

**Jihočeská univerzita v Českých Budějovicích**  
**Přírodovědecká fakulta**



**Aplikace Kancelář cyklistického  
závodu**

**Bakalářská práce**

**Jiří Míchal**

**Vedoucí práce: Ing. Václav Novák, Csc.**

**České Budějovice 2018**

---

Jihočeská univerzita v Českých Budějovicích  
Přírodovědecká fakulta

## ZADÁVACÍ PROTOKOL BAKALÁŘSKÉ PRÁCE

**Student: Jiří Michal**  
(jméno, příjmení, tituly)

**Obor – zaměření studia:** Měřicí a výpočetní technika

**Katedra/ústav, kde bude práce vypracována:** Ústav Fyziky a Biofyziky

**Školitel:** Ing. Václav Novák Csc., [vaenovak@prf.jcu.cz](mailto:vaenovak@prf.jcu.cz), tel: 602 409 272  
(jméno, příjmení, tituly, u externího š. název a adresa pracoviště, telefon, fax, e-mail)

**Garant z PFF:**  
(jméno, příjmení, tituly, katedra – jen v případě externího školitele)

**Školitel – specialista, konzultant:**  
(jméno, příjmení, tituly, u externího š. název a adresa pracoviště, telefon, fax, e-mail)

**Téma bakalářské práce: Aplikace Kancelář cyklistického závodu**

Cíle práce:

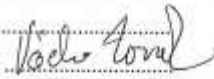
V posledních letech je, v reakci na rozšiřující se řady amatérských cyklistů, pořádáno velké množství malých cyklistických závodů, otevřených veřejnosti. Řada z nich si nemůže dovolit drahé služby profesionálních firem, zabývajících se měřením výsledků závodů, a potýká se tak s problémy spojenými s evidencí závodníků, časomírou a zpracováním výsledků.

Cílem práce studenta bude navrhnout a vytvořit aplikaci, která bude pořadatelům cyklistického závodu sloužit k následujícím činnostem:

- 1) Registraci, evidenci a správu účastníků cyklistického závodu (závodníků).
- 2) Konfiguraci kategorií závodníků, včetně parametrů a specifických způsobů, podle kterých jsou závodníci do kategorií zařazováni.
- 3) Evidenci a správu klubů, které do závodu vysílají své závodníky.
- 4) Měření času pro jednotlivé závodníky.
- 5) Vyhodnocení a prezentaci výsledků závodu.


Základní doporučená literatura: Portál Microsoft MSDN AA

---

Financování práce: .....  
Vedoucí práce: Ing. VÁCLAV NOVÁK CSc. .....podpis: 

U externích vedoucích fakultní garant práce: .....podpis: .....

Garant oboru bak. studia, pokud je obor zajišťován jinou katedrou/ústavem, než ze které je školitel (nepožaduje se u oboru biologie): .....podpis: .....

Vedoucí katedry/ústavu, kde bude práce vypracována: .....podpis: 

.....

Případný souhlas vedoucího ústavu AV: .....podpis: .....

.....

V Českých Budějovicích dne 8. 12. 2014 .....Podpis studenta: 

## Bibliografické údaje

Míchal J., 2018: Aplikace Kancelář cyklistického závodu.

[Application Office of cycling race. Bc.. Thesis, in Czech.] – 40 p.,

Faculty of Science, The University of South Bohemia, České Budějovice, Czech Republic.

## **Anotace**

Tématem práce je vytvoření aplikace pro měření a prezentaci výsledků cyklistických závodů. Práce je rozdělena na tři hlavní části. První část popisuje problematiku měření času včetně specifik časomíry cyklistického závodu. Druhá část obsahuje popis návrhu vyvíjené aplikace a popis využitých technologií. Poslední část se zabývá samotnou implementací aplikace.

Vytvořená aplikace, která je hlavním výstupem práce, je otestovaná a využitelná na závodech (nejen cyklistických), které splňují podmínky definované v textu.

## **Annotation**

The topic of this thesis is development application for time measuring and results presentation of cycling races. The thesis is divide into three main parts. The first part describes time measuring issue including timekeeping of cycling race. The second part describes the design of developed application and used technologies. The last part deals with implementation of application.

Developed application, which is the main outcome of the thesis, is tested and able to be use on races (not only on cycling races) which meet the defined conditions.



Prohlašuji, že svoji bakalářskou práci jsem vypracoval samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury.

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své bakalářské práce, a to v nezkrácené podobě elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách, a to se zachováním mého autorského práva k odevzdanému textu této kvalifikační práce. Souhlasím dále s tím, aby toutéž elektronickou cestou byly v souladu s uvedeným ustanovením zákona č. 111/1998 Sb. zveřejněny posudky školitele a oponentů práce i záznam o průběhu a výsledku obhajoby kvalifikační práce. Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

*Jindřichův Hradec, 10. 4. 2018*

*Podpis (bez něj je práce neplatná)*

### **Poděkování**

Rád bych poděkoval Ing. Václavu Novákovi, Csc. , za jeho vstřícný a trpělivý přístup při poskytování konzultací k této práci.

Velký dík patří mé ženě a dětem, za pochopení, trpělivost a podporu během studia.





# Obsah

Úvod a cíle práce.....	1
Úvod.....	1
Cíle práce.....	2
Analýza problematiky .....	3
Metody využívané organizátory.....	3
Dostupné metody a nástroje .....	4
Závěr analýzy .....	8
Požadavky a návrh řešení.....	9
Požadavky na aplikaci.....	9
Use Case diagram.....	11
Popis aktérů.....	11
Použité technologie .....	13
Implementace .....	13
Databázový model a jeho návrh.....	14
Objektově – relační model .....	17
Návrh objektového modelu.....	19
Návrh uživatelského rozhraní.....	21
HW rozhraní.....	26
Testování aplikace .....	30
Plánovaný vývoj.....	34
Závěr.....	36
Definice základních pojmů.....	37
Použité prameny a literatura.....	38
Seznam obrázků .....	39
Seznam tabulek .....	39
Seznam příloh.....	40



# Úvod a cíle práce

## Úvod

Na přelomu 70. a 80. let dvacátého století v Kalifornii (USA), položilo několik nadšených amatérských sportovců, stavbou speciálních, terénních kol, základy samostatného cyklistického odvětví, dnes obecně nazývané, stejně jako bicykly na kterých je provozováno, horská kola (mountainbiking, MTB).

V posledních dvou desetiletích se jízda na horských kolech, s dostupností těchto speciálů, stala masovou záležitostí. A přestože se i horská cyklistika profesionalizovala a je pořádána řada závodů pod záštitou národních nebo mezinárodních cyklistických svazů, stále největší podíl mezi cyklisty, věnujícími se MTB, mají bezpochyby amatéři bez platné závodní licence.

Tito cyklisté začali pro sebe pořádat nejprve malé závody v různých disciplínách tohoto odvětví. S růstem popularity horské cyklistiky počet takovýchto závodů rychle stoupl a v současné době je jich pořádáno v ČR několik desítek za sezónu. Jejich přesný počet není bohužel možné zjistit, protože neexistuje žádná jednotná evidence závodů. Za sezónu 2015 jsem jich ale napočítal v kalendáři závodů na serveru mtbs.cz [1], pouze v nejrozšířenější disciplíně – tzv. cyklomaratonu, 155.

Organizátoři těchto závodů začali, s nárůstem počtu závodníků, kteří se jejich závodů účastnili, řešit řadu problémů. Jedním z nich je i problém časomíry a rychlé publikace výsledků závodu. Existují sice produkty nebo služby, které měření časů na sportovních akcích dokáží zajistit, většina z nich je ale komerční a pro závody s malým rozpočtem tak finančně nedostupná.

Protože jsem sám amatérský cyklista a podílím se i na organizaci jednoho z těchto malých závodů, rozhodl jsem se vytvořit aplikaci, která organizátorům závodů MTB usnadní v tomto směru život.

## Cíle práce

Cílem práce je vytvořit aplikaci, která bude sloužit pořadatelům cyklistického závodu k následujícím činnostem:

- Registraci, evidenci a správě závodníků.
- Evidenci a správě klubů, které do závodu vysílají své závodníky.
- Měření času pro jednotlivé závodníky.
- Vyhodnocení a prezentaci výsledků závodu.

Z předpokladu, že aplikaci budou obsluhovat ve většině případů laici, vyplynula potřeba jednoduchého a intuitivního uživatelského rozhraní. Ze stejného důvodu jsem zvolil jako platformu pro běh aplikace operační systém MS Windows, který uživatelé důvěrně znají.

Aplikace je cílená na amatérské závody, které nejsou vázány přesnými pravidly národních cyklistických svazů, musí tedy umožňovat i variabilní konfiguraci kategorií závodníků.

Ze stejného důvodu bude možné zvolit i různé typy startu závodníků do závodu (hromadný start, postupný intervalový start...). Tato vlastnost také umožní využít aplikaci pro většinu disciplín horské cyklistiky, jako je downhill, MTB maraton nebo cross-country, případně i pro jiné sporty.

Během návrhu aplikace a později i během vlastní práce na aplikaci se objevilo ještě několik dalších požadavků, které by měla aplikace splňovat a o které jsem tedy původní zadání rozšířil. Jedná se především o tyto funkce:

- Možnost importovat do aplikace data o závodnících, např. z webového registračního systému.
- Možnost exportovat z aplikace výsledky, např. pro pozdější publikování na webových stránkách.
- Možnost využívat v aplikaci různé druhy hardwarových zařízení používaných v současné době pro časomíru, jako jsou např. externí zdroje přesného času, mechanické nebo světelné závory pro určení přesného času startu závodníka, nebo čtecí brány pro RFID čipy.

Aplikaci jsem nazval **RaceOffice**.

# Analýza problematiky

## Metody využívané organizátory

Ještě než jsem začal pracovat na samotné aplikaci, provedl jsem průzkum současného stavu měření času u několika vybraných závodů různých velikostí. Zjišťoval jsem, jaké používají pro měření času nástroje a jakou část z rozpočtu jejich závodu je spotřebována právě na časomíru. Vybíral jsem amatérské závody, u kterých byla buď tato informace veřejně dostupná, nebo u kterých jsem měl kontakt na organizátory. Cílem bylo zjistit, zda je vývoj aplikace vůbec potřebný a zda by o aplikaci vůbec byl mezi organizátory zájem. Výsledky tohoto průzkumu (platné k září 2015) jsou shrnuty v tabulce Tabulka 1

**Tabulka 1: Metody časomíry používané organizátory**

Jméno závodu	Počet závodníků	Nástroje časomíry	Část rozpočtu využitá na časomíru [%]
Jarošovská šlapka	290	MS Excel, stopky, cílová kamera	0%
Malíkov Open	123	MS Excel, stopky	0%
Střížovický upocenec	30	MS Excel, stopky	0%
MTB Stolové hory 2015	280	Stopky, papír	0%
MTB Stolové hory 2014	280	Profesionální firma, čipy	10%
Sázavský Blizák	300	Profesionální firma, čipy	10% - 15%
Březinské okruhy	50	Stopky, aplikaci pro zobrazení stopek	0%
MTB Sádecká pila	210	Stopky, cílová kamera	0%

## Dostupné metody a nástroje

V další fázi analýzy jsem zjišťoval situaci na trhu s nástroji pro časomíru. Tedy jestli vůbec existují aplikace využitelné pro měření času cyklistických závodů, jaké jsou jejich vlastnosti a zda jsou pro pořadatele malých cyklistických závodů cenově dostupné.

### Metody zajištění časomíry

Během zkoumání nástrojů časomíry na trhu jsem si uvědomil, že není možné mluvit pouze o dostupných aplikacích, jako o nástrojích pro časomíru, ale že se organizátorům závodů nabízí více způsobů, jak časomíru pro svůj závod zajistit. Obecně lze metody zajištění časomíry rozdělit podle následujících kritérií:

1. Typ aplikace a datové úložiště (online / off-line aplikace)

#### Online aplikace

Obvykle webová aplikace, často poskytující alespoň část funkcionality prostřednictvím aplikace pro tzv. smart zařízení - tedy tablety a telefony. Data závodníků i časomíry jsou uložena v databázi dostupné prostřednictvím veřejné sítě internet. Součástí aplikace bývá i on-line registrace závodníků. Nevýhodou je nutnost internetového připojení. Tento typ řešení bývá obvykle poskytován formou tzv. „software as a service“ – tedy dodavatel poskytne vlastní webový server pro běh aplikace a prezentaci výsledků, databázi, případně obsluhu.

#### Off-line aplikace

Klasická desktopová aplikace nevyžadující připojení k síti internet. Data jsou uložena lokálně, většinou v databázi. Nevýhodou oproti online aplikacím je fakt, že data nejsou okamžitě dostupná k prezentaci na internetu. Výhodou je naopak nezávislost na dostupnosti internetového připojení.

2. Cena a typ licence (komerční / zdarma)

#### Komerční aplikace

Využití aplikace je zpoplatněno, přičemž jsou využívány různé licenční modely (licence za software, licence na závodníka, časově omezená licence, atd.)

### **Aplikace zdarma**

Aplikaci je možné využívat zdarma pod různými typy licencí (Freeware, GNU GPL, atd.) Často je aplikace dodávána zdarma k zakoupenému hardwarovému řešení pro časomíru.

#### 3. Možnost využití hw zařízení pro časomíru

Aplikace dokáže spolupracovat s hardwarovými zařízeními pro časomíru, jako jsou RFID čipy, externí zdroje přesného času, startovací závory, atd. Při zakoupení hw řešení od renomovaných výrobců často bývá součástí i aplikace, která toto řešení umí využít.

#### 4. Možnost využití vlastní obsluhy

Profesionální firmy, zabývající se časomírou, využívají často vlastní řešení, které není možné získat samostatně. Celé řešení časomíry je pak dodáváno jako služba, včetně hardwarového vybavení, vyškolené obsluhy a dalších služeb, jako například registrace závodníků pomocí webové aplikace a publikace výsledků na vlastním serveru.

## **Dostupné nástroje**

V následující kapitole uvádím některé konkrétní produkty, které mě zaujaly, protože představují buď některou z hlavních metod časomíry, nebo naopak svou originalitou vybočují z metod, uváděných v kapitole Metody zajištění časomíry. Aplikace, které to umožňovaly, jsem otestoval, abych mohl posoudit jejich možné využití pro organizátory malých MTB závodů.

### **M-Sport PRO (VOLARACING)**

**Webové stránky výrobce:** <http://www.vola.fr/en/timing>

**Typ aplikace a datové úložiště:** *Off-line aplikace*



**Cena a typ licence:** *komerční (735 CHF - základní modul)*

**Možnost využití hw zařízení:** *ANO (vybrané druhy HW)*

**Možnost využití vlastní obsluhy:** *ANO*

Základní software je možné doplnit o další moduly, které rozšiřují jeho funkce o další možnosti. Např. možnost publikovat data jednoduše na webu, nebo data ukládat do databázového serveru dodavatele.

Software podporuje vybrané druhy HW, jako časové základny a stopky, nebo startovací závory.

Další podobné produkty:

- RaceDirector (<http://www.theracedirector.com/>)
- RaceTec (<http://www.racetiming.com/>)
- RunScore (<http://runscore.com/>)

## **WICLAX (Genialp)**

**Webové stránky výrobce:** *<https://www.wiclax.com>*

**Typ aplikace a datové úložiště:** *Off-line aplikace*

**Cena a typ licence:** *komerční ( 25€ / rok - základní verze, max. 300 závodníků)*

**Možnost využití hw zařízení:** *NE*

**Možnost využití vlastní obsluhy:** *ANO*

Software slouží pouze ke správě závodů, neobsahuje část pro časomíru. Časy jednotlivých závodníků dokáže systém získávat komunikací s nejvýznamnějšími systémy pro časomíru pomocí RFID čipů, jako jsou **RFID Timing, uRTime, Ipico, Bechronized, Chronolec, IDChronos** nebo **Race Result**.

Obsahuje ale velké množství pokročilých funkcí, jako např. podporu vícedenních závodů, závodů na počet kol, klasifikaci týmů, nebo servis aktuálních dat pro komentátora.

## **Webscorer**

<i>Webové stránky výrobce:</i>	<i><a href="https://www.webscorer.com">https://www.webscorer.com</a></i>
<i>Typ aplikace a datové úložiště:</i>	<i>Online aplikace</i>
<i>Cena a typ licence:</i>	<i>komerční ( 100\$ / rok - základní balíček)</i>
<i>Možnost využití hw zařízení:</i>	<i>ANO</i>
<i>Možnost využití vlastní obsluhy:</i>	<i>ANO</i>

Webscorer je množina aplikací a služeb od stejnojmenného vývojáře. Jde o aplikaci **Manual race timing** pro mobilní zařízení (OS Android, iOS) zajišťující měření a zaznamenávání časů, webovou aplikaci **Webscorer online registration** pro on-line registraci k závodu a API **Webscorer PRO embedded results subscription** pro online publikování výsledků na webu. Jednotlivé aplikace je možné získat v několika edicích s různou množinou funkcí.

## **fsTimer**

<i>Webové stránky výrobce:</i>	<i><a href="http://fstimer.org/index.htm">http://fstimer.org/index.htm</a></i>
<i>Typ aplikace a datové úložiště:</i>	<i>Off-line aplikace</i>
<i>Cena a typ licence:</i>	<i>aplikace zdarma ( GNU GPL)</i>
<i>Možnost využití hw zařízení:</i>	<i>NE</i>
<i>Možnost využití vlastní obsluhy:</i>	<i>ANO</i>

Open source, multiplatformní systém, naprogramovaný v programovacím jazyce **Python** s využitím **GTK for Python**, momentálně ve verzi 0.7. Autor deklaruje využití pro menší závody do cca 500 závodníků. Systém nabízí základní sadu funkcí pro správu závodu i časomíru, řada jich ale zatím chybí (např. více tratí v jednom závodu). Při testování se mi nepodařilo dosáhnout uspokojivé spolehlivosti.

## **Zajištění časomíry jako služba**

*Typ aplikace a datové úložiště:*      *Off-line aplikace*

*Cena a typ licence:*                      *komerční (služba)*

*Možnost využití hw zařízení:*        *ANO*

*Možnost využití vlastní obsluhy:*    *NE*

V tomto případě se vlastně nejedná o aplikaci, ale o profesionální službu, kterou firma nabízí na klíč. Používán je profesionální hardware (technologie pro identifikaci závodníků pomocí pasivních nebo aktivních čipů) i software. V základní ceně řešení bývá zajištění on-line registrace závodníků, registrace závodníků na místě startu, tisk a publikace výsledků, případně diplomů.

Obvykle je možné zajistit i další služby, jako měření mezičasů, zobrazování průběžných výsledků na velkoplošné tabuli, tisk startovních čísel, atd.

Ceny se pohybují od 10 000 Kč za základní řešení malého závodu do 100 závodníků (ceny jsem zjistil po oslovení několika firem, které tyto služby zajišťují)

Záměrně jsem neuvedl konkrétní software, protože v ČR existuje několik firem, které tyto služby dokáží zajistit, např. **Results.cz**, **SportSoft**, **Stopnito**, atd. Obvykle využívají software (a v některých případech i hardware) vyvinutý vlastními silami.

## **Závěr analýzy**

Z analýzy, kterou jsem provedl, vyplynulo, že řešení pro zajištění časomíry menšího závodu existuje celá řada. Přestože mají organizátoři na výběr z různých konceptů řešení, nevyužívají žádný. Důvodem je buď cenová nedostupnost, nebo absence vlastností, které jsou pro pořadatele důležité. Jednou z nich je i možnost využití hardwarových zařízení pro časomíru.

# Požadavky a návrh řešení

## Požadavky na aplikaci

Při návrhu vlastností aplikace jsem vycházel jednak ze zkušeností, které jsem získal při testování různých řešení časomíry, popsaných v kapitole Dostupné nástroje, jednak z vlastních zkušeností organizátora cyklistických závodů. Snažil jsem se navrhnout aplikaci tak, aby její používání nebylo příliš složité, zároveň ale tak, aby poskytovala všechny důležité funkce a přiblížila se tak co nejvíce profesionálnímu řešení.

## Funkční požadavky

Aplikace bude umožňovat:

- Založit a spravovat záznamy jednotlivých závodů.
- Pro každý závod konfigurovat parametr „Způsob startu“, který ovlivní chování aplikace při startování závodu. V tomto parametru bude možné vybrat jednu z následujících hodnot:
  - Hromadný start
  - Intervalový start
  - Start Gundersenovou metodou
  - Ruční start
- Pro každý závod vytvořit a spravovat libovolný počet tratí
- Pro každou trať konfigurovat parametr určující počet měřících bodů (mezičasů) včetně cíle.
- Pro každou trať vytvořit a spravovat libovolný počet kategorií
- Každé kategorii přidělit nepovinné parametry ročník narození a pohlaví, které budou sloužit jako selektory pro zařazování závodníků do jednotlivých kategorií.

- Založit a spravovat záznamy o klubech a členství závodníků v nich
- Založit a spravovat záznamy o závodnících a jejich příslušnosti k jednotlivým klubům
- Registrovat závodníky k jednotlivým závodům, zařazovat je v rámci závodu do kategorií a přiřazovat jim jednoznačný identifikátor (např. startovní číslo), platný pro celý závod
- Spustit časomíru pro jednotlivé závodníky, registrované v závodě, na základě způsobu určeným parametrem „Způsob startu“
- Při projetí závodníka měřícím bodem zaznamenávat hodnotu mezičasu a ke každému zaznamenanému mezičasu přiřadit konkrétního závodníka na základě startovního čísla.
- Spuštění časomíry, zaznamenání mezičasu a identifikaci závodníka na základě informací z externích zdrojů (například HW zařízení)
- V průběhu závodu, i po jeho dokončení, přidávat ručně další mezičasy, mazat je, upravovat jejich hodnoty a rušit vazbu závodníka na konkrétní mezičas.
- Během závodu zobrazovat průběžné výsledky pro diváky, např. pomocí dataprojektoru, nebo světelného panelu.
- Po dokončení závodu vytisknout výsledkovou listinu a umožnit export výsledků do souboru pro další strojové zpracování (např. publikování na webových stránkách)

## **Nefunkční požadavky**

### **Škálovatelnost**

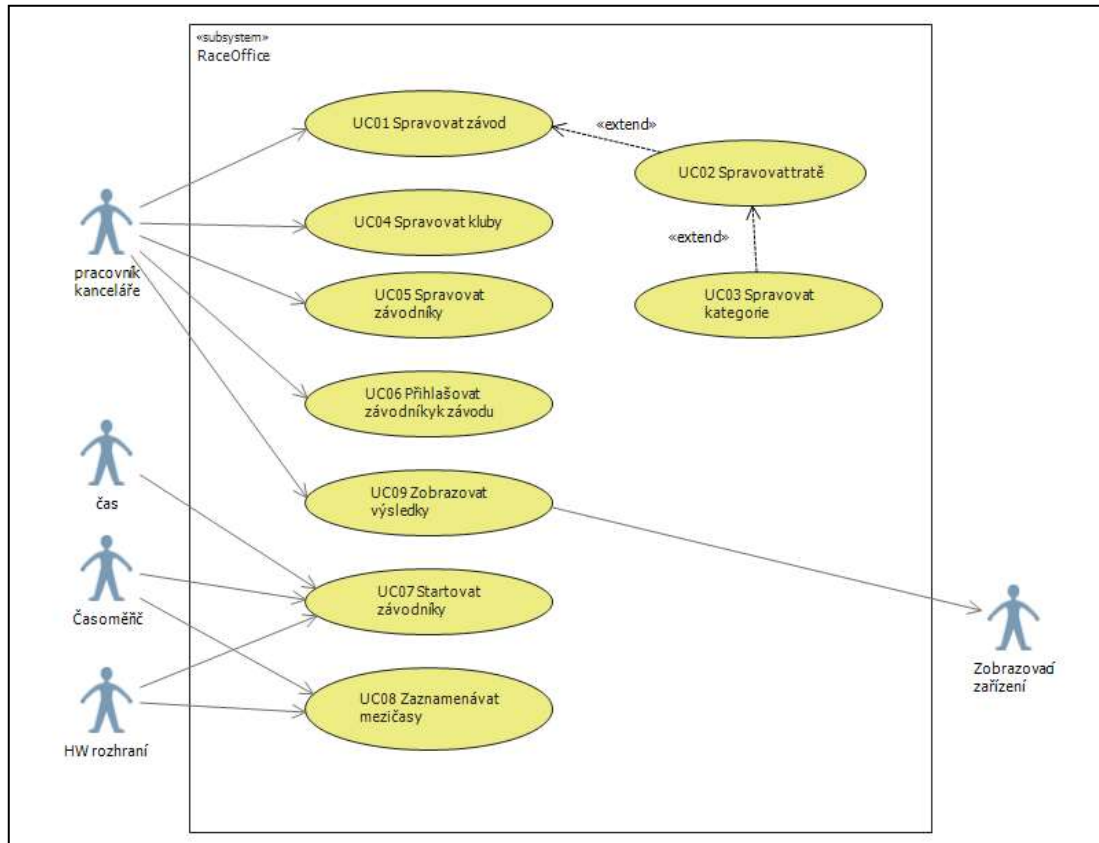
- Aplikace bude rozšiřitelná o další výstupní reporty.
- Aplikace musí umožnit současnou práci více uživatelů.

### **Rozšiřitelnost**

- Aplikace musí být v budoucnu jednoduše modifikovatelná, zejména

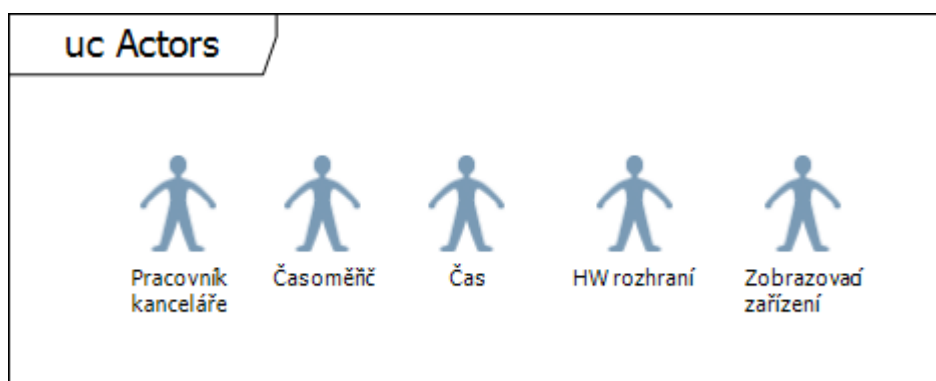
s ohledem na plánované rozšíření funkcionality.

## Use Case diagram



Obrázek 1 Use Case diagram aplikace

## Popis aktérů



Obrázek 2 Popis aktérů

## **Pracovník kanceláře**

Pracovník kanceláře má za úkol založení a správu závodu, založení a správu klubů, založení a správu závodníků a jejich přihlášení do závodu. Po dokončení závodu a uzavření časomíry tiskne výsledky, případně diplomy.

## **Čas**

Čas jako aktér startuje jednotlivé závodníky v případě automatického intervalového startu.

## **Časoměřič**

Časoměřič je zodpovědný za odstartování závodu (v případě hromadného startu), nebo jednotlivých závodníků (v případě intervalového startu, nebo startu Gundersenovou metodou). Dále má za úkol zaznamenávat časy průjezdu závodníků přes jednotlivé mezičasy, včetně cíle a identifikaci závodníků.

## **HW rozhraní**

Rozhraní pro připojení HW zařízení umožňuje integrovat do systému externí HW zařízení, které může zaznamenávat průjezd závodníků startovní nebo cílovou čarou, jejich identifikaci, případně do systému předávat časy průjezdů.

## **Zobrazovací zařízení**

HW zařízení jako např. projektor, monitor nebo světelná tabule, které umožňují prezentaci výsledků pro větší počet diváků.

# Použité technologie

Při výběru technologií, které pro vývoj aplikace využiji, jsem byl limitován pouze požadavky na aplikaci. Přestože jsem si byl v době návrhu aplikace vědom toho, že pro některé z technologií, které jsem se rozhodl použít, existují modernější alternativy, rozhodl jsem se pro následující technologie především z toho důvodu, že jsem s nimi měl již zkušenosti.



# Implementace

Implementační proces se skládal z několika postupných kroků, které popisují v této kapitole.

Nejprve byl zpracován návrh databázového modelu, který byl následně, pomocí nástrojů MS SQL, vytvořen. Z něj vychází objektově – relační model. Protože aplikace je poměrně rozsáhlá, rozhodl jsem se při jejím návrhu uplatnit principy architektury **MVC** (*Model-View-Controller*), což se následně promítlo i do návrhu diagramu tříd.

Návrh grafického rozhraní proběhl v počáteční fázi vývoje aplikace a v jeho průběhu pak byly realizovány pouze kosmetické úpravy.



## Databázový model a jeho návrh

Na základě požadavků na aplikaci jsem nejprve provedl identifikaci všech entit, které bude aplikace uchovávat v databázi, a vazeb mezi nimi.

### Identifikace entit

Tabulka 2 popisuje identifikované entity a přiřazuje jim již jména konkrétních tabulek v databázi. Pro jména tabulek, stejně jako pro jména dalších entit v aplikaci jsem použil anglické výrazy.

Tabulka 2: Identifikace entit

Název entity	Název tabulky	Popis entity
Závod	Race	Konkrétní závod
Trať	Track	Trať závodu
Kategorie	Category	Kategorie závodu
Závodník	Racer	Konkrétní závodník
Klub	Club	Konkrétní klub
Mezičas	Times	Zaznamenaný mezičas
Účast	Attendance	Představuje účast závodníka v závodě

### Identifikace vztahů

Tabulka 3 popisuje vztahy mezi entitami

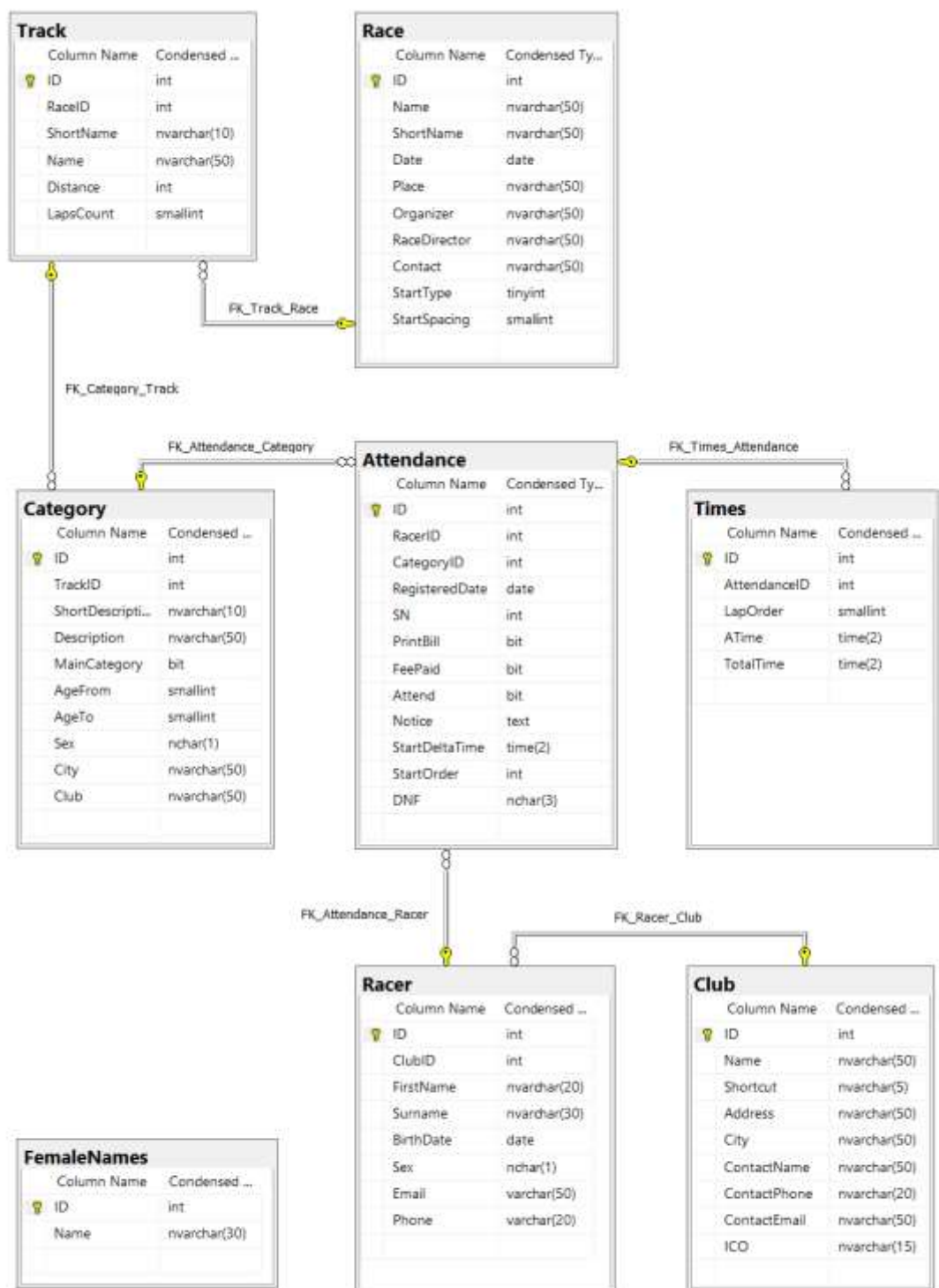
Tabulka 3: Identifikace vztahů

Entity	Vztah
Závod - Trať	1:N
Trať - Kategorie	1:N
Kategorie - Účast	1:N
Účast - Mezičas	1:N

Klub - Závodník	1:N
Závodník - Účast	1:N

### **Datový model**

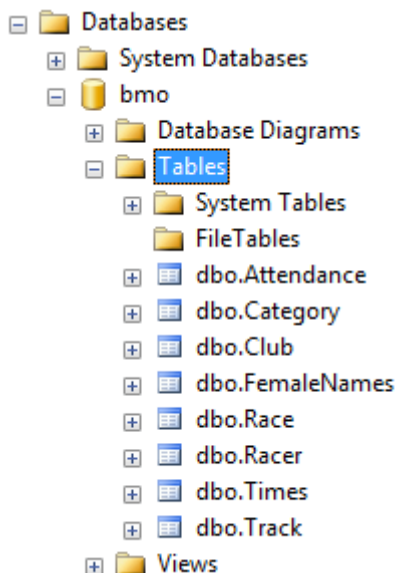
Na základě popsaných entit a jejich vztahů byl vypracován entitně-relační diagram, ve kterém jsou již doplněny atributy jednotlivých tabulek. Kromě popsaných entit obsahuje model ještě tabulku *FemaleNames*, která nemá žádný vztah ke zbytku modelu.



Obrázek 3: Datový model

## Vytvoření databáze

Databáze samotná byla vytvořena pomocí nástroje *MS SQL Management Studio*. Po vytvoření databáze jsem vygeneroval pomocí tohoto nástroje skript v jazyce SQL, který vytvoří strukturu databáze. Tento skript je uložen na přiloženém CD.

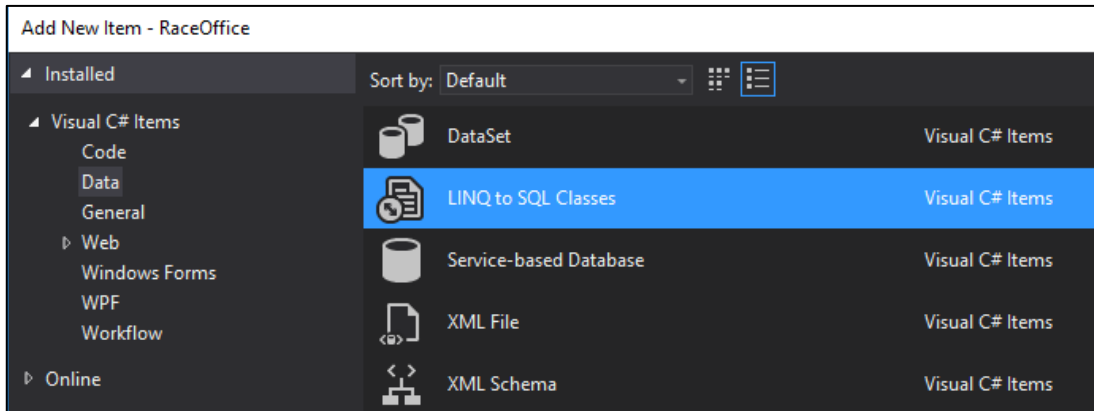


Obrázek 4: MS SQL Server Management Studio - seznam tabulek

## Objektově – relační model

V další fázi implementace jsem, již v prostředí *MS Visual Studio*, vytvořil objektově-relační model. Technika **ORM** (*Object-relational mapping, objektově-relační mapování*) zajišťuje automatické propojení dat mezi relační databází a objektovým modelem. Toto propojení mají na starost objektově relační frameworky. Pro aplikaci RaceOffice jsem zvolil Framework **LINQ to SQL**.

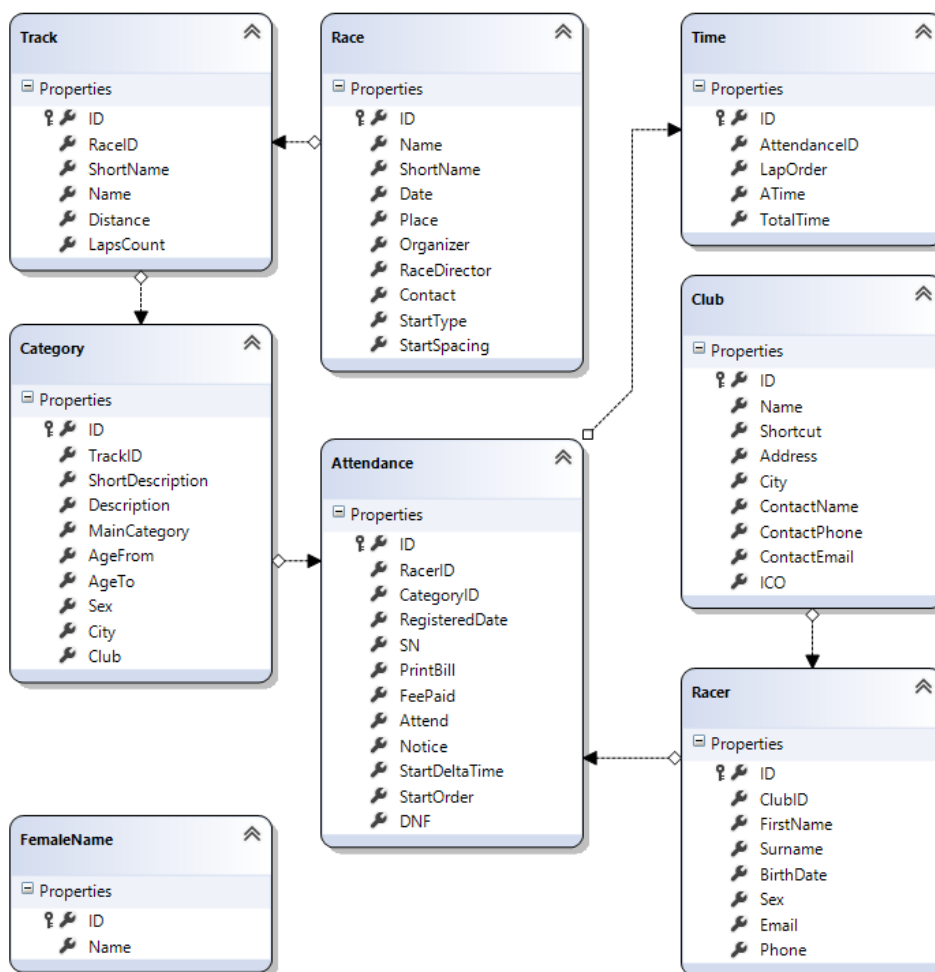
Do projektu typu Windows Forms Application, který jsem založil a pojmenoval RaceOffice, jsem přidal LINQ to SQL Classes (Obrázek 5), pojmenovaný **DataClasses**.



Obrázek 5: MS Visual Studio - přidání LINQ to SQL do projektu

Pomocí nástroje *Object Relational Designer* jsem nechal z tabulek databáze *bmo* vygenerovat třídy, které představují jednotlivé entity v databázi.

Obrázek 6 představuje konečný objektově – relační model



Obrázek 6: Objektově - relační model

## Návrh objektového modelu

V počátečních fázích vývoje aplikace jsem vytvořil jednoduchý objektový model, který do značné míry korespondoval s objektově – relačním modelem a tedy i s databázovým modelem. Jak ale aplikace rostla, přestával být kód přehledný. Kromě toho jsem věděl, že mi rozsah této práce nedovolí realizovat všechny nápady a v budoucnu budu ve vývoji aplikace pokračovat.

Abych tedy kód zpřehlednil, oddělil a usnadnil implementaci další funkcionality, rozhodl jsem se návrh přepracovat a využít principy architektury **MVC** (*Model – View – Controller*), respektive prezentačního vzoru **MVP** (*Model – View – Presenter*).

### Architektura MVC

Architektura MVC je založena na rozdělení aplikace na tři logické části (vrstvy) tak, aby aplikační logika byla oddělena od uživatelského rozhraní a jednotlivé části bylo možné měnit bez většího dopadu na části ostatní. Jednotlivé vrstvy bývají označovány jako Model, View a Controller. Odtud zkratka MVC.

Vrstva Model poskytuje a přijímá data, stará se o jejich načtení a uložení. Obsahuje také business logiku aplikace. Odkud data pocházejí, nebo jak budou zobrazena, neřeší.

Vrstva View má na starosti především zobrazení výstupu aplikace. O to, kde se data vzala, se opět nestará.

Vrstva Controller má na starosti řízení interakce mezi View a Modelem. Pokud se například změní data na vstupu, Controller se o tomto dozví a postará se, aby došlo ke změnám v Modelu, případně se postará o inicializaci změn ve View.

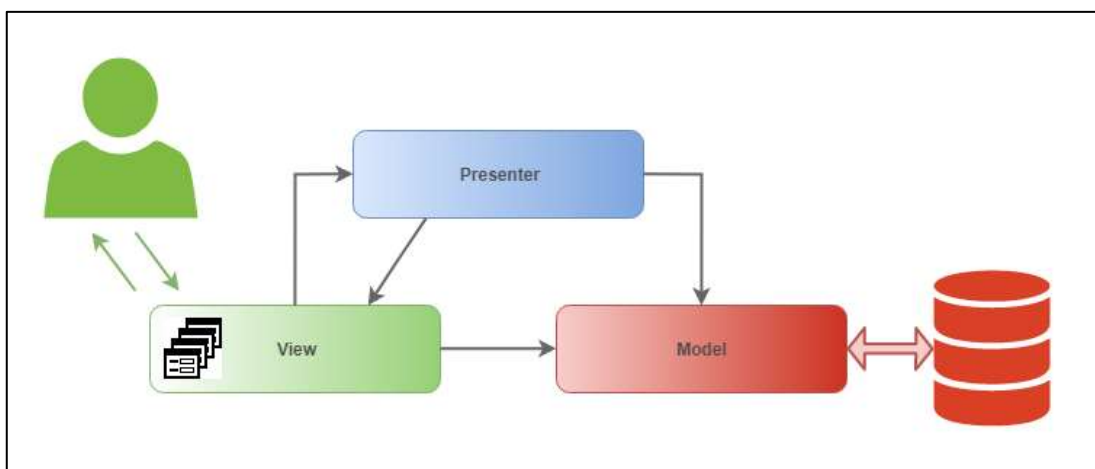
Prezentační vzor MVP, který jsem nakonec použil, je modifikací architektury MVC. Hodí se více pro platformy, které nemají oddělený vstup a výstup (jako mají klasické webové aplikace). Vrstva Controller je zde označována jako Presenter. Ten obsahuje aplikační i prezentační logiku a manipuluje s Modelem. Vrstva View se v tomto případě stará i o zpracování uživatelského vstupu (obvykle pouze tak, že v reakci na událost zavolá metodu Presenteru)

## Implementace MVP

Třídy vrstvy Model jsem už měl v podobě tříd, které automaticky vygeneroval *Object Relational Designer* frameworku LINQ to SQL. Využil jsem pouze toho, že tyto třídy je možné editovat a přesunul jsem do nich business logiku.

Vrstvu View reprezentují přímo třídy formulářů.

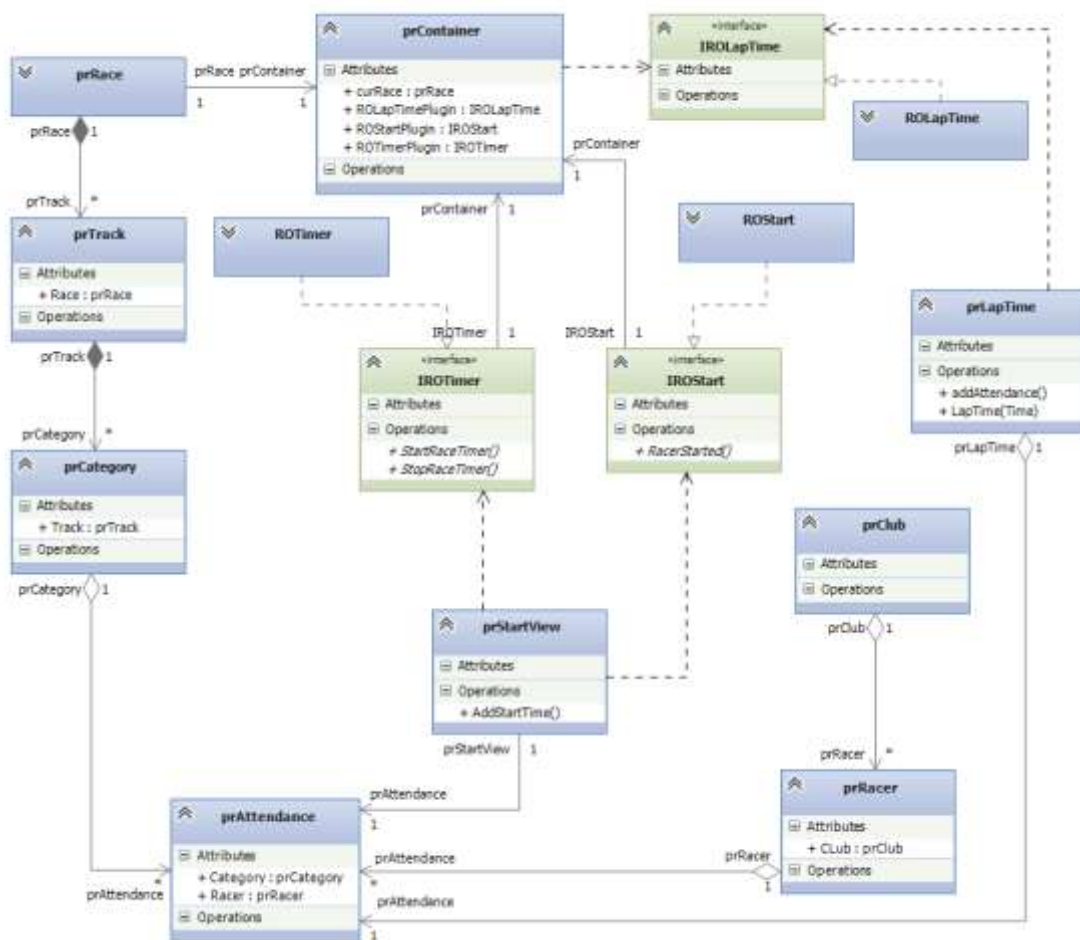
Třídy vrstvy Presenter jsou v aplikaci uloženy v souboru *presenter.cs* a až na výjimky existuje ke každé třídě z vrstvy View jedna třída z vrstvy Presenter, která obsahuje aplikační a prezentační logiku.



Obrázek 7: Blokové schéma aplikace

### Diagram tříd

V následujícím diagramu tříd (Obrázek 8) jsou, kvůli přehlednosti, zobrazeny pouze třídy z vrstvy Presenter s atributy a metodami, které jsou nejpodstatnější.



Obrázek 8: Diagram tříd

## Návrh uživatelského rozhraní

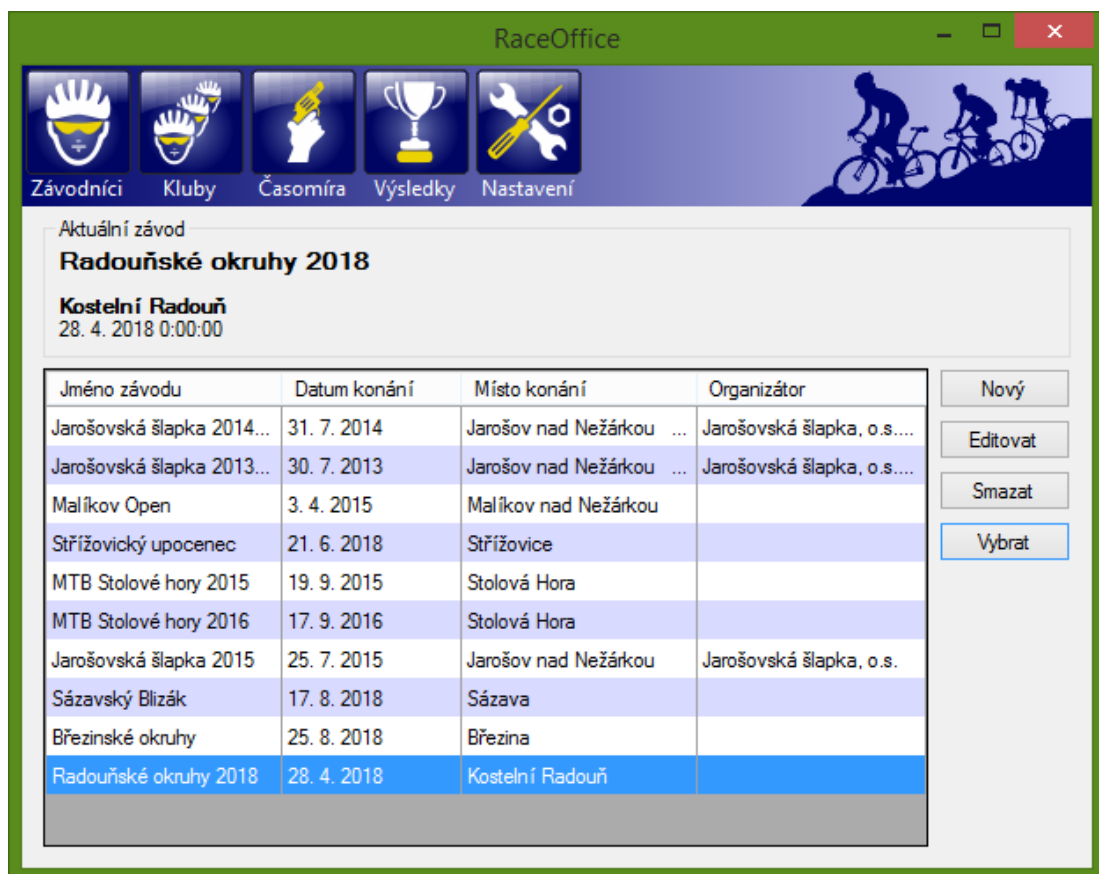
Pro vytvoření uživatelského rozhraní jsem se rozhodl použít knihovnu tříd **Windows Forms**, který je součástí frameworku .NET. V počáteční fázi jsem definoval jednotlivé formuláře, určil, jaké informace budou zobrazovat, a pomocí grafického nástroje MS Visual Studio formuláře navrhl.

### frmMain

Ústředním formulářem celé aplikace je formulář **frmMain**, který jednak obsahuje seznam spravovaných závodů a jednak hlavní menu. Pomocí tohoto menu se zobrazují ostatní formuláře – jako modální.

Tlačítka grafického menu jsem vytvořil v grafickém software pro práci s rastrovou grafikou, **Krita**.





Obrázek 9: Výchozí formulář aplikace

## Formuláře aplikace

Tabulka 4: Formuláře uživatelského rozhraní aplikace

Jméno formuláře	Popis
frmMain	Zobrazuje seznam spravovaných závodů a hlavní menu. Umožňuje spuštění většiny ostatních formulářů. Je výchozím formulářem aplikace.
frmRace	Zobrazuje vlastnosti závodu, je zobrazován při editaci a vytváření závodu z frmMain.
frmTrack	Zobrazuje vlastnosti tratě. Je zobrazován při editaci nebo vytváření tratě z frmRace.
frmCategory	Zobrazuje vlastnosti kategorie. Je zobrazován při editaci a

	vytváření kategorie z formuláře frmTrack.
frmRacers	Zobrazuje vlastnosti závodníků, umožňuje je editovat a vytvářet. Pokud je vybrán nějaký závod, zobrazuje také závodníky, kteří jsou k tomuto závodu registrováni a podrobnosti registrace. Umožňuje registrovat závodníky do závodu. Je zobrazován z frmMain
frmClub	Zobrazuje vlastnosti klubů, umožňuje je editovat a vytvářet. Je zobrazován při vytváření nového závodníka z frmRacers nebo z frmMain.
frmRaceView	Zobrazuje startovní pořadí uživatelů a jejich startovní časy, které umožňuje editovat. Umožňuje odstartovat závod a jednotlivé závodníky v závislosti na parametrech závodu. Zobrazuje zaznamenané mezičasy a umožňuje je zaznamenávat, mazat, editovat a přiřazovat k nim konkrétní uživatele. Je zobrazován z frmMain.
frmResults	Zobrazuje seznam grafických šablon a vybranou šablonu (např. výsledkovou nebo startovní listinu) zobrazí. Je zobrazován z frmMain a není modální.
frmConfig	Zobrazí parametry aplikace, které je možné konfigurovat. Především cestu a autentizační údaje k databázi, případně cestu k pluginům pro HW zařízení.

Kancelář závodu - závodníci

## Závod: Radouňské okruhy 2018

### Prohlížení závodníků a registrací

**Detail závodníka**

Jméno:  Příjmení:

Klub:

Narozen:  Pohlaví:

Email:  tel.:

**Závodník - vztah k závodu**

Trat':  Startovní číslo:

Kategorie:  Startovní čas:

Přítomen na startu:  Poznámka:

Hledat:

Příjmení	Jméno	Narozen
Froome	Chris	18. 4. 1988
Thomas	Geraint	16. 7. 1989

Hledat:

Příjmení	Jméno	Narozen	Start. číslo
Henao	Sergio	10. 2. 1990	0
Quintana	Nairo	11. 1. 1990	0
Valverde	Alejandro	10. 1. 1983	0
Porte	Richie	10. 10. 1982	0
Caruso	Damiano	28. 9. 1994	0
Sagan	Peter	13. 2. 1995	0
Majka	Rafa	30. 8. 1999	0
Sagan	Juraj	11. 4. 1998	0

Obrázek 10: Formulář frmRacers - správa závodníků

Časomíra

## Závod: Radouňské okruhy 2018

00:00:00

Start

Příjmení	Křestní jméno	Datum narození	Startovní číslo	Trat'	Kategorie	DNF
Henao	Sergio	10. 2. 1990	0	2 okruhy (50 km)	Muži (dlouhá)	
Quintana	Nairo	11. 1. 1990	0	2 okruhy (50 km)	Muži (dlouhá)	
Valverde	Alejandro	10. 1. 1983	0	2 okruhy (50 km)	Muži (dlouhá)	
Porte	Richie	10. 10. 1982	0	2 okruhy (50 km)	Muži (dlouhá)	
Caruso	Damiano	28. 9. 1994	0	2 okruhy (50 km)	Muži (dlouhá)	
Sagan	Peter	13. 2. 1995	0	2 okruhy (50 km)	Muži (dlouhá)	
Majka	Rafa	30. 8. 1999	0	2 okruhy (50 km)	Do 21 let (dlou...	
Sagan	Juraj	11. 4. 1998	0	2 okruhy (50 km)	Do 21 let (dlou...	

Obrázek 11: Formulář frmTimekeeping - start závodu

Časomíra

**Závod: Radouňské okruhy 2018**

**00:05:36**

Start Mezičasy

Čas	SN	Jméno	Příjmení	Narozen	Trat'	Kategorie	Číslo	Čas
00:00:37.65	3	Alejandro	Valverde	10. 1. 1983	2 okruhy (50 km)	Muži (dlouhá)	1	00:00:37.64
00:00:41.21	4	Richie	Porte	10. 10. 1982	2 okruhy (50 km)	Muži (dlouhá)	1	00:00:41.20
00:00:46.93	5	Damiano	Caruso	28. 9. 1994	2 okruhy (50 km)	Muži (dlouhá)	1	00:00:46.92
00:00:48.33	8	Juraj	Sagan	11. 4. 1998	2 okruhy (50 km)	Do 21 let (dlouhá)	1	00:00:48.32
00:00:50.18	2	Nairo	Quintana	11. 1. 1990	2 okruhy (50 km)	Muži (dlouhá)	1	00:00:50.18
00:00:50.38	3	Alejandro	Valverde	10. 1. 1983	2 okruhy (50 km)	Muži (dlouhá)	2	00:00:50.37
00:01:23.32	5	Damiano	Caruso	28. 9. 1994	2 okruhy (50 km)	Muži (dlouhá)	2	00:01:23.32
00:02:49.41	0							
00:03:24.38	4	Richie	Porte	10. 10. 1982	2 okruhy (50 km)	Muži (dlouhá)	2	00:03:24.38
00:03:25.02	1	Sergio	Henao	10. 2. 1990	2 okruhy (50 km)	Muži (dlouhá)	1	00:03:25.02
00:03:25.59	6	Peter	Sagan	13. 2. 1995	2 okruhy (50 km)	Muži (dlouhá)	1	00:03:25.58
00:03:26.13	7	Rafa	Majka	30. 8. 1999	2 okruhy (50 km)	Do 21 let (dlouhá)	1	00:03:26.13
00:03:26.89	8	Juraj	Sagan	11. 4. 1998	2 okruhy (50 km)	Do 21 let (dlouhá)	2	00:03:26.89
00:04:11.09	0							
00:04:27.87	0							

Zastavit časomíru Mezičas

Obrázek 12: Formulář frmTimekeeping - záznam mezičasů

## Formulář frmResults - ReportViewer

*Microsoft Report Viewer* je samostatná komponenta nástroje *SQL Server Reporting Services*.

*SQL Server 2008 Reporting Services (SSRS)* představuje serverovou platformu pro vytváření sestav, která poskytuje komplexní funkce k vytváření sestav u různých zdrojů dat. Pomocí Reporting Services je možné z relačních či multidimenzionálních zdrojů dat nebo zdrojů dat založených na standardu XML vytvořit interaktivní, tabulkové nebo grafické sestavy či sestavy s volným tvarem. [2]

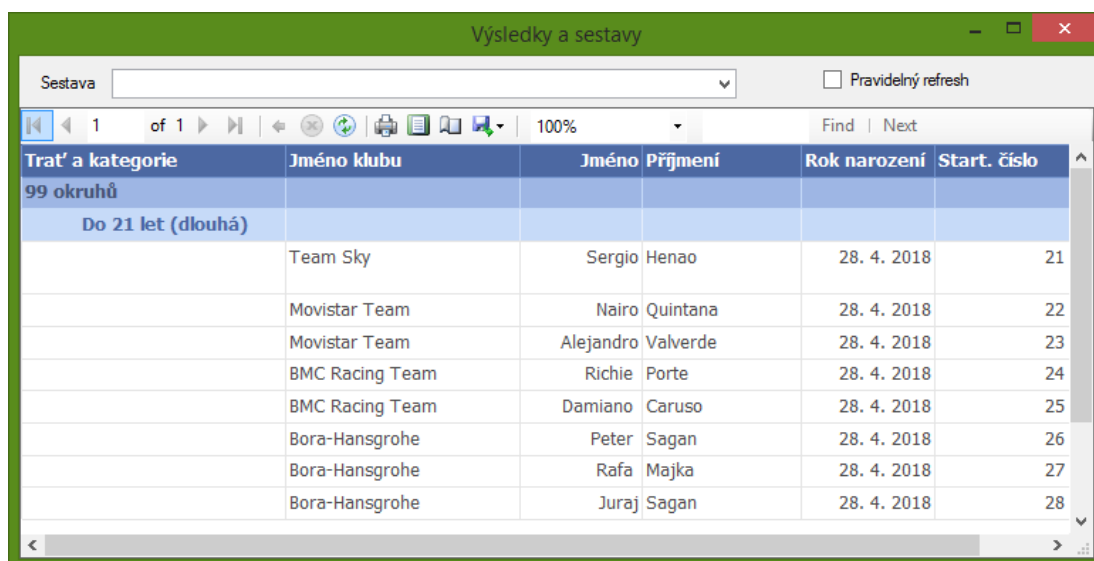
Kromě využití serverového řešení, obsahuje i část LocalReport. Pomocí Local Report je možné vyvářet reporty (tiskové sestavy) v .NET aplikacích bez nutnosti Report Serveru nebo dalších komponent. Report lze buď tisknout, zobrazovat do náhledu nebo exportovat do Wordu, Excelu a PDF. [3]

Prezentace výsledků v aplikaci je založena na technologii Report Viewer, protože tato technologie umožňuje jednoduše vytvářet a editovat šablony, do kterých se data

zobrazují. Tyto šablony je možné vytvářet a editovat jak v *MS Visual Studiu*, tak v aplikacích třetích stran, z nichž některé jsou šířené pod licenci Open Source. Protože se ale jedná o textový soubor (struktura xml), je možné ji editovat i ručně.

Možnost vytvářet a editovat vlastní šablony umožňuje uživateli generovat dokumenty s vlastním vizuálním stylem.

Data do šablony dodávají třídy **prTempStartList**, respektive **prTempResults** z vrstvy Presenter., komponenta ViewReport je součástí formuláře **frmReports**.



Trať a kategorie	Jméno klubu	Jméno	Příjmení	Rok narození	Start. číslo
99 okruhů					
Do 21 let (dlouhá)					
	Team Sky	Sergio	Henao	28. 4. 2018	21
	Movistar Team	Nairo	Quintana	28. 4. 2018	22
	Movistar Team	Alejandro	Valverde	28. 4. 2018	23
	BMC Racing Team	Richie	Porte	28. 4. 2018	24
	BMC Racing Team	Damiano	Caruso	28. 4. 2018	25
	Bora-Hansgrohe	Peter	Sagan	28. 4. 2018	26
	Bora-Hansgrohe	Rafa	Majka	28. 4. 2018	27
	Bora-Hansgrohe	Juraj	Sagan	28. 4. 2018	28

Obrázek 13: Formulář frmReports - Startovní listina

## HW rozhraní

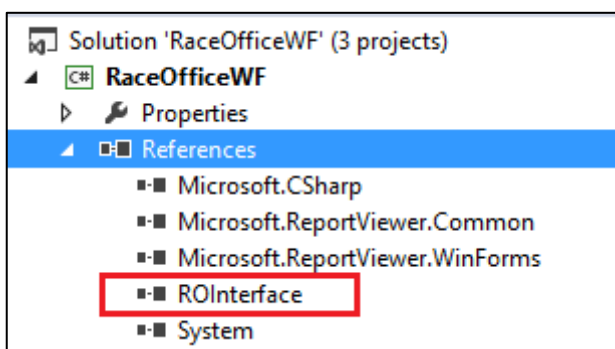
Jedním z požadavků na aplikaci, který vyplynul z provedené analýzy, byla možnost připojit k aplikaci hardwarové zařízení, které bude vykonávat některou z činností, které jinak provádí časoměřič.

Tento požadavek má tři hlavní opodstatnění:

1. OS Windows, pro které je aplikace napsána, není **RTOS** (*real-time operating system, operační systém reálného času*). Z této skutečnosti vyplývá, že mezičasy, naměřené pomocí systémového času, nemusí být zcela přesné. Ve všech sportech, pro které je aplikace uvažována, se sice časy měří s přesností maximálně na  $1 \times 10^{-2}$  s, ale měřící metoda by se mohla stát záminkou k podání protestu. Jediným způsobem, jak tento problém vyřešit, je použití nezávislého, externího časového zdroje.

2. U některých sportů, např. sjezdového lyžování, je vyžadována taková míra přesnosti, že ji není schopen zajistit časoměřič pouhou vizuální kontrolou přejetí cílové čáry a manuálním záznamem (stisknutím stopek, tlačítka...). Musí být tedy využito externí zařízení k záznamu této události – např. světelná závora.
3. S rostoucím počtem závodníků, přihlášených k závodu, přestává být v možnostech časoměřiče zaznamenat všechny události manuálně. Týká se to především hromadných závodů, kde do cíle dojíždí, nebo dobíhá velká skupina závodníků najednou. Časoměřič by měl zajistit zaznamenání mezičasu pro každého ze sportovců a ke každému zaznamenanému mezičasu přiřadit konkrétního závodníka. Od určitého počtu závodníků je nutné využít externího zařízení k identifikaci závodníků – např. cílová brána s čipy RFID.

Tato možnost je zajištěna pomocí rozhraní **IROTimer**, **IROStart** a **IROLapTime**. Tato rozhraní jsou publikována ve zvláštní knihovně tříd **ROInterface.dll**. Přidáním reference do vlastního projektu je pak možné tato rozhraní použít a vytvořit vlastní třídy, které tato rozhraní (nebo některé z nich) implementují.



Obrázek 14: Reference na knihovnu tříd ROInterface

Ve formuláři pro konfiguraci aplikace (frmConfiguration) je pak možné nastavit knihovnu tříd, která obsahuje třídy implementující popisovaná rozhraní. Tyto třídy jsou pak využívány.

Knihovna může obsahovat třídy implementující všechna tři rozhraní, nebo pouze některá. Pokud např. chceme připojit pouze startovací závora, vytvoříme třídu implementující rozhraní IROStart. Měření času a zaznamenávání mezičasů pak

zůstane v režii nativních tříd ROTimer a ROLapTime (kapitola Nativní třídy implementující rozhraní)

## Rozhraní IROTimer

```
public interface IROTimer
{
    /// <summary>
    /// Jméno pluginu, který interface implementuje
    /// </summary>
    string PluginName { get; }

    /// <summary>
    /// Aktuální čas závodu (Aktuální čas - čas 0)
    /// </summary>
    TimeSpan CurrentRaceTime { get; }

    /// <summary>
    /// Čas přečtený při posledním čtení atributu CurrentRaceTime
    /// </summary>
    TimeSpan LastReadTime { get; }

    /// <summary>
    /// Je volána při odstartování závodu (čas 0), spuštění časomíry
    /// </summary>
    void StartRaceTimer();

    /// <summary>
    /// Je volána při zastavení časomíry (manuální stop, odstartování
    posledního závodníka)
    /// </summary>
    void StopRaceTimer();

    /// <summary>
    /// Je volána bezprostředně po každé startovní události. Nastaví
    následující startovní čas
    /// </summary>
    /// <param name="NextTime">Následující startovní čas (rozdíl od času
    0)</param>
    /// <param name="Racers">Seznam závodníků, kteří v následující čas
    mají odstartovat</param>
    void SetNextRacerStartTime(TimeSpan NextTime, IList<string> Racers);

    /// <summary>
    /// Nastane, když je závodník odstartován (startovací závora,
    intervalový start)
    /// </summary>
    event EventHandler RacerStarting;
}
```

## Rozhraní IROStart

```
public interface IROStart
{
    /// <summary>
    /// Jméno pluginu, který interface implementuje
    /// </summary>
    string PluginName { get; }

    /// <summary>
    ///
    /// </summary>
    event EventHandler<ROEventArgs> RacerStarted;
}
```

## Rozhraní IROLapTime

```
public interface IROLapTime
{
    /// <summary>
    /// Jméno pluginu, který interface implementuje
    /// </summary>
    string PluginName { get; }

    /// <summary>
    /// Nastává při zaznamenání mezičasu
    /// </summary>
    event EventHandler<ROEventArgs> LapTimeDetected;
}
```

## Nativní třídy implementující rozhraní

Pokud není nakonfigurovaná žádná externí knihovna tříd s třídami implementujícími výše popsaná rozhraní, použijí se interní třídy **ROTimer**, **ROStart** a **ROLapTime**, které výše popsaná rozhraní také implementují.

Pokud taková knihovna nakonfigurovaná je, ale neobsahuje všechny tři třídy, použijí se pouze třídy, které obsahuje. Chybějící třídy jsou nahrazeny nativními třídami.

Nativní třídy využívají pro měření času systémový čas. Pro ostatní události, jako např. zaznamenání mezičasu, nebo manuální start závodníka, mají definované metody, které jsou volány obvykle po kliknutí na nějaký ovládací prvek uživatelského rozhraní. Startér tedy tyto činnosti provádí manuálně.



## Nativní třída ROLapTime

```
class ROLapTime: IROLapTime
{
    public string PluginName
    {
        get { return "Default plugin ROLapTime"; }
    }

    public event EventHandler<ROEventArgs> LapTimeDetected;

    public void ManualLapTime(int SN, TimeSpan time)
    {
        ROEventArgs eParams = new ROEventArgs();
        eParams.ID = SN;
        eParams.time = time;
        if (LapTimeDetected != null) LapTimeDetected(this, eParams);
    }
}
```

## Testování aplikace

Testování aplikace probíhalo ve třech hlavních fázích. V první fázi jsem testoval uživatelské rozhraní a ošetření neplatných dat na vstupech.

Pro druhou fázi jsem si připravil testovací plugin, který se do aplikace připojil přes HW rozhraní, simuloval připojené HW zařízení a testoval jednak vlastní rozhraní a jednak část pro časoměřiče, tedy různé typy startů a zaznamenávání mezičasů.

Ve třetí fázi proběhlo testování v ostrém provozu, na závodech na jejichž pořádání se podílím.

### Testovací plugin ROTestPlugin

Pro prvotní testování, především části pro časoměřiče a výše popsaného HW rozhraní, jsem připravil testovací plugin **ROTestPlugin**. Plugin obsahuje třídy **testROTimer** a **testROLapTime**, implementující rozhraní **IROTimer** a **IROLapTime**. Tyto třídy simulují chování HW zařízení, chovají se jako nezávislý zdroj reálného času (přestože využívají systémový čas) a generují události **RacerStarting** a **LapTimeDetected**.

Tabulka 5 obsahuje vzorek dat vygenerovaných testovacím pluginem (testy automatického záznamu mezičasů a přiřazení uživatelů)

Tabulka 5: Vzorek dat vygenerovaných pluginem ROTestPlugin

ID	AttendanceID	LapOrder	ATime	TotalTime
7575	3082	3	00:02:04.16	00:02:04.15
7576	3087	13	00:02:05.08	00:02:05.07
7577	3085	11	00:02:05.34	00:02:05.33
7578	3090	5	00:02:07.57	00:02:07.56
7579	3086	9	00:02:11.56	00:02:11.55
7580	3089	18	00:02:13.02	00:02:13.01
7581	3091	11	00:02:15.23	00:02:15.22
7582	3087	14	00:02:15.46	00:02:15.45
7583	3090	6	00:02:18.30	00:02:18.29
7584	3091	12	00:02:19.87	00:02:19.86
7585	3090	7	00:02:22.28	00:02:22.27
7586	3089	19	00:02:22.73	00:02:22.72
7587	3086	10	00:02:23.66	00:02:23.65
7588	3084	10	00:02:24.56	00:02:24.55
7589	3089	20	00:02:25.42	00:02:25.41
7590	3086	11	00:02:27.72	00:02:27.71
7591	3090	8	00:02:30.94	00:02:30.93
7592	3082	4	00:02:33.52	00:02:33.51
7593	3091	13	00:02:38.05	00:02:38.04
7594	3090	9	00:02:38.25	00:02:38.24
7595	3085	12	00:02:39.33	00:02:39.32
7596	3084	11	00:02:40.14	00:02:40.13
7597	3083	10	00:02:41.76	00:02:41.75
7598	3089	21	00:02:41.96	00:02:41.95
7599	3091	14	00:02:45.07	00:02:45.06
7600	3089	22	00:02:45.60	00:02:45.59
7601	3087	15	00:02:46.64	00:02:46.63
7602	3084	12	00:02:48.37	00:02:48.36
7603	3083	11	00:02:50.97	00:02:50.96
7604	3088	13	00:02:51.64	00:02:51.63
7605	3090	10	00:02:51.88	00:02:51.87
7606	3091	15	00:02:52.71	00:02:52.70
7607	3084	13	00:02:53.23	00:02:53.22
7608	3086	12	00:02:54.06	00:02:54.05
7609	3089	23	00:02:54.39	00:02:54.38
7610	3089	24	00:02:57.13	00:02:57.12
7611	3090	11	00:02:58.40	00:02:58.39
7612	3091	16	00:03:00.41	00:03:00.40
7613	3086	13	00:03:00.66	00:03:00.65
7614	3084	14	00:03:01.22	00:03:01.21
7615	3088	14	00:03:01.70	00:03:01.69
7616	3091	17	00:03:02.56	00:03:02.55
7617	3091	18	00:03:02.73	00:03:02.72
7618	3085	13	00:03:03.59	00:03:03.58
7619	3088	15	00:03:03.70	00:03:03.69
7620	3088	16	00:03:06.58	00:03:06.57
7621	3086	14	00:03:06.73	00:03:06.72
7622	3088	17	00:03:07.21	00:03:07.20
7623	3086	15	00:03:07.38	00:03:07.37

Čas závodu	SN	Jméno	Příjmení	Narozen	Trat'	Kategorie	Číslo	Čas
00:02:24.56	3	3	3	1. 1. 2000	99 ...	Do 21 let (dlouhá)	10	00:02:24.55
00:02:25.42	8	8	8	1. 1. 2000	99 ...	Do 21 let (dlouhá)	20	00:02:25.41
00:02:27.72	5	5	5	1. 1. 2000	99 ...	Do 21 let (dlouhá)	11	00:02:27.71
00:02:30.94	9	9	9	1. 1. 2000	99 ...	Do 21 let (dlouhá)	8	00:02:30.93
00:02:33.52	1	1	1	1. 1. 2000	99 ...	Do 21 let (dlouhá)	4	00:02:33.51
00:02:38.05	10	0	0	1. 1. 2000	99 ...	Do 21 let (dlouhá)	13	00:02:38.04
00:02:38.25	9	9	9	1. 1. 2000	99 ...	Do 21 let (dlouhá)	9	00:02:38.24
00:02:39.33	4	4	4	1. 1. 2000	99 ...	Do 21 let (dlouhá)	12	00:02:39.32
00:02:40.14	3	3	3	1. 1. 2000	99 ...	Do 21 let (dlouhá)	11	00:02:40.13
00:02:41.76	2	2	2	1. 1. 2000	99 ...	Do 21 let (dlouhá)	10	00:02:41.75
00:02:41.96	8	8	8	1. 1. 2000	99 ...	Do 21 let (dlouhá)	21	00:02:41.95
00:02:45.07	10	0	0	1. 1. 2000	99 ...	Do 21 let (dlouhá)	14	00:02:45.06
00:02:45.60	8	8	8	1. 1. 2000	99 ...	Do 21 let (dlouhá)	22	00:02:45.59
00:02:46.64	6	6	6	1. 1. 2000	99 ...	Do 21 let (dlouhá)	15	00:02:46.63
00:02:48.37	3	3	3	1. 1. 2000	99 ...	Do 21 let (dlouhá)	12	00:02:48.36
00:02:50.97	2	2	2	1. 1. 2000	99 ...	Do 21 let (dlouhá)	11	00:02:50.96
00:02:51.64	7	7	7	1. 1. 2000	99 ...	Do 21 let (dlouhá)	13	00:02:51.63
00:02:51.88	9	9	9	1. 1. 2000	99 ...	Do 21 let (dlouhá)	10	00:02:51.87
00:02:52.71	10	0	0	1. 1. 2000	99 ...	Do 21 let (dlouhá)	15	00:02:52.70
00:02:53.23	3	3	3	1. 1. 2000	99 ...	Do 21 let (dlouhá)	13	00:02:53.22
00:02:54.06	5	5	5	1. 1. 2000	99 ...	Do 21 let (dlouhá)	12	00:02:54.05
00:02:54.39	8	8	8	1. 1. 2000	99 ...	Do 21 let (dlouhá)	23	00:02:54.38
00:02:57.13	8	8	8	1. 1. 2000	99 ...	Do 21 let (dlouhá)	24	00:02:57.12
00:02:58.40	9	9	9	1. 1. 2000	99 ...	Do 21 let (dlouhá)	11	00:02:58.39
00:03:00.41	10	0	0	1. 1. 2000	99 ...	Do 21 let (dlouhá)	16	00:03:00.40
00:03:00.66	5	5	5	1. 1. 2000	99 ...	Do 21 let (dlouhá)	13	00:03:00.65
00:03:01.22	3	3	3	1. 1. 2000	99 ...	Do 21 let (dlouhá)	14	00:03:01.21
00:03:01.70	7	7	7	1. 1. 2000	99 ...	Do 21 let (dlouhá)	14	00:03:01.69
00:03:02.56	10	0	0	1. 1. 2000	99 ...	Do 21 let (dlouhá)	17	00:03:02.55
00:03:02.73	10	0	0	1. 1. 2000	99 ...	Do 21 let (dlouhá)	18	00:03:02.72
00:03:03.59	4	4	4	1. 1. 2000	99 ...	Do 21 let (dlouhá)	13	00:03:03.58
00:03:03.70	7	7	7	1. 1. 2000	99 ...	Do 21 let (dlouhá)	15	00:03:03.69
00:03:06.58	7	7	7	1. 1. 2000	99 ...	Do 21 let (dlouhá)	16	00:03:06.57
00:03:06.73	5	5	5	1. 1. 2000	99 ...	Do 21 let (dlouhá)	14	00:03:06.72
00:03:07.21	7	7	7	1. 1. 2000	99 ...	Do 21 let (dlouhá)	17	00:03:07.20
00:03:07.38	5	5	5	1. 1. 2000	99 ...	Do 21 let (dlouhá)	15	00:03:07.37

Obrázek 15: Testovací data zobrazená v rozhraní aplikace

## Testování v provozu

Pro testování v ostrém provozu jsem zvolil závod, na jehož organizaci se podílím a na kterém mám na starosti právě časomíru. Díky tomu jsem byl schopen postavit dvě pracoviště kanceláře závodu a dvě časoměřičská stanoviště. Prezentace závodníků i časomíra tak probíhala původním, zavedeným způsobem pomocí tabulek v MS Excel a stopek a paralelně ještě v aplikaci RaceOffice.

## Výsledky testů a provedené úpravy

Během testování v provozu se aplikace osvědčila. Výsledky, které byly pomocí aplikace získány, jsem nakonec mohl použít a byly k dispozici rychleji, než výsledky získané původním způsobem.

Několik problémů, které jsem zaznamenal, souviselo především s uživatelským rozhraním části obsluhované pracovníkem kanceláře (formulář frmRacers). Následuje jejich výčet a způsob řešení:

1. Pracovník kanceláře závodu je po dobu prezentace závodníků před startem maximálně vytížen. Musí do systému zadat závodníky, kteří ještě v databázi nejsou, ty, kteří tam jsou, musí vyhledat a zaregistrovat do závodu. Ukázalo se, že formulář je obtížně ovladatelný jen pomocí klávesnice.

**Řešení:** Pomocí atributu TabIndex u jednotlivých prvků formuláře upraveno pořadí získání fokusu

2. Ukázala se potřeba hledat v seznamech registrovaných a neregistrovaných závodníků – opět z důvodu rychlosti práce.

**Řešení:** Přidán TextBox **tbFilter**, který umožní vyhledat uživatele postupným vepisováním příjmení.

3. Několik problémů vzniklo zařazením závodnice ženy do mužské kategorie. Problém byl v tom, že ComboBox pro výběr pohlaví je defaultně nastaven na hodnotu „Muž“ (mužů se obvykle registruje k závodu více). V několika případech nebylo při zakládání závodnice pohlaví změněno, závodnice byla automaticky zařazena do mužské

kategorie, což ovlivnilo výsledky v ženské i mužské kategorii.

**Řešení:** Do databázového modelu byla přidána tabulka FemaleNames, která obsahuje pouze seznam ženských křestních jmen. Po opuštění TextBoxu **tbName** je kontrolováno, zda TextBox neobsahuje ženské jméno. Pokud ano, pohlaví se změní na hodnotu Žena. V případě, že je učiněn pokus o uložení uživatele s ženským jménem a pohlavím „Muž“, je zobrazeno upozornění. Pokud je učiněn pokus uložit závodníka s mužským jménem a pohlavím „žena“, je zobrazeno upozornění a nabídnuto uložení jména do databáze.

## Plánovaný vývoj

Přestože je nyní aplikace použitelná pro většinu menších cyklistických závodů, stále je možné její schopnosti rozvíjet. Již od začátku vývoje aplikace jsem věděl o některých funkcích, které by aplikace mít mohla, nebo dokonce měla, ale které nebudu schopen z důvodu rozsahu této práce realizovat. Snažil jsem se tedy aplikaci navrhout tak, aby byl její další vývoj co možná nejjednodušší.

Dále uvádím několik nových vlastností, které bych chtěl do dalších verzí implementovat.

- Podpora víceetapových závodů, nebo pohárů. V aplikaci by mělo být možné založit novou entitu Pohár, do kterého bude možné přidat jednotlivé závody. V rámci konfigurace Poháru bude možné nastavit způsob výpočtu konečného (a průběžného) pořadí – prostý součet časů, body přidělená na základě umístění v jednotlivých závodech, atd. Na základě této konfigurace pak bude aplikace schopna vyhodnocovat a prezentovat průběžné výsledky poháru.
- Podpora štafetových závodů – v aplikaci by mělo být možné založit novou entitu Štafeta, do které by bylo možné vybrat závodníky a Štafetu pak zaregistrovat do závodu.
- Podpora resources, překlad řetězců alespoň do angličtiny a možnost přepínat mezi jazykovými verzemi.
- Zajištění vysoké dostupnosti. V případě, že se databáze stane z nějakého

důvodu nedostupnou, měla by aplikace pokračovat v činnosti a data (především zachycené mezičasy) ukládat například do textového souboru. Jeho obsah by pak bylo možné do databáze naimportovat

- Možnost využívat off-line měřicí body. Tedy zřídit pracoviště časomíry pro zaznamenávání mezičasů i v místě, ze kterého není možné on-line přistupovat k databázi aplikace a zaznamenávat tak do ní data.
- Postupné rozšiřování podpory HW. Pro zařízení, u kterých do výrobce umožňuje, postupně dopisovat třídy pro připojení k aplikaci.
- Rozšíření možností aplikace tak, aby ji bylo možno využít i v dalších sportech, např. v orientačním běhu.

# Závěr

Cílem této práce bylo vytvořit aplikaci, která by sloužila pořadatelům cyklistických závodů pro většinu běžné agendy související s evidencí závodníků a měřeními a prezentací výsledků.

Motivací byla především skutečnost, že k dispozici nebylo řešení, které by splňovalo současně dvě důležitá kritéria – cenovou dostupnost a možnost používat hardwarová zařízení.

Z tohoto pohledu lze říci, že byly cíle práce splněny.

Při vývoji aplikace byl kladen důraz na to, aby mohly být snadno implementovány další funkce, které by rozšířily možnosti použití aplikace, nebo zvýšili uživatelský komfort.

V budoucnu bych také rád oslovil komunitu amatérských konstruktérů s žádostí o vývoj některých hardwarových zařízení, vhodných pro připojení k aplikaci, cenově dostupných i pro pořadatele amatérských závodů. Případně se pokusil o vývoj některých zařízení sám.

Vývoj aplikace, její návrh i testování mi přinesli mnoho cenných zkušeností. Jakkoliv jsem byl přesvědčen, že využiji především svých dosavadních znalostí, nakonec jsem musel prostudovat, prozkoumat a implementovat mnoho nových technologií.

## Definice základních pojmů

Hromadný start	Start závodu, při kterém vyrazí na start buď všichni, nebo větší skupina závodníků (např. všichni závodníci v jedné kategorii)
Intervalový start	Start závodu, při kterém vyráží na start závodníci postupně, v předem daném, konstantním časovém odstupu.
Start Gundersenovou metodou	Závodníci startují v pořadí daném výsledkovou listinou předchozího závodu nebo disciplíny s takovou časovou ztrátou na prvního s jakou došli v předchozím závodě respektive disciplíně.
MTB maraton	Závod horských kol, jehož délka a obtížnost je navržena tak, aby čas vítěze byl přibližně 2 hodiny. Trať je zpravidla realizována jako jeden okruh



# Použité prameny a literatura

- [1] CYCLING MEDIA S.R.O. Kalendář: mtbs.cz. *Mtbs.cz* [online]. 2015 [cit. 2015]. Dostupné z: <http://mtbs.cz/sekce/kalendar>
- [2] Služba SQL Server Reporting Services. *TechNet* [online]. Redmond: Microsoft, 2018 [cit. 2018-04-17]. Dostupné z: <https://technet.microsoft.com/cs-cz/library/ms159106%28v=sql.100%29.aspx?f=255&MSPPError=-2147217396>
- [3] HOLAN, Jan. Tvorba dokumentů pomoci reporting services local report. *Dotnetportal.cz* [online]. Praha: dotnetportal.cz, 2013 [cit. 2018-04-17]. Dostupné z: <https://www.dotnetportal.cz/blogy/15/Null-Reference-Exception/5359/Tvorba-dokumentu-pomoci-Reporting-Services-Local-Report>
- [4] Analyze and model your architecture. *Microsoft Developer Network* [online]. Redmond: Microsoft, 2015 [cit. 2018-04-17]. Dostupné z: <https://msdn.microsoft.com/en-us/library/57b85fsc.aspx?f=255&MSPPError=-2147217396>
- [5] C# Guide. *Microsoft Docs* [online]. Redmond: Microsoft, 2018 [cit. 2018-04-17]. Dostupné z: <https://docs.microsoft.com/en-us/dotnet/csharp/>
- [6] NAGEL, Christian, Bill EVJEN, Jay GLYNN, Morgan SKINNER a Karli WATSON. *C# 2008 Programujeme profesionálně*. Vydání první. Holandská 8, 63900 Brno: Computer Press, a.s., 2009. ISBN 978-80-251-2401-7.
- [7] UML. *ITnetwork.cz* [online]. Praha: David Čápka, 2013 [cit. 2018-04-17]. Dostupné z: <https://www.itnetwork.cz/navrh/architektury-a-dependency-injection>
- [8] *Development of Further Patterns of Enterprise Application Architecture* [online]. USA: Martin Fowler, 2000 [cit. 2018-04-17]. Dostupné z: <https://martinfowler.com/eaaDev/>

## Seznam obrázků

Obrázek 1 Use Case diagram aplikace.....	11
Obrázek 2 Popis aktérů .....	11
Obrázek 3: Datový model .....	16
Obrázek 4: MS SQL Server Management Studio - seznam tabulek .....	17
Obrázek 5: MS Visual Studio - přidání LINQ to SQL do projektu.....	18
Obrázek 6: Objektově - relační model .....	18
Obrázek 7: Blokové schéma aplikace .....	20
Obrázek 8: Diagram tříd.....	21
Obrázek 9: Výchozí formulář aplikace .....	22
Obrázek 10: Formulář frmRacers - správa závodníků .....	24
Obrázek 11: Formulář frmTimekeeping - start závodu.....	24
Obrázek 12: Formulář frmTimekeeping - záznam mezičasů .....	25
Obrázek 13: Formulář frmReports - Startovní listina .....	26
Obrázek 14: Reference na knihovnu tříd ROInterface.....	27
Obrázek 15: Testovací data zobrazená v rozhraní aplikace .....	32

## Seznam tabulek

Tabulka 1: Metody časomíry používané organizátory .....	3
Tabulka 2: Identifikace entit.....	14
Tabulka 3: Identifikace vztahů .....	14
Tabulka 4: Formuláře uživatelského rozhraní aplikace.....	22
Tabulka 5: Vzorek dat vygenerovaných pluginem ROTestPlugin .....	31

# Seznam příloh

Kancelar\_cyklistickeho\_zavodu.pdf – text bakalářské práce ve formátu pdf.

## **Aplikace**

Obsahuje zkompilovanou aplikaci

## **Dokumentace**

Obsahuje uživatelskou a instalační příručku v elektronické podobě

## **Zdrojové kódy**

Zdrojové kódy aplikace