

SZAKMAI BESZÁMOLÓ

Szakmai Gyakorlat I.

THE_00761_0_N_A

Orosz Kristóf

2025

Tartalomjegyzék

Bevezetés	3
A fogadó szervezet bemutatása	3
A szakmai gyakorlat során végzett tevékenységek bemutatása	4
A fejlesztési szakaszok részletes bemutatása	5
Tervezési szakasz	6
Backend fejlesztési szakasz	6
Frontend fejlesztési szakasz	7
Frontend – Backend integrációs szakasz	7
Tesztelési szakasz	8
Dokumentációs szakasz	8
A projekt során szerzett új ismeretek	9
Megoldott problémák	9
Reflexió a szakmai fejlődésre	10
Összegzés	11

Bevezetés

A 160 órás szakmai gyakorlatom során lehetőségem nyílt arra, hogy valódi fejlesztési projektben vegyek részt a simpleSoft Kft.-nél, amely során különféle fejlesztési szakaszokban próbálhattam ki magam. A gyakorlat során nem csupán megfigyelőként vettem részt a munkában, hanem aktívan hozzájárultam egy valós rendszer létrehozásához.

A projektemet a szakmai konzulensem segítségével készítettem el. A rendszer fejlesztését háromfős csapatban valósítottuk meg. A csapat tagjai én magam voltam, a szakmai konzulensem Dédesi Péter Úr, valamint a simpleSoft Kft. egyik munkatársa, aki a határidők betartásában segített és koordinálta a feladatok ütemezését.

A gyakorlat során nemcsak a technikai tudásom bővült, hanem fejlődtek a kommunikációs készségeim és az időgazdálkodásom is. Olyan környezetben dolgozhattam, ahol ösztönöztek a kreativitásra és a problémaorientált gondolkodásra. A gyakorlati ismeretek megszerzése mellett lehetőségem volt megtapasztalni, milyen fontos a projektmenedzsment és a rendszeres egyeztetések egy fejlesztési folyamat során. A beszámoló célja, hogy részletesen bemutassam a szakmai gyakorlatom során végzett munkát, valamint a cégnél szerzett tapasztalatokat.

A fogadó szervezet bemutatása

A szakmai gyakorlatot a simpleSoft Kft. -nél teljesítettem, amely főként egyedi szoftverfejlesztéssel, webes és mobilalkalmazások készítésével, valamint vállalati informatikai rendszerek kialakításával foglalkozik. A cég központja Miskolcon található, és kis létszámú, de rendkívül szakértő csapattal dolgozik. A vállalat több éves tapasztalattal rendelkezik az üzleti informatikai megoldások terén, és megbízható partnerként van jelen a magyar informatikai piacon.

A simpleSoft célja olyan rugalmas, egyedi igényekre szabott alkalmazások fejlesztése, amelyek segítik a partnerek hatékonyabb működését. A vállalat fejlesztői elsősorban PHP, JavaScript, C# és Java nyelveket használnak. A munkavégzés során nagy hangsúlyt fektetnek a minőségbiztosításra, a tesztelésre és az ügyfélorientált gondolkodásra.

Én is egy teljes projektet kaptam, amely során a cégnél bevett fejlesztési módszertanokat követve dolgoztam. A projekt elkészítése agilis szemlélet szerint zajlott, rendszeres megbeszélésekkel és folyamatos visszacsatolásokkal. A simpleSoft szakemberei készségesen segítettek a problémák megoldásában, és rendszeres visszajelzéseket adtak a munkámról. A cég inspiráló szakmai környezetet biztosított, amelyben sokat tanulhattam a gyakorlatias szemléletről és a csapatomunkáról. Emellett betekintést nyertem abba is, hogyan működik egy fejlesztőcég a mindennapokban, hogyan kezelik a projekteket és milyen szerepe van az ügyfélkapcsolatoknak.

A szakmai gyakorlat során végzett tevékenységek bemutatása

A szakmai gyakorlat alatt egy időpontfoglaló webalkalmazás fejlesztésén dolgoztam, amely lehetővé teszi különböző szolgáltatások időpont alapú lefoglalását adminisztrátor és felhasználói oldalon egyaránt. A projektet az elejétől kezdve építettem fel: kezdve a tervezéstől, az adatbázis kialakításán és backend funkciók megírásán át a frontend fejlesztésig és a tesztelésig. A munkát a szakmai konzulensem előre egyeztetett munkaterv alapján végeztem, amelynek részfeladatait fokozatosan valósítottam meg.

A feladatok között szerepelt az adatmodellezés, az ER-diagram és az SQL adatbázis létrehozása, valamint a szükséges PHP API-k implementálása. Ezeket fokozatosan kapcsoltam össze a frontend oldallal, ahol HTML, CSS és JavaScript használatával dinamikus felületeket készítettem. A rendszer tartalmazott regisztrációs és bejelentkezési lehetőséget, jogosultságkezelést különböző szerepkörök szerint, valamint az időpontok kezelése is teljes funkcionalitással megvalósult: létrehozás, módosítás, törlés.

A Postman alkalmazás segítségével folyamatosan teszteltem a REST API-k működését, és szükség esetén javítottam a hibákat. A projekt során törekedtem az átlátható, strukturált kódírássra, valamint arra, hogy minden funkcióhoz világos, jól dokumentált logika tartozzon. Az elvégzett munkám minden szakaszát dokumentáltam. A projekt végére egy teljesen működőképes rendszert készítettem el a szakmai konzulensem segítségével, amely megfelelt Dédesi Péter Úr elvárásainak, és számomra is komoly fejlődési lehetőséget jelentett.

A fejlesztési szakaszok részletes bemutatása

A projekt fejlesztését hat fő szakaszra osztottuk fel: tervezés, backend fejlesztés, frontend fejlesztés, frontend-backend integráció, tesztelés és dokumentáció. A tervezési szakaszban elkészítettem az ER-diagramot, az adatbázis struktúráját és a képernyőterveket. Minden funkció működését előre dokumentáltam, hogy a fejlesztés során könnyen visszakereshető legyen, mit miért valósítok meg.

A backend fejlesztés során PHP nyelven REST API-kat készítettem. A konzulensem segítségével, akitől példákat is kaptam.

A frontend fejlesztés célja egy felhasználóbarát, reszponzív kezelőfelület kialakítása volt. HTML és CSS segítségével alakítottam ki az űrlapokat, táblázatokat és gombokat, míg JavaScript-tel kezeltem a felhasználói interakciókat. Minden adatbevitelt validáltam, és visszajelzést küldtem a felhasználónak a sikeres vagy sikertelen műveletekről.

A frontend és backend integrációs szakaszban az API-k működését teszteltem éles felületen keresztül. Ellenőriztem, hogy minden adat megfelelően jelenik-e meg, és sikeresen kommunikál-e a szerverrel. Itt különös figyelmet fordítottam a jogosultságkezelés, az adatellenőrzés és a válaszüzenetek helyességére is.

A tesztelési és backend szakaszban Postman használatával végigpróbáltam minden API végpontot. Teszteltem a jogosultságok megfelelő működését, hibakezelést, adatellenőrzést és biztonságot. Emellett manuális tesztelést is végeztem a frontend oldalon. Hibák esetén javításokat végeztem és újrateszteltem az érintett funkciókat.

A dokumentációs szakaszban az egyes fejlesztési lépésekről részletes jegyzőkönyveket készítettem, amelyek tartalmazták az adott szakaszban megtett lépéseket. Ezek a dokumentumok elősegítették a fejlesztés átláthatóságát és az egyeztetések gördülékenységét.

Tervezési szakasz

A fejlesztés megkezdése előtt alapos tervezési munkát végeztem. A cél egy olyan időpontfoglaló rendszer létrehozása volt, amely átlátható, könnyen kezelhető, és külön adminisztrátori és felhasználói felülettel rendelkezik. Első lépésként elkészítettem az ER-diagramot, amelyben megterveztem a táblák közötti kapcsolatokat, kulcsokat és adattípusokat. A tervezési dokumentumban részletesen rögzítettem a rendszerrel szemben támasztott funkcionális és nem funkcionális követelményeket is.

Átgondoltam a navigációs logikát, a különböző szerepkörökből elérhető funkciókat, valamint azt is, hogyan tud a felhasználó regisztrálni, időpontot lefoglalni vagy törölni azt. A tervezés során konzultáltam a szakmai konzulensemmel, aki segített pontosítani az adatbázis struktúráját. Ez a szakasz megalapozta a későbbi fejlesztési munkát, és segített megelőzni a későbbi hibákat és félreértéseket.

Backend fejlesztési szakasz

A backend fejlesztés során PHP nyelvet és MySQL adatbázist használtam. Létrehoztam a kapcsolatot az adatbázissal, megírtam az alapvető CRUD-műveleteket (Create, Read, Update, Delete), valamint az összes API végpontot. Ezek között szerepelt az időpontok kezelése, felhasználói regisztráció, bejelentkezés, jogosultságellenőrzés, és a vélemények értékelésének kezelése is. Minden funkciót REST architektúrában valósítottam meg, így a frontend könnyen kommunikálhatott a backenddel.

A szakmai konzulensem mintákat is biztosított számomra, amelyekből megtanulhattam a jó gyakorlatokat. Törekedtem a kód újrahasznosíthatóságára, és a szétválasztott logikára. A Postman segítségével rendszeresen teszteltem az API-k működését. Ez a szakasz különösen tanulságos volt számomra.

Frontend fejlesztési szakasz

A frontend oldal HTML, CSS és JavaScript technológiákkal készült. A cél egy olyan felhasználói felület létrehozása volt, amely modern megjelenésű, reszponzív, és minden eszközön jól használható. Külön oldalt készítettem a felhasználóknak és az adminisztrátoroknak. A szolgáltatások és időpontok kezelése átlátható formában jelenik meg, és az űrlapok megfelelő validációval lettek ellátva.

A stílus kialakításánál figyeltem arra, hogy ne legyen túlszűfolt a felület. A gombok, táblázatok és az űrlapok jól tagoltan, következetes elrendezésben jelentek meg. A frontend funkciók JavaScript segítségével dinamikusan kommunikáltak a backenddel AJAX hívásokon keresztül. A fejlesztés során folyamatosan teszteltem a működést különböző böngészőkben. Fontos szempont volt a felhasználói élmény javítása és a logikus navigáció biztosítása.

Frontend – Backend integrációs szakasz

A frontend és a backend összehangolása kritikus szakasza volt a fejlesztésnek, hiszen itt dőlt el, hogy a felhasználók számára elérhető funkciók valóban működnek-e. A frontend felületek fejlesztése során gondosan előkészített űrlapokat, gombokat és táblázatokat hoztam létre, amelyek mind dinamikusan, AJAX hívásokon keresztül kommunikáltak a backend API-kkal. A kapcsolat biztosításához JSON formátumban küldtem és fogadtam az adatokat.

Fontos volt a megfelelő hibakezelés kialakítása a frontenden is: ha például az adatbázisból nem érkezett válasz, vagy hibás adatot adott vissza az API, a felhasználó jól érthető hibaüzenetet kapott. Teszteltem a teljes működési láncot minden funkció esetén – regisztráció, belépés, időpontfoglalás, foglalások kezelése, időpont módosítás, időpont törlés, valamint értékelés rögzítése.

Tesztelési szakasz

A tesztelési szakaszban célom az volt, hogy a fejlesztett rendszer minden funkciója hibamentesen működjön. Teszteltem a regisztrációt, bejelentkezést, a jogosultságkezelést, az időpontok létrehozását, módosítását és törlését, a foglalások kezelését valamint a vélemények beküldését is. Ehhez Postmant használtam, a válaszokat pedig JSON formátumban ellenőriztem.

A tesztelés nemcsak funkcionális szempontból történt, hanem figyeltem a hibakezelések megfelelő működésére is. Manuálisan is kipróbáltam minden hibalehetőséget a frontend oldalon, például üres űrlapmezők beküldését, érvénytelen adatok megadását, jogosulatlan hozzáférések tesztelését. A hibákat dokumentáltam, majd a javítás után újratesteltem az érintett funkciókat. A tesztelési fázis zárásaként egy összesített jegyzőkönyvet készítettem a sikeres és sikertelen esetekről.

Dokumentációs szakasz

A dokumentációs szakasz során a hangsúly nemcsak a technikai specifikációkon, hanem a fejlesztési szakaszokhoz kapcsolódó részletes jegyzőkönyvek elkészítésén volt. Minden egyes főbb szakaszból – tervezés, backend fejlesztés, frontend fejlesztés, frontend-backend integráció, tesztelés – külön jegyzőkönyvet vezettem, amelyekben részletesen rögzítettem az elvégzett feladatokat. Ezek a dokumentumok nemcsak a munkám nyomon követését segítették, hanem fontos kommunikációs eszközt is jelentettek a konzulenssel való egyeztetések során.

A jegyzőkönyvek szerkesztése során ügyeltem arra, hogy naprakészek, világosak és strukturáltak legyenek. Minden jegyzőkönyv tartalmazta az adott időszak tevékenységeinek listáját, az elért eredményeket. Ezek segítségével bármikor vissza tudtam tekinteni arra, milyen döntések és módosítások történtek a fejlesztés során.

A projekt során szerzett új ismeretek

A szakmai gyakorlat alatt rengeteget fejlődtem a backend fejlesztés területén. A jogosultságkezelés implementálása lehetőséget adott arra, hogy többféle felhasználói szintet kezeljek egy rendszerben.

További új terület volt számomra a REST API-k strukturált felépítése, valamint a Postman használata az API tesztelésére. A frontend fejlesztés során gyakoroltam az űrlapvalidációt, az eseménykezelést és a felhasználói élmény fokozását célzó megoldásokat.

Megismertem a JSON formátum hatékony kezelését, és létre tudtam hozni komplex JSON struktúrákat. A gyakorlat során megtanultam, hogyan kell strukturált módon dokumentálni a munkafolyamatokat, és hogyan lehet tesztelhető, moduláris kódot írni. A frontenden fejlesztési szakaszban megerősödött a HTML, CSS és JavaScript tudásom.

Megoldott problémák

Több technikai kihívással is szembe kellett néznem a gyakorlat során.

Kihívás volt a kapcsolat.php fájl kialakítása is, amely az adatbáziskapcsolatot valósította meg. Eleinte nem megfelelően zártam le a kapcsolati objektumokat, ami memóriahasználati problémákhoz vezetett.

További probléma merült fel az adatellenőrzés során: bizonyos esetekben a felhasználó hibás vagy hiányos adatokat küldött az API-nak. Bevezettem részletes hibakezelő válaszokat JSON formátumban, amelyek megkönnyítették a hibák azonosítását és kijavítását. Ezek a megoldások később a frontend fejlesztést is támogatták.

A frontend oldalon a legnagyobb problémát a különböző böngészők eltérő CSS-kezelése jelentette. Ennek kiküszöbölésére reszponzív egységeket, Flexboxot és alkalmaztam. Ezáltal egységes megjelenést tudtam biztosítani különböző eszközökön.

Reflexió a szakmai fejlődésre

A szakmai gyakorlat során lehetőségem nyílt arra, hogy mélyebb betekintést nyerjek a szoftverfejlesztés mindennapi gyakorlatába. A backend fejlesztés során különösen sokat tanultam: jobban átlátom az adatkezelés logikáját, a kód strukturálásának fontosságát, és azt is, hogyan kell hibakezelést beépíteni egy éles környezetbe közeli rendszerbe. A gyakorlat során fejlődött az önálló munkavégző képességem, ugyanakkor megtapasztaltam a csapatmunka előnyeit is.

A munka során rengeteg új ismeretet szereztem az adatbázis-kezelés, az API-fejlesztés és a frontend dizájn területén. A felhasználói élmény tervezése kapcsán is fejlődtem: tudatosabb lettem abban, hogyan kell úgy kialakítani egy felületet, hogy az könnyen használható legyen. A hibák kezeléséhez való hozzáállásom is fejlődött – nem csupán a megoldásra koncentráltam, hanem a kiváltó okok feltárására is.

A gyakorlat során fontos visszajelzéseket kaptam, amelyeket beépítettem a munkámba. Fejlődtem az időbeosztásban is: megtanultam, hogyan kell egy nagyobb projektet szakaszokra bontani, és azokat határidőre teljesíteni.

A gyakorlati tapasztalatok révén olyan problémákba ütköztem, amelyek megoldásához új gondolkodásmódra volt szükség. Például egyes API hibák elhárításához először meg kellett tanulnom logikusan szűkíteni a hibakeresést. Megértettem, miért fontos az átlátható, kommentált kód, különösen, ha más fejlesztők is dolgoznak az adott rendszerrel.

A konzulensem mindig biztatott arra, hogy először próbáljam meg önállóan megoldani a feladatokat, ez pedig hozzájárult a kreatív gondolkodás fejlődéséhez. Később már én is proaktívan kerestem megoldásokat új funkciók kialakítására, nem csak a kiadott feladatokat teljesítettem. Ilyen új funkciós ötlet volt a szolgáltatók értékelési lehetősége is.

Rájöttem, mennyire fontos a dokumentáció és a strukturált gondolkodás egy projekt során. Ezek nélkül könnyen átláthatatlanná válik a fejlesztés, és nehéz követni a változtatásokat, különösen hosszú távon.

Összegzés

A szakmai gyakorlat lezárásaként elmondhatom, hogy a megszerzett tudás, tapasztalat és a cégnél szerzett élmények jelentősen hozzájárult a jelenlegi tanulmányaimhoz. Egy összetett, működő webes alkalmazást készítettem el, amely a gyakorlat elején még távoli célnak tűnt. A gyakorlati tapasztalatok révén az elméleti ismereteimet sikerült gyakorlati tudássá alakítani.

A projekt során szembesültem valós problémákkal, amelyeket meg kellett oldanom. Ezáltal fejlődött a problémamegoldó képességem, a kreatív gondolkodásom és a technikai tudásom. Megtanultam együtt dolgozni másokkal, visszajelzéseket kezelni, dokumentálni és tesztelni. A konzulensemtől kapott minták és segítség nagyban hozzájárult ahhoz, hogy az egyes funkciókat megfelelően tudjam implementálni.

Bízom benne, hogy a jövőben hasonló feladatokkal találkozhatok, és még komplexebb rendszerek fejlesztésében vehetek majd részt.

Sikeresnek érzem projektemet, hiszen a rendszer minden funkciója végül hibamentesen működött, és a visszajelzések alapján a konzulensem is elégedett volt a végeredménnyel. Az időpontfoglaló rendszer jó példája annak, hogyan lehet az egyetemen tanult elméletet valódi, gyakorlati tudássá alakítani. Az egész fejlesztési folyamat során törekedtem arra, hogy strukturált, logikus munkamenetet kövessek.

Összességében a szakmai gyakorlat sokkal többet adott, mint pusztán technikai tudás. A jövőben is szeretnék komplex projekteken dolgozni, és továbbfejleszteni a megszerzett készségeimet.