*Kritériumok: (min 50% -> 20 pont)*

1. **(4 pont) KÖTELEZŐ ELEM**

Összefoglalva, ez a feladat arra kéri a fejlesztőt, hogy:

* hozzon létre egy dinamikusan vezérelt weboldalt a Front Controller tervezési minta felhasználásával, amelynek adatait és menüit egy konfigurációs fájlból tölti be,
* és HTML5-öt használ vízszintes menü létrehozásához,
* valamint tartalmaz egy linket egy az eredeti szervezeti weboldalára.

Részletek:

A Front Controller egy tervezési minta, amely egy központi elemet, a "front controllert" használja egy webalkalmazásban vagy keretrendszerben. Ennek az a célja, hogy centralizálja a kérések feldolgozását és irányítását, így az alkalmazás struktúrája átláthatóbb lesz, és könnyebben kezelhetőek lesznek a kérések.

A Front Controller tervezési minta lényege, hogy minden HTTP kérés egy központi vezérlőhöz irányítódik, ami dönt arról, hogy melyik konkrét kódrészlet felelős a kérés kezeléséért. Ezáltal lehetőség van közös logikai műveletek, például az autentikáció vagy az engedélyek ellenőrzése, egyszerűsítésére és központosítására.

Ha PHP kóddal dinamikussá teszed a honlapot, akkor ez megfelel a Front Controller tervezési minta kritériumainak. A PHP kód lehetővé teszi számodra, hogy dinamikusan generálj HTML-t és kezeld a kéréseket, ami lehetővé teszi a Front Controller minta implementálását.

A követelmények között szerepel, hogy minden menüpont esetén az index.php fájlt kell meghívnod, ami azt jelenti, hogy az index.php lehet a Front Controller, ahol a kéréseket kezeled és a megfelelő oldalakat töltöd be.

A PHP lehetőséget biztosít a menünevek és közös adatok olvasására egy konfigurációs fájlból is, amit egy tömbben tárolhatsz. Emellett a PHP használata lehetővé teszi a dinamikus tartalom megjelenítését és kezelését az oldalon.

Tehát a PHP kóddal való dinamikusság lehetővé teszi, hogy megvalósítsd a Front Controller tervezési mintát a honlapodban. Fontos azonban, hogy megfelelő struktúrát és logikát alkalmazz a kódodban annak érdekében, hogy a Front Controller minta hatékonyan működjön és a honlap jól skálázható legyen.

2. **(4 pont)**

* Bővítse a választott főoldalt két videóval, egyet saját könyvtárából,
* és egyet szolgáltatótól (pl.Youtube), a saját könyvtárában lévő video ne legyen több mint 5 mp, a méretkorlát miatt
* Az oldalra tegyen egy Google térképet is, ami megmutatja a választott szervezet címét.

A további oldalakat a következő pontok szerint alakítsa ki:

3. **(4 pont)**

* Készítsen egy harmadik oldalt képek, képgaléria tárolására.
* Legyen lehetőség új képek feltöltésére.
* Képfeltöltést csak bejelentkezett felhasználó tehet meg.

4. **(4 pont) [**Ehh, ez nehéznek hangzik.]

* A negyedik oldalon legyen egy kapcsolat űrlap, amelynek segítségével e-mailt lehet küldeni az oldal tulajdonosa számára, **de az e-mail küldése helyett jelenítse meg az adatokat egy új (ötödik) oldal tartalmaként.**
* Ellenőrizze megfelelően az űrlap helyes kitöltését. Az ellenőrzést végezze el kliens- és szerveroldalon is (JavaScript, PHP). **A HTML kódban ne ellenőrizze az úrlap adatait**.
* Az elküldött Űrlap adatokat ezen kívül mentse le adatbázisba is.

5. **(4 pont) [**Feltétele a 4. bekezdés ?]

* Tegye lehetővé megtekinteni egy hatodik oldalon táblázatban az előző pontban elküldött üzeneteket az adatbázisból fordított időrend szerint (a legfrissebb legyen elől).
* Írják minden üzenethez a küldés idejét és az üzenetküldő nevét. Ha nem bejelentkezett felhasználó írta, akkor: "Vendég"

6. **(4 pont)**

* Regisztráció, Belépés, Kilépés. Bővítsük a honlapot „Belépés” és „Kilépés” menüponttal a következők szerint:

a) A „Belépés” menüpont akkor látható, ha nincs bejelentkezve a felhasználó.

b) A „Kilépés” menüpont akkor látható, ha be van jelentkezve a felhasználó.

c) A „Belépés” menüpontra kattintva feljön egy oldal, ahol lehet bejelentkezni vagy regisztrálni.

d) Regisztrációkor nem léptetjük be automatikusan a felhasználót.

e) A rendszer fejlécen jelenítse meg a bejelentkezett felhasználót, ha be van lépve, a következő formában: Bejelentkezett: Családi\_név Utónév (Login\_név)

7. **(4 pont)**

* Alkalmazását töltse fel és valósítsa meg Internetes tárhelyen is. Bármelyik tárhely-szolgáltatót használhatja. (**Kötelező elem**! **A működés ez alapján lesz javítva**)

8. **(4 pont)** (**Kötelező elem**! **A forrás ez alapján lesz javítva**)

* Használja a GitHub verziókövető rendszert (<https://github.com/>).
* Ne csak a kész alkalmazást töltse fel egy lépésben, hanem a részállapotokat is legalább 5 lépésben.
* A GitHub-on a projektmunka módszert alkalmazzák: látszódjék, hogy a csoport tagjai melyik részt készítették el.

**A GIT-en saját nevet válasszanak, ami alapján be lehet azonosítani, hogy ki mit töltött fel.**

9. **(4 pont) KÖTELEZŐ ELEM**

* **Készítsen legalább 10 oldalas dokumentációt (PDF formátumban)**, amiben leírja az alkalmazás működését képernyőképekkel.
* A dokumentációban adja meg a tárhely és a GitHub projektjének URL címét
* **és írja le részletesen, hogy az előző feladatpontokat az alkalmazásban hogy és hol valósította meg**. Fontos, hogy ez utóbbit leírja a dokumentációban, mert a feladat ez

alapján lesz javítva.

* A dokumentációban adja meg az internetes tárhely és adatbázis belépéséhez szükséges FTP és URL címet, felhasználónevet és jelszót, ami szükséges a megoldás ellenőrzéséhez, valamint a GitHub címét.

**Be kell adni** a **Neptun MeetStreet**-en keresztül a dokumentációt.

Akik közösen csinálják, mindenki adja be a megoldást.

A forrásokat feltöltve a helyi web szerver (XAMPP) munka-könyvtárába az alkalmazás működőképes legyen.