|  |  |  |
| --- | --- | --- |
|  | | |
| **Závěrečná studijní práce**  **dokumentace** | | |
| **Wi-fi webradio** | | |
| Štěpán Kuzník | | |
|  | | |
|  | |  |
| **Obor:** | 18-20-M/01 INFORMAČNÍ TECHNOLOGIE  se zaměřením na počítačové sítě a programování | |
| **Třída:**  **Školní rok:** | IT4  2020/2021 | |

##### Poděkování

*Děkuji panu učiteli Ing. Petru Grussmannovi za cenné rady, panu učiteli Mgr. Marcelu Godovskému za pomoc se součástkami.*

Prohlašuji, že jsem závěrečnou práci vypracoval samostatně a uvedl veškeré použité   
informační zdroje.

Souhlasím, aby tato studijní práce byla použita k výukovým účelům na Střední průmyslové   
a umělecké škole v Opavě, Praskova 399/8.

V Opavě 10.1.2021

*podpis autora práce*

**ANOTACE**

Projekt se zabývá přehráváním hudby a rádiových stanic přes Wifi. Skládá se z hardwarové a softwarové části. Základ hardwarové části se tvoří hlavně z nízkonákladového mikročipu Wifi esp8266. Zařízení se pomocí mikročipu připojí na webovou stránku, kterou jsem vytvořil, z které přehrává hudbu z rádiových stanic.

Programová část webradia je řešena v jazyce Arduino, což je kombinace jazyků C a C++. Druhou část projektu tvoří webová aplikace vyvinutá v jazyce PHP. Tato aplikace umožňuje vzdáleně ovládat zařízení

OBSAH

[Úvod 5](#_Toc61206094)

[1 Výroba wifi webradia 6](#_Toc61206095)

[2 Využité technologie 7](#_Toc61206096)

[2.1 Hardware 7](#_Toc61206097)

[2.1.1 Seznam součástek 7](#_Toc61206098)

[2.1.2 NodeMCU 7](#_Toc61206099)

[2.2 Software 7](#_Toc61206100)

[2.2.1 Jazyk Arduino 7](#_Toc61206101)

[2.2.2 Webová aplikace 7](#_Toc61206102)

[3 Způsoby řešení a použité postupy 8](#_Toc61206103)

[3.4 Webová aplikace 8](#_Toc61206107)

[4 Výsledky řešení 9](#_Toc61206108)

[4.2 Podoba webové aplikace 9](#_Toc61206110)

[Závěr 10](#_Toc61206111)

[Seznam použitýCH INFORMAČNÍCH ZDROJů 11](#_Toc61206112)

[Seznam příloh 12](#_Toc61206113)

Úvod

V této dokumentaci Vás uvedu do projektu, popíšu Vám jaké technologie jsem využil, princip fungování, jaké jsem použil postupy a způsoby řešení.

Cílem mého projektu bylo sestavit jednoduché rádio, které bude přehrávat rádiové stanice přes HTML odkaz s použitím co nejméně součástek. Rádio jsem si vybral, jelikož se mi to zdálo jako jednoduchý projekt, který je pro mě i zároveň zajímavý

Základ mého projektu tvoří mikrokontroler NodeMCU, který má v sobě zabudovaný Wi-Fi čip ESP8266, díky kterému se mé rádio může připojit přes Wi-Fi na internetovou stanici rádia. O výstup zvuku se stará reproduktor, který jsem získal rozebráním starých beden. Kód jsem dělal ve Visual Studio Code, kde jsem použil rozšíření PlatformIO, kterému jsem dal přednost před ArduinemID, kvůli jednoduchosti přidání knihoven a kvůli větší přehlednosti.

Díky čipu ESP8266 se rádio pomocí knihovny Wi-Fi manager připojí k internetu. Aby nám rádio mohlo přehrát nějakou rádiovou stanici, musíme zadat její URL do webové aplikace.

Program je napsaný v jazyce Arduino, který je kombinace jazyků C a C++. O Připojení k internetu se postaralo ESP8266. O vypnutí rádia se stará tlačítko.

# Výroba wifi webradia

První část mého projektu představovalo sestavení samotného zařízení. Protože se jednalo o mou první zkušenost se sestavováním integrovaných obvodů, výběru a nákupu součástek předcházel výzkum dané problematiky. Nejdříve jsem se rozhodl jako základ mého zařízení použít vývojovou desku NodeMCU, která obsahuje WiFi modul ESP8266 pro komunikaci s webovou aplikací.

Poté jsem přešel k programování, abych mohl zařízení spojit přes WiFi s webovou aplikací a umožnit tak jeho ovládání. Pro tento účel jsem zvolil programovací jazyk Arduino 1.8.13.

Původně jsem chtěl použít kodek VS1053, který umožňuje dekódovat mnoho zvukových formátů – např. MP3, WMA, ACC atd., ale nemohl jsem ho sehnat, tak jsem projekt přizpůsobil k fungování bez něj.

# Využité technologie

## Hardware

### Seznam součástek

* Reproduktor
* NodeMCU ESP8266
* tlačítko

### NodeMCU

Základ mého zařízení tvoří vývojová deska NodeMCU, která již obsahuje WiFi modul ESP8266 pro komunikaci s webovou aplikací, má zabudovanou CP2102 a tlačítka.

Specifikace:

* Power supply: 5V
* 2x 15pinové rozšiřující headery, propojují všechny piny modulu ESP-12E

## Software

### Jazyk Arduino

Jazyk Arduino je až na drobné úpravy velmi podobný jazyku C nebo C++. Byl přímo vytvořen k programování integrovaných obvodů. Program sloužící k ovládání hardwarových součástek jsem napsal právě v tomto jazyce.

### HTML

Hypertext Markup Language (HTML) je v informatice název značkovacího jazyka používaného pro tvorbu webových stránek, které jsou propojeny hypertextovými odkazy. HTML je hlavním z jazyků pro vytváření stránek v systému World Wide Web, který umožňuje publikaci dokumentů na Internetu.

# Způsoby řešení a použité postupy

## Webová aplikace

Pro asynchronní web server jsem vybral knihovnu ESPAsyncWebServer

Lib\_deps = ottowinter/ESPAsyncWebServer-esphome @ 1.2.7

Knihovny nutné ke spuštění serveru. Server se spustí na portu 80.

#include “ESPAsyncWebServer.h“

#include “ESPAsyncTCP.h“

AsyncWebServer server(80);

Funkce, která zkontroluje SPIFFS systém, do kterého budeme ukládat html soubor.

if(!SPIFFS.begin()){

Serial.println(“An Error has occurred while mounting SPIFFS“);

return;

}

Výpis IP adresy, na které deska funguje.

Seriál.println(WiFi.localIP());

Přidání knihovny pro WiFiManager, díky kterému bude pro uživatele snadné se připojit k internetu.

#include “ESPAsyncWiFiManager.h“

Přidání DNS serveru

DNSServer dns;

Konfigurace WiFiManageru nastavení názvu (RadioAP)

AsyncWiFiManager wifiManager(&server,&dns);

wifiManager.autoConnect(“RadioAP“);

Seriál.println(“connected“);

Seriál.println(WiFi.localIP());

http request /url vráti url rádia ve formátu string, příkaz strcpy pak převede string na char

server.on(“/url“, http\_POST, [](AsyncWebServerRequest \*request) {

stanice = request->arg(“url“).c\_str();

strcpy(URL, stanice.c\_str());

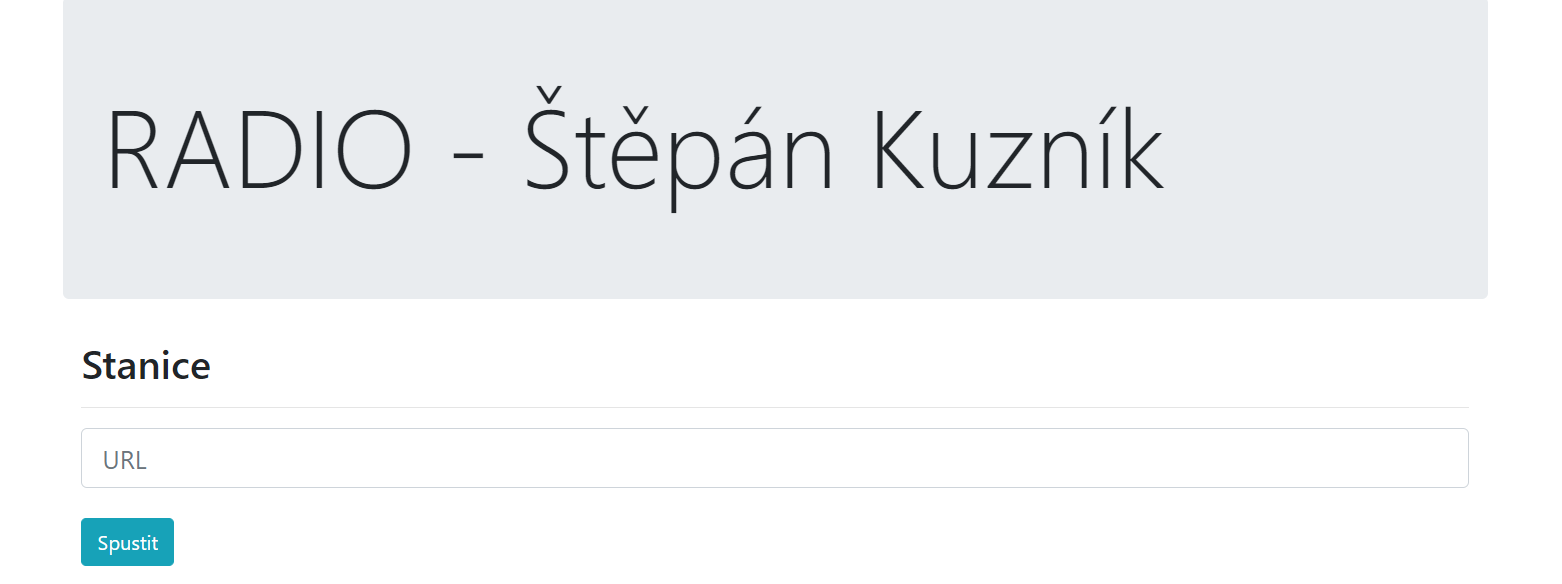
request->send\_P(200, “text/json“, “{\“result\“:\“ok\“}“);

});

# Výsledky řešení

## Podoba webové aplikace

Webová aplikace není nijak složitá, obsahuje pouze název projektu, textové pole, kde se zadává odkaz na rádiovou stanici a tlačítko pro spuštění.



**Závěr**

Cílem projektu bylo vytvořit jednoduché funkční rádio s možností ovládat jej prostřednictvím webové aplikace. Tento cíl byl splněn a zařízení i webová aplikace fungují. Při vývoji mě napadl nespočet možných změn, které by zařízení vylepšily, jako například napájení z baterie nebo umístění zařízení do krabičky. Rád bych také přidal více možností nastavení do webové aplikace, jako například možnost upravit hlasitost.

Seznam použitýCH INFORMAČNÍCH ZDROJů

[1] Knihovna WifiManager

github.com/alanswx/ESPAsyncWiFiManager

[2] Knihovna ESP8266Audio

github.com/earlephillhower/ESP8266Audio

[3] Knihovna WebServer

github.com/ottowinter/ESPAsyncjWebServer-esphome

[4] ESPAsyncWebServer s použitím SPIFFS

https://randomnerdtutorials.com/esp32-web-server-spiffs-spi-flash-file-system/

[5] https://www.hackster.io/RoboticaDIY/esp8266-how-to-make-wi-fi-radio-5bee14

Seznam příloh

č. 1 Titulní list

č. 2 Čestné prohlášení

č. 3 Poděkování

**Příloha č. 1: Titulní list**