

# PHP és MYSQL BEADANDÓ

## Hozzáférés:

felhasználónév: **blang\_DF9YEV** és jelszó: **8hu5NIJh**

URL: [http://portalbce.hu/DF9YEV/OutOfOffice\\_HW/Login.php](http://portalbce.hu/DF9YEV/OutOfOffice_HW/Login.php)

(GIT: [https://github.com/Necridus/OutOfOffice\\_HW.git](https://github.com/Necridus/OutOfOffice_HW.git))

## Szerzők és fejlesztés során elvégzett feladataik:

- **Budai Bianka**
  - felhasználókezelő admin felület (MYSQL scriptek)
    - új felhasználó hozzáadásának lehetősége
    - meglévő felhasználó és annak összes szabadságkérelmének törlése
    - felhasználó egyes adatainak módosítása, mentése
    - hibakezelés, űrlapvalidálás
    - design
- **Kimmel Martin Ádám**
  - Bejelentkező felület
    - bejelentkezési logika implementálása admin és egyszerű user alapján
    - hibakezelés
  - fő admin oldal
  - szabadságkezelő admin felületen
    - szabadságkérelmek megjelenítése elkülönítve a már korábbiakat az eljövendőktől
    - azok státuszának módosítása (MYSQL scriptek)
    - feltételes formázás JavaScript segítségével
  - Bootstrap projekthez hozzáadása és design fő vonalainak megvalósítása
  - vissza és kijelentkező gombok
  - oldalak nem megfelelő jogosultsággal való hozzáféréseinek letiltása
  - adatbázistáblák létrehozása
  - email küldés implmentálása
- **Radics Attila**
  - Fő felhasználói felület
  - felhasználó szabadságainak megjelenítő felülete (MYSQL scriptek)
    - elkülönítve a már korábbiakat az eljövendőktől
    - törlés lehetősége
  - Új szabadság hozzáadására alkalmas felület (MYSQL scriptek)
    - megfelelő hibakezelés, űrlapvalidálás
    - email küldés

## Közös munka:

- Az alapötletet közösen találtuk ki, melyik felhasználó milyen jogosultságokat kapjon, hány oldal legyen, melyen mit lehessen látni és csinálni
- Az adatbázis felépítésének megtervezése is közös munka volt
- GIT használat

## Weboldal ismertetése:

Weboldalunkon egy szabadságkezelő rendszer érhető el. Egyszerű userként lehetőség van a meglévő szabadságainkat megtekinteni, valamint új szabadságkérelmet létrehozni. Adminként bejelentkezve a szabadságkérelmek státuszát lehet módosítani, valamint meglévő felhasználókat törölni, módosítani és újakat létrehozni.

**Teszteléshez egy admin felhasználó: MartinAdmin (jelszó: 123)**

**Teszteléshez egy User: Martin (jelszó: 123)**

## Követelmények teljesülése

- Az alkalmazás tartalmazzon egy összetettebb űrlapot, melyet feldolgoztok**

Alkalmazásunk több űrlapot is tartalmaz. Az első a bejelentkező felületen van text és password típusú input fieldekkel. Ezen kívül a felhasználók új szabadságot tudnak kiírni egy olyan form segítségével, amiben a szabadság kezdő- és végdátumát adhatják meg. Adminként lehetőség van szabadságkéresek elbírálására, ami során egy select-ből választhatják ki a kívánt státuszt. Adminként lehetőség van továbbá új felhasználót hozzáadni szintén egy form segítségével, ahol a username-et és a nevet text típusúval tudja megadni az új felhasználó nevét és felhasználónevét, az email címet pedig email típusúval.

- Alkalmazatok menetkövetést**

A menetkövetést a bejelentkező felhasználó esetén a session tömb használatával alkalmazzuk (username, loggedin és isadmin változók). Űrlapfeldolgozásnál érzékeny adatok hiányában rejtett űrlapelemeket használtunk.

- Használjatok adatbázist**

Az adatbázis a következő módon épül fel:

### Users (tábla)

RecordID	ID	Email-Address	UserName	Password	Name	JobTitle_FK	IsAdmin	ValidFrom	ValidTo
int	int	varchar(50)	varchar(50)	varchar(50)	varchar(50)	int	boolean (tinyint(1))	DateTime	DateTime

### Requests (tábla)

ID	UserID	StartDate	EndDate	Status	ValidFrom	ValidTo
int	int	date	date	varchar(50)	DateTime	DateTime

### JobTitles (tábla)

ID	JobTitle	ValidFrom	ValidTo
int	varchar(50)	DateTime	DateTime

Az oldalon több esetben is lekérdezzük a szabadságkérelmeket, ugyanakkor a User nevét is tartalmazzák rendszeresen ezek a táblázatok, így join-t használunk.

A ValidFrom és ValidTo a rekord érvényességét jelöli, így valósítottuk meg a historikus adattárolást. Ha egy szabadságkérelmet törölni szeretnénk, a ValidTo értékébe beírjuk a művelet időpontját, majd új sort hozunk létre az adatbázisban, így fejlesztőként követhetjük a recordok történetét. Módosítások esetén sem közvetlen azt a recordot módosítjuk, hanem ValidTo-t a jelenlegi időre állítjuk, majd új sort hozunk létre. Minden ValidTo = NULL record valid, ezeket jelenítjük meg a felületen. A ValidFrom a record létrehozását jelöli.

- Az alkalmazást több, eltérő jogosultságú felhasználó tudja használni + Az admin jogosultságú felhasználó tudjon felvenni/törölni felhasználót**

Az alkalmazást két jogosultság szerint lehet használni: Admin és User.

Userként lehetőség van a már meglévő szabadságokat megtekinteni, még nem elmúltat törölni, valamint új szabadságigényt benyújtani.

Adminként lehetőség van felhasználók kezelésére (új felhasználó felvétele, meglévő adatainak módosítása, meglévő törlése), valamint a benyújtott szabadságkérelmek elbírálására.