

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

ELEKTRONICKÁ INFORMAČNÍ TABULE LCD

ELECTRONIC NOTICE BOARD LCD

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Michal Bureš

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. Zdeněk Bradáč, Ph.D.

BRNO 2019



Diplomová práce

magisterský navazující studijní obor **Kybernetika, automatizace a měření**
Ústav automatizace a měřicí techniky

Student: Bc. Michal Bureš

ID: 195978

Ročník: 2

Akademický rok: 2018/19

NÁZEV TÉMATU:

Elektronická informační tabule LCD

POKYNY PRO VYPRACOVÁNÍ:

1. Proveďte literární průzkum informačních panelů založených na LCD technologii.
2. Navrhněte koncepci HW a SW elektronické LCD nástěnky ovládané systémem na bázi Raspberry PI pro zobrazování informačních dat vybavené rozhraním Ethernet/WiFi.
3. Vytvořte HW rozhraní pro interakci člověk/stroj. Vybavte mikrokontrolérem, realizujte elektroniku jednotky V/V, osadte a oživte.
4. Vytvořte komplexní programové vybavení pro PC i pro mikrokontrolér a demonstруйте správnou funkci. Zaměřte se na prezentování statických i dynamických informací, zajistěte vhodnou administraci. .
5. Otestujte celkovou funkčnost, demonstруйте ji a zhodnoťte zvolené řešení.

DOPORUČENÁ LITERATURA:

Pavel Herout: Učebnice jazyka C, KOPP, 2004, IV. přepracované vydání, ISBN 80-7232-220-6

Dle pokynů vedoucího práce.

Termín zadání: 4.2.2019

Termín odevzdání: 13.5.2019

Vedoucí práce: doc. Ing. Zdeněk Bradáč, Ph.D.

Konzultant:

doc. Ing. Václav Jirsík, CSc.
předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové 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 částí druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Abstrakt

Tato diplomová práce ve svém začátku pojednává o průzkumu existujících řešení elektronických informačních LCD systémů. Účelem tohoto průzkumu je nabytí inspirace, co je uživatelům těmito profesionálními systémy nabízeno, potažmo co od těchto systémů uživatelé očekávají.

Na základě takto nabytých poznatků se dále práce věnuje návrhu vlastního systému, který může sloužit jako další alternativa profesionálních řešení firem na trhu, ve smyslu struktury konceptu řešení zadané úlohy.

Po vlastním návrhu systému práce chronologicky pojednává o zvolených „nástrojích“ použitých pro vlastní realizaci zadané úlohy Elektronické informační tabule LCD. Jak hardwarové, tak i softwarové části úlohy, která tvoří majoritní část této diplomové práce.

V závěru práce jsou prezentovány dosažené praktické výsledky.

Klíčová slova

Raspberry Pi, Raspbian, Python, Apache 2, Django, Celery, MySQL, MariaDB, webová aplikace, Threading, ADS1015, GPIO, Tkinter, GUI, Joystick

Abstract

This diploma thesis deals with the research of existing solutions of electronic information systems. The purpose of this survey is to inspire what is offered to users of these professional systems, or what users expect from these systems.

Based on the acquired knowledge, the thesis also deals with the design of its own system, which can serve as another alternative to professional solutions of companies on the market, in the sense of the concept of solution of the given task.

After the design of the system, the thesis deals chronologically with the selected "tools" used for the actual implementation of the assigned task Electronic Information Boards. Both hardware and software parts of the task, which form the majority of this thesis.

At the end of the work are presented practical results.

Keywords

Raspberry Pi, Raspbian, Python, Apache 2, Django, Celery, MySQL, MariaDB, web application, Threading, ADS1015, GPIO, Tkinter, GUI, Joystick

Bibliografická citace:

BUREŠ, Michal. Elektronická informační tabule LCD [online]. Brno, 2019 [cit. 2019-05-13]. Dostupné z: <https://www.vutbr.cz/studenti/zav-prace/detail/119254>. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. 70 s. Vedoucí práce Zdeněk Bradáč.

Prohlášení

Prohlašuji, že svou diplomovou práci na téma Elektronická informační tabule LCD, jsem vypracoval samostatně pod vedením vedoucího diplomové práce 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 uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské/diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si 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.

V Brně dne: 13. května 2019

.....
podpis autora

Poděkování

Děkuji vedoucímu diplomové práce doc. Ing. Zdeněk Bradáč, Ph.D. za odbornou pomoc a další cenné rady při zpracování mé diplomové práce. Mé poděkování patří i mé přítelkyni, která mne bezmezně podporovala při vývoji mé diplomové práce.

V Brně dne: 13. května 2019

.....
podpis autora

Obsah

Úvod.....	16
1. Průzkum informačních panelů.....	17
1.1 Informační kiosky	17
1.2 Dostupná profesionální řešení	19
1.2.1 Portfolio firmy ELVAC a.s.....	19
1.2.2 Portfolio firmy CZECH MULTIMEDIA INTERACTIVE s.r.o.	20
1.2.3 Portfolio firmy STARMON s.r.o.	20
1.2.4 Portfolio firmy Ki-Wi Digital s.r.o.	21
1.2.5 Dosažené poznatky	22
2. Návrh koncepce HW ZAŘÍZENÍ	23
2.1 Požadavky na HW zařízení	23
2.2 Řídící systém zařízení	23
2.2.1 Raspberry Pi Zero W	25
2.2.2 Raspberry Pi 3 B	25
2.2.3 Raspberry Pi 3 B+.....	26
2.3 HW pro interakci člověk – stroj.....	27
2.3.1 Joystick	27
2.3.2 Signalizace stavů Joysticku	28
2.3.3 Výběr AD převodníku	28
2.4 Návrh zapojení	29
3. Návrh koncepce SW zařízení.....	30
3.1 Operační systém	30
3.1.1 Raspbian.....	30
3.2 Programovací jazyk.....	31
3.2.1 Python	31
3.3 Webový framework.....	32

3.3.1	Flask.....	32
3.3.2	Django.....	32
3.4	Databázové prostředky.....	34
3.4.1	MySQL.....	34
3.4.2	MariaDB.....	34
3.5	GUI nástroje.....	34
3.5.1	PyQT.....	35
3.5.2	Kivy.....	35
3.5.3	Tkinter.....	36
3.6	Požadavky na SW zařízení.....	36
3.6.1	Administrace dat.....	36
3.6.2	Prezentace dat.....	37
3.7	Návrh programového vybavení.....	37
3.7.1	Návrh Webového rozhraní.....	37
3.7.2	Návrh Prezentačního rozhraní.....	39
4.	Realizace zařízení.....	40
4.1	Raspberry Pi 3 Model B.....	40
4.1.1	HDMI.....	40
4.1.2	Wi-Fi / Ethernet.....	40
4.1.3	Diskový prostor.....	41
4.2	HW pro interakci člověk-stroj.....	42
4.2.1	Signalizační LED diody.....	42
4.2.2	Stisk Joysticku.....	43
4.2.3	Pohyb Joysticku v osách X & Y.....	43
4.2.4	Deska plošného spoje.....	45
4.3	Programové vybavení.....	46
4.3.1	Webové rozhraní.....	46
4.3.1.1	Webserver.....	46

4.3.1.2	Django struktura	47
4.3.1.3	Přihlášení / Registrace	48
4.3.1.4	Úvod	50
4.3.1.5	Moje údaje	50
4.3.1.6	Moje soubory	51
4.3.1.7	Přehled uživatelů	54
4.3.1.8	Odhlášení	55
4.3.1.9	Bezpečnost.....	55
4.3.1.10	Celery.....	55
4.3.2	Prezentační GUI rozhraní	56
4.3.2.1	Základní stavba GUI.....	56
4.3.2.2	Tvorba Widgetů	57
4.3.2.3	Funkce jednotlivých tlačítek.....	58
4.3.2.4	Kontrola platnosti dat	59
4.3.2.5	Paralelní proces.....	60
4.3.2.6	Generování obsahu v GUI	61
4.3.2.7	Auto update prezentovaných dat	61
4.3.2.8	Auto prezentace	62
4.3.2.9	Defaultní stránka.....	63
5.	Závěr.....	64

Seznam použitých symbolů, zkratek a veličin

1:N	přiřazuje jednomu záznamu více záznamů z jiné tabulky
3G	třetí generace mobilních telekomunikačních technologií
5G	pátá generace mobilních telekomunikačních technologií
AD	analogově - digitální
ARM	architektura procesorů
AVI	video komprimační formát (Audio Video Interleave)
BMP	grafický formát (Bit Map)
CC	konstantní proud (Constant Current)
CPU	Central processing unit
DB	databáze
DPS	deska plošných spojů
GB	množství informace (Gigabyte)
Gbit/s	přenosová rychlost Gigabit za sekundu
GHz	jednotka frekvence (Giga Hertz - 10^9 Hertz)
GIF	Graphics Interchange Format
GND	ground
GPIO	vstupy/výstupy Raspberry Pi (General-purpose input/output)
GPU	Graphic Processing Unit
GTK	zkratka z GIMP Toolkit
GUI	Graphical User Interface
HDMI	High-Definition Multi-media
HTML	Hypertext Markup Language
HW	Hardware
Hz	Hertz
I	proud [A]
I2C	Inter-Integrated Circuit
ID	identita
IoT	Internet of Things
IP	ingress protection
IT	informační technologie

JPG	Joint Photographic Experts Group
LCD	displej z tekutých krystalů (Liquid Crystal Display)
LED	svítivá dioda (Light-Emitting Diode)
LTE	Long Term Evolution
M:N	databázová relace mezi tabulkami
MAX	maximum
MB	množství informace (megabyte)
Mbit/s	přenosová rychlost Megabit za sekundu
MHz	jednotka frekvence (Mega Hertz - 10^6 Hertz)
MicroSD	Micro Secure Digital
micro-USB	Micro Universal Serial Bus
MIN	minimum
MIT	Massachusetts Institute of Technology
MPEG	Moving Picture Experts Group
MVC	Model-View-Controller
MySQL	system pro řízení databází (My Structured Query Language)
NOOBS	New Out Of Box Software
OS	operační systém
P	výkon [W]
PDF	Portable Document Format
PIL	Python Imaging Library
PNG	Portable Network Graphics
Pull-up	rezistor
R	odpor [Ω]
SATA	Serial ATA
SCL	Synchronous Clock
SDA	Synchronous Data
SDRAM	Synchronous Dynamic Random Access Memory
SoC	System on Chip
SPI	Serial Peripheral Interface
SQL	Structured Query Language
SSD	Solid-state drive

SW	Software
U	napětí [V]
UART	Universal Asynchronous Receiver and Transmitter
UPS	Uninterruptible Power Supply
URL	Uniform Resource Locator
USB	Universal Serial Bus
Wi-Fi	wireless
WSGI	Web Server Gateway Interface

Seznam obrázků

Obrázek 1-1: Slavnostního odhalení multimediálního informačního kiosku v příjezdové hale pardubického vlakového nádraží hejtmanem Pardubického kraje Martinem Netolickým (31. 1. 2013), převzato z [5].	17
Obrázek 1-2: Různé formy provedení informačních kiosků, převzato z [17].	18
Obrázek 1-3: Trackball TSX50N8-BT1, laserový vandalproof 50mm, IP68 celek, převzato z [25].	21
Obrázek 2-1: Raspberry Pi 3 model B, převzato z [26].	24
Obrázek 2-2: Raspberry Pi Zero model W, převzato z [27].	26
Obrázek 2-3: Raspberry Pi 3 model B+, převzato z [28].	26
Obrázek 2-4: Joystick KY-023, převzato z [16].	27
Obrázek 2-5: AD převodník ADS1015, převzato z [15].	28
Obrázek 2-6: Návrh zapojení DPS Joysticku.	29
Obrázek 3-1: Work flow architektury MVC.	33
Obrázek 3-2: Neúspěšné spuštění aplikace v Kivy frameworku.	35
Obrázek 3-3: Návrh GUI Prezentační aplikace.	39
Obrázek 4-1: Ekvivalentní zapojení GPIO pinu, převzato z [9].	40
Obrázek 4-2: Popis periférií Raspberry PI 3 model B, převzato z [10].	41
Obrázek 4-3: Vnitřní zapojení Joysticku KY-023, převzato z [14].	43
Obrázek 4-4: Základní zapojení ADS1015, převzato z [22].	44
Obrázek 4-5: Programové oblasti činnosti Joysticku.	45
Obrázek 4-6: Realizovaná periferie člověk-stroj.	45
Obrázek 4-7: Pohled na přihlášení.	49
Obrázek 4-8: Pohled na registraci.	49
Obrázek 4-9: Lišta záložek Webového rozhraní - Úvodní list.	50
Obrázek 4-10: Lišta záložek Webového rozhraní - Moje údaje.	50
Obrázek 4-11: Lišta záložek Webového rozhraní - Moje soubory.	51
Obrázek 4-12: Vyhledávání v přehledu založených Událostí.	51
Obrázek 4-13: Formulář pro založení Události k zveřejnění.	52
Obrázek 4-14: Obsah DB tabulky stranky_materialy se dvěma záznamy.	52
Obrázek 4-15: Obsah DB tabulky stranky_materialy se třemi záznamy.	53

Obrázek 4-16: Obsah DB tabulky stranky_gui.	54
Obrázek 4-17: Lišta záložek Webového rozhraní - Přehled uživatelů.	54
Obrázek 4-18: Automatické odstranění Události za pomoci Celery.	56
Obrázek 4-19: Zobrazené imaginární tlačítko v GUI - bez Fullscreenu.	58
Obrázek 4-20: Výsledky po vyfiltrování platných záznamů z DB stranky_gui.	59
Obrázek 4-21: Paralelní procesy a události.	60
Obrázek 4-22: Časová náročnost generování obrazového obsahu.	61
Obrázek 4-23: Časová náročnost vyčtení záznamů z DB stranky_gui.	62
Obrázek 4-24: Výsledná Prezentační aplikace v režimu Fullscreen.	63
Obrázek 5-1: Schéma zapojení periférií.	71
Obrázek 5-2: Návrh DPS.	71
Obrázek 5-3: Přihlášení do aplikace.	72
Obrázek 5-4: Úvod.	72
Obrázek 5-5: Změna registračních údajů.	73
Obrázek 5-6: Hledání v založených Událostech.	73
Obrázek 5-7: Založení Události.	74
Obrázek 5-8: Přehled registrovaných uživatelů.	74
Obrázek 5-9: Prezentační aplikace.	75
Obrázek 5-10: Raspberry Pi s ovladačem.	76

Seznam tabulek

Tabulka 2-1: Parametry Raspberry Pi Zero model W, převzato z [27].	25
Tabulka 2-2: Parametry Raspberry Pi 3 model B, převzato z [26].	25
Tabulka 2-3: Parametry Raspberry Pi 3 model B+, převzato z [28].	26
Tabulka 2-4: Přehled AD převodníků, převzato z [22],[23],[24].	28
Tabulka 2-5: Význam pinů patice JOYSTICK.	29
Tabulka 2-6: Význam pinů patice RASPBERRY_PI.	29
Tabulka 2-7: Rozpis navržených součástek.	29
Tabulka 2-8: Význam pinů patice ADS1015.	29

ÚVOD

Klasický distribuční řetězec sdílení informací, se skládá z výroby papíru, jeho potištěním informacemi za cenu spotřeby barvy a následného doručení výtisků skupině čtenářům nebo na místa, kde ona distribuce bude probíhat. Tento distribučního řetězce, je náročný na vstupní materiály, lidskou práci, dopravu a čas. A právě čas je mnohdy určujícím kritériem, zda bylo či zda bude mít význam onu informaci sdílet, s ohledem na možnost, že informace během doručování zestárne a ztratí na relevantnosti.

V praxi je již nasazeno mnoho řešení z oblasti informačních technologií, která nedostatky klasického distribučního řetězce nahradila. Použitím prostředků IT vznikla výrazná úspora vstupních materiálů potřebných na výrobu tištěných informací a hlavně poklesl tlak člověka na přírodu při těžbě surovin, potřebných při tisku informací.

Cílem této práce je realizovat Elektronickou informační tabuli LCD, která výše zmíněné překážky klasického komunikačního řetězce překoná a nabídne uživateli, který chce informaci uveřejnit i příjemci zprávy komfort maximální možné míry rychlosti doručení zveřejňované informace, tak i možnost rychlé změny staré informace za informaci novou, aktuální.

Tato práce využívá jako stěžejní prvek Raspberry Pi 3 Model B z oblasti IT prostředků. Díky tomuto základnímu stavebnímu prvku systému je možné využít mnoha dnešních standardů, které jsou silnými nástroji pro realizaci celého systému v jednom zařízení. Právě tyto nástroje Raspberry Pi 3 Model B je nabízí. Jedná se o výkonný procesor, plnohodnotný operační systém, který nabízí možnost použití široké palety programovacích jazyků a dalších nástrojů potřebných pro realizaci mnoha procesů, kterými lze úlohu Elektronické informační tabule LCD realizovat. Připojení k síti Internet prostřednictvím Ethernetu či Wi-Fi je již samozřejmostí.

Realizované řešení musí splňovat dvě kritéria. Dostupnost uživateli, který chce informace uveřejnit, tak i snadné přečtení informací uživatelem, který má být příjemcem zprávy.

V dnešní době je Internet zaveden v drtivé většině měst i obcí prostřednictvím opticko-metalických médií nebo prostřednictvím mobilní sítě 3G, LTE či testované 5G sítě. Přístup k zařízení v této práci probíhá prostřednictvím sítě Internet za účelem maximální možné dosažitelnosti mechanismu uveřejnění informací.

Nasazení této Elektronické informační tabule LCD je navrženo do prostor školy a kategorie, do kterých je možné jednotlivé materiály kategorizovat, jsou tomuto prostředí typické. Dostupnost takto uveřejněných informací je realizováno vhodným umístěním Raspberry Pi 3 Model B do míst s velkou migrací lidí za účelem doručení zpráv potencionálním příjemcům. Zobrazení dat probíhá na libovolné obrazovce LCD připojené k zařízení, spolu s vlastním návrhem a realizací Joysticku pro ovládání aplikace s uveřejňovanými materiály.

1. PRŮZKUM INFORMAČNÍCH PANELŮ

Dnešní informační doba si žádá sdílet a dávat na vědomí velké množství zpráv co nejširší skupině lidí. Ať už se jedná o školy, úřady, banky, vlaková a autobusová nádraží, letiště, hotely, autosalóny, sportovní areály, kongresové sály, divadla, muzea, galerie, zdravotnická zařízení, výrobní prostory a jiná vhodná místa pro sdílení informací či reklam. Jsou to místa, která mají společné rysy. Velká koncentrace či migrace lidí, kteří mají podobný zájem v daném prostoru. A zde je ideální místo, kde lze charakteristické informace s lidmi sdílet či jim je nabízet v, dnes tomu vhodné, elektronické podobě.



Obrázek 1-1: Slavnostního odhalení multimediálního informačního kiosku v příjezdové hale pardubického vlakového nádraží hejtmanem Pardubického kraje Martinem Netolickým (31. 1. 2013), převzato z [5].

1.1 Informační kiosky

Česky Informační kiosky či v anglické literatuře často uváděné pod pojmem Digital Signage, jsou zařízení, která v současnosti tuto skupinu zařízení zastupují a jsou typické pro informační systémy se zobrazovacími panely LCD či LED zobrazovací technologií.

Cílem těchto zařízení je být robustním autonomním řešením, které nabízí jistý komfort a stabilitu použití jak na straně uživatele, který informaci sděluje, tak i na straně uživatele, který informaci přijímá.

Informace, které mohou být sdílené, se na profesionální úrovni firem a jejich řešeních pohybují v řadě formátů, které je možné informačním kioskem zpracovat. Například se jedná o základní obrazové formáty PNG, GIF, BMP a JPG. Co se týče psané podoby informací, profesionální řešení nabízí formáty typu z kolekce Microsoft produktů, jako jsou WORD, EXCEL a POWER POINT. Všechny již zmíněné formáty

se dají sdružit do jednoho univerzálního formátu, který je přenositelný napříč všemi platformami OS osobních počítačů a mobilních zařízení, bez nutnosti zakoupení placeného softwaru, kterým je možné dokument otevřít. Tímto formátem je PDF. Jeho prezentace v profesionálních řešeních je samozřejmostí.

Existují také informační kiosky, které podporují přehrávání videozáznamů se zvukem, v takém případě se jedná nejčastěji o formáty typu MPEG2 a formát AVI.

Typické pro takováto zařízení bývá vzdálená správa zařízení prostřednictvím sítě Internet, jejímž prostřednictvím může uživatel zařízení využívat bez nutnosti fyzické přítomnosti u zařízení. Přičemž funkcionality do přihlášeného zařízení pomocí vzdáleného připojení jsou identické, jako by byl uživatel k informačnímu zařízení připojen místně.

Mezi funkce administračního systému informačních zařízení bývá v první řadě chránění přístup do zařízení, které chrání vlastní zařízení před poškozením, tak i data jednotlivých uživatelů před zneužitím nebo neoprávněné změně zveřejňovaných dat. Po přihlášení či registraci jednotlivých uživatelů do zařízení se uživateli, dle koncepce jednotlivých výrobců, nabízí individuální přístup k nastavení plánu uveřejnění dat, která do zařízení nahrál.

Kromě různorodosti práce s daty k uveřejnění mají profesionální řešení rozdílné platformy operačních systémů, která používají. Mezi tyto operační systémy se řadí různé verze Windows, jednotlivé distribuce systému Linux a Windows 10 IoT.

Společným rysem všech informačních zařízení napříč spektrem dostupných produktů na trhu je, že řídicí systém těchto ucelených systémů je několikanásobně menší, než vlastní zobrazovací zařízení těchto systémů. Zobrazovací zařízení tvoří pouze jeho periférii. Nejčastějším zobrazovacím zařízením bývá LCD obrazovka, která je dostupná v mnoha variantách. Tím dostáváme náhled na prostorové nároky vyvíjeného zařízení.

Další společnou vlastností zařízení je potřeba zveřejněnými daty procházet na straně příjemce. Společná nutnost to sice je, ale způsoby manipulace se zveřejňovanými daty se liší. Jedni výrobci dávají přednost vlastním návrhům ovládacích periférií, jiní se vydávají cestou dotykových obrazovek, pokud je to vhodné a dostačující.



Obrázek 1-2: Různé formy provedení informačních kiosků, převzato z [17].

1.2 Dostupná profesionální řešení

Na českém i zahraničním trhu figuruje nemalý počet firem, které se těmito systémy zabývají, přičemž jejich zařízení jsou z vnějšího laického pohledu stejná. Po technické stránce však disponují dvěma různými typy přístupu řešení vnitřního systému. Konkrétně se jedná o Lokální systémy a Cloudové systémy. V čem se ale od sebe jednotliví výrobci výrazně liší, je:

- design jednotlivých provedení mechanické části informačního kiosku
- design jednotlivých provedení aplikací a jejich funkcí
- šířka podpory a servisu jejich zařízení

1.2.1 Portfolio firmy ELVAC a.s.

Jedná o ostravskou společnost, která vyvíjí vlastní aplikace a systémy do mnoha oblastí průmyslu. Od roku 2005 vyvíjí také vlastní cloudové informační systémy, provozující na systémech Windows. Z jejich nabídky se 4 kategorie produktů dotýkají tématu této práce. Jedná se o:

- Různé druhy vestavěných počítačů Digital Signage:

 použité operační systémy: Win7/Win8/WES7/WE8S/Win10/Linux

 typy procesorů: Intel 7. generace a další

 operační paměti až 32 GB

 úložná kapacita dle připojeného SSD disku

 konektivita: USB 3.0, Ethernet, HDMI, audio

- Industriální klávesnice

Jedná se o klávesnice, které jsou vyrobeny z různých odolných materiálů, jako je nerez a jsou vodotěsné. Použití této klávesnice je vhodné v náročných aplikacích, jako je průmysl či námořní aplikace či venkovní informační kiosky, které se stanou i odolnější vůči vandalství.

- Industriální kulové myši

Jedná se o převrácenou podobu statické kuličkové myši, kde se kulička netře o podložku, ale uživatel prsty otáčí její horní polokouli. Opět i tyto trackballs jsou vyráběny v různých materiálových provedeních, odolným vnějším vlivům a vandalizmu při použití ve venkovních informačních kioscích.

- Softwarové nástroje

Mezi softwarovými aplikacemi firma nabízí produkt InfoPanels, který umožňuje přehrát běžné formáty médií a spravovat vzdáleně libovolný počet displejů.

- Architektura systému

Je postavena na bázi klient-server. Architektura se skládá z 3 prvků:

Administrátorské stanice, pro kterou nabízejí InfoPanels Layout Editor a InfoPanels Explorer pro vzdálenou správu systému, jako je plánování a tvorba obsahu prezentací
Server, pro který nabízejí InfoPanels Server aplikaci, která má úlohu SQL databáze
Klientská stanice (Displej) s aplikací InfoPanels Player, která zajišťuje přehrávání zvoleného obsahu

Mnoho dalších zajímavých specifik přístupu k dané problematice, které se do této podkapitoly již nevešly, doporučuji čtenáři prostudovat na jejich stránkách.

1.2.2 Portfolio firmy CZECH MULTIMEDIA INTERACTIVE s.r.o.

Jedná se o pražskou firmu, která nabízí cloudové interiérové, exteriérové, platební a speciální kiosky. Kiosky v jejich nabídce také disponují LCD panelem pro zobrazování dat. Pro interakci člověka se strojem mají v nabídce aplikaci touch panelu, klávesnice, trackballu. Při použití kamery v informačním kiosku se stane zařízení interaktivní v návaznosti na pohyby uživatele. Vše i v odolném provedení pro exteriérové či interiérové aplikace. Jejich nabídka softwarových produktů pro zmíněný sortiment je široká.

Jedná se například o:

mediaView – určený pro rotaci prezentovaného materiálu

interactiveView – slouží pro grafickou úpravu prezentovaného obsahu

pdfView – v základní verzi umožňuje prezentaci PDF v předem daném pořadí

webView – umožňuje přizpůsobit webovou prezentaci kioskovému obsahu

WebTouch – je speciální webový prohlížeč upravený pro kiosky s dotykovou obraz.

PowerTouche – nabízený prohlížeč slouží pro dotykové obrazovky, pracující pod operačním systémem Linux Red Hat

GUI aplikace – je napsána pomocí knihovny GTK

Výrobce uvádí, že se rozhodl pro Linuxovou distribuci za účelem bezpečnosti a spolehlivosti jejich permanentně provozovaných zařízení.

1.2.3 Portfolio firmy STARMON s.r.o.

Jedná se o firmu Ing. Ondřeje Kováře z České Třebové se sídlem firmy v Chocni. Tato firma se specializuje na vlastní vývoj a výrobu informačních zařízení v sektoru veřejné dopravy a úředních desek. Veškeré produkty jsou jak ve venkovním, tak i ve

vnitřním provedení. Zajímavostí na těchto produktech je jednoduchý a čistě vypadající design při zachování komfortu ovládání. Produkty této společnosti zahrnují:

Elektronické jízdní řády – zde je zajímavou funkcionalitou zařízení hlasový výstup pro nevidomé a mód zobrazení pro slabozraké

Elektronická úřední deska – u této realizace je zajímavé použití přibližovacích tlačítek a šetřič obrazovky při nečinnosti

Venkovní reklamní displeje – umožňuje správu multimediálních spotů (videa, obrázky, html a streamovaný obsah)

RegioInfo venkovní informační kiosek – s funkcí webového kiosku s rozcestníkem preferovaných stránek a blokováním zakázaných stránek

Elektronický označnick – disponuje synchronizací času, regulací jasu displeje dle okolních podmínek s UPS či solárním napájením

Bohužel výrobce neuvádí použitý hardware. Tudíž inspiraci nabízí ve formě designu čistotě a funkční jednoduchosti těchto realizací.

1.2.4 Portfolio firmy Ki-Wi Digital s.r.o.

Jedná se o brněnskou společnost, která vyvíjí vlastní software i hardware. Mezi jejich softwarové produkty patří: Ki-Wi Server, Ki-Wi Player, Ki-Wi Kiosk, Ki-Wi Guide a Digitální úřední deska. Mezi jejich hardwarové produkty patří: Informační kiosky, Informační panely, Reklamní LCD panely a pronájem Hardware. Informační kiosky této společnosti lze využívat na všech platformách prostřednictvím internetového prohlížeče. Tato firma má za sebou mnoho kvalitních realizovaných cloudových projektů na v oblastech řešení pro maloobchod, města, obce, státní správu, reklamní agentury, pro průmyslové a výrobní podniky po celé České republice.



Obrázek 1-3: Trackball TSX50N8-BT1, laserový vandalproof 50mm, IP68 celek, převzato z [25].

1.2.5 Dosažené poznatky

Zobrazovací zařízení:

- LED – 50 x vyšší svítivost než u LCD a násobně delší životnost; proměnlivá elektrická spotřeba dle prezentovaného obsahu
- LCD – bez speciálních filtrů či bez použití odolných LCD panelů konstruovaných na přímé sluneční podmínky mají běžné panely určené především pro vnitřní použití či venkovní bez přímého slunečního světla; konstantní elektrická spotřeba bez ohledu na prezentovaný obsah

Formáty zobrazovacích zařízení:

- neomezené, dle nabídky trhu
- při realizaci velkoformátových LCD je nutné dodržovat provozní podmínky dodatečným chlazením v létě či vytápěním v zimním období

Ovládací prvky jednotlivých zařízení:

- navigační mechanická tlačítka
- klávesnice
- touchpad
- trackball – statická myš s kuličkou
- softwarová tlačítka na dotykové obrazovce
- softwarová klávesnice na dotykové obrazovce

Operační systémy:

- Windows
- Linux

Datová uložště:

- lokální v daném zařízení
- cloudové mimo dané zařízení

Databázové systémy:

- SQL
- MySQL
- Oracle

Programovací jazyky:

- C
- C++
- Python
- Java

Architektury:

- klient – server

Provedení info kiosků:

- vnitřní / venkovní

Správa dat:

- aplikací na externím zařízení
- webovým rozhraním

Vizualizace dat:

- webovým rozhraním
- GUI na zobrazovacím zařízení

2. NÁVRH KONCEPCE HW ZAŘÍZENÍ

Díky definovaným požadavkům na hardware zařízení je tato kapitola rozdělena do čtyř podkapitol. Tyto podkapitoly pojednávají o jednotlivých komponentách zapojení, které jsou potřebné k realizaci HW zadané úlohy v plném rozsahu. V samotném závěru kapitoly je navrženo zapojení všech vybraných komponent.

2.1 Požadavky na HW zařízení

Navrhované zařízení je koncipováno jako univerzální autonomní zařízení, které je situováno do prostor školy, jako alternativa papírové nástěnky pro sdílení informací. Takovéto zařízení má být variabilní pro instalaci v různých prostorách školy. Toto kritérium je splněné plánovaného nasazení v interiéru školy a použitím Raspberry Pi dle zadání, díky němuž jeho rozměry nejsou limitující pro nasazení zařízení do praxe.

Použití zobrazovací jednotky zařízení je určeno zadáním na LCD panel. Jelikož trh nabízí širokou nabídku rozměrů a designů LCD panelů, není toto kritérium překážkou uvedení zařízení do praxe i za předpokladu, že se jedná o nejrozměrnější komponentu zařízení jako celku. Naopak zařízení tím získává adaptivitu na konkrétní prostor instalace. Jelikož zařízení nemá přesně definované umístění instalace a má být univerzální, nebude výběru obrazovky věnována vlastní podkapitola. Nýbrž pro zachování univerzálnosti bude tento bod zohledněn chováním kódu zařízení, které zobrazovaný obsah automaticky přizpůsobí dle charakteru připojené zobrazovací jednotky, kterou si zařízení zidentifikuje.

Úloha realizovaného zařízení má za cíl být ovladatelnou z pohledu uživatele, který je příjemcem uveřejněných zpráv v zařízení. Proto je nutné navrhnout zařízení, které dle zadané úlohy bude využívat mikrokontrolér, který bude zajišťovat interakci mezi zařízením a člověkem. Tomuto bodu je věnována vlastní podkapitola 2.3.

Již zmíněné Raspberry Pi bude základní stavební prvek celého zařízení. Univerzálnost požití realizovaného zařízení by mohla být nevhodnou volbou ponížena, proto výběru konkrétního typu Raspberry Pi je věnována následující podkapitola 2.2.

2.2 Řídící systém zařízení

Základem zařízení je určený mikropočítač Raspberry Pi. Ten vznikl v roce 2012 jako levná alternativa výkonným počítačům, které nejsou po finanční stránce všem dostupné. Autory jsou učitelé z britské univerzity v Cambridgi, kteří hledali cenově dostupnou hardwarovou alternativu pro své studenty k výuce počítačů a programování. Za tímto účelem již v roce 2009 zakladatelé založili nadaci Raspberry Foundation, která je držitelem obchodní známky Raspberry. V době vzniku prvního minipočítače se cena pohybovala okolo 25\$ za jeden kus. Takto cenově dostupný mini počítač o velikosti

platební karty disponoval procesorem, pamětí, grafickou částí, USB portem a zvukovou kartou. Dnes je nejrozšířenější platformou jednodeskových mini počítačů, kterých si koupilo přes 12,5 milionu lidí. Od první verze Raspberry Pi ušli vývojáři veliký kus cesty a z Raspberry Pi se stal dominantní prvek na trhu ve svém oboru. Proto je vhodným prvkem pro řešení této úlohy.

Od roku 2002 byla vydána řada verzí a jejich variant, které postupně navyšovaly jak svůj výkon, počet periférií, tak i spotřebu elektrické energie za situace zachování si svých malých rozměrů. Všem verzím je společné, že nemají svůj vlastní diskový prostor jakožto interní paměť ani port na připojení SATA disku. Proto je nutné dokoupit MicroSD kartu Class 10 s minimální kapacitou 8 GB. K dnešnímu dni jsou na trhu dostupné tyto varianty zařízení:

- Raspberry Pi 1 A+
- Raspberry Pi 1 B+
- Raspberry Pi 2 B
- Raspberry Pi Zero
- Raspberry Pi 3 A+
- Raspberry Pi Zero W
- Raspberry Pi 3 B
- Raspberry Pi 3 B+

Podrobněji budou představeny poslední tři typy, které jsou nejvhodnějšími kandidáty naplnění zadání bez nutnosti dodatečného připojení externího Wi-Fi modulu k zařízení.



Obrázek 2-1: Raspberry Pi 3 model B, převzato z [26].

2.2.1 Raspberry Pi Zero W

Raspberry Pi Zero W	
CPU	1 GHz single-core ARM1176JZF-S
Architektura	ARMv6Z (32-bit)
SoC	Broadcom BCM2835
SDRAM	512 MB (sdílená s GPU)
GPU	Broadcom VideoCore IV @ 250 MHz
interní paměť	MicroSDHC
HDMI	mini-HDMI
GPIO	40 pinů
Zvuk	digitálně přes Mini-HDMI stereo audio skrz PWM na GPIO
USB	1 micro-USB (přímo z čipu BCM2837)
Ethernet	-----
Wi-Fi	802.11n
Bluetooth	Bluetooth 4.1
Wi-Fi anténa	integrovaná
Rozměry	65 mm × 30 mm × 5 mm
Příkon	0,5 W / 1,75 W

Tabulka 2-1: Parametry Raspberry Pi Zero model W, převzato z [27].

Jedná se o nejmenší Raspberry Pi, což může být v jistých aplikacích předností, tak jako celkový odběr zařízení. Je ovšem pravdou, že menší rozměry znamenají i menší výbava. Z přiložené tabulky je patrné, že oproti dvěma následujícím modelům, disponuje slabším taktem CPU, menší SDRAM, konektor HDMI je ve variantě mini, tak jako provedení jediného micro-USB. Konektivita prostřednictvím metalického připojení chybí. Jedná se o nejlevnější z Raspberry Pi při ceně 313 Kč.

2.2.2 Raspberry Pi 3 B

Raspberry Pi 3 Model B 64-bit 1GB RAM	
CPU	1.2 GHz 64-bit quad-core ARM Cortex-A53
Architektura	ARMv8-A (64/32-bit)
SoC	Broadcom BCM2837
SDRAM	1 GB (sdílená s GPU)
GPU	Broadcom VideoCore IV @ 250 MHz
interní paměť	MicroSDHC
HDMI	HDMI (rev 1.3 & 1.4)
GPIO	40 pinů
Zvuk	Analogový (přes 3,5mm jack) digitální (přes HDMI)
USB	4 porty USB 2.0
Ethernet	10/100 Mbit/s
Wi-Fi	WiFi 802.11n
Bluetooth	Bluetooth 4.1
Wi-Fi anténa	integrovaná
Rozměry	85,60 mm × 56,5 mm × 17 mm
Příkon	1,5 W / 6,7 W

Tabulka 2-2: Parametry Raspberry Pi 3 model B, převzato z [26].

Tato verze přináší nárůst výkonu procesoru o 33 % oproti předchozí variantě Raspberry Pi 2. Efektivnější instrukční sadu procesoru ARMv8, který disponuje 64-bit architekturou. I grafické jádro navýšilo svoji frekvenci o 60 %. Samozřejmostí této poslední verze je bezdrátový chip Cypress CYW43438, který zajišťuje 802.11n wireless LAN, tak Bluetooth 4.1 s integrovanou anténou. Celkový nárůst výkonu je přepočítán na 50 % při stejné konfiguraci obou verzí zařízení. Pořizovací cena této verze se pohybuje kolem 863 Kč.

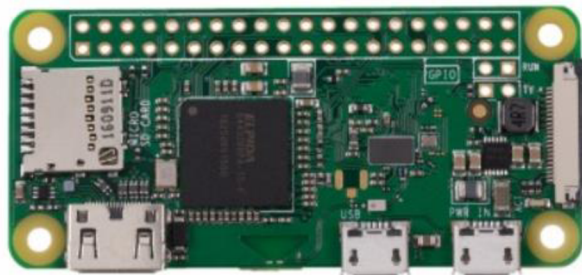
2.2.3 Raspberry Pi 3 B+

Raspberry Pi 3 Model B+ 64-bit 1GB RAM	
CPU	1.4 GHz 64-bit quad-core ARM Cortex-A53
Architektura	ARMv8-A (64/32-bit)
SoC	Optimalizovaný Broadcom BCM2837B0
SDRAM	1 GB (sdílená s GPU)
GPU	Broadcom VideoCore IV @ 250 MHz
interní paměť	MicroSDHC
HDMI	HDMI (rev 1.3 & 1.4)
GPIO	40 pinů
Zvuk	Analogový (přes 3,5mm jack) digitální (přes HDMI)
USB	4 porty USB 2.0
Ethernet	Gigabitový Ethernet
Wi-Fi	WiFi 802.11ac
Bluetooth	Bluetooth 4.2
Wi-Fi anténa	integrovaná
Rozměry	85,60 mm × 56,5 mm × 17 mm
Příkon	1,5 W / 6,7 W

Tabulka 2-3: Parametry Raspberry Pi 3 model B+, převzato z [28].

prostřednictvím Ethernetu. Pořizovací cena této verze se pohybuje kolem 900 Kč.

Tato verze přináší nárůst výkonu procesoru oproti verzi předchozí. Díky přepracovanému napájení bylo možné dosáhnout vyšší pracovní frekvence 1,4 GHz. Posunu vpřed se dočkaly i komunikační periférie, jako je Wi-Fi, které se dostalo i do pracovního pásma 5 GHz. V tomto frekvenčním pásmu se tolik neprojevuje zahlcení mnoha účastníků a následného vzájemného rušení. Rychlost Ethernetu se navýšila na dnešní 1Gbit/s. Bluetooth 4.2 přináší 2,6x nárůst rychlosti předchozí Bluetooth varianty. Připojením PoE HAT modulu je možné Raspberry Pi 3 B+ napájet



Obrázek 2-2: Raspberry Pi Zero model W, převzato z [27].



Obrázek 2-3: Raspberry Pi 3 model B+, převzato z [28].

2.3 HW pro interakci člověk – stroj

Při pohledu na paletu ovládacích prvků, které se nabízí řad výrobků profesionálních aplikací, je nutné konstatovat, že ani jedno nenabízí vhodnou variantu ovládacího prvku s mikrokontrolérem, který by šlo v této úloze s Raspberry Pi připojit. Jejich použití je sice moderní, ale přímočaré a tím tato úloha nechce být.

Jelikož Raspberry Pi má pouze digitální vstupy či výstupy, je nutné zapojení doplnit. Použitím libovolného analogově-digitálního převodníku mezi Raspberry Pi a uživatelský ovládací prvek v provedení Joysticku bude dosaženo splnění zadání v plném rozsahu.

2.3.1 Joystick

Joystick je analogové zařízení, které disponuje dvěma osami pohybu X a Y. Osa X a osa Y je reprezentována potenciometry, které mění svůj odpor v závislosti na výchylce v dané ose. Každá osa je děličem napětí, který má celkový odpor 10k Ω . V klidovém stavu mají oba dva děliče střední hodnotu odporu. To znamená, že se po uvolnění hříbku Joysticku vrátí do výchozí středové polohy.

Vlastnost vychýlení Joysticku jak v ose X, tak v ose Y současně může generovat v jistých aplikacích nežádoucí stavy a chování systému zařízení. Čím menší Joystick uživatel bude ovládat, tím více bude Joystick generovat pohyb ve dvou osách současně, i když uživatel zamýšlel pohyb pouze v jedné ose. Tuto vlastnost je nutné softwarově ošetřit, aby nedošlo k zahlcení prezentační aplikace povely. Pokud by se tak stalo, aplikace by zmatečně přemítala obsah a uživatel by se nemusel vyznat v principu ovládání.

Svislá osa Z složí pro stisk mikrospínače, který se v konstrukci Joysticku nachází a je možné ho využít jako tlačítko potvrzení volby.



Obrázek 2-4: Joystick KY-023, převzato z [16].

2.3.2 Signalizace stavů Joysticku

Během procházení obsahu, zveřejněného v zařízení, má uživatel prvotní možnost zaznamenat odezvu změnou prezentovaného obsahu. Pro vývoj i vlastní použití je vhodné zařízení doplnit signálními prvky, které budou reprezentovat stav systému či jednotlivých stavů Joysticku, jako je poloha či stisk.

Za tímto účelem zapojení bude obsahovat LED diody pro každý ze směrů v ose X, Y a pro stisk mikropínače. Napěťová logika jednotlivých GPIO pinů je 3,3V.

Pro čtení stavu mikropínače je nutné použít softwarově aktivovaného Pull-up rezistoru, jelikož stisknutí mikropínače stav přizemní.

2.3.3 Výběr AD převodníku

Jak již bylo zmíněno, Raspberry Pi nedisponuje analogovými vstupy, které jsou nutné pro připojení dvou signálů z Joysticku, které reprezentují polohu vychýlení v jednotlivých osách. Tudíž je nutné vybrat vhodný analogově-digitální převodník, který bude disponovat minimálně dvěma kanály.

Na výběr je z mnoha převodníků, jako je například MCP3008, MCP3424 a ADS1015.

převodník	pracovní napětí [V]	rozhraní	rychlost vzorkování	rozlišení	počet kanálů
MCP3008	2,7 – 5,5 V	SPI	200k/sec	10 bit	8
ADS1015	2 – 5 V	I2C	3k3/sec	12 bit	4
MCP3424	2,7 – 5,5 V	I2C	240/sec	18 bit	4

Tabulka 2-4: Přehled AD převodníků, převzato z [22],[23],[24].

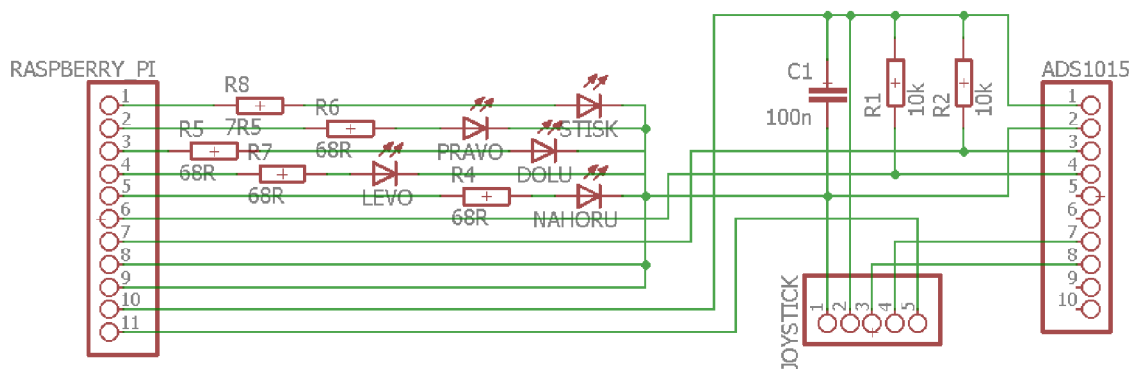
Všechny zmiňované převodníky disponují alespoň 4 kanály. Pro výběr je určující, počet kanálů, přiměřená vzorkovací rychlost s dostatečným rozlišením. Dále rozměr převodníku, jeho cena a náročnost na implementaci do zapojení. Proto je zvolen ADS1015, který se dá pořídit za 132 Kč.



Obrázek 2-5: AD převodník ADS1015, převzato z [15].

2.4 Návrh zapojení

Návrh zapojení se skládá ze čtyř prvků, které byly popsány v předchozích podkapitolách. Konkrétně se jedná o Joystick s analogově-digitálním převodníkem a signalizačními LED diodami. Vše připojené k Raspberry Pi 3 model B, které svým výkonem a výbavou maximálně splňuje předpoklady pro tuto úlohu. Navržené zapojení využívá konstrukci jednotlivých komponent, které využívají řadu pinů, jako vstupy a výstupy. Proto je dostačující vytvořit jednovrstvý plošný spoj, který jednotlivé komponenty propojí.



Obrázek 2-6: Návrh zapojení DPS Joysticku.

Patice RASPBerry_PI	
pin	účel
1, 2, 3, 4, 5	5 signalizačních diod
6, 7	I2C komunikace (SCL,SDA)
8,9	GND
10	$U_{cc} = 3,3\text{ V}$
11	stisk mikropsínače Joysticku

Tabulka 2-6: Význam pinů patice RASPBerry_PI.

Patice JOYSTICK	
pin	účel
1	GND
2	$U_{cc} = 3,3\text{ V}$
3	výstup děliče napětí osy X
4	výstup děliče napětí osy Y
5	stisk mikropsínače Joysticku

Tabulka 2-5: Význam pinů patice JOYSTICK.

Patice ADS1015	
pin	účel
1	$U_{cc} = 3,3\text{ V}$
2	GND
3	I2C komunikace SDA
4	I2C komunikace SCL
7	kanál č.0 (vstup)
8	kanál č.1 (vstup)

Tabulka 2-8: Význam pinů patice ADS1015.

počet součástek	
součástka	počet
R1206 10k	2
CKS1206 100n/50V	1
LED 3mm zelená	4
R1206 56R	4
LED 3mm modrozelená	1
R1206 5R1	1

Tabulka 2-7: Rozpis navržených součástek.

3. NÁVRH KONCEPCE SW ZAŘÍZENÍ

Kvalitnímu návrhu koncepce zařízení předchází průzkum prostředků, kterými lze úlohu řešit. Zadáním určené použití Raspberry Pi je výběr prostředků zúžen na nástroje, které lze použít na Linuxové platformě.

3.1 Operační systém

Možnosti, jaký operační systém lze na Raspberry Pi provozovat jsou nemalé. Jedná se o Linuxové distribuce, které jsou buď distribuovány pod licenci nebo v drtivé většině open-source. Linuxové platformy zaujímají na poli hardwaru významnou část, tudíž i Microsoft si nechce nechat utéci příležitost na tomto poli a zaujmou svou měrou část sektoru. Jelikož je Microsoft velkým hráčem v sektoru osobních počítačů, nebude jejich projekt Windows 10 IoT v této práci brán v úvahu. Výběr operačního systému se zaměří na zaběhlé, časem a uživateli prověřené produkty, jako jsou Debian, Raspbian, RISC OS, PiNet, Ubuntu, Arch Linux a Red Hat.

Při výběru systému pro ARM procesor, který se v Raspberry Pi skrývá, můžeme vyžadovat mnoho parametrů. V tomto případě se jedná v první řadě o stabilitu systému s programovým vybavením, jako celku. Nelze si představit nasazení neznámého a osobně neodzkoušeného operačního systému. Dále si musíme říci, zda chceme operační systém, který bude využívat výkon hardwaru pouze na zpracovávání funkcí a úkolů zadávaných pomocí konzolové řádky nebo zda část výkonu obětujeme i graficky přívětivějšímu prostředí systému. Použitý operační systém musí také dávat širokou možnost použití programových nástrojů ke splnění úlohy, pokud možno bez dodatečných instalací programové výbavy.

3.1.1 Raspbian

Nadace Raspberry Foundation představila oficiální verzi operačního systému pro mini počítače Raspberry s názvem Raspbian Stretch. Jedná se o operační systém, odvozený od Debianu 9, optimalizovaný pro Raspberry. Cílem bylo udělat hezčí a pro lidi použitelnější prostředí s nízkými hardwarovými nároky, které se odrážejí na svižném chodu systému. Předností tedy je, že se jedná o oficiální podporovanou verzi, která je již po prvním nainstalování připravená k provozu. V předchozích verzích to nebylo samozřejmostí, protože zvukový modul, modul Ethernet, Wi-Fi a Bluetooth zprvu nefungovaly bezproblémově.

Raspbian je dostupný z oficiálních stránek Raspberry Foundation spolu s podporou, jak systém snadno nainstalovat, popř. nastavit. Zakomponován je do softwaru NOOBS, který umožňuje jednoduchou instalaci různých operačních systémů. Systém obsahuje prohlížeč Chromium, díky kterému je okamžitě možné přistoupit k Internetu. Další programovou výbavou je Python 2 a Python 3, tudíž se dá okamžitě pracovat na vývoji

aplikací. Pro prohlížení PDF souborů je zde předinstalovaná aplikace Xpdf. S touto základní výbavou je tento mini počítač okamžitě připraven pro komplexní práci na vývoji aplikací, tak i využití veškerého hardwaru zařízení bez nutnosti dodatečné instalace a konfigurace.

Raspbian je možné instalovat ve třech variantách, které se liší velikostí a náročností na hardware. Jedná se o konzolovou variantu, variantu s desktopem a desktopem s předinstalovanými balíčky.

Jelikož Raspberry Pi 3 disponuje výkonným hardwarem, bude v této úloze využit Raspbian Stretch s desktopem a předinstalovanými balíčky. Zohledněno při volbě tohoto systému bylo také kontinuální podpoře autorů, kteří stále pracují na vývoji rychlejšího, efektivnějšího, stabilnějšího systému, šitého na míru.

3.2 Programovací jazyk

Programovací jazyk je základní nástroj, kterým je úlohu nutné zpracovat. Existují tři nejběžnější jazyky, které lze použít. Jedná se o jazyk C, C++ a Python. C++ a C jsou si podobné jazyky ve smyslu syntaxe, tudíž pro programátory v jazyce C se jedná o uchopitelné nástroje. Z pohledu začínajícího programátora je určitě vhodnější použití jazyka Python, který je možné se naučit v relativně krátké době používat. Toto tvrzení je podloženo i faktem, že jazyk Python se vyučuje jako první jazyk na středních školách. Z pohledu předinstalovaných aplikací v systému Raspbian Stretch, je i Python 2 a Python 3 součástí instalace.

Jak již bylo zmíněno, Raspberry Pi bylo vyvinuto za účelem podpory výuky programování a Python je jeho nedílnou součástí. To z jazyka Python činí jazyk, který bude použit v této práci pro vývoj.

3.2.1 Python

Python se řadí mezi programovací jazyky, které se dají naučit relativně za krátkou dobu, tudíž je pro vývoj aplikací vhodným nástrojem. Podporuje tři základní paradigmaty programování (funkcionální, procedurální a objektové). Jelikož se jedná o interpretovaný jazyk, kde jeho kód se překládá až za běhu programu, na chyby v kódu Vás upozorní až během spuštění. Python se řadí mezi open-source projekty, tudíž kód samotného Pythonu je volně čitelný, tak jako i kódy v Pythonu napsané. K dnešnímu dni je vydána nejvyšší verze Python 3.7.3, přičemž na Raspberry Pi jsou okamžitě po instalaci Raspbianu k dispozici výchozí verze Python 2 (verze 2.7.13) a Python 3 (verze 3.5.3). Dále disponuje širokou podporou komunity uživatelů a mnoha lektorských návodů. Python je programovací jazyk, který není cizí ani velkým firmám, jako je Google, NASA, Red Hat, MIT, YouTube, Mozilla a Ebay.

Díky svým přednostem a faktu, že je Python dostupný v základní skladbě Raspbianu, bude programovacím jazykem této práce.

3.3 Webový framework

Webové frameworky jsou nástroje, které jsou svázané s programovacím jazykem Python a umožňují programátorovi realizovat webovou nástavbu na jeho kódy. Díky těmto frameworkům nemusí programátor ztrácet čas psaním MySQL příkazů a může využít syntaxí frameworku, které komunikaci s databází prostřednictvím SQL příkazů (dotazů) obslouží na pozadí. Zástupci těchto řešení patří Django, Pyramid, WEB2PY, Bottle, Flask a Tornado.

3.3.1 Flask

Jedná se o webový mikroframework pro jazyk Python, který vznikl v roce 2010. Tento nástroj se dá naučit za kratší dobu, než jiné frameworky. Obsahuje webový server, který je nutný pro vývoj webové aplikace. Bohužel není tak obsáhlý jako jiné frameworky a neobsahuje administrační prvky, které pro vývoj webové aplikace jsou nutností. Tato skutečnost vede programátora k nutnosti si dodatečně nainstalovat knihovny, které funkce rozšíří a naimportuje je do výsledného projektu. Dále neobsahuje framework pro objektovou práci s databázemi, což opět vede k nutnosti vyhledat vhodný dodatečný nástroj. Tím je dáno, že Flask je vhodný pouze na menší projekty.

3.3.2 Django

Django je komplexní webový framework, který v sobě zahrnuje uživatelskou administraci, správu databáze a mnoho jiných funkcí. Vyznačuje se MVC architekturou svých projektů, která je logická a velice přehledná. Výhodou tohoto přístupu je to, že není vytvářen jeden dlouhý monotónní kód, ve kterém hrozí, že i sám autor ztratí přehled. V principu se jedná o oddělení logiky webu od výstupu, který je uživateli prezentován, tak i oddělení stylů (designu webu). Tuto vlastnost uživatel ocení v případě, kdy potřebuje svůj kód doplnit. Logika webu je v Django prezentována v podobě Python kódu oproti obsahu webu, který se snaží být pouze HTML stránkou, do kterého jde ovšem jistou míru Python kódu implementovat. Takto vytvořené projekty je Django schopno prezentovat prostřednictvím svého integrovaného webserveru, který je součástí instalace.

Jakmile je Django nainstalované, je nutné založit projekt, který se chystáme vyvíjet. V daném projektu je poté možné zakládat jednotlivé aplikace, které Vám logicky oddělí jednotlivé funkcionality Vašeho projektu, který je zastřešuje. Aplikací je míněn projekt nebo část projektu, který se chystáte vytvářet.

Příkladem struktury projektu a aplikace může být:

- projekt – E-shop
 - ├ aplikace1 – login uživatelů
 - ├ aplikace2 – obsah pro neregistrované uživatele
 - ├ aplikace3 – obsah pro registrované uživatele
 - ... a další logické dělení

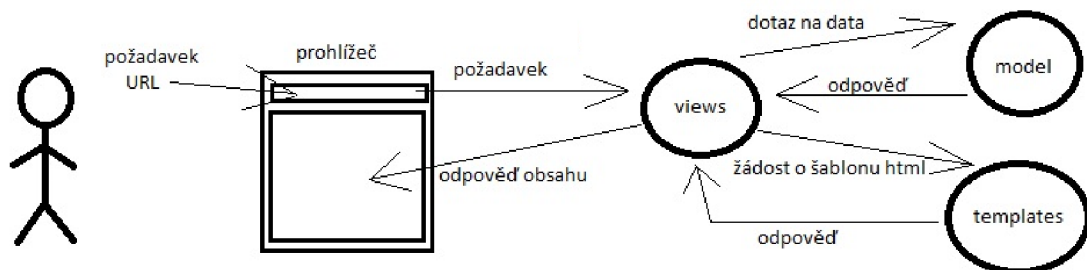
Architektura MVC (Model – View – Controller) rozděluje aplikaci do tří komponent.

Model_{MVC} – obsahuje pouze logiku a řeší databázové dotazy, vrací surová data

View_{MVC} – zobrazuje data uživateli v podobě html šablon; v Django se View_{MVC} nazývá Templates, přičemž ve stejnojmenné složce vyhledává html šablony, se kterými následně pracuje.

Controller_{MVC} – je prostředníkem mezi Modelem a Templates; v Django se Controller_{MVC} nazývá Views.

Dle URL adresy zadané uživatelem je voláno konkrétní Views, které buď přímo vykoná instrukce, nebo se spojí s konkrétním Modelem, který mu data poskytne. Následně tato hotová data (vypočítaná, upravená či vyčtená z databáze) Views předá html šabloně. Takto naplněnou html šablonu Views uživateli zobrazí.



Obrázek 3-1: Work flow architektury MVC.

Z důvodů obsáhlosti funkcí Django, které obsahuje již v základu a pro logické členění vyvíjeného projektu, bude tento framework vybrán pro realizaci této úlohy.

3.4 Databázové prostředky

Jelikož je předpoklad využití Elektronické informační tabule velkým počtem lidí, je nutné efektivně zpracovávat velké množství dat a evidovat je. Za tímto účelem existují 3 hlavní prostředky, které jsou dnes v praxi nejčastější. Dle zvoleného frameworku Django lze zvolit 4 integrované databázové enginy. Jedná se o: postgresql, mysql, sqlite3 a oracle. Nejvíce rozšířené MySQL a jeho alternativa MariaDB bude zmíněno podrobněji.

3.4.1 MySQL

Pro webové aplikace je MySQL nejpoužívanější databází, jelikož je nejspolehlivější a nejvýkonnější. Tato relační databáze je založená na jednotlivých tabulkách, kde její řádku tvoří unikátní záznamy. Sloupce těchto tabulek jsou jednotlivé atributy záznamů. Každý záznam tabulky má svůj unikátní, nezaměnitelný identifikátor. Hodnoty v tabulkách musejí nabývat definovaného datového typu. Příznak, zda se jedná o klíčový atribut a zda může nabývat nulových hodnot. Z pojmu relace vyplývá, že mezi jednotlivými záznamy a tabulkami existují vazby, které jsou jasně definovány. Např. 1:1, 1:N, M:N.

3.4.2 MariaDB

Je relační databáze, která vznikla jako odnož MySQL v dobách, kdy Oracle koupila MySQL. MariaDB si zachovává zpětnou kompatibilitu i proto, že ji založil a vývoj vede stejný člověk, který se věnoval projektu MySQL od začátku (Monty Widenius). Ve své podstatě se jedná o open-source projekt s GPL licencí a tím se četnost použití MariaDB v aplikacích stává častější záležitostí. Firmy jako je Google a Red Hat se o tuto alternativu taktéž zajímají. Pokud je číslo verze MariaDB stejné, jako číslo verze MySQL, dávají tím autoři najevo, že jde po funkční stránce o ekvivalent dané verzi MySQL.

3.5 GUI nástroje

Jedná se o knihovny, které jsou již importovány, nebo je možné dodatečně naimportovat do jazyka Python. Jejich prostřednictvím je možné vytvářet Graphical User Interface (GUI), která jsou uživatelům dobře známá jako okenní aplikace, ve kterých uživatel data přijímá a pracuje s nimi v grafickém režimu. Jedná se o psaní kódu, jehož části interpretují jednotlivé prvky GUI. Těmito prvky (Widgets) jsou například tvorba vlastního okna, tvorba tlačítek, textových polí, zaškrtnutí, výběry, menu, otevírání obrázků, práce s pozadím, práce s písmem a rozmístění prvků v okně GUI. Pro komplexní GUI je nutné všechny jmenované prvky kódem specifikovat. Existuje široká nabídka knihoven, ze které budou zmíněny tři nejzajímavější.

3.5.1 PyQT

Aplikace vytvořené tímto frameworkem jsou nejzdařilejší na design. Jedná se o multiplatformní QT napsaný v C++, které je vhodné i pro psaní aplikací pro mobilní telefony. PyQT disponuje podporou vláken, SQL databází, grafů a mnoha dalších funkcí. Dále je k dispozici program Qt Designer, jeho prostřednictvím má uživatel možnost navrhnout podobu GUI graficky. Na konci návrhu Qt Designer vygeneruje Python kód navrženého GUI. Poté role programátora spočívá v dopsání funkcionalit jednotlivým tlačítkům, logiky či akčním prvkům GUI. PyQT se zdá jako robustní řešení při návrhu GUI, ale určitě není vhodné pro začátečníky.

3.5.2 Kivy

Je framework určený pro jazyk Python. Také nedisponuje nejnovějším líbivým designem grafických prvků, ale i tak se dá použít na realizaci velice zdařilých aplikací. Hlavní předností Kivy je, že podporuje dotykové obrazovky, dokonce multi-touch obrazovky. Tím se otevírá využitelnost aplikací na dotykových obrazovkách mobilních telefonů a tabletů. Právě pro psaní her pro Android je Kivy ideálním řešením. Použitelnost projektů je především na platformách Linux, Windows, MacOS, Raspberry Pi a Android. Při mé snaze instalovat Kivy na Raspberry Pi jsem narážel na samé obtíže, i přestože jsem postupoval dle instrukcí na oficiálních stránkách kivi.org. Instalaci se povedlo realizovat za pomoci diskuzních fór a doinstalování knihoven, o kterých se oficiální zdroje nezmiňují. Následně jsem napsal pár řádek kódu pro spuštění triviálního okna „Hello World“, kde mi kód zhavaroval se zprávou o neoprávněném přístupu do paměti.

```
APKA) pi@STROJ:~/AA $ python kal.py
INFO ] [Logger      ] Record log in /home/pi/.kivy/logs/kivy_19-04-11_2.txt
INFO ] [Kivy         ] v1.11.0.dev0, git-7d6f536, 20190410
INFO ] [Python       ] v3.7.0 (default, Apr  8 2019, 00:10:21)
GCC 6.3.0 20170516]
INFO ] [Factory      ] 184 symbols loaded
INFO ] [Image        ] Providers: img_tex, img_dds, img_pil, img_gif (img_sdl2
img_ffpyplayer ignored)
INFO ] [Text         ] Provider: sdl2
INFO ] [Window       ] Provider: egl_rpi
INFO ] [GL           ] Using the "OpenGL ES 2" graphics system
INFO ] [GL           ] Backend used <sdl2>
Neoprávněný přístup do paměti (SIGSEGV)
```

Obrázek 3-2: Neúspěšné spuštění aplikace v Kivy frameworku.

Tímto chováním na mne framework neudělal kladný dojem, naopak ve mne vzbudil nedůvěru tvorby projektu jeho pomocí. Na základě těchto zkušeností jsem volbu Kivy zamítl.

3.5.3 Tkinter

Knihovna Tkinter je součástí jazyka Python a pro své použití nevyžaduje žádné dodatečné instalace. Díky této vlastnosti jsou okenní aplikace spustitelné tam, kde i kód Pythonu (Windows, Linux, MacOS). Pro použití stačí knihovnu ve vlastním Python kódu naimportovat. Bohužel tato knihovna nepatří k těm, které jsou aktivně rozvíjeny. To má za následek, že vzhled jednotlivých Widgetů není nijak design atraktivní, jako je tomu u knihovny PyQT. Na druhou stranu je knihovna prověřená časem, tudíž je funkční a stabilní.

Konstrukce kódu probíhá definováním hlavního okna a jeho parametrů, naplněním obsahu Widgety a jejich rozmístěním. Rozmístění jednotlivých Widgetů lze realizovat třemi způsoby. Konkrétními souřadnicemi umístění (Place). Zabalení určitého počtu řádků či sloupců v jedno podokno (Pack) a rozmisťování prvků v něm. Maticové umístění rozměru počtu řádků a sloupců (Grid). Poslední jmenovaná metoda je pro začínající programátory nejsnazší na porozumění a následnou aplikaci ve svém návrhu GUI. Posledním krokem je tlačítkům a jiným prvkům GUI definovat akce, které mají při aktivaci vykonat. Jednotlivé akce jsou řádky Python kódu.

Jakmile je okno inicializované, naplnění Widgety připravené, logické operace přiřazené, spustí se nekonečná smyčka. Tato smyčka GUI okno naplní Widgety, kde v řádu mikrosekund periodicky kontroluje, zda nastala událost, na kterou má reagovat. Takovou to událostí bývá stlačení tlačítka či callback. Pokud událost nastane, vykoná se přiřazená operace a vykreslí se aktuální podoba okna.

Pro tuto úlohu není design směrodatný, nýbrž funkčnost, proto právě tato knihovna bude zvolena pro tvorbu GUI.

3.6 Požadavky na SW zařízení

Hardware je v porovnání se softwarem okrajovou částí této práce. Z toho důvodu je tato kapitola obsáhlejší a definuje požadované vlastnosti chování zařízení jak z pohledu uživatele, který data uveřejňuje, tak z pohledu uživatele, který Elektronickou informační tabulí bude využívat jako příjemce zpráv. Právě i proto je na tuto část kladen důraz na stabilitu a spolehlivost řešení.

3.6.1 Administrace dat

- Uveřejnění dat má být uživatelům dostupné v maximální možné míře. Z toho vyplývá, že tato část zařízení musí být dostupná prostřednictvím Internetu.
- Přístup uživatelů prostřednictvím Internetu musí být zabezpečené.
- Zabezpečený přístup uživatelů musí nabízet registraci uživatelů.
- Uživatelé musí být k dispozici uveřejnění dat v běžných formátech (PNG, BMP, JPG, PDF)

- Uživatelé musí být možno nahraným souborům individuálně určit datum, OD kterého mají být data uveřejněna.
- Uživatelé musí být možno nahraným souborům individuálně určit datum, DO kterého mají být data uveřejněna.
- Uživatelé musí být umožněno svá data kategorizovat.
- Uživatel musí mít možnost v uložených datech vyhledávat.
- Uživatel musí mít možnost data smazat.
- Přehledný a intuitivní design.
- Přístup administrátora do administrace dat.

3.6.2 Presentace dat

- Raspberry Pi musí disponovat LCD zařízením, které nemá pevně definovaný rozměr.
- Zařízení musí být schopné přizpůsobit svůj obsah formátu použitého LCD.
- Raspberry Pi musí disponovat rozhraním stroj-člověk.
- Ovládání rozhraním stroj-člověk musí být intuitivní.
- Uživatel musí mít možnost procházet jednotlivé kategorie dat.
- Uživatel musí mít možnost procházet data v jednotlivých kategoriích.
- Presentovaná data musí být co nejaktuálnější!
- Přívětivý a přehledný design prezentační aplikace.

3.7 Návrh programového vybavení

Programové vybavení se bude skládat z nástrojů, které byly představeny v této kapitole 3. Použití této skladby programového vybavení bude cíleno na realizaci dvou rozhraní. Webového a Prezentačního. Tato rozhraní se protnou v podobě dat, která budou uložena přímo v zařízení a jsou určena k presentaci. Záznamy o těchto datech budou uloženy v MySQL databázi, kde bude možné na jedné straně zapisovat a na straně druhé vyčítat, název souboru, OD kdy má být soubor zveřejněný a DO kdy má být soubor zveřejněný. Právo zápisu či smazání záznamu bude pouze na administrační části zařízení.

3.7.1 Návrh Webového rozhraní

Návrh webového rozhraní je založen na použití frameworku Django, u kterého bude využitý integrovaný mechanismus administrace.

První aplikace projektu bude využívat existující komponenty autentizačního systému Django - login, logout, authenticate, login_require a model User. Všechny tyto nástroje budou interpretovány vlastní podobou HTML stránek, které budou v projektu Django odděleny od druhé aplikace obsahu webu. Obsah webu bude zprostředkovávat

bezplatným webserverem Apache 2 jak na lokální na IP adrese, tak i veřejné statické adrese přípojky Internetu. Obsah na veřejné přihlašovací stránce bude následující:

- login
- registrace

Druhá aplikace projektu bude soustředěna na obsah jednotlivých záložek, které budou uživateli nebo administrátorovi dostupné po přihlášení. Bude se jednat o kategorie:

- Úvodní list – všeobecné informace pro uživatele, co od prostředí může očekávat a jak s ním má zacházet
- Moje údaje – prostor pro změnu registračních údajů
- Moje soubory – prostor, kde bude možné založit událost (zveřejnit/smazat dokument)
- Přehled uživatelů – dostupné pouze administrátorovi
- Odhlášení

Webové rozhraní bude disponovat třemi databázovými tabulkami, které komplexně pokryjí chod celého zařízení. Všechny tabulky mají jednu společnou vlastnost. Majitelem kteréhokoliv souboru je právě jeden uživatel! Pokud uživatel smaže nějaký záznam, promítne se tato akce do všech příslušných tabulek. V této části bude využit databázový systém MariaDB.

DB tabulka auth_user:

Bude evidovat registrované uživatele a jejich údaje. Jedná se o tabulku, kterou již autentizační systém Django disponuje ve svém základu.

DB tabulka stranky_materialy:

Bude evidovat jednotlivé soubory, které jednotliví uživatelé nahrají do systému prostřednictvím kategorie Moje soubory. Data se budou fyzicky nacházet na lokálním disku Raspberry Pi.

DB tabulka stranky_gui:

Jelikož Tkinter neumí zobrazovat PDF soubory, musí webové rozhraní tyto soubory rozložit na formát JPG. Tento formát lze následně snadno zobrazit. Tím se dostáváme před situaci, kde uživatel bude moci nahrát PDF s více stranami. To bude mít za následek rozklad na počet JPG souborů, dle počtu stran PDF. Za tímto účelem bude tabulka **stranky_gui** evidovat přesný seznam souborů pro konečnou prezentaci.

Jakmile budou takto data připravená data k prezentaci, musíme zkontrolovat jejich platnost ve smyslu OD – DO.

3.7.2 Návrh Prezentčního rozhraní

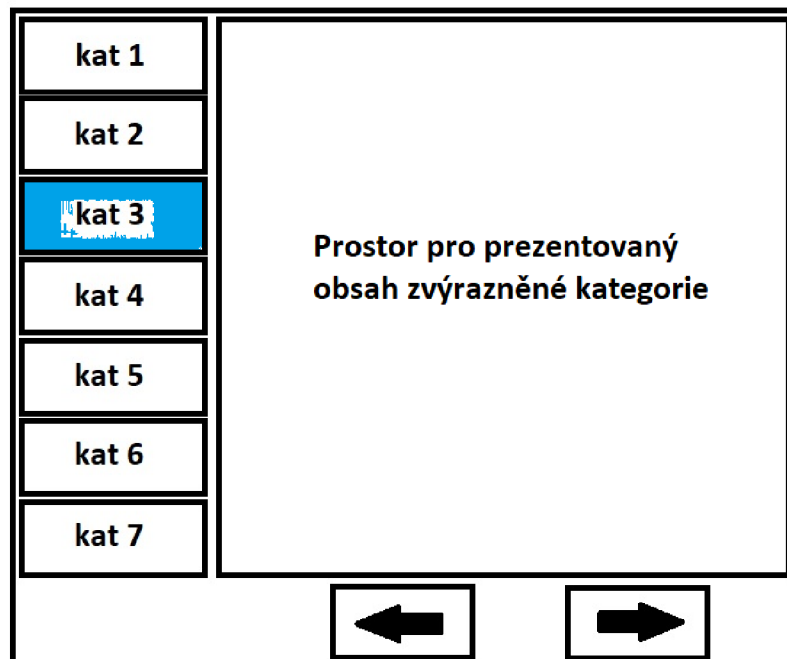
Prezentční rozhraní, alias GUI, je napsáno s využitím knihovny Tkinter. Zde bude nutné navrhnout layout Widgetů tak, aby bylo zachováno maximální možné místo na obrazovce pro zobrazení dat. Tento ohled splňuje rozmístění tlačítek jednotlivých kategorií do sloupce, jelikož běžná instalace obrazovek je na šířku. Pro případ, že instalace obrazovky bude na výšku, program GUI nesmí vytvořit tlačítka kategorií zbytečně široká. Ovládacím prvkem člověk-stroj je navržený Joystick.

S ohledem na sloupcové řazení kategorií, jeví se jako intuitivní určit pohybům Joysticku vzhůru a dolů procházení kategoriemi. Zbývající pohyby do prava a do leva se intuitivně nabízejí pro funkci listování materiálu v dané kategorii. Takto rozvržené GUI si musí vyčítat data z místního úložiště k prezentaci prostřednictvím tabulky `stranky_gui`.

Vyvolání obsahu dané kategorie bude probíhat stiskem Joysticku, respektive jeho mikropínače. Takto zvolená kategorie musí být uživateli dána srozumitelně najevo. Proto tento návrh bude počítat s barevným podbarvením tlačítka dané (aktivní) kategorie.

Přemítání obsahu pohyby Joysticku do prava či do leva bude vždy přemítat pouze obsah dané (zvýrazněné) kategorie.

Jelikož se jedná o elektronickou nástěnku, je vhodné, aby se obsah nějakým způsobem podbízel kolemjdoucím potencionálním čtenářům. Pro tento účel bude zařízení automaticky obsah přemítat po definované době nečinnosti. Nečinností je myšlen čas od posledního pohybu Joystickem uživatelem (čtenářem).



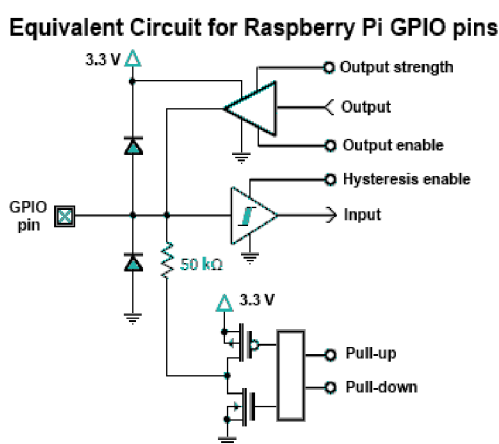
Obrázek 3-3: Návrh GUI Prezentční aplikace.

4. REALIZACE ZAŘÍZENÍ

Realizace zařízení vychází z poznatků nabytých při průzkumu profesionálních řešení na trhu, z průzkumu programového vybavení Raspberry Pi, z výběru ovládací periférie člověk-stroj a průzkumu softwarových nástrojů, dostupných pro platformu Linux.

4.1 Raspberry Pi 3 Model B

Pro realizaci zapojení je využito GPIO periférie (general-purpose input/output). Tato periférie disponuje 40 piny a jejich využití je různorodé: UART, SPI, I2C, $U_{CC} = 3,3V$ $U_{DD} = 5V$ dle legendy na obrázku pod textem (Obrázek 4-1).



Obrázek 4-1: Ekvivalentní zapojení GPIO pinu, převzato z [9].

Důležitým faktorem je, že výstupní logika jednotlivých pinů je 3,3V s maximálním zatížením jednoho pinu 16 mA. Veškeré piny GPIO jsou rozděleny do 3 bank, přičemž každá banka GPIO pinů nesmí překročit celkový proud 51mA [11][12].

Banka 1 – piny 0 až 27

Banka 2 – piny 28 až 45

Banka 3 – piny 29 až 40

Logické „0“ signál nabývá od 0 ÷ 0,8 V

Logické „1“ signál nabývá od 1,3V ÷ 3,3V

4.1.1 HDMI

Základním obrazovým výstupem je standardní HDMI konektor s podporou FullHD (1080p 30fps). Prostřednictvím toho konektoru lze sdílet i digitální zvuk ze zařízení.

4.1.2 Wi-Fi / Ethernet

Konektivita k síti Internet je zajištěna Ethernetovým modulem s rychlostí 10/100Mbit/s.

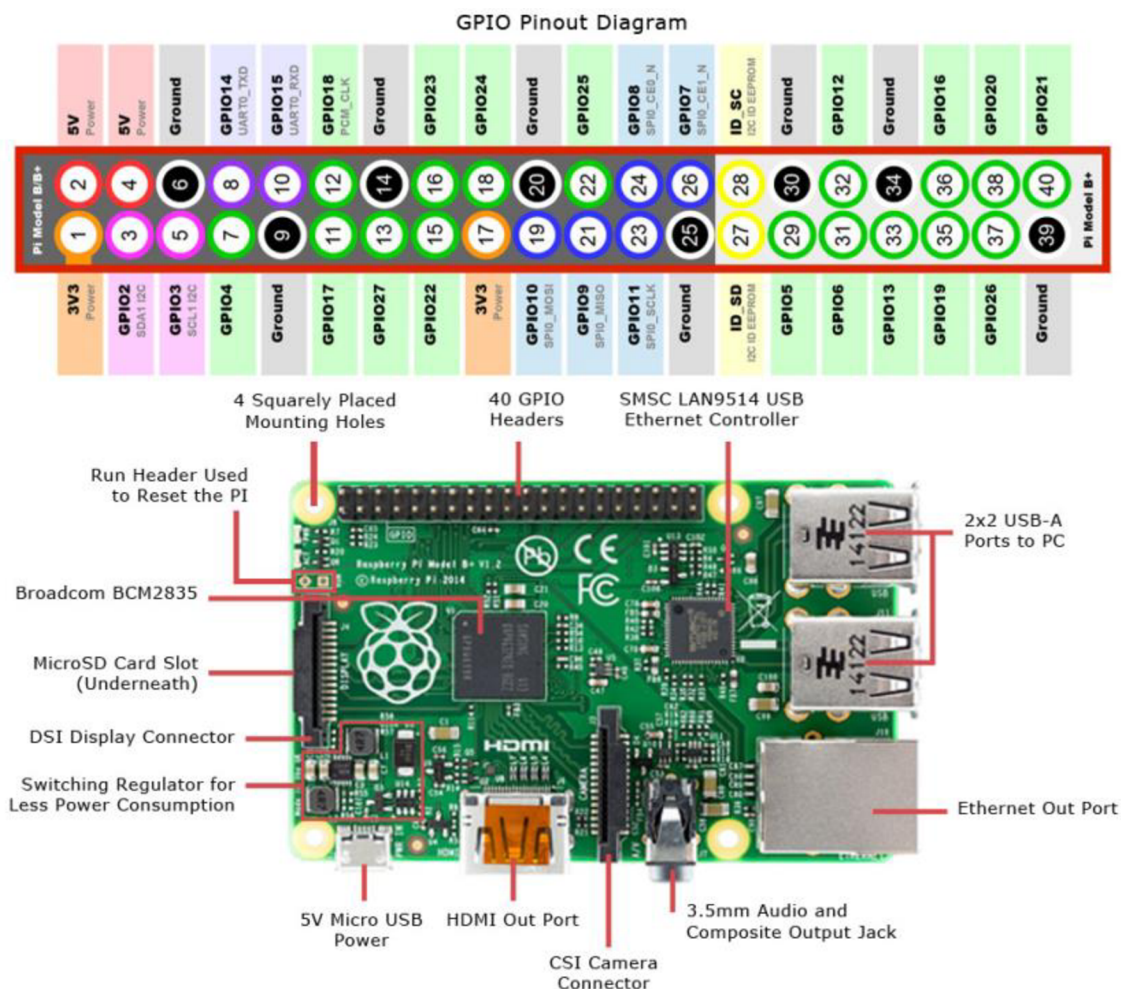
Připojení k bezdrátové síti je prostřednictvím Wi-Fi modulu, který má přímo na desce integrovanou anténu, u které je nutné mít na paměti, že její zisk bude značně omezený oproti běžným Wi-Fi zařízením. V blízkosti routru s přímou viditelností lze dosáhnout rychlosti připojení až 150Mbit/s při standardech 802.11 b/g/n.

Obě dvě periférie jsou po instalaci systému dostupné a plně funkční. Právě Wi-Fi je již během instalace Raspbianu uživateli v případě dostupnosti nabízena pro přihlášení a zadání hesla.

4.1.3 Diskový prostor

Pro zachování univerzálnosti, Raspberry Pi nedisponuje vlastním paměťovým prostorem pro instalaci operačního systému ani jiného softwaru. Za tímto účelem je vybaveno MicroSD slotem, do kterého si uživatel sám dle svých potřeb zakoupí cenově dostupnou paměťovou kartu. Dle doporučení výrobce je vhodné použít kartu Class 10 s minimální kapacitou 8 GB.

Na toto paměťové médium je nahrán, jak operační systém Raspbian, Django, MariaDB, Apache 2, Python 3 a další..., tak je využito i jako úložiště pro uživateli nahraných souborů.



Obrázek 4-2: Popis periférií Raspberry PI 3 model B, převzato z [10].

4.2 HW pro interakci člověk-stroj

V podkapitole 1.2.5 byl učiněn průzkum periférií, které jsou dostupné pro použití v této práci. Závěrem tohoto průzkumu byl návrh použití Joysticku, analogově digitálního převodníku a signalizačních LED diod jako hardwaru člověk-stroj. Tento návrh byl ctěn a realizován v plném rozsahu návrhu, doplněné o Raspberry Pi 3 Model B.

4.2.1 Signalizační LED diody

Všechny poznatky o GPIO byly při realizaci vzaty na zřetel. Požitím správných předřadných odporů LED diod a chováním kódu Joysticku tak, že v jeden okamžik může svítit pouze jedna LED dioda. Tím je dosaženo zapojení LED diod pro signalizaci pohybu LEVO, PRAVO, NAHORU, DOLU, STISK bez použití tranzistorového pole, které by sloužilo jako ochrana proti proudovému přetížení GPIO výstupů.

Při použití výstupních pinů GPIO je v kódu naimportována patřičná knihovna import RPi.GPIO as GPIO, která definuje, jaké číslování pinů bude použito. Na výběr je GPIO.BOARD, které určuje číslování pinů tak, jak jsou uspořádány na desce plošného spoje Raspberry Pi. Výběr GPIO.BCM odkazuje na číslování chipu „Broadcom - System On Chip“.

Při realizaci byly výstupní piny GPIO pro LED definovány:

- GPIO.setmode(GPIO.BCM)
- self.stisk = 26
- self.nahoru = 5
- self.dolu = 13
- self.doprava = 19
- self.doleva = 6
- GPIO.setup(self.stisk, GPIO.OUT)
- GPIO.setup(self.nahoru, GPIO.OUT)
- GPIO.setup(self.dolu, GPIO.OUT)
- GPIO.setup(self.doprava, GPIO.OUT)
- GPIO.setup(self.doleva, GPIO.OUT)

Pro jednotlivé směry byly použity zelené LED diody s úbytkem napětí na diodě 2,2V při 15mA. Takto určeným LED diodám je nutné vybrat předřadný rezistor minimální hodnoty dle rovnice (4.1):

$$R_{zelené\ diody} = \frac{U_{CC} - U_{diody}}{I_{diodou}} = \frac{3,3 - 2,2}{0,015} \doteq 67\ \Omega \quad (4.1)$$

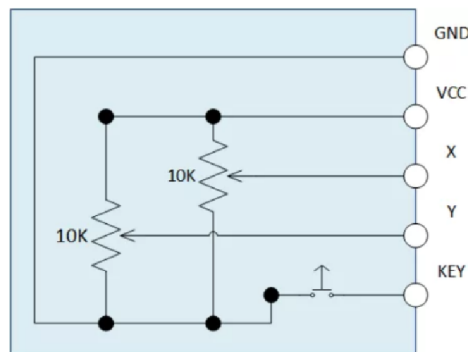
Pro signalizaci stisku bude použita modrozelená LED dioda s úbytkem napětí 3,2V při 15mA. Takto určené LED diodě je nutné vybrat předřadný rezistor minimální hodnoty dle rovnice (4.2):

$$R_{diody\ stisku} = \frac{U_{CC} - U_{diody}}{I_{diodou}} = \frac{3,3 - 3,2}{0,015} \doteq 7\ \Omega \quad (4.2)$$

4.2.2 Stisk Joysticku

Dle schéma vnitřního zapojení je patrné, že tlačítko pro stisk je svým aktivováním přizemněno. Proto je nutné použít softwarově aktivovaného Pull-up rezistoru a tímto držet signál z neaktivního tlačítka v logické „1“. Stiskem tlačítka je tento signál přizemněn a změněn na logickou „0“. Následně je docíleno vyčtení stavu. Pro docílení toho chování je v programu inicializovaný příslušný pin (tlacitko = 24) a Pull-up rezistor pomocí `GPIO.setup(self.tlacitko, GPIO.IN, pull_up_down = GPIO.PUD_UP)`.

Při stisku kloboučku Joysticku, je odeslána směrem do GUI číselná hodnota stavu Joysticku a zároveň je rozsvícena LED dioda stisku na 0,75 sekundy.



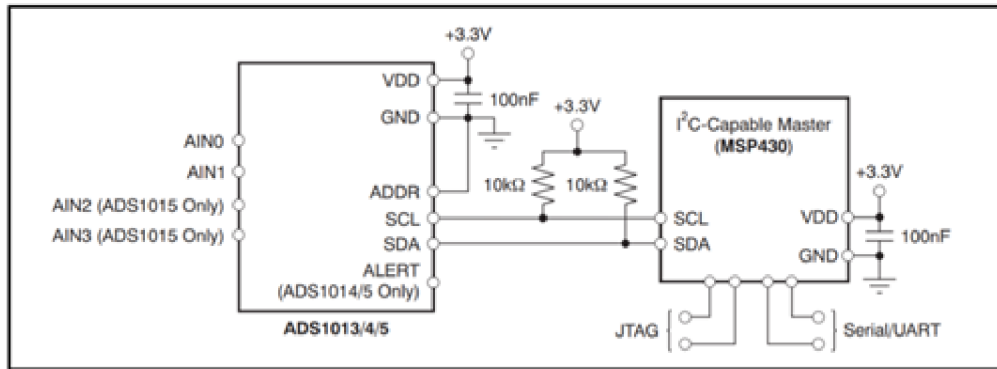
Obrázek 4-3: Vnitřní zapojení Joysticku KY-023, převzato z [14].

4.2.3 Pohyb Joysticku v osách X & Y

Joystick je analogové zařízení, proto bylo zapojení doplněno analogové-digitálním převodníkem ADS1015, u kterého byly využity kanály 0 a 1. Použití tohoto převodníku předchází importování knihovny `Adafruit_ADS1x15` a následné inicializace `Adafruit_ADS1x15.ADS1015()`. Vyčtení hodnot jednotlivých kanálů probíhá funkcí `adc.read_adc(0,1)`, kde první parametr volané funkce je číslo kanálu, druhý parametr je zesílení měřeného signálu. Zesílení je nastaveno na hodnotu 1x, tudíž měříme signál bez zesílení. Pokud jsou měřeny slabší signály, je možné zesílení nastavit až na hodnotu 16x. V případě, že rychlost vzorkování není konkrétně definována, je defaultně nastavena 1600 vzorků/sec.

Napětí na jednotlivých děličích os je $U_{cc} = 3,3V$ přivedené přímo z Raspberry Pi. Vyčítání probíhá na kanálech AIN0 a AIN1 převodníku. Takto naměřené hodnoty jsou periodicky vyčítány prostřednictvím I2C sběrnice z AD převodníku a uloženy do programových proměnných.

Dle základního zapojení z datasheetu výrobce jsou v obvodě fyzicky použity Pull-up rezistory hodnoty $10k\Omega$ pro zajištění I2C komunikace na piny SCL a SDA. Poslední doplnění spočívá v použití oddělovacího kondenzátoru mezi U_{cc} a GND s hodnotou 100nF. Zajímavostí z datasheetu je, že lze adresací připojit celkem čtyři tyto převodníky (vyčítat až 16 analogových signálů). Vše dle obrázku pod textem (Obrázek 4-4).



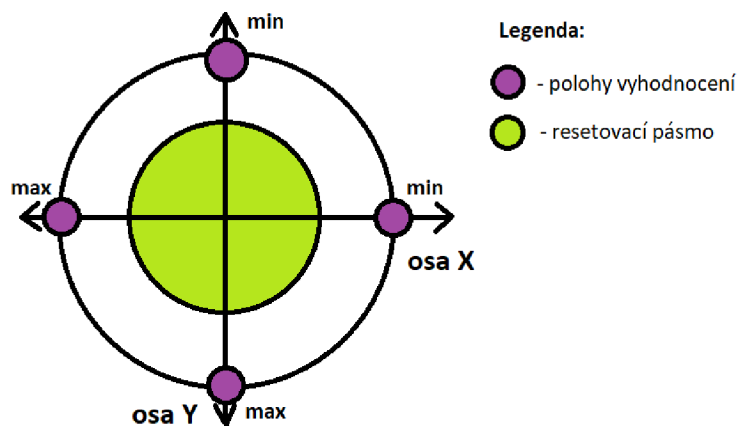
Obrázek 4-4: Základní zapojení ADS1015, převzato z [22].

Jednotlivé pohyby Joysticku musely být programově zvlášť ošetřeny ze dvou důvodů:

- v jeden okamžik nesmí svítit více než jedna LED dioda, jako ochrana proti překročení maximálního proudu GPIO
- vyhodnocení pohybů v jednotlivých osách musí mít pevné časové rozestupy jako ochranu proti zahlcení GUI příkazy od Joysticku

Poloha Joysticku je periodicky kontrolována nekonečnou smyčkou, která je vykonávána samostatným vláknem. Tato myčka periodicky vyčítá polohu v každé ose, která je reprezentována číselnou hodnotou. Program na vyhodnocení stavů Joysticku je skupina podmínek, které při jejich splnění reportují z vlákna směrem do GUI číselnou hodnotu stavu Joysticku a zároveň aktivují příslušnou LED diodu na 0,75 sekundy. Během tohoto času smyčka vlákna nevykonává další instrukce. Pokud by uživatel držel Joystick stále vychýlen ve stejné poloze, docházelo by po 0,75 sekundě k opětovnému vyhodnocení podmínky pro daný směr. Opět by došlo informování GUI o poloze Joysticku, opět by se rozsvítila LED dioda.

Toto chování je nežádoucí, a proto bylo programově ošetřeno tvorbou pomocné pomocných proměnných paměti předchozího stavu `memory_X` a `memory_Y`. Ve výchozím stavu tyto proměnné nabývají hodnoty „0“. Na obrázku pod textem (Obrázek 4-5) je znázorněno resetovací pásmo hodnot jednotlivých os, které tyto proměnné uvádí do stavu „0“. Pokud je splněna podmínka pro vyhodnocení vychýlení Joysticku jistým směrem a pomocná proměnná nabývá hodnoty „0“, je vyhodnocen pohyb v daném směru: dojde k informování GUI o poloze Joysticku, bude rozsvícena LED dioda na 0,75 sec a pomocná proměnná nabyde hodnoty „1“. Tím je dosaženo pouze jedno vyhodnocení směru. Pro opětovné vyhodnocení směru se musí uživatel vrátit do výchozí polohy Joysticku. Tím projde resetovacím pásmem, které pomocné proměnné uvolní.



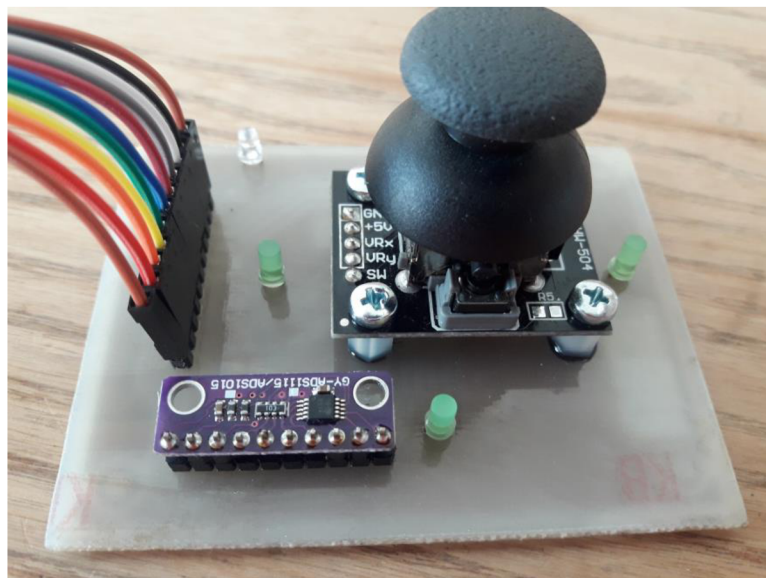
Obrázek 4-5: Programové oblasti činnosti Joysticku.

Jelikož má páka Joysticku délku cca 1,5 cm, je Joystick velice citlivý. Bylo nutné vytvořit podmínky, které vyhodnotí daný směr, pouze pokud hodnota v kolmém směru nenabývá hodnot blízkých minima či maxima.

4.2.4 Deska plošného spoje

Deska plošného spoje je realizována z jednovrstvé měděné desky, která je opatřena třemi řadami pinů pro připojení jednotlivých periférií. DPS byla vytvořena dle návrhu zapojení 2.4 za pomoci programu Autodesk Eagle.

- 1 x 11 – sloužící pro připojení k Raspberry Pi
- 1 x 5 – slouží k připojení Joysticku
- 1 x 10 – slouží k připojení AD převodníku ADS1015
- 4 x – zelené LED pro indikaci zaznamenaná směru pohybu
- 1 x – modrozelená LED pro indikaci stisku Joysticku
- 4 x – distanční sloupky



Obrázek 4-6: Realizovaná periferie člověk-stroj.

4.3 Programové vybavení

Programové vybavení Elektronické nástěnky LCD bylo realizováno za pomoci nástrojů Raspbian, Python, Django, Apache 2, MariaDB, Bootstrap, Celery, Tkinter, Threading a Python MySQL connector.

Návod na instalaci operačního systému Raspbian je dostupný na oficiálních stránkách Raspberry Foundation, proto této části práce nebude věnována vlastní podkapitola. Skladba výše jmenovaných nástrojů byla použita při řešení úlohy ve dvou hlavních částech (Webové rozhraní a Prezentační GUI).

4.3.1 Webové rozhraní

Důvodem realizace webového rozhraní je fakt, že v takovémto případě jsou kladeny výpočetní nároky pouze na zařízení, na kterém jsou webové stránky provozovány. Další předností je, že takto realizovaný přístup uživatelů je nezávislý na Prezentační části. Dostupnost administrace zařízení je takto dosažitelná z libovolného osobního počítače bez ohledu na operační systém, tak i z mobilního telefonu všech platforem.

4.3.1.1 Webserver

Webový framework Django disponuje vlastním webserverem, o kterém sami autoři na webových stránkách uvádějí, že je vhodný pro vývoj aplikací. Po jeho spuštění příkazem: `python manage.py runserver`, je dostupný na adrese: `http://127.0.0.1:8000/`.

Pro ostrý provoz doporučují použití jiného řešení. Například použití webserveru Apache 2. Tento webserver je distribuován bezplatně a je stále rozvíjen. Apache 2 je distribuce, na které je provozováno přes 50% všech serverů na celém světě. Pro jeho nasazení je nutné nakonfigurovat komunikační protokol WSGI mezi webserverem a aplikací pomocí drobné úpravy konfiguračního souboru `httpd.conf` serveru Apache 2 dle instrukcí projektu Django. Tímto souborem je definováno, kde se nacházejí složky souborů MEDIA a STATIC Django projektu, zároveň je nutné nakonfigurovat cestu do projektu Djanga a cestu ke konfiguračnímu souboru WSGI v projektu Djanga.

Jakmile je vše nastavené, zobrazí se úvodní stránka Django projektu na adrese `http://127.0.0.1/` nebo zadáním `localhost` do adresového řádku prohlížeče. Tím je vše ověřené a je možné se věnovat vývoji vlastnímu obsahu webu v Djangu.

Základní příkazy pro práci s Apachem 2 jsou:

- `/etc/init.d/apache2 start`
- `/etc/init.d/apache2 restart`
- `/etc/init.d/apache2 stop`

4.3.1.2 Django struktura

Pro řešení úlovy byl vytvořen projekt s názvem **DJproject**. Struktura obsahu webu je v projektu rozdělena do dvou částí (aplikací), a to **logg** a **stranky**. Tím bylo dosaženo logického rozdělení souborů obsahu projektu. Při realizaci velkých projektů, jako je E-shop, uživatel ocení možnost dělení obsahu do jednotlivých šuplíčků (do aplikací), ve kterých cíleně řeší pouze dílčí problematiku webu.

Každý projekt je založen příkazem, na jehož konci uvádíme požadovaný název projektu: **django-admin startproject DJproject**.

Na disku se vytvoří následující základní souborová struktura:

DJproject/

manage.py – soubor pro vykonávání operací nad projektem, jako jsou založení aplikace, migrace databáze a kolektivizace statických souborů

DJproject /

__init__.py

settings.py – hlavní konfigurační soubor projektu

urls.py – soubor s URL schématem webu, odkazující na části aplikací

wsgi.py – konfigurační soubor komunikačního protokolu

Jak již bylo zmíněno, obsah webu je rozdělen do dvou aplikací **logg** a **stranky**. Jednotlivé aplikace se zakládají příkazem, na jehož konci uvádíme požadovaný název aplikace: **python manage.py logg** či **python manage.py stranky**.

Na disku se rozšíří souborová struktura v hlavním adresáři projektu o adresáře vytvořených aplikací:

DJproject /

DJproject /

__init__.py

settings.py

urls.py

wsgi.py

stranky /

__init__.py

admin.py – administrace dané aplikace

apps.py – konfigurace dané aplikace

models.py – správa modelů databáze dané aplikace

tests.py – testování aplikace

urls.py – tento soubor, obsahující URL schéma dané aplikace je nutné vytvořit ručně

logg /

identický obsah

(**__init__.py**; **admin.py**; **apps.py**; **models.py**; **tests.py**; **urls.py**)

Tuto adresářovou strukturu je nutné ručně doplnit o složky MEDIA a STATIC. Složku STATIC je možné umístit jak přímo do hlavního adresáře projektu, tak i vytvořit jako podadresář v jednotlivých aplikacích. V tomto adresáři programátor ukládá obsah, určený pro statický obsah webu obrázky: tapety pozadí, ikona záhlaví webu a další. Realizace podadresáře v aplikaci je dosaženo logického rozdělení souborů příslušných daným aplikacím.

Úloha adresáře MEDIA je sloužit jako adresářový prostor pro data nahraná na webserver prostřednictvím webu. Její umístění je jako podadresář hlavní složky projektu.

Takto připravenou strukturu je nutné definovat do konfiguračních souborů Django projektu. Důležitým krokem je v souboru settings.py registrovat:

- všechny aplikace v části INSTALLED_APPS, které byly vytvořeny
- složku STATIC v části STATIC_URL a STATIC_ROOT
- složku MEDIA v části MEDIA_URL a MEDIA_ROOT

4.3.1.3 Přihlášení / Registrace

První aplikace v projektu Django se nazývá **logg**. Její úlohou je zajistit bezpečné přihlášení či registraci uživatelů do obsahu webu. Za tímto účelem jsou využity integrované mechanismy - **login**, **logout**, **authenticate**, **login_require**, model **User** a **auth** aplikaci.

Každá aplikace je koncipována dle architektury MVC, kde je oddělena logika od výstupu. Proto je nutné ručně vytvořit templates v každé aplikaci, kde se nachází HTML obsah. Logika aplikace se nachází v souboru views.py, který bylo také nutné ručně vytvořit.

V souborech urls.py jsou definované URL adresy:

localhost/uzivetelske_ucty/ - této URL byla přiřazena (**django.contrib.auth.urls**) aplikace **auth**, díky které lze využít integrované view pro: **login**, **logout**, **password_change**, **password_reset** a další.

localhost/uzivetelske_ucty/login/ - integrované view, které vyhledá login.html s obsahem dané stránky. Obsahem této stránky je zadání uživatelského jména a hesla. Django mechanismus **auth** provede zabezpečené přihlášení. Obsah, na který má být uživatel přesměrován po přihlášení, je dán zápisem v konfiguračním souboru settings.py **LOGIN_REDIRECT_URL = 'obsah_po_prihlaseni'**. Po přihlášení je uživatel přesměrován na Úvodní list.

Elektronická informační tabule LCD

Přihlášení Registrace

Uživatelské jméno

Heslo

Přihlášení

Obrázek 4-7: Pohled na přihlášení.

`localhost/uzivetelske_ucty/registrace/` - volá **signup** metodu ve views, která vyvolá obsah `signup.html`, jehož obsahem je registrační formulář. Formulář je následně uložen prostřednictvím views a modelu **User**, pokud je správně vyplněn po odeslání uživatelem tlačítkem Zaregistrovat se.

Elektronická informační tabule LCD

Přihlášení Registrace

Uživatelské jméno

Křestní jméno

Příjmení

E-mail

Vaše heslo (poprvé)

Vaše heslo (podruhé)

Zaregistrovat se

Obrázek 4-8: Pohled na registraci.

Po úspěšné registraci je uživatel informován o úspěšném registrování a je vyzván k přihlášení.

Takto registrovaní uživatelé jsou zapsáni do databázové tabulky **auth_user** dle modelu **User**, kterou integrovaná aplikace **auth** disponuje. Primárním klíčem této tabulky je unikátní ID záznamu tabulky, které se pouze inkrementuje.

Atributy tabulky `auth_user` jsou:

ID	unikátní pořadové číslo uživatele
password	uživatelské heslo v podobě otisku 78 znaků (hash)
last_login	záznam o posledním přihlášení
is_superuser	příznak, že uživatel je superuser
username	uživatelské jméno
first_name	uživatelské křestní jméno
last_name	uživatelské příjmení
email	uživatelská emailová adresa
is_staff	příznak, že uživatel je administrátor
is_active	příznak, kterým se blokuje uživatelský účet
date_joined	datum a čas zaregistrování uživatele do systému

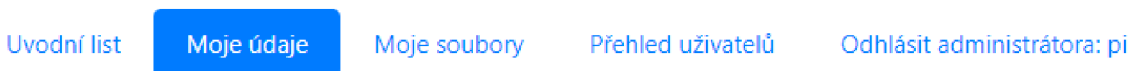
4.3.1.4 Úvod



Obrázek 4-9: Lišta záložek Webového rozhraní - Úvodní list.

Smyslem této úvodní stránky po přihlášení uživatele je informovat ho, k čemu Elektronická informační tabule LCD slouží a k čemu toto webové rozhraní slouží. Dále je informován o typech souborů, které lze do systému nahrát a jakým způsobem budou data uveřejněna. Systém je nastaven tak, že po překročení data, do kterého má být soubor zveřejněn, bude záznam smazán.

4.3.1.5 Moje údaje



Obrázek 4-10: Lišta záložek Webového rozhraní - Moje údaje.

Účelem této záložky je změna osobních informací uživatele, které uvedl během registrace. Konkrétně se jedná o uživatelské jméno, křestní jméno, příjmení, email, heslo, a zda je účet aktivní.

Tento mechanismus zapisuje do tabulky `auth_user` dle modelu `User`, za použití definovaného formuláře v `forms.py` a importované funkce `UserChangeForm` z aplikace `auth`.

4.3.1.6 Moje soubory

[Uvodní list](#)[Moje údaje](#)[Moje soubory](#)[Přehled uživatelů](#)[Odhlásit administrátora: pi](#)


Obrázek 4-11: Lišta záložek Webového rozhraní - Moje soubory.

Nejdůležitější část webového rozhraní, na kterou byl kladen největší důraz na funkcionalitu (Obrázek 4-12). Z uživatelského hlediska se skládá z možností Založit událost; vyhledání požadovaného názvu a tabulky, ve které jsou všechny záznamy buď daného uživatele, či všech uživatelů, pokud je přihlášený administrátor.

Přehled událostí v El. nástěnce

[Založit událost](#)

Zde zadej text, který v tabulce hledáš.

Název události	Zveřejnit OD	Zveřejnit DO	Kategorie	soubor	Vlastník	Smazat?
první	25. března 2019	25. prosince 2019	Obecné	uzivatel_1/777.pdf	pi	

Obrázek 4-12: Vyhledávání v přehledu založených Událostí.

Založení události probíhá danou volbou Založit událost. Tím se uživateli objeví nová stránka, která se skládá z nezbytných údajů, nutných pro uveřejnění dokumentu. Jedná se o: Název události, OD a DO kdy má být událost uveřejněna, v jaké kategorii a výběr souboru. Tyto údaje je možné zadávat, jelikož byl za tímto účelem vytvořen model s názvem **Materialy** v aplikaci **Stranky**, který obsahuje jednotlivé položky tohoto formuláře s názvem **FormNaSoubory**. Právě tento formulář v sobě skrývá nabídku kategorií, do které uživatel může kategorizovat prezentovaný materiál. Za účelem umístění Elektronické informační tabule LCD byly vytvořeny kategorie:

- Obecné
- Zkoušky
- Jídelníček
- Pro studenty
- Pro učitele
- Brigády
- Školní akce

Pokud by bylo potřeba kategorie pozměnit či rozšířit, je nutné rozšířit právě tuto nabídku (choices) právě v tomto formuláři.

Založení události do El. nástěnky

Název události:

OD kdy má být událost zveřejněna?:

DO kdy má být událost zveřejněna?:

Vyber, do které **kategorie** událost patří:

Nahraj soubor pro zveřejnění:
 Soubor nevybrán

Obrázek 4-13: Formulář pro založení Události k zveřejnění.

Po stisku Nahrát soubor se uživatelem zadané informace a soubor zpracuje. Je tím myšleno, že data v této podobě by nebyla použitelná pro Prezentační část zařízení, proto bylo nutné model i view rozšířit o funkce, které se vykonají s nahráním souboru do zařízení.

V Prezentační části není možné zobrazovat formáty PDF napřímo. Proto bylo využito nástroje pro rozklad formátu PDF na formáty JPG dle počtu stránek.

Za tímto účelem bylo view, které má na starosti naplnění formuláře a jeho kontrolu správnosti vyplněných polí, doplněno o kontrolu posledních 3 znaků z názvu souboru na PDF. V případě, že je vyhodnocena shoda s texty PDF či pdf, je využito nástroje **pdf2image** k rozkladu souboru na JPG formáty. Pokud není vyhodnocen formát PDF či pdf, dojde k uložení souboru bez jakýchkoli úprav.

Pro tuto identifikaci PDF formátu a následný rozklad na JPG bylo nutné rozšířit model **Materialy** o atributy, které budou reprezentovat dodatečnou identifikaci vlastností nahraného souboru:

je_to_pdf – příznak, který v databázové tabulce pro kontrolu identifikuje PDF

pocet_jpg – příznak, který informuje o počtu vygenerovaných stránek JPG

nazev_jpg – příznak, který informuje o názvu JPG

Pokud je nahrán soubor JPG, je jeho název v tabulce uložen bez koncovky. Pokud se jedná o soubor PDF, je jeho název JPG v tabulce uložen s podtržítkem bez koncovky, dle obrázku pod textem (Obrázek 4-14).

id	name	od_kdy	do_kdy	kategorie	vyber_souboru	user_id	je_to_pdf	pocet_jpg	nazev_jpg
15	sdsd	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor.jpg	1	0	1	soubor
16	dssfsd	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor.pdf	1	1	3	soubor_

Obrázek 4-14: Obsah DB tabulky stranky_materialy se dvěma záznamy.

Uložení souboru bylo vypracováno do přehledného adresářového systému ukládání. Za tímto účelem byl model **Materialy** doplněn funkcí, která zidentifikuje ID právě přihlášeného uživatele a tímto ID doplní název „**uživatel_ID**“ složky, do které jsou konkrétního uživatele materiály nahrávány. Tím je dosaženo identifikace uložených materiálů dle uživatelů. Jak již bylo zmíněno, pro ukládání materiálů prostřednictvím webu, je určena složka MEDIA. Pro příklad uvádím cestu k souborům, nahraným administrátorem:

DJproject/media/uzivatel_1/

V případě potřeby je možné zaměnit doplňované ID za uživatelské jméno či jiný uživatelův registrovaný údaj. Ale pozor, musí se jednat o údaj, který je nezaměnitelný!

Pokud uživatel nahraje druhý soubor se stejným názvem, Django se s tímto vypořádá přidáním podtržítka k názvu souboru a sedmi náhodně vygenerovaných znakům či číslic na konec názvu souboru dle obrázku pod textem (Obrázek 4-15).

id	name	od_kdy	do_kdy	kategorie	vyber_souboru	user_id	je_to_pdf	pocet_jpg	nazev_jpg
15	sdsd	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor.jpg	1	0	1	soubor
16	dssfsd	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor.pdf	1	1	3	soubor
17	fdfds	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor_WYgWVaY.pdf	1	1	3	soubor_WYgWVaY

Obrázek 4-15: Obsah DB tabulky stranky_materialy se třemi záznamy.

Jak již bylo zmíněno, data se nacházejí fyzicky na lokálním uložišti MEDIA v přehledných složkách. V případě PDF souborů jsou do stejné složky, kam je soubor PDF nahráván, vygenerovány JPG soubory nové s originálním názvem souboru, doplněného o podtržítka a pořadové číslo stránky původního souboru PDF. Např.:

uzivatel_1/soubor.pdf	- 1. nahraný soubor
uzivatel_1/soubor_1.jpg	- vygenerovaný 1. list JPG
uzivatel_1/soubor_2.jpg	- vygenerovaný 2. list JPG
uzivatel_1/soubor_3.jpg	- vygenerovaný 3. list JPG
uzivatel_1/ soubor_3A3xWmB.pdf	- 2. nahraný soubor (identický)
uzivatel_1/ soubor_3A3xWmB_1.jpg	- vygenerovaný 1. list JPG
uzivatel_1/ soubor_3A3xWmB_2.jpg	- vygenerovaný 2. list JPG
uzivatel_1/ soubor_3A3xWmB_3.jpg	- vygenerovaný 3. list JPG

Všechny předešlé kroky slouží k vytvoření záznamů do nové databázové tabulky s názvem **stranky_gui**, vytvořené modelem GUI v aplikaci **stranky**. V této tabulce se nacházejí záznamy tvořící seznam jednotlivých cest pouze ke všem JPG souborům určeným k prezentaci v Prezentační části. Pouze s touto tabulkou Prezentační část pracuje. Na obrázku pod textem (Obrázek 4-16) je nahrán soubor.jpg, soubor.pdf a opět soubor.pdf

id	od	do	kateg	naz	materialy_id
28	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor.jpg	15
29	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor_1.jpg	16
30	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor_2.jpg	16
31	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor_3.jpg	16
32	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor_WYgWYaY_1.jpg	17
33	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor_WYgWYaY_2.jpg	17
34	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor_WYgWYaY_3.jpg	17

Obrázek 4-16: Obsah DB tabulky stranky_gui.

V administrační části tedy existují celkem 3 databázové tabulky, které mají za úkol shromáždit či zpracovat data o uživateli a jimi nahranými soubory.

Tabulka **auth_user** – slouží k evidenci uživatelů a jejich registračních údajů; primárním klíčem této tabulky je nezaměnitelný atribut ID

Tabulka **stranky_materialy** – slouží k evidenci uživatelem nahraných materiálů a je podkladem pro výpis souborů v Administrační webové části v záložce Moje soubory; cizím klíčem této tabulky je user_id z tabulky auth_user

Tabulka **stranky_gui** – slouží jako seznam všech JPG souborů určených k prezentaci; cizím klíčem této tabulky je **materialy_id** z tabulky **stranky_materialy**

Použitím cizích klíčů bylo dosaženo smazání všech závislých záznamů v případě odstranění nadřazeného. Fyzické smazání souborů je ošetřeno tak, že pokud uživatel smaže záznam, ve view je volána metoda, která odstraní jak záznam z tabulky, tak i příslušný či příslušné soubory z adresáře.

Kontrola, zda uživatelem nahraný záznam (Událost) je stále aktuální, probíhá na pozadí periodickým kontrolováním rozdílů mezi aktuálním datem a datem zveřejnit DO. Pokud je atribut **do_kdy** starší aktuální datum, ve view je volána metoda, která odstraní jak záznam z tabulky, tak i příslušný či příslušné soubory z adresáře.

4.3.1.7 Přehled uživatelů

[Uvodní list](#)
[Moje údaje](#)
[Moje soubory](#)
[Přehled uživatelů](#)
[Odhlásit administrátora: pi](#)

Obrázek 4-17: Lišta záložek Webového rozhraní - Přehled uživatelů.

Přehled uživatelů je kategorie, která je dostupná pouze administrátorovi, kterému nabízí přehled nad uživateli v systému. Administrátor je uživatel, který má u svého záznamu v tabulce **auth_user** atribut **is_staff** = „1“. O každém uživateli se administrátor dočte: uživatelské jméno, Křestní jméno, Příjmení, E-mail, Zda je účet aktivní a datum registrace.

4.3.1.8 Odhlášení

Odhlášení probíhá prostřednictvím odkázání na URL z popisku odhlášení na hlavní liště.

localhost /uzivetelske_ucty/odhlaseni/

Jedná se o definovanou URL adresu v `urls.py`, která odkazuje na view, které spustí zabudovanou funkci **logout (request)**. Následně uživatele odkáže na úvodní přihlašovací stránku.

4.3.1.9 Bezpečnost

Obsahu webu je ošetřena integrovanou funkcí autentizačního systému Django - **login, logout, authenticate, login_required**.

login – využívá autentizace systému Django `include('django.contrib.auth.urls')`, jehož použitím vzniká URL: `localhost /uzivetelske_ucty/login/`

logout – integrovaná funkce autentizačního systému: `logout (request)`

authenticate – slouží k ověření uživatele, zda existuje

is_staff – ověří, zda se jedná o administrátora

login_required – pro použití je nutné ve view, které se stará jak o logiku na jednotlivých stránkách, tak odkazuje na stránky samotné importovat **login_required (from django.contrib.auth.decorators import login_required)**

Následné použití spočívá v přidání dekorátoru před samotné view:

```
@login_required (login_url='/uzivetelske_ucty/login/')
def po_registraci(request): #
    return render(request, 'registration/po_registraci.html')
```

V případě, že `urls.py` odkazuje na view **po_registraci** a nejste přihlášení, autentizační systém Django Vás odkáže na stránku přihlášení.

Pokud přihlášení jste, view Vám vygeneruje obsah požadované html stránky.

4.3.1.10 Celery

Pro periodickou kontrolu bylo využito nástroje Celery, který vykonává tuto kontrolu nezávisle na chodu Webové aplikace. Pro jeho použití je nutná instalace a implementace do projektu v jeho kořenovém adresáři. Zde je nutné Celery importovat v souboru `settings.py` a ručně vytvořit soubor `celery.py`, jehož prostřednictvím se registruje periodická úloha. Jakmile je Celery s projektem provázáno, prohledává jednotlivé aplikace projektu, zda se v nich vyskytuje soubor `tasks.py`. Pokud na takový soubor

narazí, vykonává jeho obsah dle požadovaného harmonogramu, určeného v `celery.py`. Vykonání jedné periodické kontroly bez smazání záznamů zabere řádově 20 milisekund. Při smazání pokusného PDF souboru o 10 stránkách, trvalo smazání záznamů ve dvou tabulkách **stranky_materialy** a **stranky_gui**, původního PDF souboru a 10 stran v JPG formátu řádově 214 milisekund (Obrázek 4-18). Těchto měření bylo možné dosáhnout spuštěním služby **worker** a **beat** příkazem: **celery -A DJproject worker -l info -B**. Jelikož tento příkaz slouží k ručnímu spuštění služby, není v řešení této úlohy použit. Nýbrž je nastaven autostart této služby v konfiguračních souborech.

```
[2019-05-09 10:46:00.033: WARNING/ForkPoolWorker-3] Dnešní datum je: 2019-05-09
[2019-05-09 10:46:00.053: INFO/ForkPoolWorker-3] Task stranky.tasks.muj_ukol[1fea32f1-48d1-48eb-8fe9-39011aa44a64] succeeded in 0.02095634399984102s: None
[2019-05-09 10:47:00.007: INFO/Beat] Scheduler: Sending due task stranky.tasks.muj_ukol1 (stranky.tasks.muj_ukol)
[2019-05-09 10:47:00.019: INFO/MainProcess] Received task: stranky.tasks.muj_ukol1[26dd1578-d0b9-49b4-bf80-e6a087036aa6]
[2019-05-09 10:47:00.026: WARNING/ForkPoolWorker-5] Dnešní datum je: 2019-05-09
[2019-05-09 10:47:00.062: WARNING/ForkPoolWorker-5] Proběhlo smazání souboru: uzivatel_1/10_stran.pdf a jeho 10 JPG souborů
[2019-05-09 10:47:00.239: INFO/ForkPoolWorker-5] Task stranky.tasks.muj_ukol1[26dd1578-d0b9-49b4-bf80-e6a087036aa6] succeeded in 0.21407890299997234s: None
[2019-05-09 10:48:00.057: INFO/Beat] Scheduler: Sending due task stranky.tasks.muj_ukol1 (stranky.tasks.muj_ukol)
[2019-05-09 10:48:00.072: INFO/MainProcess] Received task: stranky.tasks.muj_ukol1[a9f441f0-9c6f-463f-87f4-4674c1f540ae]
```

Obrázek 4-18: Automatické odstranění Události za pomoci Celery.

Zavedením vytváření seznamu platných souborů JPG pro prezentaci a kontrolu, zda nějaký záznam pozbyl platnosti vlivem stáří (atribut `DO`), bylo dosaženo, že Prezentáčnická aplikace nebude zbytečně zatěžována operacemi, které by činili její odezvy pomalejší, a snižoval by se výsledný komfort užívání.

4.3.2 Prezentáčnické GUI rozhraní

Jedná se o samostatnou aplikaci, nezávislou na Webovém administračním rozhraní. Konkrétně se jedná o samostatný Python kód. Bodem, ve kterém se Webové administrační rozhraní protíná s Prezentáčnickým rozhraním je databázová tabulka **stranky_gui**. Pokud bude tato tabulka obsahovat záznamy o datech k uveřejnění, aplikace data zobrazí. Pokud ne, informuje uživatele o prázdném obsahu kategorie. Tím je dosaženo nezávislosti.

4.3.2.1 Základní stavba GUI

Při realizaci Prezentáčnické aplikace, alias Graphical User Interface (GUI), je využita knihovna Tkinter, která je součástí instalace Python. Jedná se o samostatný kód, který se skládá ze dvou hlavních částí. První část má za úkol vytvořit vlastní okno Prezentáčnického systému s obsahem tlačítek a prezentovaných dat. Po importování knihovny Tkinter je definována třída `Application`, která dědí z třídy `Frame`. Poté je zavolána metoda `__init__()` při inicializaci prezentáčnického okna. Na samotném konci kódu je vytvořen kořen aplikace, který zajišťuje zobrazení okna, jeho ukončení a minimalizaci. Dále je nutné vytvoření objektu aplikace `Application(master=root)` a spuštění hlavního cyklu programu `mainloop()`, který čeká, jakmile nastane událost či callback, které následně vykoná. Vizualní podoba aplikace je navržena v režimu fullscreen bez roletového menu v horní části obrazovky za pomoci `root.attributes("-fullscreen", True)`. Za pomoci těchto kroků bylo možné realizovat prázdné okno Prezentáčnické aplikace.

Jelikož GUI zobrazuje data, která mohou uživatele prostřednictvím Webového rozhraní data v průběhu času libovolně měnit, je nutné, aby v druhé části kódu byl vytvořen samostatný paralelní proces (vlákno), který bude kontrolovat aktuálnost dat v databázové tabulce **stranky_gui**. Tomuto vláknu byla svěřena i úloha obsluhy Joysticku.

4.3.2.2 Tvorba Widgetů

Jakmile bylo vykonáno vše potřebné pro vytvoření okna aplikace, bylo nutné okno doplnit funkčními prvky (Widgets), nutných pro obsluhu zařízení. V tomto případě se jedná o tlačítka. Smyslem obrazovky je nabízet počet kategorií, dle počtu kategorií definovaných v Django třídě **FormNaSoubory** v forms.py aplikace stránky. Za tímto účelem je vytvořen adaptivní layout pomocí rozmístění prvků GRID, které Tkinter nabízí. Adaptivita spočívá v tom, že kód GUI přizpůsobuje pomyslnou mřížku Widgetů kategoriím, za definovaných na začátku kódu. Pokud by měla být nabídka rozšířena či jinak pozměněna, musí se tato změna promítnout právě zde: `self.DRUHYKATEGORII`. Jedná se o jediný manuální krok při práci s kódem. Zbytek funkcí je navržených tak, aby se přizpůsobily nastalým podmínkám.

Pomyslná mřížka je definována jako sloupec kategorií po levém boku. Pod posledním řádkem kategorie je definován řádek další, který slouží pro tvorbu pohybových tlačítek pro přemítání obsahu. Během realizace se vyskytla potřeba mít přehled, kde se uživatel během listování nachází. Proto mezi pohybová tlačítka byl umístěn text o aktuální stránce a celkového počtu stránek právě zvolené kategorie. Tato pohybová tlačítka s infotextem byla umístěna pod zobrazovaný obsah. Tím je řečeno, že celková výška zobrazovacího zařízení je rozdělena na (počet kategorií + 1) řádků a (sloupec kategorií + dvě tlačítka pohybu + jeden infotext) sloupců.

Jednotlivá tlačítka kategorií jsou během inicializace okna vytvořena s vlastností roztažnosti ve vertikální ose. Tím je zajištěna adaptivita tlačítek při změně počtu kategorií. V horizontální ose je tlačítkům kategorie definována pevná šířka tlačítek v jednotkách pixelů, pomocí skrytého tlačítka pod posledním tlačítkem kategorie. Skrytí tohoto tlačítka bylo realizováno za pomoci definice stylu tlačítka na FLAT styl. Tento styl vykreslí tlačítko bez jakýchkoli hran, proto je nazýváno imaginární.

Nabízí se možnost definovat šířku i výšku tlačítkům ručně v pixelech. Bohužel takto definované rozměry v pixelech nekorespondovaly s pixely obrazovky. Proto bylo využito imaginárního tlačítka, kterému bylo definováno zobrazení prázdného obrázku s rozměry 1x1 pixel. Takto realizovanému imaginárnímu tlačítku je poté možno definovat rozměry v pixelech, korespondujících s pixelem použitého zobrazovacího zařízení.

Díky strategickému umístění imaginárního tlačítka do levého dolního rohu je docílena pevná výška řádku tlačítek a infotextu (Obrázek 4-19). Šířka pohybových tlačítek a infotextu je přizpůsobitelná na šířku zobrazovacího zařízení.

Pro lepší čitelnost tlačítek, byly určeny odstupy od okrajů v ose X a Y nastavením parametrů `padx` a `pady`.



Obrázek 4-19: Zobrazené imaginární tlačítko v GUI - bez Fullscreenu.

4.3.2.3 Funkce jednotlivých tlačítek

Aby jednotlivá tlačítka kategorií mohla uživateli zobrazit obsah, byla za tímto účelem vytvořena metoda `nacteni_dat`, která ve svém začátku naváže spojení s databází `stranky_gui`. Vstupním parametrem této metody je `zvolena_kategorie`, jenž nabývá rozměru dle počtu kategorií. Ve výchozím stavu má hodnotu „0“, která vyčte obsah první kategorie z celkového počtu. Vyčtení dat z databáze je realizováno prostřednictvím `mysql.conector`, který prohledá obsah tabulky `stranky_gui` podle zvoleného filtru `zvolena_kategorie` prostřednictvím SQL dotazu. Výsledkem tohoto dotazu je uložení seznamu cest k jednotlivým souborům pro prezentaci do seznamu `cesta`.

Při realizaci volby kategorie Joystickem se nabízely dvě různé cesty řešení. První způsob mohl probíhat tak, že by se při každém pohybu nahoru či dolů načel obsah

kategorie a zobrazil v příslušné ploše. Jedná se o proveditelnou záležitost, ale z pohledu programátora se jevila jako nevhodná ve smyslu nadměrnému zatěžování výkonu zařízení. Pokud by uživatel chtěl zobrazit obsah třetí sousední kategorie, mělo by to za následek trojnásobné dotazování se databáze na výsledky a zobrazení alespoň prvního obsahu všech tří kategorií. Proto byl realizován odlišný přístup.

Svislé pohyby Joystickem přesouvají rámeček kolem tlačítka kategorie. Tímto způsobem se uživatel může třemi pohyby Joysticku přemístit tímto rámem na třetí sousední kategorii bez opakovaného dotazování databáze na výsledky. O výsledky, respektive o obsah, si uživatel požádá až následným stiskem hříbku Joysticku. Právě stisk Joysticku obsah vyvolá.

Právě zvolená kategorie je uživateli indikována modrým zbarvením tlačítka. Zabarvení kategorie tlačítka či černý rám kolem tlačítka kategorie je dostačující řešení, které uživatele informuje, v jakém stavu se Prezentační aplikace nachází ve smyslu kategorií.

Horizontální ovládací prvky pro listování v obsahu dané kategorie jsou svázána s horizontálními pohyby Joysticku. Aby byl uživatel informován o zvoleném pohybu, jsou v zapojení realizovány LED diody. Další zpětnou vazbou je krátkodobého modrého zbarvení tlačítka listování a změny jeho stylu. Po přesně definovaném čase 250 milisekund od pohybu Joysticku signalizační LED zhasne. Po přesně definovaném čase 1 sekundy od pohybu Joysticku odezní modré zbarvení tlačítka listování.

Po každém přemítnutí obsahu se mění číslo aktuálně zobrazované stránky z celkového počtu výsledků SQL dotazu. Pokud dojde k přetečení či podtečení rozsahu stránek vlivem listování v libovolném směru, pokračuje opět z protějšího konce. Z Max. → na Min. či z Mín. → na Max..

4.3.2.4 Kontrola platnosti dat

Jak již bylo v této práci zmíněno, platnost dat kontroluje Webové rozhraní. Tato kontrola spočívá v tom, zda uživatelem realizovaný záznam již nevypršel svou platností zveřejnění. Porovnává aktuální datum s datem, DO kterého uživatel chce materiál uveřejnit.

Takto odfiltrované záznamy je potřeba validovat podruhé. Jelikož nastávají situace, kdy materiály mají být teprve v budoucnu uveřejněny (Obrázek 4-20). Proto byl za tímto účelem vytvořen seznam **cesta**, do kterého jsou z každého výsledku SQL dotazu ukládány cesty k souborům, které mají datum uveřejnit OD starší, než datum aktuální. Tyto úkony řeší metoda **nacteni_dat**, při každém jejím zavolání.

id	od	do	kateg	naz	materialy_id
28	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor.jpg	15
29	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor_1.jpg	16
30	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor_2.jpg	17
31	2019-03-25	2019-12-25	Obecné	uzivatel_1/soubor_3.jpg	18
48	2019-06-01	2019-12-25	Obecné	uzivatel_1/2_stranky_1.jpg	20
49	2019-06-01	2019-12-25	Obecné	uzivatel_1/2_stranky_2.jpg	21

```

dnešní datum: 2019-05-10
cesta:
[ 'uzivatel_1/soubor.jpg',
  'uzivatel_1/soubor_1.jpg',
  'uzivatel_1/soubor_2.jpg',
  'uzivatel_1/soubor_3.jpg' ]

```

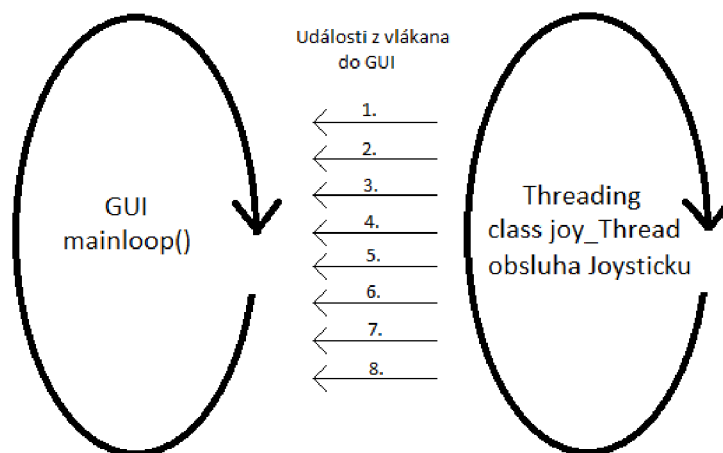
Obrázek 4-20: Výsledky po vyfiltrování platných záznamů z DB stranky_gui.

4.3.2.5 Paralelní proces

Funkcí Joysticku není pouze rozsvícení příslušné LED diody, ale i vykonat akci v GUI aplikaci. GUI je vykonáváno jako nekonečná smyčka **mainloop()**, která čeká na události či callbacky, které okamžitě vykoná. A právě použití metody **sleep()** pro rozsvícení LED diody se ukázalo jako problém při použití v GUI, jelikož během metody **sleep()** dojde k pozastavení vykonávání programy na metodou definovaný čas. Proto bylo nutné použití paralelního procesu (Obrázek 4-21), který vyše do GUI událost a rozsvícení LED diody (pozastavení vykonávání programu) přenechat právě paralelnímu procesu.

Použité Widgety ve formě tlačítek interagují s akcí kurzoru myši (najeť myši nad tlačítko, stisk tlačítka levým tlačítkem myši či stisk tlačítka pravým tlačítkem myši). Jelikož v této úloze není myš použita, nýbrž Joystick, callbacky nejsou využívány.

Konstrukce generátorů událostí je řešena použitím modulu Threading, který primárně obsluhuje pohyby Joysticku ve všech osách X & Y & Z, jak je popsáno v kapitole 4.2. Přidruženou úlohou je i generovat události, obsluhující aktuálnost dat pro zobrazení a spuštění automatické prezentace při nečinnosti Joysticku po definovaný čas.



Obrázek 4-21: Paralelní procesy a události.

Událostí, které vlákno generuje je celkem 8.

- | | |
|----------------------------------|---|
| 1. volání metody J_levo() v GUI | - obslouží přemítnutí stránky do leva |
| 2. volání metody J_pravo() v GUI | - obslouží přemítnutí stránky do prava |
| 3. volání metody J_dole() v GUI | - obslouží pohyb rámu kategorie vzhůru |
| 4. volání metody J_hore() v GUI | - obslouží pohyb rámu kategorie dolů |
| 5. volání metody clear() v GUI | - obslouží odbarvení tlačítek přemítnutí |
| 6. vykoná validaci dat v GUI | - tzv. AUTOUPDATE dat k prezentaci |
| 7. volání výběru kategorie v GUI | - obslužení volby kategorie stiskem JOYS. |
| 8. volání autorotace v GUI | - po nečinnosti spustí AUTOPREZENTACI |

Jako prostředník mezi vláknem a GUI je použit modul Queue. Jedná se o vytvoření fronty, která pracuje v režimu FIFO (First In First Out). Na straně vlákna je tato fronta plněna hodnotami, které byly na základě logiky algoritmu vlákna vygenerovány. Na straně druhé probíhá periodické vyčítání zpráv z fronty a GUI vykoná odpovídající akci. Po vyčtení zprávy z fronty dojde k uvolnění fronty.

4.3.2.6 Generování obsahu v GUI

Materiály, které uživatel nahraje prostřednictvím Webového rozhraní, projdou kontrolou na formát PDF. V případě, že je PDF detekováno, je uloženo na lokální disk a následně proběhne rozklad na JPG soubory.

Takto připravené soubory jsou kontrolovány na datumovou platnost jak na straně Webového rozhraní, tak i na straně GUI aplikace.

Kód GUI na svém počátku detekuje šířku i výšku použitého zobrazovacího zařízení, od kterých jsou odečteny rozměry výšky i šířky imaginárního tlačítka. Tímto krokem je definován rozměr plochy, do které je možné materiály zobrazit. Další vlastností takto určené plochy je určení poměru stran. Právě z poměru stran program zjistí, zda se jedná o širokoúhlou plochu či vysokou.

Charakter obrazovky je určujícím faktorem pro následnou rozměrovou úpravu prezentovaného materiálu. Při načtení každého konkrétního souboru JPG jsou načteny jeho vlastní rozměry výšky a šířky, ze kterých se určí poměr stran obrázku. Při úpravě velikosti obrázku je nutné tento poměr stran zachovat, aby nedošlo k jeho deformaci.

Pokud je plocha, do které bude obrázek širší, nežli vyšší, je právě výška rozměr, který obrázek nesmí přesáhnout. Dle poměru stran obrázku je dopočítána šířka obrázku. Program pracuje analogicky v případě, že plocha, do které bude obraz promítnut, je vyšší. Takzvaně zobrazovací zařízení bude orientované na výšku. V tomto případě je nejmenší rozměr šířka a obrázku bude dopočítána výška.

Těmito kroky je dosíleno přizpůsobení prezentovaného materiálu k ploše, která je prezentaci určena.

```
Slide číslo: 2 a byl zobrazen: uzivatel_1/soubor_1.jpg
Převod a vygenerování obrázku zabralo času: 0:00:00.661710
```

Obrázek 4-22: Časová náročnost generování obrazového obsahu.

4.3.2.7 Auto update prezentovaných dat

Důležitým faktorem, který bylo nutné vyřešit, je aktuálnost prezentovaných dat. Jelikož je Elektronická informační tabule LCD dostupná prostřednictvím Internetu, mohou nastat tři důležité momenty, se kterými je potřeba v programu GUI počítat a mít tyto okamžiky řádně ošetřeny. Jedná se o:

- vypršení platnosti prezentovaného materiálu automatickým smazáním
- odstranění prezentovaného materiálu uživatelem

- přidání materiálů k prezentaci

Všechny tyto události ovlivňují celkový počet materiálů k prezentaci každé kategorie, který je indikován v infotextu GUI aplikace.

Za tímto účelem byl realizován mechanismus periodických kontrol obsahu databázové tabulky za pomoci SQL dotazu dané kategorie. Jedná se o identickou metodu **nacteni_dat**, která je použita při vyčtení obsahu zvolené kategorie. Její volání probíhá na podmět (událost), která dorazí z vlákna. Ve vláknu lze nastavit, s jako frekvencí bude GUI metoda vláknem volána.

Díky tomuto mechanismu se aplikace chová tak, že infotext vždy ukazuje aktuální počet materiálů v kategorii, aniž by uživatel musel vykonat nějakou akci, která obsah validuje.

Pokud uživatel vykoná kteroukoliv akci Joystickem (přemítnutí obsahu vpřed či vzad, volba kategorie), jsou volány metody, které volání metody **nacteni_dat** také obsahují.

Tímto způsobem je dosaženo udržení maximální aktuálnosti prezentovaných dat.

```
Spojení s MYSQL databází bylo cíleně ukončeno.
      Vyčtení dat z DATABÁZE zabralo času: 0:00:00.021132
***-----***
```

Obrázek 4-23: Časová náročnost vyčtení záznamů z DB stránky_gui.

4.3.2.8 Auto prezentace

Jelikož cílem Elektronické informační tabule LCD je sdělit informace maximálnímu počtu čtenářů, volíme instalaci zařízení do míst s co největší migrací lidí. Pokud zařízení umístíme nevhodně, ztrácí informační tabule na svém užitku. Na druhou stranu, pokud zvolíme ideální místo, můžeme napomoci využití jejího potenciálu programovou funkcí auto prezentace. Díky této funkcionalitě můžeme zaujmout i potencionálního čtenáře, který původně neměl v plánu si prezentovaný obsah přečíst a pouze kolem zařízení procházel.

Podmětem pro spuštění auto prezentace v GUI je příchod události z paralelního procesu. Vlákno periodicky kontroluje, zda byly splněny dvě podmínky, kterými se mechanismus auto prezentace řídí. Konkrétně je nutné, aby si uživatel určil dva časové údaje:

- čas nečinnosti
- čas mezi snímky

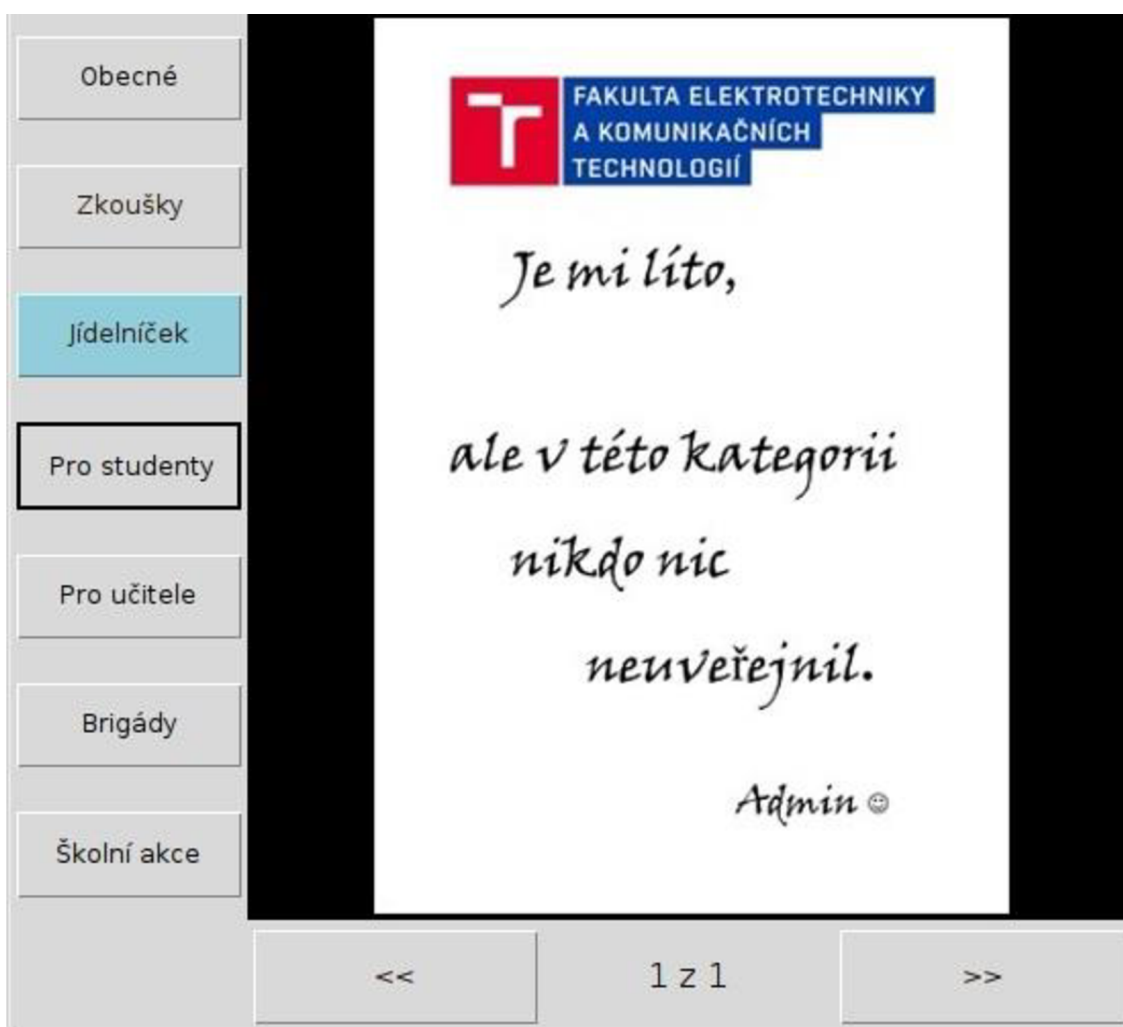
Jak již bylo řečeno, primární úlohou vlákna je vyčítání stavů Joysticku. Toho je při auto prezentaci využito. Každým pohybem je časovač nečinnosti resetován a k auto

prezentaci nedochází. Naopak, jakmile je uživatelem nastavená doba nečinnosti překročena, dojde k povolení automatického přemítání snímků.

V tento moment se spustí druhý časovač, který při dosažení uživatelem definovaného času mezi snímky vykoná přemítnutí snímku voláním metody **J_pravo()** a následného vlastního resetování časovače. Tento cyklus se opakuje do té doby, než je vykonán pohyb v kterékoliv ose Joysticku, který časovač nečinnosti resetuje.

4.3.2.9 Defaultní stránka

Při nasazení zařízení do praxe či i během provozu nastane situace, že jedna či více kategorií nebude obsahovat materiály k uveřejnění. Tato situace je ošetřena identifikací prázdné množiny výsledků k prezentaci, kde je následně volána metoda **default()**, která zobrazí informaci o prázdném obsahu (Obrázek 4-24).



Obrázek 4-24: Výsledná Prezentační aplikace v režimu Fullscreen.

5. ZÁVĚR

Cílem této diplomové práce bylo realizovat Elektronickou informační tabuli LCD, které předcházely průzkum mezi informačními panely, založených na použití LCD zobrazovačů.

První kapitola disponuje výčtem tuzemských producentů profesionálních zařízení, které uvádějí pod názvem informační kiosky. U zahraničních výrobců se užívá název z anglického jazyka, konkrétně Digital Signage. V této kapitole byla popsána široká nabídka realizovaných zařízení, tak i funkce, které tato řešení nabízí.

Na základě nabytých poznatků z první kapitoly, byl realizován návrh hardwaru zařízení, na který byl kladen důraz spolehlivosti a komfortu použití. Pro realizaci zařízení byl z portfolia Raspberry Foundation vybrán model Raspberry Pi Model 3, který disponuje výkonným čtyř jádrovým procesorem 1,2 GHz, dostatečnou pamětí 1 GB, HDMI konektorem a připojením k síti prostřednictvím Ethernet a Wi-Fi. Tento hardware zařízení bylo nutné doplnit paměťovým prostorem pro operační systém (Raspbian), programové vybavení (Python, Apache 2, Django, Tkinter a další) a data nahraná uživateli do aplikace Elektronické informační tabule v podobě 8GB paměťové karty Class 10. Dále byla realizována periférie Joystick pro interakci člověka se zařízením, který byl doplněn o analogově digitální převodník ADS1015 a signalizačními LED diodami stavů Joysticku na společné desce plošných spojů. Jako zobrazovací zařízení byl při realizaci zařízení použit LCD televizor, který je možné nahradit kterýmkoliv HDMI zařízením.

Po prostudování funkcí profesionálních systémů v první kapitole byl učiněn návrh koncepce programového vybavení Prezentační části systému, který byl během realizace rozdělen do dvou samostatných částí.

Část první slouží ke vzdálenému přístupu k Webovému rozhraní, jehož prostřednictvím má uživatel zařízení dostupné prostřednictvím sítě Internet. Na zařízení je provozován webový server Apache 2. Komunikace uživatele se zařízením je založena na architektuře klient-server. Webové rozhraní bylo realizováno frameworkem Django, který nabízí autentizační systém, jehož prostřednictvím je zaručena bezpečnost aplikace proti nežádoucímu přístupu neregistrovaných uživatelů. Obsah webového rozhraní a jeho funkce byly realizovány za pomoci jazyka Python a Django architektury Model-Views-Controller, která odděluje logiku od výstupu a činí tak práci s frameworkem přehlednou a efektivní i pro vývoj rozsáhlých aplikací. Pro práci s daty od uživatelů bylo využito objektově relační mapování, díky kterému Modely odpovídaly databázovým tabulkám. Pro design webového obsahu byl realizován sadou nástrojů Bootstrap za pomoci rozložení obsahu do mřížek.

Uživatelské funkce webového rozhraní jsou: login, registrace, seznámení s obsluhou webu, změny uživatelských údajů, zakládání událostí, přehled založených událostí,

vyhledávání v založených událostech, přehled registrovaných uživatelů (pouze administrátorovi) a odhlášení.

Prostřednictvím založení události je uživateli umožněno nahrání souboru ve formátech PNG, BMP, JPG a PDF. Této události uživatel přiřazuje vlastní název, OD a DO kdy má být událost uveřejněna a do které kategorie patří. Počet kategorií je pevně určen a charakterem koncipován do prostor školy. Při překročení data, do kterého má být událost uveřejněna, je událost se souborem či soubory vymazána.

Druhá část zařízení je zaměřena na prezentaci nahraného obsahu. Jedná se o grafické rozhraní, napsáno v Pythonu s využitím knihovny Tkinter. Toto rozhraní vyčítá cesty k souborům pro prezentaci z databázové tabulky, kterou vytváří Webové rozhraní. Design aplikace byl zkonstruován metodou mřížky, kde levý sloupec nabízí tlačítka jednotlivých kategorií, spodní řádek disponuje tlačítky pro listování v obsahu zvolené kategorie. Pro orientaci v obsahu byl mezi tlačítka listování v obsahu umístěn infotext o zobrazované stránce z celkového počtu. Vertikální pohyby Joystickem umožňují výběr kategorie, kde pro vyvolání obsahu slouží stisk hříbku Joysticku. Horizontální pohyby Joysticku slouží k listování obsahem.

Touto realizací zařízení s periferií a komplexním programovým vybavením byly splněny všechny body dle zadání. Zařízení bylo odzkoušeno a je plně funkční.

Nad rámec zadání bych doporučil tři praktická vylepšení. Pokud dojde k zapnutí libovolného modelu Raspberry Pi z portfolia Raspberry Foundation bez připojení k Internetu, zařízení nepracuje s přesným časem. Důvodem je, že ani jeden z modelů Raspberry Pi nedisponuje záložní baterií, díky které by si aktuální čas uchoval. Proto by bylo vhodné zapojení doplnit obvodem reálných hodin (např. DS3231) s knoflíkovou baterií, připojenou prostřednictvím sběrnice I2C a přesný čas cyklicky vyčítat.

Dalším námětem pro rozvoj zařízení je realizace rozesílání ověřovacích emailů během registrace, které Django umožňuje.

Pokud by na zařízení byly kladeny požadavky ovládání pomocí dotykové obrazovky, doporučuji použití otevřené knihovny Kivy, která nabízí podporu multi-dotykových aplikací.

Literatura

- [1] Rábová, Z., Hanáček, P., Peringer, P., Příkryl, P., Křena, B.: Užitečné rady pro psaní odborného textu [online]. Brno: c1993-2008, aktualizováno 2008-11-01 [cit. 2008-11-28]. Dostupné na URL:
http://www.fit.vutbr.cz/info/statnice/psani_textu.html
- [2] Kolektiv autorů: Pravidla českého pravopisu. Academia, Praha, 1993. ISBN 80-200-0475-0
- [3] Šablona pro BP/DP a prezentace v2.63 [online]. Brno: FEKT VUT, 2017 [cit. 2017-03-06]. Dostupné z: <http://latex.feec.vutbr.cz/sablona/>
- [4] Arduino Joystick s tlačítkem KY-023. *Laskarduino* [online]. \n: \n, 2016 [cit. 2018-01-01]. Dostupné z: <https://laskarduino.cz/ovladaci-prvky/132007-joystick-s-tlacitkem-ky-023.html>
- [5] TISKOVÉ ZPRÁVY: Multimediální kiosky v kraji zlepši služby nejenom turistům. *Krajský úřad Pardubického*[online]. Pardubice: \n, 2013, 31. 1. 2013 [cit. 2018-01-01]. Dostupné z: <https://www.pardubickykraj.cz/tiskove-zpravy/72994/multimedialni-kiosky-v-kraji-zlepsl-sluzby-nejenom-turistum?preiev=archiv>
- [6] Python - MySQL Database Access. *Tutorialspoint* [online]. \n: \n, 2017 [cit. 2018-01-01]. Dostupné z:
https://www.tutorialspoint.com/python/python_database_access.htm
- [7] VÝROBA PLOŠNÝCH SPOJŮ [online]. VUT v Brně, 2010 [cit. 2018-01-01]. Dostupné z:
https://www.urel.feec.vutbr.cz/web_documents/dilna/PlosneSpoje2010.pdf.
Postup návrhu plošných spojů v Eaglu. VUT v Brně.
- [8] KONSTRUKCE ELEKTRONICKÝCH ZAŘÍZENÍ: počítačový návrh plošných spojů - cvičení [online]. VUT v Brně, 1999 [cit. 2018-01-01]. Dostupné z:
<https://user.unob.cz/biolek/vyukaVUT/skripta/Eagle.pdf>. Postup návrhu plošných spojů v Eaglu. VUT v Brně. Vedoucí práce Ing. Václav Zeman
- [9] GPIO Electrical Specifications. *Mosaic-industries* [online]. [cit. 2019-05-13]. Dostupné z: <http://www.mosaic-industries.com/embedded-systems/microcontroller-projects/raspberry-pi/gpio-pin-electrical-specifications>
- [10] RPi-ovládání pomocí GPIO. *XBMC-Kodi* [online]. [cit. 2019-05-13]. Dostupné z: <https://www.xbmc-kodi.cz/prispevek-rpi-ovladani-pomoci-gpio-tlacitek-vyreseno>
- [11] Discussions. *Element14* [online]. [cit. 2019-05-13]. Dostupné z: <https://www.element14.com/community/thread/20982/1/raspberry-pi-gpio-pin-max-current?displayFullThread=true>
- [12] GPIO Pads Control2. *Scribd* [online]. [cit. 2019-05-13]. Dostupné z: <https://www.scribd.com/doc/101830961/GPIO-Pads-Control2>

- [13] Srovnání – kdy je lepší MySQL a kdy MariaDB?. *Computerworld* [online]. [cit. 2019-05-13]. Dostupné z: <https://computerworld.cz/software/srovnani-kdy-je-lepsi-mysql-a-kdy-mariadb-50258>
- [14] Arduino PS2 Joystick Schematic. *Henrysbench.capnfatz* [online]. [cit. 2019-05-13]. Dostupné z: <http://henrysbench.capnfatz.com/henrys-bench/arduino-sensors-and-input/arduino-ps2-joystick-tutorial-keyes-ky-023-deek-robot/>
- [15] I2C AD převodník ADS1115/ADS1015 12 Bit 4 kanály. *Arduino-shop* [online]. [cit. 2019-05-13]. Dostupné z: <https://arduino-shop.cz/arduino/2956-i2c-ad-prevodnik-ads1115-ads1015-12-bit-4-kanaly.html>
- [16] KY-023 Dual Axis XY Joystick Module. *All top notch* [online]. [cit. 2019-05-13]. Dostupné z: <https://alltopnotch.co.uk/product/ky-023-dual-axis-xy-axis-joystick-module/>
- [17] Obrázek. *Theendlessaisle* [online]. \n: \n, 2014, [cit. 2018-01-01]. Dostupné z: https://www.theendlessaisle.com/wp-content/uploads/2014/05/nbn_combined_large_extended5.png
- [18] MCM welcome to Newark | Newark element14. *Pinterest* [online]. [cit. 2019-05-13]. Dostupné z: <https://cz.pinterest.com/pin/408420259948485231/>
- [19] MySQL databáze - český manuál. *Junext* [online]. [cit. 2019-05-13]. Dostupné z: <https://www.junext.net/mysql/>
- [20] Python - Začínáme s GUI (12. díl). *Www.banan.cz* [online]. [cit. 2019-05-13]. Dostupné z: <https://www.banan.cz/serialy/python/Python-Zaciname-s-GUI-12-dil>
- [21] Settings. *Django* [online]. [cit. 2019-05-13]. Dostupné z: <https://docs.djangoproject.com/en/2.2/ref/settings/>
- [22] Ads1015.pdf: datasheet. *Cdn-shop.adafruit* [online]. [cit. 2019-05-13]. Dostupné z: <https://cdn-shop.adafruit.com/datasheets/ads1015.pdf>
- [23] 18bitový ADC s vývojovým boardem. *Vyvoj.hw.cz* [online]. [cit. 2019-05-13]. Dostupné z: <https://vyvoj.hw.cz/soucastky/18bitovy-adc-s-vyvojovym-boardem.html>
- [24] MICROCHIP TECHNOLOGY MCP3008-I/P. *Vyvoj.hw.cz* [online]. [cit. 2019-05-13]. Dostupné z: https://www.tme.eu/cz/details/mcp3008-i_p/prevodniky-a-d-integrované-obvody/microchip-technology/mcp3008-i-p/
- [25] Trackbally k zabudování do panelu. *Elvac* [online]. [cit. 2019-05-13]. Dostupné z: <https://www.elvac.eu/Default.aspx?tabid=701&Display=ProductDetailView&TreeType=Product&ProductId=96069&Language=cs-CZ>
- [26] Raspberry Pi 3 Model B 64-bit 1GB RAM. *Http://rpishop.cz* [online]. [cit. 2019-05-13]. Dostupné z: <http://rpishop.cz/raspberry-pi-3b/283-raspberry-pi-3-model-b-64-bit.html>
- [27] Raspberry Pi Zero W. *Http://rpishop.cz* [online]. [cit. 2019-05-13]. Dostupné z: <http://rpishop.cz/raspberry-pi-zero/647-raspberry-pi-zero-w.html>
- [28] Raspberry Pi 3 Model B+ 64-bit 1GB RAM. *Http://rpishop.cz* [online]. [cit. 2019-05-13]. Dostupné z: <http://rpishop.cz/raspberry-pi-3b/896-raspberry-pi-3-model-b-plus-64-bit-1gb-ram.html>

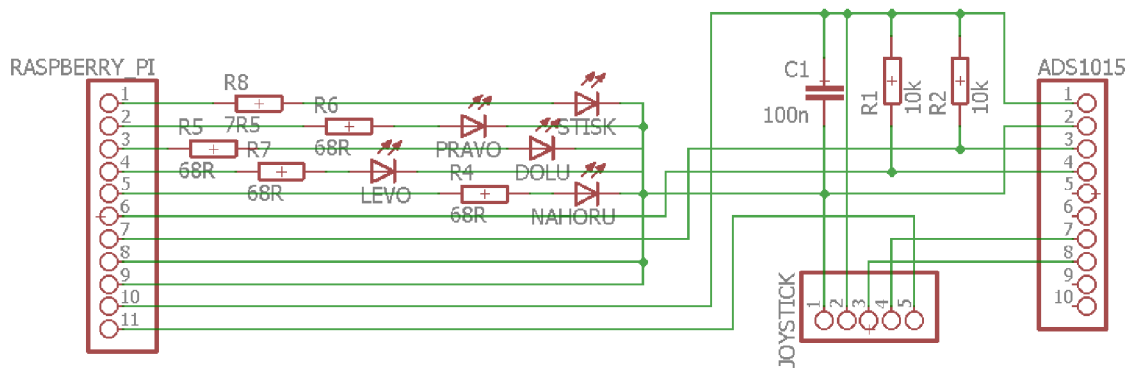
- [29] Vestavné počítače Digital Signage. *Elvac* [online]. [cit. 2019-05-13]. Dostupné z: <https://www.elvac.eu/Default.aspx?tabid=701&Display=CategoryProductListView&CategoryId=51856&language=cs-CZhtml>
- [30] Datove-zdroje. *Infopanel.s.eu* [online]. [cit. 2019-05-13]. Dostupné z: <https://infopanel.s.eu/o-systemu/software/datove-zdroje/>
- [31] Úvod. *Venkovni-displeje* [online]. [cit. 2019-05-13]. Dostupné z: <http://www.venkovni-displeje.cz/uvod.htm>
- [32] ZASTÁVKOVÉ LCD PANELY. *Herman elektronika* [online]. [cit. 2019-05-13]. Dostupné z: <http://www.herman.cz/cs/produkty/isrd/terminaly-a-zastavky/zastavkove-lcd-panely/>
- [33] Srovnání LCD displejů a LED panelů. *Herman elektronika* [online]. [cit. 2019-05-13]. Dostupné z: http://www.herman.cz/wp-content/uploads/2013/08/LCDvsLED_Herman_130608.pdf
- [34] Software. *Multimedialni-kiosky.cz* [online]. [cit. 2019-05-13]. Dostupné z: <https://multimedialni-kiosky.cz/software/>
- [35] ES_katalog_reseni_2018. *Elvac* [online]. [cit. 2019-05-13]. Dostupné z: https://www.elvac.eu/Portals/0/Docs/Katalogy/Solutions/ES_katalog_reseni_2018.pdf
- [36] Raspberry Pi: Úvod a výběr desky. *Arduino navody* [online]. [cit. 2019-05-13]. Dostupné z: <https://navody.arduino-shop.cz/navody-k-produktum/raspberry-pi-uvod-a-vyber-desky.html>
- [37] Raspberry Pi: Výběr operačního systému a první spuštění. *Arduino navody* [online]. [cit. 2019-05-13]. Dostupné z: <https://navody.arduino-shop.cz/navody-k-produktum/raspberry-pi-vyber-operacniho-systemu-a-prvni-spusteni.html>
- [38] Raspberry Pi mění svět: Seznamte se s nejzajímavějším počítačem dneška. *Hospodářské noviny* [online]. [cit. 2019-05-13]. Dostupné z: <https://tech.ihned.cz/geekosfera/c1-65195330-raspberry-pi-meni-svet-seznamte-se-s-nejzajimavejsim-pocitacem-dneska>
- [39] Úvod. *Ki-wi* [online]. [cit. 2019-05-13]. Dostupné z: <https://www.ki-wi.cz/>
- [40] Úvod. *Kivy* [online]. [cit. 2019-05-13]. Dostupné z: <https://kivy.org/#home>
- [41] User Guide. *Celeryproject* [online]. [cit. 2019-05-13]. Dostupné z: <http://docs.celeryproject.org/en/latest/userguide/periodic-tasks.html#crontab-schedules>
- [42] Učíme se Python. *Python v ČR* [online]. [cit. 2019-05-13]. Dostupné z: <https://python.cz/zacatecnici/>
- [43] Documentation: Writing your first Django app, part 1. *Django* [online]. [cit. 2019-05-13]. Dostupné z: <https://docs.djangoproject.com/en/2.2/intro/tutorial01/>
- [44] Get Started With Django Part 1: Build a Portfolio App. *Realpython* [online]. [cit. 2019-05-13]. Dostupné z: <https://realpython.com/get-started-with-django-1/>

- [45] Začínáme s MySQL a MariaDB. *OVH innovation for freedom* [online]. [cit. 2019-05-13]. Dostupné z: <https://docs.ovh.com/cz/cs/clouddb/zaciname-s-mysql-a-mariadb/>
- [46] Databáze MariaDB válčuje MySQL. *Root* [online]. [cit. 2019-05-13]. Dostupné z: <https://www.root.cz/clanky/databaze-mariadb-valcuje-mysql/>
- [47] Raspberry Pi a GPIO – co funguje. *Zdrojak.cz* [online]. [cit. 2019-05-13]. Dostupné z: <https://www.zdrojak.cz/clanky/raspberry-pi-gpio-co-funguje/>
- [48] Raspbian. *Raspberrypi* [online]. [cit. 2019-05-13]. Dostupné z: <https://www.raspberrypi.org/downloads/raspbian/>
- [49] Seznámení a konfigurace s Apach. *Zonercloud* [online]. [cit. 2019-05-13]. Dostupné z: <https://help.zonercloud.cz/inpage/seznameni-a-konfigurace-s-apache/>
- [50] Seriál Malý průvodce konfigurací Apache. *Root.cz* [online]. [cit. 2019-05-13]. Dostupné z: <https://www.root.cz/serialy/maly-pruvodce-konfiguraci-apache/>
- [51] Queue — A synchronized queue class. *Python Software Foundation* [online]. [cit. 2019-05-13]. Dostupné z: <https://docs.python.org/3/library/queue.html>
- [52] Documentation: Introduction. *Bootstrap* [online]. [cit. 2019-05-13]. Dostupné z: <https://getbootstrap.com/docs/4.3/getting-started/introduction/>
- [53] EAGLE. *Autodesk* [online]. [cit. 2019-05-13]. Dostupné z: <https://www.autodesk.com/products/eagle/overview>
- [54] User authentication in Django: Documentation. *Django* [online]. [cit. 2019-05-13]. Dostupné z: <https://docs.djangoproject.com/en/2.2/topics/auth/>

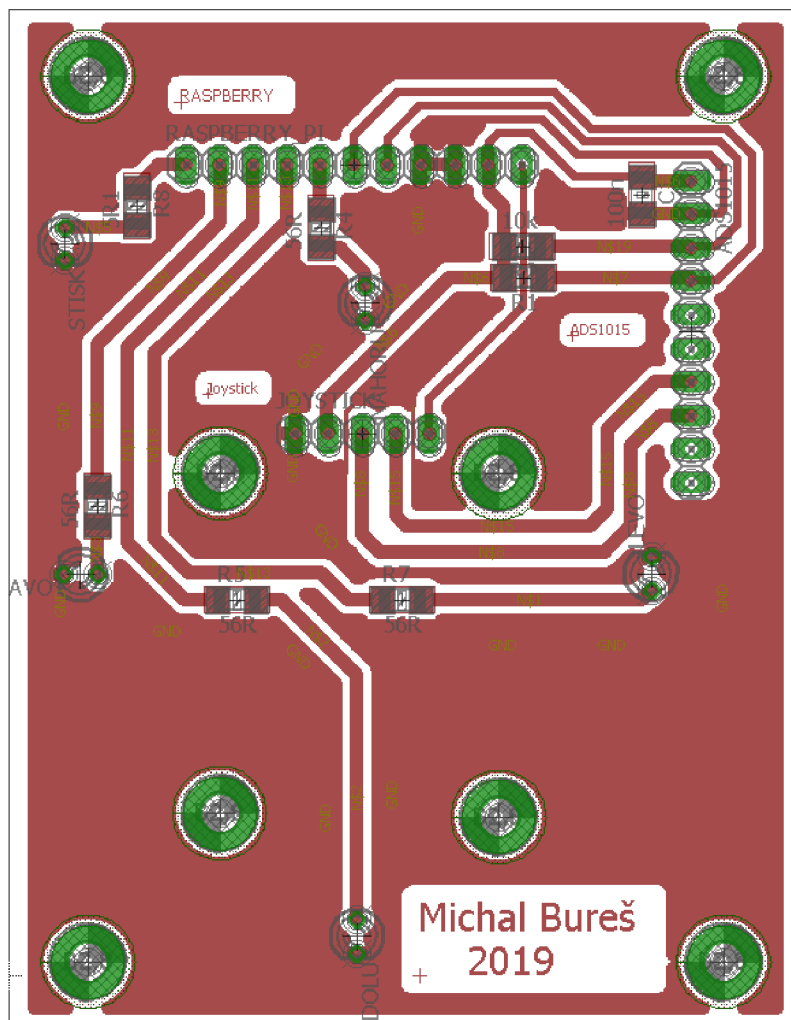
Seznam příloh

A. Schéma a návrh DPS v programu EAGLE.....	71
B. Webové rozhraní.....	72
C. Prezentační rozhraní	75
D. Realizované zařízení	76
E. Příložené CD.....	77

A. Schéma a návrh DPS v programu EAGLE



Obrázek 5-1: Schéma zapojení periférií.



Obrázek 5-2: Návrh DPS.

B. Webové rozhraní

FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH
TECHNOLOGIÍ

Elektronická informační tabule LCD

[Přihlášení](#) [Registrace](#)

Uživatelské jméno

Heslo

[Přihlášení](#)

Obrázek 5-3: Přihlášení do aplikace.

Elektronická informační tabule LCD

[Úvodní list](#) [Moje údaje](#) [Moje soubory](#) [Přehled uživatelů](#) [Odhlásit administrátora: pi](#)

Ctěný uživateli,

Vítej v Elektronické informační tabuli VUT.

Prostřednictvím této aplikace budeš moci zveřejňovat své soubory různým skupinám lidí.

Podporované formáty jsou *.BMP, *.PNG, *.JPG, *.PDF.

Tyto skupiny se dělí na:

- Obecné
- Zkoušky
- Jídelníček
- Pro studenty
- Pro učitele
- Brigády
- Školní akce

Veškeré informace, které uveřejníš budou veřejně dostupné v časovém rozmezí, které si určíš. Poté bude soubor smazán.
Máš-li nějaké dotazy či připomínky, můžeš mne kontaktovat na adrese: xburess31@stud.feec.vutbr.cz

Právě jsi se přihlásil: 13. května 2019 8:08

Obrázek 5-4: Úvod.

Elektronická informační tabule LCD

[Uvodní list](#)

Moje údaje

[Moje soubory](#)

[Přehled uživatelů](#)

[Odhlásit administrátora: pi](#)

Změna Tvých sobních údajů

Aktivní

Změnit údaje

Obrázek 5-5: Změna registračních údajů.

Elektronická informační tabule LCD

[Uvodní list](#)

[Moje údaje](#)

Moje soubory

[Přehled uživatelů](#)

[Odhlásit administrátora: pi](#)

Přehled událostí v El. nástěnce

Založit událost

Zde zadej text, který v tabulce hledáš.

Název události	Zveřejnit OD	Zveřejnit DO	Kategorie	soubor	Vlastník	Smazat?
dssfsd	25. března 2019	25. prosince 2019	Obecné	uzivatel_1/soubor.pdf	pi	
asdasd	25. března 2019	25. prosince 2019	Obecné	uzivatel_1/soubor.jpg	pi	

Obrázek 5-6: Hledání v založených Událostech.

Elektronická informační tabule LCD

[Uvodní list](#)[Moje údaje](#)[Moje soubory](#)[Přehled uživatelů](#)[Odhlásit administrátora: pi](#)

Založení události do EI. nástěnky

Název události:

OD kdy má být událost zveřejněna? :

DO kdy má být událost zveřejněna? :

Vyber, do které **kategorie** událost patří:

Nahraj soubor pro zveřejnění:

Obrázek 5-7: Založení Události.

Elektronická informační tabule LCD

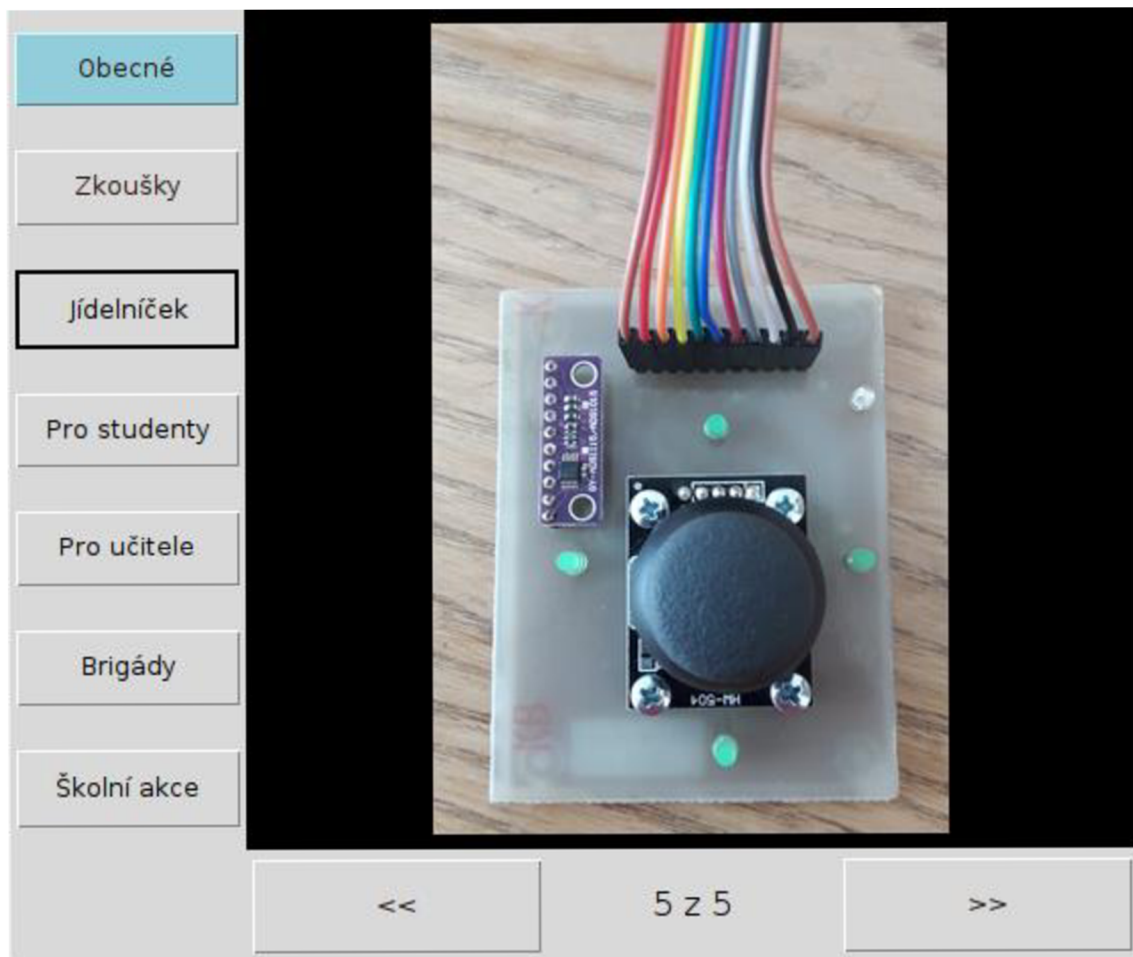
[Uvodní list](#)[Moje údaje](#)[Moje soubory](#)[Přehled uživatelů](#)[Odhlásit administrátora: pi](#)

Výpis registrovaných uživatelů

Uživatelské jméno	Křestní jméno	Příjmení	E-mail	Aktivní?	Datum registrace
pi	Michal	Bureš	xbures31@stud.feec.vutbr.cz	<input checked="" type="checkbox"/>	16. ledna 2019 17:42
Michal	Michal	Pokus	michal@michal.cz	<input checked="" type="checkbox"/>	1. dubna 2019 20:15

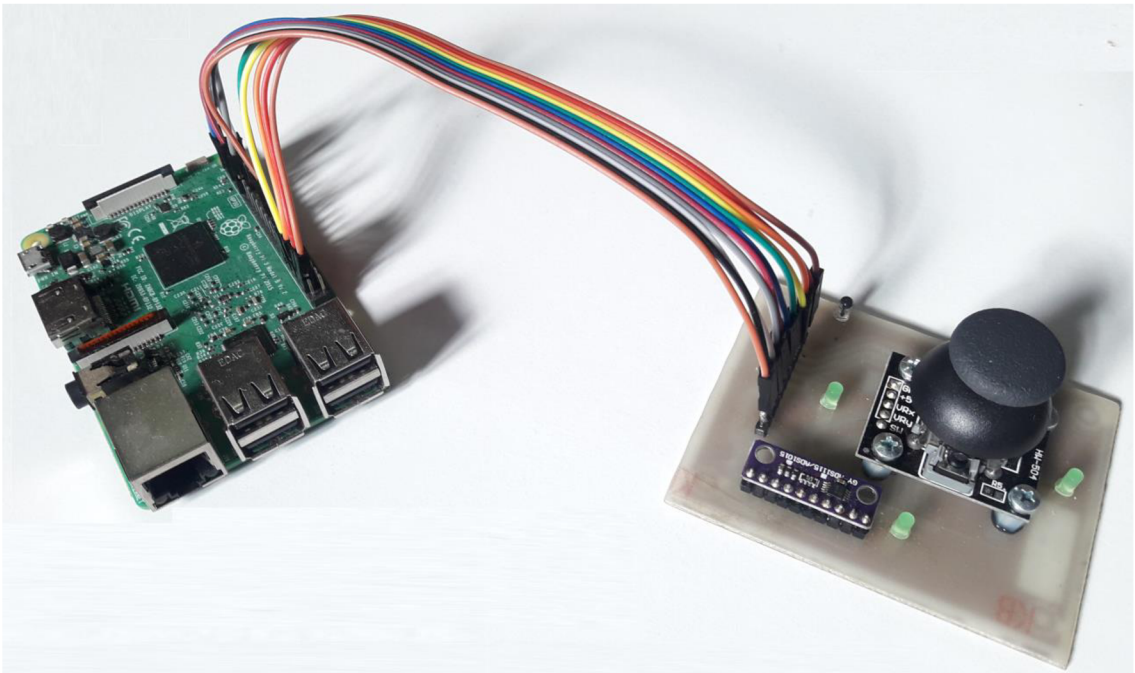
Obrázek 5-8: Přehled registrovaných uživatelů.

C. Prezentační rozhraní



Obrázek 5-9: Prezentační aplikace.

D. Realizované zařízení



Obrázek 5-10: Raspberry Pi s ovladačem.

E. Příložené CD

- Schéma a návrh DPS v programu EAGLE
- Programové vybavení Webového rozhraní
- Programové vybavení Prezentačního rozhraní
- Diplomová práce xbures31_2019_dip.pdf