

Eseményvezérelt és Vizuális Programozás (EViP)

Bevezető:

EViP Technologies, OnBoarding

(Bullet point lists...)

EViP Technologies (fiktív cég 😊)

- Szoftverfejlesztés
(C#, .NET, Universal Windows Platform)
- Workflow
Verziókövetés, code review, pull requestek,
páros programozás, online kapcsolattartás
- Infrastruktúra
github, github classroom, Moodle (online.vik.bme.hu !), Teams
- Feladat centrikus tanulás
(problem based learning)

A tananyag

- C#, mint programnyelv
- .NET alapok: szövegkezelés, I/O, kollektciók és generikus szerkezetek, Linq, lambda kifejezések
- Universal Windows Platform (UWP): UI, események, adatkötés, rajzolás
- Többszálú programozás (async metódusok)
- Kis szoftvertervezés
Model – View – View model architektúra, Command Pattern
- Visual Studio mint fejlesztő környezet
(solution, projekt, debug, unit tesztelés)
- Kód olvasás, mint képesség (vs zöld mezős projektek)

Néhány kísérlet...

- Otthon is elvégezhető laborok
 - > covid ügy kiszámíthatatlan, flexibilisek maradunk.
 - > Az egyetemi elvégzéshez hétfőn délig a Moodle alatt regisztrációt fogunk kérni, hogy tudjunk tervezni
 - > Labor összevonások dinamikusak, a fő visszajelzés úgyis a code review.
- Előre felvett előadás anyagok
(konktaktórán kiegészítve, konzultáció)
Teams alatt, a Stream kivezetés alatt áll.
- Forráskód centrikus előadások
(Példaprogramokban “EVIP” prefixű kommentárok)
- Sok kísérleti dolog... ha valami látszólag nem lenne fair, szóljatok!!!

Github és github classroom



- Személyes repository
 - > Meghívó URL a Moodleben
- Laborokon
 - > Munka saját repositoryba, külön könyvtárban a solution, könyvtár nevében a labor sorszáma, pl. LAB01)
 - > Kiinduló kód majdnem mindig lesz hozzá.
 - > Leadáskor pull request létrehozása, reviewer a labvezér.
Labvezérek listája: Moodle alatt

Tárgykövetelmények

- Laborok: részvétel min. 70%-on
 - > Otthonról is teljesíthető,
 - > Értékelt (összesen 50 pont), részvétel = min. 1 pont megszerzése
 - > Nem értékelt (pl. konzultáció, csapatos gyakorlás)
 - > Leadási határidő a labor vége utáni SZERDA éjfél (!), értékelés fáziskéséssel (ellenőrzés, kód átnézése)
- 2 db ZH: 2x25 pont (mindkettőn min. 40%)
- Félév végi jegy: 0-39p elégtelen, 40-54p elégséges, 55-69p közepes, 70-84p jó, 85-100p jeles
- Otthoni feldolgozásra kiadott anyagok és Moodle tesztek
 - > A teszttel 3-szor lehet próbálkozni (köztük min. 1 óra idővel), legyen meg 80%-ra, különben mínusz pont. Határidő VASÁRNAP éjfél!
- Munkafolyamat hiba, Moodle teszt hiány: mínusz pont (sorry, “céges” követelmények)
- Tantárgyi adatlap: <https://portal.vik.bme.hu/kepzes/targyak/VIAUBB01/>

MS Teams



Játékszabályok: lásd Culture Deck

Most Valuable Professional verseny

- A legsegítőkésebbek bónusz pontokat kapnak!
- Mindenki 3 szavazatot oszthat szét a félév végén
- A legalább 5 szavazatot kapók között arányosan 200 bónusz pontot osztunk ki.

MVP Most Valuable Professional
High Scores

😊 43 pont

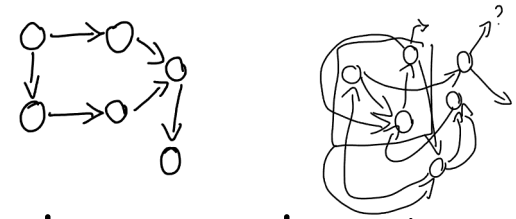
😊 35 pont

😊 22 pont

∴ Mindenki kioszthat 3 pontot (másoknak)

Arányosan
↳ 200 bónusz pont

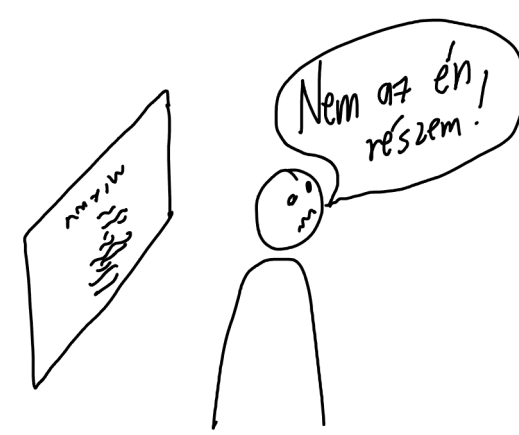
“De hát működik, nem?”



- Egy működő szoftver belülről lehet szép (karbantartható, olvasható, rendezett) és szemétdomb is (amihez egy év múlva senki nem mer hozzányúlni, mert az egész összeomlik).
- A mérnök célja az olvasható, karbantartható kód.
(Szoftver elsődleges és másodlagos értéke...)
- Ezért ha valami működik, akkor még nem biztos, hogy max. pontos. De ebben is fogunk segíteni.

A forráskód olvasása

- Az “ez nem az én részem, nem is értem, hogyan és mit csinál” kijelentés messze van a profi hozzáállástól. (Lehet, hogy aki írta, már nincs a cégnél. Viszont valakinek ki kell javítani!)
- Ezért fontos megtanulni más kódjának olvasását, áttekintését, megértését.
- Sok előadáson és laboron fogunk kész/félkész programok kódjával dolgozni. Pont ezért.
- Kis feladatokon/kódon nem azt tanulod, hogy nagy rendszerben hogyan kell elboldogulni. Ezért lesznek nagyobb programok is.
- Sok kód: az osztályok kis környezetét nézd! Egy osztálynak nincsen túl sok kapcsolata a többiekkel, így nagy kódbázisban is tud egyszerű lenni!



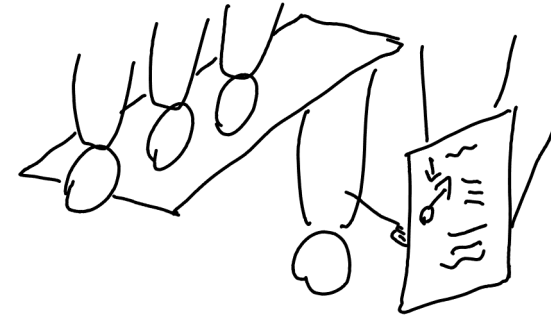
Nem ritka az ismeretlen forráskód...

“...the team figuring out where to start. It’s important for managers to respect this phase. **Teams can’t just dive into a code base and start building new functionality immediately. They have to acquaint themselves with the relevant code, think through the pitch, and go down some short dead ends to find a starting point.**”

<https://basecamp.com/shapeup/3.1-chapter-09/>

Moodle: tanulási útmutató

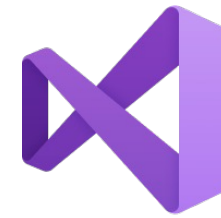
- Flipped classroom
- Otthoni felkészülési anyagok
- Tesztek (határidővel, köztes idővel)
- Minden fontos dolog ott lesz fent, vagy Teams-en az általános csatornában kihirdetve.
- Nem mindenről lesz szó a kontaktórákon.



Problem based learning

- Iparban is munka közben gyakori a stackoverflow.com olvasás.
- Haladunk konkrét feladatokkal és közben sok dolog “előkerül”.
- Ha mással akadsz el, kérdezz!
Órán, Teamsen, stackoverflown...
- Segítünk szívesen, de sokszor nem a megoldást mondjuk meg.
A keresést is tanítjuk, mert az egész életedben kelleni fog!
Lépten-nyomon!

Visual Studio 2019



- Telepítéskor workloadok kiválasztása:
 - > C# .NET
 - > Universal Windows Platform (UWP)
- Hasznos lehet UML diagramokhoz:
 - > VS2019 – Tools – Get Tools and Features – Individual Components – Code tools – Class designer

“A használhatatlan PPT”

A PPT prezentáció nem jegyzet. Kiegészíti az előadást.

A félév időbesztása

Lásd Moodle!

- 1. ZH: 7. hét (okt. 19.), HSZK, labor időpontban
- 2. ZH: 14. hét (dec. 7.), HSZK, labor időpontban

(Pótlások a pótlási héten, pótlási díj mellett.)

Culture Deck



Demók

- AttaxxPlus
- Mandelbrot

+további kis mintaalkalmazások...

összetett != bonyolult 😊