



Střední průmyslová škola strojní  
a elektrotechnická a Vyšší odborná škola,  
Liberec 1, Masarykova 3

# DATA BÁZE LETŮ PRO AEROKLUB LIBEREC

Dlouhodobá maturitní práce

Autor  
Obor  
Vedoucí práce  
Školní rok

**Tomáš Kulhavý**  
**Informační technologie**  
**Ing. Tomáš Kazda, DiS.**  
**2020/2021**



# Příhláška k maturitní práci

**Jméno a příjmení studenta**

Tomáš Kulhavý

**Název práce**

Databáze letů pro Aeroklub Liberec

**Třída**

P4

**Školní rok**

MP 2020/2021

**Přidělené role****Vedoucí práce**

Podpis(y)

Tomáš Kazda

<b>Obecná ustanovení</b>	Vypracování a odevzdání práce proběhne v souladu s platnými normami (vyhláška 177/2009 Sb.) a aktuálním dokumentem "Pokyny k vypracování prací" vydaným školou.
	Práce bude hodnocena z hlediska jejího praktického využití, zvládnutí dokumentace po věcné i formální stránce a obhajoby celé práce. Student byl seznámen s kritérii hodnocení maturitní práce.
	Práce bude odevzdána ve dvou stejnopisech vázaných pevnou nebo kroužkovou vazbou.
	Veškeré náklady na MP včetně vyhotovení obou tištěných kopií si student hradí sám.
<b>Licenční ujednání</b>	Ve smyslu § 60 (Školní dílo) autorského zákona č. 121/2000 Sb. poskytují SPŠSE a VOŠ Liberec výhradní a neomezená práva k využití této mé maturitní práce.
	Bez svolení školy se zdrím jakéhokoliv komerčního využití mé práce.
	Pro výukové účely a prezentaci školy se vzdávám nároku na odměnu za užití díla.

**Finanční rozvaha - odhad celkových finančních nákladů na práci**

v Kč	Náklady celkem	Hrazené školou
<b>výrobní</b>	0	0
<b>na služby</b>	0	0

Jedná se o MP, jejíž vypracování si škola vyžádala? **Ano Ne****Podpis studenta (vyjadřuje souhlas s uvedenými údaji a ujednáními)**

V Liberci dne 29.09.2020

**Konzultant**

Práci doporučuji

**Stanovisko předmětové komise**

Práci doporučuji

**Stanovisko třídního učitele**

Práci doporučuji

**Stanovisko garanta oboru**

Práci doporučuji

**Souhlas ředitele školy**

Práci schvaluji

Podpis

Podpis

Podpis

Podpis

Podpis

Podpis

# Zadání dlouhodobé maturitní práce

**Název**

Databáze letů pro Aeroklub Liberec

**Předmět**

WEB

**Téma**

Práce bude sloužit jako databáze pro přelety pro AK Liberec. Piloti budou nahrávat své lety a poté budou vyhodnoceny a bodovány. Poslouží jako žebříček pilotů v klubu se statistikami.

**Použité prostředky**

IGC záznamy letů, ASP.NET

**Cíle práce**

1	Vyhodnocení IGC záznamu z letu
2	Prohlížení záznamu letu
3	Statistika náletu pilotů

**Osnova práce**

1	Analýza požadavků - vytvoření katalogu, use-case
2	Volba SW technologie, zdůvodnění
3	Návrh ER/logického schématu databáze
4	Návrh UI rozhraní (Figma, XD)
5	Návrh systémové části aplikace (uživatelé, identita, administrace)
6	Rozbor IGC, návrh vizualizace dat (mapy)
7	Návrh části s vizualizací letu
8	Návrh výstupů se statistikou náletů
9	Nasazení na hosting
10	Ladění a zpětná vazba uživatelů

## Anotace (Resumé)

Práce bude sloužit jako databáze pro přelety pro Aeroklub Liberec. Piloti budou nahrávat své lety a poté budou vyhodnoceny a bodovány. Poslouží jako žebříček pilotů v klubu se statistikami.

Přináší pilotům v Aeroklubu Liberec přehled o statistikách a náletech pilotů. Může také posloužit při vybírání pilotů na velké závody jako je např. Plachtařské mistrovství České republiky juniorů.

## Summary

The work will serve as a database for flights for Aeroklub Liberec. Pilots will record their flights and then be evaluated and scored. It will serve as a ranking of pilots in the club with statistics.

It brings the pilots in the Aeroklub Liberec an overview of the statistics and raids of the pilots. It can also be used to select pilots for large competitions, such as the Czech Junior Gliding Championship.

## Čestné prohlášení

Prohlašuji, že jsem předkládanou maturitní práci vypracoval sám a uvedl jsem veškerou použitou literaturu a bibliografické citace.

V Liberci dne 11.03.2021

.....  
Tomáš Kulhavý

## Obsah

Úvod.....	1
1 Použité technologie .....	2
1.1 ASP.NET Core 5.0 .....	2
1.2 Entity Framework Core .....	2
1.3 React.....	2
1.3.1 Reducer .....	3
1.3.2 Další knihovny.....	3
1.4 Visual Studio 2019 .....	3
1.5 Visual Studio Code .....	4
1.6 MySQL.....	4
2 Soutěžní bezmotorové létání.....	6
2.1 Pravidla soutěžního bezmotorového létání.....	6
2.1.1 AAT (Assigned area task).....	6
2.1.2 Rychlostní přelet (Racing).....	7
2.1.3 Grand Prix .....	8
3 IGC.....	9
3.1 IGC struktura.....	9
3.1.1 Jednotky.....	9
3.1.2 Pojmenování souboru .....	9
3.1.3 A – FVU (Flight Verification Unit) .....	10
3.1.4 G – Security (Bezpečnostní kód) .....	10
3.1.5 H – File header (Hlavička souboru) .....	10
3.1.6 B – Fix (Jednotlivý záznam letu).....	11
3.1.7 C – Task (Úloha letu).....	11
3.1.8 D – Differential GPS (Jiný GPS záznamník).....	12
3.1.9 F – Satelit.....	12
3.1.10 L – Log book (Informace k letu).....	12
3.2 IGC záznamník.....	12
4 Práce s příchozími daty.....	14
4.1 IGC parser.....	14
4.2 Zjištění poloměrů otočných bodů .....	14
4.3 IGC-XC-Score .....	14
4.4 Počítání vzdálenosti a průměrné rychlosti .....	14
4.5 Počítání TOP 3 letů .....	15
5 Funkční stránka .....	16
6 Realizace aplikace .....	17
6.1 Návrh databáze .....	17
6.1.1 Návrh databáze letu .....	17
6.1.2 Návrh databáze uživatele.....	18
6.2 Funkcionalita stránky.....	19
6.2.1 Hlavní stránka .....	19
6.2.2 Identity .....	20
6.2.3 Výpis letů.....	22
6.2.4 Výpis letů pilota .....	23
6.2.5 Výpis pilotů.....	24
6.2.6 Výpis pořadí pilotů .....	25

6.2.7	Zobrazení letu.....	26
6.2.8	Celkové statistiky .....	27
6.2.9	Nahrávání letu .....	28
6.2.10	Feedback.....	29
	Závěr.....	30
	Seznam obrázků.....	31
	Použitá literatura.....	32
A.	Seznam přiložených souborů .....	1
B.	Další příloha .....	2





# Úvod

Tuto práci jsem si vymyslel sám, dlouho jsem uvažoval nad tím, co bych mohl dělat jako maturitní práci a pak mě napadla databáze letů a poté jejich vyhodnocení. v rámci výuky ve čtvrtém ročníku jsme začali programovat v jazyce React. Díky této práci si rozšířím znalosti řešení webových stránek v jazyce React a ASP.NET API. Zvolené téma jsem si vybral, jelikož sám jsem pilotem v Aeroklubu Liberec, tak jsem chtěl naprogramovat stránky, které budou sloužit jako žebříček pilotů v aeroklubu nebo také jako statistiky pilotů. Tato stránka by mohla pomoci členům aeroklubu při výběru pilotů na větší soutěže, jako je Mistroství České republiky nebo Mistroství regionů.

# 1 Použité technologie

## 1.1 ASP.NET Core 5.0

ASP.NET Core je univerzální framework pro vytváření webových aplikací, služeb nebo aplikací pro internet věcí (IoT) v jazyce C Sharp. Aplikace se dají nasazovat místně nebo na cloudu. Prostředí je vhodné pro vytváření webového uživatelského rozhraní a webových rozhraní API.

Nejnovější podporovaná verze frameworku .NET Core je 5.0. Tato verze je také použita v aplikaci. (1)

## 1.2 Entity Framework Core

Entity Framework je nástroj pro objektově relační zobrazení nazývané tzv. ORM (Object Relational Mapper). Je vyvíjený v jazyce C# pro platformu .NET. Podporuje dotazy LINQ, aktualizace a migrace schématu. EF Core podporuje několik databází, včetně SQL Database, která se dá používat místně, ale i na Azure Cloudu. Dále podporuje SQLite, MySQL, PostgreSQL a Azure Cosmos DB. (2)

## 1.3 React

React nebo také známý jako ReactJS je JavaScriptová knihovna pro tvorbu uživatelského rozhraní vyvíjená společností Facebook. Slouží převážně k tvorbě Single Page Application. Tato aplikace se nachází pouze na jedné stránce. Proto single page application. Je výhodný pro práci s daty, které se rychle mění. Jelikož virtualizaci zajišťuje DOM (Document Object Model). Virtuální DOM uchovává uživatelské rozhraní v paměti. DOM je pak synchronizován, kdy je to opravdu potřeba a tím se ušetří mnoho výpočetního času, který je potřeba na vykreslení DOMu.

Pro psaní kódu aplikace, která používá React, existuje syntaxe JSX, která umožňuje vytvářet HTML elementy jejich prostým zápisem pomocí tagů. Soubor je poté potřeba přeložit pomocí Babelu. (3)

### 1.3.1 Reducer

Reducer je jednoduchá funkce, která vezme dva argumenty. v kontextu nám reducer vezme aktuální stav (state) a akci (přes dispatch si vybereme action type). a vrátí nám nový stav na základě těchto dvou dat. (4)

### 1.3.2 Další knihovny

K dalším knihovnám patří:

- Reactstrap
- Formik
- React-leaflet
- Igc-parser
- Igc-xc-score
- Axios

Reactstrap slouží k zajištění funkčnosti knihovny Bootstrap s Reactem.

Formik je knihovna, která validuje vstupy, formátuje a vypořádává se s chybami, tím že v reálném čase vidíme, jestli náš vstup je validní.

React-leaflet je knihovna pro práci s mapami. v této aplikaci ji používám pro zobrazení úlohy letu a skutečně trasy letu.

Igc-parser je knihovna, která parsuje IGC soubor letu do JSONu.

Igc-xc-score je knihovna, která IGC soubor vyhodnotí a oboduje.

Axios je knihovna se zpětným voláním. Umožňuje asynchronně vykreslovat požadavky.

Další přehled použitých knihoven je v souboru package.json v adresáři ClientApp.

## 1.4 Visual Studio 2019

Visual Studio 2019 je vývojové prostředí IDE (Integrated Development Environment), které vyvíjí firma Microsoft. Visual Studio umožňuje vyvíjet aplikace v řadě programovacích jazycích. Nejpoužívanější je platforma .NET.

Obsahuje mnoho jazykových služeb. Ty slouží k tomu, aby debugger a editor podporoval ten programovací jazyk, pro který existuje daná jazyková služba. Dále obsahuje mnoho

šablon nebo také grafické rozhraní GUI (Graphic User Interface). Podporuje také například vyvíjení her v Unity.

Visual Studio 2019 jsem si zvolil, jelikož mi toto prostředí vyhovuje pro vývoj back-end aplikace. (5)

### 1.5 Visual Studio Code

Visual Studio Code je open-source editor zdrojových kódů. Vyvíjený je společností Microsoft. Podporuje verzovací nástroj Git. Obsahuje také kontextový našeptávač nebo zvýrazňovač syntaxe. Editor je naprogramovaný v jazyce JavaScript a TypeScript. Běží na frameworku Electron, na kterém běží i konkurenční Atom.

Visual Studio Code jsem si vybral, protože mi vyhovuje pro tvorbu aplikací v Reactu. a obsahuje velké množství doplňku z třetí strany, které zajišťují podporu mnoha jazyků. Má také lepší highlighting (zvýraznění) kódu, než Visual Studio 2019, proto je pro mě přehlednější. (6)

### 1.6 MySQL

MySQL je relační databázový model vytvořený švédskou firmou MySQL AB, nyní vlastněný Oracle Corporation. Je k dispozici jako free and open-source pod licencí GPL (General Public License) a zároveň pod několika komerčních placených licencí. MySQL je velmi oblíbená a častá kombinace pro základní software webového serveru.

Komunikace probíhá pomocí jazyka SQL. Jako u ostatních SQL databází se jedná o dialekt tohoto jazyka s některými rozšířeními. Jedná se o strukturovaný dotazovací jazyk, tzn. že nepíšeme to, jak má databáze data získat, ale pouze to, jaké data chceme. MySQL je optimalizovaná především na rychlost.

Databáze je založená na tabulkách. Každá tabulka obsahuje informace jednoho typu. Data jsou v databázi uložena v řádcích a v sloupcích. Každý řádek v tabulce představuje jednu položku v databázi. Každý sloupec má pevně stanovený datový typ a každý řádek má svůj unikátní identifikátor.

Při navrhování databáze musíme zvážit správný druh vztahů mezi tabulkami a entit. Nejjednodušší vztah představuje relace 1:1, kde Pk (Primary key) v jedné z tabulek odkazuje na data v druhé tabulce. Příkladem této relace může být vztah mezi státem a jeho hlavním městem. Další relací je 1:N. Ta představuje vztah, kde Pk v jedné tabulce

odkazuje na více dat v druhé tabulce. Tento vztah může reprezentovat např. třída a studenti. a poslední relací je N:M, kde každý záznam v jedné tabulce může mít více dat ve druhé. Příklad toho vztahu je např. filmy a herci.

Název souboru odpovídá názvu tabulky. Co se týče rozlišování velkých a malých písmen, tak to závisí na platformě. v kombinaci MySQL a Windows se velikost písmen nerozlišuje. Nedoporučuje se vkládat znaky s diakritikou do názvu tabulek nebo databází. (7)

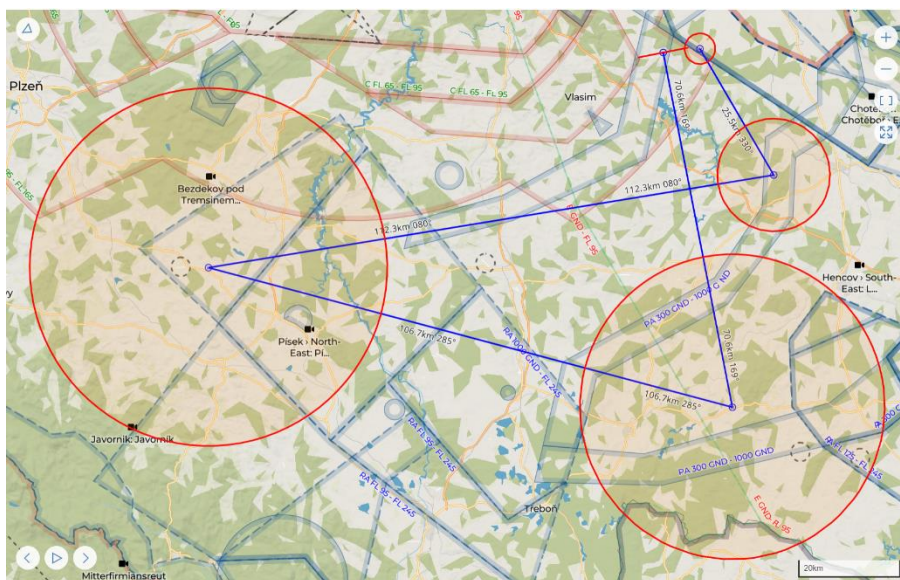
## 2 Soutěžní bezmotorové létání

### 2.1 Pravidla soutěžního bezmotorového létání

Soutěžní bezmotorové létání se řídí pravidly Mezinárodní letecké federace FAI (Fédération Aéronautique Internationale). Soutěžní létání se dělí na různé typy závodů. Závody se konají v mnoha typech tříd. Jelikož každý typ kluzáku, má jiný koeficient, tak se na závodech tyto kluzáky rozdělí na několik tříd, kde mají většinou podobný koeficient. Třídy se dělí např. na dvousedadlové většinou nebo na jednosedadlové většinou s rozpětím 15 m, 18 m nebo 21 m a více (tzv. otevřená třída). Koeficient se většinou přidělí z toho důvodu, že každý typ letadla má jiné výkonnostní vlastnosti. a aby nebyla zvýhodněna letadla s výbornými letovými vlastnostmi, tak se jejich body následně spočítají s jejich koeficientem. To znamená, že piloti s lepšími letadly musí létat rychleji. (8)

#### 2.1.1 AAT (Assigned area task)

Jedním způsobem je tzv. AAT (Assigned area task), jedná se o velké kružnice o poloměru např. až 50 km, které jsou místo standardních otočných bodů stanoveny po předem naplánované trati.



Obrázek 1 AAT (Assigned Area Task)

Pilot musí těmito kružnicemi proletět a je pouze na jeho rozhodnutí, jestli si zaletí, až na konec kružnice nebo pouze na její začátek. Jeho rozhodnutí závisí na počasí a určité predikci vývoje počasí. Jelikož je předem stanovený čas na trati, pilot musí naplánovanou trať zaletět co nejrychleji, aniž by se vrátil na letiště odletu před skončením stanoveného

času. Tento způsob úlohy je více o taktice, protože počasí se neustále mění, pokud se např. pilot rozhodne zaletět na konec kružnice, tzn. že uletí větší vzdálenost než, ti co kružnici protnou pouze na jejím začátku, tak je možné, že se během té doby změní počasí a pilot bude muset v této kružnici přistát. Jelikož mu meteorologické podmínky nedovolí pokračovat. Nebo je také možnost, že se trefí do dobrého časového intervalu a počasí bude spolupracovat a díky dobrým termickým podmínkám uletí více kilometrů za menší čas než ostatní závodníci, a tím získá bodový náskok. Pokud by se stalo, že by pilot přiletěl do cíle např. o 30 minut dříve před skončením stanoveného času, tak jeho čas na trati se bude rovnat stále tomu stanovenému času. Tzn. že se mu tím zmenší průměrná rychlost na trati a přijde tak o spoustu bodů. Jestliže pilot přiletí po stanoveném čase, tak se mu do jeho délky letu po trati počítá skutečný jeho čas na trati. v zásadě jde o to uletět co nejvíce kilometrů za předem daný čas.

Tento typ úlohy se převážně dává soutěžícím, pokud je na letový den předpověď počasí neurčitá.

### 2.1.2 Rychlostní přelet (Racing)

Druhým způsobem soutěžního létání je tzv. racing neboli rychlostí přelet. v tomto typu úlohy se naplánuje trať s pevně danými body na trati.



Obrázek 2 Rychlostní přelet

Pilot musí tuto celou naplánovanou trať zaletět, za co nejmenší čas a tím docílí vysoké průměrné rychlosti na trase. Poloměr otočných bodů je 500 m.

Tento typ úlohy se převážně zadává, pokud je předpověď počasí na letový den zřetelná a pořadatelé vědí jaký bude vývoj počasí na trati po celý den.

### 2.1.3 Grand Prix

Tento typ úlohy patří k novějším a létá se na závodech, kde je menší počet letadel. Organizátor na den vypíše trať s pevně danými doby, který piloti musí obletět. v jeden přesně daný čas se otevře startovací páska a všichni piloti musí v jeden čas odletět a trať. Jsou zde dané limity na maximální výšku a rychlost při průletu startovacím oknem. Bodování je pak jednodušší, protože všichni piloti odletěli ve stejný čas na trať a vítězí ten, co do cíle přiletí jako první.



## 3 IGC

Tento standard datových souborů byl vyvinut ve spolupráci se softwarovými vývojáři záznamových zařízení. Standard se nazývá IGC (International Gliding Commission). Jeho účelem je usnadnit zavedení technologie GPS (Global Positioning System) do soutěžního bezmotorového létání pod sportovním řádem FAI. Mezinárodní letecké federace FAI tento standard používá nejen na soutěžní létání, ale také na FAI odznaky nebo na rekordní lety. IGC standard ověřuje pozice otočných bodů, souřadnice letadla a zabezpečuje data.

Tento standard byl představen a přijat na zasedání v září roku 1994. (9)

### 3.1 IGC struktura

Každý záznam v IGC souboru začíná specifickým znakem, který identifikuje druh záznamu.

#### 3.1.1 Jednotky

Všechny data IGC záznamu jsou zaznamenávány v následujících jednotkách.

Pro záznam času se používá koordinovaný světový čas UTC (Coordinated Universal Time), je to základní čas pro všechna časová. Jedná se o souřadnicový čas a odvíjí se od základního poledníku v Greenwich. (10)

Na měření délky letu se používají metry.

Rychlost letu měříme v kilometrech za hodinu.

Datový formát letu je DDMMYY (den, měsíc, rok)

Výška letu se udává v metrech. Zaznamenává se GPS výška, ale také výška, kterou záznamové zařízení bere ze svého barometrického senzoru, který se musí každých 5 let zkalibrovat.

#### 3.1.2 Pojmenování souboru

Každý IGC záznam má své jméno, to je dané normou ISO 9660. Používá se pro všechny oficiální záznamy letu. Ostatní názvy souborů jsou povoleny pro obecné používání, ale FAI ani další soutěžní organizace jiný název neakceptují. Všechny znaky v názvu souboru jsou alfanumerický.

„A“ reprezentuje 10, „B“ je 11 a „C“ je 12. Znaky 1-9 a A-C používáme pro měsíce.

Pro zápis roku se používá modulo 10, takže každých 10 se začíná znovu.

Pro číslo letu ten daný den se začíná číslem 1.

Sériové číslo záznamové jednotky je unikátní číslo. Pro každého výrobce záznamových zařízení je k dispozici 46656 identifikačních čísel.

- YMDCXXXF.IGC
  - Y = rok
  - M = měsíc
  - D = den
  - C = výrobní ID
  - X = sériové unikátní číslo
  - F = číslo letu za den

### 3.1.3 A – FVU (Flight Verification Unit)

Tento typ záznamu začíná písmenem „A“. Specifikuje unikátní sériové číslo záznamového zařízení.

- AMNNNNN
  - A = typ záznamu
  - M = ID výrobce
  - N = unikátní ID

### 3.1.4 G – Security (Bezpečnostní kód)

Záznam „G“ ověřuje, zda data ASCII nebyla během letu nebo po něm změněna. Používá se pro ověření záznamu. Bezpečnostní kód musí být vygenerován FVU. Bezpečnostní kód musí obsahovat platné znaky ASCII tabulky.

### 3.1.5 H – File header (Hlavička souboru)

Hlavička souboru začíná písmenem „H“. Jsou zde informace o datumu, kdy byl let letěn. Dále jméno pilota, pokud je kluzák dvoumístný, uvádí se jméno druhého pilota. Také zde najdeme typ kluzáku, třídu, kam kluzák patří, registraci kluzáku a také jeho soutěžní znak. Dále jsou zde informace o verzi firmwaru záznamového zařízení, typ zařízení. Nebo

označení antény pro snímání pozice a senzoru pro snímání výšky letu. Je zde také uvedena časová zóna letu.

### 3.1.6 B – Fix (Jednotlivý záznam letu)

Fix neboli jednotlivý záznam letu, se zaznamenává pravidelně po pár vteřinách. Dá se zaznamenávat každou 1 vteřinu až 5 vteřin. Záznam začíná písmenem „B“. Obsahuje v sobě informace o čase, zeměpisné šířce a délce, validaci fixu, výška z barometru, GPS výška, přesnost fixu a pokud záznamové zařízení detekuje podle hluku motor, obsahuje i otáčky motoru.

- BHHMMSSDDMMMMMNDDDDMMMMMEVPPPPPGGGGAAAARRRR
  - HHMMSS = časový záznam
  - DDMMMMMN = zeměpisná šířka
  - DDDMMMMME = zeměpisná délka
  - V/A = validace fixu (A = valid, v = warning)
  - PPPPP = výška z barometru
  - GGGGG = GPS výška
  - AAA = přesnost fixu
  - RRRR = otáčky motoru (pokud je hluk detekován)

### 3.1.7 C – Task (Úloha letu)

Záznam, který začíná písmenem „C“ označuje úlohu letu. Obsahuje datum a čas letu, ID úlohy, která začíná číslem 0001 (označuje pořadí deklarované tratě daný den), GPS souřadnice vzletu a přistání a také GPS souřadnice otočných bodů na trati a jako poslední je název otočného bodu.

- CDMMYYHHMMSSFDFMFYIIITTEXTSTRING (informace o úloze)
- CDDMMMMMNDDDDMMMMMETEXTSTRING (otočný bod)
  - DDMMYY = datum
  - HHMMSS = čas
  - IIII = ID úlohy
  - DDMMMMMNDDDDMMMMME = GPS souřadnice

### 3.1.8 D – Differential GPS (Jiný GPS záznamník)

Záznam „D“ nám dává informace, jestli záznamové zařízení snímá GPS souřadnice pomocí jiného GPS zařízení. Pokud je absence těchto záznamů, znamená to, že záznamové zařízení používá integrovanou GPSku.

- DQSSSS
  - Q = GPS kvalifikátor
  - S = ID stanice

### 3.1.9 F – Satelit

Záznam „F“ zapisuje informace o aktuálně používaném satelitu. Vždy se zaznamená, pokud se přijímací satelit změní.

- FHHMMSSAABBCCDD
  - HHMMSS = čas
  - AABBCDD = ID satelitu

### 3.1.10 L – Log book (Informace k letu)

Tento záznam „L“ umožňuje přidat několik textových řádků ve volném formátu. Jelikož záznam „L“ není pokryt bezpečnostním kódem, umožňuje nám přidat do souboru informace i po dokončení letu, např. nějaké komentáře k letu atd.

## 3.2 IGC záznamník

IGC záznamník neboli logger, se používá pro zaznamenávání celého letu. Jehož výstupem je následně IGC soubor. Existuje několik typů loggerů.

Mezi starší patří ty, co mají v sobě nainstalovaný hardware pouze pro zaznamenávání letu. Tento logger je připojený k baterce, jelikož v sobě baterky nemají. Dále se musí pomocí kabelů připojit k externí anténě, pokud v ní v sobě nemají. Také se dají připojit pomocí kabelu k externí navigaci, kde díky IGC loggeru uvidíme naši polohu. Tyto navigace mají i palubní počítač, který dokáže spočítat aktuální klouzavost letadla, předpokládaný přílet do cíle nebo na otočný bod. Ukazují nám také základní informace o letu jako je rychlost, výška a vario.

Mezi novější patří už IGC záznamníky, co v sobě mají logger, navigaci i palubní počítač už zabudovaný. Také mají svou vlastní baterku, takže nemusí být napojený na baterku. Tyto navigace vydrží už poměrně dlouhou dobu, takže pokud pilot letí několikahodinový přelet nebo i přelet, který trvá celý den tak mu tento logger na jedno nabití stačí. Vydrží až 12 hodin.

## 4 Práce s příchozími daty

Všechna příchozí data zpracovávám ihned po vložení souboru na stránku. Data ve formátu JSON pošlu pomocí axiosu do databáze. Lety si mohu následně vylistovat všechny nebo podle pilotů. k zobrazení letu používám mapy a barograf.

### 4.1 IGC parser

IGC parser je open-sourcová knihovna, která mi celý IGC soubor pomocí regulárních výrazů rozparsuje do JSONu. Díky tomu, že jsou data v JSONu, mohu k nim přidávat další informace o letu. a také je následně poslat do FlightLogControlleru, kde je pomocí newtonsoftu deserializuji do databáze.

### 4.2 Zjištění poloměrů otočných bodů

Dále celý soubor rozparsuji pomocí regulárních výrazů ještě jednou. Abych zjistil poloměry otočných bodů, které jsou potřeba pro správné počítání uletěné vzdálenosti a průměrné rychlosti letu.

```
var RE_L = /^L.*(?:=R1)R1(?:\d{2}|\d{1})(?:\d{4}|\d{3}|\d{2})(?:\d{2}|\d{1})\d{2}.*$/;
```

Obrázek 3 Regulární výraz na zjištění poloměrů

### 4.3 IGC-XC-Score

Poté celý IGC soubor zpracuji ještě jednou, abych získal pomocí open-source knihovně IGC-XC-Score body za let.

Bodování letů je následující. Pokud je nadeklarovaná trasa uletěná, pilot získá více bodu, než kdyby let nedoletěl celý. Pokud pilot úlohu vůbec nezačal, nespočítá se žádná průměrná rychlost ani uletěná vzdálenost a nedostane žádné body.

### 4.4 Počítání vzdálenosti a průměrné rychlosti

Vzdálenost letu počítám následovně. Pomocí cyklu „for“, který mi prochází všechny otočné body, si nastavuji radius (poloměr otočných bodů, které jsem získal) a střed kružnic. Do proměnné si uložím vzdálenost mezi body v úloze a zjistím, pomocí cyklu „for“, jestli pilot proletěl daným bodem. Pokud ano, uložím si čas, kdy byl v místě toho bodu. a zvýším proměnnou „countTP“ o jedna. Do proměnné „dist“ si uložím vzdálenost

mezi body. Takto tento cyklus projede celý let. Pokud se zjistí, že pilot bodem v úloze neproletěl, nezvyšuji „countTP“ ani nepřipočítávám vzdálenost. Jakmile tento cyklus projede celý let. Porovná proměnnou „countTP“ (skutečný počet otočných bodů, které pilot proletěl) a počet otočných bodů, které byly nadeklarovány. Pokud se tyto čísla nerovnájí, spočítám vzdálenost mezi body, kterou pilot neuletěl. Tu odečtu od proměnné „dist“ a přičtu vzdálenost mezi bodem, kde pilot přistál a posledním bodem nadeklarované tratě, kterým pilot proletěl. Tímto výpočtem zjistím skutečnou uletěnou vzdálenost

Průměrnou rychlost vypočítám tak, že vydělím vzdálenost letu časem na trati.

### 4.5 Počítání TOP 3 letů

Ihned po deserializaci letu ve FlightLogControlleru, volám metodu „GiveTopBool“. Tato metoda si vytáhne z databáze všechny lety od pilota, který let nahrává z roku, kdy je nahrávaný let letěn. Tyto záznamy seřadím sestupně podle bodů za let a vezmu si pouze tři nejlepší lety.

Pokud počet letů, které jsme si vytáhli z databáze, není větší než dva, tak automaticky přidělí letu, že „Topflight“ se rovná true. a pilotovi se přičtou body za let do jeho „TopScore“ (pilotovi celkové body za nejlepší tři lety).

Jestliže počet letů je větší než dva, tak porovná nahrávaný let s lety v databázi. a pokud má nahrávaný let více bodů, tak jeho „Topflight“ se rovná true a poslední nejlepší let je nastaven na false. Dále odečteme od „TopScore“ poslední nejlepší let a přičteme body za nový nejlepší let.

## 5 Funkční stránka

Tato aplikace slouží pro všechny piloty bezmotorových letadel, kteří chtějí mít své lety uložené v databázi na internetu a také je mít ohodnocené. a chtějí porovnávat své lety s ostatními.

Piloti po nahrání letu mají možnost si svou trajektorii letu prohlédnout na mapě společně s barografem, který znázorňuje výškový profil letu. Pilotovi jsou také vypočteny základní informace o letu, jako je uletěná vzdálenost, průměrná rychlost letu nadeklarované úlohy.

Má také možnost si zobrazit celkové statistiky klubu. Kolik hodin má, který pilot nalétáno nebo kolik kilometrů celkem uletěl.

Na hlavní stránce má pilot možnost vidět pět nejlepších pilotů podle nalétaných hodin, kilometrů a hlavně bodů. Je to bodový součet za nejlepší tři lety v roce, tyto body se sčítají s ostatními roky.

Na stránce „pořadí pilotů“ může aeroklub zvážit, kterého pilota mohou vybrat k reprezentaci klubu např. na Mistroství republiky v bezmotorovém létání. Protože zde vidí bodový nálet pilotů. Bodový nálet je pro výběr pilotů nejdůležitější, jelikož body pilot dostane za nějaký počet odletěných kilometrů v úloze a za průměrnou rychlost letu.

Tato aplikace dokáže počítat přesné body pro typ tratě racing a Grand Prix. Pro úlohu AAT, ještě není naprogramovaná, tak aby počítala správně s časem, který má pilot strávit na trase.



## 6 Realizace aplikace

Tato kapitola bude popisovat, jak jednotlivé stránky fungují a k čemu slouží. Návrh databáze celé aplikace a také jak jsou jednotlivé stránky propojeny s endpointy.

### 6.1 Návrh databáze



Obrázek 4 Návrh databáze

#### 6.1.1 Návrh databáze letu

Hlavní tabulkou databáze je „FlightLogs“, který představuje jeden let. v této tabulce jsou základní informace o letu, jako je jméno pilota, datum letu nebo typ kluzáku. Každá tato tabulka má vztah 1:N, kde jeden uživatel může mít více letů a jeden let může mít pouze jednoho uživatele. Každý let má jednu tabulku „FlightLogAnalyses“, kde jsou uloženy informace o letu jako je např. délka letu, průměrná rychlost letu, uletěná vzdálenost nebo

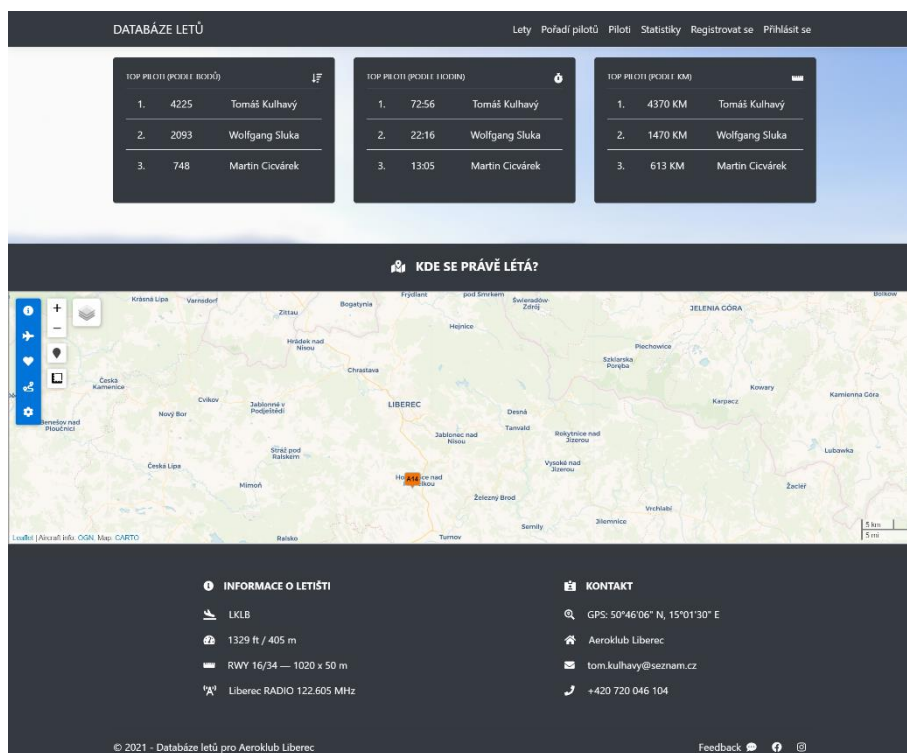
body za let. Relace mezi těmito tabulkami je 1:1. Dále má každý let tabulku „FlightTasks“, zde jsou informace o deklaraci tratě, počet otočných bodů a číslo úlohy v ten den. Relace mezi nimi je také 1:1. Každá tato tabulka má pak několik tabulek s body v úloze letu „Points“. Tato relace je 1:N. v této tabulce nalezneme zeměpisnou šířku a délku bodu a také jeho název. Další tabulkou je „Fixes“ a má vztah s „FlightLogs“ 1:N. Kolekce těchto tabulek pak představuje trasu celého letu. Je to záznam zeměpisné šířky a délky, GPS a barometrické výšky a času. Každý tento fix se zaznamenává pravidelně od 1 až 10 vteřin. Záleží, co si pilot nastaví. Tabulka „DataRecords“ má také vztah s „FlightLogs“ 1:N. a poslední tabulkou je „Extensions“, která má relaci 1:1 s tabulkami „DataRecords“ a „Fixes“. Samostatné tabulky jako je „Stats“ a „Feedback“ nemají žádnou relaci.

### 6.1.2 Návrh databáze uživatele

Hlavní tabulkou je „AspNetUsers“, který představuje jednoho uživatele. Obsahuje informace jako je email, uživatelské jméno, křestní jméno a příjmení. Dále také svoje osobní statistiky celkové náletu hodin, kilometrů a bodů. Tato tabulka má relaci N:M k tabulce „AspNetRoles“ a jsou propojené díky „AspNetUserRoles“. Díky těmto tabulkám můžeme uživatelům přidat role.

## 6.2 Funkcionalita stránky

### 6.2.1 Hlavní stránka

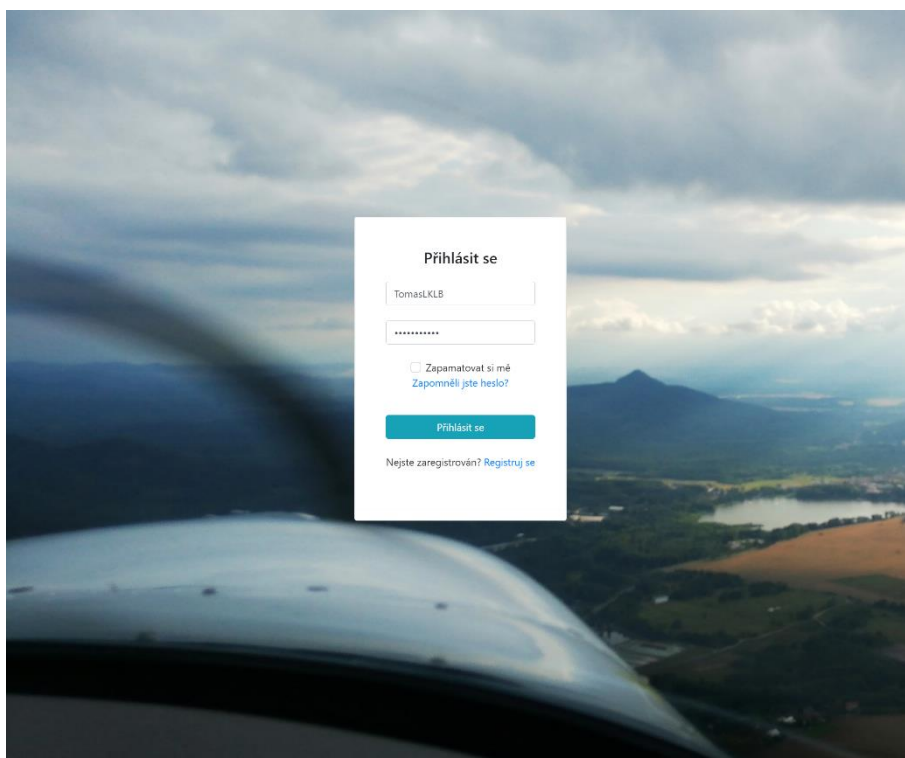


Obrázek 5 Hlavní stránka

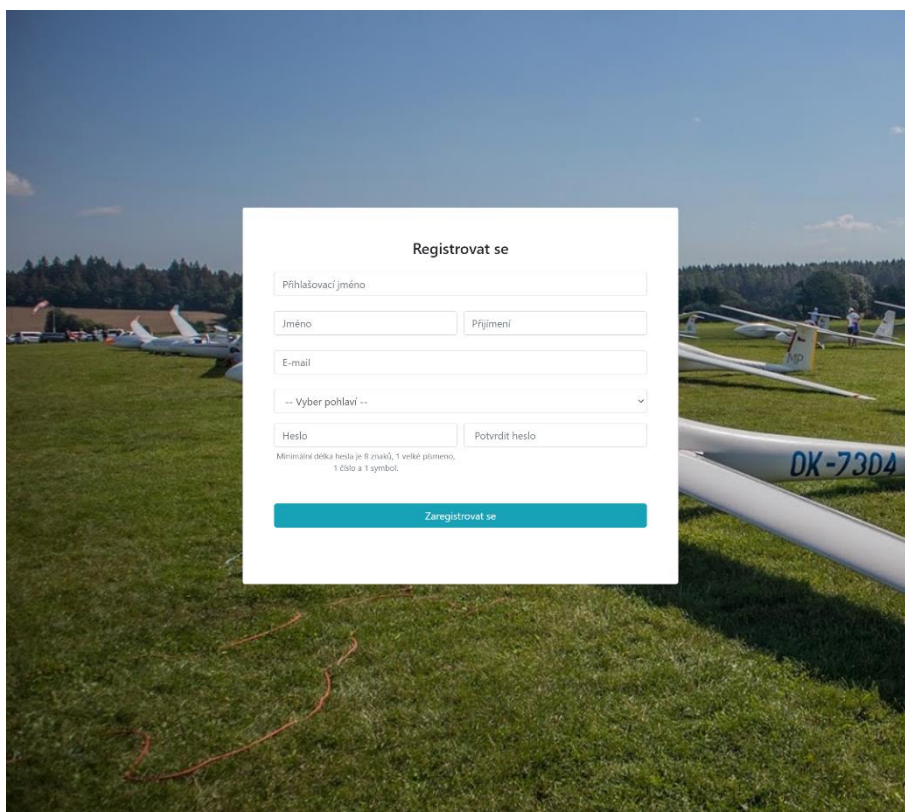
Na hlavní a zároveň úvodní stránce se nachází navigační lišta, která slouží k pohybu na stránce. v těle stránky máme tři tabulky, které nám dávají stručný přehled o nejlepších pilotech podle náletu bodů, hodin a kilometrů. a pod nimi mapu s aktuálním provozem letadel. Pomocí html tagu iframe je integrovaná ze stránky glideandseek.com. v zápatí stránky jsou informace o letišti a kontakt na mě. Také zde jsou odkazy na můj Instagram, Facebook a stránku „Feedback“, kde můžete sdělit názor na tuto aplikaci.

Tabulky s nejlepšími piloty získávají informace pomocí „FlightLogController“, který nám z databáze vybere pět nejlepších pilotů podle nalétaných bodů, hodin a kilometrů.

## 6.2.2 Identity



Obrázek 7 Stránka pro přihlášení



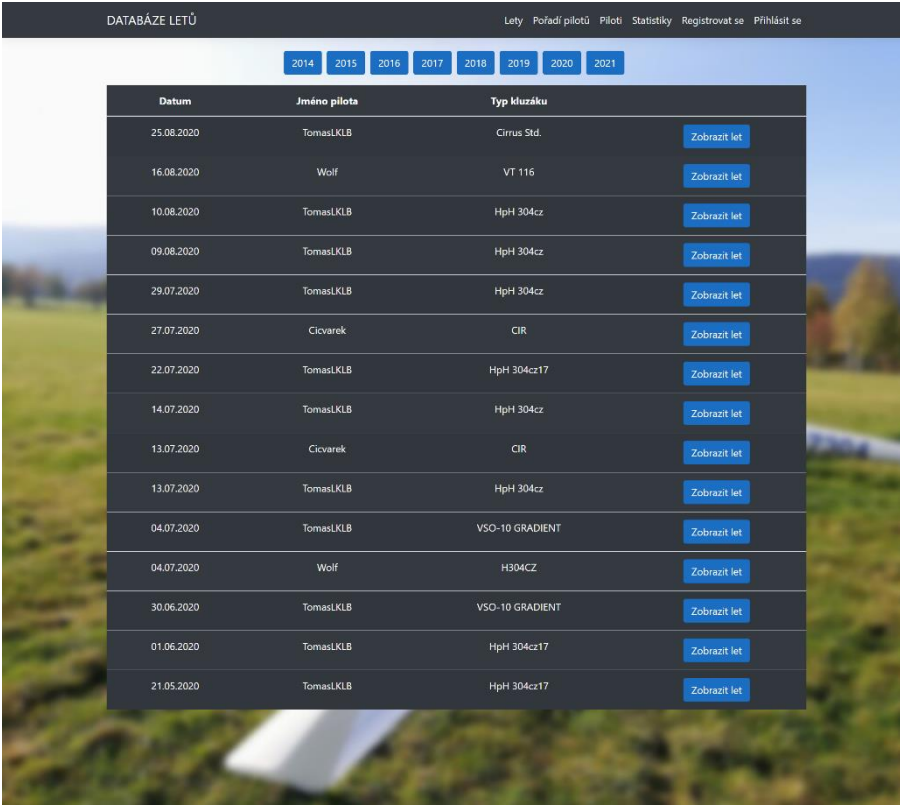
Obrázek 6 Stránka pro registraci

Stránka pro přihlášení obsahuje formulář, kde se zadávají přihlašovací údaje. Je nutné zadat uživatelské jméno a heslo. Uživatel může zaškrtnout „Zapamatovat si mě“, aby zůstal uživatel přihlášený.

Stránka pro registraci obsahuje formulář, do kterého je potřeba zadat uživatelské jméno, křestní jméno, příjmení, email, pohlaví a heslo. To musí obsahovat minimálně osm znaků, jedno velké písmeno, jedno číslo a jeden symbol.

Tyto stránky s přihlašováním a registrací obsluhuje samotné ASP a „OidcConfigurationController“.

## 6.2.3 Výpis letů



DATA BAZE LETŮ

Lety Pořadí pilotů Piloti Statistika Registrovat se Přihlásit se

2014 2015 2016 2017 2018 2019 2020 2021

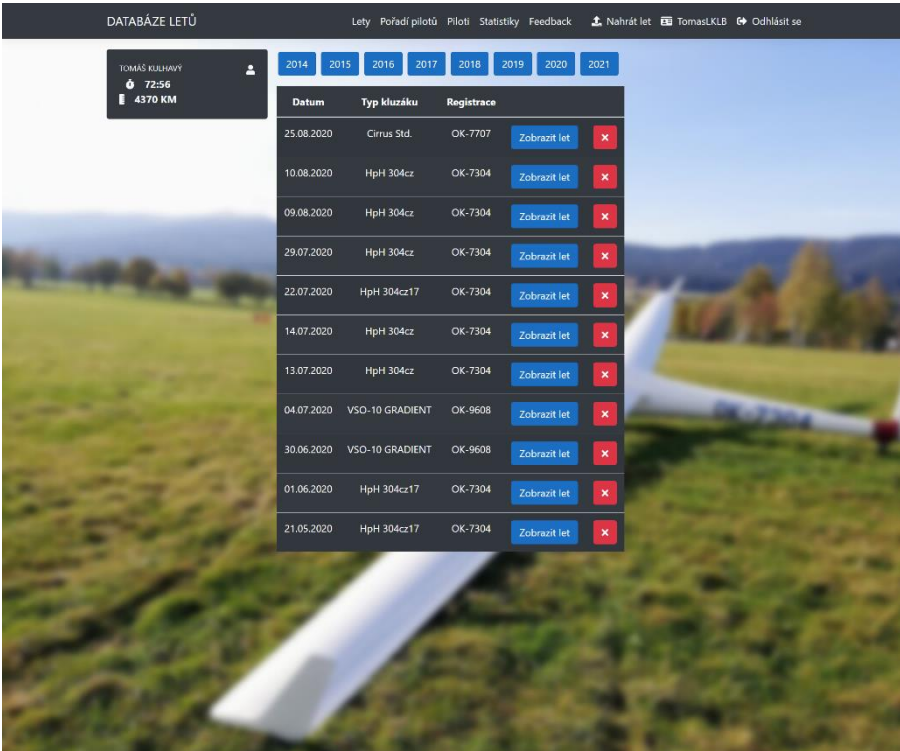
Datum	Jméno pilota	Typ kluzáku	
25.08.2020	TomasLKLB	Cirrus Std.	Zobrazit let
16.08.2020	Wolf	VT 116	Zobrazit let
10.08.2020	TomasLKLB	HpH 304cz	Zobrazit let
09.08.2020	TomasLKLB	HpH 304cz	Zobrazit let
29.07.2020	TomasLKLB	HpH 304cz	Zobrazit let
27.07.2020	Cicvarek	CIR	Zobrazit let
22.07.2020	TomasLKLB	HpH 304cz17	Zobrazit let
14.07.2020	TomasLKLB	HpH 304cz	Zobrazit let
13.07.2020	Cicvarek	CIR	Zobrazit let
13.07.2020	TomasLKLB	HpH 304cz	Zobrazit let
04.07.2020	TomasLKLB	VSO-10 GRADIENT	Zobrazit let
04.07.2020	Wolf	H304CZ	Zobrazit let
30.06.2020	TomasLKLB	VSO-10 GRADIENT	Zobrazit let
01.06.2020	TomasLKLB	HpH 304cz17	Zobrazit let
21.05.2020	TomasLKLB	HpH 304cz17	Zobrazit let

Obrázek 8 Výpis letů

Stránka s výpisem letů nám vypisuje všechny lety z roku, který si vybereme pomocí tlačítek. Lety jsou seřazené sestupně podle datumu.

Data se získávají pomocí „FlightLogControlleru“, který si z databáze vytáhne kolekci letů, podle zvoleného roku a poté je seřadí podle datumu.

## 6.2.4 Výpis letů pilota



Datum	Typ kluzáku	Registrace		
25.08.2020	Cirrus Std.	OK-7707	Zobrazit let	X
10.08.2020	HpH 304cz	OK-7304	Zobrazit let	X
09.08.2020	HpH 304cz	OK-7304	Zobrazit let	X
29.07.2020	HpH 304cz	OK-7304	Zobrazit let	X
22.07.2020	HpH 304cz17	OK-7304	Zobrazit let	X
14.07.2020	HpH 304cz	OK-7304	Zobrazit let	X
13.07.2020	HpH 304cz	OK-7304	Zobrazit let	X
04.07.2020	VSO-10 GRADIENT	OK-9608	Zobrazit let	X
30.06.2020	VSO-10 GRADIENT	OK-9608	Zobrazit let	X
01.06.2020	HpH 304cz17	OK-7304	Zobrazit let	X
21.05.2020	HpH 304cz17	OK-7304	Zobrazit let	X

Obrázek 9 Výpis letů pilota

Stránka výpis letů pilota nám zobrazuje lety zvoleného pilota ve výpisu pilotů. Lety si můžeme filtrovat podle roku, kdy byly letěny. Na levé straně od výpisu letů má pilot své osobní statistiky, jako je součet nalétaných hodin a součet nalétaných kilometrů. Když aplikace zjistí, že právě přihlášený pilot si prohlíží své lety, tak se mu zobrazí tlačítko pro odebrání letu z databáze.

Tato stránka získává data z „UserControlleru“, kterému se předávají parametry, jako je ID pilota a zvolený rok. Následně se lety seřadí podle datumu.



## 6.2.5 Výpis pilotů



Obrázek 10 Výpis pilotů

Stránka s výpisem pilotů nám ukazuje všechny zaregistrované piloty. Následně přes tlačítko nás stránka přesměruje na stránku s výpisem letů pilota.

Tyto data se získávají pomocí „UserControlleru“, který vybere všechny uživatele z databáze a seřadí je podle jména.



### 6.2.6 Výpis pořadí pilotů



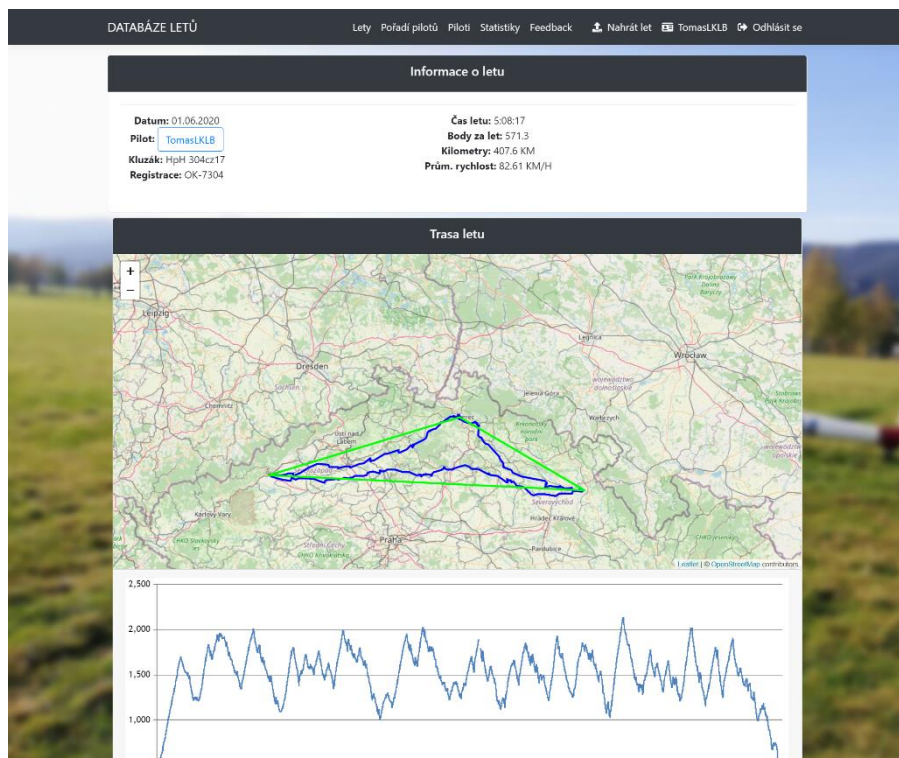
DATABÁZE LETŮ			Lety	Pořadí pilotů	Piloti	Statistiky	Registrovat se	Přihlásit se
#	Body	Jméno						
1	4225	Tomáš Kůhový						
2	2093	Wolfgang Sluka						
3	748	Martin Cicvárek						

Obrázek 11 Výpis pořadí pilotů

Stránka s výpisem pořadí pilotů nám ukazuje ten pomyslný žebříček pilotů. Který nám dává informace o tom, kolik má jaký pilot nalétaných bodů. Tyto body jsou za nejlepší tři lety za každý rok.

Tyto data jsou také z „UserControlleru“, který vybere z databáze všechny uživatele a seřadí je podle parametru „TopScore“ a pokud by se tyto body rovnaly tak ještě podle jména.

## 6.2.7 Zobrazení letu



Obrázek 12 Zobrazení letu

Tato stránka se zobrazením letu nám znázorňuje celý let a všechny další informace o letu. v horní kartě „Informace o letu“ vidíme základní informace jako je datum letu, jméno pilota, typ kluzáku a registrace. Dále pak čas letu, body za let, uletěné kilometry nebo průměrná rychlost úlohy. v kartě „Trasa letu“ můžeme vidět trasu celého letu (modrá čára) a nadeklarovaná úloha (zelená čára). Celá tato mapa je založená na leaflet. (11) Pod mapou se nachází barogram, který vykresluje výškový profil letu v metrech. Tento graf je vytvořený pomocí CanvasJS. (12) Tyto grafy jsou výhodné pro zobrazení mnoha údajů, jelikož dokáže data velmi rychle vykreslovat.

Data pro základní informace o letu stránka získává z „FlightLogControlleru“ a „AnalyseControlleru“. Předává se parametr ID letu a podle toho se v databázi vyhledá daný let. Dále pomocí „ViewControlleru“ po předání parametru ID letu se z databáze vytáhnou všechny záznamy o poloze, které se pak následně zpracují a zobrazí na mapě a barografu.

## 6.2.8 Celkové statistiky

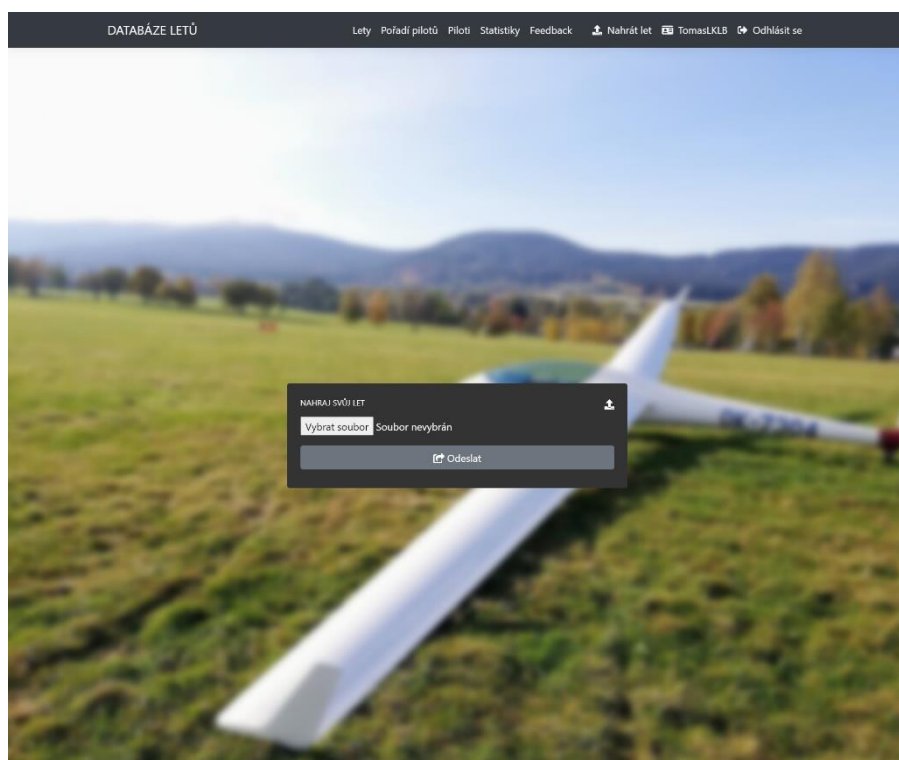


Obrázek 13 Statistiky

Stránka s celkovými statistikami slouží pro stručný přehled o náletech pilotů. v horní části jsou tři karty, kde je celkový počet letů, počet hodin a kilometrů. Pod těmito kartami se nachází grafy, kde jsou tyto statistiky znázorněny v koláčovém grafu. Tyto grafy jsou založeny na apex-charts. (13)

Stránka je obsluhována pomocí „StatisticsController“. Ten z tabulky „Stats“ získá celkové statistiky všech pilotů.

### 6.2.9 Nahrávání letu

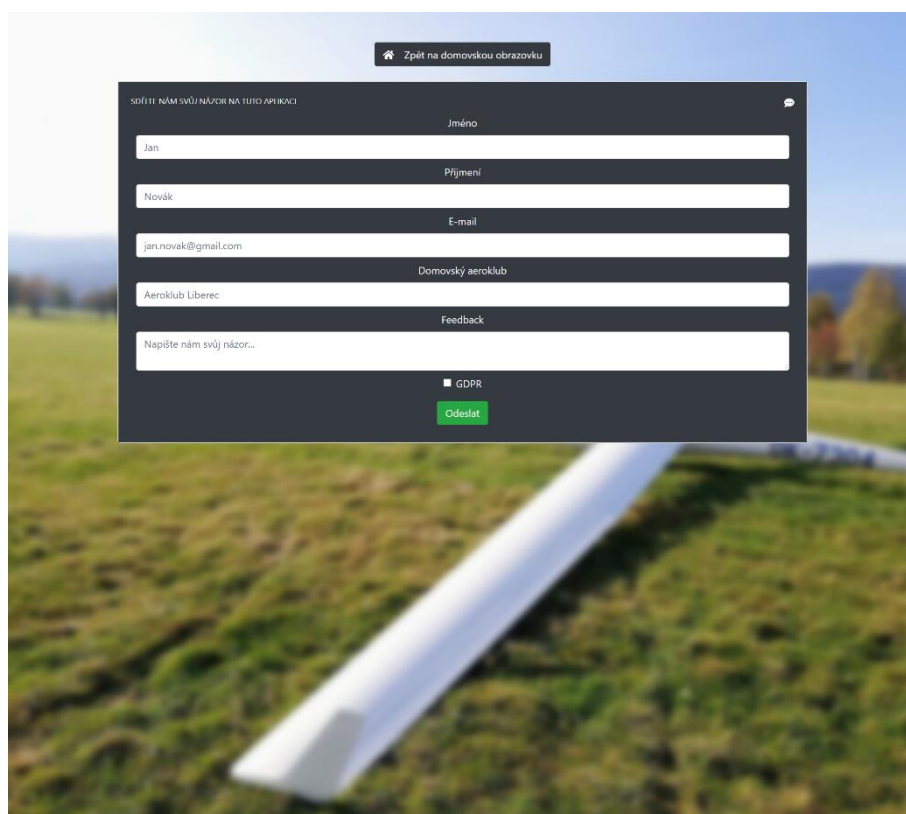


Obrázek 14 Nahrávání letu

Tato stránka slouží pro nahrávání letu do databáze. Přes tlačítko „Vybrat soubor“ uživatel vybere svůj let. Vybírání je ošetřené, že lze vybrat pouze soubor s typem IGC. Soubor, který se nahraje se následně zpracuje. Při zpracovávání se spočítá průměrná rychlost, uletěná vzdálenost a spočítají se body.

Celý let ze rozparsuje a pošle přes axios do „FlightLogControlleru“, který celý let deserializuje a uloží do databáze.

### 6.2.10 Feedback



The screenshot shows a feedback form titled "SDÍLEJTE NÁM SVŮJ NÁZOR NA TUTO APLIKACI" (Share your opinion on this application). The form is dark-themed and overlaid on a background image of a grassy field with a white pole. It contains the following fields and elements:

- A link at the top: "Zpět na domovskou obrazovku" (Back to home screen).
- Input fields for: "Jméno" (Name), "Příjmení" (Surname), "E-mail", "Domovský aeroklub" (Home aeroklub), and "Feedback" (with placeholder text "Napište nám svůj názor...").
- A checkbox labeled "GDPR".
- A green "Odeslat" (Send) button.

Obrázek 15 Feedback

Na této stránce jsou použity elementy z knihovny formik. Uživatel zde vyplní svoje jméno, příjmení, email, domovský aeroklub a svůj názor na tuto aplikaci. Je také nutné zaškrtnout checkbox se souhlasem o GDPR.

Zadané data se odešlou do „FeedbackController“, kde se zpracují a uloží do databáze. Tyto názory si pak já jako administrátor aplikace mohu prohlédnout na stránce „Feedback review“. k této stránce mám přístup pouze já.

### Závěr

Na začátku této maturitní práce jsem měl v plánu tuto aplikaci udělat v ASP.NET Razor Pages, jelikož s touto platformou jsem pracoval už pár let, ale časem jsem zjistil, že tato platforma dost omezuje programátora v jeho možnostech při vytváření webové aplikace. Proto jsem se rozhodl celou práci předělat na ASP.NET API jako backend a React jako frontend. Ze začátku to pro mě bylo náročné, jelikož při výuce jsme se React začali učit až na začátku čtvrtého ročníku. a tak jsem se musel zbytek věcí doučit sám nebo mi s mnoha problémy pomohl můj vedoucí práce, pan učitel Tomáš Kazda.

Během programování této aplikace jsem narazil na několik problémů. První problém byl v použití parseru na IGC soubor a následné ukládání dat do databáze. Jelikož v té době jsem ještě používal pouze ASP.NET Razor Pages, tak možnosti použití scriptu byly omezené. Proto jsem začal používat React. Po čase jsem zjistil, že React přináší mnohem více možností a po dohodě s vedoucím práce jsem předělal celou aplikaci. Po předělání aplikace jsem narážel na mnoho dalších, ale za to menších problémů, které jsem dokázal po nastudování dokumentace a hodinách zkoušení vyřešit.

Všechny stanovené cíle se mi podařilo úspěšně splnit. a myslím si, že i když jsem to na začátku celé maturitní práce nečekal tak se celá aplikace povedla a myslím, že má určitý potenciál. Jak už pouze jako databáze letů pro Aeroklub Liberec, tak třeba i pro celostátní používání. Odhadovaný celkový čas, který jsem strávil na této aplikaci je minimálně 300 hodin. Jelikož jsem spousta času strávil s řešením mnoha problémy při přechodu na React.

## Seznam obrázků

Obrázek 1 AAT (Assigned Area Task) .....	6
Obrázek 2 Rychlostní přelet.....	7
Obrázek 3 Regulární výraz na zjištění poloměrů.....	14
Obrázek 4 Návrh databáze.....	17
Obrázek 5 Hlavní stránka.....	19
Obrázek 6 Stránka pro registraci .....	20
Obrázek 7 Stránka pro přihlášení.....	20
Obrázek 8 Výpis letů .....	22
Obrázek 9 Výpis letů pilota.....	23
Obrázek 10 Výpis pilotů.....	24
Obrázek 11 Výpis pořadí pilotů .....	25
Obrázek 12 Zobrazení letu.....	26
Obrázek 13 Statistiky .....	27
Obrázek 14 Nahrávání letu .....	28
Obrázek 15 Feedback.....	29

## Použitá literatura

1. Microsoft. Dokumentace k ASP.NET Core. *Microsoft Docs*. [Online] Microsoft. [Citace: 15. 02 2021.] <https://docs.microsoft.com/cs-cz/aspnet/core/?view=aspnetcore-5.0>.
2. —. Dokumentace k EF Core. *Microsoft Docs*. [Online] Microsoft. [Citace: 16. 02 2021.] <https://docs.microsoft.com/cs-cz/ef/core/>.
3. Facebook. Getting started. *React Docs*. [Online] Facebook Inc. [Citace: 16. 02 2021.] <https://reactjs.org/docs/getting-started.html>.
4. React Hooks Reducer. [Online] [Citace: 04. 04 2021;.] <https://reactjs.org/docs/hooks-reference.html>.
5. Visual Studio 2019. [Online] Microsoft docs. [Citace: 19. 02 2021.] <https://visualstudio.microsoft.com/cs/vs/>.
6. Visual Studio Code. [Online] Microsoft docs. [Citace: 19. 02 2021.] [https://en.wikipedia.org/wiki/Visual\\_Studio\\_Code](https://en.wikipedia.org/wiki/Visual_Studio_Code).
7. MySQL. [Online] [Citace: 19. 02 2021.] <https://en.wikipedia.org/wiki/MySQL>.
8. FAI. [Online] [Citace: 19. 02 2021.] <https://www.fai.org/>.
9. Gliding IGC. [Online] [Citace: 19. 02 2021.] [http://www.gliding.ch/images/news/lx20/fichiers\\_igc.htm#Lrec](http://www.gliding.ch/images/news/lx20/fichiers_igc.htm#Lrec).
10. UTC time. [Online] [Citace: 19. 02 2021.] [https://en.wikipedia.org/wiki/Coordinated\\_Universal\\_Time](https://en.wikipedia.org/wiki/Coordinated_Universal_Time).
11. React Leaflet.js. [Online] [Citace: 10. 03 2021.] <https://react-leaflet.js.org/>.
12. CanvasJS. [Online] [Citace: 11. 03 2021.] <https://canvasjs.com/>.
13. Apex charts. [Online] [Citace: 11. 03 2021.] <https://apexcharts.com/>.



## A. Seznam příložených souborů

- 1) Složka s řešením Visual Studio 2019 (MP2021\_LKLB)
- 2) Složka Dokumentace
  - a. Písemná práce MP
    - i. MP2020-2021-Kulhavý-Tomáš-P4-Databaze-letu-pro-aeroklub-liberec.docx
    - ii. MP2020-2021-Kulhavý-Tomáš-P4-Databaze-letu-pro-aeroklub-liberec.pdf
  - b. Písemná práce SOČ
    - i. SOČ2020-2021-Kulhavý-Tomáš-P4-Databaze-letu-pro-aeroklub-liberec.docx
    - ii. SOČ2020-2021-Kulhavý-Tomáš-P4-Databaze-letu-pro-aeroklub-liberec.pdf
  - c. Přihláška k MP2021
- 3) Složka diagram databáze
  - a. Návrh databáze
- 4) Složka IGC
  - a. Příložené IGC soubory k testování a prohlédnutí struktury souboru
- 5) Složka SQL
  - a. Příložená databáze s nahranými lety
- 6) Složka návrh stránek
  - a. Příložené obrázky jednotlivých stránek

## **B. Další příloha**