

|  |            |   |                   |                |
|--|------------|---|-------------------|----------------|
| <b>Óbudai Egyetem</b><br>Neumann János Informatikai Kar  |            | Szoftvertervezés és -fejlesztés Intézet               |                   |                |
| <b>Tantárgy neve és kódja:</b> Szoftvertechnológia és grafikus felhasználói interfész tervezése (NIXSG1HBEE, NIXSG1LBEE)   |            | <b>Kreditérték: 5</b>                                 |                   |                |
| <i>Mérnök Informatikus BSc szak</i>  |            | <i>Esti tagozat 2022/23 tanév II. félév</i>           |                   |                |
| Tantárgy oktató(i): Kovács András, Simon-Nagy Gabriella  |            |   |                   |                |
| Előtanulmányi feltételek:<br>(kóddal)  |            | Haladó Fejlesztési Technikák (NSXHF1HBEE, NSXHF1LBEE) |                   |                |
| Heti óraszámok:  | Előadás: 1 | Tantermi gyak.: 0                                     | Laborgyakorlat: 2 | Konzultáció: 0 |
| Számonkérés módja:   | Vizsga     |   |                   |                |
| <b>A tananyag</b>  |            |   |                   |                |
| <i>Oktatási cél:</i> Az előadáson a hallgatóság megismerkedik a szoftverfejlesztés modern eszközeivel és módszertanaival, haladó verziókövetési ismeretekkel és az iparban népszerű tervezési mintákkal. A gyakorlatok során megtanulnak asztali alkalmazásokat fejleszteni WPF keretrendszerben és webes alkalmazásokat fejleszteni Javascript nyelven. |            |   |                   |                |
| <i>Tematika:</i> Az MVVM tervezési minta használata WPF keretrendszerben (vezérlők, események, adatkötés). Egyszerű játékfejlesztés WPF keretrendszerben. Javascript alapjai, DOM manipulációk, események, webapi felhasználás.  |            |   |                   |                |

| Féléves ütemezés:              |   |
|--------------------------------|---|
| Oktatási hét<br>(konzultáció)  | Témakör   |
| 1.                             | Előadás: MVC vs MVVM, UI történet, WPF bevezetés, FF bevezetés<br>Labor: WPF vezérlők és események  |
| 2.                             | Előadás: Projektmenedzsment, hagyományos módszertanok<br>Labor: WPF ablakok és adatkötés  |
| 3.                             | Előadás: Projektmenedzsment, agilis módszertanok<br>Labor: WPF logika leválasztás és commandok  |
| 4.                             | Előadás: GIT branching modellek, UML alapjai<br>Labor: WPF értesítések és templatek   |
| 5.                             | Előadás: UML tervezés<br>Labor: WPF dialógusablakok   |
| 6.                             | Előadás: Újrafelhasználás-orientált fejlesztés, DRY/OOP/SOLID/Clean elvek<br>Labor: WPF összetett üzleti alkalmazás fejlesztése, API elérés |
| 7.                             | Előadás: Tervezési minták (létrehozási minták)<br>Labor: WPF összefoglalás  |
| 8.                             | Előadás: Tervezési minták (viselkedési minták)<br>Labor: Zárthelyi dolgozat   |
| 9.                             | Előadás: Tervezési minták (strukturális minták)<br>Labor: HTML és CSS alapjai / WPF játékfejlesztés   |
| 10.                            | Előadás: Tervezési minták (strukturális minták)<br>Labor: Javascript alapjai / WPF játékfejlesztés  |
| 11.                            | Előadás: Fowler-minták, DDD, CQRS<br>Labor: Javascript DOM manipuláció / WPF játékfejlesztés  |
| 12.                            | Előadás: Mikroszolgáltatások<br>Labor: Javascript API felhasználás / WPF játékfejlesztés  |
| 13.                            | Előadás: Antipatternek<br>Labor: Javító zárthelyi dolgozat  |
| 14.                            | Előadás: Konzultáció<br>Labor: Féléves feladatok prezentálása   |
| <b>Félévközi követelmények</b> |   |

A hallgatók egy zárthelyit írnak (8. heti labor alkalom). Amennyiben a hallgató a zárthelyit nem írta meg, vagy nem teljesítette legalább 50%-os szinten, akkor a 13. heti labor alkalom során a zárthelyiből javító/pótló zárthelyit írhat. A javító zárthelyi sikeres, ha legalább 50%-os szinten teljesíti a hallgató. A zárthelyi eredménye a javító zárthelyi eredménye. A labor zárthelyin érdemjegyet lehet szerezni az alábbi skála szerint: jeles (86-100), jó (74-85), közepes (63-73), elégséges (50-62), elégtelen (0-49).

A zárthelyi dolgozatot megelőző laboralkalmakon (2,3,4,5,6. alkalom) a hallgatók önálló feladatokat oldanak meg. A feladatot a laborvezetőnek óra végéig bemutatják. Egy feladatra 1 pontot, fél pontot illetve 0 pontot lehet szerezni. Egy pont jár, hogyha az elvárások szerint elkészült a feladat. Fél pont jár, hogyha órán nem készült el teljesen, de a következő alkalomig a hallgató bemutatja. 0 pont jár, hogyha nem készült el órán értékelhető (rész)feladat. Ha egy hallgató ezen az 5 alkalmon legalább 3 pontot gyűjtött, akkor a laborvezető ezt az eredményt felajánlja zárthelyi dolgozat eredményként az alábbiak szerint: 3 pont: 70%, 3.5 pont: 75%, 4 pont: 80%, 4.5 pont: 90%, 5 pont: 100%

A felajánlott zh eredményt nem kötelező elfogadni, lehetőség van a zárthelyit megírni mindenkinek.

A 13. héten a hallgató abban az esetben is írhat javító zárthelyit, ha a zárthelyit 50%-os szint felett teljesítette. Ebben az esetben is a végső eredmény a javító zárthelyi eredménye lesz.

A hallgatók a zárthelyi dolgozat megírása után két témakör közül választhatnak: vagy webalkalmazás fejlesztéssel foglalkoznak a félév további részében, vagy játékfejlesztéssel.

Amennyiben egy hallgató webfejlesztést szeretne tanulni, akkor önállóan két rövid házi feladatot kell megvalósítania, melyben a tanult ismeretanyagot kell bemutatnia. Az előkövetelmény tárgyon (Haladó Fejlesztési Technikák) fejlesztett API végponthoz egy WPF nyelvű és egy Javascript nyelvű grafikus felületű kliensalkalmazást kell készítenie. Mindkét házi feladatnak el kell készülnie a 13. hét szerdáján éjfélig. Amennyiben a házi feladatokat nem adja be a fentebb megadott határidőig, illetve az oktató nem fogadja el megfelelő minőségűnek, akkor a hallgató különjárási díj befizetése után a 14. oktatási hét szerdáján, 23:59-ig javíthatja. Az elutasított házi feladatok javítása során a szorgalmi időszakban a hallgatónak egyszer van lehetősége ismét leadni és oktatói értékelést kérni. Ismét elutasított féléves feladatot már csak a vizsgaidőszakban meghirdetett aláíráspótló vizsgán van lehetőség bemutatni. A házi feladatok értékelése bináris: megfelelt/nem felelt meg.

Amennyiben egy hallgató játékfejlesztést szeretne tanulni, akkor 2 további hallgatóval csoportot alkotva egy játékot kell fejlesztenie WPF keretrendszerben vagy laborvezetői engedély alapján egyéb keretrendszerben (pl. Unity). A csoportokat a hallgatók alakítják ki. A játék rövid tervét, funkciólistáját a laborvezetőnek kell leadni, az általa ismertetett módon és határidőre. A játékot magát a 14. heti laboralkalmon kell prezentálni a többiek előtt. Amennyiben a bemutatáskor a programban komolyabb hiba vagy hiányosság van, akkor a bemutató után, a hét végéig (laborvezetővel egyeztetett időpontig) különjárási díjért lehet javítani. A féléves feladat értékelése bináris: megfelelt/nem felelt meg.

Amennyiben a javító zárthelyi dolgozat is lezajlott és a házi feladatok (webfejlesztés) vagy a féléves feladat (játékfejlesztés) pótbemutatása is megtörtént, minden hallgatónak rendelkeznie kell egy legalább elégséges labor zárthelyi eredménnyel és elfogadott feladatokkal. Amennyiben bármely feltétel nem teljesül, akkor aláírást csak a vizsgaidőszakban meghirdetett aláíráspótló vizsgán szerezhet, ahol csak a sikertelen komponenst vagy komponenseket (akár az összeset) kell pótolnia/javítania.

| <b>Zárthelyi dolgozatok</b>  |                               |
|--|-------------------------------|
| Oktatási hét<br>(konzultáció)  | Témakör                       |
| 8  | WPF zárthelyi dolgozat        |
| 13   | WPF javító zárthelyi dolgozat |
| <b>A félévzáró aláírás kialakításának módszere</b>   |                               |
| <p>Aláírást az a hallgató kaphat, aki a zárthelyit legalább 50%-os szinten, valamint a beadandó feladatokat is elfogadott szinten teljesítette.</p> <p>„Letiltva” bejegyzést kap az a hallgató, aki az előadások vagy a laborfoglalkozások több mint 30%-áról hiányzik.</p> <p>„Aláírás megtagadva” bejegyzést kap és aláíráspótláson vehet részt az a hallgató, aki a fent ismertetett módon nem tudta megszereznie az aláírást a szorgalmi időszakban.</p> |                               |
| <b>Pótlás módja</b>  |                               |
| Az aláíráspótláson mindenkinek a nem sikeres komponens vagy komponenseket kell pótolnia (labor zárthelyi, WPF házi feladat és Javascript házi feladat / WPF játék).  |                               |
| <b>Vizsga módja</b>  |                               |
| Az adott vizsgaalcalmon a hallgatók papíros vizsgát írnak oktatói felügyelet mellett. A vizsgán tesztkérdések (15 pont), UML rajzadási feladat (10 pont), kifejtős kérdés (10 pont) illetve tervezési minták felismerése és ismertetése feladat (3x5 pont) alkotja a számonkérést.   |                               |
| <b>Vizsgajegy kialakítása</b>  |                               |
| A vizsgajegybe 25%-os súllyal számítja a félévközi eredmény (WPF zárthelyi dolgozat jegye) 75%-os súllyal az írásbeli vizsga.  |                               |
| Minden tankörből a gyakorlatvezető nevezhet csapatokat egy házi játékkészítő/webfejlesztő versenyre. A versenyen a zsűri első díjakat, második díjakat és harmadik díjakat oszthat ki (többet is akár egy díjból).   |                               |
| Az első díjasok megajánlott jeles végső jegyet kapnak elméleti vizsga nélkül, hogyha az aláírást megszerezték.   |                               |
| A második díjasok WPF eredményét 100%-osnak tekintjük, hogyha legalább elégséges, illetve legalább elégséges írásbeli vizsga esetén a végső jegyet egygel megnöveljük.   |                               |
| A harmadik díjasok a WPF zárthelyi legalább elégséges szintje, sikeres beugró és legalább elégséges szóbeli felelet esetén egy jeggyel jobbat kapnak.  |                               |
| <b>Irodalom</b>  |                               |
| <b>Kötelező:</b>   |                               |
| Előadáson és laborfoglalkozásokon kiadott anyagok  |                               |
| <b>Ajánlott:</b>   |                               |
| Martin Fowler: Patterns of Enterprise Application Architecture, Pearson Education, 2002  |                               |
| Sike Sándor: Szoftvertchnológia és UML, ELTE Eötvös Kiadó, 2003  |                               |
| Ian Sommerville: Software engineering, Pearson Education Limited, 2015   |                               |
| <b>Egyéb segédletek:</b>   |                               |
|  |                               |