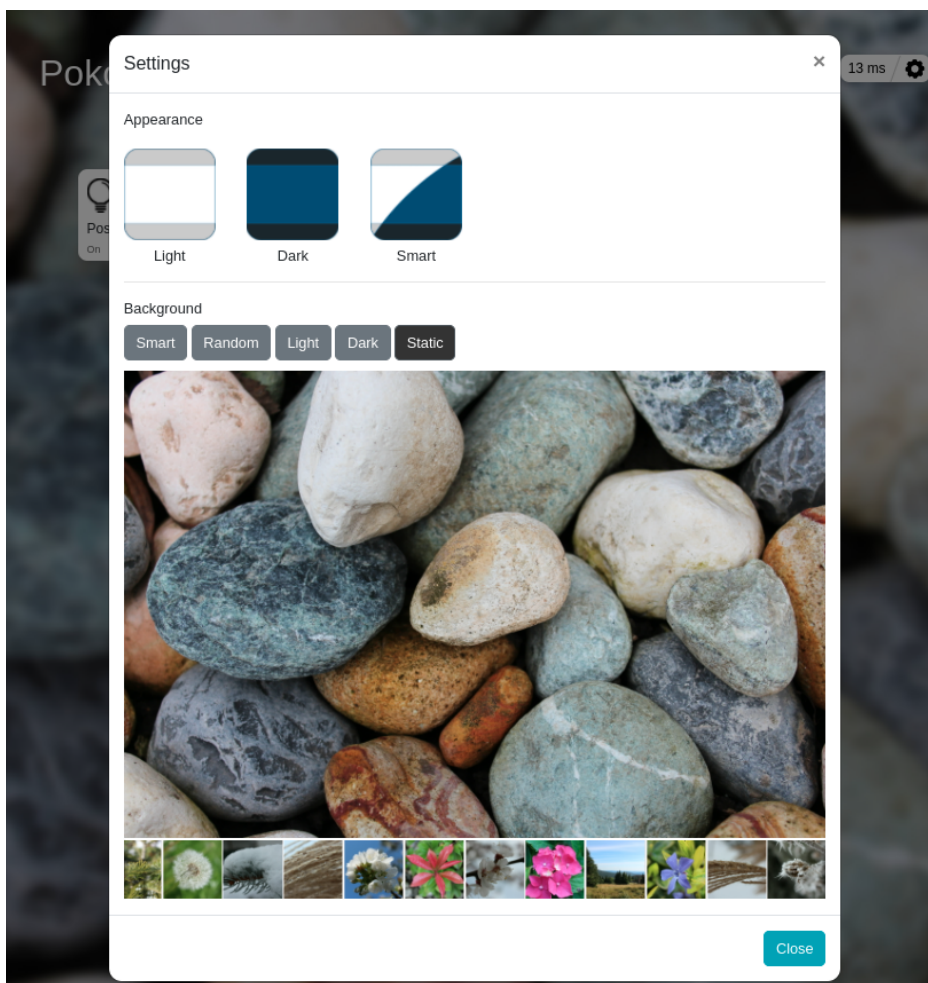
Systémová nabídka se aktivuje po kliknutí na ikonu nastavení v pravém horním rohu. Přes nabídku je možné otevřít nastavení, seznam klientů, seznam uživatelů, aktivovat upravovací mód, nebo se odhlásit.



Obrázek 2.22: Ukázka systémové nabídky v normálním módu

2.3.6 Nastavení

Neboli "settings" je nastavení, které umožňuje uživateli přizpůsobit si svůj uživatelský účet.

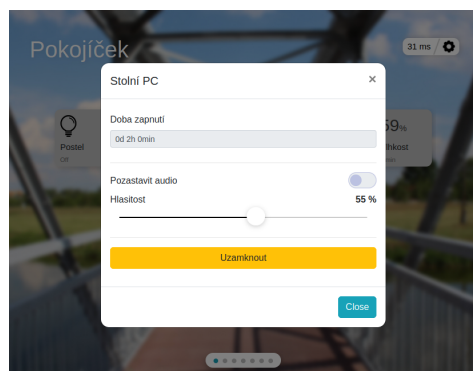


Obrázek 2.23: Modal nastavení

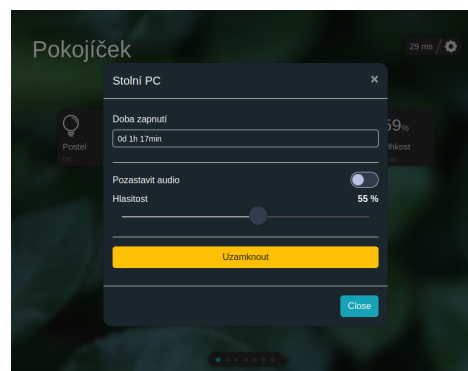
Vzhled

V horní sekci "Vzhled", neboli "Appearance", lze zvolit barevný motiv Chytré domácnosti:

- Světlý (Light) – veškeré rozhraní bude ve světlém bílo-šedém motivu
- Tmavý (Dark) – veškeré rozhraní bude v tmavém modrošedém motivu
- Chytrý (Smart) – automaticky zvolí světlý nebo tmavý mód na základě lokálního času a času východu a západu slunce. Časová signalizace pracuje v offline režimu.

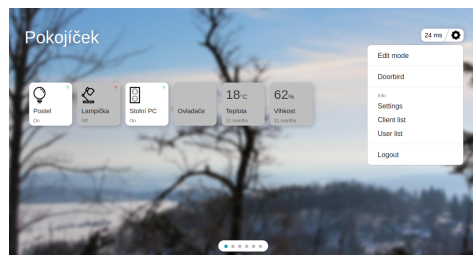


(a) Světlý motiv

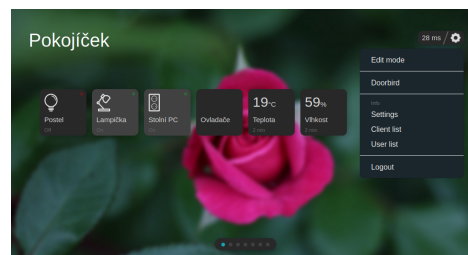


(b) Tmavý motiv

Obrázek 2.24: Ukázky motivů na normálních modalech



(a) Světlý motiv



(b) Tmavý motiv

Obrázek 2.25: Ukázky motivů na upravovacích modalech

Pozadí

V horní sekci "Pozadí", neboli "Background", lze zvolit chování automatického zobrazování pozadí:

- **Světlý (Light)** – pozadí je voleno náhodně ze všech pozadí, které Chytrá domácnost algoritmem klasifikovala jako světlá
- **Tmavý (Dark)** – pozadí je voleno náhodně ze všech pozadí, které Chytrá domácnost algoritmem klasifikovala jako tmavá
- **Chytrý (Smart)** – pozadí je voleno náhodně ze všech pozadí automaticky na základě zvoleného motivu (tedy například pokud je zvolený chytrý motiv a chytrá volba pozadí, ve dne se budou náhodně volit pouze světlá pozadí a v noci pouze tmavá pozadí – nedojde tak k oslnění uživatele v noci)
- **Náhodný (Random)** – pozadí je voleno náhodně ze všech pozadí (tmavých i světlých) nezávisle na zvoleném vzhledovém motivu
- **Statický (Static)** – pozadí je zvoleno fixně uživatelem z prohlížeče obrázků

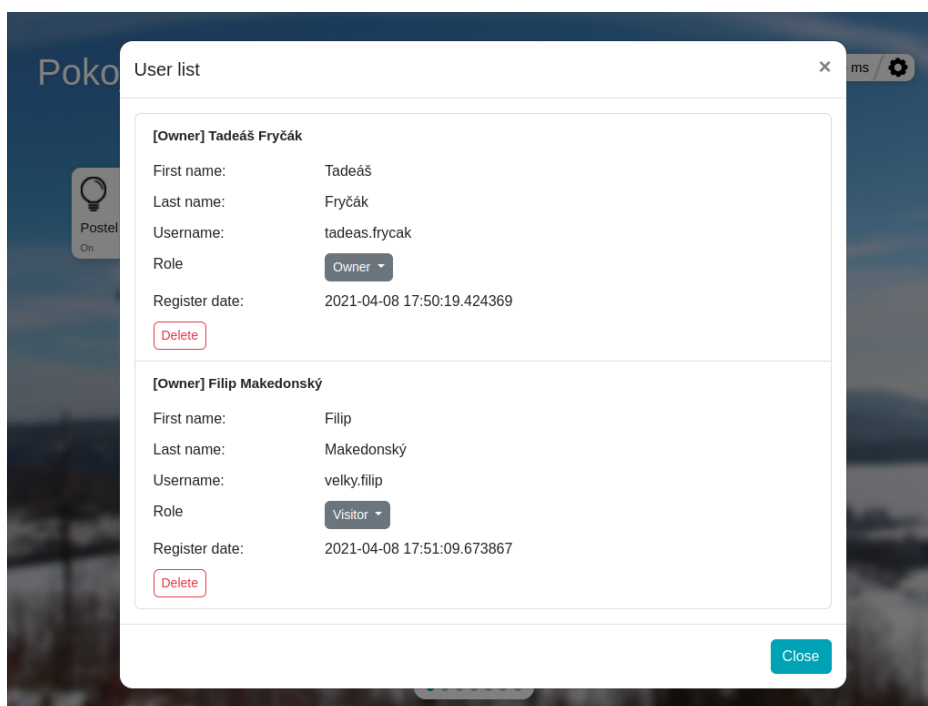
Chytrá domácnost si automaticky indexuje veškerá pozadí a algoritmem detekuje následující informace:

- **Světlost/tmavost pozadí** – na základě této informace se Chytrá domácnost rozhoduje a klasifikuje pozadí jako tmavé, či světlé
- **Nejčastější barvy** – u každého pozadí má Chytrá domácnost uloženou paletu pěti nejčastěji používaných barev v pozadí (tato funkce aktuálně není plně využívána, jedná se o připravení pro budoucí barevné přizpůsobení motivu na základě barvy pozadí)

2.3.7 Uživatelský seznam

Neboli user list, je seznam všech zaregistrovaných uživatelů s informacemi o nich. U každého zaregistrovaného uživatele je zobrazeno:

- Jméno a příjmení
- Uživatelské jméno
- Datum a čas zaregistrování
- Role (oprávnění)

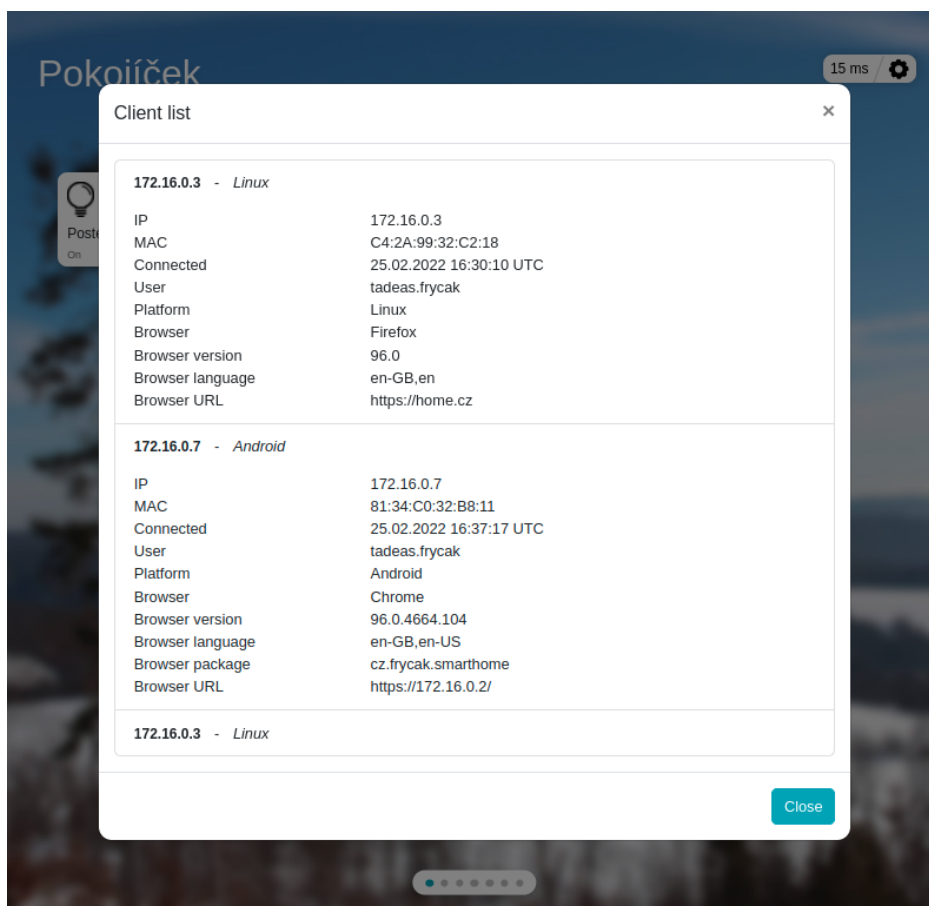


Obrázek 2.26: Seznam uživatelů

2.3.8 Klientský seznam

Neboli "client list" je seznam všech aktivních relací uživatelů. Administrátorům jsou zobrazeny tyto informace:

- IP a MAC adresa zařízení
- Datum a čas připojení
- Operační systém a prohlížeč uživatele – název, verze, jazyk
- URL adresa, kterou uživatel zadal do prohlížeče

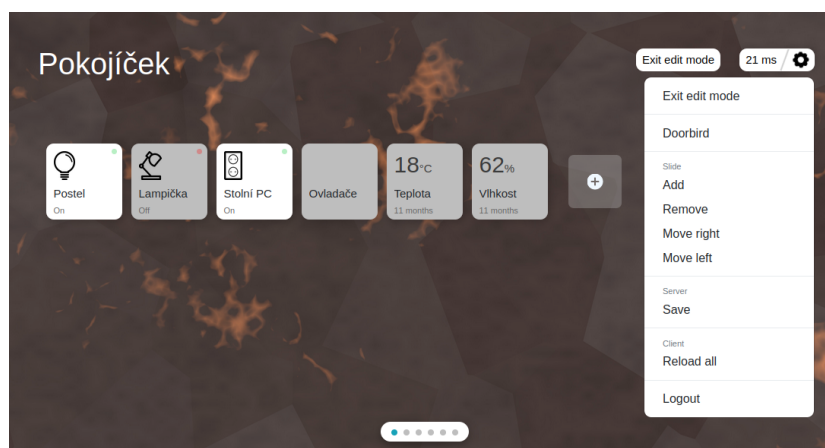


Obrázek 2.27: Seznam klientů

2.4 Upravovací mód

Upravovací mód Chytré domácnosti lze spustit po kliknutí na tlačítko "Upravovací mód" v systémové nabídce. Po jeho aktivaci se:

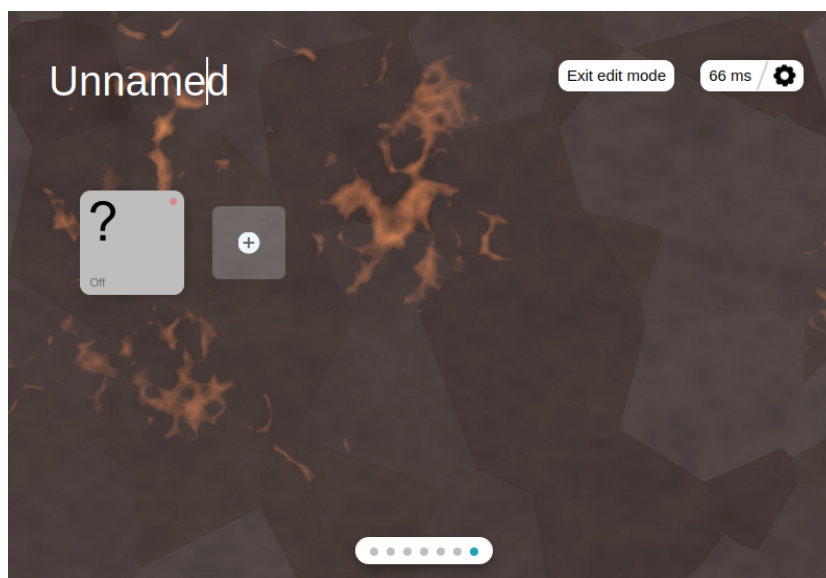
- Změní pozadí na upravovací pozadí (barevná signalizace aktivovaného upravovacího módu)
- Zobrazí tlačítko pro ukončení upravovacího módu (také dostupné v systémové nabídce)
- Umožní upravovat, přidávat, odstraňovat jednotlivé plochy (viz sekce s plochami)
- Umožní upravovat, přidávat, odstraňovat jednotlivé tily (viz sekce s tily)
- Umožní upravovat, přidávat, odstraňovat jednotlivé itemy (viz sekce s itemy)



Obrázek 2.28: Ukázka Chytré domácnosti na první ploše v upravovacím režimu

2.4.1 Plocha

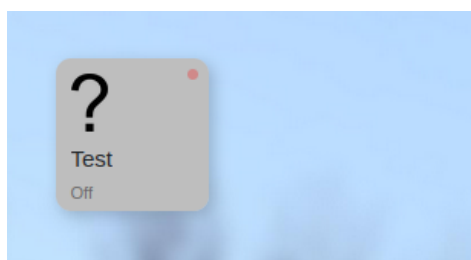
Upravení rozvržení ploch je velmi jednoduché, systémová nabídka v upravnovacím režimu nově nabízí možnosti úpravy (viz sekce se systémovou nabídkou). Název plochy se změní v textové pole, které lze po kliku na něj intuitivně upravit.



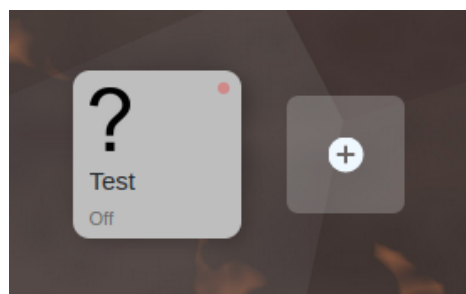
Obrázek 2.29: Ukázka možnosti úpravy názvu plochy

2.4.2 Tile

Po změně režimu z normálního do upravovacího se umožní vytvořit nový tile pomocí kliknutí na nově zobrazený poloprůhledný tile s ikonou ve znaku ”+”. Také změna pořadí tilů na ploše je velmi jednoduchá. Použil jsem systém drag and drop – stačí tedy pouze uchopit tile dlouhým kliknutím levého tlačítka myši či prstu a přetáhnout jej na nové požadovanou pozici. Další úpravy, jako například změnění typu, názvu, či ikony tilu, lze změnit v upravovacím modalu, který se otevře po krátkém kliknutí na daný tile (viz další sekce).



(a) Normální mód



(b) Upravovací mód

Obrázek 2.30: Zobrazení nového tlačítka pro přidání nového tilu

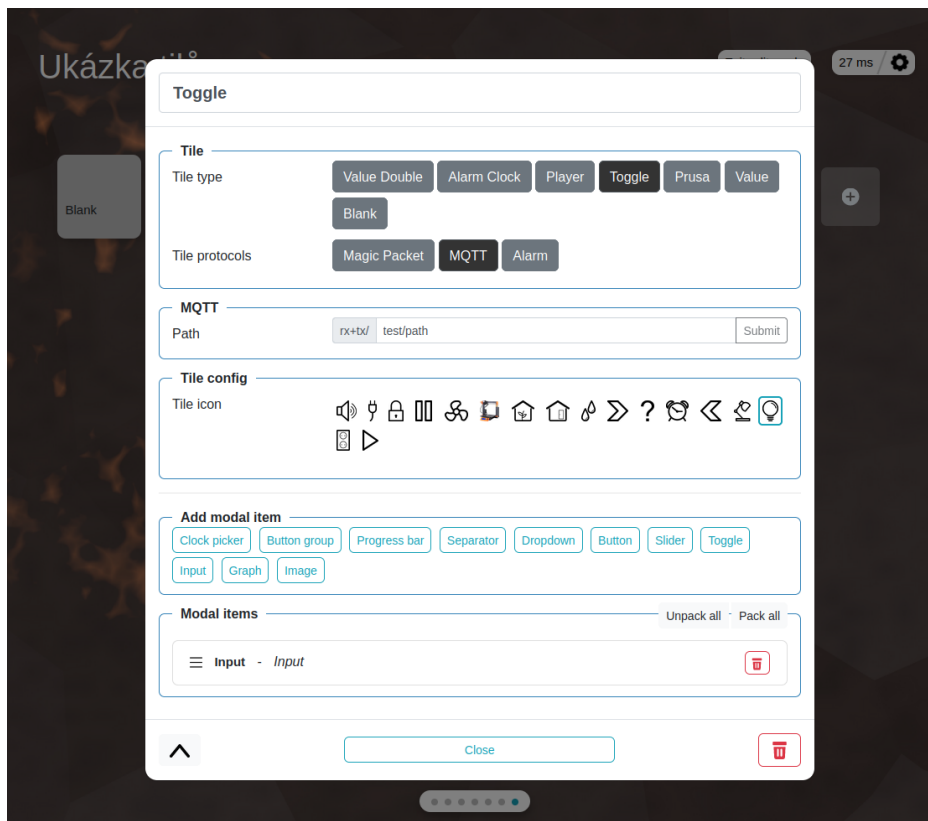
2.4.3 Modal (upravovací)

Upravovací modal (neboli ”Edit modal”) se zobrazí po krátkém stisku jakéhokoli tilu v upravovacím režimu a je možné v něm upravovat parametry tilu a spravovat itemy.

Správa tilů

V horní části upravovacího modalu se nachází prvky umožňující konfigurovat zvolený tile, je možné:

- Změnit název tilu úpravou textu ve velkém textovém poli umístěném



Obrázek 2.31: Ukázka modalu pro tile z poslední plochy

v horní části modalu

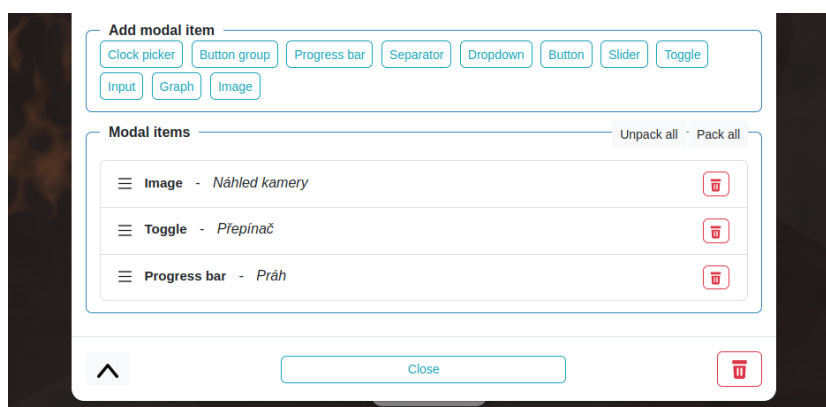
- Změnit typ kliknutím na zvolený typ – aktuální zvolený typ je zvýrazněný barvou
- Namapovat protokol/y na tile po kliknutí na vybrané protokoly v sekci "tile" – "tile protocols" (viz dále). Aktuální zvolené a aktivní protokoly jsou zvýrazněné barvou.
- Přidat item/y do normálního modalu v sekci "Přidat modal item" ("Add modal items")
- Konfigurovat, upravovat, mazat item/y z normálního modalu v sekci "Modal itemy" ("Modal items")

V pravé části patičky modalu se nachází tlačítko s ikonou koše, které po kliku na něj daný tile smaže.

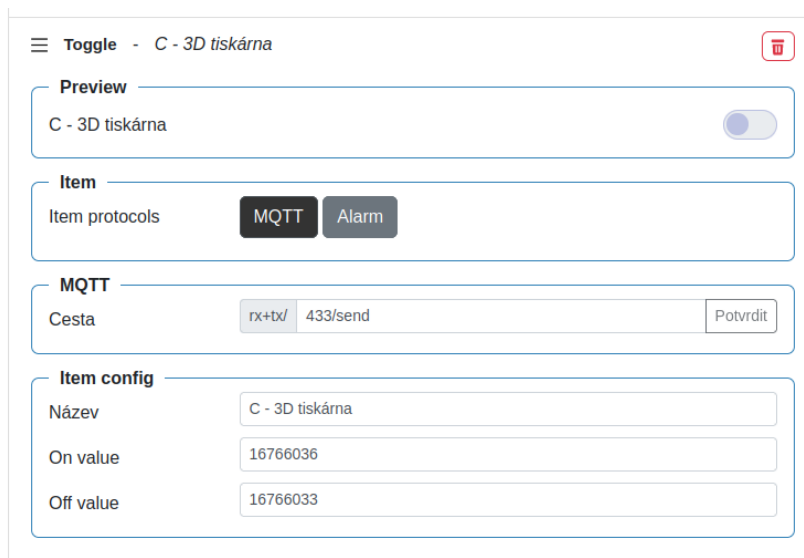
Správa itemů

Umožňuje přidat itemy do normálního modalu po kliknutí v prostřední části upravovacího modalu v sekci "Přidat item do modalu"(neboli "Add modal item") na zvolený typ itemu, který se následně přidá nad všechny ostatní aktuální itemy.

Ve spodní části modalu se nachází seznam aktuálních itemů v rozbalovací podobě, které jsou vidět v normálním modalu. Vždy je zobrazen nápis ve formátu *typ itemu - název (text) itemu*



Obrázek 2.32: Ukázka správy více itemů



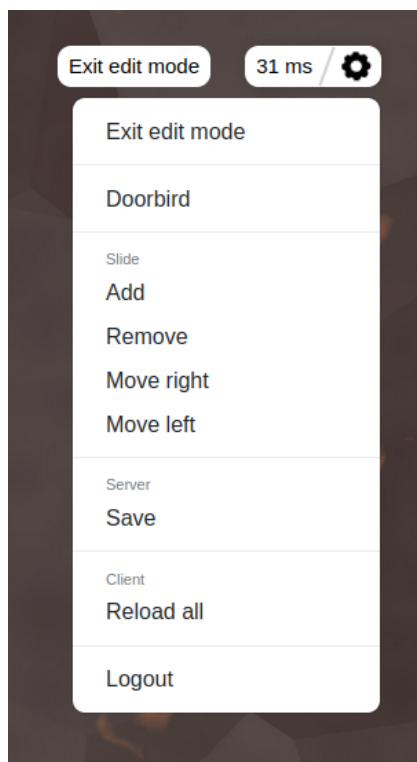
Obrázek 2.33: Ukázka správy jednoho itemu

2.4.4 Systémová nabídka

Systémová nabídka v upravovacím režimu změní svůj obsah s položkami – položky používané v normálním režimu se skryjí a položky významné pro úpravu se objeví. Tato užitečná funkce zvyšuje přehlednost mezi položkami v nabídce.

Seznam položek pro úpravu ploch:

- Tlačítko "Přidat" v sekci "Plocha" přidá novou prázdnou plochu
- Tlačítko "Odstranit" v sekci "Plocha" odebere stávající aktivní plochu se všemi tily
- Tlačítko "Posunout doprava" v sekci "Plocha" posune aktuální plochu o jednu plochu doprava
- Tlačítko "Posunout doprava" v sekci "Plocha" posune aktuální plochu o jednu plochu doprava



Obrázek 2.34: Ukázka systémové nabídky v upravovacím režimu

Mimo jiné okno systémové nabídky i tyto možnosti pro administrátory:

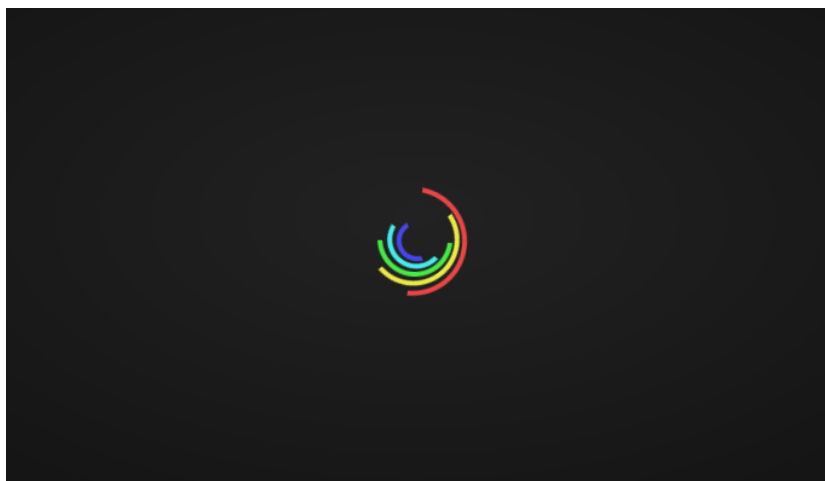
- Tlačítko "Uložit" v sekci "Server" uloží aktuální konfiguraci na serveru
- Tlačítko "Obnovit všechny" v sekci "Klient" obnoví všechny aktivní relace prohlížečů

2.5 Synchronizace

Při programování jsem kladl značný důraz na optimální synchronizaci. Vybral jsem technologii založenou na událostech (AJAX), abych byl schopný aktualizovat změny v Chytré domácnosti bez obnovování stránky prohlížeče. Synchronizaci jsem implementoval do všech částí Chytré domácnosti, včetně dynamických modulů – itemů a tilů. Jakákoli změněná informace, ať už v normálním režimu například změnou hodnoty klikem na tile, nebo v upravitelském režimu například změnou názvu plochy, se projeví na všech zařízeních ihned. U offline zařízeních je tato informace dodatečně stažena.

2.6 Načítání

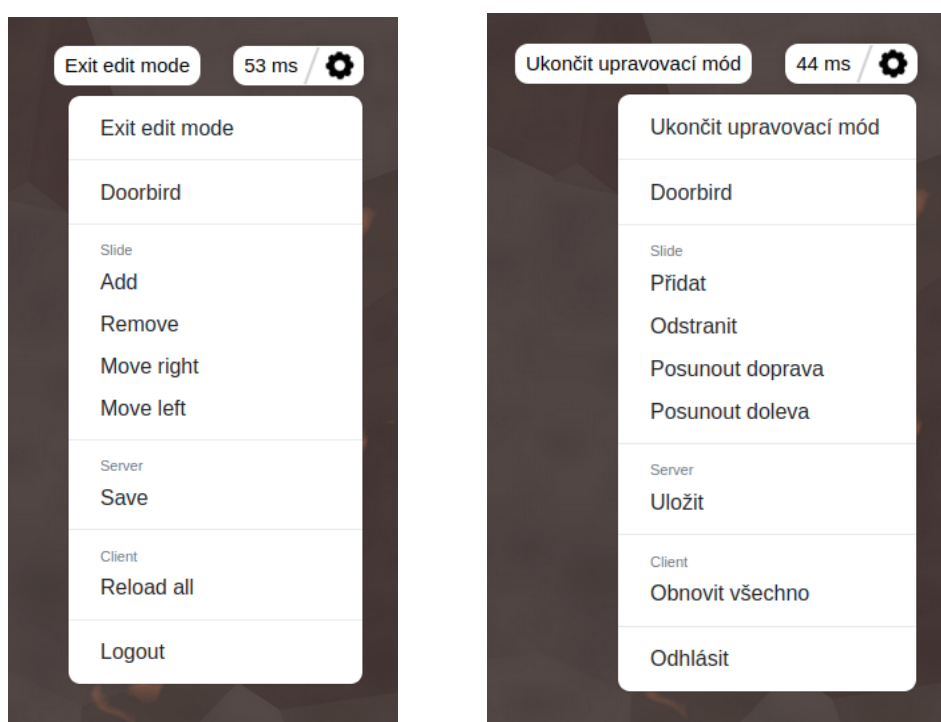
Při normálním načítání webové stránky se načítají všechny komponenty, soubory a grafické prvky časově různě dlouho a ve výsledku to může způsobit nevhledné skákání obsahu po kouskách dolů, problikávání obrázku, atp. Z těchto důvodů jsem se rozhodl implementovat tzv. "před-načítač", který se animovaně signalizuje načítání stránky, dokud nejsou veškeré soubory plně načtené. Poté se automaticky plynule odstraní.



Obrázek 2.35: Ukázka načítání před zobrazením domovské stránky

2.7 Vícejazyčnost

Chytrá domácnost automaticky upravuje texty své systémové části dle výchozího jazyka prohlížeče uživatele. Pro účely demonstrace byl v celé práci použit pro ukázky a náhledy prohlížeč s nastaveným anglickým jazykem. Aktuálně je k dispozici plně jazyk anglický a český, částečně poté i jazyk ruský.



(a) Prohlížeč v anglickém jazyce

(b) Prohlížeč v českém jazyce

Obrázek 2.36: Příklady automaticky přeložených textů dle jazyka prohlížeče

Překlady systémových textů jsou velice jednoduché, Chytrá domácnost používá standardní formát pro překlady – babel.

```

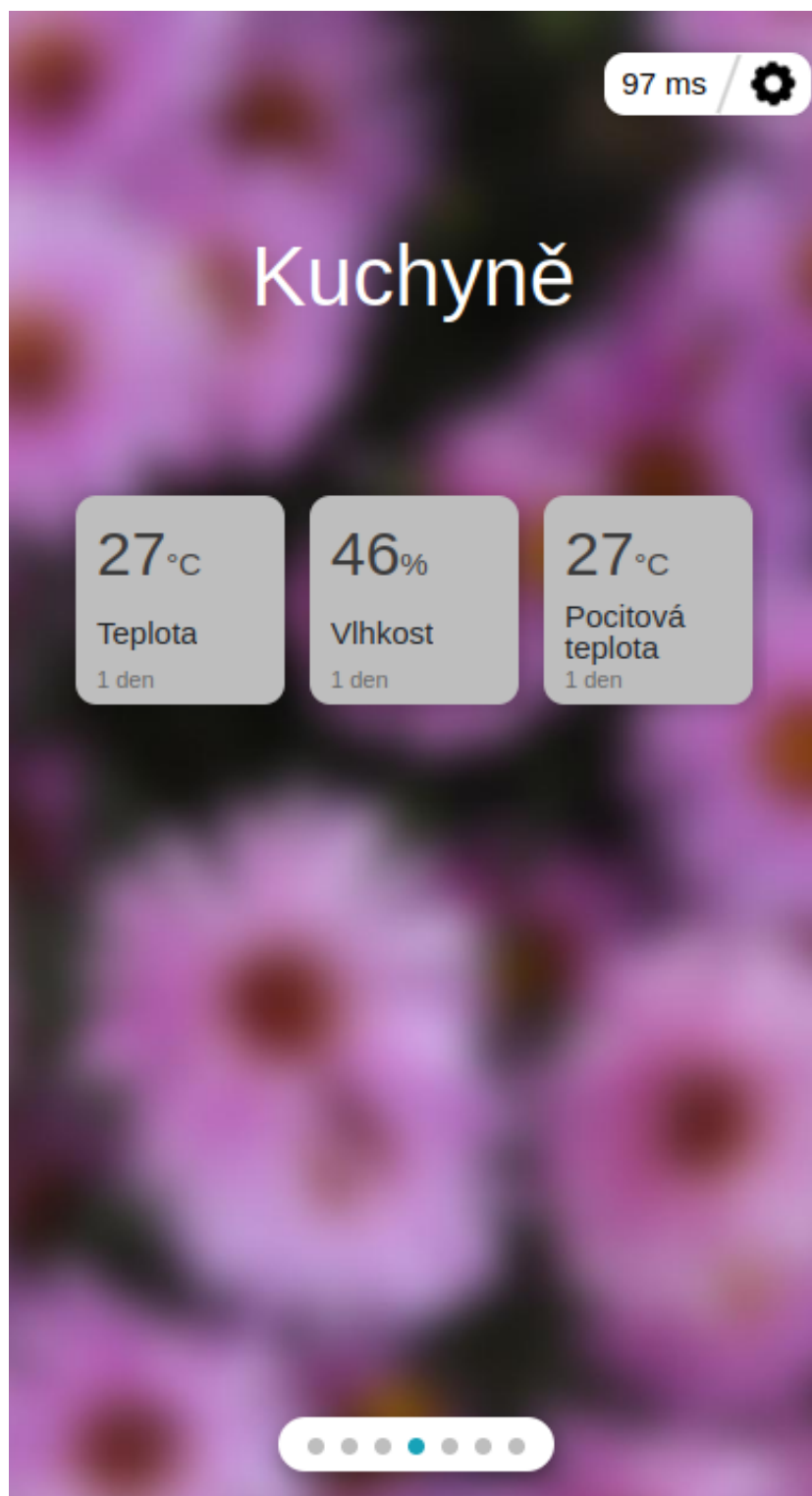
175 #: ../config/items/input.py:46
176 msgid "Confirm button"
177 msgstr "Potvrzovací tlačítko"
178
179 #: ../config/items/progress_bar.py:12
180 msgid "Progress bar"
181 msgstr "Ukazatel průběhu"
182
183 #: ../config/items/progress_bar.py:32 ../config/items/slider.py:38
184 msgid "Maximal value"
185 msgstr "Maximální hodnota"
186
187 #: ../config/items/progress_bar.py:33 ../config/items/slider.py:37
188 msgid "Minimal value"
189 msgstr "Minimální hodnota"
190
191 #: ../config/items/separator.py:12
192 msgid "Separator"
193 msgstr "Oddělovač"
194
195 #: ../config/items/slider.py:12
196 msgid "Slider"
197 msgstr "Posuvník"

```

Obrázek 2.37: Ukázka konfigurace překladu pro český jazyk

2.8 Aplikace

Součástí Chytré domácnosti je i nativní aplikace pro Android, která byla naprogramována v jazyce Java. Tato aplikace se připojuje na server stejně jako webový prohlížeč, oproti němu ale umožňuje Chytré domácnosti po získání povolení od uživatele ovládat některé části jeho mobilního telefonu skrze protokol MQTT – mobilní telefon se chová tedy stejně jako ostatní HW zařízení. Uživatel si může v Chytré domácnosti například nakonfigurovat tlačítko na prozvonění telefonu, které při stisku ihned zesílí mobilnímu telefonu hlasitost na maximum, spustí vibrační motorek na 100 ms, přehraje zvukové oznámení a několikrát po sobě aktivuje a deaktivuje blesk. Aplikace také lépe ukládá dočasné soubory, aby umožnila rychlejší spuštění aplikace.



Obrázek 2.38: Ukázka Chytré domácnosti z mobilního telefonu

Kapitola 3

Backend

Backend Chytré domácnosti je naprogramován v programovacím jazyce Python s implementací šablonového programovacího jazyku Jinja2.

3.1 Použité technologie

Backend Chytré domácnosti je naprogramován v jazyce Python za použití následujících hlavních modulů:

- Flask
- Flask-Socketio
- Flask-Login – knihovna pro zjednodušení přihlašování a registrace
- suntime – knihovna pro offline výpočet času západu a východu slunce
- opencv – knihovna pro digitální zpracování obrazu

Seznam veškerých Python balíčků včetně použitých verzí je k dispozici v příloze.

3.2 Autentizace

3.2.1 Blacklist a whitelist

Důležitým bezpečnostním prvkem je implementace znemožnění či omezení přístupu návštěvníků připojených na domácí WiFi síť k Chytré domácnosti. Smart Home umožňuje tyto základní druhy povolení/blokace přístupu zařízení:

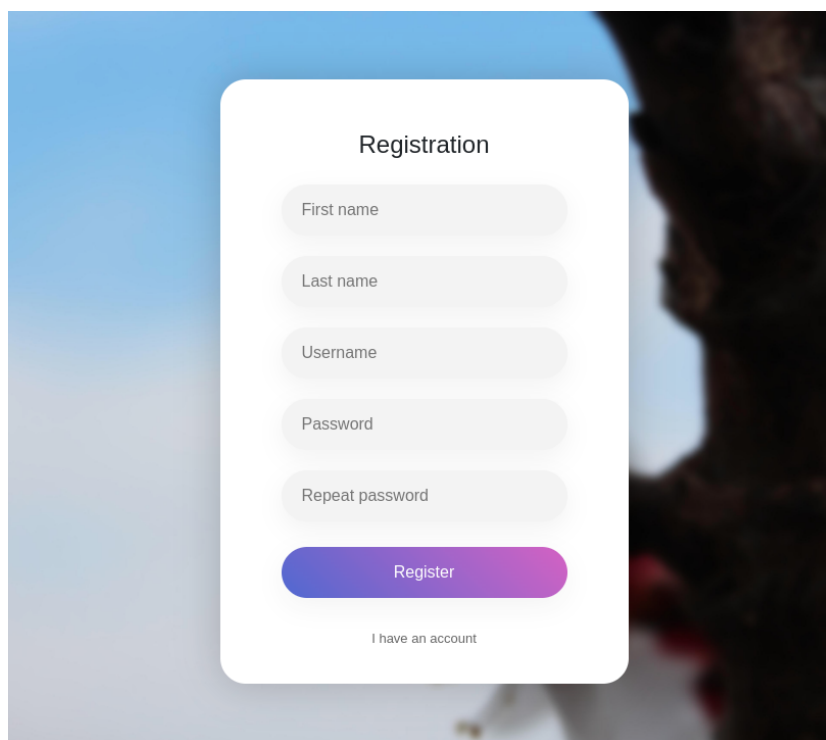
- Přiřazení nežádoucích MAC adres na blacklist – Chytrá domácnost znemožněný přístup jakémukoli zařízení s MAC adresou uvedenou na blacklistu.
- Přiřazení MAC adres domácích zařízení na whitelist – Chytrá domácnost povolí přístup jen takovým zařízením, která mají MAC adresu, která je uvedena ve whitelistu.

Použití jedné nebo kombinace z výše uvedených metod má dvě zásadní nevýhody:

- Nutná administrace zařízení pro připojení nového zařízení nebo naopak vyřazení již nepoužívaného/odpojeného zařízení.
- MAC adresy zařízení jsou na některých zařízeních volitelné a dají se měnit, což může způsobit bezpečnostní problém, pokud útočník zná MAC adresu uvedenou ve whitelistu.

3.3 Registrace

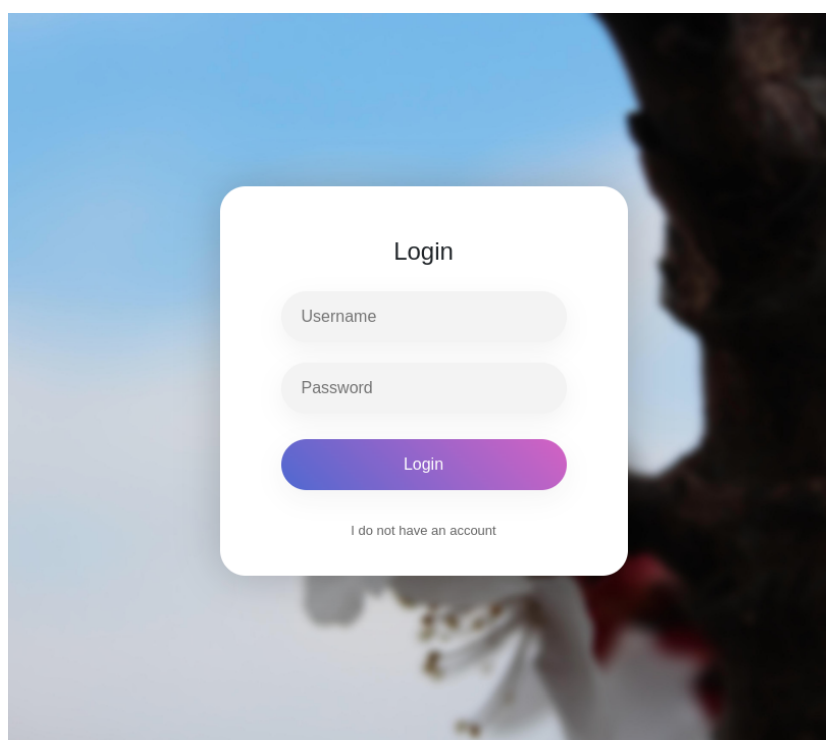
Z důvodu těchto nevýhod jsem naprogramoval přihlašovací systém a veškerý obsah Chytré domácnosti skryl za něj. Při nastavení parametru pro povolení registrací v konfiguračním souboru (viz dále) umožní Chytrá domácnost novým uživatelům využít registrační formulář a přístup do Chytré domácnosti si vytvořit. Registrační formulář je k dispozici po zadání IP adresy/domény serveru do prohlížeče a po následném kliknutí na tlačítko "Nemám ještě účet". Návrat zpět do přihlašovacího formuláře lze po kliku na tlačítko "Již mám účet"(neboli "I have an account")

A screenshot of a registration form titled "Registration". The form is centered on a white background with rounded corners, set against a blurred background of a person's profile. It contains five input fields: "First name", "Last name", "Username", "Password", and "Repeat password". Below these fields is a prominent purple "Register" button. At the bottom of the form, there is a link that says "I have an account".

Obrázek 3.1: Ukázka registračního formuláře

3.4 Přihlášení

Pokud jsou registrace povolené a uživatel si zřídil nový účet, může se nyní přihlásit pomocí přihlašovacího formuláře, který se objeví jako výchozí stránka při zadání IP adresy/domény serveru. Uživatel, který byl nově zaregistrován, obdrží základní roli návštěvníka (viz dále), kdy může volně procházet Chytrou domácnost, avšak nemůže provádět žádné akce či cokoli upravovat.



Obrázek 3.2: Ukázka přihlašovacího formuláře

3.5 Role a práva

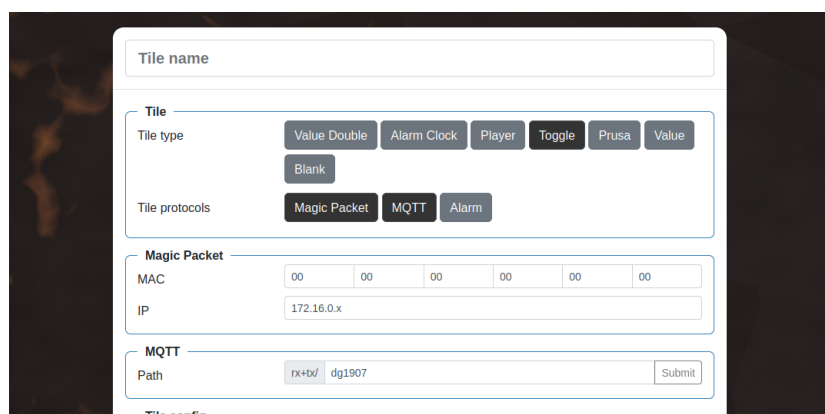
V Chytré domácnosti jsou k dispozici tyto základní role:

- **Vlastník** – Hlavní administrátor, bez omezení práv
- **Administrátor** – Má všechna práva, kromě správy uživatelů
- **Manažer** – Má všechna práva, kromě správy uživatelů a akcí týkajících se správy serveru - například nemůže zasahovat do aktuálních připojených zařízení nebo obnovovat relace.
- **Návštěvník** – Výchozí role, může akorát prohlížet zařízení v Chytré domácnosti a dívat se na jejich stav

3.6 Protokoly

V Chytré domácnost je možné namapovat protokol/y na tily a itemy. Protokoly je možné kombinovat, například na item Toggle mohu namapovat protokol Alarm i MQTT zároveň. Tato kombinace způsobí, že jakmile se aktivuje Alarm protokol (tedy nastane čas, na který je Alarm protokol nastaven), tak se aktivuje i MQTT protokol (a tedy je rozeslána informace na nakonfigurovanou cestu MQTT protokolu). Díky této možnosti je možná základní automatizace.

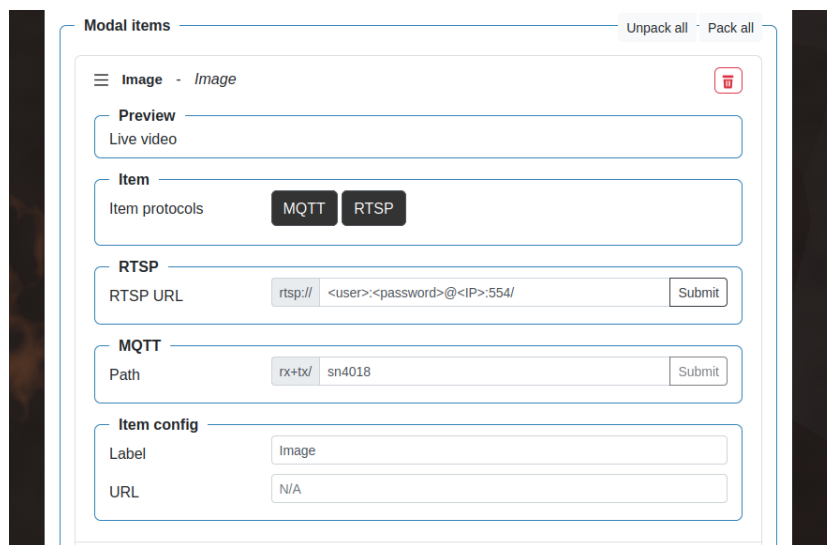
Přiřazení protokolu na tile je možné v upravovacím módu po kliknutí na vybrané protokoly v upravovacím modalu daného tilu.



Obrázek 3.3: Ukázka části modalu s protokoly pro tile

Přiřazení protokolu na item je možné provést v upravovacím režimu po rozbalení itemu kliknutím, na který chceme protokol namapovat a poté po následném kliknutí na vybrané protokoly v rozbalené části itemu.

Chytrá domácnost disponuje těmito základními protokoly, které jsou implementované přímo v jádru Chytré domácnosti:



Obrázek 3.4: Ukázka části modalu s protokoly pro item

3.6.1 MQTT

MQTT je protokol, který pracuje na principu server-klient a umožňuje jednoduchou, spolehlivou a nenáročnou komunikaci mezi IoT zařízeními, které jsou připojené do lokální sítě. [11]

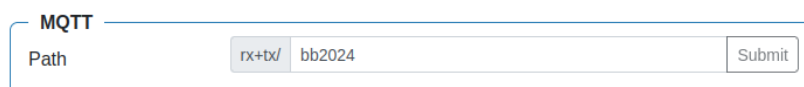
Na začátku komunikace si všichni klienti a server zdefinují hierarchicky cesty, které budou chtít přijímat a dále zpracovávat. Například při zadání:

- **pokojicek/meric/teplota** dané zařízení obdrží jakékoli zprávy od ostatních klientů/serveru, které budou mít striktně tuto cestu. Zařízení tedy neobdrží například pokojicek/meric/vlhkost nebo obyvak/meric/teplota
- **pokojicek/#** nebo **pokojicek/meric/#** dané zařízení obdrží jakékoli zprávy od ostatních klientů/serveru, které odešlou zprávu s cestou, která začíná hierarchií pokojicek nebo pokojicek/meric. Například tedy obdrží i zprávu s cestou pokojicek/meric/vlhkost.

Komunikace probíhá tak, že klient odešle do sítě zprávu se zvolenou cestou (například pokojicek/meric/teplota) a hodnotou (například 10) a všechna

ostatní zařízení, která si na začátku komunikace nadefinovala, že je tato cesta zajímavá, ji obdrží. Výše uvedený postup tedy umožňuje ovládat několik zařízení naráz. V komunikaci je dále možné nastavit několik parametrů, jako například parametr zpětnou vazbu o obdržení zprávy, atp.

V Chytré domácnosti mohou odesílat zprávy zařízení autorizované do MQTT sítě a také server.



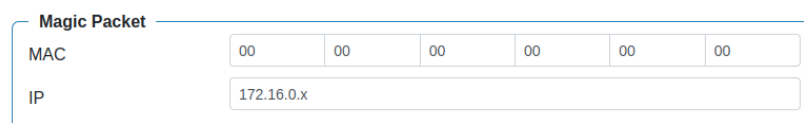
The image shows a configuration window titled "MQTT". It contains a "Path" label followed by a text input field containing "rx+tx/ bb2024". To the right of the input field is a "Submit" button.

Obrázek 3.5: Ukázka konfigurace protokolu MQTT

3.6.2 Magic Packet

Protokol Magic Packet je protokol umožňující zapnout dálkově počítač/notebook na základě MAC adresy síťové karty. Aby bylo možné počítač vzbudit, je třeba, aby měl povolené toto nastavení ve své konfiguraci v BIOSu. [12] Většina moderních počítačů a notebooků tímto nastavením disponuje a dokonce má ve výchozím stavu nastavení tuto možnost aktivovanou.

V Chytré domácnosti je poté možnost doplnit i IP adresu zařízení pro získání zpětné vazby o stavu počítače.



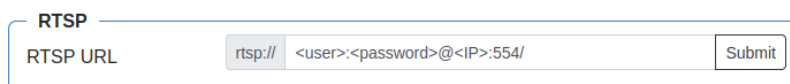
The image shows a configuration window titled "Magic Packet". It contains two rows of input fields. The first row is labeled "MAC" and has six input boxes, each containing "00". The second row is labeled "IP" and has one input box containing "172.16.0.x".

Obrázek 3.6: Ukázka konfigurace protokolu Magic Packet

3.6.3 RTSP

RTSP je protokol umožňující přístup k zobrazení živého náhledu ze streamujícího serveru. [13] Používá se například na zobrazení náhledu z kamer, chytrých domácích videotelefonů, či záznamu z NVR zařízení.

V Chytré domácnosti je možné tento protokol využít také k přenosu živého obrazu. Na bázi tohoto protokolu je postaven zabezpečovací systém pro kamery Hikvision. Zobrazování náhledu z kamer je řešeno skrze protokol MQTT a to z důvodu minimalizace zatížení hardwaru kamerového zařízení, zrychlení komunikace a zmenšení latence.

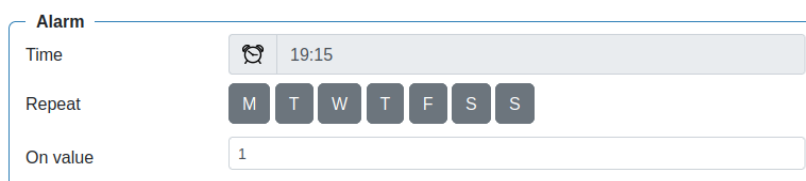


The image shows a configuration box titled "RTSP". Inside, there is a label "RTSP URL" followed by a text input field containing the placeholder text "rtsp:// <user>:<password>@<IP>:554/". To the right of the input field is a "Submit" button.

Obrázek 3.7: Ukázka konfigurace protokolu RTSP

3.6.4 Alarm

Protokol Alarm v Chytré domácnosti se používá k základní automatizaci. Po namapování tohoto protokolu na tile/item je možné nastavit čas spuštění. Vhodný je v použití s dalšími protokoly, například s protokolem MQTT.



The image shows a configuration box titled "Alarm". It contains three rows of settings: "Time" with a clock icon and the value "19:15"; "Repeat" with seven buttons labeled "M", "T", "W", "T", "F", "S", "S"; and "On value" with a text input field containing the number "1".

Obrázek 3.8: Ukázka konfigurace protokolu Alarm

3.6.5 Průša

Průša protokol – protokol, který zprostředkovává komunikaci mezi 3D tiskárnou Průša přes rozhraní Serial prostřednictvím GCode komunikace a umožňuje tak Chytré domácnosti zobrazovat data o stavu 3D tiskárny (viz tile Průša)

3.6.6 Ostatní

V Chytré domácnosti jsem používal na začátku mnohem více protokolů. Díky velkému množství paralelní kombinace protokolových serverů začalo být náročné udržovat tento soubor procesů v jednotném bezchybovém běhu. A proto jsem se rozhodl některé protokoly generalizovat a osamostatnit do vlastních služeb – jedná se například o protokol pro 3D tiskárnu Průša, který díky generalizaci navíc může běžet na jakémkoli zařízení připojeného do lokální sítě s připojením k serialu tiskárny, nebo částečně i RTSP (viz výše).

3.7 Terminál

Pro rychlejší a přehlednější debugování jsem postupem času naprogramoval grafické prostředí v terminálu. Při spuštění Chytré domácnosti v debugovacím módu nejprve zobrazím pár základních informací včetně IP adresy serveru a poté začnu logovat jednotlivé změny uživatelem, logy z protokolů, nebo chybové hlášky. Veškeré informace jsou také uloženy v log souborech.



```
Smart Home
↓ About ↓
Version IoT 11.8
Created 26.05.2019 17:29

↓ Server info ↓
URL http://172.16.0.3:5000/
Python 3.9.5
Mode gevent
OS linux 64bit
OS ver 5.11.0-38-generic

↓ Logs and other information ↓
MQTT Log (16): Sending CONNECT (u1, p1, wr0, wq0, wf0, c1, k60) client_id=b''
MQTT Log (16): Received CONNACK (0, 0)
MQTT Connected with result code 0
MQTT Log (16): Sending SUBSCRIBE (d0, m1) [(b'err/#', 0)]
MQTT Log (16): Sending SUBSCRIBE (d0, m2) [(b'warn/#', 0)]
MQTT Log (16): Sending SUBSCRIBE (d0, m3) [(b'hello/#', 0)]
MQTT Log (16): Sending SUBSCRIBE (d0, m4) [(b'debug/#', 0)]
MQTT Log (16): Sending SUBSCRIBE (d0, m5) [(b'rx/bed', 0)]
MQTT Log (16): Sending SUBSCRIBE (d0, m6) [(b'rx/bed/startup-time', 0)]
MQTT Log (16): Sending SUBSCRIBE (d0, m7) [(b'rx/bed/measure-time', 0)]
```

Obrázek 3.9: Ukázka náhledu na terminál

3.8 Struktura kódu

Chytrou domácnost jsem rozdělil na dvě části:

- **Statická část** – bez této části se Chytrá domácnost neobejde, zařizuje základní funkce a zprostředkování komunikace, patří sem například základní grafické prostředí Chytré domácnosti, backendové jádro, atp.
- **Dynamická** – jedná se o jednoduše vyměnitelné části s programovou abstraktní definicí, například protokoly, tily, itemy, ...

Strukturu kódů jsem rozdělil následovně zejména kvůli potřebě dynamičnosti:

- **/config/**
 - **/config/items/** – Python konfigurační soubory pro itemy, definují backendové chování, zpracovávání hodnot, ...
 - **/config/tiles/** – Python konfigurační soubory pro tily, definují backendové chování, zpracovávání hodnot, ukládání, ...
 - **/config/protocols/** – Python konfigurační soubory pro protokoly, definují přes co se budou jaké informace posílat
 - **/config/main.ini** – základní konfigurační soubor
 - **/config/babel.ini** – konfigurační soubor pro automatické překládání
 - **/config/whitelist.ini** – seznam povolených
 - **/config/blacklist.ini** – seznam blokových
- **/data/**
 - **/data/database.sqlite** – databáze uživatelů

- **/data/devices.json** – do tohoto souboru jsou ukládány veškeré změny v Chytré domácnosti, kvůli omezenému počtu zápisů na SD kartu v Raspberry Pi jsou data zapisována nejprve na RAM a až po určité době na SD kartu
- **/docs/** – dokumentační soubory
- **/library/** – charakteristické Python knihovny, například
 - **/library/imng.py** – image manager, zajišťuje obrazovou analýzu
 - **/library/prevent_hack.py** – kontroluje veškerý provoz s Chytrou domácností na známé útoky, validuje veškeré vstupy i na backendové části
 - **/library/validator.py** – před spuštěním validuje veškeré soubory, zda nejsou poškozené a zda jsou spustitelné
 - **/library/sun.py** – tento soubor počítá polohu slunce a na základě toho offline rozhoduje, zda je den, či noc
 - **/library/clients.py** – tento soubor se stará o správné uchování informací ohledně posledních relací uživatelů (tedy vždy, když si uživatel zobrazí Chytrou domácnost, objeví se na tom místě (ploše, modalu, ..), kde ukončil svou předchozí relaci)
 - **/library/terminal.py** – stará se o přehledné a barevné logování do terminálu v případě debugovacího režimu
 - **/library/logger.py** – loguje logy do složky logs
 - **/library/raspberry.py** – stará se o HW funkcionalitu Raspberry Pi, například měří teplotu CPU a nastavuje rychlost ventilátoru pro chlazení
 - **/library/fmng.py** – stará se o optimalizované načítání a ukládání JSON souborů (soubory se načtou pouze jednou, neukládají se zbytečně každou milisekundu, ...)

- **/library/tmng_read.py** – template manager, zajišťuje funkcionálnitu pro čtení k Flask template souborům, tento soubor tedy přečte dynamický text (například název tilu) a doplní ho na příslušné místo
- **/library/tmng_rewrite.py** – template manager, tento soubor se stará o správné uložení upravených informací načtených pomocí `tmng_read`
- **/library/tmng_rewrite.py** – template manager, tento soubor se stará o čistě zapisovací akce, například smazání tilu
- ...
- **/logs/** – logy separované po dnech a automaticky mazané po určité době
 - **/logs/auth.log** – log ukládající autorizace i pokusy o autorizaci
 - **/logs/changes.log** – log ukládající změny stavů tilů, itemů, ...
 - **/logs/changes_edit.log** – log ukládající změny v upravovacím režimu (např. změnění názvu tilu)
 - **/logs/terminal.log** – log ukládající normální konzolový výstup
- **/scripts/**
 - **/scripts/install.sh** – po spuštění nainstaluje moduly a balíčky potřebné pro spuštění Chytré domácnosti
 - **/scripts/chromium_kiosk.sh** – aktivuje grafické drivery Raspberry Pi a zobrazí jen okno s Chytrou domácností bez rámečků
 - **/scripts/line_counter.sh** – spočítá počty řádků veškerého kódu rozděleného dle typů souborů
 - **/scripts/translations.sh** – zkompileje Babel překladové soubory
 - ...

- **/static/**
 - **/css/** – veškeré CSS soubory, například Bootstrap, Animate, Swiper, CSS k itemům, tilům, ...
 - **/js/** – veškeré JS soubory, například JQuery, Fotorama, Sortable, SocketIO, Popper, ChartJS, ...
 - * **/js/items/** – JS funkcionalita k generaci a ovládání itemů; 1 item = 1 JS soubor
 - * **/js/tiles/** – JS funkcionalita k generaci a ovládání tilů; 1 tile = 1 JS soubor
 - * ...
 - **/img/**
 - * **/img/background/** – všechna pozadí, ve vnořené složce poté pozadí již připravená pro zobrazení uživateli – Chytrá domácnost pozadí automaticky kompresuje a rozmazává, aby vynikly ovládací prvky
 - * **/img/icons/** – vlastně navržené dynamické ikony, které je možné použít jako ikonu například tilu
 - * **/img/static/** – ostatní statické obrázky a statické ikony
 - **/music/** – zvukové stopy pro tile "Přehrávač"
 - **/sound/** – zvukové oznámení, například cinknutí
- **/temp/** – dočasné soubory
 - **/temp/background_data.json** – data o pozadích (jejich klasifikace na tmavá/světlá, nejčastější barvy v nich, ...)
 - **/temp/refresh_data.json** – cache aktuálních otevřených oken s pozicemi v Chytré domácnosti pro automatické otevření na stejném místě po zavření prohlížeče (uživatel tedy neztratí pozici)
- **/templates/**

- **/templates/auth/** – HTML soubory používané pro přihlašování a registrování
- **/templates/items/** – HTML kostra itemů; 1 item = 1 HTML soubor
- **/templates/tiles/** – HTML kostra tilů; 1 tile = 1 HTML soubor
- **/modal_edit/** – prvky zobrazované jen v upravovacím modalu
- **/templates/base.html** – základní HTML soubor, který je implementován všemi ostatními, je v něm základní importování knihoven, atp.
- **/templates/index.html** – soubor, který se pošle klientovi po načtení Chytré domácnosti
- **/templates/error.html** – soubor, který se zobrazí při chybové situaci (404, 403, 500, ...)
- **/templates/modal_android.html** – soubor s modalem, který se zobrazí na Android zařízeních a umožní jim stáhnout si aplikaci Chytré domácnosti přímo z webového rozhraní
- **/templates/modal_client_list.html** – soubor s modalem, který zobrazuje výčet aktivních relací (seznam všech klientů)
- **/templates/modal_user_list.html** – soubor s modalem, který zobrazuje výčet všech zaregistrovaných uživatelů (viz výše)
- **/templates/modal_settings.html** – soubor s modalem, který umožňuje uživateli si přizpůsobit prostředí Chytré domácnosti (viz výše)
- **/templates/modal_normal.html** – soubor s modalem, který se zobrazí při podržení na tilu v normálním režimu (viz výše)
- **/templates/modal_edit.html** – soubor s modalem, který se zobrazí při kliknutí na tile v upravovacím režimu (viz výše)

- **/templates/preloader.html** – soubor s načítacím kolečkem, které se točí, dokud se nenačte celá Chytrá domácnost, aby načítání nevypadalo skokově a nevzhledně (viz výše)
- **/templates/slide.html** – soubor s definicí jedné plochy (viz výše)
- **/translations/**
 - **/translations/cs/** – české jazykové překlady
 - **/translations/ru/** – ruské jazykové překlady

3.9 Konfigurace

Neuživatelskou konfiguraci Chytré domácnosti jsem vyřešil pomocí konfiguračních souborů s příponou `.ini`.

```
[default]
# Can application start?
run = yes

# Allow registrations
registrations = no

[refresh]
# Save information before browser refresh (current slide, modal, ..) and then send them back
save = yes
# Time to save these information in seconds
time = 604800

# Image manager settings
[img]
# Reclassify on start
reclassify = no
limit-dark = 60

# Set geographic position of current location to calculate sunset and sunrise
[position]
latitude = 49.638994
longitude = 17.148784

# Set log priorities
[logs]
log_only = false

changes_priority = 0
changes_edit_priority = 0
terminal_priority = 0
auth_priority = 0

# Make validations?
[validation]
json = yes
```

Obrázek 3.10: Ukázka konfiguračního `.ini` souboru

Kapitola 4

Server a jeho zabezpečení

4.1 Server

Server pro Chytrou domácnost je možné umístit na jakýkoli PC se síťovou kartou a připojením do vnitřní lokální domácí sítě. Doporučené systémové minimální požadavky jsou velmi malé:

- **CPU:** 2 jádra
- **RAM:** 1 GB
- **Disk:** 1 GB
- **OS:** Linux, Mac, Windows

Vhodný serverový počítač může být tedy například Raspberry Pi 3B, Raspberry Pi 4B, nebo jakýkoli jiný mini PC. Já zvolil Raspberry Pi 4B díky nízké spotřebě energie a dobré ceně.

4.2 Zabezpečení

Zabezpečení serveru má několik úrovní.

4.2.1 Uživatelské vstupy

Všechna komunikace přicházející od klienta je validována. Neustále probíhá kontrola na všeobecně známé útoky, jako například [XSS](#), [Path Traversal](#), [SSTI](#), nebo [SQL Inject](#). Také je kontrolována správnost příchozích dat, zda uživatel neobešel, či jinak neupravil validující podmínky napsané v JavaScriptu ve frontendové části.

Pokud je nějaký z těchto útoků detekován, či Chytrá domácnost zjistí, že uživatel odstranil kontrolní podmínky ve frontendu a pokusil se poslat Chytré domácnosti nevalidní vstup, je tento uživatel a toto zařízení automaticky zablokováno (přidáno na blacklist).

4.2.2 Nginx

Na serveru běží služba [Nginx](#), která zrychluje posílání statických souborů klientovi, ale také zvyšuje bezpečnost díky použití protokolu HTTPS s vlastně podepsaným certifikátem.

4.2.3 Firewall

Další úroveň zabezpečení Chytré domácnosti je firewall, který je nastavený velmi přísně, do sítě jsou otevřené pouze tyto porty:

- **22** – port pro vzdálené připojení pomocí protokolu SSH (autentizace probíhá přes SSH klíč)

- **80** – port pro komunikaci skrze *HTTP* protokol; veškerý provoz je přeměřovaný na port 443 a je vynucena komunikace skrze protokol *HTTPS*
- **443** – port pro výchozí komunikaci s klienty skrze protokol *HTTPS*
- **1883** – port pro možnou komunikaci s HW zařízení - skrze *MQTT* protokol

Kapitola 5

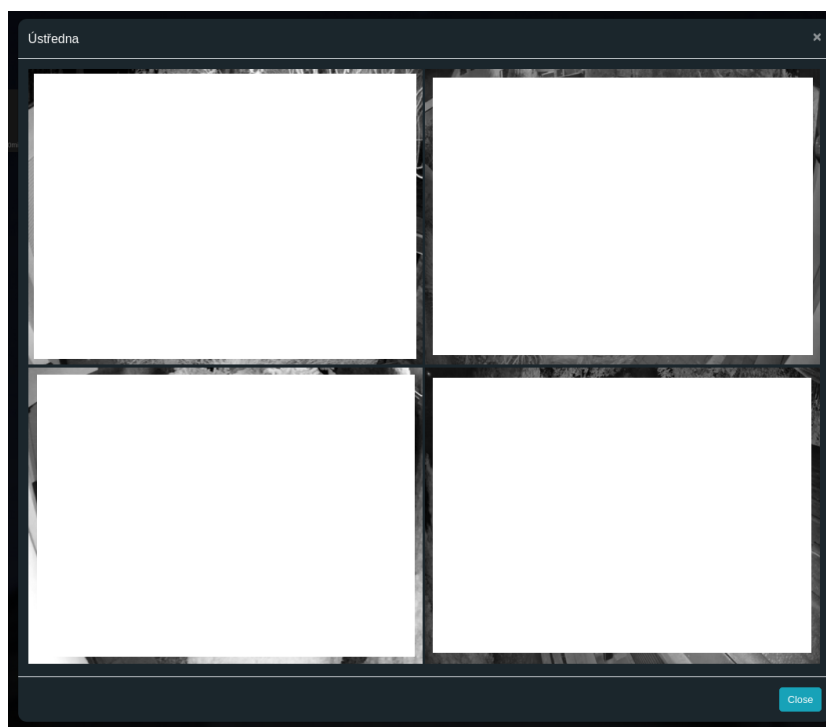
Zařzení

K Chytré domácnosti jsem navrhl, vyrobil a sestrojil několik funkčních hardwarových zařízení. Jako řídicí jednotky jsem použil hlavně jednotky s integrovanou WiFi, například ESP8266, nebo Arduina.

5.1 Kamerový systém

Kamerový systém od firmy Hikvision sice disponuje funkcemi pro zastřežení prostoru, bohužel velmi často dochází k chybám a alarm se spouští ve většině případů falešný poplach například z důvodu větru, kdy dochází k pohybu větví stromů, nebo při průchodu zvířete. Proto jsem se rozhodl naprogramovat vlastní střežící SW, který detekuje inteligentně jakýkoli pohyb člověka. Tento systém je výjimečný zejména tím, že má vysokou senzitivitu i specifitu. Systém pro zabezpečení je v konstantním provozu již zhruba 11 měsíců a mohu potvrdit, že za tuto dobu nepřehlédl jediný pohyb člověka a falešně spouštěl poplach jen ze začátku, kdy jsem ho ještě optimalizoval (senzitivita je tedy zatím 100 % a specifita je téměř 100 %). Kamerový systém funguje na základě inteligentní kombinace matematického modelu s deterministickými obrazovými metodami a také hlubokých konvolučních neuronových

sítí. Nechybí také ani vzdálené upozornění na pohyb včetně fotografií skrze moderní chatovací platformu Telegram.



Obrázek 5.1: Náhled na modal s kamerami

5.2 Domácí videotelefon

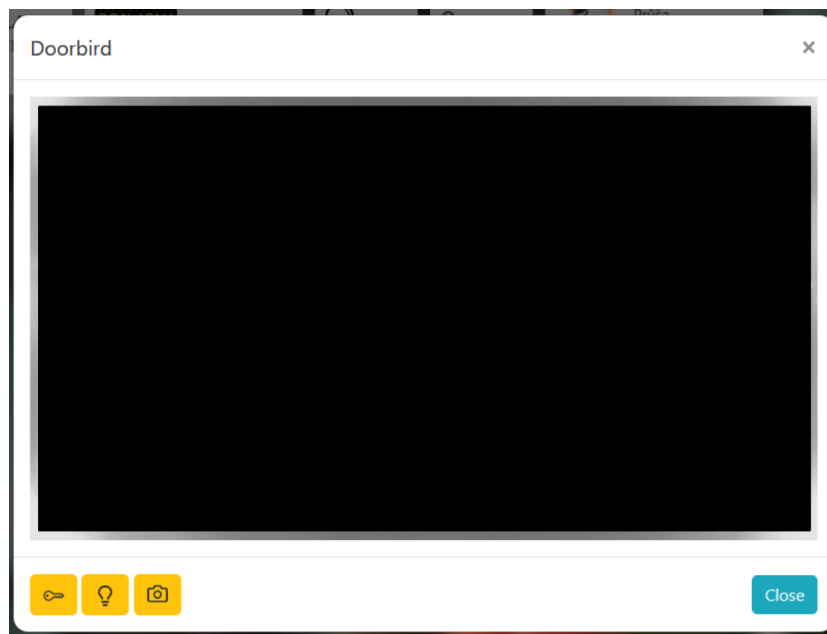
Naprogramoval jsem rozhraní pro napojení na otevřené API domácího videotelefonu od značky [Doorbird](#). Pro tuto značku jsem se rozhodl na základě splnění mých předem daných podmínek:

- Produkty dané firmy musí být k dispozici v běžných českých obchodech
- Daný produkt musí mít jakoukoli možnost ovládání pro vývojáře skrze otevřené [API](#)

Implementoval jsem několik funkcionalit, například:

- Živý náhled přes kameru Doorbirdu
- Otevření dveří pomocí spínacího relé připojeného ke zvonku
- Automatické sekvenční snímání a nahrávání po i před detekcí pohybu včetně audio stopy
- Po zapnutí automatické rozpoznávání obličeje a otevírání dveří na základě bio-identity

Tyto funkcionality jsou k dispozici po kliknutí na tlačítko "Doorbird" v systémové nabídce.

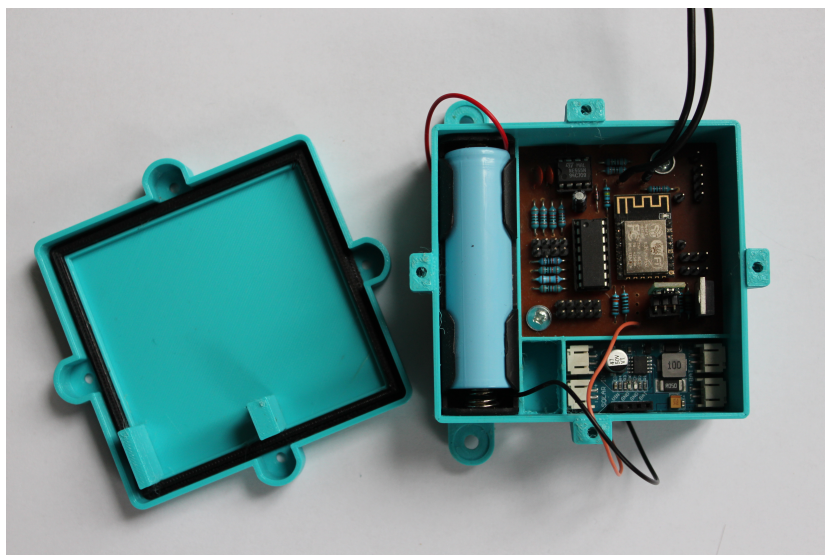


Obrázek 5.2: Náhled na modal Doorbird

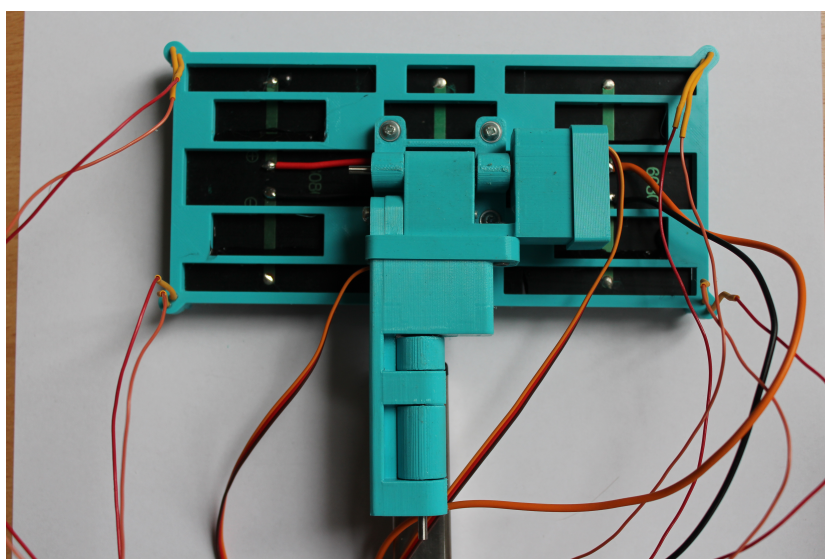
5.3 Měření ve skleníku

Jedno z mnoha HW zařízení, které jsem vyrobil, je autonomní mechanismus nezávislý na přístupu k elektrické síti, které kontroluje teplotu a vlhkost

vzduchu ve skleníku a vlhkost půdy v něm. Tato data poté odesílá do Chytré domácnosti, která je následně ukládá například do grafu či jej zobrazuje v modalech nebo na tilech.



Obrázek 5.3: Ukázka zhotovené otevřené krabičky pro řídicí jednotku



Obrázek 5.4: Ukázka zhotovené otáčecí konstrukce

Schéma a návrh plošného spoje jsem vyhotovil v programu Eagle od spo-

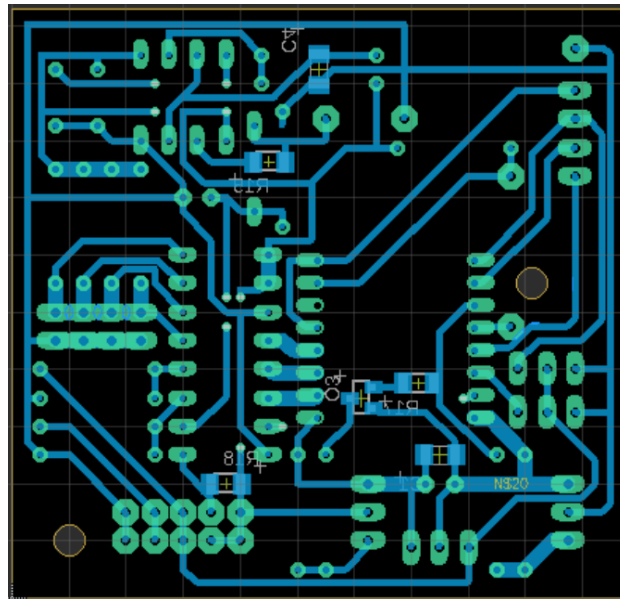
lečnosti Autodesk. Jako řídicí jednotku jsem zvolil čip ESP8266 od firmy <https://www.espressif.com/>, ke kterému jsem připojil AD převodník MCP3008. Tento převodník následně měří napětí na těchto částech:

- Napětí na 4 fotorezistorech
- Napětí na solárních panelech
- Napětí na baterii
- Napětí na výstupu z bloku měřiče vlhkosti v půdě
- Napětí na signálním výstupu z ovladače dobíjení baterie

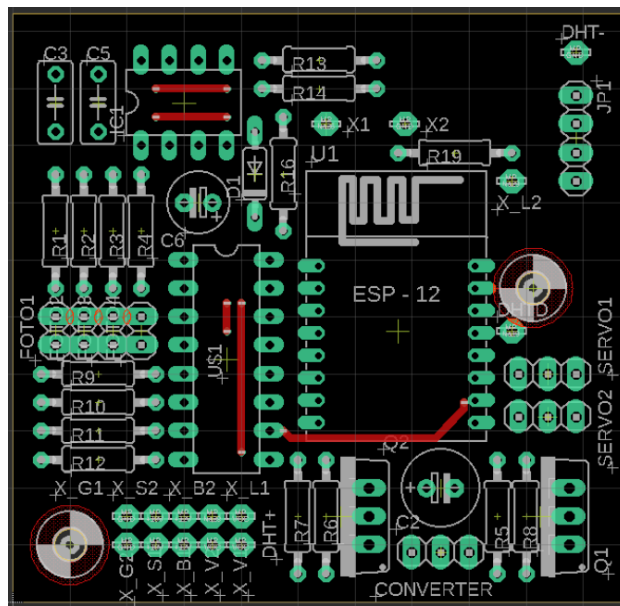
K ESP8266 je dále připojeno teplotní a vlhkostní čidlo DHT22, dvě serva pro automatické natáčení solárních panelů za sluncem a MOSFET pro odpojení veškeré elektroniky při přechodu do hlubokého spánku. Program ESP je velmi optimalizovaný, veškeré potřebné hodnoty si ukládá do RTC paměti a probouzí se v pravidelných intervalech pro měření hodnot. Pokud se baterie blíží stavu vybití, ESP informuje Chytrou domácnost a při dalším poklesu napětí na baterii začne prodlužovat spánek, díky kterému minimalizuje svou spotřebu energie.

3D návrh krabičky a otočné konstrukce jsem navrhl v CADu SolidWorks a poté jej vytiskl na 3D tiskárně Průša z materiálu PETG při standardních teplotách. Materiál PETG jsem zvolil z důvodu vysoké mechanické odolnosti.

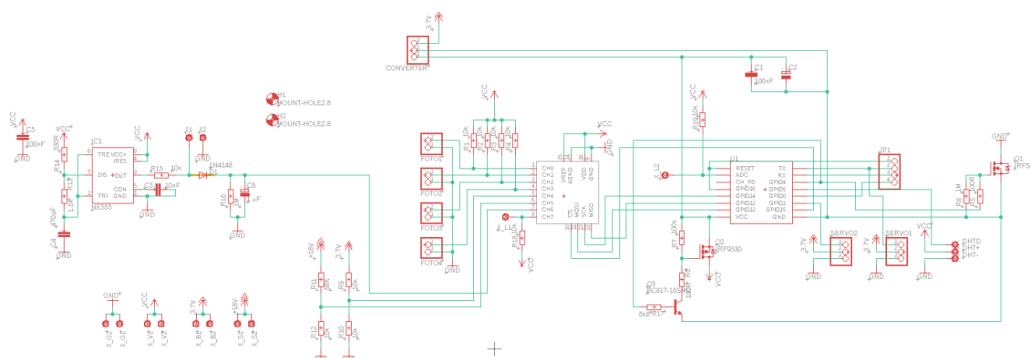
V příloze této práce jsou k dispozici přesné návrhy z programu SolidWorks



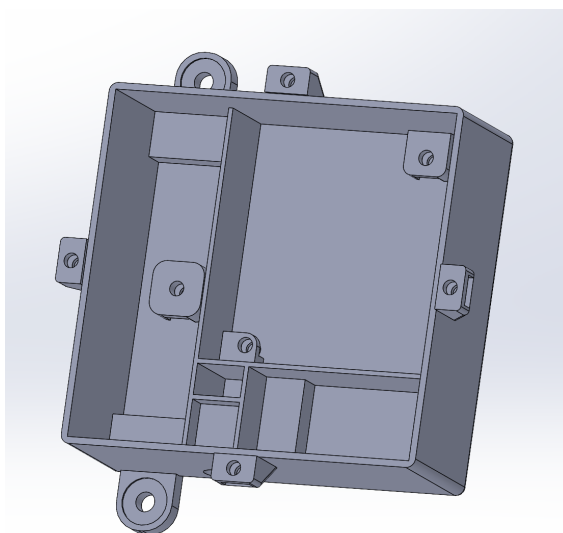
Obrázek 5.5: Návrh plošného spoje



Obrázek 5.6: Návrh osazení plošného spoje



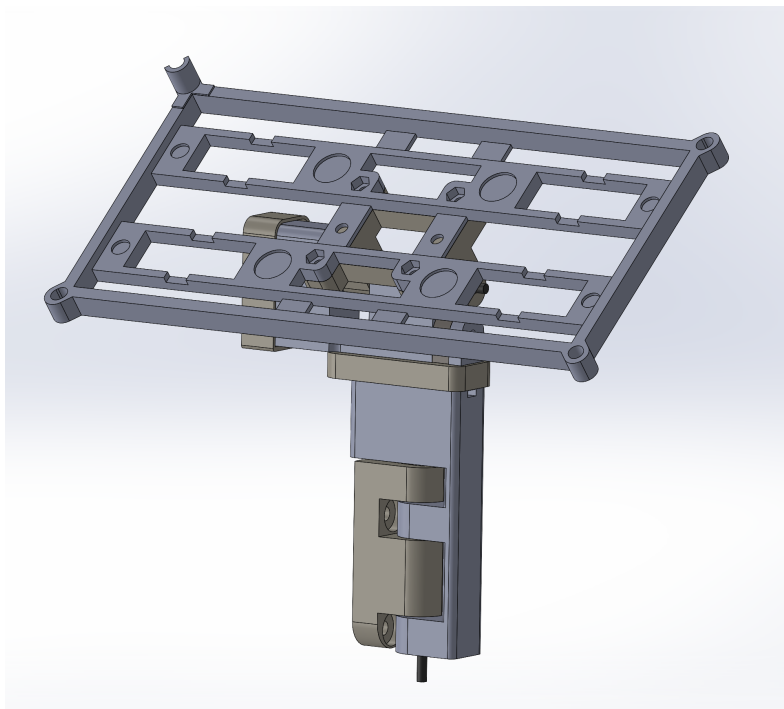
Obrázek 5.7: Schéma obvodu



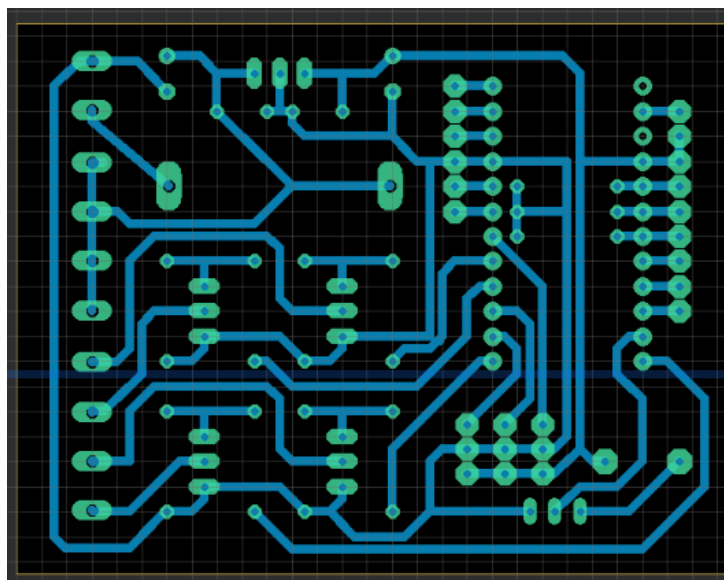
Obrázek 5.8: Ukázka 3D modelu otevřené krabičky pro řídicí jednotku

5.4 Ovládání LED

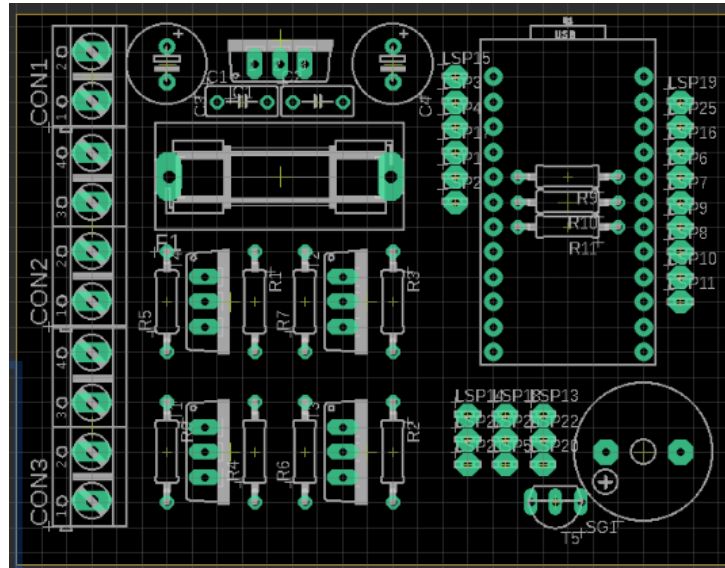
Jako další příklad HW zařízení mohu uvést například jednoduchý PWM ovladač LED pásků. Tento obvod jsem sestrojil za účelem vytvoření zařízení ovládající LED pásky pomocí pulzně šířkové modulační pod postelí. Schéma, návrh a osazení plošného spoje jsem navrhl v programu Autodesk Eagle a krabičku opět v programu SolidWorks.



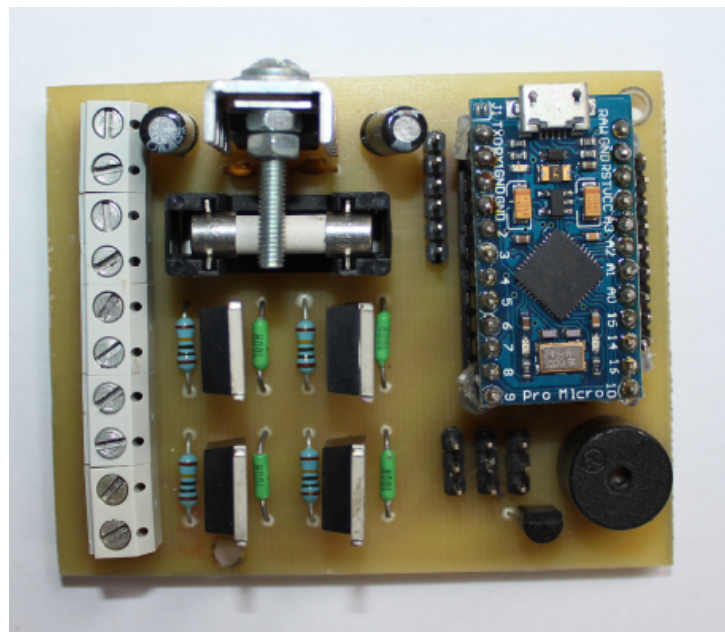
Obrázek 5.9: Ukázka 3D modelu otáčecí konstrukce



Obrázek 5.10: Návrh plošného spoje



Obrázek 5.11: Návrh osazení plošného spoje



Obrázek 5.12: Ukázka osazeného plošného spoje

Závěr

Během tří let intenzivní práce se mi podařilo vytvořit a naprogramovat Chytrou domácnost, která jednoduše a efektivně zprostředkovává ovládání a zobrazování informací jakýchkoli hardwarových zařízení včetně zařízení od výrobců třetí strany díky otevřenému API. Také jsem vyrobil několik HW zařízení prezentující schopnosti ovládání Chytré domácnosti s napojením na inteligentní funkce zejména z oboru obrazové analýzy.

Chytrá domácnost používám již aktivně dlouhodobě používám, konkrétně mi doma běží již od prosince roku 2020. Postupně jsem k domácnosti připojil kamerový systém Hikvision a domácí videotelefon Doorbird.

Každým rokem k Chytré domácnosti postupně přidávám další HW zařízení a rozšiřuji ji o další SW funkce a moduly. Aktuálně pracuji na připojení bezdrátových radiových periferií, jako je například ovladač od vstupní brány, nebo od vrat garáže.

Zdroje

1. ŠTRÁFELDA, Jan. *Co je server* [online] [cit. 2022-03-06]. Dostupné z: <https://www.strafelda.cz/server>.
2. ŠTRÁFELDA, Jan. *Frontend* [online] [cit. 2022-03-06]. Dostupné z: <https://www.strafelda.cz/frontend>.
3. ŠTRÁFELDA, Jan. *Backend* [online] [cit. 2022-03-06]. Dostupné z: <https://www.strafelda.cz/backend>.
4. ESET. *Seznam povolených* [online] [cit. 2022-03-06]. Dostupné z: <https://help.eset.com/glossary/cs-CZ/whitelist.html>.
5. ESET. *Seznam spamových adres* [online] [cit. 2022-03-06]. Dostupné z: <https://help.eset.com/glossary/cs-CZ/blacklist.html>.
6. ŠTRÁFELDA, Jan. *HTML* [online] [cit. 2022-03-06]. Dostupné z: <https://www.strafelda.cz/html>.
7. JAKPSÁTWEB. *CSS* [online] [cit. 2022-03-06]. Dostupné z: <https://www.jakpsatweb.cz/css/css-uvod.html>.
8. JAKPSÁTWEB. *JavaScript* [online] [cit. 2022-03-06]. Dostupné z: <https://www.jakpsatweb.cz/javascript/javascript-uvod.html>.
9. PYTHON. *Python* [online] [cit. 2022-03-06]. Dostupné z: <https://python.cz/>.
10. JINJA. *Jinja2* [online] [cit. 2022-03-07]. Dostupné z: <https://jinja.palletsprojects.com/en/3.0.x/>.

11. MQTT. *MQTT* [online] [cit. 2022-03-10]. Dostupné z: <https://mqtt.org/>.
12. YATRITRIVED. *Magic Packet - Wak on LAN* [online] [cit. 2022-03-10]. Dostupné z: <https://www.howtogeek.com/70374/how-to-geek-explains-what-is-wake-on-lan-and-how-do-i-enable-it/>.
13. RUETHER, Traci. *Real Time Streaming Protocol* [online] [cit. 2022-03-10]. Dostupné z: <https://www.wowza.com/blog/rtsp-the-real-time-streaming-protocol-explained>.

Seznam zkratek

Zkratka	Význam
HW	Hardware
SW	Software
IoT	Internet of Things
PC	Počítač
CAD	Computer aided design
URL	Uniform Resource Locator
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
JS	Javascript
SQL	Structured Query Language
SSTI	Server-Side Template Injection
XSS	Cross-Site Scripting
RTSP	Real-Time Streaming Protocol
SSH	Secure Shell
HTTPS	Hypertext Transfer Protocol Secure
IP	Internet Protocol
MAC	Media Access Control
DPS	Deska plošného spoje
RTC	Real-Time clock
PWM	Pulse Width Modulation
LED	Light-Emitting Diode
IO	Input-Output
BIOS	Basic Input-Output System
NVR	Network video recorder

Seznam obrázků

2.1	Příklady vyfocených pozadí	17
2.2	Ukázky nakreslených ikon	18
2.3	Náhled na první plochu Chytré domácnosti v normálním módu	20
2.4	Ukázka několika typů tilů na poslední ploše	21
2.5	Ukázka tilu typu blank	22
2.6	Ukázky stavů tilu typu Toggle	22
2.7	Ukázka tilu typu Value	23
2.8	Ukázka tilu typu Value Double	23
2.9	Ukázka tilu typu Průša	24
2.10	Ukázka tilu typu Alarm Clock	24
2.11	Ukázka tilu typu Player	25
2.12	Ukázka modalu pro tile z první plochy	26
2.13	Ukázka několika konfigurací itemu tlačítko	28
2.14	Ukázka konfigurací itemu skupina tlačítek	29
2.15	Ukázka dvou konfigurací itemu posuvník	31
2.16	Ukázka dvou konfigurací itemu přepínač	31

2.17 Ukázka konfigurací itemu rozbalovací nabídka	32
2.18 Ukázka itemu oddělovač	33
2.19 Ukázka konfigurací itemu ukazatel průběhu	34
2.20 Ukázka konfigurací itemu časový vstup	35
2.21 Ukázka konfigurací itemu vstupní pole	36
2.22 Ukázka systémové nabídky v normálním módu	38
2.23 Modal nastavení	39
2.24 Ukázky motivů na normálních modalech	40
2.25 Ukázky motivů na upravovacích modalech	40
2.26 Seznam uživatelů	42
2.27 Seznam klientů	43
2.28 Ukázka Chytré domácnosti na první ploše v upravovacím režimu	44
2.29 Ukázka možnosti úpravy názvu plochy	45
2.30 Zobrazení nového tlačítka pro přidání nového tilu	46
2.31 Ukázka modalu pro tile z poslední plochy	47
2.32 Ukázka správy více itemů	48
2.33 Ukázka správy jednoho itemu	49
2.34 Ukázka systémové nabídky v upravovacím režimu	50
2.35 Ukázka načítání před zobrazením domovské stránky	52
2.36 Příklady automaticky přeložených textů dle jazyka prohlížeče	53
2.37 Ukázka konfigurace překladu pro český jazyk	54
2.38 Ukázka Chytré domácnosti z mobilního telefonu	55

3.1	Ukázka registračního formuláře	58
3.2	Ukázka přihlašovacího formuláře	59
3.3	Ukázka části modalu s protokoly pro tile	61
3.4	Ukázka části modalu s protokoly pro item	62
3.5	Ukázka konfigurace protokolu MQTT	63
3.6	Ukázka konfigurace protokolu Magic Packet	63
3.7	Ukázka konfigurace protokolu RTSP	64
3.8	Ukázka konfigurace protokolu Alarm	64
3.9	Ukázka náhledu na terminál	66
3.10	Ukázka konfiguračního .ini souboru	73
5.1	Náhled na modal s kamerami	78
5.2	Náhled na modal Doorbird	79
5.3	Ukázka zhotovené otevřené krabičky pro řídicí jednotku	80
5.4	Ukázka zhotovené otáčecí konstrukce	80
5.5	Návrh plošného spoje	82
5.6	Návrh osazení plošného spoje	82
5.7	Schéma obvodu	83
5.8	Ukázka 3D modelu otevřené krabičky pro řídicí jednotku . . .	83
5.9	Ukázka 3D modelu otáčecí konstrukce	84
5.10	Návrh plošného spoje	84
5.11	Návrh osazení plošného spoje	85

5.12 Ukázka osazeného plošného spoje	85
--	----

Seznam příloh

1. Výkres 3D modelu mechanismu solárních panelů
2. Výkres 3D modelu krabičky k mechanismu solárních panelů
3. Výkres + schéma DPS řídicí jednotky solárních panelů
4. Výkres 3D modelu krabičky k PWM LED ovladači
5. Výkres + schéma DPS LED ovladače
6. Verze použitých knihoven v Pythonu