

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
Fakulta informačních technologií



Informační systém fotbalového klubu

Ročníkový projekt

Vlastimil Mareš

2006



## PODĚKOVÁNÍ

Na tomto místě bych rád poděkoval svému vedoucímu ročníkového projektu Ing. Zbyňku Křivkovi za připomínky a rady, které mi pomohly projekt vypracovat.

## PROHLÁŠENÍ

Prohlašuji, že jsem tento ročníkový projekt vypracoval samostatně pod vedením Ing. Zbyňka Křivky. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Brně 2. května 2006-04-22

Vlastimil Mareš

# Abstrakt

Cílem tohoto projektu je vyzkoušet si dynamickou tvorbu webových stránek, důkladně nastudovat základy skriptovacího jazyka php a pochopit databázový systém MySQL. Dále vytvořit model informačního systému malého fotbalového klubu v jazyce UML. Pomocí tohoto modelu a výše zmíněných technologií poté vytvořit samotný informační systém. Poté naplnit systém odpovídajícími daty a zhodnocení použitých technologií a jaké další technologie by bylo vhodné použít.

# Klíčová slova

Informační systém, PHP, MySQL, databázový systém

# Abstract

The point of this project is to try the creation of dynamic web pages, study the scripting language PHP and understand the database system MySQL. The next target is to create a model of the small football team information system in the UML language and make the own information system using this model and other technologies mentioned above. At last it is necessary to fill in the system with the corresponding data and discuss the technologies used and others which could be used in this project.

# Keywords

Information systém, PHP, MySQL, database system

# OBSAH

<b>1 Úvod</b> .....	<b>6</b>
<b>2 Teoretický rozbor</b> .....	<b>7</b>
<b>3 Detailní rozbor systému</b> .....	<b>9</b>
3.1 Funkce systému.....	9
3.1.1 Nepřihlášený uživatel.....	9
3.1.1.1 Články.....	10
3.1.1.2 Hráči.....	10
3.1.1.3 Zápasy.....	10
3.1.1.4 Ostatní.....	10
3.1.2 Přihlášený uživatel.....	10
3.1.2.1 Uživatel reportér.....	11
3.1.2.2 Uživatel ekonomický ředitel.....	12
3.1.2.3 Uživatel sportovní ředitel.....	12
3.1.2.4 Uživatel administrátor.....	13
3.1.3 Hesla.....	14
3.1.4 Hlídání chyb.....	14
3.1.5 Zobrazení článků.....	14
3.2 ER diagram.....	15
3.3 Use case diagram.....	
<b>4 Implementace</b> .....	<b>17</b>
4.1 Obecně k programu.....	17
4.2 Nejzajímavější části implementace.....	17
4.2.1 Užití pole \$_SESSION.....	17
4.2.2 Hesla v systému.....	17
4.2.3 Datумы v systému.....	18
4.2.4 Obrázky.....	18
4.2.5 Aktualizace údajů u hráčů.....	19
4.3 CSS styly a grafika.....	19
4.4 SQL.....	19
4.5 K programování.....	20
<b>5 Závěr</b> .....	<b>21</b>
<b>6 Literatura</b> .....	<b>22</b>
<b>7 Přílohy</b> .....	<b>23</b>
7.1 Uživatelská příručka.....	23
7.1.1 Uživatel reportér.....	23
7.1.2 Uživatel sportovní ředitel.....	23
7.1.3 Uživatel administrátor.....	25
7.1.4 Změna hesla.....	25
7.2 Jednotlivé soubory a jejich popis.....	25

# 1 Úvod

Tento projekt je zaměřen na tvorbu informačního systému, respektive internetového informačního systému. Systém je naprogramován ve skriptovacím jazyce PHP , pomocí HTML (a kaskádových stylů), javascriptu a databázového systému MySQL (zadání umožnilo výběr databázového systému. MySQL sem si zvolil z několika důvodů. Není na něj potřeba žádná licence, není složitý na pochopení ani na implementaci, v kombinaci s php jde o často používané a tím pádem dobře zdokumentované spojení a navíc mě to ve spojení s php osobně zajímalo a chtěl jsem se to naučit)

Protože se jedná o internetový informační systém malého fotbalového klubu, tak hlavní náplní projektu je internetová prezentace tohoto klubu. Na stránkách si návštěvník může přečíst články různých rubrik, přečíst si o historii klubu, vyhledat statistiky jednotlivých hráčů nebo statistiky všech týmů v klubu. Dále stránky obsahují výsledky všech týmů i rozpis jednotlivých utkání, která se ještě budou hrát. Přihlášení uživatelé pak mohou tyto záznamy přidávat/ubírat či editovat.

**Teoretický rozbor** - úvod do problematiky informačních systémů, popis jednotlivých jazyků a technologií, které byly v projektu použity.

**Detailní rozbor systému** – popis jednotlivých částí systému, uživatelů a jejich práv a všech možností spojených s používáním systému.

**Implementace** – popis vlastní implementace , popis nejdůležitějších skriptů a vypsání největších problémů a úskalí, které mě při práci potkaly.

**Závěr** – vyhodnocení a úvaha o projektu

## 2 Teoretický rozbor

**Informační systémy** - jde o komplex lidí, dat, informací, programových a technických produktů, organizačních postupů, obchodních pravidel zabezpečující sběr, přenos, aktualizaci, uchování a další zpracování dat za účelem prezentace informací pro potřeby uživatelů.

**PHP (Personal Home Page)**- je dnes velmi rozšířená technologie umožňující snadné programování na straně serveru (server-side programming). Toho lze využít k tvorbě různých interaktivních webových stránek. Stručně lze říct, že skript napsaný v PHP je vygenerován ze serveru podle zadaných kritérií a výsledek je odeslán volajícímu počítači stejným způsobem, jakým se odesílají běžné statické (HTML) stránky. Jakmile je však stránka načtena, pomocí PHP ji již není možné dále měnit.

**HTML** - Hypertext markup Language (hypertextový značkovací jazyk) - Jazyk HTML se skládá z množiny prvků, které definují dokument a návod jak jej zobrazit.

**Databáze** - určitá uspořádaná množina informací (dat) uložená na paměťovém médiu.

**Databázový systém** – je softwarové vybavení, které zajišťuje práci s databází, tzn. tvoří rozhraní mezi aplikačními programy a uloženými daty.

**MySQL** - zkratka z angl. My Structured Query Language = systém pro řízení databází. Do MySQL lze ukládat různá data (texty, obrázky atd.), s nimiž lze dále jednoduše pracovat (třídít, řadit, filtrovat apod.). Nejčastěji se MySQL používá ve spojení s jazykem PHP, které umožňuje přístup k uloženým datům. MySQL je relační databáze typu DBMS (database management system) a vychází z deklarativního programovacího jazyka SQL (Structured Query Language). Je šířen jako Open Source. Díky své licenci a rychlosti je v poslední době téměř nejoblíbenějším systémem. MySQL je malý, rychlý a jednoduchý databázový systém. Databáze MySQL má některá omezení, které obsahují jiné databázové systémy, např. robustní Oracle. Právě díky tomu dosahuje vynikající rychlosti.

**Javascript** - je multiplatformní, objektově orientovaný skriptovací jazyk, pocházející z dílny společnosti Netscape. Nyní se zpravidla používá jako interpretovaný programovací jazyk pro WWW stránky, vkládaný přímo do HTML kódu stránky. Jsou jím obvykle ovládány různé interaktivní prvky GUI (tlačítka, textová políčka) nebo tvořeny animace a efekty obrázků.

**CSS style** - zkratka pro anglický název Cascading Style Sheets, česky tabulky kaskádových stylů. Je to jazyk pro popis způsobu zobrazení stránek napsaných v jazycích HTML, XHTML nebo XML. Hlavním smyslem je umožnit návrhářům oddělit vzhled dokumentu od jeho struktury a obsahu. Původně to měl umožnit už jazyk HTML, ale v důsledku nedostatečných standardů a konkurenčního boje výrobců prohlížečů se vyvinul jinak. Starší verze HTML obsahují celou řadu elementů, které nepopisují obsah a strukturu dokumentu, ale i způsob jeho zobrazení. Z hlediska zpracování dokumentů a vyhledávání informací není takový vývoj žádoucí.



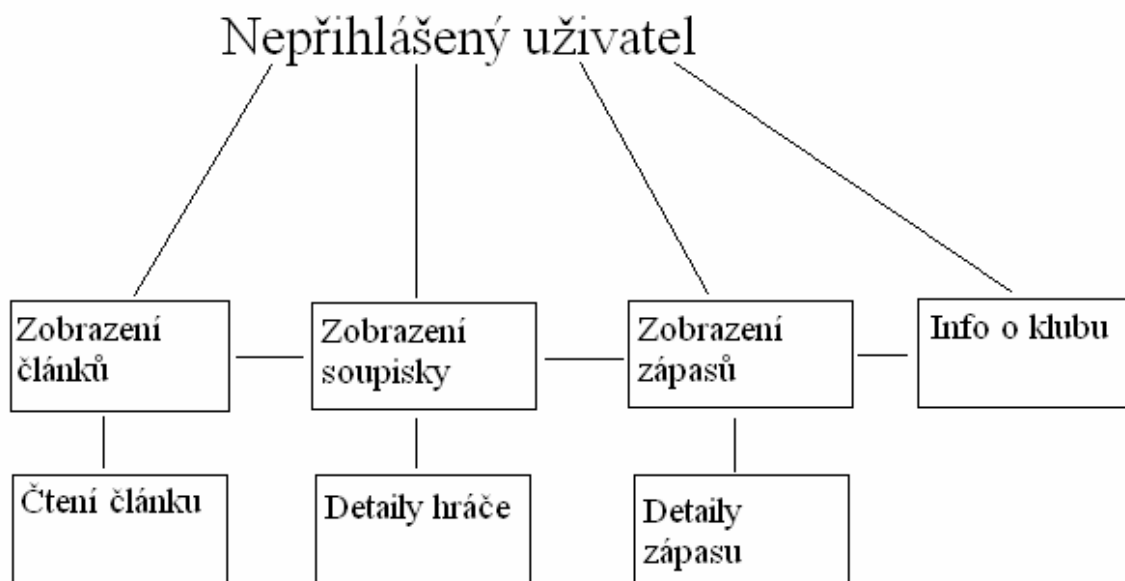
## 3 Detailní rozbor systému

### 3.1 Funkce systému

Hlavním úkolem systému je prezentace klubu na internetu. Samotný systém se potom dělí na několik hlavních částí. Část zaměřenou na články a jejich zprávu, část zaměřenou na hráče, údaje o nich a jejich statistky a na část zaměřenou na zápasy. Systém se chová jinak pro nepřihlášeného uživatele a pro přihlášeného uživatele. Část pro přihlášeného uživatele se pak ještě dělí podle práv jednotlivých uživatelů. Systém je ovládán hlavně pomocí menu v levé části obrazovky (uživatelská příručka je v přílohách).

#### 3.1.1 Nepřihlášený uživatel

Možnosti nepřihlášeného uživatele jsou značně omezené. Jedná se tedy o návštěvníka (fanouška) stránek. Nemá práva k žádné editaci a systém si může pouze prohlížet. Přístupné jsou mu všechny tři hlavní části.



Obrázek 1: možnosti nepřihlášeného uživatele

### **3.1.1.1 Články**

Může si číst články, které jsou na úvodní straně. Má zde možnost si zvolit (dole pod články je rozbalovací menu s jednotlivými rubrikami), zda chce mít zobrazeny všechny rubriky zároveň nebo pouze vždy jednu vybranou rubriku. Na stránku se zobrazuje pouze pět nejnovějších článků. Pokud má zájem i o starší články, tak pod články je odkaz, kterým se dostane do archivu článků. Zde záleží na tom, jaké články měl zobrazeny a podle toho se dostane do archivu buď všech článků nebo opět pouze jedné rubriky. Trochu mimo tyto rubriky stojí rubrika historie klubu, která má svůj vlastní odkaz v menu. Dále si může uživatel zobrazit základní údaje o klubu (kontakty, adresa hřiště...).

### **3.1.1.2 Hráči**

Uživatel si může prohlížet soupisky jednotlivých týmů klubu. Buď je může mít vypsány všechny hráče všech týmů nebo pouze hráče jednoho týmu (např. Muži A). Hráči jsou řazeni automaticky podle čísla, ale uživatel má možnost si řazení změnit podle jména nebo postu. Každý hráč má v systému uloženy podrobnosti o své osobě (počet gólů atd.). Uživatel se k nim dostane kliknutím na jméno hráče.

### **3.1.1.3 Zápasy**

Uživatel si může prohlížet rozpis zápasů jednotlivých týmů na aktuální sezonu nebo prohlížet výsledky již odehraných zápasů. Při kliknutí na zápas se opět zobrazí větší podrobnosti o něm. Uživatel si opět volí tým, jehož zápasy chce vidět. Zde kvůli přehlednosti chybí možnost zobrazení zápasů všech týmů najednou.

### **3.1.1.4 Ostatní**

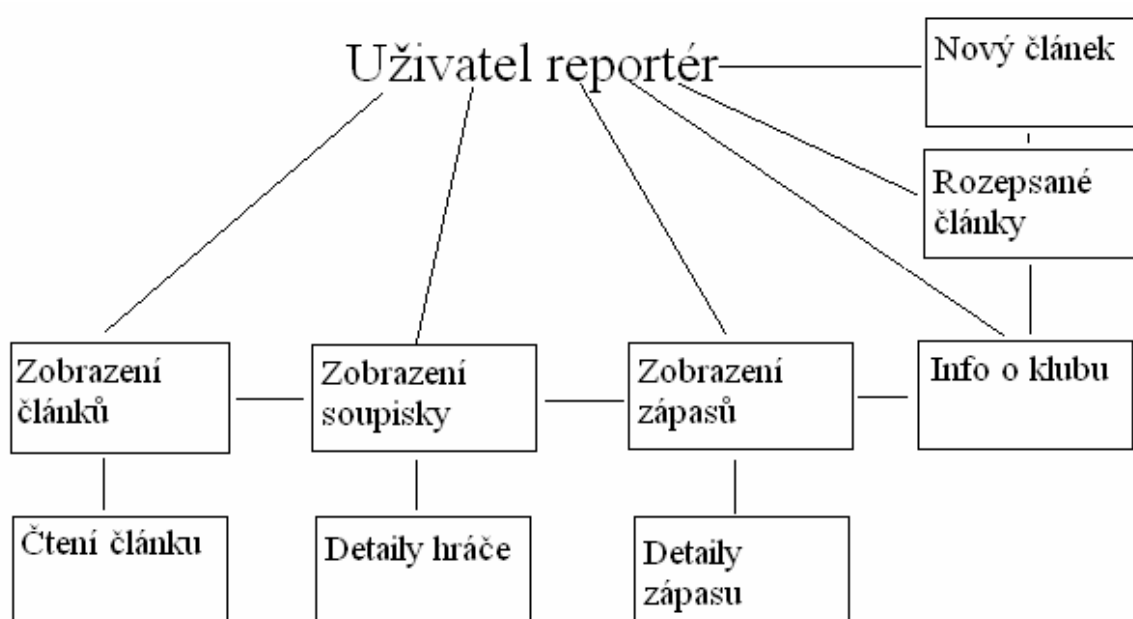
V levém sloupci je pod menu tabulka s umístěním a počtem bodů A mužstva a jeho soupeřů. V levém sloupci je zobrazen příští nejbližší zápas a příští nejbližší zápas, který se bude hrát v domácím prostředí.

## **3.1.2 Přihlášený uživatel**

V pravém horním rohu je přihlašovací dialog. Po přihlášení může uživatel již systém spravovat a editovat databázi. Záleží ovšem na jeho právech. V systému jsou čtyři různé typy uživatelů, které mohou být libovolně kombinovány.

### 3.1.2.1 Uživatel reportér

Tento uživatel je v systému, aby psal nové články nebo upravoval či mazal články staré. Žádná jiná práva v systému nemá. Jako jediný uživatel (nepočítáme-li nepřihlášeného uživatele) nemá svou speciální stránku, ale má jen přidány dvě položky v levém menu. Je to tak, protože vlastní stránka by pro něj byla zbytečná a nevyužitá. Jinak už může dělat jen to co nepřihlášený uživatel.



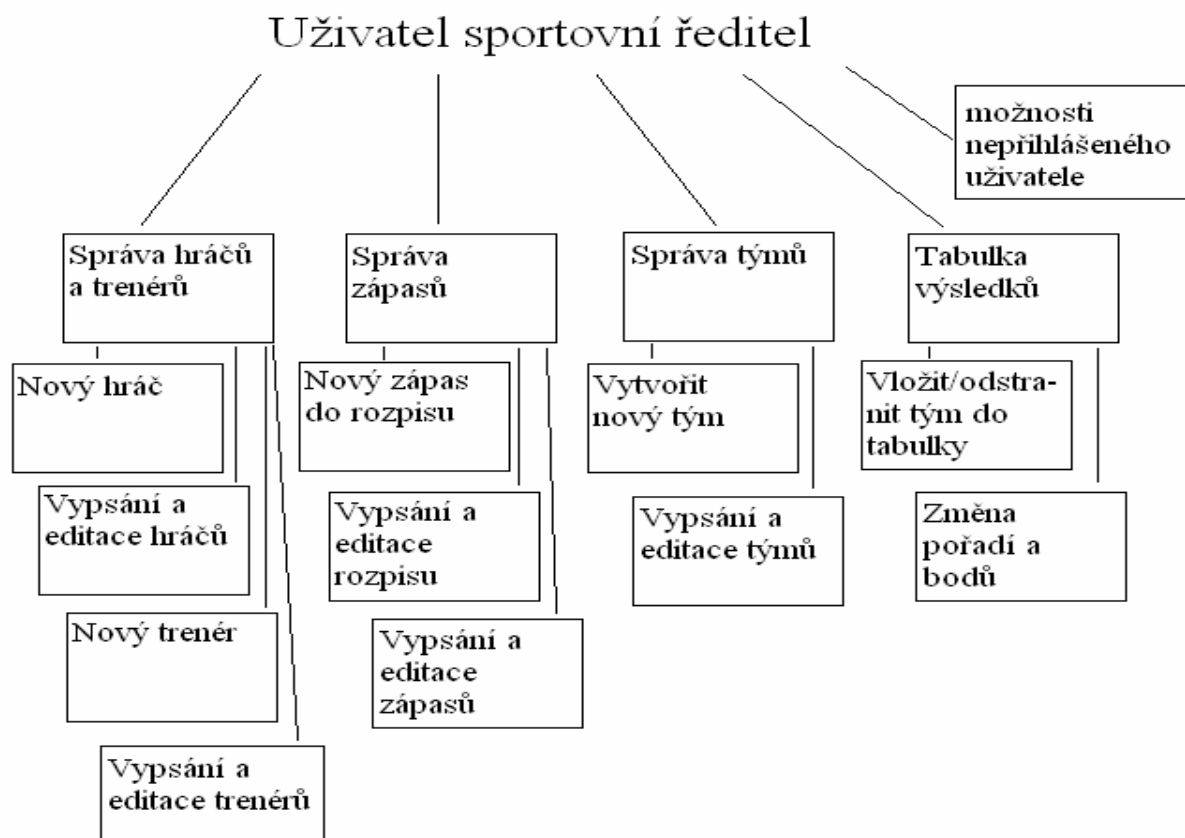
Obrázek 2: možnosti uživatele reportér

### 3.1.2.2 Uživatel ekonomický ředitel

Tento uživatel má starosti ekonomické funkce v systému. Bohužel ekonomická část systému není z časových důvodů implementována (jako jediná a nejméně důležitá část systému).

### 3.1.2.3 Sportovní ředitel

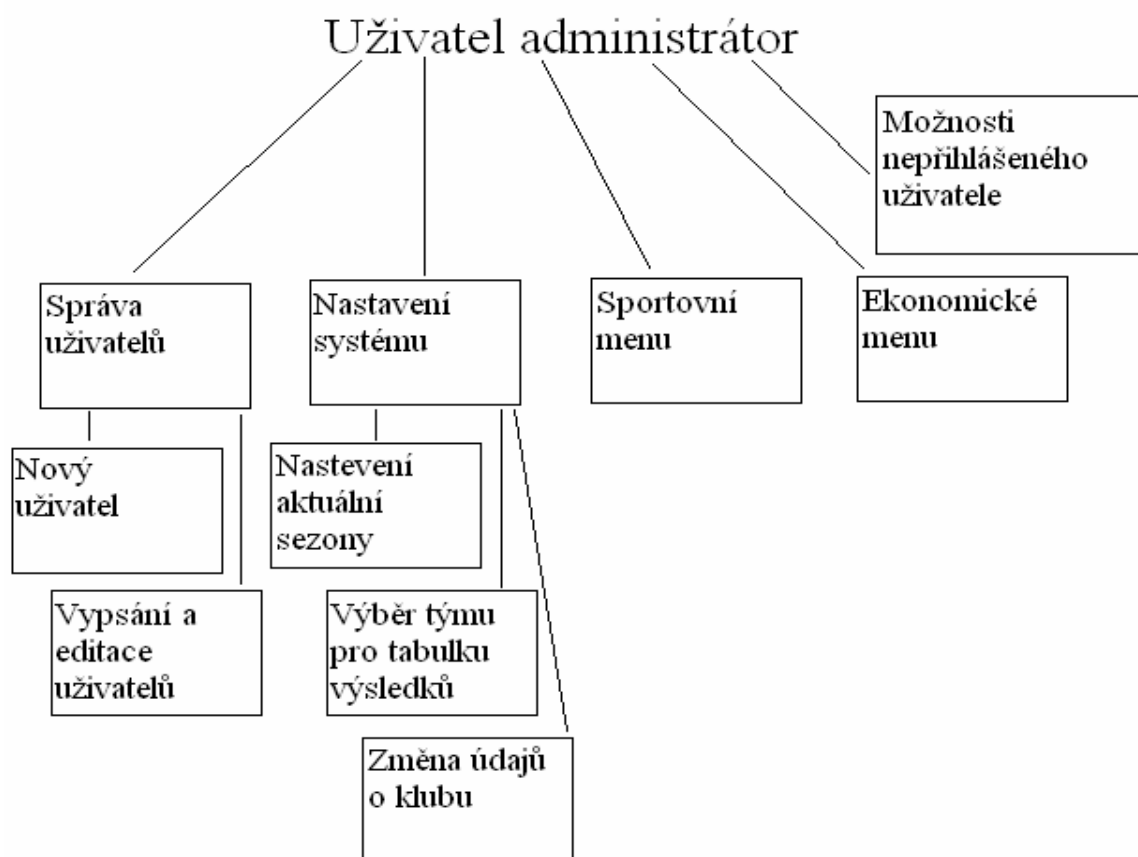
Tento uživatel má na starosti všechny části systému týkající se sportovní části. Může přidávat do databáze nové hráče nebo upravovat údaje těch již uložených. Také má práva hráče z databáze vymazat. Má na starosti také vkládání nových zápasů do rozpisu a zapisování výsledků již odehraných utkání. Opět může tyto utkání následně upravovat či mazat. Dále se stará o databázi trenérů. Také spravuje tabulku s umístěním a počtem bodů A týmu a jeho soupeřů, která se zobrazuje v levé části stránky. V jeho kompetenci je také správa týmů. Vytváří nové týmy a vkládá do nich hráče z databáze, kteří za něj hrají. Vybírá také, která sezona je aktuální, aby se potom návštěvníkům stránek zobrazovaly vždy aktuální informace. Sportovní ředitel již má svou stránku pro nejdůležitější operace.



Obrázek 3: možnosti uživatele sportovní manager

### 3.1.2.4 Administrátor

Jedná se o nejdůležitějšího uživatele systému. Administrátor má práva všech uživatelů jmenovaných výše, ale navíc má některá speciální. Jeho hlavní úlohou je správa uživatelských účtů. Je to jediný uživatel, který může vytvářet nové uživatelské účty. Dává uživatelům jednotlivá práva a může jako jediný jim následně měnit jejich hesla (nepočítáme-li, že každý uživatel může změnit svoje heslo). Ale ani on nevidí jaká hesla uživatelé mají. Dále také jako jediný může měnit informace o klubu. Jako jeho adresu, maily, telefony, logo klubu atd. Administrátor má také svou stránku s důležitými operacemi. Také nastavuje aktuální sezónu (v systému jsou to sezóny 05/06 až 10/11. Delší životnost systému není předpokládána). To je využíváno u vkládání utkání do rozpisu, kde je tato sezóna uživateli automaticky nabídnuta (může být samozřejmě změněna). Hlavní využití je ale při zobrazování zápasů, kde se automaticky zobrazí zápasy aktuální sezóny. Poté může uživatel sledovat i zápasy starších sezón.



Obrázek 4: možnosti uživatele sportovní manager

Vidíme, že zde jsou čtyři typy uživatelů, ale samozřejmě u konkrétního uživatele můžeme jednotlivé typy libovolně kombinovat a dát tak uživateli všechna práva, která potřebuje. Takže můžeme mít například uživatele, který je reportér a zároveň sportovní ředitel.

V původním návrhu systému jsem počítal ještě s uživatelem trenér, ale po hlubším zamyšlení jsem tohoto uživatele nakonec neimplementoval, protože v systému by byl nakonec prakticky zbytečný a jeho přítomnost by nevedla ke zlepšení funkčnosti, ale spíše jen k větší databázi.

### **3.1.3 Hesla**

Pro každého nového uživatele je heslo generováno automaticky a náhodně systémem. Toto heslo si pak může uživatel následně změnit. Ostatní údaje o uživateli jako je například login nebo jméno už zadává do systému administrátor a je on může tyto údaje měnit. Pro zvýšení bezpečnosti jsou hesla před přenosem hashována. Podrobnosti u popisu implementace.

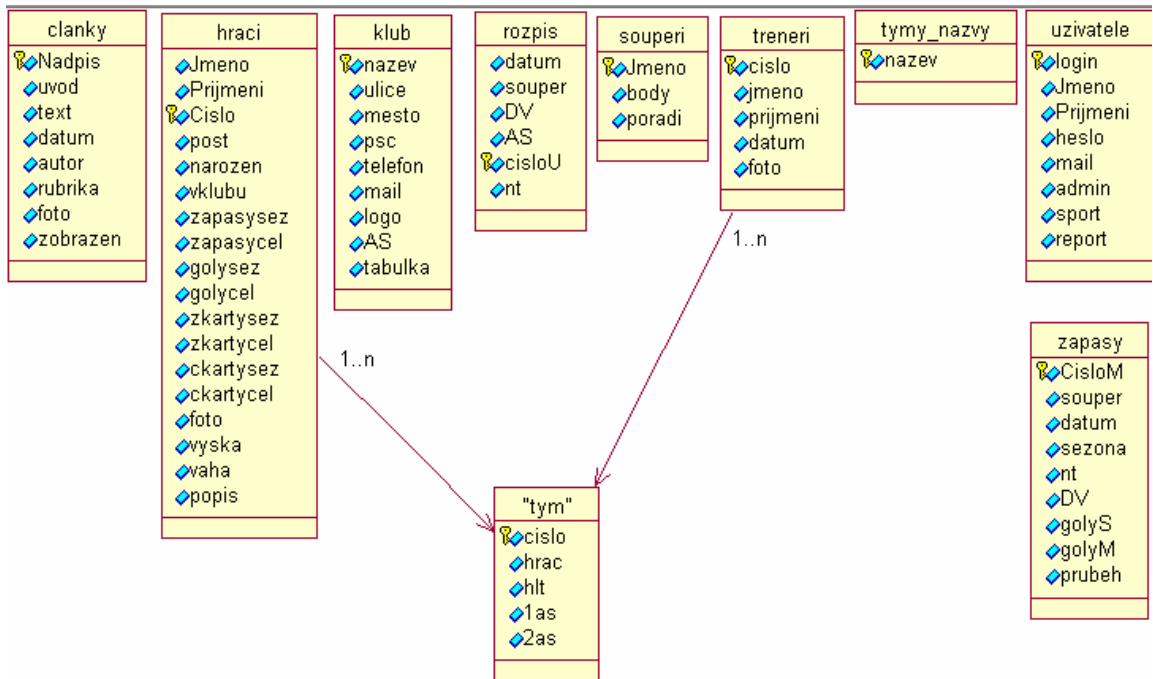
### **3.1.4 Hlídní chyby**

Systém samozřejmě uživateli příliš nevěří a proto ho při zadávání nebo úpravě údajů kontroluje. Pokud uživatel nevyplní ve formuláři povinný údaj, tak systém údaje neuloží, ale uživatele na chybu upozorní. Samozřejmě, že uživatel nepřijde o již vyplněné údaje, ty se mu opětovně vyplní stejnými hodnotami, jakými je naplnil než odeslal formulář. Pokud je nějaký údaj v tabulce databáze jedinečný, tak systém kontroluje jestli uživatel nezadával stejný a pokud ano, tak je na to opět upozorněn. Také zde samozřejmě nepřijde o již zadaná data.

### **3.1.5 Zobrazení článků**

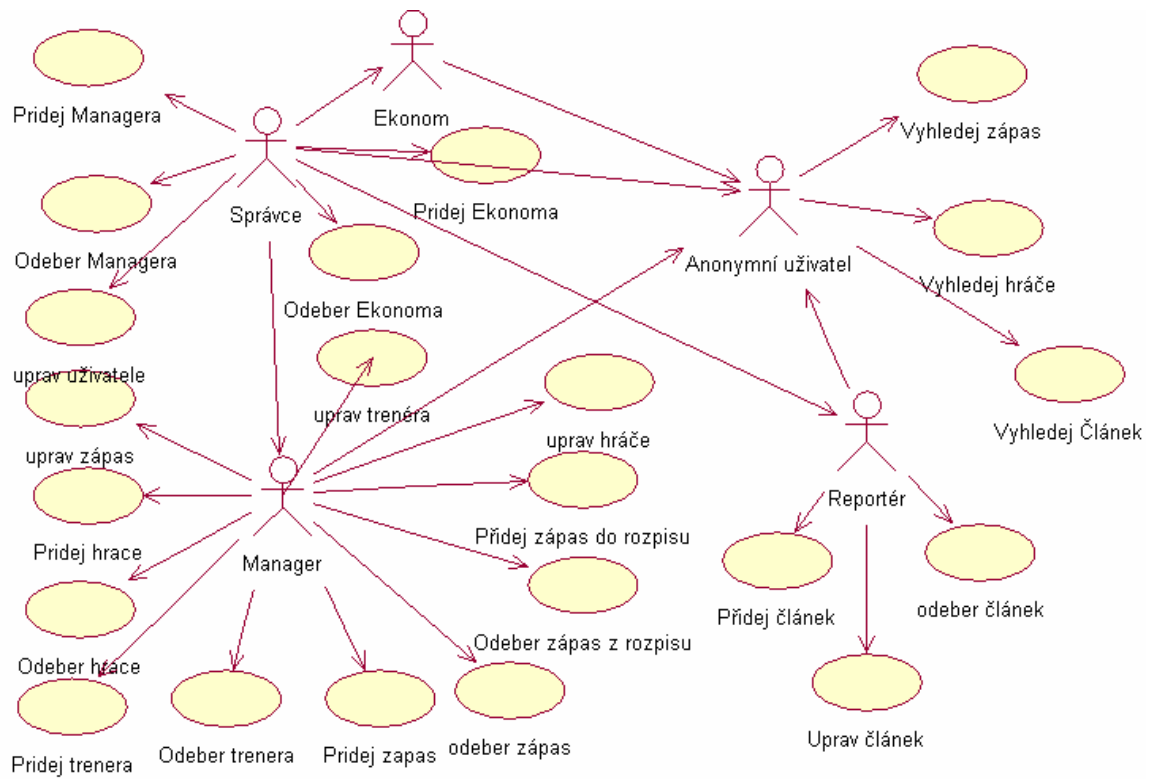
Pro zlepšení manipulace s články, byla přidána funkce zobrazení/nezobrazení článku. Je implementována z důvodu, že občas je potřeba z nějakých důvodů články stáhnout, ale nechceme ho smazat, tak je jediné řešení právě jenom jeho nezobrazení nepřihlášeným uživatelům. Další výhodou tohoto řešení je v tom, že když uživatel potřebuje uložit rozepsaný článek, tak mu jen dá atribut nezobrazovat a může ho uložit a později na něm dále pracovat a dokončit ho.

### 3.2 ER diagram databáze



Tabulka tým, znázorňuje všechny tabulky týmů v systému. Obrázek klíče značí primární klíč v tabulce. Provázání je uděláno tak, že hodnota hrac v tabulce "tym" se rovná hodnotě Cislo z tabulky hraci a hodnoty hlt, 1as, 2as se rovnají hodnotám z tabulky treneri.

### 3.3 Use case diagram





# 4 Implementace

## 4.1 Obecně k programu

V dalších částech jsem už používal buď jednoduché nebo známé konstrukce. Celý systém je založen na jednoduchých php skriptech doplněných o SQL dotazy

Programová část projektu je řešena je v programovacím jazyku PHP a ve skriptovacích jazycích HTML a Javascript (viz kapitola 2).

Program by se dal rozdělit na dvě části. Část formulářovou a část se skripty. Tyto dvě části se vždy nachází ve dvou souborech (jsou přiloženy na CD a jejich popis je v příloze). V prvním souboru je uložen formulář pro danou akci (například přidání nového uživatele do systému) a v druhém souboru je příslušný skript, který danou operaci provede s daty, které mu jsou zaslány z formuláře.

## 4.2 Nejzajímavější části implementace

### 4.2.1 Užití pole `$_SESSION`

Pokud bylo potřeba přenášet některá data a hodnoty mezi jednotlivými stránkami nebo skripty, použít jsem v tomto případě vždy pole `$_SESSION` a ne skrytá pole `<INPUT TYPE=HIDDEN...>`. Zvolil jsem takto hlavně z důvodů bezpečnosti. Další výhodou tohoto řešení je, pokud potřebujeme nějakou hodnotu použít ve více skriptech, tak ji nemusíme neustále posílat mezi skripty, ale jednou si ji uložíme a až do zavření prohlížeče k ní můžeme přistupovat.

Jako příklad dobře poslouží například soubor `login.php` v příloze. Slouží k přihlášení uživatele do systému. Do proměnné `$_SESSION['login']` se uloží `login` uživatele a na každé stránce, která vyžaduje určitá práva pro přístup, se podle hodnoty v proměnné `$_SESSION['login']` dále zjišťuje zda má i potřebná práva. Pokud ne, tak je mu přístup odepřen.

### 4.2.2 Hesla v systému

V systému má každý uživatel heslo. Při přihlašování nebo změně hesla jsou hesla kontrolována. Pro zvýšení bezpečnosti se hesla neposílají v normální podobě, ale jsou hashována. V projektu jsem na to použil funkci `php sha1()`, která vytvoří z hesla řetězec 40 znaků, ze kterého se původní řetězec nedá v rozumném čase zpátky zjistit původní heslo. V samotné databázi pak není uloženo heslo, ale jenom jeho hash. Ten je potom kontrolován s hashem, který je vytvořen z hesla, které uživatel zadal.

*if (\$zaznam['Heslo']!=sha1(\$\_POST['pass']))* tady je příklad kódu, kde se kontroluje heslo. *\$zaznam['Heslo']* je hash hesla v databázi a *\$\_POST['pass']* je heslo, které zadal uživatel.

sha1 není jediná hashovací funkce, v PHP je k dispozici např. také md5. U této funkce ale byly objeveny bezpečnostní trhliny, takže jsem použil funkci sha1.

### 4.2.3 Datумы v systému

Formát datumu, který podporuje MySQL (například 2006-04-30), je pro českého uživatele velmi nezvyklý. Proto se v systému používá klasický formát (například 30.4.2006). A na formát pro MySQL je převeden.

```
if ($datum!=""){
    $date="";
    $date.=$datum{6};
    $date.=$datum{7};
    $date.=$datum{8};
    $date.=$datum{9};
    $date.="-";
    $date.=$datum{3};
    $date.=$datum{4};
    $date.='-';
    $date.=$datum{0};
    $date.=$datum{1};
}
```

Toto ukázka kódu, který převede datum z námi používaného formátu do formátu používaného v MySQL (proměnná *\$datum* je datum zadané uživatelem a proměnná *\$date* je výsledné datum, které bude uloženo do databáze). Podobný převod se používá i pro zobrazení datumu uživateli.

### 4.2.4 Obrázky

Všechny obrázky, které se na stránkách zobrazují, jsou uloženy přímo v databázi a ne na serveru. Udělal jsem to pro větší přehlednost a ušetření diskového prostoru na serveru. U každého obrázku je hned je jasné, ke kterému záznamu databáze patří. Nevýhoda tohoto řešení je poněkud složitější ukládání a hlavně zobrazování. Obrázek si nemůžeme zobrazit jen pouhým selectem jako u všech ostatních údajů v databázi, ale je na to potřeba speciální skript. Ale jelikož data nejsou uložena přímo na serveru, tak si je nemůžeme prohlížet a mazat nebo přidávat, ale musí se k tomu využívat systém. To je ale další výhoda. Takhle může se všemi daty manipulovat opravdu jenom přihlášený uživatel. Takže když se na server dostane neoprávněná osoba, tak se k datům nedostane. Při znalosti php by sice mohla změnit kódy, ale to už je o zabezpečení fyzickému přístupu na server a to není součástí této práce. Při vkládání obrázku je hlídáno, aby jeho velikost a rozměry byly rozumné. Samotné vkládání probíhá tak, že nejprve jsou pomocí SQL příkazu INSERT vloženy do databáze všechny údaje kromě obrázku (obrázek je nepovinný údaj, takže nemusí být vložen). Poté jsou provedeny všechny kontroly a zakódování (pomocí *base64\_encode*), pokud je vše v pořádku, tak se pomocí SQL příkazu UPDATE vloží obrázek. Před zobrazením musí být ještě obrázek zpětně

dekódován pomocí *base64\_decode*. Řešení je možné si prohlédnout v příloze například v souborech *insert.php* a *zobrazit\_logo.php*.

#### 4.2.5 Aktualizace údajů u hráčů

Všechny údaje v databázi mohou být klasicky upravovány a aktualizovány pomocí k tomu určených formulářů. Hráči jsou, ale aktualizováni ještě jedním způsobem. Při zadávání výsledků zápasů se zadávají i statistiky jednotlivých hráčů v daném utkání (počet branek, karet atd.). To představuje asi nejsložitější konstrukci, co jsem v systému použil. Pod údaji o utkání se vypíše tabulka s hráči daného týmu a u nich se vyplňují údaje. Tady je problém první a to jak pojmenovat jednotlivé položky. Vyřešil jsem to pomocí indexů položek. Hráče jsem vypisoval podle čísla a pak u každého inputu jsem na konec jména dal číslo od jedné do počtu položek. Jako příklad třeba *goly1, zkarty1...* V kódu to vypadá takto *NAME="gol<?echo \$x?>"*, kde *\$x* je číslo od jedné do počtu položek a je pro každý řádek tabulky zvyšováno o jedno. Ve skriptu, který tato data zpracovává jsou poté tyto jednotlivé položky vloženy do pole. Vždy odpovídajícím způsobem, takže například hodnota z *\$\_POST['gol1']* je uložena do pole si indexem jedna *\$gol[1]*. Takže máme data seřazena stejně, jako byla vypsána ve formuláři. Poté pomocí *for* cyklu přistupujeme k určeným indexům v polích a aktualizujeme jednotlivé údaje. Děje se tak hráč po hráči, tak jak byli vypsáni ve formuláři.

Na velmi podobném principu potom funguje, také přidávání hráčů do jednotlivých týmů klubu.

#### 4.3 CSS styly a grafika

Pro zobrazení používám CSS styly. Jejich hlavní funkce je na rozdělení stránky na čtyři části - DIVy. Horní část, ve které se zobrazuje logo klubu. Na levou část, kde je hlavní menu. Pravou část je přihlašovací dialog a nejbližší zápasy. A na střední část, kde se zobrazují vybrané informace. Dále už definují jen barvy pozadí jednotlivých divů, barvu textu a různé typy odkazů. Grafika celého systému je poměrně jednoduchá. Snažil jsem se hlavně o přehlednost a intuitivnost. Jinak jsou mé schopnosti, co se designu týče, omezené a lepší vzhled se mi nepodařilo vymyslet.

#### 4.4 SQL

Pro propojení internetového rozhraní a databáze jsou logicky používány SQL příkazy, přes které se dá MySQL kompletně ovládat. Připojení k databázi:

```
$db = mysql_connect('localhost:/var/run/mysql/mysql.sock', 'xmares05', 'xxxxxxx');  
if (!$db) die('nelze se pripojit '.mysql_error());  
if (!mysql_select_db('xmares05', $db)) die('database neni dostupna '.mysql_error());
```

Místo xxxxxxxx je v kódu heslo, které tady nechci zveřejňovat. Po připojení potom pomocí php funkce MySQL\_Query() zadávám jednotlivé SQL příkazy. V programu jsou použity jen základní příkazy SELECT, INSERT, UPDATE, DELETE, ALTER TABLE, DROP TABLE

## **4.5 K programování**

Tento projekt bylo mé první setkání s PHP a první větší setkání s HTML a na kódu to bude nejspíš poznat. Postupem času jsem zjišťoval, že některé konstrukce, které jsem použil by se daly nahradit jinými, že některé části by bylo šikovnější řešit pomocí funkcí a některé věci by se daly udělat šikovněji a přehledněji, ale z časových důvodů jsem již nebyl schopen tyto nedostatky odstranit, ale funkčnost programu to nenarušuje. Spíše pozdější úpravy systému budou poněkud složitější než by tomu mohlo být.

## 5 Závěr

Mým cílem bylo vytvořit, co nejjednodušší a nejprůhlednější internetový informační systém. Zvláště se to týká části systému pro přihlášeného uživatele. Systém je vytvořen pro prezentaci fotbalového klubu, a proto jsem se zaměřil hlavně na tuto část. Myslím se, že je zpracována co se funkčnosti týče, velmi obstojně. Dále jsem zaměřil na správu článků v týmu, kterou jsem snažil udělat co nejužitečnější. Snažil jsem se držet současných trendů, ale vše jsem se snažil přizpůsobit požadavkům malého fotbalového klubu. Mé znalosti z tohoto okruhu jistě nejsou dokonalé a určitě jsou na projektu vidět určité nedostatky. Hlavně co se správy zápasů týče, kde mi výrazněji chyběla znalost z praxe a vycházel jsem jenom ze svých poznatků a znalostí. Dále pak zůstala nedokončená ekonomická část projektu. Opět jsem neměl dokonalé znalosti o ekonomickém fungování klubu, a proto zůstala tato část systému nedokončená, ale po nastudování ekonomických aspektů a fungování klubu může být v budoucnu snadno dokončena.

Z celkového pohledu, je ale systém použitelný a po přečtení uživatelské příručky, také snadno použitelný a myslím, že nároky malého fotbalového klubu by určitě splnil. Tedy až na grafiku, která je velmi jednoduchá a současné nároky již těžko snese. Ale to nebylo primárním cílem projektu.

Jak jsem již uvedl výše, tak systému by bylo možno přidat určitá rozšíření. Ať už se to týká zmiňované ekonomické části systému, tak by nebylo od věci také přidat diskusní fórum a knihu návštěv pro fanoušky. Také pod jednotlivými články by se mohla nacházet diskuse, aby se čtenáři mohli vyjádřit se svým názorem, třeba k článku nebo k události o níž článek píše. Použitelná by určitě byla nápověda pro přihlášeného uživatele. V tomto směru bych pak mohl pokračovat v tomto projektu jako na své diplomové práci. Také bych mohl v diplomové práci použít modernější a v současnosti velmi žádanou technologii asp .NET. A trochu bych se také chtěl zaměřit na grafiku a vzhled systému.

## 6 Literatura

- [1] Jiří Kosek: PHP tvorba internetových aplikací, Grada Publishing, spol s r. o., 1998
- [2] [www.wikipedia.org](http://www.wikipedia.org)

# 7 Přílohy

## 7.1 Uživatelská příručka

### 7.1.1 Uživatel reportér

#### **Práce se články**

Abyste mohli systém využívat jako reportér, musíte se nejprve přihlásit. Přihlašovací dialog se nachází v pravém horním rohu. Po vyplnění uživatelského jména a hesla již můžete systém využívat. V menu vám oproti nepřihlášenému uživateli přibyly dvě položky navíc. Jedná se o nový článek a rozepsané články.

Pokud chcete začít psát nový článek, kliknete na položku nový článek. Zde můžete článek psát. Samotný článek je zde rozdělen na několik částí, které je třeba vyplnit. Nejprve vyberete o jaký článek se jedná (historie, aktualita...). Poté zadáte nadpis. Ten se nesmí shodovat s jiným nadpisem již použitým. Pak je tu výběr obrázku, který se bude zobrazovat u článku. Vkládat se mohou jen formáty jpeg. Dále je tu úvod – část článku, která se bude společně s obrázkem zobrazovat uživateli na úvodní stránce. Poté již následuje samotný text článku. Na konci článku ještě vyplníte datum a podpis. Obě hodnoty jsou již vyplněny, ale máte možnost je měnit. Nakonec pomocí zaškrtnutí volby zobrazit, určíte zda se má článek zobrazovat nebo ne. Pokud tuto volbu nezaškrtnete, článek najdete po položkou Rozepsané články. Všechny položky

Pokud chcete najít v rozepsané články, tak klikněte v menu na položku Rozepsané články. Zobrazí se vám články podle data.

Pokud chcete upravit napsaný článek, tak na něj kliknete. Článek se vám zobrazí a pod ním najdete dvě volby. Upravit a smazat. Po kliknutí na volbu smazat, se článek vymaže z databáze. Pokud kliknete na upravit, tak se vám zobrazí stejná nabídka jako u nového článku, ale budou zde již uložené informace. Platí zde jinak stejná pravidla jako u nového článku.

### 7.1.2 Uživatel sportovní ředitel

Abyste mohli systém využívat jako sportovní ředitel, musíte se nejprve přihlásit. Přihlašovací dialog se nachází v pravém horním rohu. Po vyplnění uživatelského jména a hesla již můžete systém využívat. Po přihlášení se vám otevře stránka s nabídkou možností. Jedná se o položky Správa hráčů a trenérů, Správa zápasů, Správa týmů, Tabulka výsledků.

#### **Správa hráčů a trenérů**

Pokud chcete přidat nového hráče, klikněte na položku nový hráč. Zde můžete vyplnit jednotlivé údaje o hráči a vložit jeho fotografii. První čtyři údaje jsou povinné. Jedná se o jméno, příjmení, číslo a pozici, na které hráč hraje. Číslo musí mít hráč jedinečné jinak se mohou údaje shodovat.

Pokud chcete prohlížet a editovat údaje o hráčích, je tu položka Vypsání a editace hráčů. Otevře se vám výpis se všemi hráči. Můžete si je zobrazovat podle týmů nebo jména, čísla či příjmení. Po kliknutí na vybraného hráče se vám zobrazí detaily o něm. Pod těmito údaji najdete

dvě nabídky. Upravit a smazat. Po kliknutí na smazat, bude hráč vymazán z databáze. Po kliknutí na upravit se vám zobrazí stejná nabídka se stejnými pravidly jako u nového hráče.

Pokud chcete přidat nového trenéra, klikněte na položku nový trenér. Zde můžete vyplnit jednotlivé údaje o trenérovi a vložit jeho fotografii. Kromě fotografie jsou ostatní údaje povinné.

Pokud chcete prohlížet a editovat údaje o hráčích, je tu položka Vypsání a editace trenérů. Otevře se vám výpis se všemi trenéry. Vedle jejich jména je položka smazat, pomocí, které můžete trenéra smazat. Po kliknutí na vybraného trenéra se vám zobrazí stejná nabídka se stejnými pravidly jako u nového trenéra.

### **Správa zápasů**

Pokud chcete přidat nový zápas do rozpisu, tak klikněte na položku nový zápas do rozpisu. Zde vyplníte, který tým klubu zápas bude hrát, jeho soupeře, zda se bude utkání hrát doma či venku a datum utkání a sezónu. Všechny údaje jsou povinné.

Pokud si chcete prohlédnout nebo upravit rozpis, tak je tu položka vypsání a editace rozpisu. Zde se vypíše jednotlivé zápasy podle data. Vedle nich je položka smazat, která smaže daný zápas. Po kliknutí na zápas se zobrazí stejná nabídka se stejnými pravidly jako u nového zápasu do rozpisu.

Pokud chcete zadat výsledek už odehraného utkání nebo upravit již zadaný, klikněte na vložení a editace již odehraných zápasů. Vypíše se vám dvě tabulky. Jedna s již odehranými zápasy a jedna s rozpisem utkání. Po kliknutí na zápas z rozpisu, můžete zadat jeho výsledek a průběh a také aktualizovat údaje hráčů, kteří utkání odehráli. Pokud hráli, zatrhnete možnost hrál. Dále pak vyplníte počet branek a karet, které hráč vsítil či obdržel. Pokud nezádáte nic, tak to systém bere jako byste zadali nulu.

Pokud chcete smazat či upravit odehraný zápas, tak je tu první tabulka. Po kliknutí na zápas můžete upravovat údaje. Všechny položky jsou povinné.

### **Správa týmů**

Pokud chcete vytvořit nový tým, klikněte na položku nový tým. Zde zadáte jméno týmu a vyberete jeho trenéry a hráče. Trenéry z rolovacího menu a hráče pomocí zaškrtnutí. Jméno týmu je povinná položka a musí být jedinečná.

Pokud chcete tým smazat nebo upravit, tak klikněte na položku vypsání a editace týmů. Zde se vypíše jednotlivé týmy. Vedle nich položka pomocí, které je můžete smazat. Po kliknutí na vybraný tým můžete měnit jeho jméno i trenéry a hráče. Pravidla jsou stejná jako u nového týmu.

### **Tabulka výsledků**

Pokud chcete změnit tým, který má tabulku v pravém sloupci stránky, vyberte jej z roletového menu a dejte uložit.

Pokud chcete přidat tým do tabulky, tak klikněte na přidat nový tým do tabulky výsledků. Zde zadáte jméno týmu. Zde můžete týmy i mazat

Pokud chcete upravit pořadí týmů a jejich body, tak klikněte na upravit body a pořadí v tabulce. Zde se vám vypíše jednotlivé týmy a jejich body a pořadí, které můžete měnit.



### 7.1.3 Uživatel administrátor

Abyste mohli systém využívat jako administrátor, musíte se nejprve přihlásit. Přihlašovací dialog se nachází v pravém horním rohu. Po vyplnění uživatelského jména a hesla již můžete systém využívat. Po přihlášení se vám otevře stránka s nabídkou možností. Jedná se o položky Správa uživatelů, Nastavení systému, Sportovní menu, Ekonomické menu.

#### **Správa uživatelů**

Pokud chcete přidat nového uživatele, tak klikněte na položku nový uživatel. Zde zadáte jeho jméno, příjmení, login a mail. Všechny položky jsou povinné. Heslo se uživateli samo vygeneruje a pošle na vámi zadaný mail.

Pokud chcete měnit údaje o uživateli, tak klikněte na položku vypsání a editace uživatelů. Zde se vám vypíše tabulka se všemi uživateli. Máte zde možnost uživatele smazat. Po kliknutí na vybraného uživatele můžete upravovat jeho údaje. Zde je možné upravit i jeho heslo. Všechny položky až na heslo jsou povinné.

#### **Nastavení systému**

Pokud chcete změnit aktuální sezónu, tak ji vyberte z roletového menu a dejte změnit. Od teď to bude aktuální sezóna.

Pokud chcete změnit údaje o klubu, tak klikněte na změnit údaje na stránce o klubu. Zde můžete změnit adresu, mail, telefon a logo klubu, které se zobrazuje v horní části stránky.

### 7.1.4 Změna hesla

Abyste si mohli změnit heslo, musíte se nejprve přihlásit. Přihlašovací dialog se nachází v pravém horním rohu.

Pokud si chcete změnit heslo, klikněte na odkaz změnit heslo pod tlačítkem odhlásit vlevo nahore. Zde zadáte nejprve staré heslo a pak dvakrát nové. Pokud bude vše sedět, tak vám bude změněno.

## 7.2 Jednotlivé soubory a jejich popis

admin.php – menu administrátora

aktuality.php – archiv článků jenom z rubriky aktuality

archiv.php – archiv všech článků

clanek.php – vypsání určeného článku

ekonom.php – menu uživatele ekonomického ředitele

historie.php – vypíše články z rubriky historie

hraci.php – obsahuje formulář pro vložení nového hráče

hraciV.php – vypíše detaily o zvoleném hráči

index.php – úvodní stránka  
insert.pho – skript pro vložení článku  
insertH – skript pro vložení nového hráče  
insertR – skript pro vložení zápasu do rozpisu  
insertT – skript pro vložení nového trenéra  
insertTab – skript pro vložení týmu do tabulky pořadí  
insertTym – skript pro vložení vložení nového týmu  
insertU – skript pro vložení nového uživatele  
insertZ2 – skript pro vložení nového zápasu  
klub.php – formulář pro vložení nového článku  
login – skript pro přihlášení uživatele  
logout – skript pro odhlášení uživatele  
nastavS – menu pro nastavení systému  
nastavTab – skript pro nastavení týmu pro tabulku výsledků  
oklubu.php – výpis údajů o klubu  
report.php – archiv článků jen z rubriky reportáže  
rozepsané.php – archiv rozepsaných článků  
rozpis.php – formulář pro vložení zápasu do rozpisu  
smazatC – skript pro smazání článku  
smazatH – skript pro smazání hráče  
smazatR – skript pro smazání zápasu z rozpisu  
smazatT – skript pro smazání trenéra  
smazatTab – skript pro smazání týmu z tabulky výsledků  
smazatTym – skript pro smazání týmu  
smazatU – skript pro smazání uživatele  
smazatZ – skript pro smazání zápasu  
sport.php – menu sportovního ředitele  
spravaH.php – menu pro správu hráčů a trenérů  
spravaTymu.php – menu pro správu týmů  
spravaU.php – menu pro správu uživatelů  
spravaZ.php – menu pro správu zápasů  
tabulka.php – formulář pro vložení týmu do tabulky zápasů  
tabulkaV.php – menu pro tabulky zápasů  
trener.php – formulář pro vložení nového trenéra  
tymy.php – formulář pro vložení nového týmu  
upravitC.php – formulář pro úpravu článku  
upravitC2.php – skript pro úpravu článku  
upravitH.php – formulář pro úpravu hráčů  
upravitH2.php – skript pro úpravu článku  
upravitK.php – formulář pro úpravu údajů o klubu  
upravitK2.php – skript pro úpravu údajů o klubu  
upravitR.php – formulář pro upravení zápasu v rozpisu  
upravitR2.php – skript pro upravení zápasu rozpisu  
upravitT.php – formulář pro úpravu trenéra  
upravitT2.php – skript pro úpravu trenéra

upravitTab.php – formulář pro úpravu bodů a pořadí v tabulce zápasů  
upravitTab2.php – skript pro úpravu bodů a pořadí v tabulce zápasů  
upravitTym.php – formulář pro úpravu týmů  
upravitTym2.php – skript pro úpravu týmů  
upravitU.php – formulář pro úpravu uživatele  
upravitU2.php – skript pro úpravu uživatele  
upravitZ.php – formulář pro úpravu zápasu  
upravitZ2.php – skript pro úpravu zápasu  
Vlastik.css – kaskádový styl  
vypisH.php – výpis soupisky  
vypisH2.php – skript pro určení podle čeho se bude řadit  
vypisR.php – vypíše rozpis  
vypisT.php – vypíše trenéry  
vypisTym – vypíše týmy  
vypisU.php – vypíše uživatele  
vypisZ.php – vypíše detaily o zápase  
zapas.php – formulář pro přidání zápasu  
zapasy.php – vypíše zápasy  
zapasy2.php – skript, pro výběr, kterého rýmu zápasy budou zobrazeny  
zapasy3.php – skript pro výběr sezóny, která bude zobrazena  
zmenaH.php – formulář pro změnu hesla  
zmenit\_heslo.php – skript pro změnu hesla  
zobrazitH.php – skript pro zobrazení fota u hráče  
zobrazitT.php – skript pro zobrazení fota trenéra  
zobrazit\_logo.php – skript pro zobrazení loga klubu  
zobrazit\_obrázek.php – skript pro zobrazení obrázku u článku  
zobrazRozepsane.php – zobrazí rozepsané články