

ZÁRÓ DOLGOZATI ÖSSZEFOGLALÁS

(Szövegszerkesztővel töltendő ki! Benyújtandó pdf formátumban 1 példányban)

Hallgató neve: Nagy Alexander István
Neptun kódja QJ9GWZ
Munkarend Nappali
Szak/specializáció Gazdaságinformatika

Értesítési címe: 2040 Budaörs, Felsőhatár utca 9
Telefon (mobil) 0630 121 1890
e-mail címe: anagy989@gmail.com

A záró dolgozat Egyedi alkalmazásfejlesztése, dokumentálása és
címe: bevezetése

A záró dolgozat készítésének helye:
Vállalat neve: Vincotech Hungária Kft.
Vállalat címe: Bicske, Kossuth u. 59, 2060
Külső konzulens
Neve, beosztása: Ducsai László, Senoir IT Engineer
E-mail cím, laszlo.ducsai@vincotech.com, 06305301593
telefonszám:

ÖSSZEFOGLALÁS

(Benyújtandó pdf formátumban 1 példányban. Szövegszerkesztővel töltendő ki!)

A szakdolgozatom témájának kiválasztásánál fontos szempont volt, hogy olyan témát válasszak, ami kapcsolódik a felsőoktatási tanulmányaimhoz, illetve a gyakorlati helyemül választott Vincotech Hungária Kft. profiljához, és az ott betöltött pozíciómhöz illeszkedjen. Így hát a választásom, mint a címből is kiderült az egyedi alkalmazásfejlesztése, dokumentálása és annak bevezetésére esett.

Már középiskolában is a C# illetve SQL volt az amiket tanultam. Szerencsére az egyetemi tanulmányaim alatt is ezek a nyelveket tudtam továbbtanulni, illetve sikerült egy olyan helyszínt találni a szakmai gyakorlatomnak, ahol ezeket a nyelveket a gyakorlatban is tudtam alkalmazni.

Először nem igazán értettem, hogy mi értelme van megtanulni a programozási elveket, mivel a kódolás a lényeg. Később mikor a gyakorlatomat végeztem és elkészültem pár kisebb projekttel és megkaptam az első nagy munkámat, fejlesztés közben eszembe jutottak ezek az elvek. És akkor jöttem rá, hogy igazából azok mindennek az alapja. Ott van a stratégia elv ami leírja a programok részekre való bontását, ami elengedhetetlen egy nagyobb alkalmazásnál. Vagy a taktikai elvek, amik segítenek a kódunk finomabbá, jobbá tételében. Ezen kívül a technológiai elvek, technikai elvek és esztétikai-ergonómiai elvek is, amiket akarva akaratlanul használunk és jó, ha ismerjük őket.

A való életben is egy egyszerűnek tűnő cselekvés, mint például a kávéfőzés is leírható algoritmussal. Az algoritmus konkrétan megmondja, hogy egy adott feladat elejétől a végéig, milyen cselekvéseket, milyen sorrendben kell végrehajtanunk. Három szerkezeti egységre tudjuk lebontani, szekvencia (műveletsor), szelekció (kiválasztás) és iteráció (ismétlés, ciklus).

Egy alkalmazás elkészítése nem csak abból áll, hogy leülünk és egy programozási nyelvet kiválasztva elkezdünk eszeveszettül gépelni, mint ahogy a filmekben az látható. Ez ennél sokkal bonyolultabb folyamat. Szükség van egy feladatleírásra ami alapján meg tudjuk tervezni az alkalmazást, akár elemezni a felhasználóréteget akiknek készül és ez alapján a felületet elkészíteni. Ezt követően neki eshetünk a kódolásnak is, amit követhet egy hosszas

tesztelés, hibakeresés. Személy szerint a következő lépés az amit leginkább szeretek, a hibajavítás. Mindezek után, ha egy igazán jól működő programot akarunk készíteni a hatékonyságvizsgálattal még jobban bele tudunk merülni egyes kódrészletekbe ami fejlesztésével, mindig tudunk új technikákat tanulni.

Mielőtt kiadnánk az alkalmazásunkat, egy jól megírt felhasználói és egy fejlesztői dokumentáció megírása elengedhetetlen. Előbbi a felhasználóknak, hogy könnyedén el tudják sajátítani a program használatát, amit készítettünk. Utóbbi az esetleges későbbi fejlesztéshez mind nekünk, de leginkább, ha más szeretne hozzá nyúlni, tudja mit hol keressen.

Miután bevezettük az alkalmazásunkat, és élesbe megy minden a tesztek és hibajavítások ellenére előjöhetnek hibák amiket javítanunk kell, akár azonnal. Későbbiekben esetleges frissítéseket kiadni.