**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU**

**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA**

**Diplomski studij**

**ENASTAVNIK**

**Diplomski rad**

**Student: Mato Lučić, br. indeksa Axxxx**

**mlucic@etfos.hr**

**Mentor: Doc.dr.sc. Alfonzo Baumgartner**

**Osijek, Lipanj 2017.**

# Uvod

Tema diplomskog rada je napraviti interaktivnu aplikaciju namjenjenu nastavnicima, učenicima i roditeljima za vođenje evidencije o uspjehu učenika. Nastavnici trebaju imati mogućnost prilagodbe aplikacije svojim potrebama i sve mogućnosti administratorskog pristupa. Učenici mogu imati pristup pregledu unešenih podataka koji su njihovi te pristup kvizovima znanja iz pojedinog predmeta. Roditelji mogu pristupiti samo unešenim podatcima u bazu te pregledati ocjene, opravdane sate, neopravdane sate i rezultate kvizova znanja.

Aplikacija je napravljena u web tehnologijama. Programski jezici i web tehnologije korišteni za ovu aplikaciju su: C#, ASP.NET, AngularJS, HTML, JavaScript, CSS, SQL. Korištenje navedenih web tehnologija omogućio je programski paket Visual Studio. Uvid u bazu podataka te manipulaciju podatcima olakšao je SQL Server Managment Studio. Provjeru pozadinskog odvijanja rada sa podatcima omogućuje programski paket Postman.

U narednim poglavljima biti će objašnjene navedene tehnologije, njihova upotreba u realizaciji web aplikacije.

## C# programski jezik

C# je programski jezik koji je dizajniran za izgradnju aplikacija koje se pokreću na .NET platformi. C# je jednostavan, siguran i moćan objektno orijentirani programski jezik. Razvijen je 2000. godine od strane tvrtke Microsoft u laboratoriju pod vodstvom Andersa Hejlsberga, Scotta Wilatmutha i Petera Goldea. C# je potpuno objektno orijentirani jezik što znači da svi elementi unutar njega prestavljaju objekt. Objekt predstavlja strukturu koja sadrži podatkovne elemente, kao i metode i njihove međusobne interakcije. Iduća bitna stavka koja je implementirana u C# programsku jeziku je generičko programiranje. Pojam označava stil programiranja u kojem se pišu algoritmi uz uvjet da se tip podataka specificira prilikom implementacije algoritma ovisno o tipu parametara koji su potrebni algoritmu. Generičko programiranje i generički tipovi podataka su uvedeni da se spriječi dupliciranje koda jer se kreira samo jedna jedinstvena metoda ili klasa koja može biti bilo kojeg tipa prilikom implementacije. Microsoft je od 2014. godine omogućio programerima da do tada korišteni kompajler koji je kompajlirao C# kod, kodnog imena Rosyn preoblikuje u kompajler koji će biti open-source. Taj novi kompajler je dobio ime C# tako da danas C# kod kompajlira njegov kompajler (C#) što uvelike olakšava te ubrzava kompajliranje, pronalaženje greški u kodu, omogućuje brži rad i slično. Kako je rečeno u gornjem tekstu C# je objektno odijentirani programski jezik kojeg prate paradigme poput klasa, objekt, metoda, varijable, enkapsulacije, nasljeđivanja, apstraktne klase i interfejsi.

Klasa je osnovna programska cjelina svakog objektno orijentiranog programskog jezika. Ona predstavlja generalizaciju pojma strukture iz programskog jezika C. Kao i struktura, klasa je samo nacrt po kome se strukturira objekt. Konstruira se ključnom riječju *class*.

Objekt nastaje instanciranjem klase, odnosno stvaranjem varijable koja u sebi drži vrijednost napisane klase te omogućuje pristup svim varijablama i metodama koji su u klasi.

Metoda je procedura dodjeljena klasi objekta. Omogućuje objektu da odradi funkciju koja je potrebna u aplikaciji.

Varijabla je memorijska lokacija čiji se sadržaj može mijenjati tokom izvođenja programa. Svaka varijabla u programu predstavlja memorijsku lokaciju i memorijsko zauzeće. Zauzeće ovisi o tipu varijable koji se može mijenjati. Varijablu se može zamisliti kao veliku ladicu gdje se mogu staviti stvari pod uvjetom da su istog tipa.

Enkapsulacija znači pravo pristupa svojstava i metoda unutar neke klase. Postoji private, public i protected pristup. Private pristup dozvoljava pristup samo unutar klase u kojoj se varijaba ili metoda nalazi. Public omogućuje pristup varijabli, metodi iz bilo koje klase. Protected dozvoljava pristup unutar klase u kojoj se nalazi varijabla ili metoda i klasi koja nasljeđuje tu klasu.

Nasljeđivanje omogućuje ponovnu upotrebljivost koda ovisno o pravu pristupa (enkapsulaciji).

Apstraktne klase služe kao opis ponašanja klase koja ih nasljeđuje. Sadrže metode koje će se implementirati u klasama koje ih nasljeđuju. Kreiranje instance apstraktne klase nije moguće. Apstraktna klasa se kreira ključnom riječju abstract.

Interfesi su kosturi klasa koje ih nasljeđuju. Unutar interfejsa se definiraju metode i prava pristupa za svaku metodu implementacijom interfejsa potrebno je nasljediti sve metode definirane u interfejsu.

## ASP.NET

ASP.NET je '.NET' (eng. *Dot-Net*) bazirano okruženje, gdje se aplikacije mogu razvijati u bilo kojem .NET kompatibilnom jeziku uključujući C#, VB.NET i JScript.NET. Drugim riječima, cijeli '.NET Framework' je na raspolaganju bilo kojoj ASP.NET aplikaciji. Programeri mogu lako iskoristiti pogodnosti ove tehnologije, koja uključuje CLR (eng. *Common Language Runtime*) okruženje, nasljeđivanje i slično.

ASP.NET je dizajniran da radi s 'WYSIWYG' (eng. *What You See Is What You Get*) HTML editorima i drugim alatima, uključujući Visual Studio .NET koji ne samo da značajno olakšava web programiranje već omogućava i korištenje svih pogodnosti koje on pruža, uključujući razvoj web aplikacija u grafičkom okruženju.

Za razliku od klasičnog ASP-a, ASP.NET programski model je više objektno orijentiran što ga dodatno približava standardnim objektno orijentiranim razvojnim okolinama i jezicima kao što su C++, C#, VB i slično.

ASP.NET se temelji na MVC pristupu te aplikaciji omogućuje olakšani pristup bazi pomoću Entity Frameworka i dohvaćanje podataka pomoću Web API-ja.

### MVC

MVC je kratica za generalnu strukturu .NET aplikacije i glasi „Model-View-Controller“.

Model – domena oko koje je sagrađna aplikacija. Zasnovan na stvarnom slijedu i njegovim objektima. Primjer je preslike baze podataka.

View – vizualni prikaz danog modela u samom kontekstu. Realiziran ne preko HTML-a, CSS-a, JavaScripta.

Controller – koordinator koji je veza između Viewa i Modela. Realiziran je u C# programskom jeziku. Korisniku omogućuje pristup bazi.

### Entity Framework

Entity Framework (EF) je preporučena tehnologija pristupa podacima za aplikacije u .NET-u. To je objektno-relacijski maper koji omogućuje .NET programerima rad s relacijskim podacima pomoću objekata specifičnih za pristup bazi podataka. Programerima omogućuje olakšan pristup bazi podataka sa manje pisanja kooda, ulaska u logiku rada sa bazom te prijenosom podataka i slično.

Entity Framework omogućuje dva pristupa podatcima: Code first i Database first.

Code first je takav pristup u kojem se stvaraju modeli kao reprezentativni primjer baze podataka te Entity Framework s obzirom na modele stvara bazu podataka. Modelima je potrebno definirati i vezu između modela kako bi tablice mogle dobro funkcionirati i stvorile logički pristup podatcima. U koliko određena klasa treba imati pristup većoj količini podataka druge klase, koriste se kolekcije objekata, a ukoliko je potrebno pristupati samo određenom podatku iz klase onda je dovoljno u klasi navesti samo objekt klase kao atribut.

Database first pristup je obrnuti pristup gdje se stvore tablice u bazi podataka koje Entity Framework porepozna i izgenerira modele.

U ovoj aplikaciji je korišten Code first pristup.

### 1.1.3. WebAPI

WebAPI okruženje se koristi za izradu RESTful aplikacija. Korištenjem ovog okruženja omogućujemo upravljanje HTTP zahtjevima. WebAPI nam omogućuje komunikaciju između sučelja koji je namijenjen korisniku prilikom upravljanja aplikacije i pozadinskog dijela aplikacije koji manipulira podatcima. Pozadinski dio zovemo backand, a dio namijenjen korisniku za interakciju sa bazom zovemo frontend.

## AngularJS

AngularJS je JavaScript MVC framework napravljen od strane Google-a koji omogućava stvaranje dobro struktuiranih, lakih za testiranje i održavanje frontend aplikacija.

AngularJS je MVC framework koji definira brojne koncepte za pravilno organiziranje web aplikacije.

Aplikacija je napravljena uz pomoć MVC modula koji su spomenuti u poglavlju 1.2.1. AngularJS proširuje HTML pružajući [direktive](https://docs.angularjs.org/guide/directive) oznakama pomoću kojih se može izgraditi moćna i dinamična web aplikacija. Uz HTML, omogućuje i CSS stilske alate za oblikovanje i uređivanje HTML-a.

## SQL

**Structured Query Language** (SQL) je najpopularniji [računalni jezik](https://hr.wikipedia.org/wiki/Ra%C4%8Dunalni_jezik) za izradu, traženje, ažuriranje i brisanje podataka iz [relacijskih baza podataka](https://hr.wikipedia.org/w/index.php?title=Relacijska_baza_podataka&action=edit&redlink=1). SQL je stardardiziran preko standarda [ANSI](https://hr.wikipedia.org/wiki/ANSI) i [ISO](https://hr.wikipedia.org/wiki/ISO).

# 2.Glavni dio