

Feladat címe: Étterem-webapp

KÖVETELMÉNY SPECIFIKÁCIÓ (SRS)

Csoport tagok:

Név:	Neptun kód:
Boros Bence	Q8V8J4
Zagyva István	FA8QJX
Tóth Bence	WDFP8X

Tartalom

1. Bevezetés:	3
2. Áttekintés:	3
3. A rendszer funkciói:	4
3.1 A rendelések blokk:	6
3.2 Szakács fő blokk:	7
4. Használhatóság:	7
4.1 A felhasználók tudásszintje	7
4.2 A sebesség más rendszerekhez viszonyítva	8
4.3 Az alkalmazás felhasználói felülete	8
5. Megbízhatóság:.....	8
6. Megbízhatóság:.....	9
7. Támogatottság:.....	10
8. Tervezési korlátozások:	10
8.1 Igényekhez igazodás	10
8.2 Fejlesztői eszköz és programozási nyelvek	11
8.3 Módszertan	11
9. Felhasznált kész komponensek:.....	11
10. Interfészek:	12
10.1 Felhasználói interfészek	12
11. Alkalmazott szabványok:	12
12. Mellékletek	13

1. Bevezetés:

Introduction

Az étterem webapp, egy olyan webes felületű alkalmazás, amely egy étterem belső rendelés - igazgatási rendszerét hivatott segíteni, egészen konkrétan az adott asztalnál a rendelést felvevő pincér munkáját segíti, és egyszerűsíti.

2. Áttekintés:

Overview

Felhasználók és főbb funkciók:

A felhasználókat két szerepkörre oszthatjuk:

A pincér felhasználók, a korábban már említett módon a pincér munkáját segíti, jelen esetben nem papír alapon kerül rögzítésre a rendelés, hanem az alkalmazáson keresztül.

A pincér felhasználók funkciói:

- Asztal foglalás: Először lehetősége nyílik asztalt választani, melyhez rögzítésre kerül majd a rendelés,
- Rendelés felvétele: Ez a funkció teszi lehetővé, hogy a vendégek által, az étlapról kiválasztott ételeket rögzítse a pincér, mely étlapot kategóriákra bontva lát a pincér az alkalmazáson keresztül, valamint lehetősége van az adott ételekből a mennyiséget is kiválasztani.
- Rendelés leadása: Ez a menüpont azt követően jelenik meg, amint minden ételből, italból ki lett választva igény szerint, és a pincér megnyomja az étlap alatt található **Rendelés** gombot, ezt követően megjelenik a weboldal jobb oldalán, egy oldalszéli menüsáv, mely tartalmazza az adott asztalhoz tartozó összes megrendelt étlap elemet, és a mennyiséget. Majd az ezen az oldalszéli menüsávon, az összesítő lista alatt megtalálható **Megrendelés** gombot megnyomva, a rendelés leadásra kerül, és ezen a ponton jön elő az

alkalmazás másik, a felhasználókat illető csoportja, továbbá a pincér a rendelés leadása után vissza léphet a Főoldalra, azaz ahol az asztalok vannak, és fel vehet másik rendeléseket.

A szakács felhasználók funkciói:

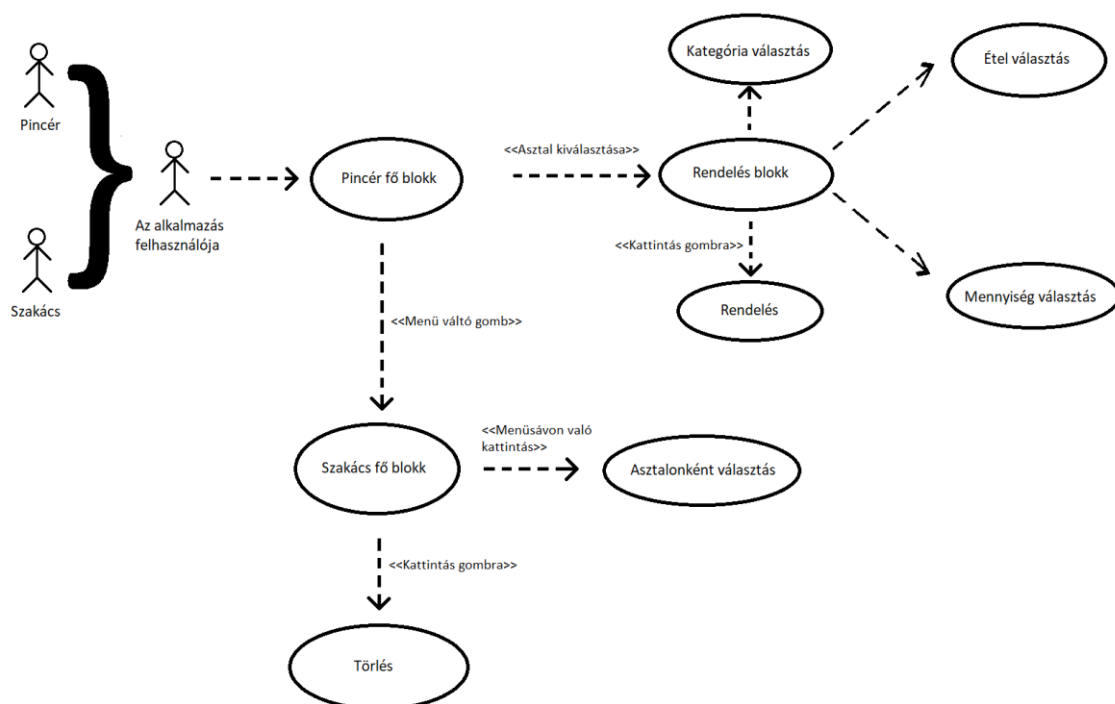
- Asztalonkénti rendelések: szakács felhasználók számára a felület egészen más, mert számukra lehetőség van az asztalok sorszáma szerint külön választva letisztult, és átlátható módon megtekinteni a rendeléseket,
- Rendelés véglegesítése/törlése: Amennyiben az adott asztalhoz tartozó egyik megrendelt elemet sikerült elkészíteni, azaz kivitelre készen áll, a szakács felhasználóknak lehetősége van eltávolítani az adott elemet a listából. Természetesen minden elemre külön- külön van lehetőség az eltávolításra.

Feltételezések, függőségek és korlátozások

A webalkalmazás oldalait bármely internet-hozzáféréssel rendelkező számítógép böngészőjéből el lehet érni, egyéb előfeltétele, vagy igénye nincs.

3. A rendszer funkciói:

Specific Requirements



Szakács: Olyan alkalmazott, aki a szakács szerepkörbe tartozva van bejelentkezve.

Pincér: Olyan alkalmazott, aki a pincér szerepkörbe tartozva van bejelentkezve.

Bejelentkezés:

Előfeltétele: Internetkapcsolat.

Leírás: Ez az oldal ad lehetőséget a szerepkörök szerint történő bejelentkezésre.

Szokásos működés: Megjelenít két mezőt, a felhasználó név-jelszó páros beírásához.

Pincér fő blokk:

Előfeltétele: Internetkapcsolat, pincér szerepköri bejelentkezés.

Leírás: Ez az oldal az, ahol a pincérnek lehetősége van kiválasztani az asztalt a rendelés felvétele előtt.

Szokásos működés: Megjeleníti az étterem összes asztalát, melyről az asztal kiválasztásával (kattintással) lehet tovább lépni a rendelések felvételére.

3.1 A rendelések blokk:

Előfeltétele: A Pincér fő blokkban való sikeres asztal választás.

Leírás: Az asztal kiválasztását követően erre az oldalra navigál át a program, melyben öt funkció található, amelyek segítségével lehetséges egy rendelés teljes lebonyolítása.

Szokásos működés: A kategória kiválasztását követően az adott ételekből / italokból kiválasztani a vendégek igényének megfelelő mennyiséget, majd leadni a rendelést.

- **Kategória választás funkció:** Az étterem étlap-adatbázisában tárolt ételek / és italok mindegyike rendelkezik egy kategória azonosítóval, mely a program ezen funkciójához szükséges, mert a rendelések funkcióhoz tartozó oldal felső részén található fülek, ezen kategóriákat reprezentálják.
- **Étel választás funkció:** Ezen funkció az, amely az étlap-adatbázisból betölti az ételeket és italokat, a kategória kiválasztást követően, és annak megfelelően.
- **Mennyiség választás funkció:** Ezen funkció minden egyes, az adatbázisból betöltött étlap elem mellett megtalálható, itt lehet kiválasztani, hogy a megrendelni kívánt elemből mennyi kell.

3.2 Szakács fő blokk:

Előfeltétele: Internetkapcsolat, szakács szerepköri bejelentkezés.

Leírás: Ez az oldal az, ahol a szakácsnak lehetősége van megnézni az asztalonkénti rendeléseket, valamint a kész rendeléseket eltávolítani.

Szokásos működés: Megjeleníti az asztalokat (fülek formájában, az oldal tetején), alatta pedig a kiválasztott asztalhoz tartozó rendeléseket, a rendelés mellett meg a gombot, ami az adott rendelést eltávolítja a listából.

- **Asztal kiválasztása funkció:** Ez a funkció a lap tetején megjelenő sáv, mely az adatbázis alapján megjeleníti az asztalokat számozva, illetve rákattintva az egyikre ki is listázza az adott asztalhoz leadott rendeléseket.
- **Törlés funkció:** Ez a funkció egy gomb formájában jelenik meg, amely minden, az adott asztalhoz tárolt rendelés-lista elem mellett megjelenik, és lehetőséget ad ezen lista elem az adatbázisból, és ennél fogva a listából történő eltávolításra. Egyféle biztonsági rendszert is tartalmaz ez a funkció, mert kattintást követően egy felugró ablak formájában megkérdezi, hogy biztosan törölni kívánja-e a lista elemet.

4. Használhatóság:

Usability

A termék egy adott feladatcsoport egyszerűen történő ellátására készült. Az oldal jól áttekinthető és könnyen kezelhető. Fontos szempont a könnyű hozzáférhetőség mellett az egyszerű használhatóság.

4.1 A felhasználók tudásszintje

Nincsenek benne haladó számítógépes tudást igénylő elemek, egyszerű, és a lehető legfelhasználóbarátabb. Az alapvető kezelési ismeretek elsajátítását követően bárki képes az alkalmazás funkcióit kezelni.

4.2 A sebesség más rendszerekhez viszonyítva

Az alkalmazás sebessége, a felhasznált modern technológiák és a program dinamikus felépítésének köszönhetően gyorsabb, mint más, meglevő rendszerek. Az alkalmazás menüpontjai, funkciói között való navigálás egyszerű és gyors, azaz nem kell egy adott funkció eléréséhez sok szinten keresztül kutakodni.

4.3 Az alkalmazás felhasználói felülete

Az alkalmazás weboldalának felhasználói felületei egyszerű felépítésűek. A fontos funkciók könnyen megtalálhatóak, és szem előtt vannak. A zsúfoltságot elkerülése érdekében, az alkalmazás felületein a funkciók kényelmesen vannak elhelyezve. A főmenü funkcióját leszámítva, bármely funkcióra, vagy blokkra kattintva nem egy új oldalra navigálja át a felhasználót, hanem az adott oldalon marad.

5. Megbízhatóság:

Reliability

A szolgáltatás az alkalmazás szándékos leállítását leszámítva mindig elérhető. Szándékos leállítás alatt az adatbázis, és az alkalmazás más egyéb részeinek időleges karbantartása miatt történhet. Valamint függ a szolgáltatás teljesítménye az egyidejűleg kiszolgált felhasználók számától, de tekintve hogy ez az alkalmazás egy zártkörű, viszonylag alacsony felhasználói létszámot szolgál ki, ez nem jelenthet problémát. A meghibásodásokat 24 órán belül javítani kell a szolgáltatás zökkenőmentes használata érdekében. Az adatbázisról egy naprakész backup

tárolódik és az esetleges hardveres meghibásodások esetén minimális adatvesztéssel lehet helyreállítani a szolgáltatást.

6. Megbízhatóság:

Performance

A válaszidők az alkalmazáson belül alacsonyak, ez köszönhető a jól megírt alkalmazásnak, valamint a szervergép erőforrásainak. A főbb funkcióknál, egy másodpercnél kisebb válaszidőre számítunk. Az alkalmazás egyféle adatbázist használ, melyben helyet kapnak a felhasználói, és étlap elemek is. Az adatbázis mérete a felhasználók számától és a tárolt étlap-elemek mennyiségétől függ. Ennél fogva az adatbázis mérete lehet jelentősen nagy is, és ilyenkor romlik az adatbázis lekérdezéseinek gyorsasága, ezeket igyekszünk kiküszöbölni különböző adatbázis-kezelési módszerekkel.

Alkalmazásunk használatához szükséges gépigény: csupán egy böngésző futtatásához kell elegendőnek lennie:

- Processzor: Intel I3 vagy újabb
- Memória: 2GB
- Merevlemez: nincs szükség tárhelyre, mivel webalkalmazás
- Monitor és egyéb perifériák az alkalmazás használatához
- Internet elérés, mivel csekély méretű képek lesznek jelen, nem muszáj a nagy sávszélesség, az adatok betöltése a lényeg.

A szerver esetleges konfigurációja a következő lehet (ajánlott):

- Processzor: 2x Intel Xeon 3,2Ghz (6,12 mag)
- Memória: 32GB DDR3 2000Mhz CL6-7
- Merevlemez: 4TB, raid technológiával, hozzá raid vezérlő
- Hálózati kártya: Dual Gb Ethernet (10/100/1000Mbps)
- Alaplap: Intel SC5650DP
- Tápegység: 750W

- Sáv szélesség : BIX hálózatra kötve 1Gbit/1Gbit

7. Támogatottság:

Supportability

Alkalmazásunkat úgy készítjük el, hogy a későbbi fejlesztés és karbantartás egyszerű legyen olyan emberek számára is, akik nem voltak benne az alkalmazás megírásában. Az adatbázist és az alkalmazást átláthatóvá tesszük, a szükséges alkalmazásokkal és a technológia ismeretével a karbantartás zökkenőmentesen folyhat.

Fontosabb szempontok a támogatottsággal kapcsolatban:

- Fejlesztőkörnyezetnek a Visual Studio Code-t, és a PyCharmot használjuk amely segítségre van az alkalmazás működésének leellenőrzésében.
- Elemezni tudjuk a kód komplexitását és stabilitását a Code Analysis segítségével, lehetőség szerint törekszünk a minél nagyobb stabilitásra
- A későbbi karbantartók számára létrehozunk egy fejlesztői útmutatót a szoftver működésével és a kód felépítésével kapcsolatban

8. Tervezési korlátozások:

Design Constraint

8.1 Igényekhez igazodás

A megrendelő és a felhasználók igényeinek megvalósítása is fontos, ugyanis a megrendelő által előírt követelményeknek eleget kell tenni, valamint a felhasználók visszajelzéseinek is igyekszünk megfelelni.

- kívánt funkciók, szolgáltatások megvalósítása, amelyeket a megrendelők előírnak.
- korszerű alkalmazás létrehozása, amely a legfejlettebb technológiákat használja, így egy dinamikusan működő webalkalmazást igyekszünk megvalósítani
- megfelelő biztonsággal rendelkező alkalmazás létrehozása, amely védi a felhasználók adatait, a támadások ellen

8.2 Fejlesztői eszköz és programozási nyelvek

A fejlesztő eszközök a Visual Studio Code, és a PyCharm volt.

A választott programozási nyelvek: a backend részhez, a Python 3.7.5-ös verziója, a frontend részhez az Angular keretrendszer (TypeScript) 8-as verziója.

8.3 Módszertan

Az alkalmazás fejlesztése során az agilis módszertan által egyik definiált módszerben, a SCRUM-ban dolgozott a csapat.

9. Felhasznált kész komponensek:

Third-party Components

Alkalmazásunk más szoftverből való komponenst nem használ fel. Ezen felül az alkalmazásunk implementációjában törekszünk az egyéni programkód írására, amely nem használ fel más műsorszolgáltatóktól vett komponenseket.

10. Interfészek:

Interfaces

Alkalmazásunk nem definiál külön felületeket más alkalmazások számára, csak azokat melyek a felhasználók, illetve az étlap-elemek elérésére, és a velük történő kommunikálásra szolgál.

10.1 Felhasználói interfészek

A felhasználó a böngészőn keresztül éri el az alkalmazást.

A felhasználói interfészre vonatkozó követelmények:

- Könnyű átláthatóság a felületen
- Legyen jól kezelhető, letisztult, válaszképes
- Egységes téma, stílusok, megjelenítése

A felhasználók szempontjából előnyös a könnyű kezelhetőség, gyorsan elsajátítható funkciók összessége.

11. Alkalmazott szabványok:

Az alkalmazás implementációja összességében megköveteli az objektumorientáltságot, mivel bár a Python nyelv nem, de az Angular mint keretrendszer, ami magába foglalja a Scrit nyelvet, mely viszont követi az Objektum orientált elveket. Törvény által előírt szabványok: Adatvédelmi törvény, a felhasználók adatait úgy kell tárolni, hogy harmadik fél kezébe ne kerülhessenek, valamint ne élhessenek vele vissza. Ehhez szükséges egy megfelelően biztonságos alkalmazás, valamint adatbázis, amely védve van a különböző támadások ellen, pl.: SQL injection.

12. Mellékletek

Appendix

Az alkalmazás megtalálható:

[https://github.com/Ej-Tim/Pincer Alkalmasas/tree/dev](https://github.com/Ej-Tim/Pincer_Alkalmasas/tree/dev)