

## Okvirni plan i program studentske prakse

Tjedan	Ponedjeljak	Utorak	Srijeda	Četvrtak	Petak
1.	<b>Upoznavanje i uvod u OOP</b> kratko ponavljanje osnovnim principa objektno-orijentiranog programiranja	<b>WebAPI</b> HTTP protokol, Request/response, CRUD <b>GIT</b>	<b>Baze podataka</b> ponavljanje osnovnih pojmova o bazama podataka i relacija među bazama, upoznavanje sa Ado.Net-om	<b>Multilayer arhitektura</b> Upoznavanje sa pojmom multilayer arhitekture	<b>Asinkrono programiranje</b> Upoznavanje sa pojmom i primjena <b>Razrada projektnog zadatka</b>
2.	<b>Dependency injection</b> Upoznavanje sa pojmom DI-a, kako implementirati DI te upoznavanje sa Autofac-om			<b>Izrada backend dijela projektnog zadatka</b>	
3.	<b>Izrada backend dijela projektnog zadatka</b>				
4.	<b>Izrada backend dijela projektnog zadatka</b>		<b>Uvod u React</b> Upoznavanje s osnovnim konceptima JavaScripta/Reacta	<b>Class components/function components</b>	<b>Mobx</b> -state management
5.	<b>Izrada frontend dijela zadatka</b>				<b>Završetak projekta i prezentiranje projekta</b>

Pozdrav studenti,

Šaljem vam okvirni plan i program prakse koja kreće u ponedjeljak, 1. veljače 2021. godine. Obzirom na trenutnu situaciju u kojoj se nalazimo, praksu ćemo održati online. Plan je okviran jer ovisi o vašem napretku. Ukoliko budemo imali teškoća sa backend dijelom zadatka, možda će izrada tog dijela zadatka trajati dulje, ali može i kraće (ukoliko backend dio zadatka bude dobro napredovao). Ovo je samo kratki uvid u ono što ćemo sve raditi.

Za komunikaciju ćemo koristiti alate kao što su **Zoom**, **Skype** (ako nemate račun, svakako ga kreirajte) te **Slack** (instalirajte si i tu aplikaciju do ponedjeljka, a ja ću vam poslati pozivnicu na mailove s kojih ste se prvotno javili). Praksa će trajati 5 tjedana tj. 25 radnih dana po 8 h dnevno. Planirani završetak prakse je 5.3.2021.

U ponedjeljek, 1.2.2021. ćemo se upoznati putem aplikacije Zoom te vas molim da se uključite u poziv u 8:00h ujutro. Pozivnice za meeting ću vam poslati putem maila. Na prvom sastanku ćemo se upoznati s alatima koje ćete koristiti, planom i programom i ponoviti prethodna znanja iz OOP-a. Praksa je zamišljena tako da u prvih tjedan dana dobijate teme (u koje ću vas ja na početku dana kratko uputiti) te ćete istraživati o navedenoj temi i izraditi jedan jednostavni projekt na navedenu temu. Prvih tjedan dana ćete raditi samostalno kako biste na kraju tjedna mogli grupno krenuti u izradu konkretnog projektnog zadatka.

Sa samim projektnim zadatkom ćemo se upoznati odmah na početku, ali željela bih da svakako razmislite o projektnoj temi. Ukoliko budete imali neku želju/ideju i ostatak ekipe se s time bude slagao, možemo projektni zadatak usmjeriti prema toj temi (ne mora biti tema koju ćemo vam mi predložiti).

Za ponedjeljak bih željela da instalirate sljedeće programe na svoja računala (ukoliko ih još nemate):

1. **Visual Studio 2019** (community verziju je moguće skinuti ukoliko imate vaše ettaedu podatke s faxa)
2. **Visual studio Code**
3. **SourceTree**
4. **Postman**

Ukoliko nemate račun na **GitHubu**, svakako ga kreirajte jer će vam biti potrebno.

U nastavku šaljem popis literature koja vam može pomoći u učenju i rješavanju projektnog zadatka. Neka vas ne plaši količina literature, navedeni su različiti primjeri (knjive, tutoriali, video materijali, blogovi) kako bi svatko pronašao materijal koji njemu najviše odgovara.

**Čujemo se u ponedjeljak u 8 h!**

## Git

- General - <https://git-scm.com/doc>
- How Git works: <https://app.pluralsight.com/library/courses/mastering-git>
- GitHub for Windows Users - [https://mva.microsoft.com/en-US/training-courses/github-for-windows-users-16749?l=KTNeW39wC\\_6006218965](https://mva.microsoft.com/en-US/training-courses/github-for-windows-users-16749?l=KTNeW39wC_6006218965)
- Git Succinctly - <https://www.syncfusion.com/ebooks/git>
- GitFlow - <https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow>

## C#

- C# Fundamentals for Absolute Beginners - [https://mva.microsoft.com/en-US/training-courses/c-fundamentals-for-absolute-beginners-16169?l=Lvld4EQIC\\_2706218949](https://mva.microsoft.com/en-US/training-courses/c-fundamentals-for-absolute-beginners-16169?l=Lvld4EQIC_2706218949)
- C# Programming Yellow Book - <http://www.robmiles.com/c-yellow-book/>
- Naming guidelines - <https://docs.microsoft.com/en-us/dotnet/standard/design-guidelines/naming-guidelines>
- C# tutorial for beginners - <https://www.youtube.com/playlist?list=PLAC325451207E3105>
- Programming with C# - Objective 1.1 Multithreading and asynchronous processing - <https://www.youtube.com/watch?v=RhLLxew4-TY>

## OOP via C#

- General - <http://www.c-sharpcorner.com/UploadFile/84c85b/object-oriented-programming-using-C-Sharp-net/>
- General - <https://www.youtube.com/watch?v=e7Yj6vLyYOI>

## Async/await

- <https://msdn.microsoft.com/library/hh191443%28vs.110%29.aspx?f=255&MSPPErr=-2147217396>
- <https://blog.stephencleary.com/2012/02/async-and-await.html>

## Architecture

- Design Patterns - <http://www.dofactory.com/net/design-patterns>
- Design Patterns in general: <https://app.pluralsight.com/library/courses/patterns-library>
- Composition over inheritance - [http://en.wikipedia.org/wiki/Composition\\_over\\_inheritance](http://en.wikipedia.org/wiki/Composition_over_inheritance)
- Multilayered architecture
  - General - [http://en.wikipedia.org/wiki/Multilayered\\_architecture](http://en.wikipedia.org/wiki/Multilayered_architecture)
  - Onion architecture- <http://jeffreypalermo.com/blog/onion-architecture-part-4-after-four-years/>
  - Multilayered architecture - <http://www.codeproject.com/Articles/552487/Understanding-Multilayered-Architecture-in-NET>
  - Multilayered Architectures in .Net-Web - <http://mono.hr/pdf/Multi-Layered%20Architectures%20in%20.Net-Web.pdf>
  - Multilayered Architectures in .Net Workshop-Web - <http://mono.hr/pdf/Multi-layered%20Architectures%20in%20.Net%20Workshop-Web.pdf>
  - Example: <https://github.com/MonoSoftware/DIPracticeLayeredArchitecture>

## Dependency Injection

- General
  - <https://docs.microsoft.com/en-us/aspnet/core/fundamentals/dependency-injection>
  - <https://msdn.microsoft.com/en-us/library/ff921152.aspx>
- (Ninject based) - <http://www.mono.hr/Pdf/Dependency-Injection-in-practice-CodeCAMP.pdf>
- (AutoFac based) - <https://autofac.readthedocs.io/en/latest/getting-started/index.html>
- Examples: <https://github.com/MonoSoftware/DependencyInjectionInPractice> (Follow tags for step by step guide)
  - Dependency Injection in ASP.NET Core 2.0 - <https://code.msdn.microsoft.com/Dependency-Injection-in-ce9c8edb>
  - Dependency Injection - <https://channel9.msdn.com/Shows/Visual-Studio-Toolbox/Dependency-Injection>

## ORMs

- SQL
  - SQL Database Fundamentals - [https://mva.microsoft.com/en-US/training-courses/sql-database-fundamentals-16944?l=w7qq6nAID\\_6805121157](https://mva.microsoft.com/en-US/training-courses/sql-database-fundamentals-16944?l=w7qq6nAID_6805121157)
  - Introduction to SQL - <https://app.pluralsight.com/library/courses/introduction-to-sql>
  - SQL Queries Succinctly - [https://www.syncfusion.com/ebooks/sql\\_queries\\_succinctly](https://www.syncfusion.com/ebooks/sql_queries_succinctly)
- EF (Entity Framework)
  - Entity Framework Core - [https://www.syncfusion.com/ebooks/entity\\_framework\\_core\\_succinctly](https://www.syncfusion.com/ebooks/entity_framework_core_succinctly)
  - Home of EF - <http://www.asp.net/entity-framework>
  - An Introduction to Entity Framework with Rowan Miller - <https://channel9.msdn.com/Blogs/Seth-Juarez/An-Introduction-to-Entity-Framework-with-Rowan-Miller>

## Repository Pattern

- Repository pattern, done right - <http://www.codeproject.com/Articles/526874/Repositorypluspattern-cplusedoneplusright>
- Repository and Unit of Work Patterns - <http://www.asp.net/mvc/overview/older-versions/getting-started-with-ef-5-using-mvc-4/implementing-the-repository-and-unit-of-work-patterns-in-an-asp-net-mvc-application>
- Unit of Work - a Design Pattern - <https://mono.software/2017/01/13/unit-of-work-a-design-pattern/>

## Entry level books (ASP.NET MVC, ASP. NET Web API, Entity framework, Angular JS ...)

- <https://www.syncfusion.com/resources/techportal/ebooks>

## ASP.NET MVC & WebAPI

- ASP.NET Core - Beginner - [https://mva.microsoft.com/en-US/training-courses/aspnet-core-beginner-18153?l=VM5gy36dE\\_6611787171](https://mva.microsoft.com/en-US/training-courses/aspnet-core-beginner-18153?l=VM5gy36dE_6611787171)
- ASP.NET Core API - <https://app.pluralsight.com/library/courses/asp-dotnet-core-api-building-first>
- Little ASP.NET Core Book - <https://s3.amazonaws.com/recaffeinate-files/LittleAspNetCoreBook.pdf>
- ASP.NET WebAPI <https://www.syncfusion.com/resources/techportal/details/ebooks/webapi>

## HTML

- HTML5 & CSS3 Fundamentals: Development for Absolute Beginners - [https://mva.microsoft.com/en-US/training-courses/html5-css3-fundamentals-development-for-absolute-beginners-14207?l=Y4COscFfB\\_7500115888](https://mva.microsoft.com/en-US/training-courses/html5-css3-fundamentals-development-for-absolute-beginners-14207?l=Y4COscFfB_7500115888)
- How to Code in HTML5 and CSS3 - <http://howtocodeinhtml.com/>
- HTML, CSS, JS - <https://app.pluralsight.com/library/courses/html-css-javascript-big-picture/>

## JavaScript

- JavaScript Fundamentals for Absolute Beginners - [https://mva.microsoft.com/en-US/training-courses/javascript-fundamentals-for-absolute-beginners-14194?l=L4YIbtUxE\\_900115881](https://mva.microsoft.com/en-US/training-courses/javascript-fundamentals-for-absolute-beginners-14194?l=L4YIbtUxE_900115881)
- Pluralsight Javascript Path, you can do the Javascript Skill IQ test beforehand - <https://app.pluralsight.com/paths/skills/javascript>
- JavaScript Succinctly - <https://www.syncfusion.com/ebooks/javascript>
- Eloquent JavaScript - [https://eloquentjavascript.net/Eloquent\\_JavaScript.pdf](https://eloquentjavascript.net/Eloquent_JavaScript.pdf)

- How to Learn JavaScript Properly - <http://javascriptissexy.com/how-to-learn-javascript-properly/>
- OOP:
  - Objects - <http://javascriptissexy.com/javascript-objects-in-detail/>
  - Prototype - <http://javascriptissexy.com/javascript-prototype-in-plain-detailed-language/>
  - OOP Concepts - <http://javascriptissexy.com/oop-in-javascript-what-you-need-to-know/>
- Advanced:
  - Variable scope - <http://javascriptissexy.com/javascript-variable-scope-and-hoisting-explained/>
  - Closures - <http://javascriptissexy.com/understand-javascript-closures-with-ease/>
  - this keyword - <http://javascriptissexy.com/understand-javascripts-this-with-clarity-and-master-it/>
  - Callback - <http://javascriptissexy.com/understand-javascript-callback-functions-and-use-them/>
  - Method chaining - <http://javascriptissexy.com/beautiful-javascript-easily-create-chainable-cascading-methods-for-expressiveness/>
  - Apply, Call, Bind methods - <http://javascriptissexy.com/javascript-apply-call-and-bind-methods-are-essential-for-javascript-professionals/>
  - Promises - <https://medium.com/@bluepnume/learn-about-promises-before-you-start-using-async-await-eb148164a9c8>
- Miscellaneous
  - 1. Simple tips - <http://javascriptissexy.com/12-simple-yet-powerful-javascript-tips/>
  - 2. Intermediate and advanced - <http://javascriptissexy.com/learn-intermediate-and-advanced-javascript/>

## ReactJs

- React.js Succinctly - [https://www.syncfusion.com/ebooks/reactjs\\_succinctly](https://www.syncfusion.com/ebooks/reactjs_succinctly)
- Pluralsight React path, you can do the React Skill IQ test beforehand - <https://app.pluralsight.com/paths/skills/react>
- <https://app.pluralsight.com/library/courses/react-redux-react-router-es6>
- The Road to Learn React - <https://leanpub.com/the-road-to-learn-react>
- Awesome React - <https://github.com/enaqx/awesome-react>
- <https://reactjs.org/tutorial/tutorial.html>
- <https://reactjs.org/docs/hello-world.html>
- <https://reactjs.org/docs/introducing-jsx.html>
- <https://reacttraining.com/react-router/web/guides/philosophy>
- <https://github.com/enaqx/awesome-react#react-general-tutorials>
- <https://mobx.js.org/> or <https://redux.js.org/introduction>
- <https://scrimba.com/g/glearnreact>