Sadržaj

[1. Uvod 2](#_Toc195579467)

[2. Opis korištenih tehnologija 4](#_Toc195579468)

[2.1 HTML, CSS i JavaScript 4](#_Toc195579469)

[2.2 PHP 5](#_Toc195579470)

[2.3 XAMPP 5](#_Toc195579471)

[2.4 MySQL 6](#_Toc195579472)

[3. Arhitektura baze podataka 7](#_Toc195579473)

[3.1 Logička razina 7](#_Toc195579474)

[3.2 Fizikalna razina 8](#_Toc195579475)

[4. Modeli baze podataka 9](#_Toc195579476)

# Uvod

Knjižnica je ustanova koja prikuplja, čuva, obrađuje i omogućuje pristup različitim vrstama pisanih, tiskanih i elektroničkih izvora znanja. Osim knjiga, knjižnice često nude i druge medije poput časopisa, filmova i digitalnih izvora te služe kao važno mjesto za obrazovanje, istraživanje i razonodu.

U školskom okruženju knjižnica ima posebnu ulogu jer omogućuje učenicima i nastavnicima pristup dodatnim izvorima informacija koji su ključni za učenje i razvoj. Upravo zbog važnosti knjižnice kao obrazovnog centra, ključno je osigurati da njezini resursi budu lako dostupni svim korisnicima.

Tradicionalni način upravljanja knjižnicom, koji se temelji na ručnom vođenju evidencija o posudbama i dostupnosti knjiga, postaje sve neefikasnije u suvremenom digitalnom dobu. Često dolazi do problema kao što su gubitak podataka, sporost u pronalasku informacija te nemogućnost pravovremenog informiranja korisnika o stanju knjiga.

Razvojem tehnologije i sve većom prisutnošću interneta, škole sve više prepoznaju potrebu za digitalizacijom svojih usluga, uključujući i rad knjižnica. Digitalni sustavi omogućuju jednostavnije pretraživanje, rezervaciju i upravljanje knjižničnom građom, čime se štedi vrijeme i povećava dostupnost informacija.

Motivacija za izradu ovog završnog rada bila je upravo želja za modernizacijom školskog sustava i olakšavanjem pristupa knjižničnim uslugama. Sustav za upravljanje školskom knjižnicom koji je izrađen u sklopu ovog rada omogućava korisnicima da putem interneta pretraže dostupne knjige, rezerviraju željene naslove i unaprijed se informiraju o stanju knjižnice, bez potrebe za osobnim dolaskom. Time se približava suvremenim potrebama korisnika i potiče njihovu veću interakciju s dostupnim resursima.

U tradicionalnim školskim knjižnicama posudba knjiga često se obavlja ručno, pri čemu knjižničar vodi evidenciju korisnika i knjiga pomoću bilježnica ili jednostavnih obrazaca. Ovakav način rada može uzrokovati razne probleme kao što su gubitak podataka, netočno vođenje evidencije, sporost u pronalasku informacija te nemogućnost brze provjere dostupnosti knjiga.

Jedan od značajnih problema je i ograničen pristup informacija o dostupnosti knjiga za korisnike koji nisu fizički prisutni u školi. U suvremenom društvu, koje sve više teži digitalizaciji usluga, korisnici očekuju da im informacije budu dostupne jednostavno i brzo, bez obzira na njihovu lokaciju.

Cilj ovog završnog rada je izraditi sustav koji omogućuje online pregled dostupnih knjiga u školskoj knjižnici i njihovu rezervaciju putem web stranice. Na taj način korisnik može unaprijed provjeriti koje su knjige dostupne te rezervirati željeni naslov. Nakon što izvrši rezervaciju, korisnik dolazi u školu kako bi osobno preuzeo knjigu i posudio je za čitanje.

Ovim sustavom knjižnica se približava modernim standardima, poboljšava se preglednost i dostupnost informacija te se korisnicima štedi vrijeme koje bi inače potrošili dolaskom samo radi provjere dostupnosti knjiga.

# Opis korištenih tehnologija

Za izradu sustava za upravljanje školskom knjižnicom korišten je skup tehnologija koje zajedno omogućuju izradu dinamične, funkcionalne i vizualno privlačne web aplikacije. Svaka tehnologija ima svoju posebnu ulogu u procesu izrade sustava i omogućava ostvarivanje specifičnih funkcionalnosti, od izgradnje korisničkog sučelja do upravljanja bazom podataka.

## HTML, CSS i JavaScript

**HTML** (*eng. HyperText Markup Language*) osnovni je građevni blok svake web stranice. Njegova glavna funkcija je strukturirati sadržaj stranice, definirajući elemente poput zaglavlja, odlomka, tablica, obrazaca i poveznica. Bez HTML-a web stranice ne bi mogle sadržavati organizirani tekst, slike, videozapise ili obrasce koji omogućuju interakciju s korisnikom. HTML omogućuje značenjsko označavanje sadržaja, čineći ga čitljivim i za korisnike i za tražilice.

**CSS** (*eng. Cascading Style Sheets*) koristi za oblikovanje HTML elementa. Dok HTML daje strukturu stranici, CSS definira njezin izgled – boje, fontove, margine, veličine elemenata, animacije i raspored. Na taj način CSS omogućava vizualnu prilagodbu stranice potrebama korisnika i modernim dizajnerskim trendovima. Pomoću CSS-a, web stranica postaje responzivna, što znači d se automatski prilagođava različitim veličinama zaslona kao što su računala, tableti i pametni telefoni.

**JavaScript** je programski jezik koji omogućava dodavanje dinamičkih elemenata web stranicama. U kontekstu ovog projekta JavaScript je korišten za poboljšanje korisničkog iskustva, kao što je automatsko ažuriranje rezultata pretrage bez potrebe za ponovnim učitavanjem cijele stranice (tzv. Dinamička interakcija). JavaScript omogućava i obradu korisničkih unosa, validaciju obrazaca i rad s elementima DOM strukture stranice.

**Primjena u završnom radu:**

U ovom projektu, **HTML** je korišten za izradu osnovne strukture web stranice – oblikovanje stranica za početni opis knjižnice i stranicu za pretraživanje knjiga.

**CSS** je korišten za stiliziranje stranice, definiranje izgleda alatne trake, kartica i rezultata pretrage.

**JavaScript** je upotrijebljen kako bi omogućio interaktivno ponašanje stranice, kao što je filtriranje rezultata pretrage i dinamički prikaz podataka bez potrebe za ponovnim učitavanjem stranice.

## PHP

PHP (*eng. Hypertext Preprocessor*) je skriptni jezik koji se izvršava na strani poslužitelja. Osnovna uloga PHP-a je stvaranje dinamičkog sadržaja, komunikacija s bazom podataka te obrada korisničkih zahtjeva. PHP je široko korišten zbog svoje jednostavnosti, brzine i velike kompatibilnosti s bazama podataka, osobito MySQL-om.

PHP kod se nalazi između korisnika i baze podataka. Kada korisnik unese podatke ili pošalje zahtjev (npr. pretragu), PHP ga obrađuje, komunicira s bazom podataka i vraća rezultate na pregledan način.

**Primjena u završnom radu:**

PHP je korišten za povezivanje web stranice s MySQL bazom podataka. Kada korisnik unese ime autora, naslov knjige ili bilo koji drugi kriterij pretrage, PHP obrađuje te podatke, šalje ih u bazu, dohvaća rezultate i prikazuje ih korisniku na stranici.

## XAMPP

XAMPP (*eng. Cross-Platform, Apache, MySQL, PHP, Perl*) je besplatan softverski paket koji omogućava lokalno postavljanje web poslužitelja. Ime XAMPP sastavljeno je od početnih slova glavnih komponenti: Apcache (poslužitelj), MySQL (baza podataka), PHP i Perl (programski jezici).

XAMPP simulira pravo serversko okruženje na lokalnom računalu, što programerima omogućuje da testiraju i razvijaju svoje aplikacije prije nego što ih učine dostupnima korisnicima na internetu.

**Primjena u završnom radu:**

XAMPP je korišten za lokalno pokretanje web stranice i baze podataka. Omogućio je rad za upravljanje školskom knjižnicom na mom računalu, gdje sam mogao testirati funkcionalnosti web stranice, provjeravati pretragu knjiga, rezervaciju i komunikaciju s bazom podataka prije nego što bi sustav eventualno bio premješten na pravi server.

## MySQL

MySQL je sustav za upravljanje relacijskom bazom podataka (*eng. RDBMS – Relational Database Menagment System*). Omogućuje pohranu velikih količina podataka u tablice koje su međusobno povezane i omogućuju brzo pretraživanje, ažuriranje i organizaciju informacija.

MySQL koristi standardni jezik SQL (eng. Structed Query Language)za komunikaciju s podatcima.

MySQL baze su popularne u svijetu zbog svoje brzine, stabilnosti i mogućnosti besplatnog korištenja u otvorenim projektima. Mnoge velike aplikacije i web stranice koriste MySQL upravo zbog njegove pouzdanosti.

**Primjena u završnom radu:**

MySQL je korišten za izradu baze podataka koji sadrže podatke o knjigama u knjižnici. Kreirana je tablica koja sadrži informacije poput imena i prezimena autora te naslova, mjesta izdavanja i godine izdavanja knjige.

Prilikom pretraživanja korisnički upit filtrira podatke iz ove tablice i prikazuje korisniku samo tražene knjige koje su dostupne.

# Arhitektura baze podataka

Baza podataka predstavlja organizirani skup podataka koji omogućuje učinkovito upravljanje i pristup informacijama. U kontekstu ovog završnog rada, baza podataka koristi se za pohranu podataka o knjigama koje se nalaze u školskoj knjižnici, uključujući informacije o autorima, naslovima, godinama izdanja, dostupnosti i slično.

Arhitektura baze podataka može se promatrati kroz 3 osnovne razine: logičku, fizičku i konceptualnu. Ove razine omogućuju odvajanje pogleda korisnika od tehničke implementacije baze podataka.

### Logička razina

Logička razina prikazuje način na koji su podatci organizirani i međusobno povezani u samoj bazi. Na ovoj razini definiraju se tablice, njihova polja (atributi), tipovi podataka, primarni i strani ključevi te odnosi među tablicama.

U ovom projektu logička razina obuhvaća tablicu knjiga\_inventara\_csv, koja sadrži sljedeće atribute:

* Broj
* Datum
* Prezime i ime autora
* Naslov\_mjesto\_godina
* Svezak\_God\_Br
* Dobavljač
* Vrijednost (euro)
* Uvez
* Signatura
* Napomena

Svaki atribut u ovoj tablici predstavlja jednu jedinstvenu knjigu te omogućuje filtriranje i pretragu prema tim kriterijima.

### Fizikalna razina

Fizikalna razina odnosi se na način koji su podatci fizički pohranjeni na disku. U slučaju ove baze, podatci se spremaju u MySQL sustavu koji brine o strukturiranom zapisu podataka na lokalnom poslužitelju (putem XAMPP-a).

Na ovoj razini brigu vodi **DBMS** (Sustav za upravljanje bazom podataka), koji upravlja:

* Načinom pohrane podataka
* Korištenjem memorije
* Indeksima za bržu pretragu
* Pristupom podatcima i njegovom zaštitom

Fizikalni prikaz korisnik ne vidi direktno, ali on osigurava pouzdanu i sigurnu pohranu podataka.

# Modeli baze podataka

Model baze podataka predstavlja konceptualni okvir prema kojem se strukturiraju podatci u bazi. Odabir modela ima veliki utjecaj na funkcionalnost, brzinu i jednostavnost korištenja baze.

U nastavku ću objasniti neke najpoznatije modele baza podataka:

#### Hijerarhijski model

U hijerarhijskom modelu podatci su organizirani poput stabla. Svaki roditeljski zapis može imati više djece, ali svako dijete ima samo jednog roditelja. Ovaj model danas se rijetko koristi jer je teško prilagodljiv kompleksnijim odnosima među podatcima.

Primjer hijerarhijskog modela je struktura mapa i datoteka na računalu.

#### Mrežni model

Mrežni model je sličan hijerarhijskom, ali omogućuje da jedan zapis ima više nadređenih i podređeni zapisa. Time je omogućeno složenije povezivanje podataka. Koristio se u ranijim fazama razvoja baze podataka, ali danas je gotovo u potpunosti zamijenjen relacijskim modelom.

#### Plošni model

Plošni (ili jednostavni) model predstavlja podatke u obliku jedne tablice. Najčešće se koristi kod manjih baza i za demonstraciju osnovnih principa.

Primjer je Excel tablica u kojoj se svi podatci nalaze u jednom radnom listu.

U ovom projektu korištena je baza slična plošnom modelu jer se svi podatci nalaze unutar jedne tablice koja se pretražuje i filtrira.

#### Objektni model

Objektni model se temelji na principima objektno-orijentiranog programiranja. Podaci se pohranjuju kao objekti koji sadrže kompleksne ili multimedijske podatke, poput slika, zvučnih zapisa, videozapisa, dokumenata te podataka koji zahtijevaju naprednu obradu i pohranu.

#### Relacijski model

Relacijski model je danas najrašireniji i najčešće korišteni model baze podataka. Temelji se na organizaciji podataka u tablice (relacije) koje su međusobno povezane.

Svaka tablica ima redove (zapise) i stupce (atribute).Ključne značajke relacijskog modela uključuju:

* Jednostavne primarne ključeve
* Odnose među tablicama poput stranih ključeva
* Izbjegavanje ponavljanja podataka (redundancije)
* Mogućnost jednostavne nadogradnje baze

U ovom projektu korišten je relacijski model jer omogućuje pouzdanu, fleksibilnu i učinkovitu organizaciju podataka o knjigama. Kroz SQL naredbe omogućena je jednostavna manipulacija podatcima: unos, izmjena i brisanje.