



Bakalářská práce

Herní stanice založená na Raspberry Pi

Studijní program:

B0613A140005 Informační technologie

Studijní obor:

Aplikovaná informatika

Autor práce:

Martin Veselý

Vedoucí práce:

Ing. Jana Kolaja Ehlerová, Ph.D.

Ústav nových technologií a aplikované informatiky

Liberec 2023



Zadání bakalářské práce

Herní stanice založená na Raspberry Pi

<i>Jméno a příjmení:</i>	Martin Veselý
<i>Osobní číslo:</i>	M20000050
<i>Studijní program:</i>	B0613A140005 Informační technologie
<i>Specializace:</i>	Aplikovaná informatika
<i>Zadávající katedra:</i>	Ústav nových technologií a aplikované informatiky
<i>Akademický rok:</i>	2022/2023

Zásady pro vypracování:

1. Proveďte rešerši dostupných herních zařízení založených na minipočítačích upřednostňujících alternativní ovládání k myši a klávesnici.
2. Navrhněte vlastní zařízení spolu se sadou her.
3. Implementujte řešení, zajistěte konektivitu na Internet a zabezpečte zařízení na lokální i síťové úrovni.
4. Otestujte řešení, navrhněte vylepšení a možnosti rozšíření.

Rozsah grafických prací: dle potřeby dokumentace
Rozsah pracovní zprávy: 30 – 40 stran
Forma zpracování práce: tištěná/elektronická
Jazyk práce: Čeština

Seznam odborné literatury:

- [1] MONK, Simon. Raspberry Pi cookbook. Second edition. Beijing: O'Reilly, 2016. ISBN 978-1-491-93910-9.
- [2] NYSTROM, Robert. Game programming patterns. [London]: Genever Benning, [2014]. ISBN 978-0-9905829-0-8.
- [3] FULLERTON, Tracy. Workshop herního designu: průvodce tvorbou inovativních her. Přeložil Daniel DOLENSKÝ. V Praze: Nakladatelství Akademie múzických umění v Praze, 2021. ISBN 978-80-7331-568-9.

Vedoucí práce: Ing. Jana Kolaja Ehlerová, Ph.D.
Ústav nových technologií a aplikované informatiky

Datum zadání práce: 12. října 2022
Předpokládaný termín odevzdání: 22. května 2023

prof. Ing. Zdeněk Plíva, Ph.D.
děkan

L.S.

Ing. Josef Novák, Ph.D.
vedoucí ústavu

V Liberci dne 19. října 2022

Prohlášení

Prohlašuji, že svou bakalářskou práci jsem vypracoval samostatně jako původní dílo s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Jsem si vědom toho, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu Technické univerzity v Liberci.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti Technickou univerzitu v Liberci; v tomto případě má Technická univerzita v Liberci právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Současně čestně prohlašuji, že text elektronické podoby práce vložený do IS/STAG se shoduje s textem tištěné podoby práce.

Beru na vědomí, že má bakalářská práce bude zveřejněna Technickou univerzitou v Liberci v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů.

Jsem si vědom následků, které podle zákona o vysokých školách mohou vyplývat z porušení tohoto prohlášení.

Herní stanice založená na Raspberry Pi

Abstrakt

V bakalářské práci byla nejprve provedena analýza řešení podobných projektů, díky které bylo možné navrhnout vlastní koncept herní konzole spolu s jejím systémem. Po zvolení vhodného konceptu byla nejprve realizována softwarová stránka projektu, do které spadalo nastavení samotného Raspberry Pi, do něhož byl nainstalován desktopový OS, následně bylo zařízení připojeno k internetu, byl nastaven systém a byla zajištěna ochrana zařízení. Následovala realizace herního prostředí jakožto uživatelského rozhraní, ve kterém je možné herní konzoli nastavit, nebo především hrát hry. Práce se dále zabývala připojením, nastavením a testováním jednotlivých vstupů systému Grove, jež byl použit pro vstupy jako tlačítka či joystick. Cílem této části bylo propojení těchto modulů, k již vytvořenému uživatelskému rozhraní a odstranění chyb, aby bylo možné ovládat herní konzoli. Po této části byly moduly připravené pro instalaci do krytu konzole. V další části práce byl vytvořen dřevěný kryt pro herní konzoli, kterému předcházelo modelování krytu ve 3D programu, správné rozkreslení jednotlivých komponent a zvolení vhodných rozměrů. Po realizaci krytu byly všechny části spojeny do výsledné herní konzole. Ta byla následně otestována, poupravena dle výsledků testování a byly navrženy možnosti rozšíření bakalářské práce.

Klíčová slova: Raspberry Pi, herní konzole, herní prostředí, systém Grove

Game station based on Raspberry Pi

Abstract

In the bachelor's thesis, an analysis of the solutions of similar projects was first carried out, thanks to which it was possible to design a suitable concept of a game console along with its system. Once a suitable concept was chosen, the software side of the project was implemented, which involved setting up the Raspberry Pi itself. Into which the desktop OS was installed, then the device was connected to the internet and the system was set up to suit this work and to properly protect the device. This was followed by the implementation of the gaming environment as a user interface in which the game console can be set up or, more importantly, games can be played. The work also dealt with the connection, setup and testing of the different inputs of the Grove system, which was used for inputs such as buttons or joystick. The aim of this part was to connect these modules, to the already created user interface and to remove errors in order to control the game console. After this part, the modules were ready to be installed in the console housing. In the next part of the work, a wooden cover for the game console was created, which was preceded by modelling the cover in a 3D program, correctly drawing out the individual components and choosing the appropriate shape and size. Once the cover was realised, all parts were joined together to form the final gaming console. This was subsequently tested, modified according to the results of the testing and possibilities for extending the bachelor thesis were suggested.

Keywords: Raspberry Pi, game console, game environment, Grove system

Poděkování

Rád bych poděkoval vedoucí mé bakalářské práce Ing. Janě Kolajové Ehlerové, Ph.D. za odborné rady, užitečné připomínky a pomoc vždy, když byla třeba.

Obsah

Seznam obrázků	11
Seznam zkratek	12
1 Úvod	13
2 Rešerše	14
2.1 Teorie a motivace her	14
2.2 Projekty s podobným zaměřením - software	14
2.2.1 Recalbox	15
2.2.2 RetroPie	15
2.3 Projekty s podobným zaměřením - herní konzole	15
2.3.1 RetroPie Arcade Cabinet	15
2.3.2 Recalbox konzole	16
2.3.3 PiGRRL 2	16
3 Použitý hardware	17
3.1 Raspberry Pi 4	17
3.2 Waveshare 10.1" LCD displej	17
3.3 Rozhraní Grove	18
3.4 Reproduktory Verk E-02A	18
3.5 Grove Base HAT	18
3.6 Grove – Thumb Joystick	19
3.7 Grove – Dvojité tlačítko	19
3.8 Grove – Ultrazvukový senzor vzdálenosti	20
3.9 Grove – Vibrační modul	21
3.10 Grove – Mini PIR pohybový senzor	21
3.11 Grove – Vícebarevná flash LED (5 mm)	21
3.12 Zapojení modulů a pinout	22
4 Software	24
4.1 Nastavení RPI	24
4.1.1 Příprava systému	24
4.1.2 Nastavení ssh připojení	24
4.1.3 Výběr a instalace knihoven	25
4.1.4 Automatické aktualizace her a RPI	25
4.2 Herní prostředí konzole	25

4.2.1	Diagram a funkce herního prostředí	27
4.2.2	Ukázky herního prostředí	28
4.2.3	Spouštění a správa her	28
4.2.4	Databáze	29
4.3	Inicializace a kalibrace vstupů	29
4.3.1	Dvojitá tlačítka a joystick	30
4.3.2	Mini PIR pohybový senzor	30
4.3.3	Ultrazvukový senzor vzdálenosti	30
4.4	Vytváření miniatur her	30
4.5	Výběr a modifikace her	31
4.5.1	Had	31
4.5.2	Tetris	31
4.5.3	Pong	32
5	Návrh a výroba krytu pro konzoli	33
5.1	Návrh	33
5.1.1	Příprava pro navrhování krytu konzole	33
5.1.2	Návrh krytu konzole	33
5.2	Výroba	34
5.2.1	Výběr technologie	34
5.2.2	Materiál pro výrobu	35
5.2.3	Počátek výroby	35
5.2.4	Lepení dílů v celek	35
5.2.5	Opracování a lakování	36
5.2.6	Osazení krytu komponenty	36
5.2.7	Ukázky realizované herní konzole	36
6	Testování řešení	37
6.1	Testování jednotlivých částí řešení	37
6.1.1	Testování herního prostředí	37
6.1.2	Testování připojení k databázi	37
6.1.3	Testování her	38
6.1.4	Testování připojených modulů	38
6.2	Testování z pohledu hráče	38
7	Cenová kalkulace práce	39
8	Možnosti rozšíření práce	40
8.1	Rozšíření herního prostředí	40
8.2	Rozšíření her	40
8.3	Rozšíření ovládání herní konzole	41
8.4	Rozšíření sítě konzolí	41
9	Závěr	43
	Použitá literatura	44

Seznam obrázků

2.1	konzole RetroPie Arcade Cabinet [14]	16
2.2	konzole PiGRRL [1]	16
3.1	Raspberry Pi 4	17
3.2	Waveshare 10	17
3.3	System Grove	18
3.4	Grove Base HAT	18
3.5	Grove - Thumb Joystick	19
3.6	Grove - Thumb Joystick - diagram [10]	19
3.7	Grove - Dvojité tlačítko	20
3.8	Grove - Dvojité tlačítko - diagram [10]	20
3.9	Grove - ultrazvukový senzor vzdálenosti	20
3.10	Grove - ultrazvukový senzor vzdálenosti - diagram [10]	20
3.11	Grove - Mini PIR pohybový senzor	21
3.12	Grove - Mini PIR pohybový senzor - diagram [10]	21
3.13	Grove - Vícebarevná flash LED (5 mm)	22
3.14	Grove - Vícebarevná flash LED (5 mm) - diagram [10]	22
3.15	Umístění zapojení modulů do Grove Base HAT	22
3.16	Tabulka zapojení jednotlivých modulů [10]	22
3.17	Základní schéma zapojení konzole	23
4.1	Diagram funkcionality konzole	26
4.2	Diagram herního prostředí	27
4.3	Úvodní obrazovka se seznamem her	28
4.4	Zobrazení žebříčku hry, možnost spuštění hry	28
4.5	Tabulka pro uchování skóre her	29
5.1	Model konzole z boční strany	34
5.2	Model konzole z přední strany	34
5.3	Vyrobená herní konzole z boční strany	36
5.4	Vyrobená herní konzole z přední strany	36
7.1	Tabulka s cenovou kalkuací projektu	39
8.1	Síť více herních konzolí	41

Seznam zkratk

RPI	Raspberry Pi
OS	Operační systém
SSH	Secure Shell
USB	Universal Serial Bus
HDMI	High-Definition Multimedia Interface
NC	Not connected (není připojeno)

1 Úvod

Videohry jsou významnou součástí moderní kultury již několik desetiletí, první domácí konzole se objevily v 70. letech 20. století. Od té doby se video herní průmysl vyvinul a rozrostl do obrovského globálního trhu, který generuje miliardové příjmy. Současně s technologickým pokrokem se v herním průmyslu zvýšila nostalgie po klasických videohrách, což vedlo ke vzniku retro herních konzolí.

Raspberry Pi je univerzální jednodeskový počítač, který si získal oblibu jako platforma pro různorodé projekty, jak profesionální, tak pro zábavu. Skládání herních konzolí je toho příkladem. Cílem bakalářské práce je navrhnout a vyrobit konzoli, v rámci které dojde ke spojení výkonu počítače Raspberry Pi 4 s moduly systému Grove a vytvořit tak retro herní konzoli. Fyzická podoba konzole bude mířit do retro podoby a hlavním materiálem na krytu konzole bude dominovat dřevo. Výsledkem tedy bude návrh a výroba konzole, která poskytne maximální nostalgický herní zážitek s použitím moderních technologií.

V rámci bakalářské práce bude celý proces návrhu a výroby herní konzole zdokumentován včetně jednoduchého vlastního OS s využitím Raspberry Pi 4 a modulů systému Grove. Podrobněji bakalářská práce v první části seznámí čtenáře se základy a motivací her, základy herních konzolí založených na jednodeskových počítačích a se softwarem, který byl v těchto konzolích využitý. V druhé části práce jsou popsány moduly systému Grove a čtenář je seznámen s hardwarem, jež je v práci použitý. Následující část pojednává o vytváření grafického rozhraní jako OS konzole, popisuje přípravu RPI a informace o softwarové stránce této práce. Další část přibližuje, jakým způsobem byl realizován kryt herní konzole. V poslední části práce je popsáno testování řešení a dále jsou sepsány možnosti, jakými by bylo možné práci rozšířit.

2 Rešerše

2.1 Teorie a motivace her

Hraní her patří již od dávných dob do součásti lidské kultury. Dříve se jednalo samozřejmě o pohybové a stolní hry, dnes již převládá digitální forma her. Ty si v posledních dekáдах vybudovaly silnou pozici jak sociální, tak ekonomickou a často i edukativní. Hry mohou sloužit také pro experimentální účely, nejčastěji se pak jedná o hry, ve kterých hráč dělá rozhodnutí o budoucím ději, tento typ příběhových her často využívá tzv. motýlího efektu, kde má pak každé rozhodnutí vliv na následující děj a hra může skončit vícero způsoby. Takovým příkladem jsou hry Detroit: Become Human nebo Until Dawn.

Motivace her je ale především zdroj zábavy, odpočinku a dnes již i jako forma profesionálního sportu, respektive e-sportu. Oproti sledování filmů jsou hry interaktivní, obzvláště pokud lze jejich děj měnit dle hráče. Často se ve hrách vyskytují také různé hádanky či logické úlohy, které pomáhají hráče trénovat a rozvíjet jeho kreativitu a přemýšlení. Další kategorie her jsou hry založené na formě umění, cílem je tedy předat hráči co nejlepší grafický nebo emoční zážitek. Obecným cílem hry je ale motivovat hráče, aby u hry strávil dostatek času pro budování postupu, případně aby se zdokonaloval v nějaké konkrétní činnosti. [3]

Základní teorie her spočívá v poskytnutí souboru pravidel a cílů, jež musí hráč dodržovat, nebo kterých musí dosáhnout při hraní. Díky tomu je hráč motivován k naplnění těchto cílů. Hry je možné hrát jako samostatně (tzv. singleplayer), nebo ve více hráčích (tzv. multiplayer), kde hráči mohou kooperovat pro získání stanovených cílů. Hry mohou ale na hráče mít i negativní následky, konkrétně se může jednat o závislost, zdraví při nadměrném hraní a nedostatku spánku či vyčlenění se ze sociálního života.

O vývoj her se dnes starají týmy složené z různorodých odvětví, nejde totiž pouze o vývoj samostatné hry, ale také o marketing, dobré načasování vydání hry, kvalitní děj a hrátelnost. Na tu je kladen obrovský důraz, jelikož může potenciálního hráče nejrychleji demotivovat od hraní hry. [3]

2.2 Projekty s podobným zaměřením - software

V této podkapitole jsou uvedeny příklady retro herní prostředí s podporou RPI. V této práci bylo herní prostředí vytvářeno vlastní, ačkoliv uvedené varianty jsou typu open-source, jedná se o software, jež je poskytnut zdarma a jež má otevřený

zdrojový kód. Systém pro tuto práci byl vytvářen zcela individuálně pouze pro tuto práci, aby bylo možné nastavit vše dle vlastních potřeb.

2.2.1 Recalbox

Recalbox je open-source software, jedná se o retro herní systém, ve kterém je možné hrát retro hry. Systém je možné nainstalovat jak na RPI, tak na různé další konzole a PC, celkem je systém dostupný na více než 100 zařízení/OS, např. Apple II. nebo Nintendo DS, Wii atd.

Tento systém poskytuje jak herní prostředí, tak knihovnu s širokým výběrem her. Podporovány jsou také herní ovladače (drátové i bezdrátové). Systém je jednoduchý na instalaci, konfiguraci i následné hraní a je ideální volbou pro podobné projekty se základem na RPI. [22]

2.2.2 RetroPie

Stejně jako Recalbox v kapitole 2.2.1 je RetroPie open-source software sloužící jako retro herní prostředí pro RPI a PC. Oproti Recalboxu není podporovaný na tak velké škále OS, jen na cca polovině, ale podporuje také jak nové tak starší herní ovladače. Stejně jako v Recalboxu má systém předinstalované hry, je ale možné dodatečně další hry nainstalovat. Hráč si také může nastavit libovolné téma herního prostředí dle vlastní volby.

Lze jej nainstalovat do již nakonfigurovaného RPI s OS Debian, ale je také možnost nainstalovat pouze image tohoto systému, čímž lze ušetřit mnoho paměti. V RetroPie prostředí jsou kladeny požadavky i na úsporu energie a je tedy možné detailně nastavit systém dle požadavků uživatele. [23]

Další podobné systémy sloužící jako prostředí pro herní konzole založené na RPI jsou např. Batocera.linux a Lakka.

2.3 Projekty s podobným zaměřením - herní konzole

V této podkapitole jsou uvedeny příklady projektů, ve kterých byly vytvářeny herní konzole založené na RPI. Tyto konkrétní příklady byly zvoleny, jelikož se liší stylem využití a dva z příkladů využívají herní prostředí z kapitoly 2.2. Projektů s podobným zaměřením je ale více, liší se svou velikostí, vzhledem i použitým hardwarem.

2.3.1 RetroPie Arcade Cabinet

V rámci tohoto projektu byl využit herní systém RetroPie popsáný v kapitole 2.2.2. Jedná se o konzoli vyrobenou ze dřevěného obalu s monitorem o úhlopříčce 24 palců. Konzole je ovládaná tlačítky a joysticky a je možná hra dvou hráčů najednou. Tento projekt se z uvedených nejvíce podobá konceptu této práce, ať už svou velikostí, tak způsobem ovládání a dřevěnou konstrukcí. [14]

2.3.2 Recalbox konzole

Tato konzole je založena na prostředí Recalbox uvedeném v kapitole 2.2.1. Jedná se o komerční kit, který je možné zakoupit a sestavit. Výsledný produkt je retro herní konzole založena na RPI. Jedná se ale pouze o konzoli, nikoli kompletní sadu s monitorem.

Aktuálně dostupnou verzí je verze "Mini+", která obsahuje RPI Zero 2 W, plastový kryt, ovladač, kabely na propojení, microSD kartu a pouzdro. Jedná se o přenosnou konzoli, kterou pro hraní stačí připojit k monitoru a zdroji elektřiny. Cena tohoto produktu je 65,55 €. [21]

Druhou dostupnou verzí je verze "Premium", jež obsahuje RPI 4 (2 GB RAM), kabely na připojení, microSD kartu a výběr jednoho z plastových krytů kopírujících design herních konzolí Playstation 1, nebo Nintendo 1. generace. Tento kit neobsahuje žádné vstupní zařízení, hráč tedy musí využít vlastní ovladač či klávesnici. Cena tohoto kitu je 125,84 € ve verzi NESPI (Nintendo) a 110,72€ ve verzi PISTATION (Playstation). [12]

2.3.3 PiGRRL 2

PiGRRL 2 je přenosná konzole, jejíž jádro tvoří RPI. Jedná se o volnočasový projekt, o který se autor podělil na fórech a zveřejnil i CAD model pro možnost vytisknout si kryt a další součástky na 3D tiskárně. Mimo RPI je třeba zakoupit ještě několik dalších komponent jako tlačítka, gamepad, reproduktor, baterie, kabely a displej. Veřejné jsou i manuály na sestavení a zprovoznění této konzole. Herní prostředí této konzole je RetroPie zmíněný v kapitole 2.2.2. Výhodou této konzole je její malá velikost a tradiční retro design, který tato konzole kopíruje. Díky systému RetroPie je hráči poskytnuta rozsáhlá knihovna her. [1]



Obrázek 2.1: konzole RetroPie Arcade Cabinet [14]



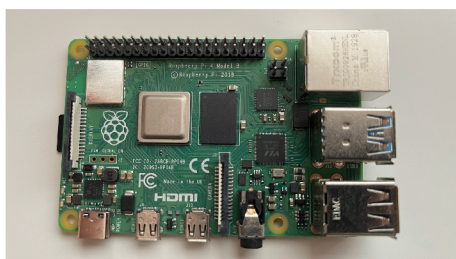
Obrázek 2.2: konzole PiGRRL [1]

3 Použitý hardware

V této kapitole je sepsán seznam prvků a příslušenství, které bylo v této práci využito. V rámci podkapitol modulů systému Grove je v každé podkapitole uvedeno zapojení daného modulu spolu s popisem testování funkcionality a připojení modulu k danému GPIO pinu.

3.1 Raspberry Pi 4

Pro tento projekt byl použit jednodeskový počítač Raspberry Pi 4 model B, jež disponuje čtyřjádrovým procesorem a 4 GB RAM paměti. Dále má integrované technologie Wifi a Bluetooth. Jako OS byl zvolen Raspberry Pi OS 64 bit ve verzi desktop. Jedná se o OS Raspbian, konkrétně verzi 11, tedy označení bullseye. Pro zapojení vstupů do RPI je možné využít 4 USB-A (2x 3.0), 2x Mini HDMI, Audio 3,5 mm Jack, Gigabitový ethernet a GPIO piny pro zapojení Grove Base HAT. Prívod energie je zajištěn přes USB-C port a systém a data jsou ukládána na microSD kartu. [19] [20]



Obrázek 3.1: Raspberry Pi 4



Obrázek 3.2: Waveshare 10

3.2 Waveshare 10.1" LCD displej

V konceptu herní konzole byl zvolen displej Waveshare s úhlopříčkou 10,1 palce (25,654 cm). Rozlišení displeje je 1024 x 600 pixelů. K RPI je připojen pomocí dvou kabelů, pro přenos obrazu je to HDMI – Mini HDMI, dále USB-A – microUSB kabel pro napájení displeje z RPI. Technologie použitá na tomto displeji je LCD. [26]

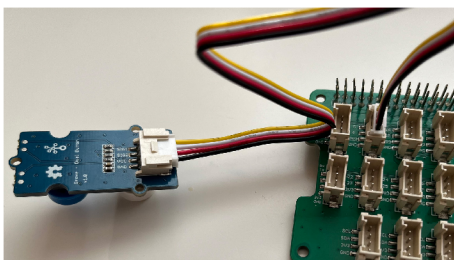
Displej je k RPI možno připojit i pomocí pinů na zadní straně. Tímto způsobem je možné ovládat displej i dotykově, což ale kvůli retro koncepci konzole není na

místě. Toto připojení v našem případě není možné ze dvou důvodů. Na RPI je třeba zanechat volné piny pro připojení Grove Base HAT a také je třeba, aby se k RPI dalo dostat kvůli případnému servisování konzole.

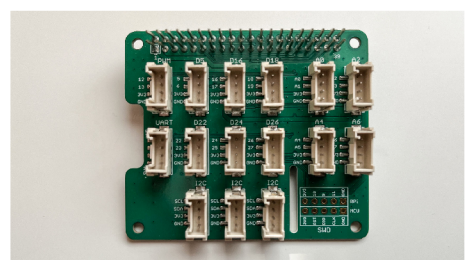
Displej je zobrazen na obrázku 3.2.

3.3 Rozhraní Grove

V této práci byl použit standardizovaný systém Grove. Jeho moduly byly zvoleny kvůli jednoduchému připojení za pomoci standardizovaných konektorů, širokému výběru modulů a různých zařízení, snadné implementaci, jelikož pro každý modul existuje minimálně jedna knihovna pro připojení k GPIO pinům na RPI. Výhodou rozhraní Grove jsou dále implementované knihovny, které lze pro připojení Grove zařízení využít.



Obrázek 3.3: Systém Grove



Obrázek 3.4: Grove Base HAT

3.4 Reproduktry Verk E-02A

Pro zajištění ozvučení byly zvoleny reproduktory Verk E-02A, jelikož je pro napájení využít USB-A kabel a mohou být napájeny přímo zdrojem RPI. K připojení pro přenos dat slouží 3.5 mm Jack kabel, zapojen do RPI. Kvůli omezenému místu v krytu konzole byly reproduktory vyjmuty z plastového krytu a do konzole byly vloženy pouze samostatně.

3.5 Grove Base HAT

Grove Base HAT je rozšiřující deska, která je připojená na RPI pomocí GPIO pinů. Námí využitá deska byla zakoupena od výrobce Seeed Studio. Deska obsahuje 15 Grove portů (6 digitálních, 4 analogové, 3 I2C, 1 PWM, 1 UART), GPIO digitální piny a SWD rozhraní pro připojení mikrokontrolerů (např. UART, I2C). Base HAT je pro tento projekt zásadní, jelikož je díky němu umožněno využít rozhraní Grove.

Base HAT a k němu připojené příslušenství jsou napájeny přes zdroj RPI. Pro připojení base HAT musí RPI obsahovat 40 GPIO pinů, na které je base HAT nasunut, tím pádem lze využít i jiné modely RPI. [10]

Grove Base HAT je zobrazen na obrázku 3.4

3.6 Grove – Thumb Joystick

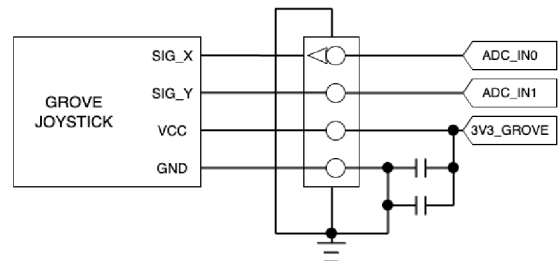
Grove Thumb Joystick je vstupní zařízení, jež obsahuje dva otočné ovládací prvky na získání pozice osy x a osy y. Samotný joystick je také tlačítkem, které lze stisknout a vyvolat nějakou akci. Joystick je připojen na analogový Grove konektor. Pro ovládání joysticku byla zvolena knihovna od Grove se třídou GroveThumbJoystick. Díky této knihovně je v cyklu volána funkce, která vrací hodnoty x a y, jež jsou v programu následně porovnávány, aby byl získán směr, kam joystick směřuje a mohla být volána správná funkce.

Grove Thumb Joystick je připojen na analogový konektor A0. Při testování funkčnosti byla vytvořena instance třídy GroveThumbJoystick, na kterou je následně volána funkce `.value()`. Z té je navracena hodnota souřadnic x a y. Stisk tlačítka je detekován, pokud hodnota souřadnice x nabyde hodnoty přesahující 900. [7]

Při realizaci ovládání konzole joystickem bylo zjištěno, že není možné, aby hráčův pohyb směřoval přesně po osách x a y. Proto byla po testování zvolena tolerance, o kolik se může hráčův pohyb vychylovat z této osy.



Obrázek 3.5: Grove - Thumb Joystick

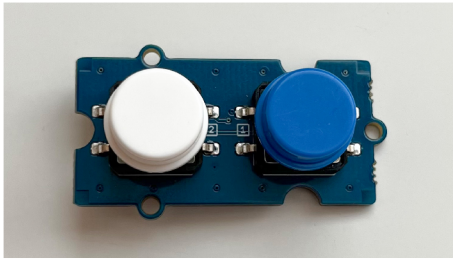


Obrázek 3.6: Grove - Thumb Joystick - diagram [10]

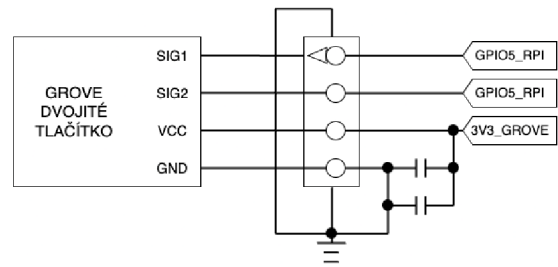
3.7 Grove – Dvojité tlačítko

Dvojité tlačítko se standardizací Grove bylo zvoleno jakožto další vstupní zařízení, díky kterému bude konzole ovládána. V tomto projektu jsou použité celkem 2 ks tohoto dvojitého tlačítka, celkem tedy 4 tlačítka. Dvojitá tlačítka byla konkrétně zvolená pro ušetření součástí, kabelů a pro snížení pořizovací ceny konzole.

Pro zajištění funkce zmíněných tlačítek byla nejprve použita knihovna Grove, která se ovšem ukázala jako nevhodná, poněvadž při jejím použití se začala objevovat chyba, která uváděla, že stisk tlačítka nebyl rozpoznán. Tento problém byl vyřešen použitím alternativní knihovny gpiozero. Konkrétně použitím její funkce `when_pressed()`. Poté vše fungovalo správně a byl tak odstraněn problém s nedetekovanými stisky tlačítek. [11] [4]



Obrázek 3.7: Grove - Dvojité tlačítko



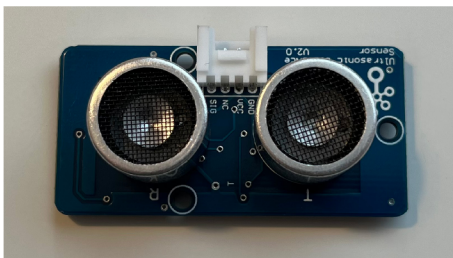
Obrázek 3.8: Grove - Dvojité tlačítko - diagram [10]

3.8 Grove – Ultrazvukový senzor vzdálenosti

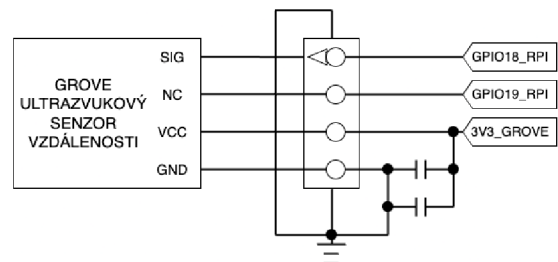
Ultrazvukový senzor vzdálenosti byl do tohoto projektu zvolen jako alternativní ovládání konzole mimo tlačítek a joysticku. Senzor dokáže měřit vzdálenost pomocí ultrazvukových vln s přesností do 2 mm. Pro měření vzdálenosti slouží přijímač a vysílač, jejichž hodnoty jsou nadále softwarově zpracovány a jejichž výsledkem je vzdálenost od tělesa či objektu. [8]

Senzor je připojen k RPI přes systém Grove, konkrétně k digitálnímu konektoru. Využita byla nejprve knihovna Grove, která ale stejně jako u tlačítka v kapitole 3.7, nefungovala správně a její výsledné hodnoty nebyly přesné. Zvolena proto byla alternativní knihovna gpiozero, konkrétně třída DistanceSensor, jejíž vstupy jsou 2 GPIO piny, na kterých je připojen přijímač a vysílač. Pro získání vzdálenosti je třeba na instanci této třídy zavolat funkci `.distance()`. Pro získání hodnoty v milimetrech je třeba vynásobit výsledek této funkce číslem 10. [8] [4]

Pro testování byl senzor připojen k digitálnímu konektoru D18. Přijímač byl tedy připojen k GPIO pinu 18 a vysílač k GPIO pinu 19. Byla vytvořena již zmíněná instance třídy DistanceSensor, jejíž funkcionality odpovídala očekávání a byla tak použita dále v tomto projektu.



Obrázek 3.9: Grove - ultrazvukový senzor vzdálenosti



Obrázek 3.10: Grove - ultrazvukový senzor vzdálenosti - diagram [10]

3.9 Grove – Vibrační modul

Vibrační modul nebylo možné s žádnou knihovnou zprovoznit. Sensor se spouštěl jen v polovině případů, navíc byly jeho vibrace slabé a v dřevěné konstrukci krytu by nebyly téměř zaznamenatelné. Modul byl tedy z projektu vyřazen a dále nevyužit. [9]

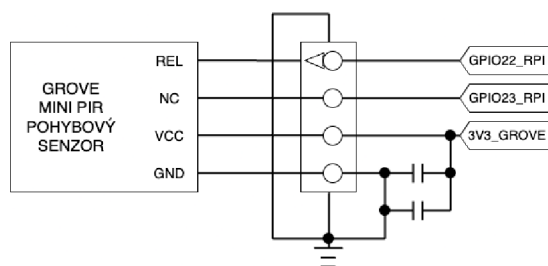
3.10 Grove – Mini PIR pohybový senzor

Z důvodu zaznamenávání pohybu okolo konzole, která měla být díky tomu probouzena z hibernace, byl použit pohybový senzor Grove - Mini PIR. Senzor využívá pro detekci pohybu infračervené záření a reaguje na změnu tepelné signatury, díky které je pohyb zaznamenáván. Maximální udávaná vzdálenost pro detekci pohybu je 3 až 5 metrů. [5]

Senzor je připojen k RPI kabelem systému Grove na digitální konektor Base HATu. Pro jeho inicializaci byla využita knihovna Grove, konkrétně třída GroveMiniPIRMotionSensor. Při testování byl tento senzor připojen na konektor D22. Při vytváření instance třídy bylo třeba uvést pouze GPIO pin 22. Následně byla volána funkce `.on_detect()`, která vracela bool proměnnou, jež se nastavila na true v případě, že byl zaznamenán pohyb. Funkčnost senzoru byla tímto ověřena. [5]



Obrázek 3.11: Grove - Mini PIR pohybový senzor



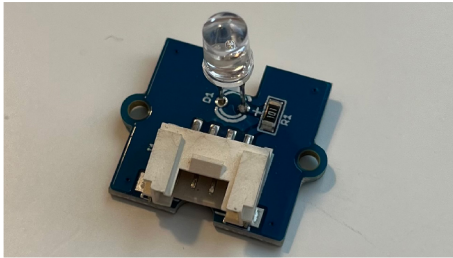
Obrázek 3.12: Grove - Mini PIR pohybový senzor - diagram [10]

3.11 Grove – Vícebarevná flash LED (5 mm)

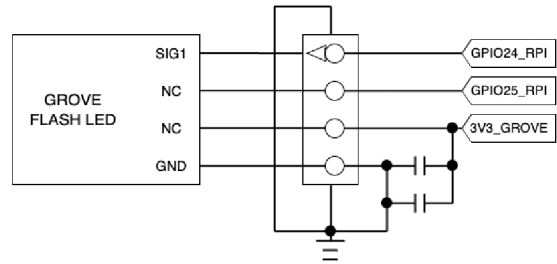
Vícebarevná flash LED byla pro tento projekt zvolena kvůli interakci s uživatelem. LED dioda využívá připojení přes systém Grove, konkrétně se připojuje na digitální konektor. Pro testování byla dioda zapojena na digitální GPIO pin 5. Využita byla knihovna Grove a její instance třídy GroveLed. Pro rozsvícení se na instanci této třídy volala funkce `.on()`, pro vypnutí diody následně funkce `.off()`. [6]

Při testování této diody bylo zjištěno, že nelze nastavit poměr rgb barev a při rozsvícení blikají postupně všechny barvy. Interakce s uživatelem může být tedy omezená, jelikož se předpokládalo, že díky nastavitelné barvě diody, bude možné ohlašovat více informací. Pro interakci s uživatelem za pomoci diod bylo zvoleno

překonání minimálně 3. nejlepšího skóre z žebříčku a tedy nový zápis do databáze. V takovém případě začnou diody po dobu 3 vteřin blikat.



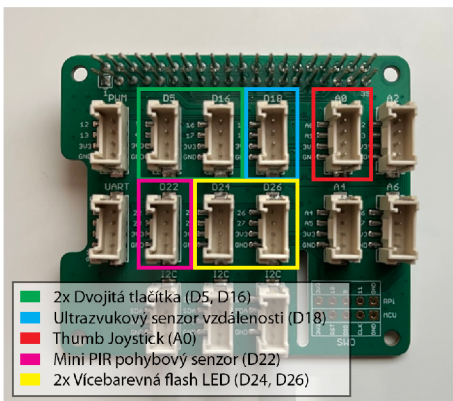
Obrázek 3.13: Grove - Vícebarevná flash LED (5 mm)



Obrázek 3.14: Grove - Vícebarevná flash LED (5 mm) - diagram [10]

3.12 Zapojení modulů a pinout

V této kapitole je uvedeno schéma zapojení modulů systému Grove do konzole. Dále je zobrazen Grove Base HAT s vyznačením, do kterých konektorů jsou jednotlivé moduly zapojeny. Detailní přehled jednotlivých propojení je patrný z tabulky na obrázku 3.16.



Obrázek 3.15: Umístění zapojení modulů do Grove Base HAT

Thumb Joystick SIG_X ADC_IN0 SIG_Y ADC_IN1 VCC 3V3 GROVE GND GND	Ultrazvukový senzor SIG GPIO18_RPI NC VCC 3V3 GROVE GND GND
Dvojitě tlačítko 1 SIG1 GPIO5_RPI SIG2 GPIO5_RPI VCC 3V3 GROVE GND GND	Mini PIR senzor REL GPIO22_RPI NC VCC 3V3 GROVE GND GND
Dvojitě tlačítko 2 SIG1 GPIO16_RPI SIG2 GPIO17_RPI VCC 3V3 GROVE GND GND	Flash LED 1 SIG GPIO24_RPI NC NC GND GND
Flash LED 2 SIG GPIO26_RPI NC NC GND GND	

Obrázek 3.16: Tabulka zapojení jednotlivých modulů [10]

4 Software

V této kapitole je popsáno základní nastavení RPI, využití knihovny a také návrh a vývoj OS pro herní konzoli. Následně kapitola popisuje mapování jednotlivých vstupů, výběr her a jejich bližší popis.

4.1 Nastavení RPI

4.1.1 Příprava systému

Na RPI byl nainstalován Raspberry Pi OS desktop ve verzi 11 s označením bullseye. Prvním základním nastavením bylo připojení k Wifi a následná aktualizace OS. Byl vytvořen uživatel „student“ s heslem, který byl přidán do skupiny sudoers, aby mohl provádět rootovské příkazy. Pro programování byl použit jazyk Python ve verzi 3.9.7. [19] [20] [17]

Zdrojové kódy včetně her a herního prostředí jsou uloženy na platformě Github v privátním repozitáři, ze kterého jsou stahovány jednotlivé hry a jejich aktualizace. Dále je zde umístěný i soubor s údaji pro přístup k databázi. Z důvodu zachování přístupu budou v rámci bakalářské práce všechny soubory umístěny ve veřejném repozitáři na platformě GitHub, jedinou výjimkou bude již zmíněný soubor s údaji pro přístup k databázi.

Z důvodů zajištění požadovaného zabezpečení nebude zařízení obsahovat vstupní zařízení (např. klávesnici), které by mohlo být hrozbou pro prolomení bezpečnostních prvků. V případě fyzického odcizení zařízení je možné bezpečnostní prvky prolomit a fyzicky tak napadnout zařízení.

4.1.2 Nastavení sshpřipojení

Pro vzdálenou správu RPI bylo třeba povolit ssh připojení, toho se docílilo přes raspi-config. Následně byla zjištěna ip adresa RPI a proběhl test připojení. Následně kvůli bezpečnosti připojení na RPI bylo třeba znemožnit přihlášení na roota přes heslo, proto bylo třeba přepsat sshd_config, konkrétně parametr PermitRootLogin nastavit na prohibit-password. Dále byl přidán ssh klíč pro bezpečnější připojení v domácí síti místo hesla. [20] [13]

4.1.3 Výběr a instalace knihoven

Pro tento projekt bylo potřeba nainstalovat několik knihoven, které byly následně využívány pro připojení modulů, vytváření OS pro herní konzoli a spouštění her. Tyto instalace byly realizovány přes správce balíčků apt, nebo pip3, který slouží pro jazyk Python. OS, hry a manipulace s připojeným hardwarem byla zajištěna jazykem Python a jeho knihovnami (sys, os...). [13]

Pro připojení, správu a tok dat z modulů systému Grove byly nainstalovány knihovny Grove a gpiozero. Knihovna Grove je oficiální knihovnou systému Grove, proto se s touto knihovnou snadno pracovalo a připojení bylo možné ve většině případů správně navázat. Knihovna gpiozero byla zvolena v případech, kdy nastal s Grove knihovnou problém, nebo řešila přístup ke vstupům snadněji. Tak tomu bylo např. u řízení tlačítek.

Pro vytváření OS pro herní konzoli a spouštění her byly zvoleny knihovny Tkinter a Pygame. Knihovna Tkinter je hojně využívána pro vytváření desktopových aplikací, kde je třeba řídit vstupy, reagovat na ně a pohybovat se v několika různých oknech na obrazovce. Vše přitom lze pohodlně graficky zpracovat. Pro vytváření her v jazyce Python je vhodné zvolit knihovnu Pygame, jelikož disponuje přehlednými manuály pro řešení této problematiky. Knihovna řeší i animace, které jsou ve hrách nezbytné. [24] [16]

4.1.4 Automatické aktualizace her a RPI

Pro automatické aktualizace her byl přidán crontab, který jednou za 20 minut ověří, zda není nahraná nová verze hry na portálu Github, kde jsou hry uloženy a odkud jsou distribuovány. Tohoto je docíleno příkazem „git pull“, který je v crontabu spouštěn.

Aktualizace OS a správce balíčků u RPI byla zřízena přes crontab, ve kterém je vykonáván skript, který jednotlivé části aktualizuje. Konkrétně jsou aktualizovány služby apt, pip3 a RPI. [13]

Přes crontab je také vykonáváno spouštění skriptu herního prostředí po spuštění RPI. Pro tento úkon byl použit parametr @reboot místo časových údajů a příkaz se vykoná ihned po spuštění systému. [20]

4.2 Herní prostředí konzole

Tato kapitola popisuje návrh, cíle a řešení herního prostředí pro konzoli.

Ke spuštění her a správu základního nastavení bylo třeba vytvořit pro uživatele jednoduchý OS. Cílem OS bylo přepínání mezi hrami, následně jejich spuštění, nastavení zvuku a uživatelského jména pro zapsání výsledků do databáze, ze které jsou zobrazovány data do žebříčků jednotlivých her.

K vytvoření herního prostředí byla použita knihovna Tkinter viz kapitola 4.1.3. OS konzole byl vytvořen ze tří oken. Hlavní menu, jež obsahuje výběr her formou horizontálního seznamu. Okno s nastavením, které poskytuje seznam, ve kterém lze spravovat nastavení konzole. Okno samotné hry, kde se nachází žebříček výsledků

konkrétní hry a tlačítko pro spuštění hry. OS byl navržen minimalistickým retro designem pro dosažení jednoduchého ovládání a vzhledu.

Hlavním skriptem, jež spouští herní prostředí je menu_rpi.py. Skript obsahuje funkce open_settings(), button_sound(), start_game() a game_change(). Zbytek kódu probíhá ve skriptu a běží v cyklu knihovny Tkinter .mainloop(). [24]

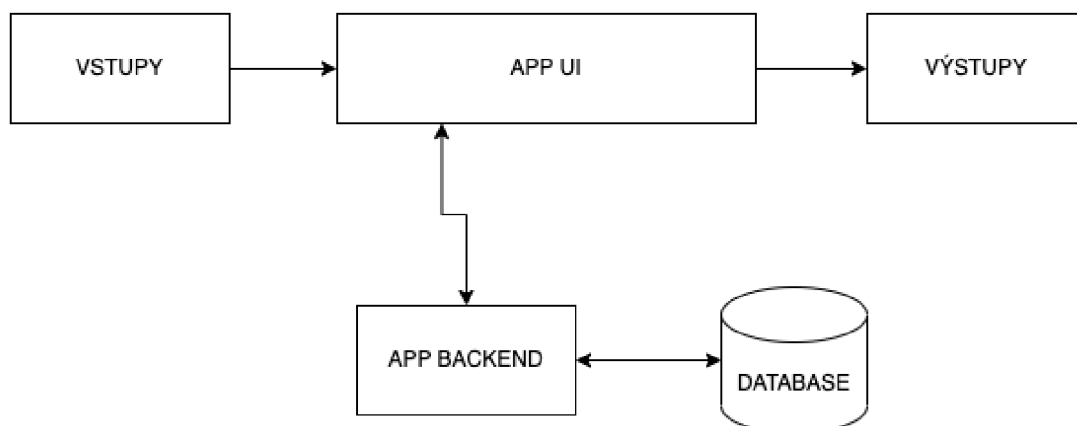
Nejprve je vytvořeno základní okno, jemuž jsou přidány parametry. Následně je vykonáno volání na inicializaci her a na hlavním okně je přidáno vykreslení miniatur her. Dále je provedena inicializace vstupů systému Grove a na Tkinter okno jsou přidány listenery na stisk tlačítek, které volají již uvedené funkce.

Open_settings() slouží k zobrazení nastavení konzole. V nastavení je zobrazen seznam, ve kterém je uživateli umožněn pohyb joystickem nahoru a dolů.

Button_sound() nahraje zvuk ve formátu mp3 ze složky /sounds/button/ a zvuk přehraje. Zvuky slouží pro lepší interakci mezi konzolí a uživatelem.

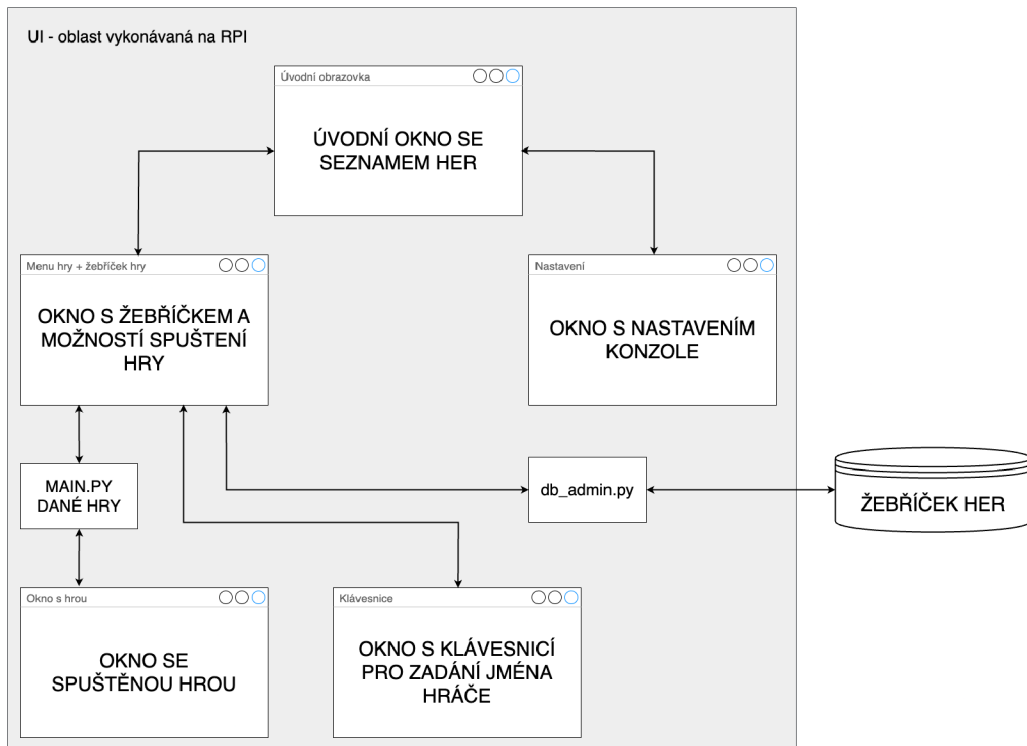
Funkce start_game() slouží pro zobrazení okna s miniaturou hry na pozadí, ve kterém je vykreslen žebříček 3 nejlepších výsledků hry a tlačítko s možností zahájení hry. Po spuštění hry je vytvořeno nové okno, ve kterém je spuštěna hra. Po ukončení hry je hráč navrácen zpět do nabídky s žebříčkem a zahájením hry.

Game_change() slouží pro přepínání her v seznamu na hlavní obrazovce. V systému je uchována globální proměnná "current_game"(int), díky které je v nabídce dynamicky měněna miniatura a název hry. Tato proměnná umožňuje i spuštění hry a zápis do databáze.



Obrázek 4.1: Diagram funkcionality konzole

4.2.1 Diagram a funkce herního prostředí



Obrázek 4.2: Diagram herního prostředí

Na obrázku 4.2 je zachycen diagram herního prostředí a jednotlivé možnosti, jak se uživatel může mezi okny pohybovat.

Při zapnutí konzole se hráč ocitne na úvodní obrazovce. Na úvodní obrazovce jsou zobrazeny miniatury jednotlivých dostupných her, které je možné joystickem posouvat vlevo či vpravo. Tak hráč může vybrat jednu z her. Z úvodní obrazovky je po výběru hry možnost zobrazit okno s žebříčkem této hry.

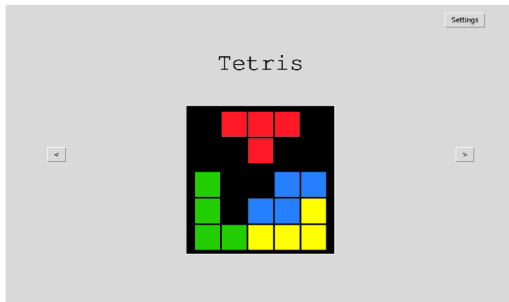
Žebříček obsahuje 3 nejlepší výsledky z dané hry. Na pozadí tohoto okna je v programu volána funkce ze souboru `db_admin.py` pro přístup do databáze, jak lze z diagramu vidět. Z databáze jsou záznamy čtené a zobrazované právě do žebříčku. Dále jsou do databáze zapisovány záznamy po ukončení hry, tedy pouze v případě, že hráčovo skóre je vyšší než alespoň 3. v žebříčku dané hry.

V tomto okně s žebříčkem skóre je možné spustit danou hru, či se naopak vrátit na úvodní obrazovku. Po spuštění hry je hráči zobrazena klávesnice, ve které se pohybuje joystickem a potvrzuje výběr tlačítkem. Po zadání své přezdívky je mu spuštěna daná hra. Po ukončení hry je hráč navrácen na obrazovku se skóre a může tak hru opakovaně spustit.

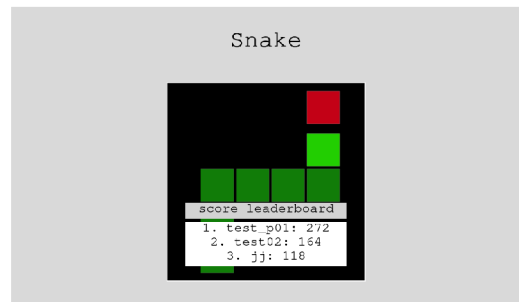
Z úvodní obrazovky je hráči druhým tlačítkem umožněno zobrazit okno s nastavením. V nastavení může měnit hlasitost a restartovat či vypnout konzoli.

4.2.2 Ukázky herního prostředí

V této kapitole jsou zobrazeny jednotlivé obrazovky herního prostředí. Jejich funkcionalita a návaznost je popsána v kapitole 4.2.1.



Obrázek 4.3: Úvodní obrazovka se seznamem her



Obrázek 4.4: Zobrazení žebříčku hry, možnost spuštění hry

4.2.3 Spouštění a správa her

Pro spuštění jednotlivé hry je volána funkce třídy `GameList`, která obsahuje instance třídy `Game`. Z instancí tříd `Game` je zvolena taková hra, jež si hráč v úvodní menu zvolil. Následně je již na konkrétní instanci třídy `Game` volaná funkce `play()`. Před spuštěním skriptu hry je třeba změnit postavení v adresáři do složky, ve které je hra uložena. Tento krok je nutné provést kvůli importování jednotlivých souborů uvnitř samotné hry. Při změně postavení v adresáři je do proměnné uloženo současné postavení v adresáři, na které se automaticky po ukončení hry program vrátí.

Hry jsou nastavené, aby jejich návratová hodnota bylo skóre, jehož hráč při hraní dosáhl. Důvod je jednotné zapisování do databáze mimo kód samostatných her pouze na jednom místě kódu ve třídě `GameList`.

Odebírání a přidávání her je provedeno dynamicky tak, aby bylo jednoduché pro správce konzole. Pro přidání nové hry je třeba přidat zdrojové kódy do složky `/games/"nová hra"`. Miniatura hry, zobrazovaná v hlavním menu, je třeba přidat do složky `/pict/"x.jpg"`. Název miniatury musí být číslo (číslování od 0) a formát obrázku `.jpg`, například tedy `"4.jpg"`. Po přidání souborů je třeba zapsat název a umístění souborů na nový řádek do souboru `games.txt`. Soubor je formátovaný následovně.

`GameName`, `GameMainLocation`, `PictureLocation`

`GameName` ... název hry

`GameMainLocation` ... absolutní cesta k umístění `.py` souboru s hrou *

`PictureLocation` ... absolutní cesta k umístění miniatury *

* absolutní cesty začínají v adresáři, kde je umístěn soubor `menu_rpi.py`

4.2.4 Databáze

V tomto projektu byla zvolena PostgreSQL databáze, která podporuje standard ANSI SQL:2008. Manipulace s daty byla tedy vykonávána v jazyce SQL. Pro hostování databáze byla zvolena platforma Render.com. Databáze obsahuje jedinou tabulku, díky které je umožněno ukládat záznamy o skóre z her. Jeden záznam obsahuje informaci o hře, hráči a skóre. Alternativou bylo i zvolení ukládání dat do paměti konzole. Volba databáze byla však vhodnější pro potencionální rozšíření práce, konkrétně pokud by bylo více konzolí, které by žebříčky sdílely mezi sebou. Rozšířený by mohl být také profil hráče, který by obsahoval informace jako seznam her, získané dovednosti, osobní žebříčky atd. [15]

Ukládání skóre a aktualizace databáze a žebříčků je provedena, pokud hráč dosáhne lepšího výsledku, než je již v databázi zapsán. V databázi jsou pro každou hru uvedeny pouze 3 nejlepší výsledky pro každou hru. V případě, že hráč dosáhne skóre shodného s 3. pozicí, jeho skóre není do databáze ani žebříčku zapsáno. Pokud je naopak skóre hráče vyšší, 3. pozice v žebříčku je automaticky odstraněna a do databáze je proveden zápis nového skóre. Třídění skóre je provedeno spolu s výpisem z databáze příkazem ORDER BY score DESC.

score	
game	str
player	str
score	int

Obrázek 4.5: Tabulka pro uchování skóre her

Databáze obsahuje jedinou tabulku "score", jež je zobrazena na obrázku 4.5. Tato tabulka obsahuje záznamy ze všech her. Jejimi parametry jsou "game", jež je název hry, "player", jež udává název hráče a "score", což je výsledné skóre hry. Takovéto schéma databáze bylo zvoleno, jelikož pro aktuální práci není třeba uchovávat více záznamů. Naopak bylo cílem zvolit a již připojit herní prostředí k databázi pro potencionální rozšiřování projektu v budoucnu, kde by bylo vytížení a použití databáze větší.

4.3 Inicializace a kalibrace vstupů

Nejen pro ovládání her, ale také OS, bylo třeba připojit moduly, jež pracují jako vstupy. Myš a klávesnici dle zadání práce konzole nevyužívá vůbec, pouze případně k servisování konzole, pokud by selhalo připojení přes ssh. Konkrétně je tedy konzole ovládaná tlačítky a joystickem. Toho bylo docíleno namapováním jednotlivých vstupů na klávesy, např. pokud bylo stisknuto tlačítko 1, softwarově byla volána funkce, jež simuluje stisk klávesy 'p'. Ve skriptu OS byla přiřazena příslušná funkce, jež se vykonala, pokud byla stisknuta klávesa 'p'. Mapování joysticku bylo navrženo

stejným způsobem, jeho vstup se převáděl na klávesy šipek. Řešení vstupů joysticku, a jejich kalibrace, je popsáno v kapitole 3.6.

Inicializace těchto modulů je provedena v souboru `grove_controls.py`, který obsahuje třídy pro inicializaci jednotlivých vstupů systému Grove. V tomto souboru jsou naimportovány knihovny `gpiozero`, `Grove` a `pyautogui`.

4.3.1 Dvojitá tlačítka a joystick

Třída `Button_controller` je inicializována zadáním pinu (`int`) a tlačítka, jehož spuštění má být vyvoláno (`char`). U třídy `Joystick_controller` je pro inicializaci třeba přidat pouze číslo portu (`int`), v případě našeho projektu je to 0, jelikož je joystick zapojený na analogový port A0. Po inicializaci těchto zařízení jsou jednotlivé vnitřní funkce volány automaticky. [11] [7]

4.3.2 Mini PIR pohybový senzor

Třída `PIRMotionSensor` je inicializována zadáním pinu (`int`), na kterém je Mini PIR senzor připojen do Grove Base HATu. Tato třída zajišťuje funkcionalitu pro uspávání displeje, pokud není před konzolí zaznamenán pohyb ze senzoru. Třída je naprogramovaná, aby po 30 sekundách bez detekce pohybu, vypnula HDMI port a vypnula tak displej. Jakmile je pohyb zaznamenán, port je opět zapnutý a displej konzole se rozsvítí. Potencionální využití by mohl senzor mít i u hraní her jako alternativní ovládání. [5]

4.3.3 Ultrazvukový senzor vzdálenosti

Třída `UltrasonicRanger` zajišťuje inicializování senzoru pro měření vzdálenosti za pomoci ultrazvukových vln. Pro inicializaci této třídy je třeba uvést pin, do kterého je senzor zapojen.

Ačkoliv proběhlo úspěšné testování knihovny `gpiozero` pro tento senzor, při pokusu o použití a zasazení senzoru do celku konzole, nastala znovu stejná již dříve vyřešená chyba. Při inicializaci instance třídy `DistanceSensor` byla vyvolávána výjimka chybného zapojení senzoru, konkrétně žádného signálu z pinu s přijímačem. Tento problém byl odstraněn vytvořením jiné třídy, jež byla popsána na oficiálních stránkách Seeed studia pro práci s tímto senzorem. Tato třída přistupuje a pracuje s jednotlivými GPIO piny. Po vytvoření instance této nově vytvořené třídy byly hodnoty zachycené senzorem správné. [8]

4.4 Vytváření miniatur her

Miniatury her byly vytvářeny v programu Adobe Illustrator ve verzi 2023 27.4.1. Při vytváření miniatur bylo kladeno na zachování retro designu ikon, jelikož hry i konzole samotná takový koncept dodržuje. Ikonu jsou tedy grafiky jednoduché, vyskytuje se na nich malé množství barev a objekty umístěné v nich jsou převážně

čtverce a základní tvary. Formát exportovaných souborů je .jpg a rozlišení miniatur je 500 x 500 pixelů při 150 PPI (pixel na palec).

4.5 Výběr a modifikace her

Herní konzole a její OS byl navržen tak, aby bylo možné přidávat další hry. V projektu byly vybrány a upraveny celkem 3 hry, které mají nastiňovat, jak konzole funguje a jak se ovládá. Všechny hry, stejně jako OS konzole, byly naprogramovány v jazyce Python.

Základ her byl získán z portálu GitHub z veřejných repozitářů. Následně byly hry upravovány a modifikovány. Cílem byla postupná optimalizace herního prostředí, aby bylo přidávání nových her co nejméně obtížné a nemusely se mnoho upravovat. Důvodem tohoto postupu je zpřístupnění konzole pro širší spektrum různých her získaných kdekoliv na internetu. Jakákoliv nová hra musela splňovat pouze využití knihoven Tkinter nebo Pygame.

4.5.1 Had

První hra zvolená pro tento projekt je hra Had. V této hře je úkolem hráče sbírat jablka, díky čemuž nabírá skóre a velikost jeho hada se zvětšuje, což činí hru složitější. Hráč má na začátku hry 5 životů. Jakmile had naboří do vlastního těla nebo do horní či spodní hrany pole, o jeden život přichází. Pokud hráči životy dojdou, hra končí. V případě, že je skóre hráče vyšší než třetí nejvyšší skóre v žebříčku, je jeho skóre uloženo do žebříčku.

Základ této hry byl získán z portálu GitHub z repozitáře Snake-Pygame od autora TkhiienLok. Hra byla následně přepracována a upravena pro konzoli. Nevyhovující kód byl přepsán do objektů, aby bylo možné opakované spuštění a inicializaci hry. Tato konkrétní hra byla zvolena kvůli jejímu využití knihoven Pygame a Tkinter. [25]

Jednalo se o první přidanou hru do konzole, na které bylo testováno chování konzole, konfigurace ovládání a nastavování přidání nových her. Po přidání této hry byl kód herního prostředí optimalizován, aby se zjednodušilo přidání dalších her.

4.5.2 Tetris

Další zvolenou hrou se stala hra Tetris známá z 80. let minulého století. Cílem hráče je skládat postupně padající tvary do řad, které po uzavření celé řady mizí a získat tak co nejvyšší skóre. Každý tvar lze přesouvat doleva a doprava a také jej otáčet o 90 stupňů vlevo či vpravo. Hra končí, jakmile je tvar umístěný mimo herní plochu směrem nahoru.

Základ této hry byl získán z portálu GitHub od autora rajatdiptabiswas z veřejného repozitáře tetris-Pygame. U této hry byla snaha co nejvíce zachovat originál, aby bylo zjištěno, jak je dále třeba optimalizovat herní prostředí pro pohodlné přidávání nových her. [18]

Změny původní hry byly nastaveny pouze v rozlišení okna a drobných změn pro správné zobrazení na displeji konzole. U opakovaného spouštění hry Tetris nastal stejný problém jako u hry had. Hra není uzpůsobena opakované inicializaci Pygame třídy a prostředí, bylo proto nutné hru upravit více, aby byl tento problém odstraněn.

4.5.3 Pong

Poslední hrou zvolenou pro demonstraci funkcionality konzole, byla zvolena hra Pong. Oproti hrám zmíněných v kapitolách 4.5.1 a 4.5.2 je tato hra určena pro dva hráče. Hra obsahuje dvě pálky a míček. Každý hráč pohybuje svou pálkou nahoru a dolů a snaží se dostat míček za soupeřovu pátku. Jakmile se to hráči podaří, získává bod. V tomto případě ovládá každý hráč svou pátku a snaží se získat bod. Pálka hráče vlevo je ovládána dvěma tlačítky a pálka hráče vpravo je ovládána pohybem joysticku nahoru a dolů.

Základ této hry byl získán z portálu GitHub od autora Amey-Thakur z veřejného repozitáře PONG-GAME. Tento kód byl upraven pro správné zobrazení na konzoli. Původní hra byla pro jednoho hráče, výsledný kód byl přepsán, aby bylo možné hru hrát se dvěma hráči proti sobě. [2]

Hru vyhraje hráč, který jako první dosáhne 11 bodů. V případě skóre 10:10 a více, musí hráč vyhrát o 2 body, tedy např. 12:10, nebo 16:14. Do tabulky skóre je zaznamenáván rozdíl, o kolik jeden z hráčů zvítězil. Maximální možné skóre je tedy 11 v případě výhry 11:0.

5 Návrh a výroba krytu pro konzoli

Design konzole byl nejprve navržen ve 3D prostředí, díky čemuž bylo snadné představit si výsledný produkt a zároveň odhalit různá úskalí při následné výrobě krytu. Jakmile koncept krytu splňoval cíle a funkčnost, byla zahájena jeho výroba.

Vyrobený kryt byl následně osazen RPI a všemi moduly. Následně bylo testováno, zda je konzole funkční a zda byl koncept vytvořen a vyroben správně. Po testování hardwarové stránky projektu byla testována konzole jako celek, testovalo se tedy propojení hardwaru a softwaru.

5.1 Návrh

5.1.1 Příprava pro navrhování krytu konzole

Při vytváření návrhu na vzhled konzole bylo nutné nejprve znát rozměry všech komponentů, ze kterých se výsledná konzole skládá. Bylo nutné znát také vstupy, které do konzole budou zapojené, konkrétně přívod elektrické energie a ethernetový kabel. V návrhu bylo nutné počítat také se servisováním konzole, proto musel být přístup k modulům, kabelům a RPI snadný. Pro 3D modelování byl v tomto projektu zvolen program Sketchup, verze 22.0.353. Postprodukční program pro správu textur, světel, odrazů a animací byl zvolen Artlantis, verze 6.0.1.4.

První částí návrhu bylo překreslení všech komponentů do 3D formátu. Důvodem bylo odhalení potencionálních problémů s usazením komponentů nebo servisováním konzole. Tímto bylo možné zobrazit po návrhu konceptu krytu kompletní konzoli i s osazením.

Prvním krokem bylo nalezení výkresu pro daný díl, pokud nebyl výkres veřejně dostupný, bylo nutné díl změřit. Veškeré modelování bylo v této fázi bez jakýchkoliv textur či barev. Tyto operace budou provedeny až pro finální koncept v programu Artlantis. Překreslený díl byl uložen a připraven na zasazení do výsledné skupiny.

5.1.2 Návrh krytu konzole

Následoval návrh samotného krytu, kde bylo prvním krokem získání inspirace a črtání nákresu na papír. Kvůli servisování si bylo třeba uvědomit, jaké díly budou pevně zasazené, tedy nerozebíratelné a jaké půjdou z konzole vyjmout pro správu hardwaru. Musel být zachován přístup k RPI. Toho bylo docíleno na zadní straně krytu. Přední strana obsahuje vstupy (tlačítka, joystick a senzory). Právě přední deska

spolu s připevněnými vstupy byla zvolena pro případné vyjmutí. Přístup do konzole byl tímto zajištěn vpředu i vzadu. Dalším nutným cílem při návrhu bylo zajištění pohodlného ovládání. Tlačítka a joystick musela být umístěna ergonomicky pro ruku hráče, jednalo se o dosah a vyvarování se hran. U senzorů muselo být zajištěno správné umístění. Oba senzory pro měření před sebou potřebují volný prostor, nic jim tedy nesmělo překážet ve směru měření.

Samostatný návrh krytu byl realizován dle již zmíněného nákresu. Kvůli výrobě bylo třeba kryt konzole rozdělit na přesné díly, ze kterých se reálný kryt bude skládat dohromady. Konkrétně se jednalo o vyjímatelné kusy, boky, spodní a vrchní stranu a díly uvnitř konzole. Vše bylo tedy navrženo tak, aby návrh opravdu odpovídal reálné výrobě. Jakmile byl každý z těchto dílů vymodelován, byl vytvořen poslední soubor, do kterého se postupně všechny díly krytu a následně komponenty přidávaly. Výsledkem tohoto souboru byl koncept konzole, jež byl připraven na postprodukcii.

Posledním krokem před realizací krytu konzole bylo přidání textur a vytvoření snímků, jež měly detailně přibližovat reálný vzhled konzole.



Obrázek 5.1: Model konzole z boční strany



Obrázek 5.2: Model konzole z přední strany

Na výše uvedených obrázcích 5.1 a 5.2 je zobrazen finální návrh krytu konzole, dle kterého byla následně realizována výroba.

5.2 Výroba

5.2.1 Výběr technologie

Při výběru technologie byla zvolená standardní výroba z přírodních materiálů oproti tisku na 3D tiskárně. Ačkoliv bylo třeba i tak 3D model vytvořit pro přesné modelování konceptu krytu konzole. Standardní postup byl zvolen pro lepší kvalitu zpracování, zejména pak kvůli retro stylu, jež měla konzole mít a také kvůli dřevěnému vzhledu, jež byl autorovi této práce subjektivně bližší.

5.2.2 Materiál pro výrobu

Pro výrobu konzole byla zakoupena překližka Multiplex bříza tloušťky 12 mm skládající se z 9 slepovaných vrstev. Základní formát této překližkové desky byl 1200 x 600 mm. Dále bylo zakoupeno lepidlo na dřevo Bison PU Max, tmel na dřevo značky Soudal, základová barva na dřevo Luxol, lazura na dřevo Luxol a tmavý matný lak na dřevo Luxol.

Veškeré potřebné vybavení bylo použito vlastní a nebylo třeba žádné dokupovat.

5.2.3 Počátek výroby

První fází výroby bylo vytištění plánů získaných z 3D modelu. Každý kus krytu konzole byl s tolerancí přibližně zakreslen na deskový formát. Tolerance byla zvolena kvůli následnému opracování přesnějšími nástroji. Po zakreslení všech tvarů stěn krytu byly tyto kusy vyřezány z formátu pomocí stolní kotoučové pily. Dalším krokem bylo následné zpracování jednotlivých tvarů přímočarou pilou, díky které bylo dosaženo vyšší přesnosti řezání hran. Pro dosažení této přesnosti byl zvolen jemný plátek na zakřivené řezy, který umožnil správnou manipulaci s dřevěnými deskami. Posledním krokem této části výroby bylo dobroušení hran pásovou bruskou, aby bylo zbaveno ostrých třísek a hran.

Do jednotlivých tvarů bylo třeba zakreslit a následně odstranit materiál pro jednotlivé díry tlačítek a senzorů. Technologie výroby jednotlivých děr byly nejprve otestovány na zbytkovém materiálu, jelikož vrstvy překližky není složité vytrhat a dosavadní postup tak zmařit. Pro tento krok byly zvoleny naostřené nástroje pro maximální přesnost a rezný efekt. Po načrtnutí díry byla využita frézka, vrtáky do dřeva s hroty a forstnerovy vrtáky pro vytvoření děr. Každá z vytvořených děr byla zároveň otestována příslušným zařízením, které do ní bylo zasunuto. Díry a jejich hrany byly v poslední fázi zabroušeny pro pohodlnou manipulaci při výrobě.

5.2.4 Lepení dílů v celek

Jednotlivé části krytu konzole byly nejprve bez lepení poskládány v tížený celek pro vizualizaci hran a jejich správnému doléhání. Po potřebných úpravách a správnému natočení byly zakresleny potřebné kóty pro následnou manipulaci a lepení částí. Při lepení kusů v celek bylo použito speciální lepidlo na dřevo, které bylo nanášeno na příslušné hrany, jež na sebe navazovaly a spojovaly se. Po nanášení lepidla na všechny části bylo třeba co nejrychleji srovnat stěny do správných úhlů a následně celý kryt konzole stáhnout svěrkami na následujících 24 hodin pro správné zasychání lepidla.

Dále bylo nejprve třeba obrousit lepidlo, jež ze spár vyteklo na stěny. Dalším krokem bylo použití tmelu na dřevo, které bylo vneseno do hran, ze kterých lepidlo vyteklo, nebo se do nich správně nedostalo. Po vytvrdnutí bylo třeba obrousit přebytek tmelu a pomocí brusky a smirkového papíru zahladit hrany na jejich finální vzhled. Tímto krokem byla dokončena konstrukce krytu konzole po konstrukční stránce.

5.2.5 Opracování a lakování

Před lakováním bylo nutné, aby byl kryt konzole jemně zbroušen a důkladně očištěn. Poté na něj byla nanесena základová barva dřeva, jež zároveň slouží pro ochranu dřeva. Po správném vyschnutí barvy byl tento krok zopakován ještě jednou. Dále byl kryt konzole znovu přebroušen a očištěn. Následně byl kryt konzole napuštěn lazurou na dřeva pro správnou odpudivost vody a ošetření dřeva. Posledním krokem byl nátěr krytu konzole matným vodou ředitelným lakem na dřeva. Po zaschnutí byl kryt konzole takto připraven pro osazení jednotlivými komponenty.

5.2.6 Osazení krytu komponenty

U zhotoveného krytu bylo třeba odebrat zadní stranu pro přístup k zapojení jednotlivých komponentů. Nejprve byly do krytu umístěny a upevněny vstupní zařízení (tlačítka, senzory), následně byl do krytu připevněn displej, reproduktory a samotné RPI. Následně byl veškerý hardware propojen s RPI. Kryt konzole byl takto osazen a připraven na testování nejprve funkčnosti, následně pak testování jako výsledku práce.

5.2.7 Ukázky realizované herní konzole



Obrázek 5.3: Vyrobená herní konzole z boční strany



Obrázek 5.4: Vyrobená herní konzole z přední strany

Na obrázcích, 5.3 a 5.4, je zobrazena herní konzole po její realizaci spolu se všemi vstupy a výstupy.

6 Testování řešení

V této kapitole je popsáno, jakým způsobem byly testovány jednotlivé části výsledné konzole. Takto otestované části byly následně spojeny v celek v podobě konzole, na jejíž testování bylo nahlíženo z pohledu uživatele tedy z pohledu hráče.

6.1 Testování jednotlivých částí řešení

V této kapitole je uvedeno, jakým způsobem byla testována softwarová a hardwarová část této práce. Při testování byla snaha o zachycení výjimek či špatných vstupů či výstupů funkcí, správná optimalizace programů a celkově správný chod konzole a jejího prostředí.

6.1.1 Testování herního prostředí

Herní prostředí bylo testováno průběžně spolu s jeho vytvářením mimo RPI. Pro testování sloužil manuální výpis informací a proměnných daných funkcí. Testována zde byla primárně data, jež byla předávána mezi funkcemi pro správný chod celého systému konzole, správná inicializace a spuštění knihovny Tkinter.

Po nasazení herního prostředí na RPI byla testována správná funkcionality všech užitých knihoven. Testováno bylo tak zobrazení na displeji, správné otevírání oken herního prostředí, správné přehrávání zvuků s možností nastavení hlasitosti.

6.1.2 Testování připojení k databázi

Pro testování databáze bylo zvoleno testování přes tzv. pytesty. Tedy pro každou využitou funkci, jež se připojovala k databázi, četla, zapisovala či pracovala s daty, byl vytvořen test. Testy porovnávají očekávanou hodnotu s návratovou hodnotou dané funkce. Pokud se hodnoty shodují, test je označen jako úspěšný. V případě neúspěšného testu je vypsána návratová hodnota funkce, jež je porovnána s očekávanou hodnotou. Díky tomuto výpisu je dále možné najít a odstranit chybu v dané funkci.

V konečném testování pro práci s databází byly splněny všechny testy pro dané funkce.

6.1.3 Testování her

U her bylo testováno především jejich opakovatelné spuštění. Problémem zde byla inicializace instance knihovny Pygame, jež musí být po skončení hry řádně ukončena. Pro samotné testování bylo užito výpisů do konzole a následné ladění kódu. Po odstranění chyb s opakovanou inicializací byla hra přidána do herního prostředí a následovalo testování z pohledu hráče. Při tomto testování bylo nastavováno především zobrazení, tedy velikost oken či objektů samotné hry. Testováno bylo i ovládání hry pomocí Grove příslušenství, zejména správná inicializace instancí tříd, jež s tímto příslušenstvím pracují.

Posledním krokem testování her byla návratová hodnota skóre, jež je návratová hodnota po ukončení každé hry. Testování této části proběhlo pomocí výpisů do konzole, kde bylo porovnáváno očekávané skóre z instance dané hry s hodnotou, jež byla posílána do hlavního programu. Kvůli práci s databází byl testován i návratový datový typ.

6.1.4 Testování připojených modulů

Testování modulů systému Grove bylo provedeno formou výpisu do konzole. Toto testování bylo realizováno nejprve zvlášť pro každý modul, viz kapitola 3. Následně bylo testováno zapojení všech Grove modulů najednou a jejich nezávislé fungování v herním prostředí a samotných hrách. Testovány byly i neobvyklé vstupy jako dlouhé podržení tlačítka či jeho několikanásobné zmáčknutí v rychlém sledu.

6.2 Testování z pohledu hráče

V této části byla herní konzole testována za účelem zlepšení uživatelského zážitku. Cílem bylo dosažení jednoduchého ovládání konzole a her. Při tomto testování byly získávány informace k případnému rozšíření této práce. Rozměr displeje v této práci byl po testování konzole zhodnocen jako adekvátní pro tento typ herní konzole.

Při testování ovládání byl důraz kladen na ergonomii ovládání a umístění jednotlivých vstupů konzole. Ovládání konzole bylo intuitivní, reakce vstupů byly nastaveny správně. Individuální by u této problematiky byla ergonomie, jelikož přesná pozice joysticku a tlačítek byla zvolena dle uvážení autora. Tedy hráči se znatelně menší či větší dlaní by mohlo být po chvíli ovládání nepohodlné.

7 Cenová kalkulace práce

V této kapitole jsou vypsány jednotlivé položky spolu s pořizovací cenou, za kterou byly nakoupeny. Bylo použito vlastní opotřebované nářadí, které tím pádem není v kalkulaci zahrnuto. Stejně tak využitý software nebylo nutné pořizovat, jelikož byl již vlastněn a také není předmětem kalkulace. Nicméně k využitému softwaru existují volně přístupné bezplatné alternativy, jež je možné využít.

produkt	cena za kus	počet kusů	cena celkem
Grove - joystick	112,0 Kč	1	112,0 Kč
Grove - mini PIR senzor	155,0 Kč	1	155,0 Kč
Grove - flash LED	39,0 Kč	3	117,0 Kč
Grove - vibrační modul	80,0 Kč	1	80,0 Kč
Grove - dvojité tlačítko	66,0 Kč	2	132,0 Kč
Waveshare 10,1" LCD displej	2 699,0 Kč	1	2 699,0 Kč
Grove - ultrazvukový senzor vzdálenosti	120,0 Kč	1	120,0 Kč
Grove - 5ks kabelů s konektory	109,0 Kč	1	109,0 Kč
Raspberry Pi 4 model B - 4GB RAM	1 650,0 Kč	1	1 650,0 Kč
Grove - Base HAT	309,0 Kč	1	309,0 Kč
Raspberry Pi USB-C napájecí zdroj	247,0 Kč	1	247,0 Kč
Reproduktor Verk E-02A	119,0 Kč	1	119,0 Kč
Raspberry Pi 32GB microSDHC sada	318,0 Kč	1	318,0 Kč
Multiplex překližka bříza 12 mm tl. - formát	799,0 Kč	1	799,0 Kč

celková cena konzole	6 966,0 Kč
----------------------	------------

Obrázek 7.1: Tabulka s cenovou kalkulací projektu

V tabulce zobrazené na obrázku 7.1 jsou zobrazeny jednotlivé ceny nakoupených produktů. V případě Raspberry Pi 4 modelu B 4 GB RAM a Grove - Base HAT se jedná o aktuální cenu, za kterou se zařízení prodávají, jelikož bylo toto příslušenství po dobu projektu zapůjčené od vedoucí práce.

Nevyužitý byl v této práci Grove - vibrační modul a jedna Grove - flash LED, cena výsledného zařízení by tak byla možná ponížít o 119 Kč.

8 Možnosti rozšíření práce

8.1 Rozšíření herního prostředí

Herní prostředí konzole by bylo možné rozšířit přidáním více oken. Přidáno by tedy mohlo být okno pro zobrazení podrobnějších žebříčků, ve kterém by byly obsaženy dodatečné informace o průběhu hry či konkrétním postupu daného hráče. Dále okno se seznamem her, ve kterém by bylo možné hry odstraňovat, třídit či přidávat do oblíbených. Na úvodní obrazovce by tedy přibyla možnost zobrazit pouze oblíbené hry. Případným rozšířením herního prostředí by bylo přidání herních profilů, ve kterém by mohl hráč získávat trofeje či sběratelské předměty z jednotlivých her, ale mohl by se také spojit s ostatními hráči, jež by si mohl přidat do přátel a vyznat je tak k soupeření ve skóre u konkrétní hry.

Další rozšíření by jistě bylo v sekci nastavení, aby si mohl hráč herní prostředí více přizpůsobit dle vlastních potřeb. Aktuálně lze v konzoli nastavit pouze zvuk, jinak nastavení obsahuje možnost vypnutí či restartování konzole. Do nastavení by tedy bylo přidáno nastavení velikosti písma, volba barevného zobrazení herního prostředí a nastavení jasů displeje.

Z pohledu herního prostředí jako programu by bylo možné herní prostředí optimalizovat a zajistit automatické přizpůsobení více typům displejů a hardwaru. Herní prostředí by tak bylo možné použít i na jiném zařízení, jelikož aktuální herní prostředí bylo vytvářeno přímo na hardware zmíněný v kapitole 3.

8.2 Rozšíření her

Rozšíření v části her by bylo možné z pohledu přidávání nových her. Hry by tak bylo možné přidat dynamicky z portálu či knihovny a stačilo by je pouze nainstalovat. Aktuální přidávání her je vytvořeno za účelem demonstrace a simulace funkcionality herní konzole a hry jsou přidávány a upravovány manuálně.

Pro knihovnu her by mohl být využit jeden ze softwarů zmíněných v kapitole 2.2. Jelikož je software zdarma dostupný, bylo by možné naprogramovat herní prostředí tak, aby se na něm tyto hry daly spustit. Knihovna by tak byla značně rozšířena o nové hry.

8.3 Rozšíření ovládání herní konzole

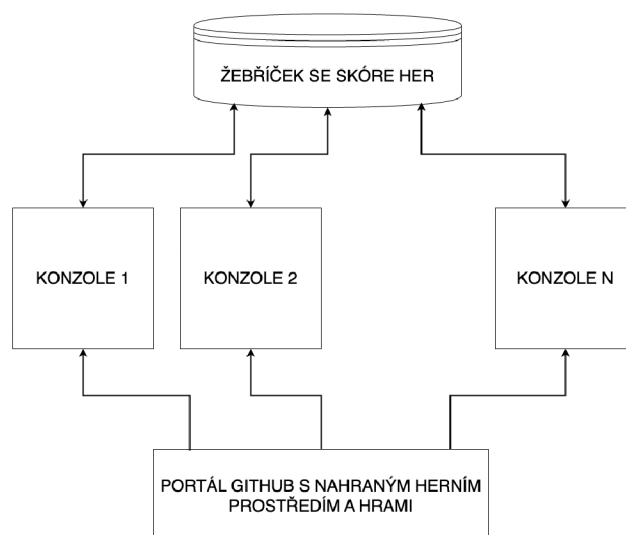
Herní konzole by mohla být rozšířena o drátový ovladač, jež by byl zapojen přes port USB. Na ovladači by se nacházel joystick, sada tlačítek a případně i malý mikrofón a reproduktor. Herní ovladač by byl přínosem, pokud by hráč zobrazoval herní prostředí na větší obrazovku, v případě samostatné konzole by byla výhoda ovladače pouze pohodlnější ovládání. Ovladač by byl vytvořen v retro stylu podobně jako samotná herní konzole.

Možnost rozšíření této práce by bylo vytvoření webové či mobilní aplikace pro správu her či uživatelského profilu popsaného v kapitole 8.1. Zde by si hráč mohl přidávat nové hry, vyzývat přátele k soutěžím o skóre či sdílet získané trofeje s ostatními hráči. Toto rozšíření by naopak s retro stylem konzole nekorespondovalo.

Rozšířit by také tato práce šla využitím senzoru pro měření vzdálenosti. Ačkoliv byl tento senzor v práci uveden, otestován a připraven na použití, nebyla nalezena správná aplikace, ve které by mohl být senzor využit pro zjednodušení ovládání konzole či hry. Pro využití tohoto senzoru v této práci bylo zamýšlené přidat hru tanky, gorily, střelbu lukem nebo bowling. Žádná z uvedených her nebyla dostupná ve správné variantě pro zamýšlené hraní. Pro demonstraci funkcionality senzoru tak byla přidána aplikace vypisující vzdálenost, jež senzor zachycuje.

8.4 Rozšíření sítě konzolí

Koncept konzole, jež je popsána v této práci, byl vytvořen a připraven, aby z něj bylo možné vytvořit síť nezávisle fungujících totožných konzolí. Herní konzole z této práce by se vyrobila ve více kusech a herní prostředí by bylo pouze staženo z portálu Github. Bylo by pouze nutné nastavit jednotlivé RPI v každé konzoli, konkrétně by byla nezbytná instalace knihoven, nastavení crontabu a zajištění zabezpečení.



Obrázek 8.1: Síť více herních konzolí

Nezávislá funkcionálna konzolí a jejich připojení k portálu Github a databázi je zobrazeno na diagramu na obrázku 8.1. Dalším cílem vytvoření této sítě konzolí by bylo hraní her mezi těmito zařízeními. Hráči by tam mohli hrát stejnou hru spolu či proti sobě v reálném čase. Pro tuto síť byla připravena i databáze, která by byla využita pro sdílení žebříčků napříč více stanicemi.

9 Závěr

Cílem bakalářské práce bylo nejprve seznámení se s projekty zabývající se herními konzolemi, jejichž základem jsou jednodeskové počítače. Dalším cílem bylo sestavit retro herní konzoli, která měla uživateli zajistit nostalgický zážitek ve spojení s použitím moderních technologií. Konzole měla mít zajištěnou konektivitu k internetu společně se zabezpečením.

Vytvořit takovou konzoli obnášelo několik důležitých kroků od analýzy možností hardwarového řešení, softwarového řešení, návrhu a výrobu krytu konzole až po testování jednotlivých částí. Součástí provedení bylo připojení konzole k internetu, zajištění automatických aktualizací a zabezpečení. Dále byla zpracována cenová kalkulace za navrženou konzoli.

V neposlední řadě pak bylo uvažováno o možnostech rozšíření jednotlivých částí. V rámci samotného projektu se autor musel vypořádat s celou řadou problémů, které bylo nutno vyřešit. Výsledné zařízení bylo finálně testováno a shledáno jako vhodné pro běžné použití.

V rámci bakalářské práce byl tak vytvořen jeden funkční prototyp herní konzole. Stojí za zvážení, zdali se v budoucnu nevěnovat již avizovaným možnostem rozšíření a vytvořit síť těchto herních konzolí, které by sdílely herní prostředí, databázi a následně i hraní her. Pro takové rozšíření bylo herní prostředí, hry i databáze připravena.

Použitá literatura

- [1] adafruit. *PiGRRL 2 - Raspberry Pi Game Console*. 2023. URL: <https://www.thingiverse.com/thing:1277483> (cit. 22.02.2023).
- [2] Amey-Thakur. *PONG-GAME*. URL: <https://github.com/Amey-Thakur/PONG-GAME/tree/main> (cit. 03.04.2023).
- [3] Tracy Fullerton. *Workshop herního designu. průvodce tvorbou inovativních her*. 1. vyd. V Praze: Nakladatelství Akademie múzických umění, 2021. ISBN: 978-80-7331-568-9. (Cit. 15.12.2022).
- [4] *GPIO Zero*. *gpiozero*. 2019. URL: <https://gpiozero.readthedocs.io/en/stable/index.html> (cit. 12.01.2023).
- [5] *Grove - PIR Motion Sensor*. 2023. URL: https://wiki.seeedstudio.com/Grove-PIR_Motion_Sensor/ (cit. 22.02.2023).
- [6] *Grove - Red LED*. 2023. URL: https://wiki.seeedstudio.com/Grove-Red_LED/ (cit. 22.02.2023).
- [7] *Grove - Thumb Joystick*. 2023. URL: https://wiki.seeedstudio.com/Grove-Thumb_Joystick/ (cit. 22.02.2023).
- [8] *Grove - Ultrasonic Ranger*. 2023. URL: https://wiki.seeedstudio.com/Grove-Ultrasonic_Ranger/ (cit. 22.02.2023).
- [9] *Grove - Vibration Motor*. 2023. URL: https://wiki.seeedstudio.com/Grove-Vibration_Motor/ (cit. 22.02.2023).
- [10] *Grove Base Hat for Raspberry Pi*. 2023. URL: https://wiki.seeedstudio.com/Grove_Base_Hat_for_Raspberry_Pi/ (cit. 12.01.2023).
- [11] *Grove Dual Button*. 2023. URL: <https://wiki.seeedstudio.com/Grove-Dual-Button/> (cit. 22.02.2023).
- [12] *Kit "Premium"Recalbox - NESPI/PiStation*. 2023. URL: https://www.kubii.com/fr/kits- consoles-recalbox/3252-1663-kit-premium-recalbox-3272496312197.html%5C#/version_kit_premium_recalbox-pistation (cit. 16.12.2022).
- [13] Simon Monk. *Raspberry Pi cookbook. Software and hardware problems and solutions*. 2. vyd. Sebastopol: O'Reilly, 2013. ISBN: 978-1449365226. (Cit. 23.03.2023).
- [14] Mike Murray. *BUILD A RETROPIE BARTOP ARCADE CABINET*. 2023. URL: <https://www.thegeekpub.com/4506/build-retropie-bartop-arcade-cabinet/> (cit. 15.12.2022).

- [15] *PostgreSQL*. 2023. URL: <https://render.com/docs/databases> (cit. 12. 03. 2023).
- [16] *Pygame. pygame documentation*. URL: <https://www.pygame.org/docs/> (cit. 29. 01. 2023).
- [17] *Python. Python 3.9.16 documentation*. 2023. URL: <https://docs.python.org/3.9/> (cit. 29. 01. 2023).
- [18] rajatdiptabiswas. *Tetris*. URL: <https://github.com/rajatdiptabiswas/tetris-pygame> (cit. 26. 03. 2023).
- [19] *Raspberry Pi. Raspberry Pi OS*. 2023. URL: <https://www.raspberrypi.com/software/> (cit. 22. 02. 2023).
- [20] *Raspberry Pi. Raspberry Pi Documentation*. 2023. URL: <https://www.raspberrypi.com/documentation/> (cit. 22. 02. 2023).
- [21] *Recalbox Mini+ Kit*. 2023. URL: <https://www.kubii.com/en/kits/3480-recalbox-mini-kit-3272496308312.html> (cit. 16. 12. 2022).
- [22] *Recalbox Wiki*. 2023. URL: <https://wiki.recalbox.com/en/home> (cit. 15. 12. 2022).
- [23] *RetroPie Docs*. 2022. URL: <https://retropie.org.uk/docs/> (cit. 15. 12. 2022).
- [24] *TkDocs*. 2022. URL: <http://tkdocs.com/tutorial/index.html> (cit. 29. 01. 2023).
- [25] TkhienLok. *Snake-Pygame*. URL: <https://github.com/TkhienLok/Snake-Pygame/commits/master> (cit. 20. 03. 2023).
- [26] *Waveshare 10.1"LCD displej, IPS, 1024×600, HDMI, dotykový, rezistivní*. 2023. URL: <https://rpishop.cz/lcd-oled-displeje-hat/2101-waveshare-101-hdmi-lcd.html> (cit. 19. 01. 2023).

A Přílohy

Zdrojové kódy herního prostředí a her jsou umístěny ve veřejném repozitáři GitHubu na tomto [odkazu](#).