

Katedra informatiky
Přírodovědecká fakulta
Univerzita Palackého v Olomouci

BAKALÁŘSKÁ PRÁCE

Foucaultovo kyvadlo

Ukládání dat z Foucaultova kyvadla a jejich prezentace



2021

Vedoucí práce: doc. RNDr. Mi-
chal Krupka, Ph.D.

Pavel Cvek

Studijní obor: Aplikovaná informatika,
kombinovaná forma

Bibliografické údaje

Autor: Pavel Cvek
Název práce: Foucaultovo kyvadlo (Ukládání dat z Foucaultova kyvadla a jejich prezentace)
Typ práce: bakalářská práce
Pracoviště: Katedra informatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci
Rok obhajoby: 2021
Studijní obor: Aplikovaná informatika, kombinovaná forma
Vedoucí práce: doc. RNDr. Michal Krupka, Ph.D.
Počet stran: 31
Přílohy: 1 CD/DVD
Jazyk práce: český

Bibliographic info

Author: Pavel Cvek
Title: Foucault pendulum (Data saving from Foucault pendulum and their presentation)
Thesis type: bachelor thesis
Department: Department of Computer Science, Faculty of Science, Palacký University Olomouc
Year of defense: 2021
Study field: Applied Computer Science, combined form
Supervisor: doc. RNDr. Michal Krupka, Ph.D.
Page count: 31
Supplements: 1 CD/DVD
Thesis language: Czech

Anotace

Webová prezentace založená na sběru dat z Foucaultova kyvadla Přírodovědecké fakulty Univerzity Palackého v Olomouci, která veřejnosti a studentům vhodným způsobem prezentuje naměřený pohyb spolu s informacemi o chování Foucaultova kyvadla.

Synopsis

Web presentation based on data collected from the Foucault pendulum at Palacký University in Olomouc, which provides the measured movement to the public and students in an appropriate manner, together with information about the behavior of the Foucault pendulum.

Klíčová slova: Foucaultovo kyvadlo, Arduino, webová prezentace, JavaScript, PHP

Keywords: Foucault Pendulum, Arduino, web presentation, JavaScript, PHP

Děkuji panu prof. RNDr. Tomášovi Opatrnému, Dr. za ochotu a spolupráci a také děkuji panu doc. RNDr. Michalovi Krupkovi, Ph.D. za odborné vedení této práce.

Místopřísežně prohlašuji, že jsem celou práci včetně příloh vypracoval/a samostatně a za použití pouze zdrojů citovaných v textu práce a uvedených v seznamu literatury.

datum odevzdání práce

podpis autora

Obsah

1	Úvod	8
2	Arduino	9
2.1	Komunikace	9
2.1.1	Arduino	9
2.1.2	NodeMcu	10
2.1.3	Zpracování na straně webového serveru	11
2.2	HTTP a HTTPS	11
3	Použité technologie	12
3.1	PHP	12
3.2	JavaScript	12
3.2.1	Knihovna CK Editor	12
3.3	HTML	13
3.4	CSS	13
3.5	MariaDB	13
3.6	Arduino	13
4	Webová prezentace	13
4.1	veřejná část webové prezentace	14
4.1.1	Vypsání naměřených dat	15
4.1.2	Načtení naměřených dat	15
4.1.3	Animace kyvadla	16
4.1.4	Grafy	17
4.1.5	Galerie	19
4.1.6	Zapomenuté heslo	20
4.2	Administrace webové prezentace	20
4.2.1	Příchozí data	21
4.2.2	Uživatelé	21
4.2.3	Editace textu	22
4.2.4	Galerie	22
4.3	Kontrola fotodetektorů	22
5	Uživatelská příručka	23
5.1	Veřejná část webové prezentace	23
5.1.1	Navigační menu	23
5.1.2	Přijatá data	23
5.1.3	Možnosti galerie	24
5.2	Administrace webové prezentace	25
5.2.1	Přihlášení	25
5.2.2	Zapomenuté heslo	25
5.2.3	Změna hesla	25
5.2.4	Kontrola kyvadla	25

5.2.5	Příchozí data	25
5.2.6	Editace uživatelů	26
5.2.7	Editace titulní strany	26
5.2.8	Editace galerie	26
Závěr		27
Conclusions		28
A Testování aplikace		29
B Obsah přiloženého CD		30
Literatura		31

Seznam obrázků

1	Ukázka měření pomocí programu Lighthouse	14
2	Ukázka grafu Stokesových parametrů	19
3	Ukázka grafu směru kmitů	20
4	Ukázka zasílaného emailu	23
5	Animace pohybu Foucaultova kyvadla	24
6	Ukázka galerie	24

Seznam zdrojových kódů

1	Zahájení seriové komunikace s NodeMcu na Arduinu	10
---	--	----

1 Úvod

V prostorách u hlavního schodiště přírodovědecké fakulty Univerzity Palackého se nachází Foucaultovo kyvadlo. Jeho pohyb, který je sledován pomocí čtyř laserových svazků a fotodetektorů zpracovává Arduino, které současně zajišťuje jeho další pohyb pomocí elektromagnetu v podlaze. Správný chod a seřízení senzorů Foucaultova kyvadla v prostorách přírodovědecké fakulty má na starost pan prof. Tomáš Opatrný.

Cílem této práce je možnost naměřená data Arduinem ukládat a následně je prezentovat pomocí vytvořené webové prezentace foucault.optics.upol.cz. Uživatel může zjistit více o účelu Foucaultova kyvadla a dozvědět se informace o Foucaultovu kyvadlu v přírodovědecké fakultě. Součástí webové prezentace je animace pohybu Foucaultova kyvadla, kterou je možné sledovat v průběhu času, nebo sledovat aktuální pohyb Foucaultova kyvadla a grafy zobrazující časový vývoj směru kmitů a Stokesovy parametry. Webovou prezentaci jsem vytvořil v českém a anglickém jazyce.

Součástí je také administrace webové prezentace, kde je možné sledovat příchozí data v nezměněné podobě, měnit obsah webové prezentace, vytvářet uživatele a současně jim posílat pozvánky do administrátorské části.

Vzhled webové prezentace jsem se snažil vytvořit podle webových stránek Univerzity Palackého v Olomouci, zejména záhlaví a zápatí. Logo jsem použil podle grafického manuálu Univerzity Palackého v Olomouci. Celou veřejnou část webové prezentace jsem uzpůsobil dle požadavků a návrhu pana prof. Tomáše Opatrného, kterému jsem vytvořil přístup do testovací verze webové aplikace. Změny které jsem na této stránce zveřejňoval jsem s panem prof. Tomášem Opatrným konzultoval emailem. Vzhled administrace webové prezentace jsem vytvořil podle sebe. Jednotlivé možnosti administrace jsem vytvářel dle budoucí potřeby, ve které by mohlo dojít ke změně textu, obrázků v galerii, nebo uživatelů, které by pan prof. Tomáš Opatrný mohl případně upravit bez zásahu programátora.

V první části se věnuji popisu převodu dat mezi Arduinem, bezdrátovým modulem označeným NodeMcu a webovým serverem. V Druhé části popisují použité technologie a ve třetí se zabývám postupy, které jsem použil při vytváření webové prezentace. V poslední části se zaměřuji na uživatelskou příručku.

2 Arduino

Arduino je vývojová elektronická deska založená na bázi jednočipového počítače s flash pamětí a otevřeným softwarem. Programům běžícím v Arduinu se říká "sketch" a základní jádro tvoří funkce `setup()`, která je volána pouze jednou při spuštění Arduina a určuje základní nastavení proměnných, knihoven, pinů, typů komunikace a všech počátečních hodnot. Poté se spustí funkce `loop()`, která je volána v nekonečné smyčce a zajišťuje programu možnost měnit se, reagovat a aktivně ovládat Arduino. Bez těchto dvou funkcí není možné spustit žádný program na Arduinu. Pro nahrávání zdrojového kódu do Arduina se používá vývojové prostředí s otevřeným zdrojovým kódem Arduino IDE, které je možné použít i pro ostatní výrobce jednočipových počítačů kompatibilních s Arduinem. Nahrávání zdrojového kódu do Arduina probíhá pomocí sériové komunikace u novějších modelů Arduina, které mají wifi modul, a je možné nahrávat zdrojový kód pomocí cloudu. Sériová komunikace se u Arduina využívá i ke komunikaci s dalším Arduinem, nebo pro výstupy a vstupy při komunikaci s počítačem. Každé Arduino má v době psaní této práce k dispozici nejméně jednu hardwarovou sériovou komunikaci, kterou je možné využít k připojení jiného Arduina, nebo počítače. [2] [3]

Pokud Arduino obsahuje pouze jeden hardwarový sériový port ke komunikaci a je vyžadována současná komunikace počítače i dalšího Arduina, je tato sériová komunikace využita přednostně ke komunikaci s počítačem a případná komunikace s druhým Arduinem je ignorována, dokud nedojde k odpojení počítače. Tento problém je možné řešit pomocí softwarové sériové komunikace, která je k dispozici v knihovně `SoftwareSerial.h`, a obsahuje vývojové prostředí Arduino IDE. Detailní popis této knihovny je k dispozici na www.arduino.cc. Díky softwarové sériové komunikaci je možné rozšířit hardwarovou sériovou komunikaci o jednu další a umožní komunikaci počítače a zároveň dalšího Arduina současně.

2.1 Komunikace

2.1.1 Arduino

Vzhledem k tomu, že v době psaní této práce je pro sběr dat používána deska Arduino Due neumožňující připojení k bezdrátové síti přírodovědecké fakulty, které je nezbytné pro jakoukoli vzdálenou komunikaci s webovým serverem, rozhodl jsem se pro připojení samostatné vývojové platformy NodeMcu s wifi modulem ESP8266EX. Jedná se samostatnou vývojovou desku, která je, stejně jako Arduino, založena na bázi jednočipového počítače s vlastní flash pamětí a je kompatibilní s Arduinem, i programovacím prostředím Arduino IDE. K použití nezávislého systému NodeMcu mě vedl fakt, že je kvůli nezávislému výpočetnímu výkonu na Arduinu možné zajistit spojení s bezdrátovou sítí přírodovědecké fakulty a následnou komunikací se serverem fakulty pomocí protokolu HTTP. Díky této nezávislé konfiguraci, kdy je možné zajistit předání dat mezi Arduinem a vývojovou platformou NodeMcu pomocí sériové komunikace, bylo možné zane-

chat zdrojový kód na Arduinu, který zajišťuje řízení Foucaultova kyvadla bez větších zásahů a další výpočetní zátěže pro Arduino. K tomuto řešení mě vedly i úpravy zdrojového kódu panem prof. Tomášem Opatrným, který potřebuje mít k dispozici možnost úpravy zdrojového kódu. Proto jsem ve zdrojovém kódu v Arduinu zajistil pouze ve funkci `setup` zahájení sériové komunikace mezi Arduinem a NodeMcu. V průběhu testování jsem zjistil, že sériová komunikace není navázána, pokud se Arduino i NodeMcu zapnou ve stejnou dobu, a proto jsem ještě přidal úpravu, která zahájí sériovou komunikaci a následně restartuje NodeMcu a spustí se do resetovacího pinu na NodeMcu z Arduina napájení 1.

Dále jsem ve funkci `loop` zajistil k usnadnění předání dat předávací textový řetězec označený `sendData`, do kterého je možné libovolně zadávat všechny hodnoty, které je potřeba předat a uložit na server fakulty. Tento textový řetězec je předán pomocí funkce `Serial3.print(sendData)`, za předpokladu, že je k dispozici propojení sériové komunikace s NodeMcu a současně probíhá čtení sériové komunikace z NodeMcu. Takto jsou zjišťovány základní informace o probíhající komunikaci z NodeMcu s bezdrátovou sítí fakulty a serverem katedry, který zpracovává přijatá data bez nutnosti připojení k počítači.

```
1 //zahájení seriové komunikace s NodeMcu
2 Serial3.begin(9600);
3 delay(300);
4 Serial.println("Resetuji NodeMcu");
5 //zapnutí napájení resetovacího pinu
6 pinMode(pinWifiReset, OUTPUT);
7 delay(100);
8 digitalWrite(pinWifiReset, HIGH);
9 Serial.println("hotovo");
10 delay(100);
11 //vypnutí napájení resetovacího pinu
12 pinMode(pinWifiReset, INPUT);
```

Zdrojový kód 1: Zahájení seriové komunikace s NodeMcu na Arduinu

2.1.2 NodeMcu

NodeMcu je samostatná vývojová elektronická deska s bezdrátovým modulem kompatibilní s Arduinem. NodeMcu má k dispozici pouze jeden hardwarový port pro komunikaci. Rozhodl jsem se tedy o jeho rozšíření za pomoci softwarové sériové komunikace, protože v průběhu testování se nahrávání upraveného zdrojového kódu do NodeMcu ukázalo jako problémové, pokud bylo k NodeMcu připojeno Arduino, což by bylo v reálném prostředí problém vzhledem k fyzické přístupnosti. K zajištění komunikace s bezdrátovou sítí jsem použil knihovnu `ESP8266WiFi.h` a knihovnu `WiFiClient.h`. Obě knihovny je možné nahrát z vývojového prostředí Arduino IDE. Ve zdrojovém kódu jsem připravil možnost

ke změně názvu bezdrátové sítě a jejího hesla, protože v případě změny názvu bezdrátové sítě, nebo jejího hesla, je komunikace přerušena. Naprogramoval jsem tedy kontrolu připojení k bezdrátové síti, která je předána pomocí sériové komunikace do Arduina pro zpětnou kontrolu připojení. K odesílání dat jsem zvolil HTTP klienta z knihovny `ESP8266HTTPClient.h`, kterou je možné nahrát z vývojového prostředí Arduino IDE. Protože knihovna `ESP8266HTTPClient.h` umožňuje spojení pomocí protokolu HTTP i HTTPS, rozhodl jsem se naprogramovat přepínač mezi protokolem HTTP a HTTPS. Jelikož se nejedná o důvěrná data, která NodeMcu posílá, zvážil jsem využití primárně protokolu HTTP, který nevyžaduje další nastavení. HTTPS protokol vyžaduje k připojení veřejný klíč, který je potřeba při každé změně certifikátu měnit.

2.1.3 Zpracování na straně webového serveru

Ke zpracování dat na straně serveru jsem naprogramoval jednoduchou webovou stránku v jazyce PHP, kde v první řadě proběhne ověření komunikačního klíče a označení NodeMcu zjištěním, že se v SQL databázi nachází záznam, který odpovídá zaslanému komunikačnímu klíči a označení NodeMcu. Pokud je tato podmínka splněna, dojde k uložení příchozích dat v nezměněné podobě a časové stopě odpovídající času přijetí dat do SQL databáze.

2.2 HTTP a HTTPS

HTTP je zkratka Hypertext Transfer Protocol. Jedná se o protokol poskytovaný na aplikační vrstvě, který používá k přenosu dat transportní vrstvu TCP zajišťující spolehlivé předání dat. HTTP protokol je používán pro přenos hypertextových dokumentů a jiných typů souborů. Protokol HTTP používá model klient server a funguje na způsob požadavek - odpověď. Protokol HTTP neumožňuje šifrování dat, a proto se jedná o protokol, který není bezpečný využívat k posílání citlivých dat. K tomuto účelu se používá protokol HTTPS, který svoji komunikaci šifruje nad transportní vrstvou TCP pomocí SSL nebo TLS certifikátu, což je zásadní rozdíl mezi protokolem HTTP a HTTPS. Dále také typicky HTTP protokol přenáší data pomocí portu 80 a protokol HTTPS pomocí portu 443. Certifikát je tedy možné vygenerovat pro webový server, například na letsencrypt.org, kde jsou dostupné aplikace na různé platformy a webové servery umožňující automatické prodlužování certifikátu vzhledem k tomu, že má platnost přibližně tři měsíce.

Z důvodů větší bezpečnosti šifrování dat a zároveň možnosti ověřit si pomocí certifikátu pravost stránek, (kdy je těžší podvrhnout v serveru DNS dotazovanou webovou stránku za falešnou webovou stránku), používám k webové prezentaci protokol HTTPS, který je konfigurovaný na straně webového serveru na automatické HTTP na HTTPS. Ke sběru dat z NodeMcu používám protokol HTTP, kde je automatické směrování na webovou stránku, která zajišťuje komunikaci v konfiguraci serveru, vypnuté. Důvod, proč jsem zvolil k přenosu dat nezabezpečenou komunikaci pomocí protokolu HTTP, je právě jeho šifrování, kdy by docházelo

kvůli automatické změně TLS certifikátu každé tři měsíce k výpadku posílaných dat, dokud by nedošlo ke změně veřejného klíče certifikátu na NodeMcu. To je však vzhledem k fyzické dostupnosti (je potřeba rozdělat podlahu u Foucaultova kyvadla) komplikované. Vzhledem k současnému stavu a také tomu, že jsou data veřejně dostupná a nejsou tedy důvěrná, je lepší využít protokol HTTP. Pro případ, že by bylo možné využít fakultou optiky TLS certifikát s delší dobou platnosti (alespoň rok, zejména bez automatické obnovy TLS certifikátu), jsem připravil v NodeMcu možnost využít přepnutí na HTTPS komunikaci. [1]

3 Použité technologie

Webové technologie, které jsem v této práci použil, jsem přizpůsobil podle možností katedry optiky přírodovědecké fakulty, na základě emailové komunikace se správcem serveru katedry optiky. K posílání dat jsem vybíral technologii kompatibilní se současnou technologií sběru dat, která je založená na Arduinu.

3.1 PHP

PHP je dynamicky typovaný skriptovací jazyk, který pracuje na straně serveru. PHP je nejrozšířenější skriptovací jazyk pro web. Vzhledem k tomu, že při vývoji PHP může dojít k odstranění některých funkcí a může tak dojít k nefunkčnosti webových stránek, jsem vývoj soustředil na verzi 7.4, kterou má v plánu používat fakulta optiky Univerzity Palackého. Pro případ přechodu k novější verzi jsem ve virtuálním prostředí přizpůsobil webové stránky na současnou nejnovější PHP verzi 8.0. Je tedy možné k bezproblémovému provozu webových stránek použít PHP verzi 7.4 nebo 8.0. [4] [5]

3.2 JavaScript

JavaScript je dynamicky typovaný objektově orientovaný skriptovací jazyk, který je spouštěný na straně klienta a nezatěžuje tak server. K interpretaci JavaScriptu dochází přímo v prohlížeči. JavaScript je událostmi řízený, umožňuje tedy reagovat na události a měnit obsah webových stránek bez opětovného načtení webové stránky. Na JavaScriptu je založena značná část této práce, bez něj by nebylo možné znázornit animaci pohybu Foucaultova kyvadla a dynamické načítání dat a změn webových stránek bez nutnosti opětovného načtení webové stránky. [6] [7]

3.2.1 Knihovna CK Editor

Knihovna CK Editor je knihovna třetí strany, jedná se o JavaScriptovou knihovnu určenou k úpravě webových stránek bez potřebné znalosti HTML nebo CSS. Je to knihovna s otevřeným zdrojovým kódem a v této práci je použita k úpravě

textu na titulní webové stránce. Bližší informace o této knihovně jsou dostupné na stránkách ckeditor.com.

3.3 HTML

HTML je značkový jazyk určený zejména k tvorbě webových stránek a jedná se o nejrozšířenější značkový jazyk, který určuje sémantický význam obsahu webové stránky. [8]

3.4 CSS

CSS je zkratka kaskádových stylů používajících se k úpravě vzhledu dokumentů. Kaskáda se v CSS používá při deklaraci stejných vlastností selektorů. Větší váha selektorů má přednost před nižší vahou selektorů. V případě, že mají selektory stejnou váhu, je použitý selektor, který je ve zdrojovém kódu poslední a důležitost, která obchází pravidlo kaskády pomocí vlastnosti `important!`. K přehlednému psaní zdrojového kódu v CSS se používají metodiky, například BEM. K psaní zdrojového kódu jsem se rozhodl použít vlastní metodiku, ale kdybych psal práci znovu od začátku, rozhodl bych se použít metodiku BEM kvůli lepší přehlednosti zdrojového kódu. [9]

3.5 MariaDB

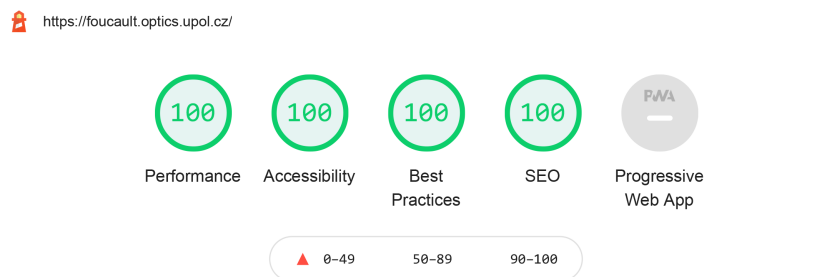
Jedná se o SQL databázi s otevřeným zdrojovým kódem, je to nástupce MySQL. MariaDB je v této práci použita zejména proto, že je používána fakultou optiky Univerzity Palackého, a v této práci slouží k uložení příchozích dat z Foucaultova kyvadla a dalších nezbytných částí, o kterých je třeba ukládat záznamy. [4] [10]

3.6 Arduino

Arduino je vývojová elektronická deska založená na bázi jednočipového počítače s flash pamětí a otevřeným softwarem. Arduino je jedna z použitých technologií pro sběr dat z Foucaultova kyvadla a jeho další řízení. Bližší informace o technologii Arduino jsem popsal na začátku kapitoly o Arduino 2.

4 Webová prezentace

Tato část popisuje postupy, které jsem použil při vytváření webové prezentace. Vzhled webové prezentace jsem navrhl a přizpůsobil pro mobilní zařízení, notebooky a monitory. Veřejnou část webové prezentace jsem přizpůsobil nejznámějším webovým prohlížečům Google Chrome, Mozilla Firefox, Opera, Safari a Microsoft Edge, aby bylo možné stránky zobrazit na co největším počtu zařízení.



Obrázek 1: Ukázka měření pomocí programu Lighthouse

Administraci jsem upravoval pro webové prohlížeče Google Chrome, Mozilla Firefox a Microsoft Edge. Rychlost načítání jsem kontroloval pomocí aplikace Lighthouse, kterou má k dispozici vývojářské prostředí Google Chrome 1. Strukturu HTML kódu jsem kontroloval pomocí nástroje validator.w3.org a strukturu CSS kódu jsem kontroloval pomocí nástroje w3.org/css-validator.

4.1 veřejná část webové prezentace

Pro webovou prezentaci jsem se rozhodl použít jednoduchou webovou strukturu a rozdělit ji na tři hlavní sekce:

- *Popis Foucaultova kyvadla* je zobrazen na začátku webové prezentace a obsahuje základní popis Foucaultova kyvadla.
- *Přijátá data* jedná se o nejzajímavější část webové prezentace a je zobrazena uprostřed. V této části je vykreslena animace kyvadla přenášena v reálném čase v průběhu dne, nebo je možné vybrat a zobrazit jiný den. V této části jsou i grafy ukazující hodnoty a časový vývoj směru kmitů a Stokesových parametrů.
- *Fotogalerie* na webové prezentaci se nachází jako poslední a zobrazuje galerii.

V navigaci, v horní části webové prezentace, jsem přidal odkazy na jednotlivé sekce. Struktura je společná pro českou i anglickou verzi webu a její vzhled jsem se snažil přizpůsobit stránkám Univerzity Palackého. Části "Přijátá data" a "Fotogalerie" jsem rozdělil na menší části, aby bylo možné přidávat, nebo upravovat dodatečné informace bez zásahu do zdrojového kódu. V administraci jsem naprogramoval úpravu textů jednotlivých částí. Kvůli většímu pohodlí uživatele jsem se rozhodl nenačítat naměřená data přímo v titulní stránce webové prezentace, místo toho jsem vytvořil webovou stránku, kterou z titulní strany volám pomocí JavaScriptové třídy `XMLHttpRequest()`. Tato JavaScriptová třída umožňuje výměnu dat webového klienta se serverem bez nutnosti opětovného načtení webové stránky. Díky této JavaScriptové třídě je tak možné měnit načtené hodnoty podle datumu přímo ve stránce bez nutnosti opětovného načtení. JavaScript využívám

i k vykreslení v HTML elementu `<canvas>`, který používám ke grafickému vykreslení Foucaultova kyvadla a jednotlivých grafů. V případě, že má uživatel webového klienta, který neobsahuje, nebo má zakázaný JavaScript, jsem použil HTML element `<noscript>`, který uživatele informuje, že bez něj není možné správně webovou prezentaci zobrazit.

4.1.1 Vypsání naměřených dat

K vypsání řetězce obsahující data, která jsou uložena v databázi a byla naměřena Arduinem, jsem u Foucaultova kyvadla naprogramoval webovou stránku `"/data-load"`. Tato webová stránka vypisuje řetězec hodnot obsahující Stokesovy parametry, stupně, stav jednotlivých senzorů a datum načtených dat. Datum naměřených hodnot jsem naprogramoval podle aktuálního dne a přidal možnost jej změnit pomocí HTTP POST metody. Protože používám k uložení dat textový řetězec, je pořadí prvních hodnot nutných pro správné zobrazení animace Foucaultova kyvadla a grafů dané, a založené na domluvě s panem prof. Tomášem Opatrným. Textový řetězec jsem k uložení dat použil, aby bylo možné měnit jeho strukturu bez zásahu do zdrojového kódu webové části a přizpůsobovat posílaná data podle potřeby pana prof. Tomáše Opatrného. Textový řetězec, který vytváří tato webová stránka, načítám do titulní strany webové prezentace.

4.1.2 Načtení naměřených dat

V JavaScriptu jsem naprogramoval funkci `loadData()` k načítání textového řetězce, který vytváří webová stránka `"/data-load"`. Tato funkce je volána při načtení webové prezentace pomocí události `onload="loadData()"`, nebo v případě, že uživatel změní datum pomocí události `onchange="loadData()"`. Při volání této funkce dochází k zastavení časovače animace Foucaultova kyvadla a vytvoření instance třídy `XMLHttpRequest()` pomocí které je vytvořen HTTP POST metodou požadavek na webovou stránku `"/data-load"` se zasláným parametrem `den`. Zároveň dochází k vytvoření události, kterou zpracovávám obsluhou události `xhttp.onreadystatechange`. Při zpracování obsluhou události je volána lokální funkce, kterou jsem zajistil rozdělením načteného datového řetězce do globálních proměnných. Dále dochází k načtení maximální možné hodnoty, kterou je možné nastavit posuvníkem, vymazáním neaktuálních dat z grafů a voláním funkcí k nastavení nových hodnot grafů a animace Foucaultova kyvadla. V případě, že by došlo k chybě během nastavování, nebo načítání dat, jsem vytvořil vyjímku, kterou jsem zajistil vykreslení prázdných grafů, smazání animace pohybu Foucaultova kyvadla, namísto které se zobrazí pouze informace o nedostupnosti dat a do konzole webového prohlížeče dojde k vypsání hlášení, že došlo k chybě při načítání. Původně jsem zvažoval funkci načítat data v určitém intervalu, aby se zachovávala vykreslení aktuálních dat na otevřené stránce, ale tuto možnost jsem nakonec nevytvořil, protože by docházelo ke zbytečnému zpomalování serveru u uživatelů, kteří by nechali otevřenou webovou aplikaci. K načtení aktuálních dat je tedy nutné webovou prezentaci znovu načíst.

4.1.3 Animace kyvadla

K animaci pohybu Foucaultova kyvadla jsem původně vytvořil funkci. Funkce ale nesplňovala moje požadavky na změny, které jsem postupně přidával a které ji začaly dělat nepřehlednou. Proto jsem se rozhodl funkci přepsat na třídu `Pendulum`. Další důvod byl princip zapouzdření, který mi umožnil skrýt přístup k atributům a metodám, aby nedošlo k nedovoleným změnám. Do konstruktoru třídy jsem přidal proměnnou s identifikátorem HTML elementu `<canvas>` do kterého vykresluji trajektorii Foucaultova kyvadla. Díky tomu je možné identifikátor zadat při vytváření instance třídy a je tím zachována snadná přenositelnost zdrojového kódu. Během vytváření instance třídy jsem nastavil výpočet velikosti `<canvas>`, kterým jsem zajistil přizpůsobení velikosti k případným změnám `<canvas>` bez nutnosti úpravy atributů třídy. Ke změnám atributů jsem vytvořil jednotlivé metody:

- `setName(text)` tato metoda slouží ke změně nadpisu pro jinou než českou verzi webové prezentace. Jako parametr přijímá textový řetězec.
- `setUnavailableDataText(text)` tato metoda slouží ke změně textu v případě, že nejsou dostupná data pro jinou než českou verzi webové prezentace. Jako parametr přijímá textový řetězec.
- `setTime(sec = 0)` tato metoda umožňuje změnu zobrazeného času. Jako parametr přijímá číslo, které určuje zobrazený čas v sekundách.
- `availableData(bool = false)` tato metoda určuje, zda jsou dostupná data. Přijímá parametr pravdivostní hodnoty. Výchozím nastavením je nedostupnost dat.
- `setStokesParameters(value, stokesArray)` v této metodě dochází k výpočtu X a Y souřadnic ze Stokesových parametrů, které určují přesný pohyb Foucaultova kyvadla. Tato metoda vyžaduje číslo určující pozici v poli Stokesových parametrů a dvourozměrné pole Stokesových parametrů, kdy první rozměr určuje sloupec a druhý rozměr textový řetězec Stokesových parametrů, které rozdělují čárka.

Ke grafickému vykreslení nadpisu, schodiště, úhloměru a senzorů jsem vytvořil metodu `clearDraw()`, která nejdříve zavolá metodu smazání celého kreslicího pole, `clear()` a následně metodu `draw()` k jeho opětovnému vykreslení. Metoda `clearDraw()` je volána z metody `drawMove(i)` vykreslující pozici, ve které se nachází Foucaultovo kyvadlo. Tato metoda přijímá číslo, které určuje zobrazovanou pozici Foucaultova kyvadla. Abych mohl vytvořit animaci pohybu Foucaultova kyvadla, vytvořil jsem novou třídu `pendulumMoveAnimation(object)`, která při vytváření instance jako parametr přijímá instanci třídy `Pendulum`. Tato funkce vytváří interval, ve kterém dochází ke smazání kreslicího pole, posunutí Foucaultova kyvadla na další pozici a jeho vykreslení na této pozici. Interval jsem vytvořil pomocí metody `setInterval(this.move, this.interval)` třídy `Window`, která

volá zadanou funkci v pravidelném časovém intervalu, dokud nedojde k zavření okna nebo zavolání metody `clearInterval()`. K co nejpřesnějšímu pohybu animace pohybu Foucaulova kyvadla podle skutečného Foucaultova kyvadla, kdy celý pohyb trvá přibližně deset sekund, jsem použil výpočet `this.interval = 10000 / 628;` ve kterém číslo 628 určuje počet pozic, které jsou spočítány ze Stokesových parametrů. Spuštění animace pohybu Foucaultova kyvadla je možné pomocí atributu `start` a zastavení animace pohybu Foucaultova kyvadla pomocí atributu `stop`.

Uživatel webové prezentace může posouvat animaci pohybu Foucaultova kyvadla v průběhu dne pomocí posuvníku a tlačítek plus a mínus, které jsou umístěny pod kreslicí plochou s animací pohybu Foucaultova kyvadla. K tlačítkům jsem vytvořil funkce `plusSlider()` a `minusSlider()`, které zastaví současnou animaci, posunou posuvník dopředu nebo dozadu a zavolají funkci `changeTimeAnimation()`, která je volána i při změně hodnoty přímo na posuvníku. Tato funkce tak zajišťuje uživateli webové prezentace možnost posouvat trajektorii Foucaultova kyvadla v průběhu dne. Uživatel také může spustit animaci trajektorie, která znázorňuje chování Foucaultova kyvadla v průběhu dne. K tomu jsem vytvořil tlačítka spustit a zastavit, které volají při jejich stisknutí funkce `animateDaily()` a `AnimationStop()`. Funkce `animateDaily()` napřed zastaví animaci pohybu Foucaultova kyvadla a následně vykreslí trajektorii s kyvadlem na první pozici. K tomu jsem vytvořil interval, který volá každých patnáct milisekund funkci, `trajectoryDaily()` ve které kontroluji, zda nebylo dosaženo poslední naměřené hodnoty, a v případě, že bylo dosaženo, spustí animaci pohybu Foucaultova kyvadla na poslední pozici. Pokud nebylo dosaženo poslední naměřené hodnoty, kontroluji, jestli jsou v pořádku všechny senzory a pokud ano, je vykreslena trajektorie a Foucaultovo kyvadlo na první pozici, ve které se nachází. V případě, že v daný moment měl některý ze senzorů poruchu, je místo trajektorie vykreslen otazník a informace o nedostupnosti dat. Spuštění tlačítkem jsem vytvořil od bodu ve kterém je nastaven posuvník a v případě, že je posuvník na poslední hodnotě, posuvník umístuji na začátek.

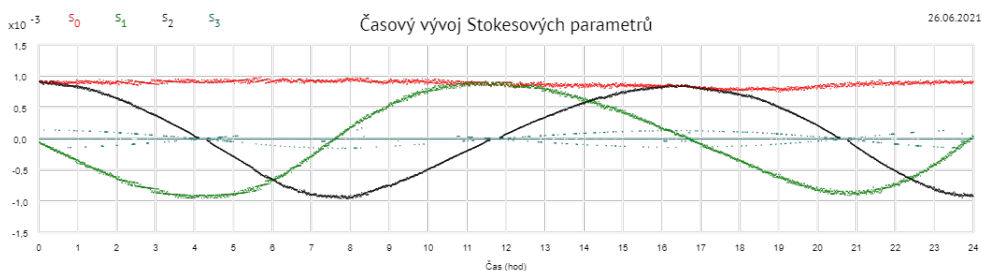
4.1.4 Grafy

Aby bylo možné vytvořit větší množství grafů bez nutnosti opakování zdrojového kódu, vytvořil jsem vlastní třídu `TimeGraph`. Ke grafickému zobrazení jsem se rozhodl pro kreslicí plátno dostupné pomocí HTML elementu `<canvas>`, který oproti vykreslení pomocí SVG. SVG je značkovací jazyk pro popis dvojrozměrné grafiky. Kreslicí plátno jsem použil, protože nabízí rychlejší práci s větším množstvím dat a také dochází k vykreslování v kreslicím plátně a je vytvořen pouze jeden objekt. Oproti tomu SVG tvoří na každou hodnotu speciální objekt. Kreslicí plátno jsem tedy využil, protože grafy zobrazují časový vývoj v průběhu dne a teoreticky by mohlo dojít k vytvoření až 86 400 hodnotám u grafu, který zobrazuje časový vývoj směrů kmitů, a 345 600 hodnotám v případě grafu, který zobrazuje časový vývoj Stokesových parametrů. K takovému množství dat by došlo, pokud by přicházela data z Foucaultova kyvadla přicházela v intervalu každou

sekundu. V době psaní této práce přichází data z Foucaultova kyvadla přibližně každých 30 sekund, což je průměrně 3000 zobrazených hodnot u časového vývoje směrů kmitů a 12000 zobrazených hodnot u časového vývoje Stokesových parametrů. Třidu `TimeGraph` jsem tedy vytvořil, aby bylo možné upravovat interval zasílaných dat přímo panem prof. Tomášem Opatrným bez potřebného zásahu do zdrojového kódu.

Možnost nastavení kreslicího plátna jsem přidal do konstruktoru třídy `TimeGraph`. Vytvořením instance třídy `TimeGraph` je tedy nutné zadat identifikátor elementu s kreslicím plátnem. Dále jsem vytvořil výpočet velikosti kreslicí plochy ke snadnější úpravě velikosti plochy bez nutnosti změnit zdrojový kód třídy, `TimeGraph` ke kterému dochází při vytvoření instance třídy. K úpravě jednotlivých instancí třídy `TimeGraph` jsem vytvořil metody:

- `setMaxValue(x, y, minusGraph)` tato metoda slouží k nastavení maximální hodnoty X a Y souřadnic, a také k nastavení, zda bude graf zobrazovat mínusové hodnoty. Přijímá tedy jako parametry dvě čísla, pomocí kterých provádím výpočet správného umístění hodnot vzhledem k velikosti grafu, z nichž první určuje X osu druhý Y osu. Pokud by tyto hodnoty byly špatně stanoveny, mohlo by docházet k vykreslování mimo kreslicí plochu. Třetí parametr je pravdivostní hodnota, pomocí které dochází k upravení kreslicího prostoru a určuje, zda bude graf zobrazovat i mínusové hodnoty.
- `setName(text)`, `setXName(text)`, `setYName(text)` tyto metody umožňují měnit název grafu, název osy X a název osy Y. Přijímají jako parametr textový řetězec.
- `setDescription(x, y, rightY = "")` pomocí této metody je možné nastavit popis hodnot u osy X a osy Y. Pro první dva parametry, které tato metoda přijímá, je vyžadováno pole. Graf je navržen na dvacetpět políček v X souřadnici a sedm v Y souřadnici. Vyžaduje tedy ke správnému popisu hodnot v prvním poli dvacetpět hodnot a ve druhém sedm hodnot. Osu Y je možné doplnit o horní popis, takzvaný upperstand v posledním parametru, který přijímá textový řetězec.
- `draw()` pomocí této metody dochází k vykreslení názvu, struktury a popisků. V této metodě nedochází k vykreslení hodnot. Metoda nevyžaduje žádný parametr.
- `drawPoint(x, y, color)` pomocí této metody je možné kreslit jednotlivé body, které zobrazují hodnoty v grafu. Metoda vyžaduje jako první parametr hodnotu osy X. Druhý parametr určuje hodnotu osy Y a třetí vyžaduje textový řetězec, který odpovídá některé barvě ve formátu názvu barvy v anglickém jazyce, v RGB formátu, nebo HEX formátu.
- `clearGraph()`, `clearPoint()` tyto metody slouží k mazání kreslicího pole. Druhá metoda, kromě mazání, volá metodu, `draw()` pomocí které je nakreslena základní struktura grafu.



Obrázek 2: Ukázka grafu Stokesových parametrů

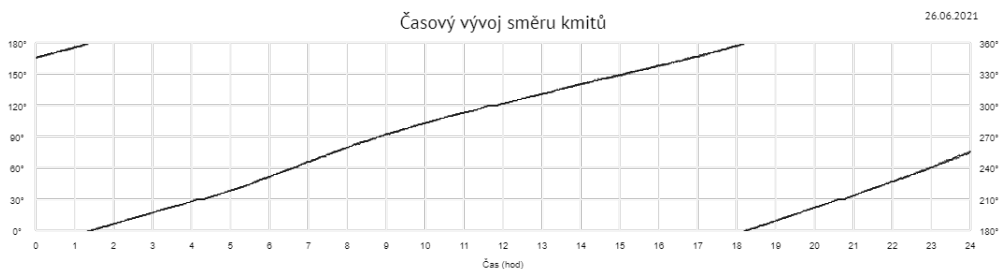
- `drawDescription(text, x, y, color="000000", fontSize=12, upperLine="", underline="")` pomocí této metody je možné graf doplnit o další popis. Hlavní popis je možné doplnit o text zobrazený nad textem, takzvaný `upperline`, nebo o text zobrazený pod textem, takzvaný `underline`. Vyžaduje textový řetězec, souřadnici X a Y, které určují, kde bude daný text vykreslen. Textový řetězec, který odpovídá některé barvě ve formátu názvu barvy v anglickém jazyce, v RGB formátu, nebo HEX formátu, velikost písma, které je ve výchozím nastavení 12 pixelů, předposlední parametr je textový řetězec, který je zobrazený nad hlavním textem a poslední parametr je textový řetězec zobrazený pod hlavním textem.

K počátečnímu nastavení potřebných hodnot do vytvořených instancí třídy `TimeGraph` jsem vytvořil funkci `setGraphData()`. V této funkci dochází i k vykreslení názvu, struktury grafu a jejich popisků. Hodnoty v této funkci jsem určil podle jazykové verze webových stránek. Překlad do anglické verze u jednotlivých grafů a správnému zobrazení hodnot v anglickém jazyce tak dochází v této funkci.

K vykreslení hodnot jsem vytvořil funkci `drawGraphData(sensorsArray)`. V této funkci dochází k postupnému procházení načtených hodnot a jejich vykreslení do jednotlivých grafů. V případě, že některý senzor nefunguje, nejsou správně počítány všechny zobrazené hodnoty a došlo by tak k jejich zkrácení. Proto v této funkci ověřuji, jestli byly funkční všechny senzory v době poslání dat. V případě, že byl nefunkční některý senzor, jsou data přeskočena v grafu a nejsou vykreslena. Do této funkce jsem přidal popis, který určuje u grafu Stokesových parametrů, barvu jednotlivých Stokesových parametrů, ve které byla data vykreslena a také popis s datem, ve kterém byla data načtena. Jednotlivé grafy je možné vidět na obrázku 2 a 3.

4.1.5 Galerie

Galerie jsem se rozhodl načítat přímo na titulní stránce, protože nebyla potřeba upravovat zobrazené obrázky v průběhu prohlížení webové prezentace uživatelem. Při načítání procházím informace o jednotlivých obrázcích uložených v databázi. Mezi načítané informace patří název obrázku, místo jeho uložení, autor obrázku, popis v českém jazyce pro českou verzi webové prezentace a v



Obrázek 3: Ukázka grafu směru kmitů

anglickém jazyce pro anglickou verzi webové prezentace, a také zda má být obrázek zobrazen. K rychlejšímu načítání titulní stránky jsem u obrázků nastavil takzvané líné nahrávání ``. To zajistí načítání obrázku až v případě, že se uživatel nachází ve webové prezentaci u daného obrázku. Ke zvětšení obrázků jsem vytvořil jednoduchou funkci, která zobrazí připravený zvětšený obrázek a načte do něj informace o zobrazovaném obrázků.

4.1.6 Zapomenuté heslo

V případě, že uživatel zapomene heslo, jsem vytvořil webovou stránku, na kterou jsem přidal odkaz z formuláře přihlášení. Jedná se o jednoduchý webový formulář, který vyžaduje uživatelské jméno a email uživatele. Při ověření, zda se uživatel nachází v databázi, dochází k vytvoření požadavku o změnu hesla a zápis do databáze. Současně je uživateli pomocí funkce `mail()` odeslán email s vytvořeným odkazem pro obnovu hesla. Odesílání emailu jsem omezil na jednu žádost za den v případě, že se žádost o obnovu nachází v nevyřízeném stavu. Když uživatel otevře odkaz, který byl zaslán v emailu se správným klíčem, zobrazí se uživateli webový formulář k zadání nového hesla a jeho potvrzení. Po změně hesla dojde k úpravě informací o uživateli v databázi a uložení informace o změně.

4.2 Administrace webové prezentace

Administraci webové prezentace jsem rozdělil pro větší monitory na dvě navigační menu a sekci s daty a pro mobilní zařízení na rozevírací navigační menu a sekci s daty. Na horní navigační menu jsem vytvořil odkaz na titulní stranu prezentace pro snadné přepínání. Dále je zde možnost změnit heslo a odhlásit se z administrace webové prezentace. V levém navigačním menu jsou jednotlivé funkce administrace. Hlavní sekci tvoří kontejner, který mění svůj obsah podle volby uživatele z levého navigačního menu. Ke změně obsahu používám dříve zmíněnou třídu `XMLHttpRequest()`, která umožňuje komunikaci webového klienta a serveru. Je tedy možné s pomocí HTML DOM načtení nového obsahu na webové stránce bez nutnosti jejího opětovného načtení. HTML DOM je mo-

del objektu dokumentu a umožňuje pomocí JavaScriptu výměnu všech HTML elementů na webové stránce. Proto jsem ke každé potřebné činnosti vytvořil samostatnou webovou stránku napsanou v PHP. Všechny webové stránky jsem rozdělil do složek, které jsou pojmenovány podle zobrazovaného obsahu.

4.2.1 Příchozí data

Protože se jedná o uzavřenou část webové prezentace a nehrozí tak přetížení serveru, přidal jsem načítání dat v intervalu pomocí metody `setInterval(showPendulumData, 10000)` ve které dochází každých deset sekund k volání funkce `showPendulumData()`. Tento interval je možné zrušit výběrem konkrétního časového úseku. Maximální počet zobrazených hodnot při prvním načtení jsem nastavil na 360. Při testování jsem zjistil, že při větším množství zobrazovaných dat mají pomalejší zařízení problém je všechna vykreslit a dochází tak ke zpomalení webové stránky. Maximální limit zobrazených hodnot jsem nastavil na deset tisíc a minimální na jednu hodnotu.

4.2.2 Uživatelé

Uživatele ukládám do tabulky v databázi. Abych zachoval bezpečnost hesel a nebylo možné je přečíst z tabulky, v databázi jsem se rozhodl využít Hash generátor Argon2i, ke kterému má PHP podporu ve funkci `password_hash($promena, PASSWORD_ARGON2I)` při nastavení druhého parametru `PASSWORD_ARGON2I`. Tato funkce vytváří z původního hesla dlouhý textový řetězec. K ověření, zda uživatel napsal správné heslo, využívám PHP funkci `password_verify($password, $hash)`, která převede zadané heslo na Hash, ověří s uloženým heslem v tabulce v databázi a vrátí pravdivostní hodnotu. Detailní popis Hash generátoru Argon je na stránkách [argon2.online](#).

Protože se jedná o uzavřený systém, který vyžaduje pozvánku, rozhodl jsem se vytvořit formulář, který je možné zobrazit při kliknutí myši na tlačítko "přidat nového uživatele". Po vyplnění požadovaných údajů a kliknutí na tlačítko "registrovat" je uživateli na zadaný email poslán email, který ho informuje o vytvoření účtu a heslu, kterým se může do administrace přihlásit. Heslo, které uživateli přijde, zadává administrátor, který uživateli posílá pozvánku, aby bylo pro případ potřeby možné zadat jednoduché heslo. Aby administrátor nemusel při každém vytváření uživatele vymýšlet heslo, přidal jsem tlačítko "klíč", který pomocí třídy Math vygeneruje z připraveného řetězce nové heslo.

Ke smazání uživatelů jsem vytvořil tlačítko v editaci, které je možné vyvolat při stisknutí tlačítka "editace". Editaci jsem nastavil tak, aby ověřovala přihlášeného uživatele, a aby se přihlášený uživatel nemohl upravit, nebo smazat. Zároveň jsem tuto možnost vypnul u účtu pana prof. Tomáše Opatrného, kterého není možné v editaci upravit, nebo smazat vůbec.

4.2.3 Editace textu

Titulní stranu jsem rozdělil na jednoduché části, které je možné editovat bez znalosti HTML a CSS jazyka. Rozdělení jsem provedl, abych zabránil nechtěnému upravení grafických prvků stránky, které není možné bez znalosti programovacích jazyků provést. Každá část je uložena v samostatném HTML souboru, který se načítá na titulní straně v příslušném místě. K editování jednotlivých souborů jsem vytvořil tlačítka, která po stisknutí načtou HTML soubor do připraveného kontejneru. Aby bylo možné načtený text editovat bez znalosti HTML, používám JavaScriptovou knihovnu CK Editor, která převádí HTML elementy na čitelný text. Při stisknutí tlačítka "uložit" knihovna CK Editor zobrazený text převádí zpět na HTML elementy. K uložení upraveného textu jsem vytvořil stránku, ve které jsem nastavil filtrování některých HTML elementů, aby nedošlo k narušení struktury titulní stránky vložením škodlivého zdrojového kódu. Uložení probíhá opětovným načtením upravované stránky a přepsáním stávajícího textu novým pomocí PHP funkce `fwrite()`.

4.2.4 Galerie

U přidávání obrázků jsem se rozhodl rozdělit nahrávání obrázků do připravené složky a informace o obrázku ukládat do tabulky v databázi. Aby nebylo možné nahrát jiný soubor, než obrázek s koncovkou JPG, JPEG a GIF, nastavil jsem při nahrávání kontrolu souboru. Nastavil jsem i kontrolu názvu obrázku, aby nedošlo k nechtěnému uložení a smazání původního obrázku. K editaci každého obrázku jsem vytvořil formulář, ke kterému jsem vytvořil tlačítko editace. Při kliknutí na toto tlačítko je volána JavaScriptová funkce, která pomocí HTML DOM upraví obsah formuláře na vybraný obrázek. Editaci jsem nastavil k úpravě informací uložených v tabulce v databázi bez možnosti vyměnit obrázek.

Při mazání některého obrázku jsem nastavil smazání informace o daném obrázku z tabulky v databázi, u které dochází ke kontrole, zda smazání proběhlo v pořádku. Pokud ano, dochází ke smazání obrázku ze složky.

4.3 Kontrola fotodetektorů

Ke kontrole, jestli všechny fotodetektory pracují správně, jsem vytvořil webovou stránku, která monitoruje příchozí data z Arduina. Je to z toho důvodu, že může dojít k fyzické manipulaci s fotodetektory, například jejich otočením, a tak dojde k tomu, že Arduino nebude schopno provádět výpočet pohybu Foucaultova kyvadla. Tato webová stránka kontroluje, jestli je v příchozích datech zaznamenán nefunkční fotodetektor a následně na to upozorní emailovou zprávou pana prof. Tomáše Opatrného, který pak může fotodetektor seřídít, viz. ukázka posílaného emailu 4. Současně jsem naprogramoval do této webové prezentace kontrolu času příchozích dat. Data z Foucaultova kyvadla přichází v psaní této práce přibližně každých třicet sekund, pokud tedy data nejsou posílána dvacet minut, pošle pomocí PHP funkce `mail()` upozornění emailovou zprávou pro kon-

Došlo k výpadku senzoru

Senzor 1 funguje

Senzor 2 nefunguje

Senzor 3 funguje

Senzor 4 funguje

Výpadek začal v: 2021-07-06 19:20:01

Obrázek 4: Ukázka zasílaného emailu

trolu Foucaultova kyvadla. Tato webová stránka je určena pro plánovač úloh na serveru, protože bez interakce uživatele nebo serverem není možné, aby proběhlo její vyhodnocení a následné odeslání emailem.

5 Uživatelská příručka

Tato kapitola slouží jako dokumentace pro uživatele. Popisuje základní funkce, které jsou k dispozici ve veřejné části webové prezentace i v její uzavřené části označovanou jako administrace webové prezentace.

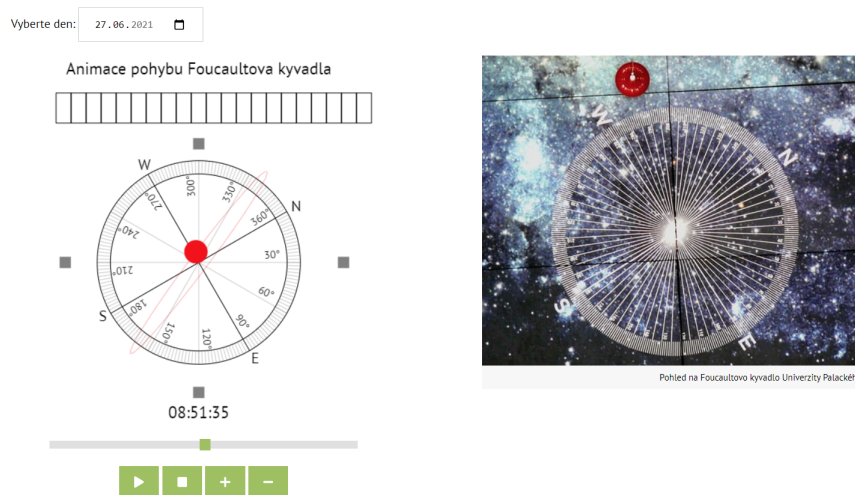
5.1 Veřejná část webové prezentace

5.1.1 Navigační menu

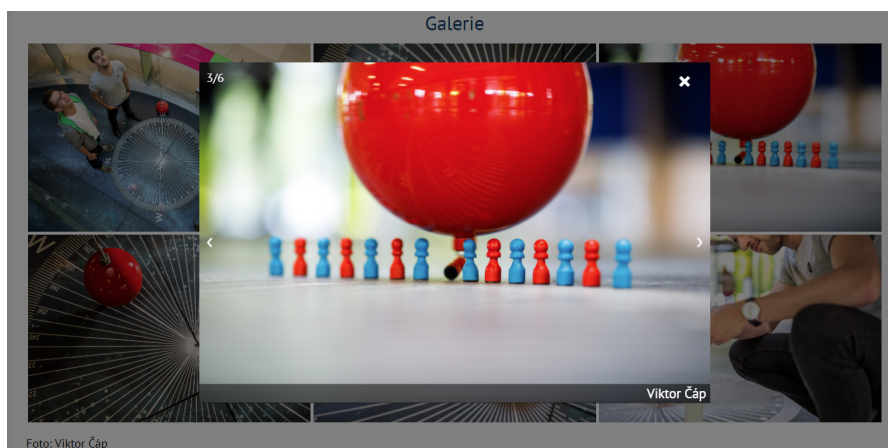
Základem veřejné části webové prezentace je navigační menu, které je rozděleno na dvě sekce. První slouží zejména pro přihlášení do Administrace, následnému přepínání mezi administrací a titulní stranou webové prezentace, odhlášení z administrace a přepínání mezi anglickou a českou webovou prezentací. Druhá část navigace umožňuje snadný přechod do části webové prezentace, která uživatele zajímá. Webová prezentace je rozdělena na tři části: Popis kyvadla, Přijatá data a Fotogalerie.

5.1.2 Přijatá data

Při zobrazení stránky se nastaví jako výchozí aktuální den a dojde k načtení hodnot do animace pohybu Foucaultova kyvadla a grafu. Aktuální den je možné změnit pomocí výběru dne, který se nachází v této části. Při změně dne dojde k novému načtení hodnot naměřených ve vybraném dnu a jejich zobrazení v animaci pohybu Foucaultova kyvadla a grafu. Výběr dne je společný pro animaci pohybu Foucaultova kyvadla i pro grafy. U animace pohybu Foucaultova kyvadla je možné pomocí posuvníku, nebo tlačítek plus a mínus, vybrat konkrétní trajektorii v průběhu dne. Pokud nebyla dostupná data, je místo animace pohybu



Obrázek 5: Animace pohybu Foucaultova kyvadla



Obrázek 6: Ukázka galerie

Foucaultova kyvadla zobrazen červený otazník a doplňující text. Dále je možné sputit tlačítkem "přehrávání" trajektorii pohybu zaznamenanou v průběhu dne, případně ji zastavit tlačítkem "zastavit". Ukázka animace pohybu Foucaultova kyvadla na obrázku 5.

5.1.3 Možnosti galerie

V části galerie je možné kliknout do obrázku a zvětšit ho, a následně kliknutím myši na šipky vlevo a vpravo měnit obrázky. Zavřít zvětšený náhled lze kliknutím myši na křížek v pravém horním rohu, nebo kliknutím myši mimo otevřený obrázek. Ukázka galerie na obrázku 6.

5.2 Administrace webové prezentace

V této části jsou popsány možnosti, které mohou vyžadovat příslušná oprávnění. Pokud Vám není v administraci webové prezentace některá z níže popsaných možností dostupná, nemáte k jejímu použití příslušné oprávnění.

5.2.1 Přihlášení

Přihlášení do administrace webové prezentace je možné pomocí formuláře, který je dostupný v horním menu. Při kliknutí na "přihlásit" po vyplnění správného uživatelského jména a hesla dojde k automatickému přesměrování do administrátorské části webové prezentace. Po úspěšném přihlášení lze libovolně přepínat mezi administrátorskou a veřejnou částí v horním menu.

5.2.2 Zapomenuté heslo

V případě zapomenutého hesla je možné vytvořit požadavek na změnu hesla, který je dostupný ve formuláři přihlášení. Po kliknutí na odkaz "zapomenuté heslo" dojde k přesměrování na formulář pro obnovu hesla. Po zadání uživatelského jména a emailu dojde k odeslání emailu na zadaný email, ve kterém jsou popsány další pokyny, jak postupovat. Žádost o vytvoření nového hesla je možné vytvořit jednou za den. Po úspěšné změně hesla lze vytvořit další žádost o změnu hesla.

5.2.3 Změna hesla

Po přihlášení do administrace webové prezentace je možné provést změnu hesla pro přihlašování, tato možnost je dostupná v horním menu po najetí myši na uživatelské jméno a následném kliknutí na "změna hesla".

5.2.4 Kontrola kyvadla

Kontrola kyvadla je dostupná jako první možnost v levém menu v administraci webové prezentace a jedná se o souhrn informací o Arduinu a senzorech za poslední den.

5.2.5 Příchozí data

Příchozí data jsou zobrazena jako první po přihlášení do administrace webové prezentace a jsou zde naměřené hodnoty v nezměněné podobě. Je zde možné vybrat počet zobrazených hodnot při změně čísla vlevo nahoře. Dále je možné vybrat den a čas zobrazených hodnot pomocí tlačítka "filtrovat podle času". Po kliknutí je zobrazen formulář, ve kterém je možné vybrat den a čas první hodnoty a poslední. K zobrazení vybraných hodnot dojde po kliknutí myši na tlačítko "zobrazit". V případě výběru nějakého konkrétního dne a času je v tomto formuláři dostupné tlačítko "obnovit" na zrušení výběru a zobrazení aktuálních

hodnot. Obnovit výběr na poslední naměřené hodnoty lze také při opětovném načtení webové stránky například pomocí klávesy F5.

5.2.6 Editace uživatelů

V editaci uživatelů, která je v levém menu označena jako "uživatelé", je možné přidat nového uživatele při kliknutí na tlačítko "přidat nového uživatele". Po kliknutí myši se zobrazí formulář, ve kterém je potřeba zadat údaje, práva a následně kliknout na tlačítko "registrovat", které pošle vytvořenému uživateli na zadaný email heslo a informaci o vytvoření účtu. Heslo pro nového uživatele je možné vytvořit pomocí tlačítka "klíč" dostupného v poli "heslo".

Dále je možné v editaci uživatelů editovat práva uživatele, nebo uživatele smazat pomocí tlačítka "editace" vedle vypsání uživatele.

5.2.7 Editace titulní strany

V editaci titulní strany, která je v levém menu označena jako "editace", lze měnit informace na titulní straně. Hlavní stránka je rozdělena na několik úseků a každý lze editovat pomocí daného tlačítka v české, nebo anglické verzi podle výběru. Po kliknutí myši na tlačítko se zobrazí textový editor, který vykreslí aktuální článek na titulní straně. Pokud je editor bez textu, je tento úsek na titulní straně prázdný. K uložení změn v daném úseku titulní strany slouží tlačítko "Uložit" dostupné pod oknem s textovým editorem. Při úspěšném uložení se zobrazí červený text "uloženo" a článek na titulní straně se ihned změní.

5.2.8 Editace galerie

Editace galerie je možná pomocí tlačítka v levém menu označeného jako "galerie". Po kliknutí myši na tlačítko "přidat nový obrázek" dojde k zobrazení formuláře, ve kterém je možné nahrát nový obrázek. Pro nahrání je nastaveno omezení na formáty JPG, JPEG a GIF s maximální velikostí 500KB. V případě úspěšného nahrání obrázku se zaškrtnutou možností "zobrazit v galerii", je tento obrázek zobrazen v galerii na titulní straně.

Po kliknutí myši na tlačítko "editace" se zobrazí formulář s daným obrázkem, kde je možné změnit zobrazené údaje o vybraném obrázku, jestli se má daný obrázek zobrazit na titulní straně v galerii, nebo lze tento obrázek smazat.

Závěr

Hlavní náplní této práce bylo zajistit a rozšířit možnosti Foucaultova kyvadla, které se nachází v budově přírodovědecké fakulty Univerzity Palackého. Zejména šlo o ukládání naměřených dat a jejich následné webové prezentace, kterou jsem rozdělil na dvě části - veřejnou a uzavřenou. Mým hlavním cílem bylo zajistit, aby tato práce byla co nejvíce flexibilní a zajišťovala co nejméně úprav ze strany programátora při pozdějších úpravách zejména proto, že bude spravována panem prof. Tomášem Opatrným. Je tedy možné nezávisle na přidání dalších funkcí kdykoli měnit naměřená data, která se budou posílat a ukládat, editovat texty doplňující prezentaci, a přidávat další uživatele či obrázky. Také jsem dbal na to, aby použité technologie byly plně kompatibilní se serverem, které má k dispozici fakulta optiky přírodovědecké fakulty a vyžadovaly minimum úprav.

Pro veřejnost jsou naměřená data znázorněna graficky a pohyb kyvadla je animovaný v reálném čase na titulní straně webové prezentace včetně doplňujících textů a možnosti vybrat si animaci a grafy se starší naměřenou hodnotou. Rozhodl jsem se proto udělat přístup do uzavřené části, který je určen zejména studentům zabývajících se problematikou Foucaultova kyvadla bez možnosti registrace, přístup je tedy řízený a je možný pouze na pozvání administrátora stránek. Současně je v této části i administrace titulní strany. Administraci titulní strany jsem omezil na práva, která jsou možná každému uživateli nastavit.

Původně jsem zamýšlel vytvořit samostatnou aplikaci pro prezentaci na chytré telefony, ale této myšlenky jsem se nakonec vzdal, protože přizpůsobení webových stránek umožnilo bezproblémovou webovou prezentaci naměřených dat.

Conclusions

The major aim of this project was to ensure and extend the capabilities of Foucault pendulum, which is placed in Palacky University's Faculty of Science building. It was, in particular, the storage of measured data and the subsequent web presentation of that data, which I divided into two parts: open and closed. My primary goal was to make this work as adaptable as possible, with as little modifications by the programmer as feasible in subsequent adjustments, especially because it would be maintained by prof. Tomáš Opatrný. As a result, regardless of the inclusion of additional features, it is possible to alter the measured data to be transmitted and preserved at any time, edit the accompanying words, and add other users or pictures. I also ensured that the technologies utilized were entirely compatible with the server, which is available to the Faculty of Science's Faculty of Optics and only required a little amount of configuration.

On the main page of the online presentation, the measured values are visually shown, and the movement of the pendulum is animated in real time, with extra messages and the option to pick animation and graphs with an older measured value. As a result, I decided to create access to the closed section, which is meant specifically for students working with the topic of Foucault's pendulum, without the need to register, and therefore access is regulated and available only on the site administrator's invitation. At the same time, this section is responsible for the administration of the title page. I have restricted access to the cover page to rights that may be assigned to each user.

I had planned to build a separate application for smartphone display, but I finally abandoned that plan because the website's customisation allowed for a seamless web presentation of the measured data.

A Testování aplikace

K otestování webové prezentace je potřeba webový server Apache, MariaDB a PHP 7.4 nebo 8.0. K tomu lze použít například program [Xampp](#) s PHP verzí 7.4 nebo 8.0. Pokyny pro správné nahrání do programu XAMPP, nebo na jiný webový server, jsou dostupné v `readme.txt` na přiloženém CD. Ke správnému otestování příchozích dat a animaci kyvadla a grafů, je nezbytné Foucaultovo kyvadlo s Arduinem. Proto jsem vygeneroval SQL soubor příchozích dat nasbíraných z Foucaultova kyvadla, která je možná nahrát do vytvořené databáze a jsou uložena ve složce "Data dostupná na přiloženém CD".

Veřejnou část webové prezentace je rovněž možné otestovat na webových stránkách s prezentací foucault.optics.upol.cz, data jsou reálně přenašena z Arduina a může se stát, že bude posunutý senzor. V takovém případě grafy a animace nebudou zobrazovat hodnoty a pro zobrazení animace kyvadla a grafů lze použít historii měření, například 25.06.2021, kde je možné pozorovat experiment útlumu pohybu Foucaulova kyvadla mezi 8. a 10. hodinou ranní. K vytvoření přístupu na testování uzavřené administrátorské části je možné mě kontaktovat na emailu pavel.cvek01@upol.cz.

B Obsah přiloženého CD

Tato část popisuje obsah přiloženého CD

require/

Tento adresář obsahuje PHP soubory nezbytné pro správné připojování k SQL databázi na serveru.

install/

Tento adresář obsahuje instalační PHP soubor, který vytvoří tabulky v SQL databázi a vytvoří administrátorský přístup k webovým stránkám. Instalační php soubor je rovněž možné použít k opravě poškozené SQL tabulky.

publicHTML/

Tento adresář obsahuje všechny potřebné soubory včetně obrázků a ikon pro nahrání na webový server.

Arduino/

Tento adresář obsahuje zdrojový kód pro Arduino.

NodeMcu/

Tento adresář obsahuje zdrojový kód pro NodeMcu.

doc/

Text práce ve formátu PDF, vytvořený s použitím závazného stylu KI PřF UP v Olomouci pro závěrečné práce, včetně všech příloh, a všechny soubory potřebné pro bezproblémové vygenerování PDF dokumentu textu (v ZIP archivu), tj. zdrojový kód textu, vložené obrázky, apod.

readme.txt

Instrukce jak postupovat při kopírování souborů na webový server.

Navíc CD obsahuje:

data/

Tento adresář obsahuje ostrá ukázková data uložená ve formátu SQL. Data jsou nasbírána v období od 05.02.2021 do 16.08.2021

Literatura

- [1] PUŽMANOVÁ Rita. *TCP/IP v kostce*, Kopp, 2009, ISBN: 978-80-7232-388-3
- [2] SELECKÝ Matúš. *Arduino*, Computer Press, 2016, ISBN: 978-80-251-4840-2
- [3] VODA Zbyšek; tým HW KITCHEN. *Průvodce světem Arduina 2.vydání*, Spark-Fun, 2017, ISBN: 978-80-87106-93-8
- [4] WELLING Luke; THOMSON Laura. *Mistrovství PHP a MySQL*, Computer Press, 2017, ISBN: 978-80-251-4892-1
- [5] PHP.NET, *PHP Reference*. [online]. Dostupné z: <https://www.php.net/manual/en/langref.php/>
- [6] W3SCHOOLS. *JavaScript and HTML DOM Reference* [online]. Dostupné z: <https://www.w3schools.com/jsref/default.asp/>
- [7] MDN WEB DOCS. *JavaScript Reference* [online]. Dostupné z: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/>
- [8] HYSLOP Bruce; CASTRO Elizabeth. *HTML5 a CSS3*, Computer Press, 2012, ISBN: 978-80-251-3733-8
- [9] GASSTON Peter *CSS3*, Computer Press, 2016, ISBN: 978-80-251-4641-5
- [10] W3SCHOOLS. *MySQL Reference* [online]. Dostupné z: https://www.w3schools.com/mysql/mysql_sql.asp