Bemutatkozás - Név - Szak – Csapat

A mi feladatunk volt a \*\*hibajegyek felvétele, tárolása, kezelése\*\*, illetve azokról különféle \*\*statisztikák\*\* készítése a mi rendszerünk és a többi rendszer számára. A rendszerünk megtervezése során törekedtünk arra, hogy minden nagyobb részfeladatot \*\*apró taskokra\*\* bontsunk, és hogy mindenki dolgozzon minden rétegben. Mindenkinek tesztelnie kellett a másik által írt szolgáltatásokat, ennek köszönhetően tudtuk más kódját validálni és jutottunk el oda, hogy a tervezést követően mindenki tudott gond nélkül dolgozni a nagyobb taskokon is. A heti \*\*demózás\*\*ok során \*\*SCRUM-os\*\* megközelítést alkalmaztunk, azaz mindig az aktuálisan már működő alkalmazásunkat mutattuk be, majd átbeszéltük, hogy ki mit csinált a héten, esetleg milyen kódok vannak, amelyek még nem készültek el teljesen és emiatt nem kerültek még be az alkalmazásba. Ezekután mindig átbeszéltük, hogy hogyan lehetne bővíteni a rendszerünket esetleg miket kellene kijavítani benne. Az Issue-tracker rendszerben az iteratív fejlesztés jelentette a legnagyobb kihívást a csapat számára, ugyanis folyamatosan új igényeket kaptunk és ennek hála nagyon sokszor változott az adatmodellünk, valamint a szolgáltatásaink.

Fő feladatom a IssueTracker oldalon történő hibajegyek felvételének és módosításának implementálása volt, mely elég komplex volt, ami azt jelenti, hogy egészen az adatbázis rétegtől kezdve, az ezen feladatot/okat megvalósító serviceken kersztül a megjelenítési rétegig mindenbe belekóstolhattam. Létrehozásnál ügyelni kellett arra, hogy a későbbiekben ezt a HelpDesk modul könnyedén, kevés adat megadásával tudja kezelni remote ejb segítségével. A hibajegyek létrehozásánál figyelembe kellett vennem azt is, hogy sok adat összefügg, pl. egy esetleges cég kiválasztása után a saját hibajegy típusait tudja kiválasztani, amihez AJAX-ot használtam. A hibajegyek módosításánál ügyeltem a felhasználó barát kinézetre, így nagyban hasonlít a hibajegyek részletes nézetére, nyilván azzal a különbséggel, hogy itt az adott jogokkal rendelkező „dolgozó” tudja szerkeszteni is a kívánt adatokat. Feladataim közé tartozott még egy olyan logolás konfigurációja, mely az IssueTracker modul saját logjait egy külön mappába szedi szét, és ezek közül is az error-okat/hibákat egy külön txt fájlba írja, ezzel is megkönnyítve az esetleges hibák keresését. Továbbá a JUnit teszteket is írtunk a servicekhez, melyből én is kivettem a részem. Szándékosan különböző servicekhez írtunk teszteket, mint amit megírtunk, így rákényszerítve magunkat a pontosabb működésbeli ellenőrzésre, így nem csak teszteltünk, de megértettük, hogyan is működnek a programunk nem általunk írt részei is. Kódunk felügyeletére coberturát használtunk, ennek köszönhetően a kódunk jobb minőségű lehet, ennek a reportja megtalálható a jenkinsen is