

Időkönyvelő

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	Erőforrás-terv, munkaidő nyilvántartás	7
5.2	Technikai 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

Időkönyvelő

2.2 A feladat rövid ismertetése

Munkahelyünkön gyakran előfordulhat, hogy felettesünk pontosan tudni szeretné, melyik feladattal mennyi időt töltöttünk aznap. Ezt persze egyszerűen áthidalhatjuk azzal, hogy egy határidőnaplóba rendszeresen jegyzeteljük, hogy épp mivel foglalkoztunk. De van erre egy sokkal kényelmesebb megoldás is: mi lenne, ha a számítógépünk mindezt magától végezné el, beavatkozás nélkül?

A cél egy olyan háttérben futó alkalmazás elkészítése, mely folyamatosan jegyzi a fókuszban levő alkalmazás ablakot, nyitásának, zárásának, az ablakok közti váltásoknak pontos idejét, majd ezekről különböző statisztikákat és diagramokat állít össze.



3 Elvárások a feladattal kapcsolatban

3.1 Operációs rendszer, környezet

- Windows 7, 8.1, 10

3.2 Felhasználható programozási nyelv

- C#

3.3 Megoldás formátuma

- C# forrásállomány
- Teljes projekt környezet
- Forráskód dokumentáció
- Technikai dokumentáció (odt és pdf formátumban)
- Erőforrás-terv és munkaidő nyilvántartás
- Felhasználói dokumentáció

3.4 Szoftverfejlesztés

A feladat egy olyan alkalmazás készítése, mely figyeli és eltárolja, hogy éppen melyik futó alkalmazás van fókuszban, mettől meddig volt ebben az állapotban, és mennyi ideig tartott a használata. A jegyzett adatokból a szoftver készítsen statisztikákat, melyeknek paramétereit és tartalmát a felhasználó szabadon adhassa meg.

A fejlesztés közben be kell tartani a meghatározott kódolási konvenciókat, melyeket a gyakorlatvezető szab meg.

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 felület, grafikai elemek
 - Menürendszer
 - Beállítások
 - Összetett szűrő
 - Statisztika

- 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

A szoftver a háttérben végezze működését, és csak a fókuszban lévő programok aktivitásának pontos dátumát és időtartamát kell nyomon követnie. A felhasználói felületnek alkalmasnak kell lennie arra, hogy az addig begyűjtött adatokból különféle statisztikákat jelenítsen meg. A kirajzolt adathalmaz időtartamát a felhasználó adhassa meg.

4.1 Megjelenés

- A program indításakor jelenjenek meg az éppen aktuális nap adatai: az adott napon percre pontosan melyik program mennyi ideig volt fókuszban.
- A begyűjtött adatok rendezhető táblázatban, illetve diagramon jelenjenek meg. A megjelenítéshez lehessen minél több releváns paramétert meghatározni.

4.2 Funkciók

- A fókuszban lévő programok fókuszban töltött idejét folyamatosan mentse el.
- A mentett adatokból paraméterezhető szűrők segítségével lehessen megjeleníteni az alkalmazások használatának statisztikáit. A paraméterek alapesetben a statisztika kezdeti- és végdátuma, valamint a statisztika tartalma (pl.: használati idő, aktiválás számossága, átlagos aktív állapotban töltött idők, ezen statisztikák egymáshoz képesti százalékos aránya, első és utolsó aktiválás dátuma és ideje az intervallumon belül, stb.). A rögzített adatokból egyéb, itt fel nem tüntetett statisztika is implementálható, ami használható adatokat tartalmaz.
- Az adatok alapértelmezés szerint minden paramétere szerint rendezhető táblázatban jelenjenek meg. Ezek a paraméterek: alkalmazás neve, a keresési intervallumon belül megjelenített statisztikák.
- A statisztikák igény szerint diagramon is kerüljenek megjelenítésre, mely tartalmazza a szükséges adatokat. Pl.: alkalmazások megnevezése, használatuk ideje százalékos arányban, aktiválásuk száma százalékos vagy pontos számosság szerint, stb.
- Plusz funkcionalitásként több kiválasztható diagram típust is be lehet építeni, így a megjelenítéshez használt diagram típusa újabb paraméter lehet: torta, oszlop, halmozott oszlop, fánk, stb. Szintén plusz feladatként a diagramok grafikai tartalma is bővíthető: skálázás, szintvonalak, maximum értékek, stb.

5 Dokumentáció

5.1 Erőforrás-terv, munkaidő nyilvántartás

A specifikáció birtokában a projekt résztvevői készítsenek erőforrás-tervet. 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 Technikai dokumentáció

Az elkészült kódot, függvényeket és osztályokat megfelelő kommentekkel kell ellátni, továbbá el kell készíteni a teljes alkalmazás dokumentációját.

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

A technikai dokumentáció szövegezésénél előírás, hogy a nem hozzáértő személyek számára is feldolgozható legyen, így az egyes fogalmak, rövidítések, idegen kifejezések magyarázatát a dokumentumnak tartalmaznia kell.

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
- Ergonomikus kialakítás
- 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
- Fejlesztői dokumentáció színvonala
- Dokumentált tesztelés
- Erőforrás felhasználásának pontos nyilvántartása

6.3 Projekt megvalósításának piaci jellegű értékelése

A projekt lezárultával összehasonlításra kerül a kezdeti erőforrás-terv, valamint a megvalósítás során dokumentált munka. Ezen dokumentumok elemzéséből levezetésre kerülnek azok a jellemző 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: Időkönyvelő

Feladat rövid ismertetése: Időkönyvelés, mely megmutatja, hogy melyik alkalmazás mikor volt a fókuszban

Felhasznált programozási nyelv: C#

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