Proposed topics for laboratory

No.	Tutorial name	Chapters to be contained (not limited to…)	Additional information
1	MVC	a) Setting up the dev environment b) Fundamentals c) Connecting to backend d) Building forms e) Data Validations f) Authentication and Authorization g) Deployment	Entity framework(AdventureWorks/Northwind database) 2 students
2	React (Hook and Redux)	a) The basics b) Setting up the dev environment c) Form Validation d) Routing e) Working with Lists f) Authentication g) Using Hooks h) Testing i) Deployment	AdventureWorks/Northwind database 2 students
3	Angular (9)	j) The basics (data, property, event binding), directives k) Setting up the dev environment l) Services and dependency injection m) Handling forms n) Http requests o) Authentication and route protection p) Deployment	AdventureWorks/Northwind database 2 students
4	Vue	a) The basics b) Setting up the dev environment c) Using and creating directives d) Routing in Vuew e) Form Validation f) Authentication g) Deployment	AdventureWorks/Northwind database 2 students

5	Blazor	a) Setting up the environment b) Introduction to Blazor c) Introduction to Razor d) Components e) Routing f) Forms g) Backend – Entity Framework Core h) Security i) Deployment	AdventureWorks/Northwind database 2 students
6	Universal Windows Platform	a) Setting up the environment b) Basics of UWP c) Data and Async Tasks d) Navigation e) UWP Architecture f) Controls g) App lifecycle h) Capabilities i) Deployment	AdventureWorks/Northwind database 2 students
7	Rust	a) Setting up the environment b) Language Fundamentals c) Types and Variables d) Control Flow e) Data structures f) Functions g) Traits h) Lifetime and memory i) Circular references j) Testing	AdventureWorks/Northwind database 2 students
8	Scala programming language	a) Setting up the environment b) Basics of Scala c) OOP in Scala d) Functional programming in Scala e) Pattern matching f) Deployment	2 students

9	GoLang	a) Setting up the dev environment b) Programming fundamentals c) Grouping data d) Structs, pointers, functions, grouping data e) Channels f) Error handling g) Testing	2 students
10	Any other Functional Programming Language		
11	SOLID principles	a) Single-responsibility Principle b) Open-closed Principle c) Liskov substitution principle d) Interface segregation principle e) Dependency Inversion principle	.NET, C# 1 student
12	Design Patterns 1 (Structural Patterns)	a) Adapter b) Bridge c) Composite d) Decorator e) Façade f) Flyweight g) Proxy	.NET, C# 1 student
13	Design Patterns 2(Behavioral Patterns)	a) Chain of Resp. b) Command c) Interpreter d) Iterator e) Mediator f) Memento g) Observer h) State i) Strategy j) Template Method k) Visitor	.NET, C# 1 student

14	Design Patterns 3(Creational Patterns)	a) Abstract Factory b) Builder c) Factory Method d) Prototype e) Singleton	.NET, C# 1 student
15	Design Patterns 4	a) Layered architecture b) Event-driven architecture c) Microkernel architecture d) Microservices architecture e) Space-based architecture	.NET, C# 1 student
16	Micro services		.NET, C#, AdventureWorks/Northwind database 2 students
17	Web services vs Web API		.NET, C#, AdventureWorks/Northwind database 1 student
18	Async Programming		.NET, C# 1 student
19	Blockchain		
20	Service based (SOA) vs microservces		2 students
21	Messaging Hubs(SignalR)		.NET, C# 2 students
22	Messaging Hubs(RabbitMQ)		Java 2 students
23	Scaled Agile vs Agile	a) In depth description of Agile methodology b) In depth description of SAFE c) Comparison and differences	1 student
24	Micro Front Ends		2 students
25	Python	a) Python setup b) Object and data structures c) Operators, statements, methods and functions d) OOP in Python e) Error handling f) Python Decorators and generators g) Advanced modules h) Advanced objects and structures	2 students

26	Entity Framework using Code-First workflow	a) Basic information and introduction b) Lazy loading c) Eager loading d) Explicit loading e) LINQ to query data f) Working with data (add, remove, update)	.NET, C# 2 students
27	Entity Framework using Database-First workflow	a) Basic information and introduction b) Lazy loading c) Eager loading d) Explicit loading e) LINQ to query data Working with data (add, remove, update)	.NET, C# 2 students
28	ASP.Net Core MVC With Entity Framework Core	Just like the normal MVC project only that is targeting .NET Core	2 students
29	Flutter Development	https://www.udemy.com/course/flutter-bootcamp-with-dart/	Mobile development 2 students
30	C# Unity development 2D		2 students
31	C# Unity development 3D		2 students
32	Complete JavaScript Course	a) Setting up the environment b) JavaScript Basics c) JavaScript objects and functions d) DOM manipulation and Events e) Async JavaScript f) ES6	2 students
33	Web Development 2020	a) Introduction to HTML (Basic and Internediate) b) Introduction to CSS (Basic and Internediate) c) Introduction to Boostrap(Basic and Internediate) d) Introduction to JavaScript(Basic and Internediate) e) NodeJS … … Syllabus recommendation: https://www.udemy.com/course/the-complete-web-development-bootcamp/	3 students

34	Guide to Git	a) Introduction b) Installation c) Basic Git commands d) Visual merge. Diff tool e) Comparisons f) Branching and merging g) Rebasing h) Stashing i) Tagging	2 students
35	NodeJS applications	Follow the Syllabus from: https://www.udemy.com/course/the-complete-nodejs-developer-course-2/	2 students
36	MongoDB databases	Follow the Syllabus from: https://www.udemy.com/course/mongodb-the-complete-developers-guide/	2 students
37	Unit testing for C#	Follow the course content from: https://www.udemy.com/course/unit-testing-csharp/	1 student
38	Unit testing with NUnit and Moq	https://www.udemy.com/course/nunit-moq/	1 student
39	Appium -Mobile Automation Testing	Follow the syllabus from: https://www.udemy.com/course/mobile-automation-using-appiumselenium-3/	1 student
40	Robot Framework Test Automation	https://www.udemy.com/course/robot-framework-level-1/ https://www.udemy.com/course/robot-framework-2/	1 student
41	Selenium WebDriver - Java, Cucumber BDD	Follow the following syllabus: https://www.udemy.com/course/cucumber-bdd-selenium-java-complete-automation-course/	1 student

1. Each student/team must inform in any mean possible (email address, verbal, etc.) about the selected topic.
2. Each team has a maximum 10 weeks(incl. the holiday) to develop the required material for the selected topic.
3. The required material is the following:
    a) A docx. Document about the selected topic, with respect to the syllabus indicated in the current document. The team is not restricted to the proposed syllabus and is encouraged to develop more than indicated in the 3rd column
    b) A Git address containing all the source code(if any), or a USB stick, CD, WeTransfer, with all the seeds and code evolution
    c) A presentation no longer than 15 minutes that will be presented in front of the other students
4. Language for documentation and coding is ENGLISH
5. Please be aware that the documentation will be the subject of Plagiarism detection and documentation with a higher than 25% possible plagiarism will need to be modified
6. Language for c) – presentation can be Romanian
7. Due date for submitting a) and b) is 8 May 24:00 – after this date the mark is downgraded by 2 points from the final grade

Presentation of c) will take place in weeks 13 & 14 (25-29May / 1-5 June)

The Grading formula is as it follows: 80%*(**a & b**) + 20%*(**c**)