Óbuda University John von Neumann Faculty of Informatics	Institute of Software Engineering

Name and code: Advanced programming techniques (NSXHF1EBNE) **Credits:** 4

Computer Science BSc *Daytime 2020/21 year I. semester*

Subject lecturers: Balázs Elemér, Benkő Gábor, Csábrádi Attila, Dr. Erdélyi Krisztina, Haydu Lénárt, Dr. Kertész Gábor, Kovács András, Romhányi Ármin, Sárdi Tamás, Simon-Nagy Gabriella, Sipos Miklós, Szabó-Resch Zsolt,

Prerequisites: (with code)				
Weekly hours:	Lecture: 0	Seminar: 0	Lab. hours: 3	Consultation: 0
Way of assessment:	Midyear grade			

Course description

Goal: Familiarize the students with the advanced topics of C# programming. One lesson from the weekly three is held as a lecture.

Course description: Advanced techniques of the C# language (Lambda expressions, LINQ, Entity Framework, Attributes, Reflection, DLL, Unit tests, Mock, Processes and threads);

Lecture schedule

Education week	Topic
1	*Lecture*: Rules, Delegate/event, Func/Action, Lambda, Logger example *CSharp*: Simple FeedbackProcessor
2	*Lecture*: Extension methods, LINQ, XML/JSON, XLINQ/JsonLINQ, People.xml *CSharp*: XLINQ War on westeros.xml
3	*Lecture*: Dotnet execution/DLL, Reflection, XmlSerializer *CSharp*: Reflection / Validator
4	*Lecture*: Layering, ORM, SQL-first vs Code-first, EF EMP demo (sql first) *CSharp*: DbContext, Entities, Db Seed, Select/NonCrud example (code first) *Project*: (Oct/01) Bitbucket/atlassian account, git repository, readme.md
5	*Lecture*: Project expectations, Layering example, Repository pattern *CSharp*: Project example: One SLN, Multiple CSPROJ, DB access, Menu
6	*Lecture*: GIT quickstart + DLL (chaser) via GIT *CSharp*: ZH practice *Project*: (Oct/12) Solution with empty projects
7	*Lecture*: Dotnet GC, Dotnet versions *CSharp*: ZH (together, outside regular schedule) *Project*: (Oct/19) Database + dbcontext + entities
8	*Lecture*: Unit tests, theory, simple example *CSharp*: Unit tests using nUnit *Project*: (Oct/30) Menu + All list operations
9	*Lecture*: Unit tests using MOQ *CSharp*: Project review
10	*Lecture*: Processes, IPC (theory + examples) *CSharp*: Project review *Project*: (Nov/15) All crud+noncrud functionality
11	*Lecture*: Threads 1 (Thread+Task syntax and examples) *CSharp*: Thread/Task usage
12	*Lecture*: Threads 2 (Task extras, synchronization) *CSharp*: Thread synchronization
13	*Lecture*: Layered architectures closing thoughts *CSharp*: Project work consultation *Project*: (Dec/06) Gitstat deadline
14	*CSharp*: ZH retakes (together, outside regular schedule) *Project*: Project work defense

Midterm requirements

Attendance on the practices is obligatory. Before the practices the students must watch the lecture videos.

The students will write one mid-semester test, writing the mid-semester test is obligatory. If a student doesn't write or doesn't pass the mid-semester test, a re-test is possible on the last week. A re-test is also possible if a student wants to re-write one of the tests. In this case, it is always the re-test grade is taken into account.

If a student doesn't have a passed test at the end of the regular semester, a last re-test is possible in the exam season.

The students have to create a project work on their own, that shows their advanced C# programming skills:

- The expectations must be met that are mentioned in the prog3_layers_requirements document:
- Stylecop/Fxcop-valid code, Doxygen developer documentation
- Single-user, single-branch GIT repository
- Usage of a database + Entity Framework to access it
- Usage of LINQ
- Layered architecture (minimum 5 sub-projects: Console app + business logic DLL + repository DLL + data access DLL + Unit test DLL)
- Unit tests (typically for the logic classes)

The project work has to be submitted until 06/DEC 23:59. If that deadline is not met, or the teacher doesn't accept the quality of the project work or it doesn't fulfil the bare minimum expectations listed above, then the student will have to present their project work in the exam season.

Midterm Test Scheduling

Education week	Topic
7	FIRST MIDTERM
14	RE-TEST if necessary

Midterm grade calculation methods

Félévközi jegyet az a hallgató kaphat, aki a zárthelyit legalább 50%-os szinten, valamint a beadandó feladatot megfelelően teljesítette. A hallgató érdemjegye a zárthelyi jegye lesz, amit a projektfeladat egy jeggyel ronthat, illetve javíthat.

"Letiltva" bejegyzést kap az a hallgató, aki a laborfoglalkozások több mint 30%-áról hiányzik. Szintén "Letiltva" bejegyzést kap az a hallgató, aki egyáltalán nem adott le féléves feladatot.

"Elégtelen" bejegyzést kap és félévközi jegy pótláson vehet részt az a hallgató, aki a fentiek alapján nem teljesíti a zárthelyit 50%-os szinten, vagy a féléves feladatra kapott osztályzata elégtelen.

Method of replacement

In a mid-semester re-test, the test can be rewritten.

In the exam season re-test, students will have to solve a combined task from the FULL contents of the semester.

Presentation of the project works will be done afterwards, if necessary.

Type of exam

Exam grade calculation methods

Mid-semester grade can only be given to a student who passed all two tests and who submitted an accepted project work. The grade will be the test grade, that can be changed with one grade by the project work.

"Signature refused" entry will be given to any student who misses more than 30% of the practice sessions. (TVSZ 23.§). Also "Signature refused" entry will be given to any student who did not submit a project work.

"Failed" grade will be given to any student who doesn't have the successful practice test or the project grade is "Fail".

References

Obligatory:

Lab presentations, practice materials
https://users.nik.uni-obuda.hu/prog3/

Recommended:

Others: