



Semestrální projekt: Databázové a informační systémy

## **Informační systém pro sezónu Formule 1**

Vysoká škola Báňská – Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky  
Katedra informatiky

## Obsah

<b>1. Specifikace zadání .....</b>	<b>3</b>
<b>2. Datový model .....</b>	<b>4</b>
2.1. ER diagram .....	4
2.2. Lineární zápis typů entit .....	4
2.3. Datový slovník .....	5
2.4. Integritní omezení .....	7
<b>3. Stavová analýza .....</b>	<b>7</b>
<b>4. Funkční analýza .....</b>	<b>7</b>
4.1. Seznam funkcí .....	7
4.2. Detailní popis funkcí .....	10
<b>5. Návrh uživatelského rozhraní .....</b>	<b>13</b>
5.1. Menu .....	13
5.2. Návrh formulářů .....	15

## 1. Specifikace zadání

### 1.1. Proč

Je potřeba zjednodušený informační systém pro evidenci jedné sezóny Formule 1.

### 1.2. K čemu

Týmy mají pod kontrolou, které motory jsou ve Formulích svých jezdců namontovány. Jsou odpovědní za správnost informací, které se týkají jejich týmu.

FIA kontroluje, zda všechny strany, které jsou v kontaktu s IS vkládají nebo udržují aktuální a pravdivé informace.

Organizátor je povinen po ukončení GP vložit výsledky svého závodu pro evidenci výsledných tabulek

Při přístupu do veřejné části si každý návštěvník může zobrazit informace, které jsou v databázi

### 1.3. Kdo

**Administrátor** – správce celé databáze, na začátku sezóny buď vymaže databázi, nebo vytvoří novou, může informace aktualizovat, vkládat a číst

**Týmy** – Každý tým může aktualizovat data o sobě (týmu) a o svých jezdcích

**Organizátor GP** (GrandPrix – Velká cena) – Organizátor jednotlivých GP může upravit/vložit výsledky své GP

**FIA** (Mezinárodní automobilová federace) – jako organizátor může všechny data kontrolovat a upravovat.

**Návštěvník** – návštěvník může data vidět, ale nesmí je upravovat

### 1.4. Vstupy

U týmů evidujeme jako název, sídlo, dodavatele pneumatik, dodavatele motorů.

U jezdců budeme evidovat název jméno, příjmení, startovní číslo, národnost, datum narození.

Všichni jezdci budou mít na výběr několik motorů, ale vždy budou mít jen jeden namontovaný (přidělený)

U GP evidujeme datum, délku jednoho okruhu, počet kol v závodě.

U každé GP je uvedeno se budou evidovat výsledky, tedy pořadí v závodě, jezdec, ztrátu na vítěze a informaci, zda dojel do cíle.

### 1.5. Výstupy

Seznam jezdců

Seznam týmů

Seznam GP a okruhů

Výsledky jednotlivých GP

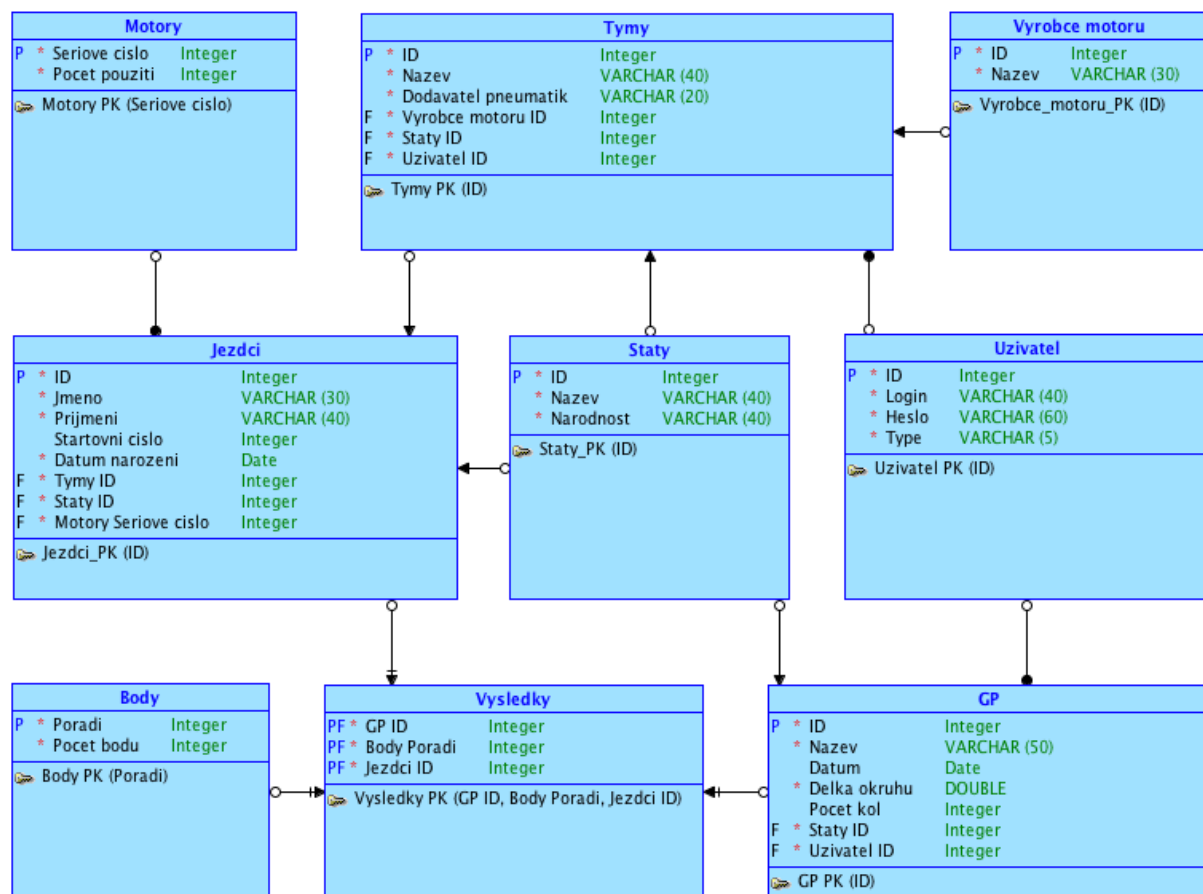
Pořadí jezdců v sezóně a poháru konstruktérů

### 1.6. Funkce

Je potřeba vést evidenci velkých cen, umístění jak jezdců, tak i Poháru konstruktérů (pořadí týmů), ale i využití komponent jednotlivých jezdců (v našem případě stačí vést evidenci motorů). Po IS je vyžadována běžná práce s daty (přidání, úprava) a zobrazení několik statistik.

## 2. Datový model

### 2.1. ER diagram



### 2.2. Lineární zápis typů entit

Legenda: **Primární klíč**, *cizí klíč*, atribut

Motory(**Seriove\_cislo**, Pocet\_pouziti)

Vyrobc\_motoru(**ID**, Nazev)

Staty(**ID**, Nazev, Narodnost)

Tymy(**ID**, Nazev, Dodavatel\_pneumatik, Vyrobc\_motoru\_ID, Staty\_ID, Uzivatel\_ID)

Jezdci(**ID**, Jmeno, Prijmeni, Startovni\_cislo, Datum\_narozeni, Tymy\_ID, Staty\_ID, Motor\_Seriove\_cislo)

Body(**Poradi**, Pocet\_bodu)

GP(**ID**, Nazev, Datum, Delka\_okruhu, Pocet\_kol, Staty\_ID, Uzivatel\_ID)

Vysledky(GP\_ID, Body\_Poradi, Jezdci\_ID)

Uzivatel(**ID**, Login, Heslo, Typ)

## 2.3. Datový slovník

### Motory

Název	Typ	Velikost	Klíč	Null	Index	Popis
Serieove_cislo	Int	10	PK	Ne	Ano	Sériové číslo motoru
Pocet_pouziti	Int	10	Ne	Ne	NE	Počet použití motoru

### Vyrobce\_motoru

Název	Typ	Velikost	Klíč	Null	Index	Popis
ID	Int	10	PK	Ne	Ano	Identifikace výrobce
Nazev	Varchar	30	Ne	Ne	Ne	Název společnosti

### Staty

Název	Typ	Velikost	Klíč	Null	Index	Popis
ID	Int	10	PK	Ne	Ano	Identifikace státu
Nazev	Varchar	40	Ne	Ne	Ne	Název státu
Narodnost	Varchar	40	Ne	Ne	Ne	Národnost státu

### Tymy

Název	Typ	Velikost	Klíč	Null	Index	Popis
ID	Int	10	PK	Ne	Ano	Identifikace týmu
Nazev	Varchar	40	Ne	Ne	Ne	Název týmu
Dodavatel_pneumatik	Varchar	20	Ne	Ne	Ne	Dodavatel pneu.
Vyrobce_motoru_ID	Int	10	FK	Ne	Ano	Identifikace výrobce
Staty_ID	Int	10	FK	Ne	Ano	Identifikace státu
Uzivatel_ID	Int	10	FK	Ne	Ano	Ident. Uživatele

### Uzivatel

Název	Typ	Velikost	Klíč	Null	Index	Popis
ID	Int	10	PFK	Ne	Ano	Identifikace uživatele
Login	Varchar	40	Ne	Ne	Ne	Login uživatele
Heslo	Varchar	60	Ne	Ne	Ne	Heslo uživatele
Type	Varchar	5	Ne	Ne	Ne	Identifikace uživatele

## Jezdci

Název	Typ	Velikost	Klíč	Null	Index	Popis
ID	Int	10	PK	Ne	Ano	Identifikace jezdce
Jmeno	Varchar	30	Ne	Ne	Ne	Křestní jméno jezdce
Prijmeni	Varchar	40	Ne	Ne	Ne	Příjmení jezdce
Startovni_cislo	Int	10	Ne	Ano	Ne	Startovní číslo jezdce
Datum_narozeni	Date		Ne	Ne	Ne	Datum narození
Tymy_ID	Int	10	FK	Ne	Ano	Identifikace týmu
Staty_ID	Int	10	FK	Ne	Ano	Identifikace státu
Motory_seriove_cislo	Int	10	FK	Ne	Ano	Identifikace motoru

## GP

Název	Typ	Velikost	Klíč	Null	Index	Popis
ID	Int	10	PFK	Ne	Ano	Identifikace GP
Nazev	Varchar	50	Ne	Ano	Ne	Název velké ceny
Datum	Date		Ne	Ne	Ne	Datum GP
Delka_okruhu	Int	10	Ne	Ne	Ne	Délka 1 okruhu
Pocet_kol	Int	10	Ne	Ano	Ne	Počet kol v GP
Staty_ID	Int	10	FK	Ne	Ano	Identifikace státu
Uzivatel_ID	Int	10	FK	Ne	Ano	Ident. uživatele

## Body

Název	Typ	Velikost	Klíč	Null	Index	Popis
Poradi	Int	10	PK	Ne	Ano	Pořadí
Pocet_bodu	Int	10	Ne	Ne	Ne	Počet bodů za umístění

## Vysledky

Název	Typ	Velikost	Klíč	Null	Index	Popis
GP_ID	Int	10	PFK	Ne	Ano	Identifikace GP
Body_Poradi	Int	10	PFK	Ne	Ano	Počet bodů
Jezdci_ID	Int	10	PFK	Ne	Ano	Identifikace jezdce

## 2.4. Integritní omezení

1. Uživatel.Type musí mít hodnotu: `admin`, `FIA`, `GP` nebo `team`.
2. Uživatel.ID nesmí být použit v attributech Tmy.ID a GP.ID zároveň

## 3. Stavová analýza

### Stavy GP

- **Neodjetá** - Datum GP je ještě v budoucnosti (GP.Datum > aktuální datum)
- **Probíhající** – Datum GP je dnes (GP.Datum = aktuální datum)
- **Odjetá** – Datum GP je již v minulosti (GP.Datum < aktuální datum)

## 4. Funkční analýza

### 4.1. Seznam funkcí

#### 1. Evidence uživatelů

*Tabulka:* Uživatel, GP, Tmy

*Role:* Admin

##### 1.1. Vložení uživatele

##### 1.2. Úprava uživatele

Každý tým může upravit pouze heslo pouze sobě

*Role:* Uživatel pouze svůj záznam

##### 1.3. Smazání uživatele

U propojených atributů v tabulkách GP nebo Tmy se smažou příslušné záznamy

##### 1.4. Seznam uživatelů

##### 1.5. Detail uživatele

#### 2. Evidence GP

*Tabulka:* GP

*Role:* FIA, Admin

##### 2.1. Vložení GP

##### 2.2. Úprava GP

*Role navíc:* Uživatel GP pro svůj záznam

##### 2.3. Smazání GP

Po smazání záznamu se smažou i odpovídající záznamy v tabulce `Vysledky`

##### 2.4. Seznam GP

S možností filtrace nebo třídění atributů – Délka, počet kol, datum

*Role:* všichni včetně veřejnosti

##### 2.5. Detail GP

*Role:* všichni včetně veřejnosti

### 3. Evidence výsledků GP

*Tabulka:* Vysledky

*Role:* všichni včetně veřejnosti

#### 3.1. Vložení výsledků

*Role pouze:* FIA, Admin, Uživatel pro svou GP

Vložit záznam, pokud GP je probíhající nebo je odjetá, jen uživatel pro svou GP

#### 3.2. Úprava výsledků

*Role pouze:* FIA, Admin, Uživatel pro svou GP

Upravit záznam, pokud GP je probíhající nebo je odjetá, jen uživatel pro svou GP

#### 3.3. Smazání výsledku

*Role pouze:* FIA, Admin

#### 3.4. Výpis výsledků pro zadanou GP

#### 3.5. Výpis tabulky pořadí jezdců v sezóně (body)

#### 3.6. Výpis tabulky pořadí v poháru konstruktérů (body)

#### 3.7. Výpis pořadí ve všech odjetých GP pro zadaného jezdce

#### 3.8. Výpis pořadí ve všech odjetých GP, kde bodoval, pro zadaného jezdce

#### 3.9. Výpis jezdců v odjetých GP pro zadané umístění

### 4. Evidence bodů za umístění

*Tabulka:* Body

*Role:* FIA, Admin

#### 4.1. Vložení pořadí

#### 4.2. Úprava pořadí

#### 4.3. Smazání pořadí

#### 4.4. Výpis všech pořadí

*Role:* všichni včetně veřejnosti

### 5. Evidence států

*Tabulka:* Staty, Tymy, Jezdci

*Role:* FIA, Admin

#### 5.1. Vložení státu

#### 5.2. Úprava státu

#### 5.3. Smazání státu

#### 5.4. Výpis všech států

#### 5.5. Výpis týmů, podle zadaného státu

*Role:* všichni včetně veřejnosti

#### 5.6. Výpis jezdců, podle zadaného státu

*Role:* všichni včetně veřejnosti



## 6. Evidence výrobců motorů

*Tabulka:* Vyrobc\_e\_motoru

*Role:* FIA, Admin

- 6.1. Vložení výrobce motorů
  - 6.2. Úprava výrobce motorů
  - 6.3. Smazání výrobce motorů
  - 6.4. Výpis všech výrobců motorů
- Role:* všichni včetně veřejnosti

## 7. Evidence týmů

*Tabulka:* Tmy, Vyrobc\_e\_motoru

*Role:* FIA, Admin

- 7.1. Vložení týmu
  - 7.2. Úprava týmu
- Role:* FIA, Admin, Uživatel pouze svůj záznam
- 7.3. Smazání týmu
  - 7.4. Výpis všech týmů
- Role:* všichni včetně veřejnosti
- 7.5. Detail týmu
- Role:* všichni včetně veřejnosti
- 7.6. Výpis všech jezdců tohoto týmu
- Role:* všichni včetně veřejnosti

## 8. Evidence motorů

*Tabulka:* Motory

*Role:* FIA, Admin

- 8.1. Vložení motoru
  - 8.2. Smazání motoru
  - 8.3. Výpis všech motorů
- Role navíc:* Uživatel Team
- 8.4. Výpis všech namontovaných motorů
- Role navíc:* Uživatel Team

## 9. Evidence jezdců

*Tabulka:* Jezdci, Motory

*Role:* FIA, Admin

- 9.1. Vložení jezdce
  - 9.2. Úprava jezdce
- Role navíc:* Uživatel (tým) pouze jezdce svého týmu
- 9.3. Smazání jezdce
  - 9.4. Výpis všech jezdců
  - 9.5. Změnit namontovaný motor
- Při namontování motoru se atribut Motory.Pocet\_pouziti inkrementuje o 1
- 9.6. Detail jezdce

## 4.2. Detailní popis funkcí

### Funkce 1. Evidence uživatelů

#### 1.1 Vložení uživatele

*Vstup:* \$id – odpovídá Tymy.ID nebo GP.ID, v závislosti na \$typ\_uzivatele  
\$login\_uzivatele – odpovídá atributu Uzivatel.Login  
\$heslo\_uzivatele – odpovídá atributu Uzivatel.Heslo  
\$typ\_uzivatele – odpovídá atributu Uzivatel.Type  
\$info1 ... \$info6 – další atributy na základě výběru typu uživatele ve formuláři

*Popis:* Tato procedura bude vkládat nového uživatele. Podle vstupu \$typ\_uzivatele se vloží, případně upraví atributy v jiných tabulkách. Procedura bude transakce.

1. Proměnná \$id\_uzivatele odpovídá poslední největší hodnotě Uzivatel.ID + 1.

*SELECT MAX(ID) FROM Uzivatel*

2. Pokud \$typ\_uzivatele = `GP`  
Proměnná \$id\_gp odpovídá poslední největší hodnotě GP.ID + 1.  
*SELECT MAX(ID) FROM GP*

Data se vloží do příslušných tabulek.

*INSERT INTO Uzivatel VALUES (\$id\_uzivatele, \$login\_uzivatele, \$heslo\_uzivatele, `gp`)*

*INSERT INTO GP VALUES (\$id\_gp, \$info1, \$info2, \$info3, \$info 4, \$info 5, \$info6, \$id\_uzivatele )*

3. Pokud \$typ\_uzivatele = `team`  
Proměnná \$id\_team odpovídá poslední největší hodnotě Team.ID + 1

*SELECT MAX(ID) FROM Team*

Data se vloží do příslušných tabulek. Vstupní parametr info 6 se nebude využívat.

*INSERT INTO Uzivatel VALUES (\$id\_uzivatele, \$login\_uzivatele, \$heslo\_uzivatele, `team`)*

*INSERT INTO Tymy VALUES (\$id\_team, \$info1, \$info2, \$info3, \$info 4, \$info 5 , \$id\_uzivatele)*

4. Pokud \$typ\_uzivatele = `FIA`, po vykonání procedura končí

*INSERT INTO Uzivatel VALUES (\$id\_uzivatele, \$login\_uzivatele, \$heslo\_uzivatele, \$typ\_uzivatele)*

5. Pro \$typ\_uzivatele = `admin` se neprovede nic, protože tento účet je výchozí uživatel. Pro jiné typy se také nic neprovede, protože nejsou nadefinované jiné typy.
6. Procedura končí

### 1.5 Detail uživatele

*Vstup:* \$id\_uzivatele – odpovídá atributu Uzivatel.ID

*Popis:* Pro vypisovaného uživatele vypíšeme informace, podle jeho typu. Tento výpis se bude realizovat jako procedura a jako transakce.

1. Je potřeba zjistit typ uživatele

```
SELECT Type FROM Uzivatel WHERE ID = $id_uzivatele
```

2. Pokud Type = `team`, po vykonání procedura končí

```
SELECT Uzivatel.ID, Tymy.Nazev, Dodavatel_pneumatik,  
       Vyrobc_motoru.Nazev  
FROM Uzivatel  
JOIN Tymy ON Uzivatel.ID = Tymy.Uzivatel_ID  
JOIN Vyrobc_motoru ON Tymy.Vyrobc_motoru_ID =  
       Vyrobc_motoru.ID  
JOIN Staty ON Tymy.Staty_ID = Staty.ID  
WHERE Uzivatel.ID = $id_uzivatele
```

3. Pokud Typ = `GP`, po vykonání procedura končí

```
SELECT Uzivatel.ID, GP.Nazev, Delka_okruhu,  
       Pocet_kol, Staty.Nazev  
FROM Uzivatel  
JOIN GP ON Uzivatel.ID = GP.Uzivatel_ID  
JOIN Staty ON GP.Staty_ID = Staty.ID  
WHERE Uzivatel.ID = $id_uzivatele
```

4. Pro jiné hodnoty, po vykonání procedura končí

```
SELECT ID, Type FROM Uzivatel WHERE ID = $id_uzivatele
```

### Funkce 3. Evidence výsledků GP

#### 3.5 Výpis tabulky pořadí jezdců v sezóně (body)

*Popis:* Bodový výpis všech jezdců v sezóně.

```
SELECT Jmeno, Prijmeni, SUM(Pocet_bodu)  
FROM Vysledky  
LEFT JOIN Body ON Vysledky.Body_Poradi = Body.Poradi  
LEFT JOIN Jezdci ON Jezdci.ID = Vysledky.Jezdci_ID  
GROUP BY Jmeno, Prijmeni  
ORDER BY SUM(Pocet_bodu) DESC
```

#### 3.6 Výpis tabulky pořadí v poháru konstruktérů (body)

*Popis:* Bodový výpis všech týmů v poháru konstruktérů pro jednotlivé týmy.

```
SELECT Tymy.ID, Nazev, SUM (Pocet_bodu)  
FROM Vysledky  
LEFT JOIN Body ON Vysledky.Body_Poradi = Body.Poradi  
LEFT JOIN Jezdci ON Jezdci.ID=Vysledky.Jezdci_ID  
LEFT JOIN tymy ON tymy.ID=Jezdci.Tymy_ID  
GROUP BY Tymy.ID, Nazev
```

*ORDER BY SUM(Pocet\_bodu) DESC*

### 3.8 Výpis jezdců ve všech odjetých GP pro zadané umístění

*Vstup:* \$cislo – číselný vstup uživatele

\$typ – typ číselného vstupu

\$vypis – zobrazení buď pořadí nebo bodů

*Popis:* Vypíše se pouze ty GP a požadovanou vlastnost (pořadí nebo body), kde zadaný jezdec bodoval. Procedura bude transakce.

1. Zjistíme poslední pozici, která je bodovaná. Tuto hodnotu si uložíme do proměnné \$maximum.

*SELECT MAX(Poradi) FROM Body WHERE NOT Pocet\_bodu = 0*

Následují SELECTy, které podle zadaných kritérií vypíší požadované informace.

2. Pokud \$typ='startovni' a \$vypis = 'body', po vykonání procedura končí.

*SELECT GP.Nazev, Body.Pocet\_bodu  
FROM Vysledky  
JOIN Body ON Vysledky.Body\_Poradi = Body.Poradi  
JOIN Jezdci ON Vysledky.Jezdci\_ID = Jezdci.ID  
JOIN GP ON Vysledky.GP\_ID = GP.ID  
WHERE Jezdci.Startovni\_cislo = \$cislo  
AND Vysledky.Body\_poradi <= \$maximum  
AND Vysledky.Body\_poradi > 0*

3. Pokud \$typ='id' a \$vypis = 'body', po vykonání procedura končí

*SELECT GP.Nazev, Body.Pocet\_bodu  
FROM Vysledky  
JOIN Body ON Vysledky.Body\_Poradi = Body.Poradi  
JOIN GP ON Vysledky.GP\_ID = GP.ID  
WHERE Vysledky.Jezdci\_ID = \$cislo  
AND Vysledky.Body\_poradi <= \$maximum  
AND Vysledky.Body\_poradi > 0*

4. Pokud \$typ='startovni' a \$vypis = 'poradi', po vykonání procedura končí

*SELECT GP.Nazev, Vysledky.Body\_poradi  
FROM Vysledky  
JOIN Jezdci ON Vysledky.Jezdci\_ID = Jezdci.ID  
JOIN GP ON Vysledky.GP\_ID = GP.ID  
WHERE Jezdci.Startovni\_cislo = \$cislo  
AND Vysledky.Body\_poradi <= \$maximum  
AND Vysledky.Body\_poradi > 0*

5. Pokud \$typ='id' a \$vypis = 'body', po vykonání procedura končí

```
SELECT GP.Nazev, Vysledky.Body_poradi  
FROM Vysledky  
JOIN GP ON Vysledky.GP_ID = GP.ID  
WHERE Vysledky.Jezdci_ID = $cislo  
AND Vysledky.Body_poradi <= $maximum  
AND Vysledky.Body_poradi > 0
```

## 5. Návrh uživatelského rozhraní

### 5.1. Menu

#### 1. Jezdci

- 1.1. Vložení jezdce – funkce 9.1 (role: FIA, Admin)
- 1.2. Výpis všech jezdců – funkce 9.4
  - Editace jezdce (role: FIA, Uživatel team – svého jezdce) – funkce 9.2
  - Smazání jezdce (role: FIA, Admin) – funkce 9.3
  - Detail jezdce

#### 2. Týmy

- 2.1. Vložení týmu (role: FIA, Admin) – funkce 7.1
- 2.2. Výpis všech týmů – funkce 7.4
  - Editace týmu (role: FIA, Admin, Uživatel team – sám sebe) – funkce 7.2
  - Smazání týmu (role: FIA, Admin) – funkce 7.3
  - Detail týmu
    - Zobrazení jezdců tohoto týmu – funkce 7.6
    - Viz Menu 1.2 – pouze své jezdce
    - Změnit namontovaný motor (role: FIA, Admin, Uživatel team – svého jezdce) – funkce 9.5

#### 3. GP

- 3.1. Vložení GP (role: FIA, Admin) – funkce 2.1
- 3.2. Přehled vytvořených GP – funkce 2.4
  - Úprava GP (role: FIA, Admin, Uživatel GP – svou vlastní GP) – funkce 2.2
  - Smazání GP (role: FIA, Admin) – funkce 2.3
  - Detail GP – funkce 2.5
    - Vložení výsledku (role: FIA, Admin, Uživatel GP – svou vlastní GP) – funkce 3.1
    - Úprava výsledku (role: FIA, Admin, Uživatel GP – svou vlastní GP) – funkce 3.2
    - Smazání výsledku (role: FIA, Admin, Uživatel GP – svou vlastní GP) – funkce 3.3
    - Zobrazení výsledků – funkce 3.4

#### 4. Tabulky

- 4.1. Výpis pořadí jezdců – funkce 3.4
  - Výpis pořadí v odjetých GP – funkce 3.6
  - Výpis pořadí v odjetých GP, pouze kde bodoval – funkce 3.7
  - Výpis jezdců v odjeté GP pro zadané umístění – funkce 3.8
- 4.2. Výpis pořadí poháru konstruktérů – funkce 3.5

## 5. Administrace (role: Admin, FIA)

### 5.1. Uživatelé (role: Admin)

5.1.1. Vložení uživatele – funkce 1.1

5.1.2. Přehled uživatelů – funkce 1.4

Úprava uživatele – funkce 1.2

Smazat uživatele – funkce 1.3

Detail uživatele – funkce 1.5

### 5.2. Body (role: FIA, Admin)

5.2.1. Výpis všech pořadí

Vložení pořadí – v4.1

Úprava vložení – funkce 4.2

Smazání pořadí – funkce 4.3

### 5.3. Evidence států (role: FIA, Admin)

Podobné, jako Body (viz Menu 4.2)

### 5.4. Výrobce motorů (role: FIA, Admin)

Podobné, jako Body (viz Menu 4.2)

### 5.5. Motory (role: FIA, Admin)

Podobné, jako Body (viz Menu 4.2)

Výpis všech namontovaných motorů – funkce 8.4

## 5.2. Návrh formulářů

Detail vložení výsledků GP – funkce 4.1

obdobně úprava výsledků – funkce 4.2

**Vložení výsledků pro GP Monaka** ← Název GP – funkce 2.5

Datum: 28. května 2017 ← Datum GP – funkce 2.5

Pořadí	Body	Jméno jezdce	Nedokončil
1	25	<input type="text" value="Jezdec"/> ▾	<input type="checkbox"/>
2	18	<input type="text" value="Jezdec"/> ▾	<input type="checkbox"/>
3	15	<input type="text" value="Jezdec"/> ▾	<input checked="" type="checkbox"/>
.	.	.	.
.	.	.	.
21	0	<input type="text" value="Jezdec"/> ▾	<input checked="" type="checkbox"/>
22	0	<input type="text" value="Jezdec"/> ▾	<input checked="" type="checkbox"/>

← Funkce 9.4 – výpis všech jezdců

↑ Funkce 3.1 - vložení  
případně  
Funkce 3.2 - úprava

Funkce 4.4 – výpis všech pořadí

## 5.6. Vložení uživatele – funkce 1.1

### Vložení uživatele

Login

Typ uživatele

Heslo

Typ uživatele

▼

\*\*\*\*\*

Výběr typu uživatele  
FIA/Admin/GP/team

Na základě výběru typu uživatele se  
dynamicky doplní formulář:  
GP – formulář funkce 2.4  
Team – formulář funkce 7.4  
FIA – nepřidá se žádný formulář

Zadání hesla

Místo pro dynamicky vygenerovaný formulář

Uložení formuláře – funkce 1.1

Uložit