

# Okosreceptkönyv

## Projektfeladat specifikáció



**Informatikai Biztonsági és  
Adatvédelmi Tanácsadó Kft.**

# 1 Tartalomjegyzék

1	Tartalomjegyzék.....	2
2	Bevezetés .....	3
2.1	A feladat címe .....	3
2.2	A feladat rövid ismertetése .....	3
3	Elvárások a feladattal kapcsolatban.....	4
3.1	Operációs rendszer, környezet.....	4
3.2	Felhasználható programozási nyelv .....	4
3.3	Megoldás formátuma.....	4
3.4	Szoftverfejlesztés .....	4
3.5	Modulok.....	4
4	Szoftver specifikáció.....	6
4.1	Megjelenés .....	6
4.2	Funkciók.....	6
5	Dokumentáció .....	7
5.1	Erőforrásterv, munkaidő-nyilvántartás .....	7
5.2	Üzemeltetői dokumentáció .....	7
5.3	Forráskód dokumentáció.....	7
5.4	Felhasználói dokumentáció .....	7
6	A projekt értékelése.....	8
6.1	A feladat értékelésének felhasználó oldali szempontjai.....	8
6.2	A feladat értékelésének technikai szempontjai.....	8
6.3	Projekt megvalósításának piaci jellegű értékelése .....	8
7	Projekt adatlap.....	9

## 2 Bevezetés

### 2.1 A feladat címe

Okosreceptkönyv

### 2.2 A feladat rövid ismertetése

Napjainkban a miket körülvevő világ lassan minden eleme „okos”, így jogosan merülhet fel az igény egy okosreceptkönyv elkészítésére is. A hagyományos receptkönyvekben csak névre, esetleg kategóriára (leves, desszert stb.) lehet keresni. Az okosreceptkönyv annyival tud többet, hogy a felhasználó beírhatja, milyen alapanyagokat talál otthon a kamrában, az alkalmazás pedig megjeleníti azokat a recepteket, amelyekben szerepelnek a megadott összetevők.



## 3 Elvárások a feladattal kapcsolatban

### 3.1 Operációs rendszer, környezet

- Windows 10+, asztali alkalmazás (desktop application)

### 3.2 Felhasználható programozási nyelv

- Nincs megkötés
- Ajánlott nyelvek, keretrendszerek: Java, C#, Dart; WPF, UWP, Flutter

### 3.3 Megoldás formátuma

- Forráskód állományok
- Teljes projekt környezet
- Forráskód dokumentáció
- Üzemeltetői dokumentáció (odt/docx és pdf formátumban)
- Felhasználói dokumentáció
- Erőforrásterv és munkaidő-nyilvántartás

### 3.4 Szoftverfejlesztés

A feladat egy olyan program készítése, melyhez recepteket adhatunk hozzá, illetve kereshetünk köztük. A szűrőnek az egyszerű név- és kategóriakeresésen túl képesnek kell lennie arra, hogy az étel összetevői közt is keressen.

A fejlesztés közben be kell tartani a Clean Code alapszabályait.

Az elkészült megoldásnak maradéktalanul meg kell valósítania a 4. fejezetben megfogalmazott követelményeket. Amelyik követelmény nincs pontosan definiálva, ott a megvalósítás során a fejlesztő szabad kezet kap. Fontos, hogy a választott megoldás megfelelő színvonalú legyen mind felhasználói, mind fejlesztői szempontból.

### 3.5 Modulok

A projekt keretében történő megvalósítás egy lehetséges felbontási lehetősége az alábbi:

- Adatbázis tervezése, kivitelezése, interfész megírása
- Felhasználóbarát front end (GUI) tervezése
  - Applikáció felülete, grafikai elemek
  - Menürendszer
  - Beállítások
- Back end kivitelezése
- Projektvezetéssel kapcsolatos dokumentációk, nyilvántartások vezetése, feladatok összehangolása, felhasználói dokumentáció elkészítése, tesztelés.

Lehetőség szerint a fejlesztői dokumentációkat minden esetben a ténylegesen fejlesztést végző projektagok készítsék el. A felhasználói dokumentáció külön egységet képezhet, melyet azonban érdemes a teszteléssel összekapcsolni a megfelelő minőség biztosítása érdekében.

## 4 Szoftver specifikáció

Az alkalmazás elsődleges célja, hogy a recepteket tárolja. A másik és egyben a bonyolultabb célja, hogy úgy lehessen keresni a receptek között, hogy a megadott hozzávalókat tartalmazó recepteket kiszűrje. A felhasználó ez alapján láthatja, hogy a „kamra” tartalma milyen ételek/italok elkészítését teszi lehetővé, illetve milyen hozzávalókat szükséges még beszerezni.

### 4.1 Megjelenés

- A programnak átláthatónak és könnyen kezelhetőnek kell lennie
- Teljesen el legyen különítve a keresés és a receptek menedzselése

### 4.2 Funkciók

- A receptek legyenek tipizálva, mint pl.: vegetáriánus, csirke, sertés, leves, főétel stb. Egy recept több típushoz is sorolható
- Elkülönülő egyszerű és összetett keresés
- Egyszerű keresésnél lehessen típusra, vagy névre szűrni
- Az összetett keresés esetén hozzávalókra lehessen szűrni. Ebben az esetben eredményképpen adjon ki olyan recepteket, melyekhez szükségesek a megadott hozzávalók, vagy csak pár dolgot kell még beszerezni az étel elkészítéséhez
- A receptek neve, illetve a hozzávalók listája mellett egy elkészítési leírás is kapjon helyet
- A recepteket egy 1-től 10-ig terjedő skálán lehessen pontozni
- Plusz funkcióként legyen lehetőség egy képet is menteni a recepthoz

## 5 Dokumentáció

### 5.1 Erőforrásterv, munkaidő-nyilvántartás

A specifikáció birtokában a projekt résztvevői készítsenek erőforrástervet. Ez tartalmazza a feladatban részt vevő projektagokat, akik legyenek hozzárendelve a tervezés során azonosított részfeladatokhoz. Minden részfeladat mellé kerüljön egy munkaidő ráfordítási becslés munkaóraban számolva. Ezt a tervet a tényleges fejlesztés előtt le kell adni. A feladat megoldása során az elvégzett munkáról készüljön nyilvántartás részfeladatonként és személyenként a tényleges munkaórák számának megjelölésével. A projekt végén a két dokumentum összehasonlításra, az eltérések elemzésre kerülnek.

### 5.2 Üzemeltetői dokumentáció

Az üzemeltetői dokumentáció célja, hogy a rendszer üzemeltetőinek támogatást adjon a termék üzemeltetésének elsajátításához. Tartalmazza többek között a rendszer architekturális felépítését (alkalmazás stack elemei és azok közti kapcsolat leírása), az alkalmazás fordításához, fejlesztéséhez és futtatásához szükséges követelményeket, technológiákat, továbbá a konfigurációs állományok leírását (ha vannak).

A dokumentációnak a feladat bonyolultságától függő hosszúságúnak kell lennie, maximális terjedelem nincs meghatározva.

### 5.3 Forráskód dokumentáció

A fontosabb függvények és osztályok előtt szerepelnie kell megjegyzéseknek, melyeknek tartalmazniuk kell az azt követő metódus rövid szöveges – akár magyar nyelvű – leírását. A forráskód dokumentációt a munka során folyamatosan kell készíteni.

### 5.4 Felhasználói dokumentáció

Az alkalmazás használatának részletes bemutatása képernyőképekkel, funkciók pontos leírásával.

## 6 A projekt értékelése

### 6.1 A feladat értékelésének felhasználó oldali szempontjai

A működő alkalmazás tesztelése alapján az alábbiak a legfontosabb jellemzők:

- Kiírást teljes egészében lefedő funkcionalitás
- Kényelmes használat
- Igényes felhasználói felület
- Stabil működés
- Igényes felhasználói dokumentáció

### 6.2 A feladat értékelésének technikai szempontjai

Informatikai szakmai szempontból a megoldás értékelésének alapja:

- Kódkép, a kód tisztasága, kommentelés minősége
- Kódolási konvenciók betartása (Clean Code)
- Dokumentációk színvonala
- Dokumentált tesztelés
- Erőforrás felhasználásának pontos nyilvántartása

### 6.3 Projekt megvalósításának piaci jellegű értékelése

A projekt lezárultával összehasonlításra kerül a kezdeti erőforrásterv, valamint a megvalósítás során dokumentált munka. Ezen dokumentumok elemzéséből levezetésre kerülnek azok a problémák, melyek a piaci környezetben jellemzően megjelennek. Végigtekintjük ezen problémák okait, következményeit, lehetséges elkerülésüknek vagy hatásuk mérséklésének módjait. A jellemző hibák ebből a megközelítésből:

- Határidő csúszása
- Nem megfelelő minőség
- Hiányos, vagy elmaradó tesztelés
- Használhatatlan, pontatlan dokumentáció
- Pontatlan erőforrás becslés
- Aránytalanul magas önköltség
- Az elkészült termék továbbfejlesztésének, karbantartásának nehézségei

A fentiek értékelésén túl fejlesztői szemszögből elemezzük a megvalósítás tapasztalatait, a lehetséges továbbfejlesztés, átalakítás, támogatás kérdéseit és piaci lehetőségeit.



## 7 Projekt adatlap

Projekt neve: Okosreceptkönyv

Feladat rövid ismertetése: Receptkönyv, melyben nem csak névre, de hozzávalókra is lehet szűrni

Specifikációt összeállította: Szabó Gábor Ferenc, Csörögi Gyula Attila, Apáti László, Varga Bence