

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Webové aplikace – 2. projekt

Implementační dokumentace

Tým 08 – Organizátor schůzek

Lukáš Havlíček	(xhavli46)
Jakub Sadílek	(xsadil07)
Jaroslav Katrušák	(xkatru00)
Filip Weigel	(xweige01)

1 Zadání

Implementujte webovou aplikaci vhodnou pro organizaci schůzek a jiné domlouvání termínů událostí (akcí). Aplikace umožní libovolnému návštěvníkovi vygenerovat novou událost (akci). Uživatel bude mít možnost vytvořit několik alternativních termínů konání události (akce), např. podle dnů, nebo hodin. Aplikace by uživateli měla umožnit automaticky generovat termíny podle zadaných kritérií (např. každou celou hodinu v pracovní době uvedených dnů, pracovní dobu stanovuje uživatel).

Aplikace pro nově vytvořenou událost vygeneruje náhodný řetězec (řetězce) o vhodné délce pomocí kterého bude možné podávat organizátorovi události zpětnou vazbu k termínu události a prohlížet výsledky. Aplikace vygeneruje URL obsahující výše uvedené řetězce pro snadný přístup k hlasování a výsledkům hlasování.

Očekává se, že organizátor akce bude výše uvedené URL distribuovat zájemcům o aukci vlastními silami mimo implementovanou aplikaci.

Aplikace automaticky smaže všechna data k události 14 dnů po nastaveném termínu konání akce.

2 Implementace

Projekt je implementován v jazycích HTML, PHP a Javascript. Frontend je implementován pomocí CSS a frameworku Bootstrap.

2.1 Backend

Úvodní stránka je implementována v souboru `index.php`. Manuálně si lze přidat termín, nebo lze termín generovat automaticky podle zvolených kritérií, kterými jsou počáteční datum a čas, a koncový datum a čas včetně rozestupů mezi jednotlivými termíny. Dále při automatickém generování lze zvolit pro které dny v týdnu si přeje uživatel generovat termíny. Standartně jsou zvoleny pracovní dny v týdnu.

Automatické generování termínu je implementováno pomocí cyklů `for`. Nejdříve je pomocí cyklu zjištěno, které dny jsou vybrány pro automatické generování a poté následuje druhý cyklus, který postupně od počátečního času k času koncovému generuje termíny s požadovaným rozestupem.

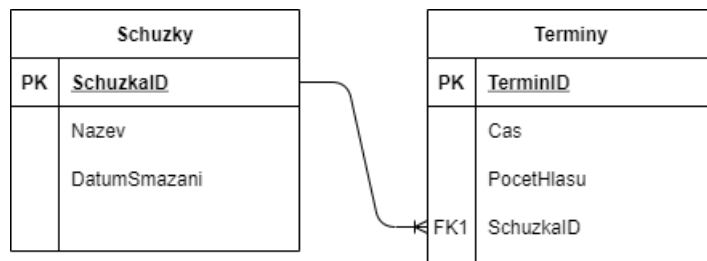
Po vygenerování termínu (manuálně/automaticky) se uživateli termíny zobrazí. Nevyhovující termíny lze editovat, nebo odebrat. Odebrání a editace termínu je realizována pomocí `javascriptu`, při kterém je využito i `jquery` pro výběr prvků z samotné stránky. Poté je nutné vyplnit název události a následně je možné vygenerovat událost pomocí tlačítka `založit hlasování`. Backend přijme informace o události a termínech, přidělí ji ID, vytvoří nové záznamy v databázi událostí a termínů k dané události.

`hlasovani.php?id=x`, kde `x` je ID události. Na tento link je uživatel přesměřován po založení hlasování. Backend dle požadavku GET vybere termíny z dané události a termíny jsou zobrazeny uživateli. Uživatel může zvolit termíny, pro které chce hlasovat - implementovány pomocí `checkboxu` - a odešle je pomocí metody POST. Následně backend POST přijme a pro každý zvolený termín inkrementuje v databázi počet hlasů o jeden hlas. Stejněmu uživateli je zamezeno hlasovat vícekrát pro danou událost.

Pro uživatele není hlasování povinné a pokud si nepřeje hlasovat, může se rovnou podívat na výsledky hlasování pomocí tlačítka `Výsledky`. Po stisknutí tlačítka je uživatel přesměřován na `vysledky.php?id=x`, kde `x` je ID události, backend pomocí metody GET dostane informace o požadované události a obratem zobrazí uživateli aktuální výsledky.

2.2 SQL databáze

Databáze je implementována v jazyce SQL. Skript pro vytvoření databáze je obsažen ve souboru `SQL.txt` a konfigurace pro připojení k databázi je obsažena v souboru `config.php`. Databáze se skládá ze dvou tabulek. První tabulka `Schuzky`, která obsahuje primární klíč `SchuzkaID`, dále název schůzky a datum smazání schůzky (14 dní po uplynutí schůzky). Mazání je realizováno pomocí databázového eventů. Druhá tabulka obsahuje vygenerované termíny k právě jedné schůzce. Je složena z `TerminID`, který je primárním klíčem, časem pro daný termín, dále počtem hlasů a cizím klíčem `SchuzkaID`, který odkazuje do tabulky `Schuzky`. Při smazání schůzky dojde k automatickému smazání všech termínů, které byly se schůzkou provázány pomocí `on delete cascade`



Obrázek 1: Schéma SQL databáze

2.3 Frontend

Frontend je implementován pomocí HTML a pro vzhled je použit `css` předpis, který se nachází v souboru `style.css`. V souboru `style.css` jsou specifikovány barvy, velikosti písma, barvy písma a podobně. Rozložení stránky je řešeno pomocí frameworku `bootstrap` ve verzi 4.3.1 a pro jeho správnou funkcionálnitu je třeba mít nalinkovaný `javascript` ve verzi 5.0.0 a výše. `Bootstrap` byl zvolen z důvodu rychlé realizace rozložení stránky a taktéž protože jsme se s ním mohli setkat i v jiných projektech. Frontend je implementován v souborech `index.php`, `hlasovani.php` a `vysledky.php`.

3 Závěr

Při vypracovávání projektu jsme nenarazili na žádné větší obtíže. Komunikace probíhala pomocí serveru `Discord`, na kterém mohl každý člen diskutovat řešení projektu a připomínky k projektu. Pro verzování projektu jsme zvolili server `GitHub`¹. V projektu se podařilo splnit všechny požadavky zadání a projekt lze tedy považovat za úspěšně dokončený. vypracovaný projekt je dostupný na² a³

¹www.github.com/filaseek/wap

²<http://www.stud.fit.vutbr.cz/~xhavli46/wap/>

³<http://www.wap.borec.cz/>