



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ**
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

UNIVERZÁLNÍ KOMUNIKAČNÍ CENTRUM PRO INTELIGENTNÍ DOMÁCNOST S VYUŽITÍM OPENHAB

**UNIVERSAL COMMUNICATION CENTER FOR THE SMART HOME BASED ON PLATFORM
OPENHAB**

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

JAN VESELKA

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. DOMINIK KOVÁČ

BRNO 2015



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student: Jan Veselka

ID: 146128

Ročník: 3

Akademický rok: 2014/2015

NÁZEV TÉMATU:

Univerzální komunikační centrum pro inteligentní domácnost s využitím OpenHAB

POKYNY PRO VYPRACOVÁNÍ:

V rámci teoretické části bakalářské práce se seznámte se systémem openHAB (open Home Automation Bus). Prostudujte a zdokumentujte dostupné bindings (balíčky zajišťující podporu komunikace daného zařízení s openHAB) sloužící pro ovládání inteligentních periférií v domě. OpenHAB zprovozněte na zařízení Raspberry Pi a vytvořte demonstrátor, který bude znázorňovat funkčnost openHABu s perifériemi z řady Homematic. V rámci platformy openHAB vytvořte aplikaci pro smartphone, která umožní periférie ovládat a monitorovat. Celý systém se snažte navrhovat tak, aby byl jednoduše rozšiřitelný o další prvky / senzory domácí automatizace. Vše pečlivě sepište do podoby bakalářské práce.

DOPORUČENÁ LITERATURA:

- [1] HERSENT, Olivier, David BOSWARTHICK a Omar ELLOUMI. The internet of things: applications to the smart grid and building automation. Hoboken: John Wiley, 2012, xxv, 344 s. ISBN 1119994357.
[2] GOODWIN, Steven. Smart home automation with linux and raspberry pi. Berkeley: Apress. ISBN 978-1430258872.

Termín zadání: 9.2.2015

Termín odevzdání: 2.6.2015

Vedoucí práce: Ing. Dominik Kováč

Konzultanti bakalářské práce:

doc. Ing. Jiří Mišurec, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Elektronických periférií v domácnosti je v dnešní době nespočet, proto je výhodné ovládat je v pohodlí a přehledně, například za pomoci chytrého mobilního telefonu. Tato práce řeší právě tuto možnost ovládání jednotlivých prvků domácnosti za použití OpenHAB (Open Home Automation Bus). V dokumentu je postupně vysvětleno jakou architekturu samotný OpenHAB má a jsou zde popsány jednotlivé prvky této architektury. Druhá část práce se zabývá konfigurací OpenHAB, konkrétně syntaxí jednotlivých souborů a také seznámení s vývojovým prostředím. Třetí část práce se zaměřuje na již přednastavené vazby mezi systémem OpenHAB a periferiemi. Čtvrtá část práce se zabývá přístrojem Raspberry Pi a možností použít jej jako server pro OpenHAB. V závěru práce je uveden návod pro zprovoznění tohoto systému na Raspberry Pi a příklad jeho následné konfigurace za použití prvků a senzorů firmy Homematic.

KLÍČOVÁ SLOVA

OpenHAB, inteligentní domácnost, raspberry pi, vazby, periférie

ABSTRACT

In households is increasing number of electronic peripheries. It is therefore appropriate to control these peripheries remotely, with comfort, for example by smart phone. This thesis focuses on opportunity to control every single element of household by OpenHAB (Open Home Automation Bus). In this thesis is gradually explained which architecture OpenHAB have and it's described individual elements of this architecture. Second part of this thesis deal with configuration of OpenHAB. Specifically with syntax of individual files and acquaint user with development environment. Third part of thesis is focused on already set bindings between system OpenHAB and peripheries. Fourt part focuses on device Raspberry Pi and opportunity to use it as a server for OpenHAB. In conclusion are given instructions for launching this system on Raspberry Pi and a example of following configuration with use of elements and sensors from Homematic.

KEYWORDS

OpenHAB, intelligent home, raspberry pi, bindings, periphery

VESELKA, Jan *Univerzální komunikační centrum pro inteligentní domácnost s využitím OpenHAB*: semestrální projekt. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2015. 56 s. Vedoucí práce byl Ing. Dominik Kováč

PROHLÁŠENÍ

Prohlašuji, že svůj semestrální projekt na téma „Univerzální komunikační centrum pro inteligentní domácnost s využitím OpenHAB“ jsem vypracoval samostatně pod vedením vedoucího semestrálního projektu a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedeného semestrálního projektu dále prohlašuji, že v souvislosti s vytvořením tohoto semestrálního projektu jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

(podpis autora)

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Dominiku Kováčovi za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno

.....

(podpis autora)

PODĚKOVÁNÍ

Výzkum popsáný v tomto semestrálním projektu byl realizován v laboratořích podpořených z projektu SIX; registrační číslo CZ.1.05/2.1.00/03.0072, operační program Výzkum a vývoj pro inovace.

Brno

.....
(podpis autora)

OBSAH

Úvod	12
1 Inteligentní domácnosti	13
1.1 Insight Home	13
1.2 HAIDY	13
1.3 Savant	14
1.4 Open Home Automation Bus	14
1.5 HomeMatic	14
1.5.1 CCU	15
2 Systém OpenHAB	16
2.1 Architektura OpenHAB	16
2.1.1 Sběrnice událostí	16
2.1.2 Stavové uložště	16
2.1.3 Uživatelské rozhraní	16
2.1.4 OpenHAB Runtime	17
2.1.5 Logování stavů	17
2.1.6 Konzole	18
2.1.7 Vazby	18
2.1.8 Zabezpečení	18
2.1.9 Rest	19
2.1.10 Server-Push	19
2.1.11 Konfigurace	19
2.1.12 Položky	20
2.1.13 Mapy stránek	21
2.1.14 Skripty	24
2.1.15 Pravidla	25
2.1.16 Akce	26
2.1.17 Prostředí OpenHAB Designer	27
2.2 OpenHAB vazby	27
2.2.1 Asterisk	28
2.2.2 Homematic	28
2.2.3 HTTP	29
2.2.4 Sériová komunikace	29
2.2.5 KNX	29
2.2.6 Weather	30

3	Raspberry Pi	33
3.1	Specifika Raspberry Pi	33
4	Realizování demonstrátoru inteligentní domácnosti pomocí RPi a OpenHABu.	34
4.1	Instalace přípravku Raspberry Pi	34
4.1.1	Raspbian	34
4.1.2	Java	35
4.1.3	OpenHAB	37
4.2	Vytvoření demonstrátoru inteligentní domácnosti	38
4.2.1	Konfigurace OpenHABu	39
4.2.2	Vytvoření vzhledu aplikace	41
4.2.3	Aplikace pro smartphone	43
5	Závěr	47
	Literatura	48
	Seznam symbolů, veličin a zkratk	50
	Seznam příloh	51
A	Konfigurace CCU	52
A.1	Použitá zařízení	52

SEZNAM OBRÁZKŮ

2.1	Komunikace na sběrnici událostí [12].	17
2.2	Vykreslení elementu Colorpicker	22
2.3	Rámeček	22
2.4	Element graf	23
2.5	OpenHAB Designer	27
2.6	Základní rozvržení [12]	31
2.7	Úprava vzhledu rozvržení	31
3.1	Raspberry Pi verze B [18]	33
4.1	Nástroj SD formatter	34
4.2	Nabootované GUI Raspbianu	35
4.3	Přihlášení do systému	36
4.4	Grafické rozhraní	36
4.5	Rozbalení JDK (Java Development Kit)	36
4.6	Start Runtime	37
4.7	OpenHAB demoverze [12]	37
4.8	Grafické rozhraní GreenT UI [12] (demoverze)	38
4.9	Zobrazení aplikace pomocí GreenT UI	42
4.10	Zobrazení prvků Homematic	42
4.11	Předpověď počasí za pomoci vazby Weather	42
4.12	Zobrazení podlaží	43
4.13	Zobrazení skupiny P_Obyvak	43
4.14	Aplikace OpenHAB na Google play	44
4.15	Menu aplikace	44
4.16	Nastavení aplikace	44
4.17	Hlavní menu	45
4.18	Koupelna - Demo	45
4.19	Obývací pokoj - Demo	45
4.20	Homematic 1	45
4.21	Homematic 2	45
4.22	Příkazový řádek - vazby Weather, Homematic a ruční test	46
A.1	Napájení CCU	53
A.2	Načítání CCU	53
A.3	USB ovladač a zapojení	54
A.4	Načítání GUI	54
A.5	Zobrazené GUI	54
A.6	Učení zařízení	55
A.7	Učení neaktivní	55

A.8	Učení aktivní	55
A.9	Inbox se zařízeními	55
A.10	Dokončení instalace zařízení	56

SEZNAM TABULEK

2.1	Seznam Podporovaných položek v OpenHAB [12]	20
2.2	Seznam vybraných elementů [12]	21

ÚVOD

Automatizace domácnosti se stává v posledních letech populárnější, vzhledem k pokroku v informačních a komunikačních technologiích. Tak lze velmi jednoduše zvýšit komfort pomocí dálkového ovládání jednotlivých periférií domu, např. vytápění, klimatizace, světel nebo spotřebičů, bez nutnosti fyzické přítomnosti uživatele. Taková automatizace se zaměřuje zejména na zvýšení kontroly, monitorování a administrativy těchto systémů. Propojení je realizováno jak skrze bezdrátové připojení, tak s pomocí klasické kabeláže. Navíc jako velká výhoda se jeví také možnost naprosté kontroly spotřeby energie v celém komplexu, tudíž takováto automatizace vede k celkovému snížení spotřeby energie. Zdánlivě jedinou nevýhodou inteligentní domácnosti je její cena. Ta vzroste přibližně o 20% – 30% původní ceny domácnosti. Další nepříjemností může být přetechnizování domácnosti a neschopnost některých členů domácnosti ovládat jednotlivé periférie. I přes to všechno zájem o takovéto domácnosti stále roste [23].

Cíl této práce je seznámení s vybranými systémy, které poskytují automatizaci domácnosti – popis architektury, konfigurace a instalace systému OpenHAB, přičemž je kladen důraz na teoretický rozbor tohoto systému. Dále je práce zaměřena na systém HomeMatic, spolu s přípravkem Raspberry Pi, který má za úkol demonstrovat možnou automatizaci domácnosti.

Pro praktickou část práce byl pro demonstraci automatizace domácnosti zvolen systém OpenHAB. Jeho velkou výhodou je možnost spojit více technologií automatizace dohromady (KNX, Homematic, aj.). Díky jednoduchému rozšiřování o další prvky a senzory domácí automatizace a dále díky možnosti komunikace se smartphonem se jeví jako ideální řešení tohoto problému. Jako prvky a senzory domácí automatizace byly použity zařízení firmy Homematic, jejichž data bude systém OpenHAB interpretovat.

V závěru bude diskutováno zhodnocení výsledků, kterých bylo dosaženo.

1 INTELIGENTNÍ DOMÁCNOSTI

V dnešní době se používají systémy a komunikační nástroje jako je KNX, Asterisk, Homematic, Sériová komunikace nebo také Bluetooth a nesčetná škála dalších systémů a nástrojů. Takovéto množství systémů a automatizace je náročná na správu, proto se tato práce více zaměřila na jeden konkrétní systém.

Na světovém trhu je nespočet systémů, jako ilustrace byly popsány následující systémy: Insight Home, HAIDY, Savant, HomeMatic, OpenHAB (Open Home Automation Bus).

1.1 Insight Home

Srdcem tohoto systému InHome (Insight Home) je tzv. „centrální systém“, který pomáhá, díky perifériím, automatizovat provoz domácnosti. Ovládá například zabezpečení, regulační prvky, osvětlení a jiné řízení domácnosti nebo také hudbu, video a VoIP hovory. Výhodou InHome systému je v řešení komunikace, využívá několik typů komunikace v systému:

- Wifi a jiná bezdrátová řešení,
- KNX,
- optická síť,
- strukturovaná kabeláž,
- VoIP,
- VPN.

Jako nevýhoda se jeví fakt, že uživatel si nemůže sám navrhnout nebo měnit aplikaci, přidávat/odebírat komponenty. Navíc tento systém není Open Source [8].

1.2 HAIDY

Dalším systémem ve výčtu je systém HAIDY. Tento systém patří k levnějším v uvedeném výčtu. Vše pracuje na podobném principu jako předchozí systém. Nabízí například solární systém (ploché a vakuové solární kolektory) nebo ohřev vody bazénu. V základu nabízí dva produkty, a to:

- HAIDY Home,
- HAIDY Plus.

Produkt HAIDY Home nabízí od řízení vytápění, až po osvětlení nebo zabezpečení. Ovládání je intuitivní, i přes pokročilé funkce, které nabízí.

Produkt HAIDY Plus se jeví jako komerční tah, jelikož nenabízí znatelné rozdíly. Jediným znatelným rozdílem je zde rapidní nárůst ceny [7].

1.3 Savant

Tento systém používá operační systém Apple. Tím pádem je možno použít iPad, iPhone, iPod atp. Systém je promyšlen do větších detailů, ovšem opět zde není možnost vlastních uživatelských úprav aplikace, a také není Open Source. Je zde na výběr z několika možností uživatelských rozhraní, které ovšem nejsou sjednocené, a tudíž je zde velké množství aplikací, které musí uživatel obsluhovat. Co se týče komunikace v tomto systému, je opět identická s předchozími systémy [22].

1.4 Open Home Automation Bus

Tento systém sjednocuje veškeré možnosti automatizace domácnosti do jediné aplikace. Díky základům položených na Javě je možno tuto aplikaci velice jednoduše upravit dle potřeb uživatele. Další výhodou je fakt, že toto řešení je Open Source, a tudíž uživatel ušetří značné množství financí. Je také možné přidávat a odebírat komponenty, a to i za chodu aplikace, díky jádru celého systému – tzv. „Runtime“.

Pomocí vazeb, kterých je nyní kolem 40 předpřipravených, je možné spojit aplikaci OpenHAB téměř se všemi dnes dostupnými komponenty domácí automatizace, a tím docílit všech výhod, které nabízejí zde uvedené systémy automatizace.

Ve výsledku tedy nabízí:

- Modifikovatelnost aplikace,
- jednoduchost,
- možnost přidání/odebrání komponentů,
- sjednocení celé automatizace domácnosti do jedné aplikace,
- finanční nenáročnost.

Ovšem je třeba mít na paměti, že základy programování jsou téměř nutné pro jakoukoliv modifikaci aplikace. A pokud chceme modifikovat věci, jako je například rozlišení grafu, fonty textů, apod., je nutná vyšší znalost programovacího jazyka Java.

Z tohoto pohledu se OpenHAB jeví, jako nejlevnější a zároveň nejefektivnější, pokud uživatel zvládá základy Javy. Pro řešení naší problematiky byl vybrán právě OpenHAB.

1.5 HomeMatic

Je systém, pomocí kterého lze z části automatizovat domácnost. Obsahuje například ovládání světel, vytápění, sledování počasí, zabezpečení a řízení přístupu. Jádrem celého tohoto systému je CCU (Central Control Unit).

1.5.1 CCU

Zajišťuje provoz a monitorování funkcí, stejně tak jako hlášení jejich stavů. Také zajišťuje konfiguraci rozhraní jednotlivých komponentů . Zejména sestavuje spojení mezi Ethernetem a HomeMatic systémem. Díky tomuto spojení má uživatel přístup k celé domácí technice skrze lokální počítač, který je v jeho LAN síti nebo také pomocí internetu vzdáleně. Pro aplikaci popisovanou zde bude použito čidlo teploty a vlhkosti vzduchu, dále magnetické čidlo otevření okna.

Pro správnou komunikaci mezi jednotlivými zařízeními a CCU je třeba provést několik kroků. První krok je konfigurace CCU. Poté, před samotnou komunikací CCU a například teploměrem HomeMatic, je nutné spárovat obě zařízení a dále zařízení nakonfigurovat v GUI ve webovém prohlížeči [2].

2 SYSTÉM OPENHAB

2.1 Architektura OpenHAB

Projekt OpenHAB (Open Home Automation Bus) se zabývá takovouto automatizací domácnosti. OpenHAB je tzv. serverová aplikace, která nemá vysoké nároky na hardware a potřebuje k běhu JVM (Java Virtual Machine). Tudíž může běžet na různé architektuře jako je například 32/64bit systém i ARM (Advanced RISC (Reduced Instruction Set Computing) Machine). Mezi zástupce ARM patří kupříkladu Arduino a Raspberry Pi. Klientská aplikace může běžet jako webové rozhraní nebo jako aplikace pod Androidem či iOS. Lze také sledovat stav serveru na dálku, v takovém případě je možné využít klientské aplikace pro mobilní zařízení. Komunikace probíhá na dvou různých interních kanálech [12]:

- Asynchronní sběrnici událostí.
- Stavové uložště.

2.1.1 Sběrnice událostí

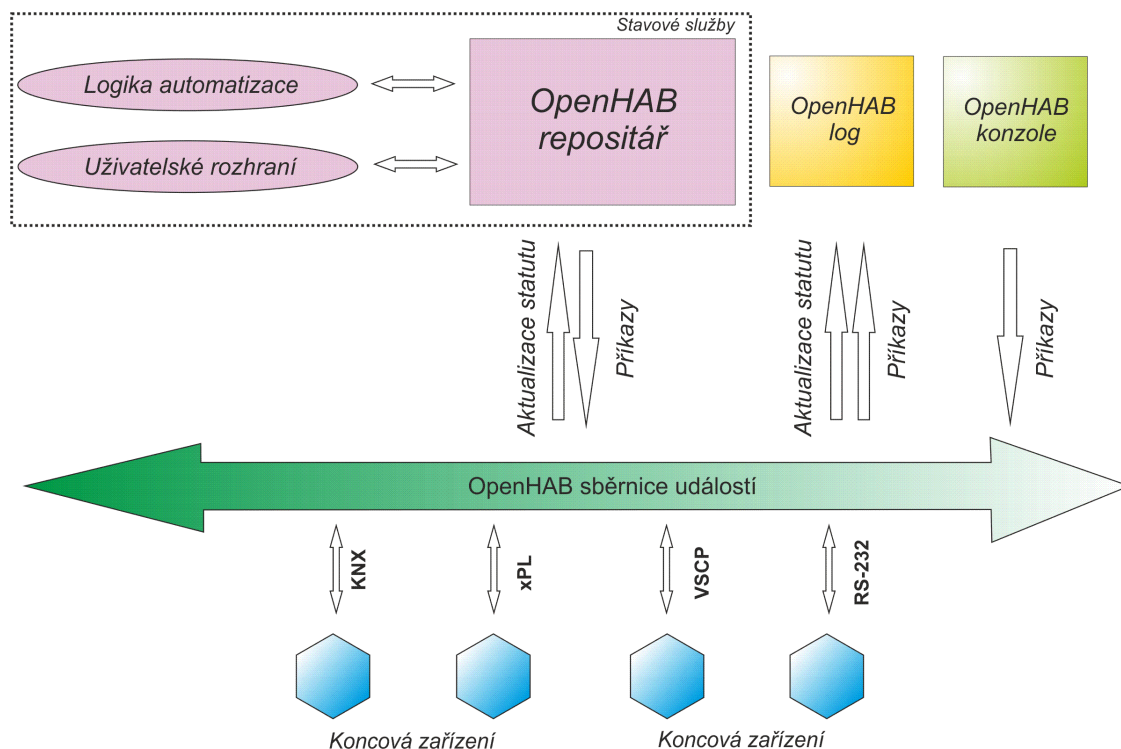
Sběrnice událostí (obr.2.1) je základní služba OpenHAB, která umožňuje komunikaci mezi jednotlivými prvky architektury. Jsou zde hlavně dva typy událostí, a to příkazy, které spouští akce nebo mění stav zařízení/položky, a pak stavové aktualizace, které mají za úkol informovat o změně stavu zařízení/položky. To zajistí velice malou potřebu párování jednotlivých zařízení, tudíž je zajištěna dynamičnost systému [12].

2.1.2 Stavové uložště

Nebo také repozitář (obr.2.1). Je zaveden pro ukládání funkcí, které byly realizovány stavovými prvky. Také tyto informace synchronizuje mezi ostatními balíčky. Je možno uložit jeho obsah do databáze nebo na disk [12].

2.1.3 Uživatelské rozhraní

Standardní uživatelské rozhraní (obr.2.1) pro OpenHAB je založeno na frameworku WebApp.Net. Jedná se o jednoduchý, výkonný framework s využitím technologie AJAX (Asynchronous JavaScript and XML). Obsahuje kompletní sadu k použití, která nám pomůže vytvořit, snadno a rychle, pokročilé mobilní webové aplikace. Tento framework je přístupný skrze jakýkoliv webový prohlížeč, iPhone, iPad, ale také skrze zařízení používající systém Android [12].



Obr. 2.1: Komunikace na sběrnici událostí [12].

2.1.4 OpenHAB Runtime

OpenHAB Runtime je sada OSGi (Open Services Gateway initiative) svazků umístěných na OSGi rámeček (Equinox). Jedná se tedy čistě o JAVA řešení a je nutno spustit JVM pro funkčnost. Základy položené na OSGi umožňují přidávání a odebrání funkcí za běhu bez nutnosti zastavování služby [12].

V prostředí OpenHAB Konzoli, lze poté několika příkazy manipulovat s položkami: [12]

- `openhab send <položka><příkaz>` - pošle příkaz položce,
- `openhab update <položka><stav>` - pošle aktualizaci položce,
- `openhab status <položka>` - zobrazí současný stav statutu položky,
- `openhab items [<vzor>]` - zobrazí jména a typy všech položek, shodujících se ve vzoru,
- `openhab say <věta>` - pošle zprávu skrze TTS (text-to-speech) na hostovací stroj.

2.1.5 Logování stavů

OpenHAB log (obr.2.1) umožňuje zapisovat stavy položek do tzv. log souborů. Lze konfigurovat celou syntax vytvářených souborů, tudíž je zde zajištěna velká flexi-

bilita. Tato konfigurace je realizována v souboru `logging.persist`, přičemž veškeré logy jsou zapisovány do souboru, který se nachází v `{openhab_home}/logs` [12].

2.1.6 Konzole

Konzolí (obr.2.1) je uživateli umožněno, kromě klasického grafického rozhraní, odesílání příkazů a čtení aktuálních stavů položek [12].

2.1.7 Vazby

Vazby (z anglického Bindings) jsou balíčky, které jsou používány pro rozšíření funkcí OpenHAB. Pomocí vazeb lze získat přístup např. ke komunikačnímu softwaru Asterisk nebo je možné připojit se ke sběrnici domácí automatizace KNX (Sběrnice Konnex bus). V tuto chvíli je přes 40 různých vazeb, které rozšiřují jednotlivé funkce OpenHAB [12].

2.1.8 Zabezpečení

Pro zabezpečení komunikace s OpenHAB jsou zde dva mechanismy a to [12]:

- HTTPS.
- Autentizace.

HTTPS

HTTPS (Hypertext Transfer Protocol Secure) je zabezpečovací a zdokonalená verze protokolu HTTP, který chrání například před odposloucháváním nebo falšování dat. HTTPS sice klasicky používá port 443, ovšem OpenHAB využil port 8443, který je také možné použít. Pro šifrování dat protokol SSL (Secure Sockets Layer), tj. vrstva vložená mezi transportní vrstvu (např. TCP/IP) a aplikační vrstvu (např. HTTP) [12].

Autentizace

Všechny požadavky na autentizaci jsou odeslány na přístupový bod JAAS (Java authentication & authorization Service), která zajišťuje povolení autentizovat uživatele. Implicitně je zde soubor `users.cfg`, ve kterém může uživatel vytvořit seznam uživatelů a hesel. Tento soubor poté slouží pro autentizaci uživatelů [12].

Konfigurace zabezpečení

Máme ovšem možnost implementovat pokročilejší konfiguraci, ke které je nutno zahrnout soubor pro JAAS LoginModules (`{openhabs_home}/etc/login.conf`). Lze vybrat z následujících možností zabezpečení [12]:

- ON zabezpečení je povoleno globálně,
- OFF zabezpečení je globálně zakázáno,
- EXTERNAL všechny požadavky, které nepřicházejí z interní sítě, musí být autorizovány (Pro rozeznání interní a externí IP (Internet Protocol) adresy slouží soubor `openhabs.cfg`, ve kterém lze nastavit masku sítě. Každá IP adresa, která je v rozmezí této masky bude označena za interní a nemusí být autorizována).

2.1.9 Rest

Rest API (Representational State Transfer Application Programming Interface) slouží ke komunikaci OpenHAB s ostatními systémy (stavy položek, aktualizace stavů, posílání příkazů). Zajišťuje také přístup k mapám stránek, tzn. že je to rozhraní, které je umožňuje použití vzdáleného uživatelského rozhraní (např. JavaScript klienti). Rest API podporuje různé typy odpovědí, z toho plyne, že lze definovat formát odpovědi po přidání příslušného typu do hlavičky HTTP [12].

2.1.10 Server-Push

Pro funkci Server-Push je použit Atmosphere framework. Tímto je umožněno použít long-polling (dlouhé dotazování), HTTP streamování nebo websocket, podle toho co podporuje klient. Pokud využijeme pro každého klienta unikátní ID, je zajištěné snížení využití sítě. taktéž probíhá automatické porovnávání zpráv - pokud je zpráva aktuální identická se zprávou předcházející, tuto zprávu server ani neodesílá [12].

2.1.11 Konfigurace

Konfigurace je realizována pouze textovými soubory, které je možno editovat v mnoha různých textových editorech, ovšem pro náročnější konfigurace je příhodnější použít aplikaci, která je nad vývojovým prostředím Eclipse, a to OpenHAB Designer. Konfigurace je rozdělena do několika souborů a tedy i skupin, a to [12]:

- Položky (Items).
- Mapy Stránek (Sitemaps).
- Skripty (Scripts).
- Pravidla (Rules).

- Akce (Actions).

2.1.12 Položky

Položky jsou objekty, které mohou být přečteny nebo do nich můžou být hodnoty zapsány, za cílem interakce s nimi. Také mohou být svázány vazbou (bindings), např. pro čtení statutu např. KNX (Konnex Bus). Mohou být definovány v souborech, které se nachází v `{openhab_home}/configurations/items`. Příponu `.items` musí mít všechny soubory definice položky. Položky jsou typicky definovány pomocí OpenHAB Designer úpravou definičního souboru položky. Pokud tak učiníte, budete moci využívat plnou podporu vývojového prostředí, jako je například kontrola syntaxe [12].

Název položky	Popis	Typy příkazů
Color	Informace barvy (RGB)	OnOff, Percent, HSB ¹
Contact	Status položky (kontakt dveří/oken)	-
DateTime	Obsahuje datum a čas	-
Dimmer	Obsahuje hodnotu stmívače v %	OnOff, Percent
Group	Sdružuje položky do jedné skupiny	-
Number	Ukládá hodnoty ve formátu čísla	Decimal
Rollershutter	Typicky používané pro žaluzie	UpDown, Percent
String	Text	String
Switch	Přepínač světél	OnOff

Tab. 2.1: Seznam Podporovaných položek v OpenHAB [12]

Syntaxe položek

Položky jsou definovány následovně (Část v hranatých závorkách není povinná) [12]:

```
Typ Jméno["popisek"] [<Ikona>] [(skupina1,...)] [{konfigurace}]
```

- Typ - označuje druh položky. Jejich výčet se nachází v tabulce č.1.
- Jméno - je název položky, který využijeme pro volání této položky.
- Popisek - je text, který bude zobrazen v uživatelském rozhraní (pokud se jedná o typ, který pracuje s čísly může být použit pro formátování výstupu).
- Ikona - použije se pro přiřazení ikony, lze použít i vlastní. Ikona odkazuje na soubor obrázku, který se nachází v `{openhab_home}/webapps/images/`.

¹HSB (Hue (tón), Saturation (sytylost), Brightness (jas)) je barevný model, který nejlépe odpovídá vnímání barev lidským okem.

- Skupina - Položku můžeme umístit do více skupin a tím zamezit opakovanému definování položky v jednotlivých skupinách.
- Konfigurace - zde lze nastavit vazby jako je např. typ spojení, adresa hardware, atp.

2.1.13 Mapy stránek

Mapy stránek se používají k vytvoření elementu uživatelského rozhraní, aby OpenHAB položky byly dostupné z různých nadstaveb. V podstatě se jedná o deklarativní definice uživatelského rozhraní. Krátkým skriptem lze definovat strukturu a obsah obrazovky uživatelského rozhraní. Soubory jsou uloženy v `{openhbab_home}/configurations/sitemaps` [12].

Syntaxe map stránek

Mapy stránek mohou být složeny seskupením různých prvků uživatelského rozhraní do oblastí, které budou poskytnuté OpenHAB. V souboru definic mohou být použity následující elementy (tab. 2.2) [12]:

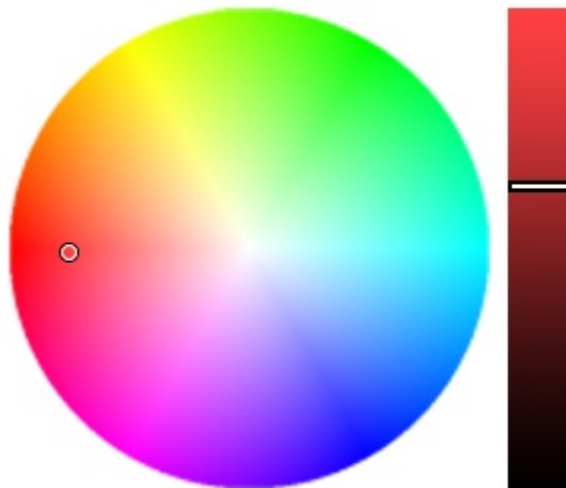
Jméno elementu	Popis
Frame	Oblast s různými elementy / dalším vnořeným rámečkem
Group	Vykreslí prvky dané skupiny definované v souboru definice
Colorpicker	Vykreslí dialog pro výběr barvy
Chart	Vykreslí graf
Image	Vykreslí obrázek
Selection	Vykreslí položku výběru
Slider	Vykreslí posuvník
Switch	Vykreslí přepínač
Video	Vykreslí vybrané video

Tab. 2.2: Seznam vybraných elementů [12]

Dialog pro výběr barvy

Tento element vykreslí dialog pro výběr barvy (obr. 2.2). Atribut `sendFrequency` určuje námi zvolenou frekvenci obnovy tohoto elementu (reálné číslo neboli `int`). Tento element v současné době není podporován v GUI GreenT UI (více o GreenT UI na 4.1.3).

```
Colorpicker[item="Nazev"][label="Popis"]
[icon="ikona"][sendFrequency="čas_obnovy"].
```

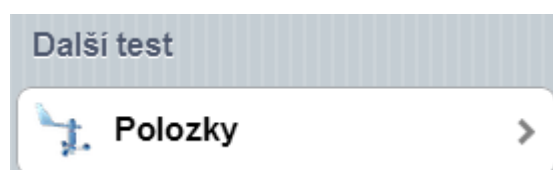


Obr. 2.2: Vykreslení elementu Colorpicker

Rámeček

Dále zde máme element **Frame** (obr. 2.3), který má za úkol vytvořit oblast ve které mohou být další elementy, popřípadě vnořený rámeček. Vpodstatě se jedná o oblast, která nám vizuálně odděluje prostor, ve kterém mohou být položky, které zde umístíme.

```
Frame[item="Název"] [label="Popis"] [icon="Ikona"] {
  [Další_elementy_mapy_sítě].
}
```



Obr. 2.3: Rámeček

Skupina

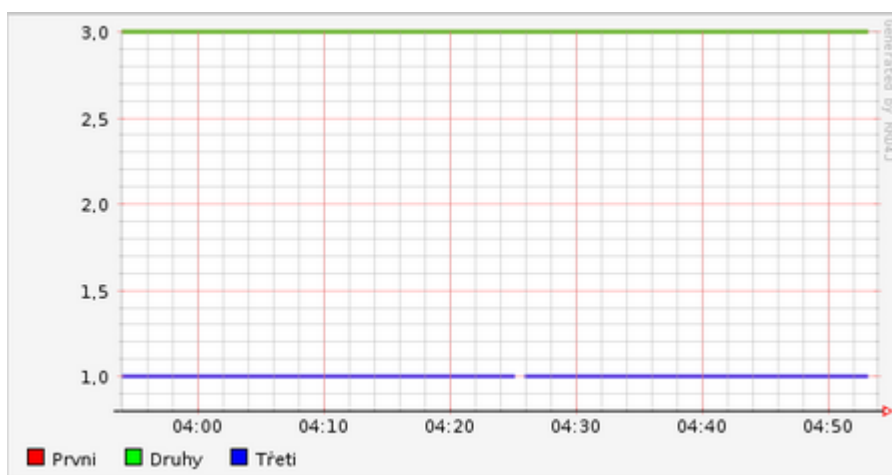
Element **Group** vykreslí prvky dané skupiny, kterou jsme definovali v definici položek.

```
Group[item="Název"] [label="Popis"] [icon="Ikona"] .
```

Graf

Má za úkol vykreslit graf (obr.2.4). Máme možnost nastavit osu x (period=m/h/d), ale také rychlost obnovování informací (refresh="milisekundy").

```
Chart[item="Název"] [label="Popis"] [icon="Ikona"] [period=m/h/d]
[refresh=ms].
```



Obr. 2.4: Element graf

Obrázek

Vykreslí obrázek dle nastavení. `Url=` určuje cestu k obrázku a `refresh=ms` nám udává čas v milisekundách za který se nám obrázek opět načte.

```
Image[item="Název"] [label="Popis"] [icon="Ikona"] url="url_obrázku"
[refresh=ms].
```

Výběr

Vykreslí nám položku výběru, tzn. že je zde možnost výběru z několika možností, kdy pouze jedna může být aktivována. Jednotlivé výběry pojmenujeme v definici výběru (např. 1=Večeře, 2=Návštěva, 0=Vypnuto).

```
Selection[item="Název"] [label="Popis"] [icon="Ikona"]
[mappings="Definice_výběrů"].
```

Přepínač

Má stejnou syntaxi jako element výběru.

Posuvník

Vykreslí posuvník s možností vypnout/zapnout.

```
Slider[item="Název"][label="Popis"][icon="Ikona"]  
[sendFrequency="frekvence"][switchEnabled="ON/OFF"].
```

Video

Zde je nutno také nastavit url videa.

```
Video[item="Název"][icon="Ikona"] url="url_připojovaného_video".
```

2.1.14 Skripty

OpenHAB přichází s velmi silným výrazovým jazykem, který může být použit pro definování skriptů. Skript je blok kódu, který může být definován uživatelem, který může být volán, a tudíž používán z různých míst kódu. Veškeré skripty jsou uloženy v souboru, který se nachází v `openhab.home/configurations/scripts`. Název souboru označuje název skriptu (bez přípony). Je zde použit stejný výrazový jazyk jako v Xtend odnoži Javy, přičemž syntaxe je velmi podobná programovacímu jazyku Java a má mnoho funkcí, které umožňují psát kódy stručně. Navíc není třeba skripty kompilovat, takže je lze interpretovat za běhu aplikace. Aby bylo možné se skripty dělat něco užitečného, OpenHAB poskytuje přístup k:

- Všem definovaným položkám (zajištěn přístup dle názvu)
- Všechny výčet stavů/příkazů (ON, OFF, Increase, atp.)
- Veškeré standardní akce (aby bylo zajištěno „že se něco stane“).

Je zde také podpora prostředí (OpenHAB Designer), který slouží mimo jiné i pro opravy syntaxe, zvýraznění, atp. Při kombinaci těchto funkcí, lze jednoduše napsat:

```
if(Teplota.state<18){  
sendCommand(Topení,ON).
```

Kód uvedený níže sleduje, zda-li je stav teploty menší než 18°C. Pokud nižší je, tak se pošle příkaz ON na položku Topení [12].

Volání skriptů

Skript je identifikován svým názvem souboru, pokud název souboru je `demo.script`, název skriptu je pouze `demo` (bez přípony). Každý skript má návratovou hodnotu, která je výsledkem posledního výrazu v něm (může být NULL). Skripty lze volat z různých míst:

- Z rámce pravidel prostřednictvím: `callScript("názevSkriptu")` akce

- Ze záznamu v kalendáři ve vašem Google kalendáři - stačí napsat `callScript("názevSkriptu")`, v položce kalendáře.
- Z XMPP konzole zadáním `>callScript("názevSkriptu")` (do konzole lze zadat také např. `>Teplota.state`, což je příkaz, kterým vypíšeme hodnotu v položce `Teplota`) [12].

2.1.15 Pravidla

Pro vyvolání reakce na událost slouží konfigurační soubor pravidel. Pravidla jsou zapsána v souboru `{openhab.home}/configurations/rules`. Soubor s pravidly je textový soubor s následující strukturou [12]:

- Import.
- Deklarace proměnných.
- Pravidla.

Import

Zápis je uskutečněn klíčovým slovem `import #názevKnihovny`. Syntaxe je téměř stejná jako v jazyce Java [12].

Deklarace proměnných

Tato sekce může být použita pro deklarování proměnných, které by měly být dostupné všem pravidlům v tomto souboru. Můžeme deklarovat proměnné s nebo bez počátečních hodnot, modifikovatelné (klíčové slovo `var`) nebo pouze pro čtení (`val`) [27].

Spouštěcí podmínky

Každé pravidlo může mít jakýkoliv počet spouštěcích podmínek, ale musí mít alespoň jedno. Prováděcí blok obsahuje kód (skript), který bude proveden, když spouštěcí podmínka bude splněna. Jsou zde tři druhy spouštěčů pravidel:

- Spouštěče založeny na události položky. Můžeme odposlouchávat příkazy pro konkrétní položku: aktualizaci stavu nebo změny stavu (aktualizace může stav ponechat beze změn). Je taktéž možné zvolit, zda bude zachycen konkrétní nebo jakýkoliv příkaz / stav.
- Časové spouště. Můžeme použít některé předdefinované výrazy pro časovače nebo použít expresi Cron [13].
- Systémové spouště. Vykonání akce např. při zapnutí/vypínání systému [12].

Syntaxe

```
rule "Pravidlo"
when
    <SPOUŠTĚCÍ_PODMÍNKA1> or
    <SPOUŠTĚCÍ_PODMÍNKA2> or
    <SPOUŠTĚCÍ_PODMÍNKA3>
    ...
then
    <PROVÁDĚCÍ_BLOK>
end
```

Toto uvedené pravidlo znázorňuje možnou syntaxi kódu. "Pravidlo" je název pravidla a pokud je splněna (v našem případě alespoň jedna) z nadefinovaných spouštěcích podmínek, provede se část kódu s prováděcím blokem, tj. odešle se například příkaz pro upravení hodnot nebo aktualizaci položky a mnoho dalšího.

2.1.16 Akce

Akce jsou předdefinované Java metody, které jsou automaticky staticky importované a mohou být použity v rámci skriptů a pravidel k provádění OpenHAB konkrétních operací. Tyto akce lze rozdělit na dvě skupiny, a to [12]:

- Akce jádra
- Akce položek.

Akce jádra

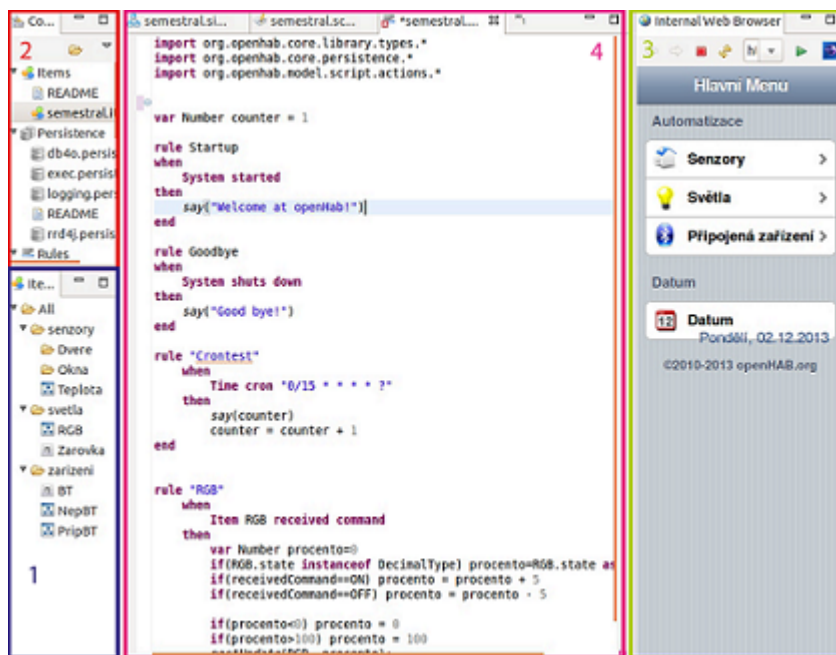
Je zde množství akcí, jako například příkaz položce skrz sběrnici událostí nebo také poslání položce aktualizaci (např. `postUpdate (Položka, Aktualizace)`). Patří zde i log událostí nebo akce související se zvukem a jeho manipulací (převod textu na řeč, atp.) [12].

Akce položek

Je nám taktéž umožněno, abychom skrze vazby prováděli jednotlivé akce. Každá vazba má svůj vlastní výčet akcí, které může provádět. Například se připojí k Twitteru a pošle zprávu: `sendTweet (Zpráva)` [12].

2.1.17 Prostředí OpenHAB Designer

Toto prostředí (obr. 2.5) zjednodušuje a zefektivňuje konfiguraci souborů nutných pro běh OpenHAB. Jedná se o Eclipse RCP (Rich Client Platform) aplikaci, což znamená, že je zde minimální potřeba modulů pro výstavbu klientské aplikace [21]. Jsou zde čtyři okna s:



Obr. 2.5: OpenHAB Designer

1. položkami,
2. konfiguračními soubory,
3. vestavěným webovým prohlížečem,
4. pracovním oknem.

Tento nástroj zvyšuje přehlednost výsledného kódu díky zvýrazňování syntaxe. Lze také zobrazit výsledné uživatelské rozhraní ve vestavěném prohlížeči. Syntaxe lze klávesovou zkratkou CTRL+mezera doplnit díky našeptávači.

2.2 OpenHAB vazby

Jak již bylo zmíněno, vazby umožňují propojit OpenHAB systém s perifériemi. V současné době jich je přibližně 40. Tudíž je možné propojit již automatizovanou domácnost k systému OpenHAB – například zabezpečení nebo některé jiné systémy automatizace domácnosti.

2.2.1 Asterisk

Tato vazba se používá k tomu, aby mohla probíhat komunikace mezi OpenHAB a opensource (otevřený software [11]) Asterisk implementovanou telefonní ústřednou PBX (Private Branch eXchange). S touto vazbou lze detekovat příchozí hovory nebo zjistit, zda-li někdo momentálně neuskutečňuje telefonní hovor. Lze také implementovat další vazby, a tak například zobrazit ID volajícího na TV pomocí vazby Samsung TV. Samozřejmě musíme nakonfigurovat vazbu. V souboru `.items` je možné tuto konfiguraci provést, a to velice jednoduchým způsobem – stačí pouze přidat k dané položce `{asterisk=active}`. Tato vazba lze použít pro položky typu Přepínač a Text [12].

2.2.2 Homematic

Tato vazba podporuje nesčetné množství zařízení od firmy Homematic. Základ těchto zařízení spočívá v tom, že je zde jedna hlavní řídicí jednotka (například CCU1, CCU2, LAN adaptér a CUL), která propojuje ostatní zařízení skrze bezdrátový protokol HomeMatic. Pro aplikaci, kterou se zabývá tato práce bude použito CCU1. Konfigurace se provádí v souboru `openhab.cfg` následovně [12]:

- `homematic:host` – Hostname / IP adresa Řídicí jednotky
- `homematic:callback.host` – Nepovinné. IP adresa / Hostname lokálního hostitele (ale ne `localhost` nebo `127.0.0.1`) Callback (Zpětné volání) je funkce při níž server pro vzdálený přístup po ověření odpojí připojení a poté zavolá počítač uživatele zpět [1].
- `homematic:callback.port` – Nepovinné. Číslo portu pro server zpětného volání. Základní nastavení je 9123 [12].

Příklad konfigurace souboru `openhab.cfg`:

```
homematic:host=Moje_zakladna
homematic:callback.host=Muj_Host
homematic:callback.port=9123.
```

Příklad konfigurace položek:

```
Number Teplota "Momentální Teplota [%.1f °C]" <temp> (pObyvak,pPocasi)
{homematic="IEQ0053616:1#TEMPERATURE"}
Number Vlhkost "Vlhkost [%d %%]" <waterdrop> (pObyvak,pPocasi)
{homematic="IEQ0053616:1#HUMIDITY"}
Textitem=Teplota
Textitem=Vlhkost
```

Tato konfigurace nastaví čísla s názvem „Teplota“ a „Vlhkost“ s popisem „Momentální teplota“ a „Vlhkost“. Dále je přiřazeno ID daného zařízení, pro sestavení spojení mezi systémy OpenHAB a HomeMatic [12].

2.2.3 HTTP

Tato vazba umožňuje, aby OpenHAB mohl volat URL, když nastane nějaká podmínka. Je nutné umístit tento svazek HTTP vazby do {OpenHAB_home}/addons složky a dále nakonfigurovat informace o vazbě do konfigurace položek. Výsledná konfigurace může vypadat takto [12]:

```
Number Weather_Temperature "Temperature [%..1f-°C]" <temperature>
(Wetter){http="<[http://weather.yahooapis.com/forecastrss?
w=638242&u=c:60000:XSLT(demo_yahoo_weather.xsl)]">{.
```

Lze také použít vypisování z cache paměti a to následovně v konfiguraci [12]:

```
http:weatherCache.url=http://weather.yahooapis.com/forecastrss?
w=566473&u=c
http:weatherCache.updateInterval=60000.
```

A následovně v nastavení položek [12]:

```
Number temp{http="<[weatherCache:10000:XSLT
(demo_yahoo_weather_temperature.xsl)]">{.
Number windSpd{http="<[weatherCache:10000:XSLT
(demo_yahoo_weather_wind_speed.xsl)]">{.
```

2.2.4 Sériová komunikace

K tomu abychom mohli připojit skrze vazbu Sériové zařízení, je třeba provést konfiguraci nastavení a to v souboru položek. Syntaxe je velice jednoduchá – stačí za položku přidat `serial="port"`. Kde port identifikuje sériový port hostovacího systému, například: "COM1" na operačním systému Windows, "/dev/ttyS0" na operačním systému Linux nebo "dev/tty.PL2303-0000103D" na systému Mac OS [12].

```
Switch Tlacitko_HW "Dvere" (Vstup) {serial="/dev/ttyS0"} [12].
```

2.2.5 KNX

KNX je standardizovaná, síťová komunikace (V Evropě známé také jako EIB – European Instalation Bus). Používá se v inteligentních budovách, jelikož jeho možnosti zahrnují například kontrolu světla, systém topení/ventilací, bednění a žaluzií, monitorování alarmů, správu energie nebo distribuci audio/video. Výhodami je [10][4]:

- jednoduchá instalace,
- jednoduché a málo nákladné změny/rozšíření,
- decentralizovaný systém, takže funguje i při výpadku,
- úspora energie a nákladů.

Tato vazba umožňuje připojit k OpenHAB také KNX instalace domácí automatizace. Je potřeba nastavit konfiguraci v souboru `openhab.cfg` a dále nakonfigurovat informace v souboru položek. Syntaxe je vysvětlena níže [12]:

```
knx=" [<] [<IDdb>:] <hlAS> [[+ [<] <OdposlechAS>] + [<] <OdposlechAS> ..] ,
[<] [<IDdb>:] <hlAS> [[+ [<] <OdposlechAS>] + [<] <OdposlechAS> ..] ".
```

Části v hranatých závorkách jsou nepovinné. Každý úsek oddělený čarkou odpovídá KNX datovými body. Většinou bývá definovaný pouze jeden datový bod (`IDdb`) pro přijetí příkazu na OpenHAB položku. Pro každý typ položky je možno použít pouze určené typy příkazů (tab. 2.1). Lze přiřadit jednu skupinu KNX adresy (`hlAS`) a několik naslouchacích adres skupin (`OdposlechAS`) pro každý typ příkazu. Volitelný znak `<` říká, zdali adresa skupiny daného datového bodu přijíma požadavek pro čtení na KNX sběrnici (pokud ano, znak zde je). Příklad by tedy vypadal takto [12]:

```
Stmívač TestStmívače (Světla) {knx="1/3/20+0/3/20, 1/3/21, 1/3/22+
0/3/22+0/8/15"}.
```

2.2.6 Weather

Vazba s názvem „Weather“ umožňuje získávání dat, která se týkají předpovědi počasí. K disozici je několik poskytovatelů [26]:

- ForecastIo,
- OpenWeatherMap,
- Wunderground,
- WorldWeatherOnline,
- Hamweather,
- Yahoo.

Většina poskytovatelů požaduje takzvaný „API klíč“ (Application Programming Interface key). Vyjímkou je poskytovatel Yahoo, který tento klíč nevyžaduje. Tuto vazbu je nutno nakonfigurovat v souboru `openhab.cfg` a v souboru položek. Konfigurační soubor je potřeba v následujících řádcích v úseku vazby „Weather“, nastavit například takto [26]:

```
weather:apikey.ForecastIo=52f5713fb0365f42d9aa6e3f2f902f95
weather:location.doma.latitude=49.10
```

```

weather:location.doma.longitude=16.35
weather:location.doma.provider=ForecastIo
weather:location.doma.language=en
weather:location.doma.updateInterval=4

```

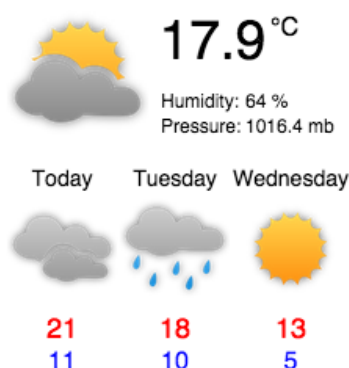
Do řádku „apikey.ForecastIo“ se zadává API klíč. Část „location.doma“, ve většině řádků, udává název lokace. Řádky „latitude“ a „longitude“ uvádějí zeměpisnou výšku a šířku místa, pro které nastavujeme předpověď. Položka „provider“ určuje poskytovatele předpovědi, „language“ nastavuje jazyk předpovědi. Nakonec „updateInterval“ určuje interval obnovování dat. Této hodnotě je potřeba věnovat pozornost, protože někteří poskytovatelé, jako například „ForecastIo“, mají omezený počet odeslání dat za den. Nastavení v tomto příkladu je provedeno pro oblast Brno, poskytovatel ForecastIo, jazyk angličtina, interval obnovení 4minuty.

Syntaxe v souboru položek je vysvětlena níže. Kompletní výpis možností nastavení předpovědi je uveden na [26]:

```

Number Teplota_Dnes "Teplota [%.1f°C]" <temperature>
{weather="locationId=doma,type=temperature,property=current"}

```



Obr. 2.6: Základní rozvržení [12]



Obr. 2.7: Úprava vzhledu rozvržení

Je možné použít tzv. vylepšené „rozvržení HTML“ (z anglického HTML layouts) pro tuto vazbu. Pro to je nutné stáhnout z oficiálních stránek OpenHAB [12] složku `weather-data`, která obsahuje několik typů obrázků a příklad rozložení HTML pro tuto vazbu. Složka `layout` obsahuje soubory `example.css`, `example.html` a `example.js`. Názvy souborů musí být jednotné, avšak je možné názvy změnit (např. z „example“ na „pocasi“). V mapách stránek je poté možné zadat:

```
Webview url="/weather?locationId=home&layout=pocasi&iconset=colorful"
```

Zde poté `locationID=` uvádí název lokace, která byla zvolena v konfiguračním souboru `openhab.cfg`, dále `layout=` uvádí zvolený jednotný název souborů ve složce `layout` a `iconset=` uvádí jaký typ obrázku bude zvolen (na výběr je z: `colorful`, `dark`, `flat_black`, `flat_colorful`, `flat_white` a `light`). Výsledná demo verze je zobrazena na obrázku (obr. 2.6).

Konfigurační soubory lze možné editovat a získat odlišný vzhled, dle vlastních potřeb a přání. Uživatel je zde omezen pouze svou fantazií a znalostí syntaxe HTML a CSS (obr. 2.7).

3 RASPBERRY PI

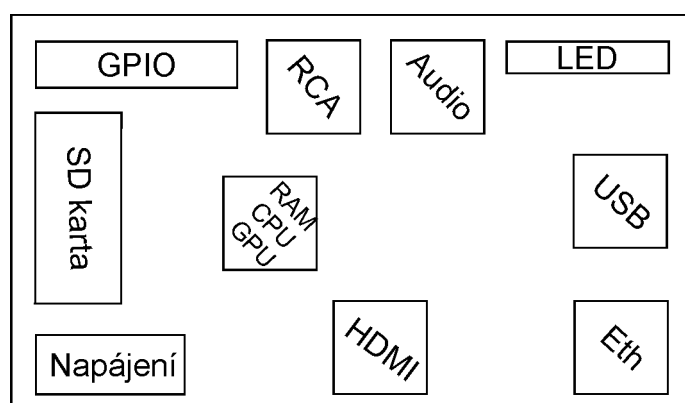
Raspberry Pi (obr. 3.1) je jednodeskový počítač, který svou velikostí nepřevyšuje velikost kreditní karty. Vyvíjí jej nadace Raspberry Pi Foundation [14] za účelem podpory výuky informatiky na školách. Raspberry Pi nalézá své využití v mnoha ohledech. Jako příklad lze uvést realizaci nenáročných aplikací [6]:

- automatizace domácnosti,
- quadrokoptéra,
- multimediální centrum pro XBMC (Xbox Media Center),
- meteostanice.

Kromě tohoto stručného výčtu možností použití je zde samozřejmě mnoho jiných. Díky minimálním rozměrům a poměru cenu/výkon se představivosti opravdu meze nekladou [17].

3.1 Specifika Raspberry Pi

Nyní jsou dostupné dvě verze, a to verze A a verze B, přičemž dále budeme mluvit o verzi B. Raspberry Pi má jednojádrový procesor architektury ARM (ARM11 Broadcom BCM2835), taktovaný na frekvenci 0,7GHz a paměť RAM o velikosti 512MB. Výbava obsahuje dva porty USB 2.0, HDMI (High-Definition Multimedia Interface) a RCA pro video výstup, a pro audio výstup zde máme 3,5mm jack. Slot pro SD kartu. Piny GPIO (General Purpose Input/Output) a jako velká výhoda verze B oproti verzi A je síťové rozhraní 10/100 Ethernet [6][16][19].



Obr. 3.1: Raspberry Pi verze B [18]

4 REALIZOVÁNÍ DEMOSTRÁTORU INTELGENTNÍ DOMÁCNOSTI POMOCÍ RPI A OPENHABU.

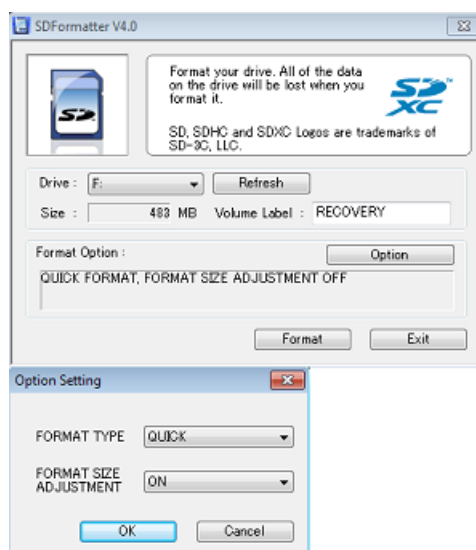
Zde bude popsána část instalace systému Raspbian na RPi, další část instalace a konfigurace OpenHAB a vytvoření demonstrátoru. V příloze je poté podrobně rozebrána konfigurace CCU firmy Homematic A.

4.1 Instalace přípravku Raspberry Pi

Je zde možnost zprovoznit OpenHAB na Raspberry Pi. Raspberry Pi nám dovoluje vybrat si z několika různých operačních systémů, na kterých lze zprovoznit JVM (Linux/Windows). Ovšem pro řešení zadané problematiky se na základě doporučení z [12] zvolilo Raspbian.

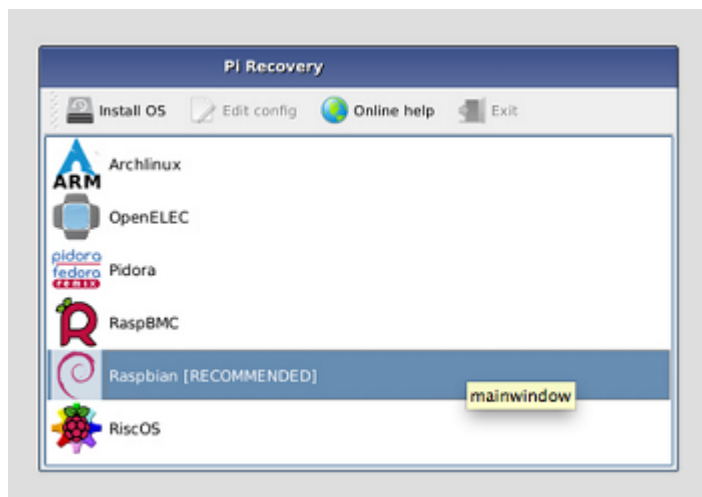
4.1.1 Raspbian

Jedná se o operační systém, který je zdarma, založený na operačním systému Debian a optimalizovaný pro Raspberry Pi. Tento operační systém je souhrn několika základních programů a služeb pro chod Raspberry Pi. Raspbian obsahuje ale také více než 35000 balíčků softwaru, přepraveného k instalaci na Raspberry Pi [20]. Tento operační systém je nutno nainstalovat na SD kartu, která se vkládá do Raspberry Pi (obr.3.1). Nejdříve je však nutno SD kartu naformátovat, to provedeme pomocí



Obr. 4.1: Nástroj SD formatter

nástroje SDFormatter [5] (obr. 4.1). V nastavení tohoto nástroje je třeba zvolit „Format size adjustment - ON“, touto volbou využijeme plnou kapacitu SD karty. Poté můžeme zkopírovat obraz OS (v našem případě Raspbian z [15]) na SD kartu, karta se umístí do Raspberry Pi a nabootuje se zvolený systém. V nabídnutém GUI se tento systém nainstaluje (obr 4.2).



Obr. 4.2: Nabootované GUI Raspbianu

GUI Raspbianu

Ovládání formátovacího nástroje a instalace systému je intuitivní a zvládne ji i mírně pokročilý uživatel. Po instalaci systému je v základu nastaveno přihlašovací jméno a heslo následovně (obr. 4.3):

- jméno - „raspberry“,
- heslo - „pi“.

Poté lze zadat příkaz pro start uživatelského rozhraní - **startx**. Uživatelské rozhraní je velice podobné uživatelskému rozhraní Windows, Ubuntu aj. (obr. 4.4).

4.1.2 Java

Nyní je nutné nainstalovat Javu. Javu lze zdarma stáhnout z [9]. Následující příkazy provádíme v příkazovém řádku/konzoli. Rozbalíme stažený soubor, to provedeme příkazem **sudo tar -zxvf název_souboru -C /opt** (obr. 4.5). Nyní je Raspberry Pi připraveno pro implementaci softwaru OpenHAB.

```

Debian GNU/Linux wheezy/sid raspberrypi tty4
raspberrypi login: pi
Password:
Last login: Mon Dec 30 19:54:15 UTC 2013 on tty1
Linux raspberrypi 3.1.9+ #2 Mon Apr 16 04:53:15 EST 2012 armv6l

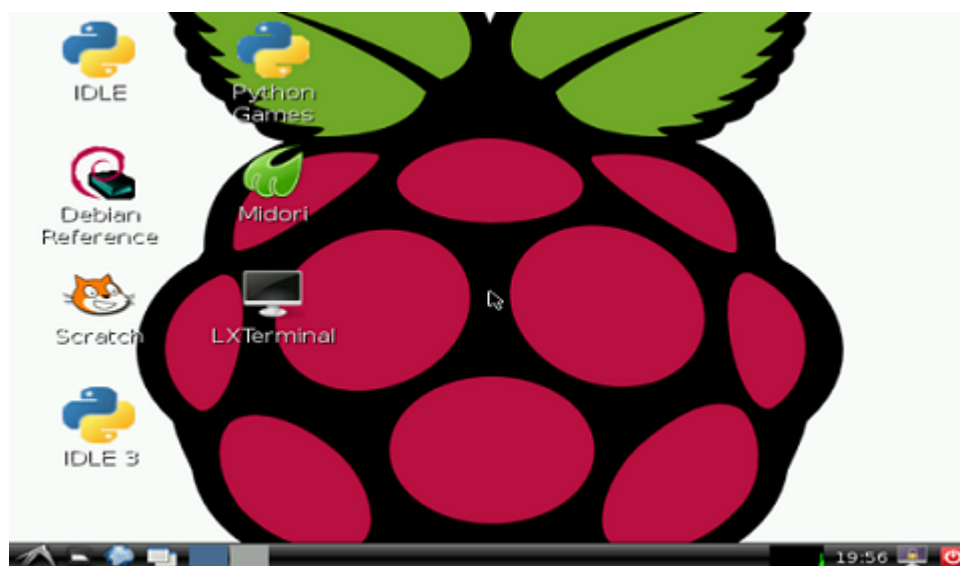
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

Type 'startx' to launch a graphical session
pi@raspberrypi ~ $ startx_

```

Obr. 4.3: Přihlášení do systému



Obr. 4.4: Grafické rozhraní

```

xvesel61@xvesel61-VirtualBox:~$ ls -l
celkem 56
drwxr-xr-x 2 xvesel61 xvesel61 4096 lis  6 23:58 Dokumenty
-rw-r--r-- 1 xvesel61 xvesel61 8980 lis  6 23:47 examples.desktop
drwxr-xr-x 2 xvesel61 xvesel61 4096 lis  6 23:58 Hudba
drwxr-xr-x 2 xvesel61 xvesel61 4096 lis 11 13:08 News
drwxr-xr-x 2 xvesel61 xvesel61 4096 lis  6 23:58 Obrázky
drwxr-xr-x 3 xvesel61 xvesel61 4096 pro 11 18:38 Plocha
drwxr-xr-x 2 xvesel61 xvesel61 4096 pro 12 00:27 Stažené
drwxr-xr-x 2 xvesel61 xvesel61 4096 lis  6 23:58 Šablony
drwxrwxr-x 2 xvesel61 xvesel61 4096 lis 25 21:56 Ubuntu One
drwxr-xr-x 2 xvesel61 xvesel61 4096 lis  6 23:58 Veřejné
drwxr-xr-x 2 xvesel61 xvesel61 4096 lis  6 23:58 Videa
drwxr-xr-x 3 xvesel61 xvesel61 4096 lis 11 20:00 workspace
xvesel61@xvesel61-VirtualBox:~$ cd Stažené
xvesel61@xvesel61-VirtualBox:~/Stažené$ sudo tar -zxvf jdk-8-linux-arm.tar.gz -C /opt

```

Obr. 4.5: Rozbalení JDK (Java Development Kit)

4.1.3 OpenHAB

Celý OpenHAB je, stejně jako Java, zdarma. Pro běh OpenHAB je nutné spustit Runtime, který lze stáhnout z [24] stejně jako veškeré vazby. Také se musí vytvořit uživatelský konfigurační soubor. Stávající soubor `openhab_default.cfg` se zkopíruje do adresáře, ve kterém se nachází zmíněný soubor a pojmenuje se `openhab.cfg`. Pro rychlý start je nutno stáhnout DEMO z [12] a zkopírovat soubory do patřičných složek. Poté lze Runtime spustit příkazem `sh start.sh` (obr.4.6).

```
xvesel61@xvesel61-VirtualBox:~/Plocha/RunTime$ sh start.sh
Launching the openHAB runtime...
```

Obr. 4.6: Start Runtime

Jakmile se Runtime spustí, potrvá načtení veškerých souborů pár chvil a poté je možné do webového prohlížeče zadat následující adresu pro zobrazení klasického GUI: `http://localhost:8080/openhab.app?sitemap=název_mapy_stránek`. Výsledné zobrazení lze vidět na obr.4.7. Je také možnost využít rozhraní GreenT UI.



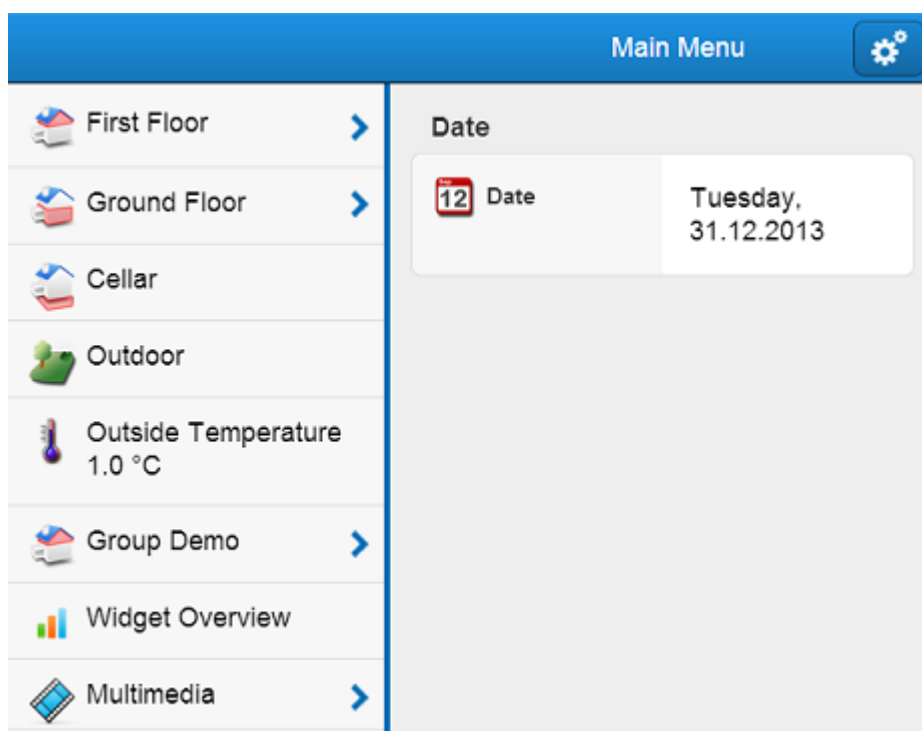
Obr. 4.7: OpenHAB demoverze [12]

GreenT UI

Toto rozhraní bylo vyvinuto s Sencha 2.0 JavaScript frameworkem. Komunikace je zajištěna pomocí REST API. Také komunikuje podobně jako klasické rozhraní s

Runtimem, pouze s tím rozdílem, že nabízí více možností nastavení a to od graficky příjemnějších prostředí (obr. 4.8, tak po podporu zabezpečení nebo zahrnutí více jazyků. Lze nastavit zda-li nebudou probíhat animace (pro méně výkonné stroje). Pro realizaci tohoto rozhraní je nutno [12]:

1. stáhnout z [12] GreenT UI,
2. rozbalit jej do adresáře `webapps`, nachází se v kořenovém adresáři Runtimu,
3. adresář `greent.items` umístit do adresáře `configurations/items`,
4. přidat vazbu OpenHAB Exec mezi ostatní vazby (lze stáhnout z [12]),
5. do webového prohlížeče zadat `http://localhost:8080/greent` (může se zadat port 8443 a protokol HTTPS pro HTTPS kódování).



Obr. 4.8: Grafické rozhraní GreenT UI [12] (demoverze)

4.2 Vytvoření demonstrátoru inteligentní domácnosti

K vytváření kódu byl použit nástroj OpenHAB Designer (kap. 2.1.17). Bylo třeba vytvořit kód v souborech s příponami `items` (položky), `sitemap` (mapy stránek), `rules` (pravidla), `persist` (perzistence).

4.2.1 Konfigurace OpenHABu

Perzistence

Zajišťuje ukládání stavů položek. V tomto řešení se používá pro uložení aktuální teploty a její vykreslování do grafu. V základním nastavení grafu je rozlišení grafu pouze 240x480 bodů (výsledný kód je umístěn společně s touto prací). Soubor `org.openhab.persistence.rrd4j-verze.jar` lze dekompileovat a je zde možné, mimo jiné, upravit rozlišení, fonty a barvy grafu upravením příslušných hodnot v kódu. Nachází se v adresáři s vazbami.

Bylo třeba upravit soubor perzistencí, aby byla data grafu s počasím uložena každou minutu, při každé aktualizaci a aby se data obnovila při startu systému.

```
Pocasi_Graf*:strategy=everyUpdate,restoreOnStartup,everyMinute
```

Mapy stránek

Výsledný kód je umístěn společně s touto prací. Jedná se mapu stránek s názvem **Semestral** a popiskem **Hlavní menu**. Hlavní menu obsahuje 3 rámečky (Frame), a to:

- Demo,
- HomeMatic,
- Počasí.

Rámeček „Demo“ obsahuje demoverzi aplikace, podobné demoverzi původní, kterou lze stáhnout z oficiálních stránek [12]. Rámeček „HomeMatic“ obsahuje získaná data pomocí systému Homematic a navíc umožňuje komunikaci s rádiem ovládanou zásuvku firmy Homematic. Rámeček „Počasí“ obsahuje interpretované hodnoty získané pomocí vazby Weather. Kromě výše uvedených rámečků je na hlavní straně v základu zobrazené datum, čas, dnešní teplota a graf dnešní momentální teploty, jehož hodnoty se aktualizují každých deset sekund.

Pravidla

Pravidla byla převzata z [12] a následně upravena. Výsledný kód je umístěn společně s touto prací. Příkaz `Time cron "0 0/5 * * * ?"` nastavuje Cron v pravidlu s názvem **Inicializace světel**, tzn. kdy se má daný skript spustit. V tomto případě se skript spustí každou půlhodinu, každý den, každého měsíce, každého roku.

Dále pravidlo generuje náhodné hodnoty (generace nastává každých 30minut) pro každou položku ve skupině **Světla** od 0,0 až po 1,0 a tyto hodnoty porovnává s číslem 0,5. Pokud je podmínka splněna, dané položce je přiřazena hodnota **ON**, v opačném případě hodnota **OFF**.

Pravidlo pro generaci náhodné teploty funguje obdobně, pouze výpočet výsledné teploty je odlišný. K číslu 20,3 je přičten výsledek uvedené rovnice (obr. ??).

Položky

Z každého typu položky je v následujícím textu popsána jedna.

Skupina

Group P_Obyvak "Obývací pokoj" <video> (P).

Jedná se o skupinu s názvem P_Obyvak. Bude mít popisek Obývací pokoj, s obrázkem videa u popisku. Nakonec se přiřadí do skupiny s názvem P (Přízemí).

Stmívač

Dimmer Svetla_P_Obyvak_Stul "Osvětlení stolu" (P_Obyvak,Svetla).

Jedná se o položku stmívač s názvem Svetla_P_Obyvak_Stul, který má popisek "Osvětlení stolu" a dále se přiřazuje do skupin P_Obyvak a Svetla.

Přepínač

Switch Topeni_P_Obyvak "Topení" <heating> (P_Obyvak,Topeni).

Tímto se vytvoří přepínač s názvem Topeni_P_Obyvak, s popiskem "Topení" a obrázku topení u popisku. Opět je přiřazen do dvou skupin, a to P_Obyvak a Topeni.

Rolety

Rollershutter Roleta_P_Obyvak "Rolety" (P_Obyvak,Rolety).

Roleta pojmenovaná Roleta_P_Obyvak, s popiskem Rolety a přeřazená do skupin P_Obyvak a Rolety.

Datum

DateTime Datum "Datum: [%1\$tA,%1\$td.%1\$tm.%1\$tY]" <calendar>.

Jde o datum se jménem Datum a popiskem Datum: název dne, den.měsíc.rok. Veškeré možnosti nastavení formátování jsou dostupné online na [3]. Položka má obrázek kalendáře u popisku a může na něj být navázána ntp vazba, která může mít např. takovouto podobu {ntp="Europe/Prague"}.

Přepínač Homematic

Switch Zasuvka_HomeMatic "Zásuvka" (HomeMatic) {homematic="address=KEQ0177042, channel=1, parameter=STATE"}

Jedná se o přepínač s názvem `Zasuvka_Homematic` a popiskem „Zásuvka“. Patří do skupiny `Homematic`. Je použita vazba `Homematic` s následujícími parametry:

- `address=KEQ0177042` - jde o adresu datapointu¹, kterou lze zjistit v GUI `Homematic`.
- `channel=1` - číslo kanálu pro komunikaci. Taktéž zjistitelné (popř. nastavitelné) v GUI `Homematic`.
- `parameter=STATE` - název datapointu, např. `PRESS_SHORT`, `LEVEL`, apodobně.

Weather vazba - předpověď

`Number Teplota_Dnes "Teplota [%1f °C]" {weather="locationId=home, type=temperature, property=current"}`

Výše uvedené vytvoří číslo s názvem `Teplota_Dnes` a popiskem `Teplota X°C`. Přičímž `X` je hodnota s přesností na jedno desetinné místo. Vazba `Weather` má následující parametry:

- `LocationID=home` - Jde o název lokace, která se uvádí v konfiguračním souboru `openhab.cfg`.
- `type=temperature` - Jde o typ hodnoty, která má být prezentována, např. `athmosphere`, `wind`, `temperature` atp.
- `property=current` - Je to vlastnost typu. Například u typu `temperature` jsou k dispozici následující vlastnosti: `current`, `min`, `max`, `feel`, `dewpoint`, `minMax`.
- `forecast=X` - Kromě zde uvedených parametrů existuje také `forecast=X`, kde `X` udává, kolik dní dopředu se budou data k předpovědi odebírat.

4.2.2 Vytvoření vzhledu aplikace

Pro webovou aplikaci existuje více řešení - bylo vybráno `GreenT`. Pro `smartphone` byla použita a upravena aplikace `HABDroid`.

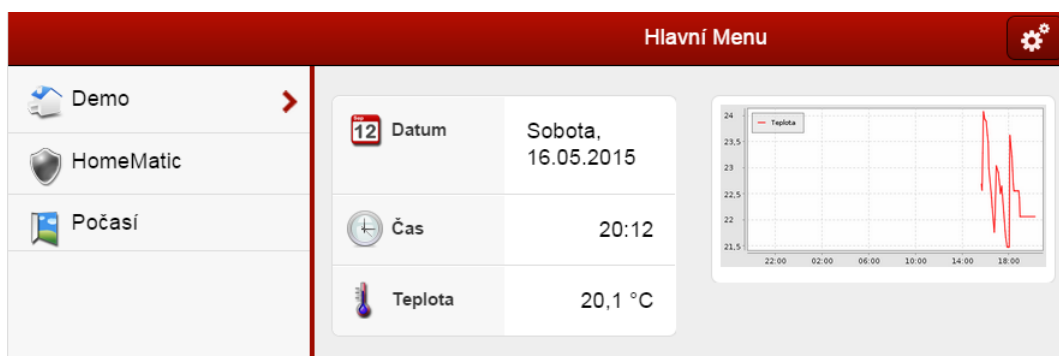
Pro web

Aplikace byla tvořena a optimalizována pro prostředí `GreenT UI` pro verzi runtime 1.6.2. Hlavní menu aplikace je patrné z obrázku (obr. 4.9).

Při stisknutí textové položky „HomeMatic“ dojde k zobrazení instalovaných prvků `Homematic` (obr. 4.10).

Analogicky při stisknutí textové položky „Počasí“ dojde k zobrazení interpretovaných dat, které poskytuje `ForecastIO`. Je zde ukázka některých funkcí, které vazba `Weather` umožňuje jako například: teplota, vlhkost, procento oblačnosti nebo tlak. Součástí vazby je také možnost předpovědi počasí na pět až sedm dní (odvíjí

¹Datapoint – identifikovatelný prvek v souboru dat [12].

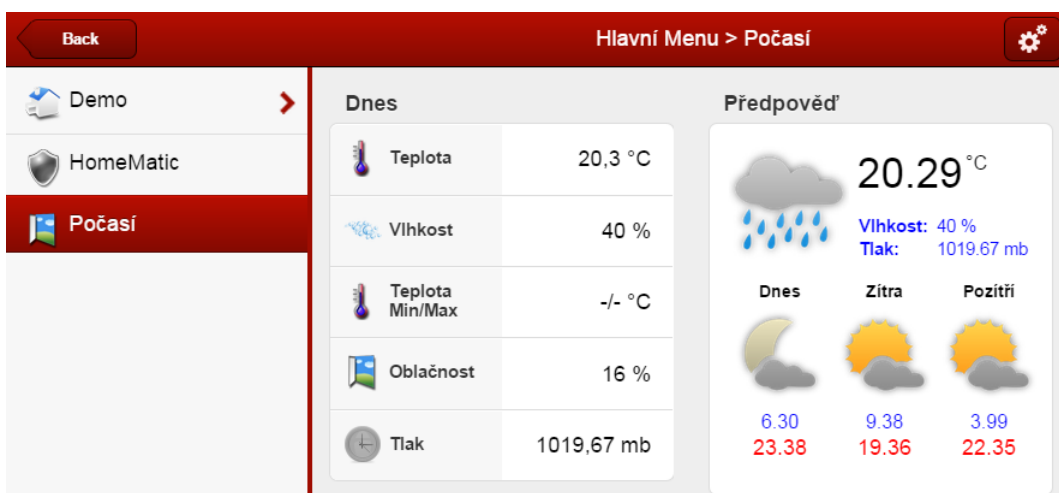


Obr. 4.9: Zobrazení aplikace pomocí GreenT UI



Obr. 4.10: Zobrazení prvků Homematic

se od zvoleného poskytovatele). V příkladu je uvedena základní předpověď na tři dny (obr. 4.11).



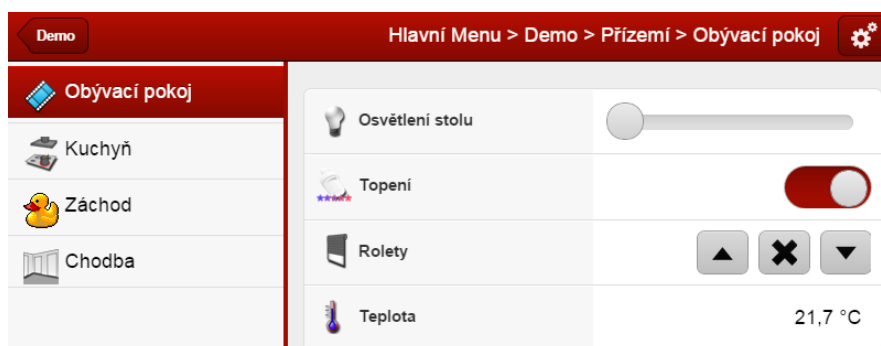
Obr. 4.11: Předpověď počasí za pomoci vazby Weather

Textové pole „Demo“ skrývá výčet několika základních možností, které umožňuje

OpenHAB použít. Obsahuje dvě podpoložky (Přízemí a První patro, obr. 4.12), které obsahují několik dalších položek (např. Obývací pokoj, koupelna atp.).



Obr. 4.12: Zobrazení podlaží



Obr. 4.13: Zobrazení skupiny P_Obyvak

Při výběru například položky „Obývací pokoj“ dojde k zobrazení celé skupiny s názvem P_Obyvak . To znamená, že zobrazeny budou položky: Osvětlení stolu, Topení, Rolety a Teplota (obr. 4.13).

4.2.3 Aplikace pro smartphone

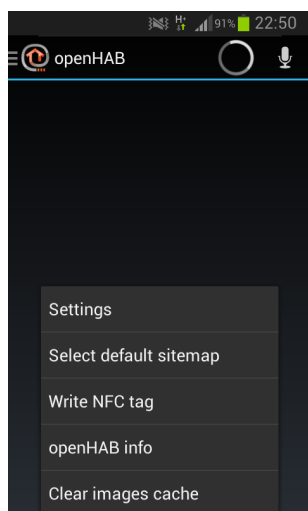
Vývojový tým OpenHAB nabízí aplikaci OpenHAB zdarma ke stažení z Google play (obr. 4.14).

Po spuštění aplikace a její načtení, se zobrazí demo aplikace OpenHAB. Po stisku menu tlačítka na smartphonu se zobrazí menu (obr. 4.15). Pro nastavení OpenHAB url, případně přihlašovacího jména a hesla, je nutno vybrat položku „settings“ (obr. 4.16).

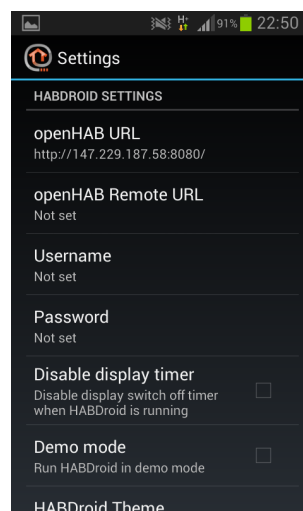
Po zadání korektní OpenHAB URL je možné aktualizovat zobrazení map stránek (je možnost existence několika map stránek, a poté z nich vybrat takovou, která má být zobrazena). Po výběru mapy stránek se načte (obr. 4.17). Stejně jako při zobrazení za pomoci GreenT UI jsou zde tři textové položky, které obsahují další položky. Po výběru položek „Demo“ - „Přízemí“ - „Obývací pokoj“ se zobrazí položky ze skupiny P_Obyvak (obr. 4.19). Stejným způsobem lze najít taktéž koupelnu (obr. 4.18).



Obr. 4.14: Aplikace OpenHAB na Google play



Obr. 4.15: Menu aplikace



Obr. 4.16: Nastavení aplikace

Když bude vybrána položka „HomeMatic“ dojde ke zobrazení položek, které interpretují hodnoty získávané pomocí jednotlivých zařízení firmy Homematic.

Byl testován magnetický kontakt, rádiem ovládaná zásuvka (pro test byla do zásuvky zapojena lampa jako signalizace funkčnosti) a teploměr / vlhkoměr (obr. 4.20 a obr. 4.21). Veškeré testy probíhali v místnosti.

Z výpisu runtime lze vidět, že OpenHAB získává data od poskytovatele ForecastIO a to:

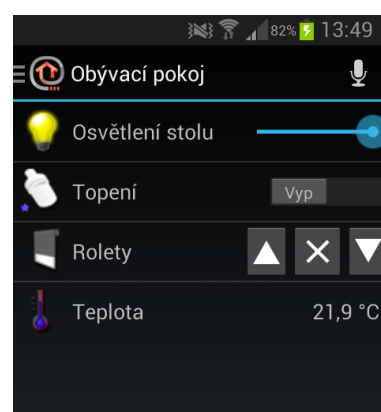
- teploty,
- oblačnost,
- tlak,
- vlhkost,
- komplexní slovní předpověď.



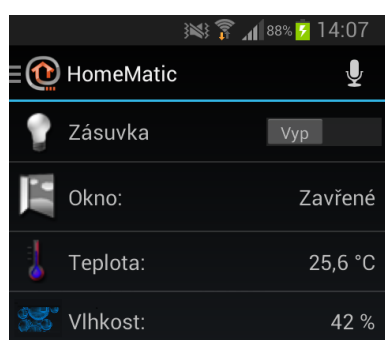
Obr. 4.17: Hlavní menu



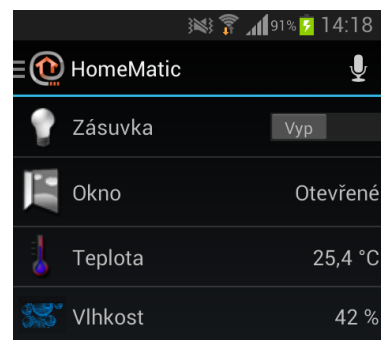
Obr. 4.18: Koupelna - Demo



Obr. 4.19: Obývací pokoj - Demo



Obr. 4.20: Homematic 1



Obr. 4.21: Homematic 2

Dále si lze povšimnout ručního testu rolet v obývacím pokoji a v ložnici, topení a světel v ložnici, a také rádiem ovládané zásuvky firmy Homematic (obr. 4.22).

Kromě zobrazovaných hodnot lze také vidět předpověď komplexních podmínek v anglickém jazyce. I když tato informace nebyla zakomponována do aplikace, je

vhodné ji zmínit jako možnost implementace. Také je zde zaznamenána aktualizace času a data.

```
2015-05-20 13:41:34.553 [INFO ] [runtime.busevents] - Roleta_P_Obyvak received command DOWN
2015-05-20 13:41:35.044 [INFO ] [runtime.busevents] - Roleta_P_Obyvak received command STOP
2015-05-20 13:41:35.353 [INFO ] [runtime.busevents] - Roleta_P_Obyvak received command UP
2015-05-20 13:41:38.145 [INFO ] [runtime.busevents] - Teplota_Dnes state updated to 16.67
2015-05-20 13:41:38.147 [INFO ] [runtime.busevents] - Teplota_min1 state updated to 8.09
2015-05-20 13:41:38.148 [INFO ] [runtime.busevents] - Teplota_min state updated to 11.40
2015-05-20 13:41:38.151 [INFO ] [runtime.busevents] - Mraky_Dnes state updated to 99.00
2015-05-20 13:41:38.152 [INFO ] [runtime.busevents] - Teplota_min2 state updated to 8.12
2015-05-20 13:41:38.154 [INFO ] [runtime.busevents] - Tlak state updated to 1010.32
2015-05-20 13:41:38.155 [INFO ] [runtime.busevents] - Vlhkost_Dnes state updated to 77.00
2015-05-20 13:41:38.161 [INFO ] [runtime.busevents] - Podminky_Komplex state updated to Light
rain throughout the week, with temperatures rising to 27°C on Tuesday.
2015-05-20 13:41:38.167 [INFO ] [runtime.busevents] - Teplota_max state updated to 17.49
2015-05-20 13:41:40.638 [INFO ] [runtime.busevents] - Topeni_PP_Loznice received command OFF
2015-05-20 13:41:41.396 [INFO ] [runtime.busevents] - Svetla_PP_Loznice Strop received command ON
2015-05-20 13:41:42.407 [INFO ] [runtime.busevents] - Roleta_PP_Loznice received command UP
2015-05-20 13:41:42.540 [INFO ] [runtime.busevents] - Roleta_PP_Loznice received command UP
2015-05-20 13:41:44.334 [INFO ] [runtime.busevents] - Cas state updated to 2015-05-20T13:41:44
2015-05-20 13:41:44.340 [INFO ] [runtime.busevents] - Datum state updated to 2015-05-20T13:41:44
2015-05-20 13:41:46.642 [INFO ] [runtime.busevents] - Zasuvka_HomeMatic received command ON
2015-05-20 13:41:47.886 [INFO ] [runtime.busevents] - Zasuvka_HomeMatic received command OFF
```

Obr. 4.22: Příkazový řádek - vazby Weather, Homematic a ruční test

5 ZÁVĚR

Tato práce byla zaměřena na možnosti automatizace domácnosti, popis jejich architektury, konfigurace a instalace systémů, přičemž je kladen důraz na teoretický rozbor systému OpenHAB. V rámci práce byly zdokumentovány vybrané vazby sloužící pro ovládání inteligentních periférií v domě. OpenHAB byl zprovozněn na zařízení Raspberry Pi a následně byl vytvořen demonstrátor pro znázornění funkčnosti komunikace OpenHABu s perifériemi z řady Homematic. Byla vytvořena mapa stránek, kterou je možné interpretovat v aplikaci OpenHAB pro Android (HABDroid).

Demonstrátor Raspberry Pi byl ovšem touto aplikací enormně zatížen a vykazoval delší odezvy na uživatelské interakce (jedna až patnáct sekund). Zároveň při zvyšujícím se počtu vazeb narůstala doba načítání Runtime (na několik desítek minut při použití veškerých dostupných vazeb, kterých je cca šedesát). Avšak pro toto zadání není potřeba tolik vazeb implementovat a postačí jich šest až deset.

Navíc byla použita vazba Weather, která umožňuje získávat data týkající se předpovědi počasí. Byla použita jako demonstrace rozšíření. Použitá metoda automatizace domácnosti se jeví jako nenáročná pro realizaci, navíc s možností jednoduchého rozšíření o další prvky domácí automatizace.

LITERATURA

- [1] Callback. [online]. [cit. 2013-11-17]. Dostupné z:
[http://technet.microsoft.com/cs-cz/library/cc784461\(v=ws.10\).aspx](http://technet.microsoft.com/cs-cz/library/cc784461(v=ws.10).aspx)
- [2] Conrad [online]. 2014 [cit. 2014-03-16]. Dostupné z:
<http://www.conrad.com/ce/>
- [3] DateTime [online]. [cit. 2015-15-05]. Dostupné z:
<http://docs.oracle.com/javase/7/docs/api/java/util/Formatter.html#dt>
- [4] EIB. [online]. [cit. 2013-11-17]. Dostupné z: <http://cs.wikipedia.org/wiki/EIBA>
- [5] Formátování SD. SD Association [online]. 2013 [cit. 2013-12-15]. Dostupné z:
https://www.sdcard.org/downloads/formatter_4/
- [6] GPIO. Wikipedia [online]. 15.9.2013 [cit. 2013-12-15]. Dostupné z:
http://en.wikipedia.org/wiki/General-purpose_input/output
- [7] HAIDY [online]. 2013 [cit. 2014-03-03]. Dostupné z:
<http://www.haidy.cz>
- [8] Insight Home [online]. 2013 [cit. 2014-03-03]. Dostupné z:
<http://www.insighthome.eu/inHome.html>
- [9] Java. JDK8 [online]. 2013 [cit. 2013-12-15]. Dostupné z:
<https://jdk8.java.net/download.html>
- [10] KNX. [online]. [cit. 2013-11-17]. Dostupné z:
[http://en.wikipedia.org/wiki/KNX_\(standard\)](http://en.wikipedia.org/wiki/KNX_(standard))
- [11] Open Source. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001 [cit. 2013-11-17].
Dostupné z: http://en.wikipedia.org/wiki/Open-source_software
- [12] Openhab [online]. 2013 [cit. 2013-10-12]. Dostupné z:
<https://github.com/openhab/openhab/wiki>
- [13] Quartz [online]. [cit. 2013-11-03]. Dostupné z:
<http://www.quartz-scheduler.org/documentation/quartz-2.1.x/tutorials/tutorial-lesson-06>
- [14] Raspberry Pi. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2013-12-01]. Dostupné z:
http://en.wikipedia.org/wiki/Raspberry_Pi

- [15] Raspberry Pi OS. Raspberry Pi [online]. 2013 [cit. 2013-12-15]. Dostupné z: <http://www.raspberrypi.org/downloads>
- [16] Raspberry Pi. Spomocnik [online]. 17.12.2012 [cit. 2013-12-01]. Dostupné z: <http://spomocnik.rvp.cz/clanek/16199/raspberry-pi-a-jeho-potencial-proskolstvi.html>
- [17] Raspberry Pi. Pingbin [online]. 2013 [cit. 2013-12-15]. Dostupné z: <http://pingbin.com/2012/12/30-cool-ideas-raspberry-pi-project/>
- [18] Raspberry Pi. Raspberry Pi [online]. 2013 [cit. 2013-12-15]. Dostupné z: <http://www.raspberrypi.org/wp-content/uploads/2011/07/RaspiModelB.png>
- [19] Raspberry Pi. Root [online]. 1.9.2011 [cit. 2013-12-01]. Dostupné z: <http://www.root.cz/clanky/raspberry-pi-miniurni-arm-pocitac-za-parstovek>
- [20] Raspbian [online]. [cit. 2013-12-01]. Dostupné z: <http://www.raspbian.org>
- [21] RCP. Eclipse wiki [online]. [cit. 2013-12-02]. Dostupné z: http://wiki.eclipse.org/Rich_Client_Platform
- [22] Savant [online]. 2013 [cit. 2014-03-03]. Dostupné z: <http://www.savantsystems.com/>
- [23] SOCIETY, IEEE Power. POWERCON 2012 Auckland [IEEE PES International Conference on Power System Technology: POWERCON : Auckland, New Zealand : October 30 (Tuesday)-November 2 (Friday) [2012]]. Piscataway, N.J.: IEEE. ISBN 978-146-7328-685.
- [24] Stažení RunTime. OpenHAB [online]. 2013 [cit. 2013-12-15]. Dostupné z: <https://code.google.com/p/openhab/downloads/list>
- [25] WebApp.Net [online]. [cit. 2013-11-06]. Dostupné z: <http://trywebapp.net>
- [26] Weather [online]. [cit. 2015-11-05]. Dostupné z: <https://github.com/openhab/openhab/wiki/Weather-Binding>
- [27] Xtend [cit. 2013-11-03]. Dostupné z: <http://www.eclipse.org/xtend/documentation.html>

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

OpenHAB otevřená sběrnice domácí automatizace – Open Home Automation Bus

JVM Java virtuální stroj – Java Virtual Machine

ARM Pokročilý RISC stroj – Advanced RISC Machine

RISC Redukovaná instrukční sada – Reduced Instruction Set Computing

AJAX asynchronní javascript a XML – Asynchronous JavaScript and XML

XML Rozšířený značkovací jazyk – Extensible Markup Language

OSGi Open Services Gateway initiative

KNX sběrnice KNX – Konnex Bus

EIB European Instalation Bus

HTTPS Hypertext Transfer Protocol Secure

SSL Secure Socket Layer

JAAS Java Authentication & Autohrization Service

Rest API Representational State Transfer Application Programming Interface

Eclipse RCP Eclipse Rich Client Platform

PBX Private Branch eXchange

XBMC Xbox Media Center

HDMI High-Definition Multimedia Interface

GPIO General Purpose Inpup/Output

SEZNAM PŘÍLOH

A Konfigurace CCU	52
A.1 Použitá zařízení	52

A KONFIGURACE CCU

Následuje série pokynů krok po kroku, které navádějí ke správné instalaci. Začíná se pokynem k vsunutí baterií a síťového adaptéru (obr.A.1). Následně se zobrazí informace, že se CCU načítá (žlutá, servisní LED bliká v intervalech a zelená, napájecí LED svítí). Načtení je hotovo jakmile žlutá LED přestane blikat a zelená LED svítí nepřetržitě (obr.A.2). Průvodce nabádá pro instalaci USB ovladače a poté k propojení CCU a počítače USB kabelem (obr.A.3). Nyní je základní nastavení hotovo a může se přejít do GUI pomocí webového prohlížeče zadáním adresy (v tomto případě): „10.101.81.51“ nebo „http://home.matic“. Po zadání výše uvedené adresy se GUI načte (obr.A.4), a poté se zobrazí (obr.A.5). CCU nezná jednotlivé zařízení, a proto je nutné tyto zařízení tzv. naučit. To se provede buď podržením libovolného tlačítka CCU na dobu delší než 5sekund nebo zvolením volby „naučit zařízení“, která se nachází v hlavní liště GUI (obr.A.6). Pokud byla zvolena možnost stisku tlačítka na CCU, display zobrazí hlášení, že je na jednu minutu zapnutý režim učení. Nyní musí být v daném intervalu (1 minuta) stisknuto tlačítko učení i na zařízení, které má být naučeno. Po uplynutí jedné minuty CCU zobrazí nalezené zařízení. Pokud byla zvolena možnost učení skrze GUI, tak se pouze vybere volba „naučit zařízení“, a poté se v zobrazeném okně (obr.A.7) zvolí možnost „BidCoS RF teach“. Opět je analogicky režim učení zapnut na dobu jedné minuty (obr.A.8). V inboxu se poté nachází nalezené zařízení (obr.A.9). V záložce nastavení v GUI, lze zařízení otestovat, zdali je funkční, a poté volbou „dokončit“, dokončit instalaci zařízení (obr.A.10).

A.1 Použitá zařízení

Pro demonstraci možností využití systému Homematic, byly zvoleny následující zařízení:

- magnetický kontakt HM-Sec-SC,
- teploměr/vlhkoměr HM-WDS10-TH-O,
- rádiem ovládaná zásuvka HM-LC-Sw1-Pl-2,
- centrála CCU1 HM-Cen-3-1.



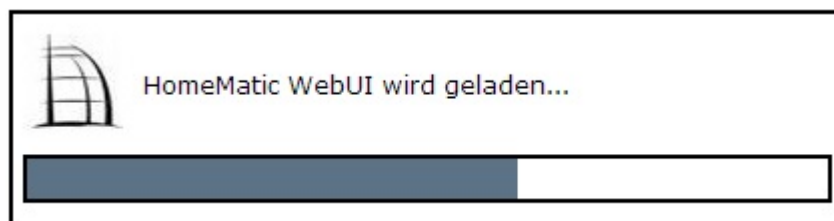
Obr. A.1: Napájení CCU



Obr. A.2: Načítání CCU



Obr. A.3: USB ovladač a zapojení



Obr. A.4: Načítání GUI



Obr. A.5: Zobrazené GUI

Admin Domů > Zobrazení stavu > Kanály		Alarmmeldungen (4)		Odhliždit se	
		Serviceemeldungen (3)			
Bomskri	Operace	Obříbené	Zobrazení stavu	Programy a zkratky	Nastavení
Naučte zařízení					
Název	Místo	Obchod	Poslední aktualizováno	Postavení	
Filtr	Filtr	Filtr			
HM-Sac-SC KE0018264: 1		Oleřnice	28/01/2020 14:47:35		

Obr. A.6: Učení zařízení

Naučte zařízení	
<p>BidCoS RF (radio)</p> <p>BidCoS-RF - Varianta 1: Přímý Teach</p> <p>Naučte režim není aktivní</p> <p>Režim BidCoS RF teach</p> <p>Pokud máte zařízení, které se naučí v přímém přístupu, přímá výuka je možná snadno: Stiskněte tlačítko "režimu Teach BidCoS RF". To znamená, že centrum minuty je umístěn v režimu učení. Během této doby, pak Arlemlaster učení přístroj musí být aktivován. Pro více informací, obraťte se na provozní a instalační manuál příslušného zařízení.</p>	<p>BidCoS-RF - Varianta 2: Učíte se s pořadovým číslem</p> <p>Zadejte sériové číslo:</p> <p>Naučte zařízení</p> <p>Pokud nemáte na výuku jednotku v přímém přístupu, vstup do sériového čísla a stiskněte tlačítko "Training Device".</p> <p>Poznámka: Ne každý přístroj BidCoS RF podporuje Teach podle sériového čísla.</p>
<p>BidCoS-Wired</p> <p>BidCoS-Wired - Varianta 1: Automatické Teach- case Ve</p> <p>musíte se naučit v přímém přístupu jednotky, přímá výuka je možná snadno: Spuště programovací proceduru na přístroj se učí. Instrukce k aktivaci režimu učení na domácí automatické zařízení, naleznete v příručce k zařízení. Panel se pak učí zařízení bez provedení jakékoli operace.</p>	<p>BidCoS-Wired - Varianta 2: Hledat na zařízení</p> <p>Hledat zařízení</p> <p>Pokud nemáte na výuku jednotku v přímém přístupu, Teach pomocí funkce vyhledávání je možné. Udělej to, stiskněte tlačítko "Vyhledat zařízení" Centrální pak dozví nezávisle všechny nové BidCoS-drátové zařízení.</p>
<p>Zpátky</p> <p>Inbox (0 nové vybavení)</p>	

Obr. A.7: Učení neaktivní

Naučte zařízení	
<p>BidCoS RF (radio)</p> <p>BidCoS-RF - Varianta 1: Přímý Teach</p> <p>Naučte režim je stále aktivní 24s</p> <p>Režim BidCoS RF teach</p> <p>Pokud máte zařízení, které se naučí v přímém přístupu, přímá výuka je možná snadno: Stiskněte tlačítko "režimu Teach BidCoS RF". To znamená, že centrum minuty je umístěn v režimu učení. Během této doby, pak Arlemlaster učení přístroj musí být aktivován. Pro více informací, obraťte se na provozní a instalační manuál příslušného zařízení.</p>	<p>BidCoS-RF - Varianta 2: Učíte se s pořadovým číslem</p> <p>Zadejte sériové číslo:</p> <p>Naučte zařízení</p> <p>Pokud nemáte na výuku jednotku v přímém přístupu, vstup do sériového čísla a stiskněte tlačítko "Training Device".</p> <p>Poznámka: Ne každý přístroj BidCoS RF podporuje Teach podle sériového čísla.</p>
<p>BidCoS-Wired</p> <p>BidCoS-Wired - Varianta 1: Automatické Teach- case Ve</p> <p>musíte se naučit v přímém přístupu jednotky, přímá výuka je možná snadno: Spuště programovací proceduru na přístroj se učí. Instrukce k aktivaci režimu učení na domácí automatické zařízení, naleznete v příručce k zařízení. Panel se pak učí zařízení bez provedení jakékoli operace.</p>	<p>BidCoS-Wired - Varianta 2: Hledat na zařízení</p> <p>Hledat zařízení</p> <p>Pokud nemáte na výuku jednotku v přímém přístupu, Teach pomocí funkce vyhledávání je možné. Udělej to, stiskněte tlačítko "Vyhledat zařízení" Centrální pak dozví nezávisle všechny nové BidCoS-drátové zařízení.</p>
<p>Zpátky</p> <p>Inbox (0 nové vybavení)</p>	

Obr. A.8: Učení aktivní

Naučte zařízení	
<p>BidCoS RF (radio)</p> <p>BidCoS-RF - Varianta 1: Přímý Teach</p> <p>Naučte režim je stále aktivní 24s</p> <p>Režim BidCoS RF teach</p> <p>Pokud máte zařízení, které se naučí v přímém přístupu, přímá výuka je možná snadno: Stiskněte tlačítko "režimu Teach BidCoS RF". To znamená, že centrum minuty je umístěn v režimu učení. Během této doby, pak Arlemlaster učení přístroj musí být aktivován. Pro více informací, obraťte se na provozní a instalační manuál příslušného zařízení.</p>	<p>BidCoS-RF - Varianta 2: Učíte se s pořadovým číslem</p> <p>Zadejte sériové číslo:</p> <p>Naučte zařízení</p> <p>Pokud nemáte na výuku jednotku v přímém přístupu, vstup do sériového čísla a stiskněte tlačítko "Training Device".</p> <p>Poznámka: Ne každý přístroj BidCoS RF podporuje Teach podle sériového čísla.</p>
<p>BidCoS-Wired</p> <p>BidCoS-Wired - Varianta 1: Automatické Teach- case Ve</p> <p>musíte se naučit v přímém přístupu jednotky, přímá výuka je možná snadno: Spuště programovací proceduru na přístroj se učí. Instrukce k aktivaci režimu učení na domácí automatické zařízení, naleznete v příručce k zařízení. Panel se pak učí zařízení bez provedení jakékoli operace.</p>	<p>BidCoS-Wired - Varianta 2: Hledat na zařízení</p> <p>Hledat zařízení</p> <p>Pokud nemáte na výuku jednotku v přímém přístupu, Teach pomocí funkce vyhledávání je možné. Udělej to, stiskněte tlačítko "Vyhledat zařízení" Centrální pak dozví nezávisle všechny nové BidCoS-drátové zařízení.</p>
<p>Zpátky</p> <p>Inbox (0 nové vybavení)</p>	

Obr. A.9: Inbox se zařízeními

● Alarmmeldungen (0)
 ● Servicemeldungen (6)
 Odhlásit se

Programy a zkratky		Nastavení			Naučte zařízení	
Výrobní číslo	Interface / Kategorie	Převodovka režim	Místo	Funkční zkoušky	Akce	Dokončeno
KEQ0177 042	BidCoS-RF	Standardní		<div>Test</div> <div style="background-color: green; color: white; text-align: center;">OK</div> <div>•: •: •</div>	<div>Vymazat</div> <div>Sada</div> <div> <input checked="" type="checkbox"/> provozovat <input checked="" type="checkbox"/> viditelný <input type="checkbox"/> přihlásit </div>	Dokončeno
KEQ0177 042: 1	Přijímač	Standardní		<div>Test</div> <div style="background-color: green; color: white; text-align: center;">OK</div> <div>•: •: •</div>	<div>Sada</div> <div> <input checked="" type="checkbox"/> provozovat <input checked="" type="checkbox"/> viditelný <input type="checkbox"/> přihlásit </div>	<input type="checkbox"/>

Obr. A.10: Dokončení instalace zařízení