Műszaki rendszerterv

A kívánt rendszer specifikációi alapján a megvalósítás az alábbi szempontok szerint történjék:

Szerverkörnyezet

A biztonság magas prioritású, ezért linux operációs rendszer alatt php-fpm és nginx webszerver legyen a kiszolgáló környezet, a fejlesztéskor elérhető legfrissebb stabil verzióval. Adatbáziskezelőt illetően licencdíj, könnyű kezelhetőség és megbízhatósági megfontolások alapján PostgreSQL vagy MySQL kerül telepítésre.

Mivel a végleges fizikai környezet nem ismert, a hordozhatóság érdekében Docker konténerben legyen elhelyezve a teljes, önállóan működő rendszer. A konténer létrehozásakor hozza létre automatikusan az adatbázis-sémát. A Docker beállítófájlok és az alkalmazás forráskód GIT verziókezelőbe legyen feltöltve.

Adatbázis-kiszolgáló

Törekedjünk az adatbázis séma létrehozásakor a tárolt eljárások és specifikus adattípusok mellőzésére, hogy esetleges jövőbeni migráció során a legkevesebb módosításra legyen szükség az alkalmazásréteg forráskódjában.

Szem előtt kell tartani azonban az adatok integritásának megőrzését, ezért:

- Legyen idegen kulcs a kívánságlista táblán a felhasználó tábla felé
- Legyen idegen kulcs a kívánság elem táblán a kívánságlista tábla felé
- Legyen duplikációt kizáró index a felhasználói azonosítóra
- Legyen duplikációt kizáró index a kívánságlista nevére felhasználónként

Alkalmazás-kiszolgáló

A biztonságot szem előtt tartva, kerüljük a komplex tartalomkezelő keretrendszereket, kerülendő a fejlesztés után kiderülő biztonsági réseket. Ezek frissítése, verzióváltása a hozzá készült kód stabilitást veszélyeztetheti. Az egyedi keretrendszer azonban nélkülözheti a bevált keretrendszerekben alkalmazott biztonsági technikákat, ezért a feladatmeghatározásban kiemelt szempontokon felül az alábbiakat implementáljuk:

- a HTTP fejlécekben lehetőleg fedjük el a PHP és NGINX verzióját, módosítsuk a Session ID nevét
- jelszó tárolása SHA2 algoritmussal, sózással
- a natív jelszóról már kliens oldalon történjék hash képzés, ráadásul NONCE használatával, hogy minden bejelentkezéskor más leképzett jelszó jelentkezzék az adatforgalomban
- a böngésző ne jegyezze meg a bejelentkezési ablakon a felhasználói azonosítót és jelszót
- korlátozzuk a sikertelen bejelentkezések számát
- adminisztrátor saját magát ne törölhesse, ne módosíthassa
- CORS alkalmazása
- index.php átnevezése

A feladat összetettségéhez képest elegendő egyedi, egyszerű útvonalvezérlőt írni - nem szükséges robusztus MVC alkalmazása. A HTML megbízható rendereléshez e projektben a Smarty sablonkezelő kielégítő, illetve egy kész BootStrap template használunk fel.

Két szintű konfigurációs beállításokat alkalmazzunk, egy felső szintű, környezeti paramétereket (melyek az üzemeltetési környezetre vonatkoznak, pl. adatbázis elérése), illetve egy második szintű, a konkrét feladathoz kapcsolódó paraméterek.

Maga a webes elérés a /*kivansag*/ context alatt legyen elérhető, amennyiben reverse proxy mögé kerül az alkalmazás. A későbbi bővíthetőség érdekében lehetőleg kevés változtatással további context is bevezethető legyen.

Az átláthatóság érdekében törekedni kell az osztályok használatára a kisebb komplexitású feladategységeknél is, illetve PHPDOC formátumú kommentek alkalmazását, hogy a forráskód dokumentációját automatikusan lehessen generálni.

HTTPS protokoll használata elengedhetetlen, ezt azonban jelen scope nem tartalmazza, az üzemeltetés feladata lesz.