**AKADEMIJA TEHNIČKO – UMETNIČKIH STRUKOVNIH STUDIJA-BEOGRAD**

**ODSEK VISOKA ŠKOLA ELEKTROTEHNIKE I RAČUNARSVA**

**Milanović Danilo**

**Implementacija infromacionog sistema na primeru muzeja**

**- završni rad –**



Beograd, 2023

**REZIME:**

Ovo istraživanje se fokusira na projektovanje i implementaciju informacionog sistema za muzej s ciljem unapređenja muzejskih usluga i efikasnog upravljanja muzejskim resursima. Metodologija obuhvata analizu korisničkih potreba, odabir tehnoloških rešenja i razradu sistema. Rad se oslanja na uverenje da informacioni sistemi mogu značajno promeniti način na koji muzeji pružaju usluge i upravljaju svojim aktivnostima.

Kao rezultat istraživanja, sistem je dizajniran i uspešno implementiran, omogućavajući korisnicima bogatije i interaktivnije muzejsko iskustvo, kao i olakšavanje administrativnih zadataka muzejskog tima.

**Ključne reči:** informacioni sistem, muzej, korisnici, administracija, implementacija.

**ABSTRACT:**

This research focuses on the design and implementation of an information system for a museum aimed at enhancing museum services and efficient management of museum resources. The methodology involves analyzing user needs, selecting technological solutions, and system development. The work is based on the belief that information systems can significantly transform the way museums provide services and manage their activities.

As a result of the research, the system has been designed and successfully implemented, enabling users to have a richer and more interactive museum experience while streamlining administrative tasks for the museum team.

**Keywords:** information system, museum, users, administration, implementation.

|  |
| --- |
| Kadnidat: **Milanović Danilo** |
| 1. Broj indeksa: **is-21/20** |
| 1. Studiski program: **Informacioni sistemi** |
| 1. Tema: **implementacija infromacionog sistema na primeru muzeja** |
| Osnovni zadaci: |
| **1. Analiza poslovnih procesa** |
| **2. Projektovanje baze podataka**  **3. Praktična realizacija informaciong sistema**  Mentor:  Beograd, septembar 2023.godine.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Mr. Miloš Pejanović |
|  |
|  |
|  |

Sadržaj

[1. Uvod 5](#_Toc146225065)

[2. Tehnologije 6](#_Toc146225066)

[2.1. XAMPP i PHP 8.2 6](#_Toc146225067)

[2.2. JavaScript, Ajax i jQuery 6](#_Toc146225068)

[2.3. HTML i CSS 7](#_Toc146225069)

[2.4. SQL 7](#_Toc146225070)

[2.5. Bootstrap 7](#_Toc146225071)

[2.6. Power Designer 7](#_Toc146225072)

[3. Modelovanje 8](#_Toc146225073)

[3.1 Use case dijagram 9](#_Toc146225074)

[3.2 Dijagram sekvenci 12](#_Toc146225075)

[3.2.1 Dijagram sekvenci korisnika 12](#_Toc146225076)

[3.2.2 Dijagram sekvenci administratora 14](#_Toc146225077)

[3.3 Dijagram aktivnosti 16](#_Toc146225078)

[4. Baza podatka 18](#_Toc146225079)

[4.1 Konceptualni model baze podataka 18](#_Toc146225080)

[4.2 Fizički model baze podataka 20](#_Toc146225081)

[5. Implementacija 22](#_Toc146225082)

[6. Testiranje aplikacije 31](#_Toc146225083)

[6.1 Testiranje korisničkih zahteva 31](#_Toc146225084)

[6.2 Testiranje administrativnih zahteva 34](#_Toc146225085)

[7. Zaključak 37](#_Toc146225086)

[8. Indeks pojmova 38](#_Toc146225087)

[9. Literatura 39](#_Toc146225088)

[10. IZJAVA O AKADEMSKOJ ČESTITOSTI 40](#_Toc146225089)

# 1. Uvod

Kroz ovaj uvod, predstavićemo ključne aspekte u vezi sa temom rada, istraživanjem koje je sprovedeno, teorijskim osnovama, rešenjima i analizama obrađenim u radu, kao i strukturom samog rada.

U okviru ovog rada, pažljivo smo istražili proces projektovanja i implementacije informacionog sistema za muzej. Fokus je stavljen na analizu korisničkih potreba i odabir odgovarajućih tehnoloških rešenja.

Analiziraćemo praktične aspekte sistema kako za korisnike, tako i za administraciju muzeja, uz istraživanje izazova i implementaciju rešenja. Pored toga, obratićemo pažnju na merenje efikasnosti informacionog sistema i njegov uticaj na muzejske operacije.

Struktura rada će biti organizovana u sekcije koje detaljno opisuju metodologiju projektovanja, teorijske osnove, analize i rezultate istraživanja, pružajući celovit pregled procesa implementacije informacionog sistema u muzejskom okruženju. U ovom radu ćemo obraditi:

U prvom delu je uvod. Nakon njega sledi drugi deo, u kome ćemo se baviti tehnologijama koje su korišćene u izradi aplikacije. Ovde će biti razmatrane različite tehnologije, uključujući XAMPP i PHP 8.2, JavaScript, Ajax i jQuery, HTML i CSS, SQL (Structured Query Language) i Bootstrap.

Modelovanje kao treće poglavlje koje ćemo obraditi fokusira se na proces modeliranja informacionog sistema. Ovde ćemo razmotriti različite modele, uključujući use case dijagram, dijagram sekvenci korisnika i administratora, kao i dijagram aktivnosti.

U četvrtom poglavlju, razmatraćemo sve aspekte baze podataka. Ovde će biti prikazani konceptualni model baze podataka, kao i fizički model baze podataka.

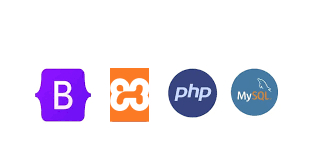
Implementacija kao peto poglavlje, posvećeno je detaljima implementacije aplikacije. Čitaoci će saznati kako su tehnologije i modeli primenjeni u stvarnom okruženju.

Testiranje predstavlja šesto poglavlje, i bavi se procesom testiranja aplikacije. Prikazuje kako su korisnički i administrativni zahtevi proveravani i testirani kako bi se osigurala funkcionalnost i ispravnost sistema.

Zaključak: Konačno, sedmo poglavlje donosi zaključke rada, ističući glavne tačke, izazove i doprinose implementiranog sistema.

# 2. Tehnologije

Razvoj informacionog sistema za muzej zahtevao je pažljivo odabiranje alata i tehnologija kako bi se postigla funkcionalnost i estetska privlačnost sistema. PHP je korišćen kao osnovni programski jezik za izradu serverske logike. Pomoću PHP-a omogućena je dinamička generacija sadržaja, interakcija sa bazom podataka i obrada korisničkih zahteva.

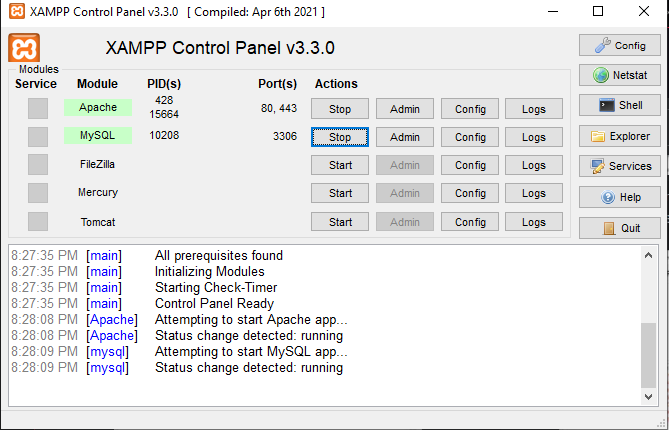


*Slika 2.1 – Alati korišćeni u izradi završnog rada.*

## 2.1. XAMPP i PHP 8.2

XAMPP Control Panel (Slika 2.1.1) je korisni alat za upravljanje svim ključnim komponentama .XAMPP razvojnog okruženja, uključujući Apache web server, MySQL bazu podataka i PHPXAMPP je posebno koristan alat za razvoj web aplikacija. On integriše Apache web server, MySQL bazu podataka, PHP programski jezik i druge komponente u jednostavan paket. Korišćenjem XAMPP-a, programeri mogu brzo postaviti lokalno razvojno okruženje na svom računaru. Najnovija verzija XAMPP-a podržava PHP verziju 8.2, što omogućava programerima da iskoriste najnovije funkcionalnosti i poboljšanja u PHP jeziku.

U kombinaciji, ove tehnologije čine moćan alatni set za izradu, testiranje i optimizaciju web aplikacija i baza podataka. Omogućavaju programerima da efikasno razvijaju i upravljaju svojim projektima bez potrebe za pristupom javnim web serverima tokom razvojnog procesa.



*Slika 2.1.1 – Slika XAMPP kontrolnog panela.*

## 2.2. JavaScript, Ajax i jQuery

JavaScript je esencijalni jezik za web razvoj, omogućavajući interaktivne web stranice i čini vitalni deo web aplikacija. Ajax omogućava brze promene na stranici bez potrebe za ponovnim učitavanjem, dok jQuery i drugi framework-ovi pojednostavljuju razvoj.

## 2.3. HTML i CSS

HTML je osnovni jezik za strukturu web stranica, definisanje elemenata kao što su naslovi i paragrafi. CSS se koristi za stilizaciju elemenata. Ovi jezici zajedno formiraju osnovu za dizajniranje web stranica. Dodatno, alati poput SCSS (SASS) olakšavaju stilizaciju i održavanje koda za privlačne i odzivne web stranice.

## 2.4. SQL

SQL (Structured Query Language) je standardni jezik za upravljanje i manipulaciju relacionalnim bazama podataka kao što je MySQL. SQL omogućava izradu upita za čitanje, pisanje, ažuriranje i brisanje podataka u bazi. Ovaj jezik je ključan za efikasno upravljanje i ekstrakciju podataka iz baze. Programeri i administratori baza podataka koriste SQL za izradu kompleksnih upita i procedura koje omogućavaju obradu podataka.

## 2.5. Bootstrap

Bootstrap je popularni open-source framework za razvoj responzivnih i vizuelno privlačnih web stranica. Ovaj framework pruža kolekciju CSS stilova i JavaScript komponenata koje olakšavaju dizajniranje modernih web interfejsa. Bootstrap omogućava brzu i doslednu izradu web stranica i aplikacija, čime se štedi vreme i resursi programera.

## 2.6. Power Designer

PowerDesigner je visoko efikasan alat za modeliranje baza podataka i poslovnih procesa. Omogućava kreiranje vizualnih dijagrama kao što su UML dijagrami, dijagrami sekvenci, i dijagrami aktivnosti, čime olakšava projektovanje sistema. Osim toga, automatski generiše SQL kod za kreiranje baza podataka, što ubrzava proces implementacije. Alat podržava timski rad i deljenje projekata u različitim formatima. Sa svojim intuitivnim interfejsom i bogatim setom funkcionalnosti, PowerDesigner je nezamenjiv alat za inženjere softvera i baza podataka koji žele precizno definisati i dokumentovati svoje projekte.

# 3. Modelovanje

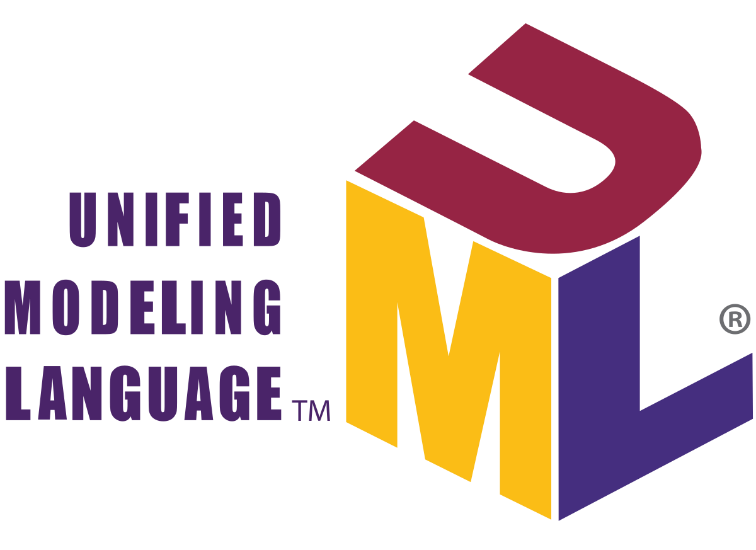
Tokom studiranja smo se bavili UML (Unified Modeling Language) modelovanjem, što predstavlja standardni jezik za vizualizaciju, projektovanje i dokumentovanje softverskih sistema. Ova veština modelovanja postala je esencijalna komponenta našeg obrazovanja jer pruža uvid u kompleksne aspekte softverskog inženjeringa.

Učenje i praktikovanje UML modelovanja omogućili su nam razumevanje struktura i ponašanja softverskih sistema na dubljem nivou. UML je prepoznatljiv po bogatom setu notacija i dijagrama koji služe kao jezik komunikacije između timova razvoja i ostalih relevantnih interesnih grupa.

Jedna od ključnih prednosti UML-a jeste sposobnost da se apstraktne ideje prevedu u konkretnu vizualnu formu. Kroz korišćenje različitih UML dijagrama, kao što su dijagrami klasa, dijagrami stanja, dijagrami sekvence i mnogi drugi, mogli smo precizno predstaviti strukturu, funkcionalnosti i interakcije unutar softverskih sistema.

Važno je napomenuti da je UML modelovanje igralo ključnu ulogu u olakšavanju komunikacije unutar timova za razvoj softvera. Vizualni prikazi i dijagrami omogućavaju članovima tima da brže i jasnije razumeju složene koncepte i ideje, što ubrzava procese razvoja i donošenja odluka.

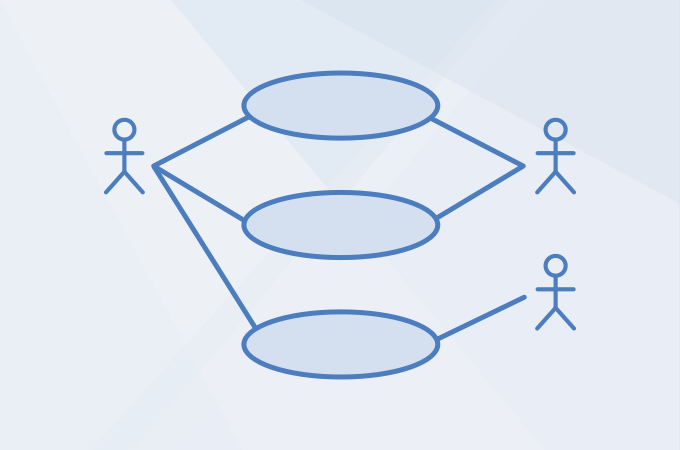
Kroz sve ovo, UML modelovanje nije samo postalo deo našeg obrazovanja, već i snažan alat koji će nam koristiti u budućoj karijeri kao softverskim inženjerima. Omogućava nam precizno dokumentovanje dizajna softverskih sistema, što je ključno za razvoj i održavanje visokokvalitetnog softvera u profesionalnom okruženju.

**

*Slika 3.1 – Logotip UML-a.*

## 3.1 Use case dijagram

Use Case dijagrami su ključni element UML modelovanja i igraju značajnu ulogu u identifikaciji i dokumentovanju funkcionalnih zahteva softverskog sistema. Ovaj tip dijagrama pomaže timu za razvoj softvera da bolje razume kako će korisnici interagovati sa sistemom i koji su to ključni scenariji ili funkcionalnosti koje sistem treba da podrži.



*Slika3.1.1 – Use case dijagram.*

**Funkcionalnosti korinika na sistemu:**

**Novi korisnik:**

*Registracija:* Neregistrovani korisnik može se registrovati na platformi popunjavanjem registracionog obrasca i stvaranjem korisničkog naloga.

*Prijava:* Nakon registracije, korisnik može da se prijavi na svoj nalog kako bi pristupio dodatnim funkcijama rezervacije.

*Pregled izložbi:* Novi korisnik takođe može pregledavati dostupne izložbe i informacije o njima, baš kao i gost.

**Registrovani korisnik:**

*Rezervacija izložbe:* Registrovani korisnik može da rezerviše karte za određenu izložbu. Bira datum i vreme posete i potvrđuje rezervaciju.

*Upravljanje rezervacijama:* Korisnik može pregledati svoje postojeće rezervacije. Takođe može otkazati rezervaciju ako je potrebno.

*Pregled izložbi:* Registrovani korisnik i dalje ima pristup pregledu izložbi i informacijama o njima.

**Administrator sistema**: ima širok spektar funkcionalnosti koje mu omogućavaju efikasno upravljanje informacionim sistemom muzeja.

*Dodavanje Nove Izložbe:* Administrator može dodavati nove izložbe u sistem. Prilikom dodavanja izložbe, administrator unosi informacije kao što su naziv izložbe, lokacija, datum i vreme izložbe, opis i slično.

*Uređivanje informacija o izložbama:* Postojeće informacije o izložbama mogu se uređivati. To uključuje izmenu datuma, vremena, opisa ili drugih relevantnih detalja vezanih za izložbu.

*Brisanje**izložbe*: Administrator može obrisati izložbu iz sistema ako više nije aktuelna ili je potrebno ukloniti je iz ponude.

*Dodavanje novog eksponata:* Eksponati koji će biti izloženi na određenoj izložbi mogu se dodati u sistem. Administrator unosi informacije o svakom eksponatu, uključujući naziv, opis, sliku i slično.

*Uređivanje informacija o eksponatima:* Postojeći eksponati mogu se uređivati. To omogućava administratoru da ažurira opis, sliku ili druge podatke o eksponatu.

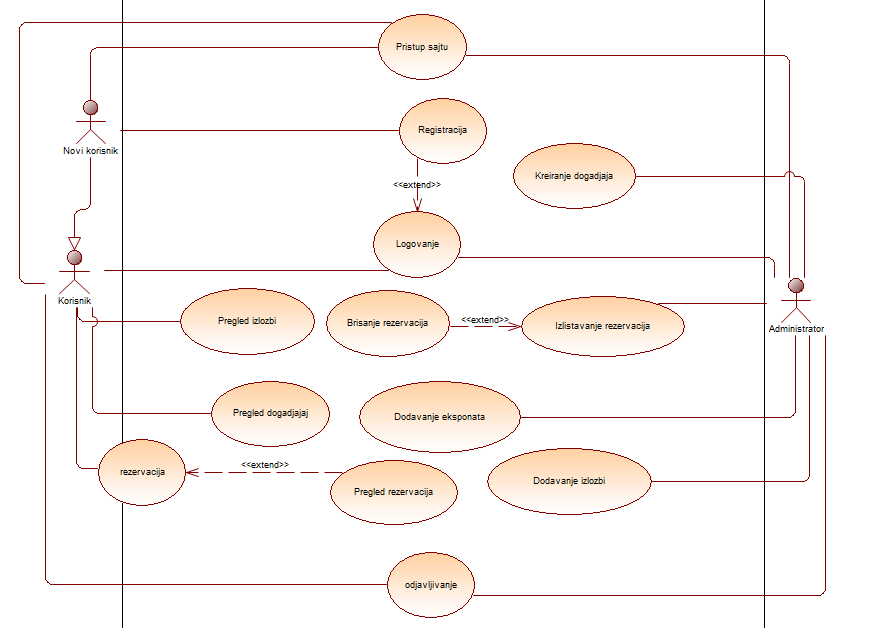
*Brisanje eksponata:* Administrator može ukloniti eksponat iz sistema ako više nije deo izložbe ili ako je potrebno ukloniti ga iz baze podataka.

*Upravljanje korisnicima:* Administrator može pregledati korisničke profile, brisati neaktivne ili nepoželjne korisnike i vršiti administraciju korisničkim nalozima.

*Upravljanje rezervacijama:*Administrator ima pristup informacijama o svim rezervacijama. Može pregledati, potvrditi, odbiti ili otkazati rezervacije na zahtev korisnika.

*Dodavanje događaja:* Administrator može da kreira nove događaje, pridruži im odgovarajuće eksponate i unese sve relevantne informacije. Ova funkcionalnost omogućava bolje planiranje i upravljanje događajima.

Ove funkcije omogućavaju administratoru da efikasno upravlja svim aspektima informacionog sistema za rezervaciju izložbi i obezbedi kvalitetno iskustvo korisnicima.



*Slika3.1.2 – Use case dijagram.*

## 3.2 Dijagram sekvenci

Dijagram aktivnosti je vizualni alat koji se koristi za modeliranje tokova aktivnosti unutar sistema ili procesa.

### 3.2.1 Dijagram sekvenci korisnika

Kao što je prikazano na slici (Slika 3.2.1.1) dijagram sekvenci koji ilustruje složenu interakciju između korisnika i sistema za rezervaciju izložbi. Ovaj dijagram prikazuje kako korisnik komunicira sa sistemom, koristeći različite funkcionalnosti sistema kako bi ostvario svoje potrebe i želje vezane za posete muzeju. Sistem za rezervaciju izložbi pruža korisnicima bogato i intuitivno iskustvo u planiranju poseta muzeju. Ovaj sistem omogućava korisnicima da istraže dostupne izložbe, eksponate i događaje, da pronađu informacije koje ih interesuju i da rezervišu karte za posetu.

Korisnik započinje svoju interakciju s sistemom pregledom dostupnih izložbi, eksponata i događaja. Ova početna aktivnost omogućava korisniku da se upozna sa ponudom i izabere šta želi da poseti. Sistem nudi raznovrsne opcije i sadržaje kako bi zadovoljio različite interesovanja posetilaca.

Registracija i prijava su ključni koraci za pristup sistemu. Ukoliko korisnik nema već kreiran korisnički nalog, može se registrovati. Tokom registracije, korisnik unosi svoje lične informacije i kreira nalog. Nakon kreiranja naloga, korisnik može da se prijavi koristeći svoje korisničko ime i lozinku. Ova funkcionalnost osigurava siguran pristup sistemu i personalizovano iskustvo.

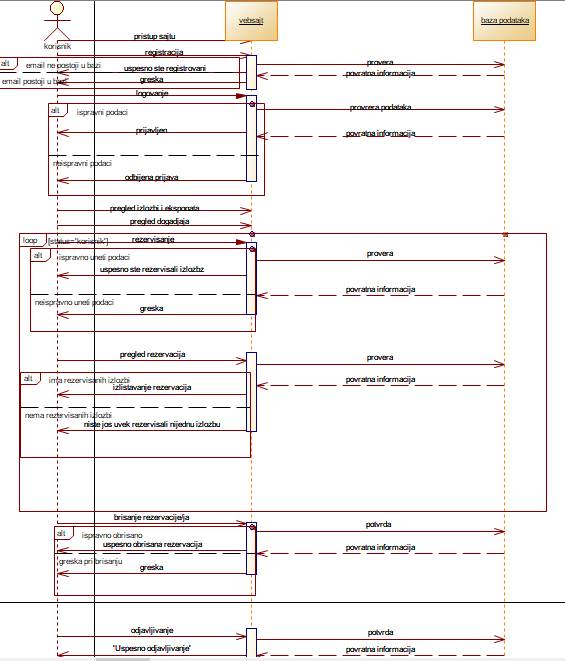
Kada korisnik pronađe izložbu ili eksponat koji ga interesuje, može da pregleda detaljnije informacije. To uključuje čitanje opisa, pregled slika i proveru dostupnih datuma posete. Ova funkcionalnost omogućava korisnicima da se dublje upuste u sadržaj izložbi i donesu informisane odluke o svojim posetama.

Rezervacija je ključna aktivnost za korisnike koji žele da posete određene izložbe. Korisnik ima opciju da rezerviše kartu za posetu. Ova aktivnost uključuje odabir datuma i vremena posete, broja karata. Sistem omogućava korisnicima da jednostavno rezervišu karte i obezbedi svoje mesto na željenim izložbi.

Upravljanje Rezervacijama je još jedna važna funkcionalnost sistema. Korisnik može da pregleda ili eventualno otkazuje svoje postojeće rezervacije. Ova fleksibilnost omogućava korisnicima da bolje planiraju svoje posete muzeju i da se prilagode promenljivim okolnostima.

Na kraju svih aktivnosti, korisnik se može odjaviti sa svog korisničkog naloga kako bi zaštitio svoje podatke i završio svoju sesiju na sistemu za rezervaciju izložbi. Ovo je važna funkcionalnost za sigurnost korisnika i očuvanje njihovih privatnih informacija.

Ove aktivnosti čine korisničko iskustvo na sistemu za rezervaciju izložbi bogatim i korisnim. Sistem omogućava korisnicima da pronađu, rezervišu i uživaju u posetama muzeju na jednostavan i efikasan način.



*Slika 3.2.1 – Dijagram sekvenci korisnika.*

### 3.2.2 Dijagram sekvenci administratora

Kao što možemo primetiti na slici (Slika 3.2.2.1) koja pruža uvid u interakciju između administratora i sistema muzeja. Ovaj dijagram prikazuje kako administrator koristi različite funkcionalnosti sistema kako bi upravljao sadržajem i pružio podršku korisnicima.

Administrator započinje svoju interakciju pregledom celokupnog sadržaja sistema, uključujući izložbe, eksponate, lokacije i događaje. Ovo mu omogućava da ima kompletan pregled nad svim elementima u sistemu.

Ima privilegije za upravljanje izložbama i eksponatima, što uključuje dodavanje novih, uređivanje postojećih informacija i brisanje neželjenih elemenata. Takođe može da upravlja lokacijama, dodavati nove ili ažurirati informacije o postojećim.

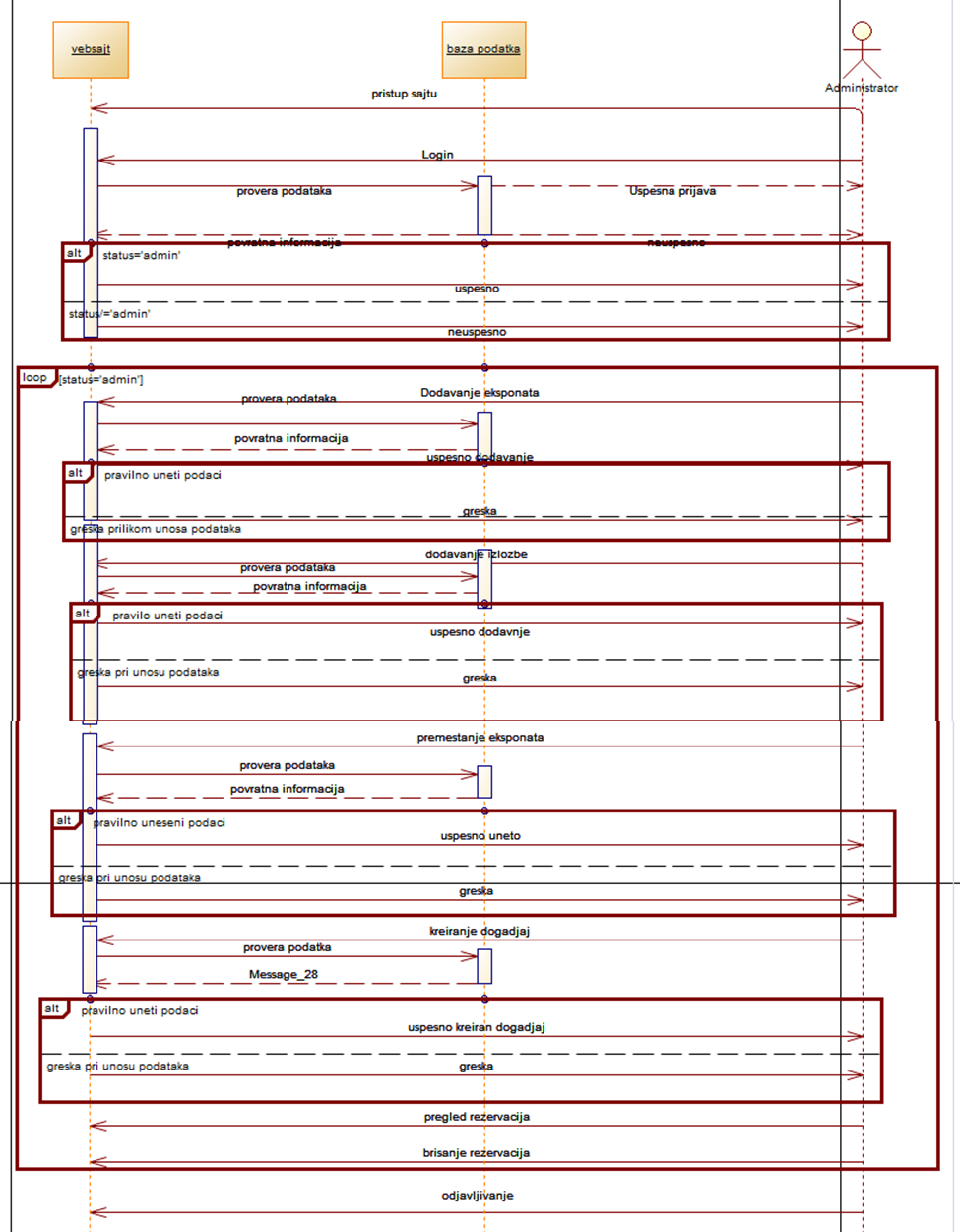
Administrator ima ovlašćenja za upravljanje događajima, uključujući dodavanje, uređivanje događaja. Može precizirati datume, vremena i ostale detalje vezane za održavanje događaja.

Pored toga, administrator može premestiti eksponate sa jedne lokacije na drugu. Ovo omogućava fleksibilno upravljanje postavkama muzeja i izložbi.

Upravljanje rezervacijama je takođe deo administratorove uloge. Može pregledavati sve rezervacije, vršiti njihovo uređivanje i otkazivanje. Ova funkcionalnost mu omogućava da pruži podršku korisnicima u vezi sa njihovim rezervacijama.

Na kraju interakcije, administrator se može bezbedno odjaviti sa svog naloga kako bi osigurao sigurnost i zaštitu svojih podataka.

Administrator ima ključnu ulogu u efikasnom upravljanju sistemom za rezervaciju izložbi i obezbeđivanju kvalitetnog korisničkog iskustva. Ove funkcionalnosti omogućavaju mu da održava sadržaj sistema, upravlja događajima i rezervacijama, pruža podršku korisnicima i donosi informisane odluke.



*Slika 3.2.2.1 – Dijagram sekvenci administratora.*

## 3.3 Dijagram aktivnosti

Sistem za rezervaciju izložbi predstavlja informacioni sistem koji omogućava korisnicima da istraže i isplaniraju svoje posete muzejima, izložbama i događajima na jednostavan i efikasan način. Ovaj sistem pruža niz funkcionalnosti kako bi korisnicima omogućio da pronađu, rezervišu i upravljaju svojim posetama muzeju ili izložbi.

Ključna aktivnost koja označava početak korisničkog iskustva je pregled dostupnih izložbi i eksponata. Korisnik započinje svoju interakciju sa sistemom istražujući bogatu ponudu izložbi, prikupljajući informacije o dostupnim događajima i datumima. Ova aktivnost omogućava korisnicima da se upoznaju sa ponudom i izaberu šta ih najviše interesuje.

Da bi u potpunosti iskoristili funkcionalnosti sistema, korisnici imaju opciju registracije ili prijave. Novi korisnici kreiraju svoje korisničke naloge unoseći svoje lične informacije i postavljajući svoje korisničko ime, lozinku, email-adresu, broj telefona. Registrovani korisnici jednostavno ulaze u sistem koristeći svoje već postojeće korisničke podatke. Ova aktivnost olakšava pristup korisničkom iskustvu i omogućava personalizaciju..

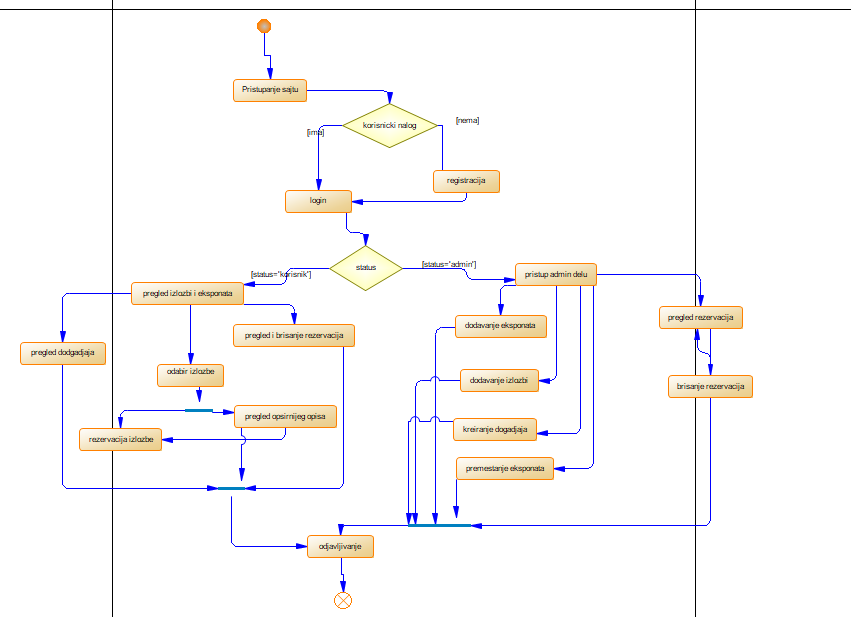
Nakon što korisnik pronađe izložbu ili eksponat koji ih interesuje, mogu da pristupe detaljnim informacijama o samoj izložbi. Ova aktivnost uključuje pregled opisa, slika, datuma i vremena posete, kao i svih relevantnih informacija koje bi mogle da pomognu korisniku pri odlučivanju i odabiru.

Rezervacija karata predstavlja ključnu aktivnost sistema. Korisnici imaju mogućnost da rezervišu karte za posetu izložbi. Ova aktivnost podrazumeva izbor datuma i vremena posete, kao i broja karata koji su im potrebni.

Upravljanje rezervacijama je aktivnost koja pruža korisnicima mogućnost pregleda, uređivanja ili otkazivanja svojih postojećih rezervacija. Ova funkcionalnost omogućava korisnicima fleksibilnost u planiranju svojih poseta muzeju i priliku da prilagode svoje planove u skladu sa promenama u rasporedu.

Kao finalna aktivnost u korisničkom iskustvu, odjava sa korisničkog naloga je važan korak koji pomaže u zaštiti ličnih podataka korisnika i pruža siguran završetak sesije.

Administrator sistema ima dodatne funkcionalnosti, uključujući dodavanje novih izložbi, eksponata i događaja u sistem, kao i upravljanje postojećim sadržajem. Administrator takođe može premestiti eksponate sa jedne lokacije na drugu. Takođe administrator može da vrši pregled rezervacija za određene izložbe, kao i da ih uređuje. Ove funkcionalnosti omogućavaju administratoru da održava i ažurira sistem kako bi pružio najnovije informacije korisnicima.



*Slika 3.3.1 – Dijagram aktivnosti.*

# 4. Baza podatka

Informacioni sistem će se sastojati od niza tabela koje će služiti za skladištenje podataka i olakšati efikasno upravljanje informacijama. Ove tabele će omogućiti praćenje rezervacija izložbi, upravljanje tokom događaja i druge relevantne funkcionalnosti.

U narednom delu, predstavićemo osnovne tabele koje će biti ključni elementi informacionog sistema. Svaka od ovih tabela će sadržavati specifična polja koja će nam omogućiti organizaciju i manipulaciju podacima na odgovarajući način.

## 4.1 Konceptualni model baze podataka

Baza podataka sistema za rezervaciju izložbi u muzeju sastoji se od nekoliko ključnih tabela koje čuvaju važne informacije o eksponatima, izložbama, korisnicima i rezervacijama (Slika 4.1.1). Ove tabele igraju ključnu ulogu u funkcionalnosti sistema i omogućavaju korisnicima da pretražuju izložbe, rezervišu karte i upravljaju svojim rezervacijama.

Tabela “EKSPONATI“ čuva informacije o eksponatima, uključujući njihov jedinstveni identifikator, lokaciju gde se nalaze, opis eksponata, povezanost sa događajima i izložbama, naziv eksponata i putanju do slike eksponata.

Tabela “IZLOZBA“ sadrži podatke o izložbama, uključujući identifikator izložbe, povezanost sa određenom lokacijom, naziv izložbe, datum održavanja izložbe, opis izložbe i putanju do slike koja je predstavlja.

Korisnici sistema registrovani su u tabeli “KORISNIK”, gde svaki korisnik ima jedinstveni identifikator. Ova tabela čuva lične podatke korisnika, uključujući ime, prezime, broj telefona, e-mail adresu i šifru.

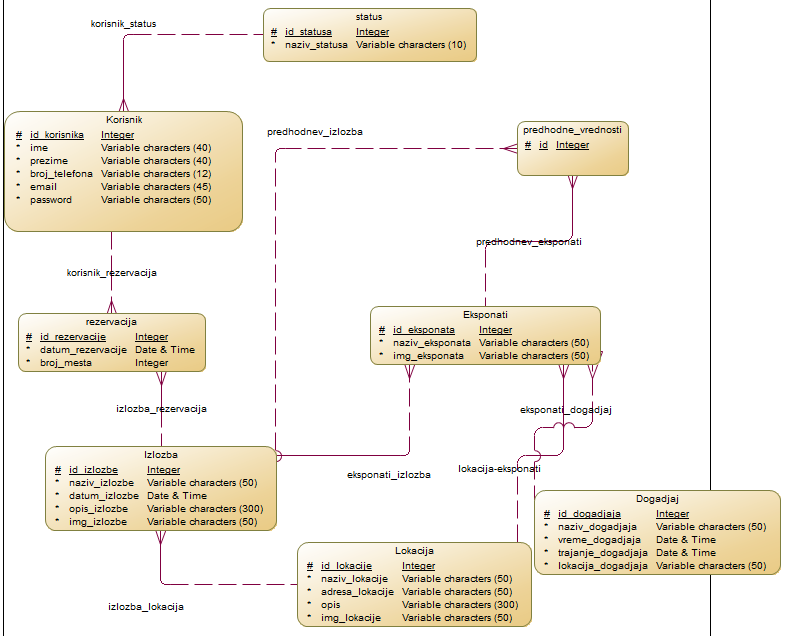
Tabela “LOKACIJA“ čuva informacije o lokacijama muzeja, uključujući njihov identifikator, naziv, adresu, opis i putanju do slike lokacije.

Tabela “PREDHODNE\_VREDNOSTI“ se koristi za praćenje promena u atributima eksponata i povezana je sa eksponatima i izložbama, omogućavajući praćenje istorije promena prilikom dodavanja dogadjaja. Kada dogadjaj istekne eksponati se vraćaju u svoje izlozbe u kojima su bile pre kreiranja dogadjaja.

Tabela “STATUS“ predstavlja bitan deo strukture baze podataka koja se koristi kako bi se zadovoljila normativna forma, posebno treća normativna forma (3NF). Ova tabela ima ključnu ulogu u organizaciji i skladištenju podataka na način koji obezbeđuje minimalnu redundanciju i očuvanje referencijalne integritete podataka.

I na kraju tabela “REZERVACIJA“ čuva informacije o rezervacijama karata za izložbe, uključujući identifikator rezervacije, povezanost sa izložbama i korisnicima, datum i vreme rezervacije.

Ove tabele čine srce baze podataka sistema za rezervaciju izložbi i omogućavaju efikasno upravljanje informacijama i interakciju korisnika sa sistemom. Sistem omogućava korisnicima da pregledaju dostupne izložbe, eksponate, vrše rezervacije i upravljaju svojim rezervacijama.



*Slika 4.1.1 – Konceptualni model baze podatka.*

## 4.2 Fizički model baze podataka

Nakon prelaska iz konceptualnog modela (Slika 4.1.1) u fizički model baze podataka, primetili smo da nekoliko ključnih tabela sada uključuje strane ključeve (Foreign Keys). Ovi strani ključevi su od suštinskog značaja jer omogućavaju da se veze između tabela jasno i precizno definišu kako bi se postigla tačnost, doslednost i integritet podataka u celom sistemu.

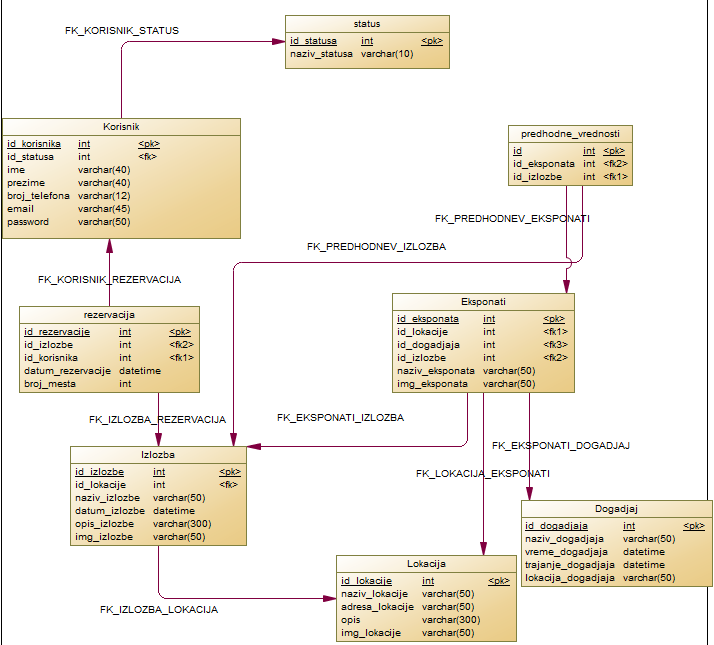
Konceptualni model baze podataka pruža apstraktan pregled podataka i njihovih veza, dok fizički model uvodi detaljne tehničke specifikacije koje omogućavaju konkretno skladištenje i manipulaciju podacima. Prilikom prelaska na fizički model, ključni aspekti koji se uzimaju u obzir uključuju strukturu tabela, tipove podataka i način na koji se uspostavljaju veze između tih tabela.

U našem slučaju, implementacija stranih ključeva postala je neophodna kako bismo adekvatno opisali i upravljali vezama između različitih entiteta u našem sistemu. Ovi strani ključevi funkcionišu kao mostovi koji povezuju entitete i omogućavaju nam da precizno pratimo kako se podaci razmenjuju između tabela.

Kroz proces dodavanja stranih ključeva u odgovarajuće tabele, obezbeđujemo da svaki redak podataka bude tačno povezan sa odgovarajućim redom u drugoj tabeli. Kao što je prikazano na slici (Slika 4.2.1), kroz strani ključ “FK\_EKSPONATI\_DOGADJAJ“ u tabeli “EKSPONATI“, mi definišemo vezu između eksponata i događaja, koristeći “ID\_DOGADJAJA“ kao referencu na primarni ključ u tabeli “DOGADJAJ“. Ovo osigurava da svaki eksponat bude povezan sa tačno određenim događajem, pružajući dragocenu informaciju.

Pored toga, koristeći klauzule “on delete restrict” i “on update restrict” u definicijama stranih ključeva, mi postavljamo pravila za očuvanje integriteta podataka. Na primer, restrikcija na brisanje ili ažuriranje redova u povezanim tabelama osigurava da se podaci ne naruše nehotičnim ili neovlašćenim akcijama. Ovo je posebno važno za očuvanje doslednosti i tačnosti podataka u bazi.

Primena stranih ključeva u fizičkom modelu baze podataka igra ključnu ulogu u postizanju preciznosti i integriteta podataka. Oni omogućavaju da se veze između tabela jasno definišu i da se obezbedi doslednost podataka u celom sistemu. Kroz pažljivu implementaciju stranih ključeva, stvaramo čvrstu osnovu za efikasno upravljanje podacima i olakšavamo rad sa složenim bazama podataka u širokom spektru informacionih sistema i poslovne primene.



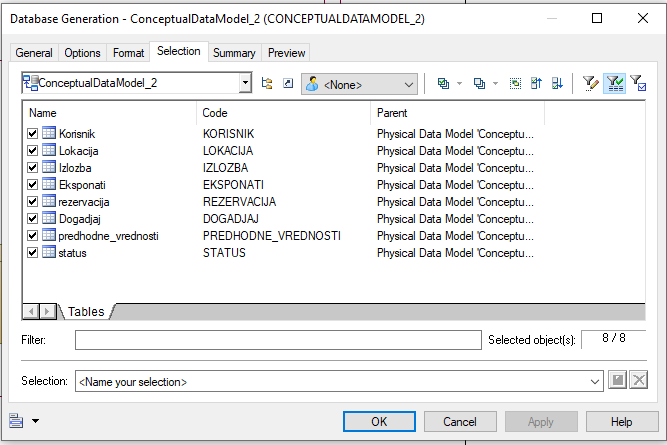
*Slika 4.2.1 – Fizički model baze podatka.*

# 5. Implementacija

Implementacija je ključna faza u razvoju sistema za rezervaciju izložbi u muzeju jer označava konkretan prelazak sa dizajna na funkcionalan softver. U ovoj fazi detaljno ćemo opisati kako smo realizovali glavne komponente sistema i kako su se te komponente integrisale u jedinstveni informacioni sistem. Implementacija obuhvata backend i frontend aspekte, kao i sve tehničke detalje koji su neophodni za pravilan rad sistema.

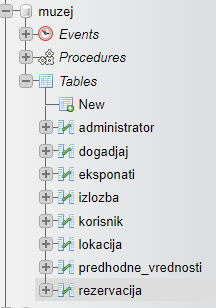
U softverskom razvoju, implementacija se najčešće odnosi na pisanje koda i razvoj informacionih sistema prema specifikacijama. Programeri koriste programski jezik i alate kako bi stvorili funkcionalni softver.

Nakon što smo detaljno definisali fizički model podataka u programu Power Designer (Slika 4.2.1), sledeći korak u implementaciji našeg informaciong sistema je bio generisanje SQL koda (Slika 5.1) iz ovog modela. Ovaj SQL kod je esencijalan jer precizno opisuje strukturu naše baze podataka, uključujući tabele, veze između njih, ključeve i ograničenja integriteta podataka. Generisanjem SQL koda iz modela osigurali smo da baza podataka tačno odražava našu zamisao i da će raditi sa našim informacionim sistemom bez problema.



*Slika 5.1 – Generisanje baze podataka.*

Kako bismo preneli ovu strukturu u radnu MySQL bazu podataka, koristili smo popularni alat za upravljanje bazama podataka po imenu phpMyAdmin (Slika 5.2). PhpMyAdmin omogućava jednostavan unos SQL koda i njegovo izvršavanje na ciljnoj MySQL bazi podataka.



*Slika 5.2 – Prikaz phpMyAdmin nakon izvršene skripte.*

Ovaj proces prenosa je ključan za stvaranje osnove naše aplikacije jer omogućava da podaci budu čvrsto integrisani sa samim sistemom. Kada se SQL kod izvrši, tabele, ključevi i veze su postavljeni, što znači da se podaci mogu uspešno čuvati, čitati i manipulisati u okviru samog informacionog sistema.

Nakon što smo uspešno generisali SQL kod iz našeg modela i preneli ga u MySQL bazu podataka putem phpMyAdmin-a, sledeći korak bio je uspostavljanje veze između naše PHP aplikacije i te baze podataka.

Da bismo omogućili PHP skriptama da komuniciraju sa MySQL bazom podataka, koristili smo PHP-ovu ugrađenu funkcionalnost za uspostavljanje veze sa bazom (Slika 5.3).

Takođe, osiguravamo se tako što pomoću ugrađene funkcije proveravamo da li je uspešno povezana baza podatka sa serevrom (Slika 5.3).

$server="localhost";// Server na kojem je MySQL baza podataka

$kor="root";// Korisničko ime za pristup bazi

$pas="";// Lozinka za pristup bazi, ostavljena prazna jer se radi na lokalnom serveru

$dbime="muzej";// Ime baze podataka kojoj želimo da pristupimo

$db=mysqli\_connect($server, $kor, $pas, $dbime);

if ($db->connect\_error) {

    die("Greška pri povezivanju sa bazom podataka: " . $db->connect\_error);

} else {

    echo "Uspešno povezano sa bazom podataka!";

}

*Slika 5.3 – Konekcija sa bazom.*

Zatvaranje veze sa bazom podataka je obavezan korak u svakom PHP projektu. To oslobađa resurse na serveru i sprečava neželjene promene podataka. Osim toga, poboljšava skalabilnost aplikacije i smanjuje rizik od bezbednosnih problema. U PHP-u, veza se zatvara jednostavnom funkcijom close(). Ovo je bitno za efikasno i bezbedno upravljanje bazom podataka.

Da bi naša aplikacija najbolje funkcionisala, odlučili smo da centralizujemo upravljanje bazom podataka u datoteci funkcije.php. Ova datoteka je srce našeg projekta i sadrži sve važne funkcije, uključujući i konekciju sa bazom podataka.

Ova strategija ima mnoge prednosti. Prvo, omogućava nam da održavamo čist i organizovan kod. Sva logika povezana sa bazom podataka je smeštena na jednom mestu, što olakšava pronalaženje i uređivanje koda. Takođe, smanjuje ponavljanje istog koda na različitim stranicama, čime se povećava efikasnost razvoja i održavanja.

Kako bismo koristili funkcije iz funkcije.php, jednostavno ih uključujemo na svakoj stranici koja zahteva pristup bazi podataka. To postižemo korišćenjem require\_once (“funkcije.php”). Ova funkcija osigurava da se datoteka učita samo jednom, bez obzira koliko puta je pozivamo, što čini naš kod čistijim i bržim.

Na ovaj način, naš inforamcioni sistem postaje stabilniji i lakši za održavanje. Ako u budućnosti bude potrebno promeniti detalje vezane za bazu podataka, to ćemo uraditi samo u funkcije.php, a te promene će automatski biti primenjene na svim stranicama koje koriste ovu datoteku. Ovo povećava efikasnost razvoja i smanjuje rizik od grešaka.

Navigacioni meni u našem informacionom sistemu ( Slika 5.4) igra važnu ulogu u olakšavanju kretanja kroz različite delove sistema. Struktura menija je planirana kako bi odražavala glavne sekcije i funkcionalnosti informacionog sistema, omogućavajući korisnicima brzo snalaženje.

Takođe, meni je prilagodljiv različitim uređajima, čineći ga upotrebljivim na računarima, tabletima i pametnim telefonima. Ovaj dizajn menija unapređuje korisničko iskustvo, čineći navigaciju brzom i jednostavnom, bez obzira na uređaj koji koristi.

<div id="mainNavigation">

      <nav role="navigation">

        <div class="py-3 text-center border-bottom">

          <a href="index.php"><img src="img/logo.png" alt="" class="invert" width='3%' height='3%' href='index.php'></a>

        </div>

      </nav>

      <div class="navbar-expand-md">

        <div class="navbar-dark text-center my-2">

          <button class="navbar-toggler w-75" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavDropdown" aria-controls="navbarNavDropdown" aria-expanded="false" aria-label="Toggle navigation">

            <span class="navbar-toggler-icon"></span> <span class="align-middle">Menu</span>

          </button>

        </div>

        <div class="text-center mt-3 collapse navbar-collapse" id="navbarNavDropdown">

          <ul class="navbar-nav mx-auto ">

            <li class="nav-item">

              <a class="nav-link active" aria-current="page" href="index.php">Početna</a>

            </li>

            <li class="nav-item">

              <a class="nav-link" href="izlozbe.php">Izlozbe</a>

            </li>

            <li class="nav-item">

              <a class="nav-link " href="dogadjaji.php">Dogadjaji</a>

            </li>

            <li class="nav-item">

            <a   style='float:right' href="admin.php">Admin deo</a>

            </li>

            <?php if(isset($\_SESSION['ime'])){

                      echo "</li>

                      <li class='nav-item'>

                      <a   style='float:right' href='login.php?odjava'>Odjavi se </a>

                      </li>";

                      }else{

                      echo "</li>

                      <li class='nav-item'>

                      <a   style='float:right' href='login.php?odjava'>Prijavi se</a>

                      </li>";

                      }

    ?>

*Slika 5.4 – kod za prikazivanje navigacionog menija aplikacije.*

Registracija korisnika je ključni korak za pristupanje platformi. Korisnici popunjavaju formu za registraciju, unoseći svoje osnovne informacije, uključujući ime, prezime, e-mail adresu i lozinku.

Da bismo osigurali tačnost podataka, implementirali smo validaciju unosa. To uključuje proveru ispravnosti e-mail adrese i provravanje da li postoji već kreiran nalog za unetu e-mail adresu. Nakon validacije, korisnikov nalog se kreira u bazi podataka (Slika 5.5).

if(isset($\_POST['btnreg'])){

    $ime=$\_POST['ime'];

    $prezime=$\_POST['prezime'];

    $email=$\_POST['email'];

    $password=$\_POST['password'];

    $broj\_telefona=$\_POST['telefon'];

$query = "SELECT email FROM korisnik WHERE email='$email'";

$result = mysqli\_query($db, $query);

if(mysqli\_num\_rows($result) > 0){

    echo "<div class='alert alert-danger'>Email postoji u bazi podataka";

}else{

$upit="INSERT INTO korisnik (ime, prezime, email, broj\_telefona, status, password) VALUES('$ime', '$prezime', '$email', '$broj\_telefona', 'korisnik','$password')";

$rez=mysqli\_query($db, $upit) ;

echo "<div class='alert alert-success' >Uspesno ste se registrovali<br></div>";

echo "<div class='d-grid gap-2'>

<a class='btn btn-success'  name='log' href='login.php'>Uloguj se</a>

</div>";

*Slika 5.5 – Unos novog korisnika u bazu.*

Nakon uspešne registracije ili prijavljivanja na platformu, korisnikova interakcija postaje personalizovana i sigurna zahvaljujući sesijama i kolačićima (Slika 5.6 I Slika 5.7).

Kolačići su male tekstualne datoteke na uređaju korisnika. Oni pamte informacije između sesija. Najčešće čuvaju autentifikacione podatke, tako da korisnici ostaju prijavljeni između poseta. Takođe, mogu se koristiti za personalizaciju i analizu ponašanja korisnika.

setcookie("id", $row["id"] , time()+60\*60\*24\*30);

setcookie("email", $email, time()+60\*60\*24\*30);

setcookie("ime", $row['ime'], time()+60\*60\*24\*30);

setcookie("status", $row['status'], time()+60\*60\*24\*30);

*Slika 5.6 – Kreiranje kolačića.*

Ukoliko se pronađu odgovarajući kolačići, funkcija (Slika 5.7) koristi njihove informacije za identifikaciju korisnika. Na osnovu tih podataka, kreira se ili obnavlja sesija za korisnika. Sesija je jedinstvena za svakog korisnika i čuva važne informacije tokom njihove posete, kao što su identitet, privilegije i druge podatke.

function proveraKolacica(){

//Ako kolačići postoje generišu se promenljive sesije

if(isset($\_COOKIE['id']) and isset($\_COOKIE['email']) and isset($\_COOKIE['ime'])and isset($\_COOKIE['status'])){

$\_SESSION['id']=$\_COOKIE['id'];

$\_SESSION['email']=$\_COOKIE['email'];

$\_SESSION['ime']=$\_COOKIE['ime'];

$\_SESSION['status']=$\_COOKIE['status'];}

*Slika 5.7 – Funkcija za kreiranje promenjivih sesija.*

Da bismo osigurali pristup samo registrovanim korisnicima na određenim delovima web stranice, implementiramo funkcionalnost za proveru korisničkog statusa. Na početku stranice s ograničenim pristupom, sistem proverava da li je korisnik prijavljen ili registrovan (Slika 5.8). Ovo se postiže putem funkcije (Slika 5.9) koja proverava prisustvo odgovarajućih sesija ili kolačića na uređaju korisnika.

Ako sistem utvrdi da korisnik nije prijavljen, automatski će ga preusmeriti na stranicu za prijavljivanje. Ovim se osigurava da samo registrovani korisnici mogu pristupiti zaštićenim resursima.

<?php

    session\_start();

    require\_once('funkcije.php');

    proveraKolacica();

    prijava();

    ?>

*Slika 5.8 – Provera na početku stranice.*

function prijava()

{

//Ako korisnik nije prijavljen odmah se prosleđuje na stranicu za prijavu

if(!isset($\_SESSION['id'])) header("Location: login.php");

}

*Slika 5.9 – Provera prijavljenog korisnika.*

Naš informacioni sistem, kao što je prikazano na slici (Slika 5.4), omogućava korisnicima koji su uspešno ulogovani na svoje naloge da jednostavno i sigurno upravljaju svojim sesijama i odjave se sa sistema. Ovaj deo korisničkog iskustva je ključan kako bi se osiguralo da korisnici imaju potpunu kontrolu nad svojim interakcijama sa sistemom.

Kada je korisnik ulogovan, primetiće prisustvo opcije “Odjavi se” na navigacionom meniju ili sličnom mestu na web stranici. Kada korisnik odabere ovu opciju i pritisne dugme “Odjava”, web aplikacija šalje zahtev serveru da se obavi operacija odjave. Ovaj zahtev sadrži informacije koje potvrđuju da je korisnik inicirao postupak odjave.

Nakon što server primi ovaj zahtev, pokreće se proces završetka sesije korisnika. Ovo uključuje gašenje ili “ubijanje” trenutne sesije, čime se prekida veza između korisnika i sistema (Slika 5.10). Osim toga, svi kolačići koji su postavljeni tokom sesije korisnika se poništavaju kako bi se osiguralo da nijedni osetljivi podaci ne ostanu dostupni na uređaju korisnika nakon odjave.

function odjava(){

//Ako se korisnik odjavljuje, uništavaju se promenljive sesije, sesija i kolačići

setcookie("id", "", time()-1);

setcookie("email", "", time()-1);

setcookie("ime", "", time()-1);

setcookie("status", "", time()-1);

unset($\_SESSION['id']);

unset($\_SESSION['email']);

unset($\_SESSION['ime']);

unset($\_SESSION['status']);

session\_destroy();

}

*Slika 5.10 – “Ubijanje” sesija i prekiadnje kolačića.*

Administrativni deo sistema (Admin deo) (Slika 5.4) predstavlja privilegovani segment informacionog sistema koji je rezervisan isključivo za korisnike sa odgovarajućim administratorskim privilegijama. Ovaj deo sistema omogućava administratorima da vrše napredne funkcije upravljanja, nadzora i održavanja.

Prava pristupa admin delu imaju samo korisnici koji su registrovani kao administratori i imaju odgovarajući status na svojim korisničkim nalozima. To znači da samo admnistratori mogu pristupiti ovom delu sistema, čime se čuva sigurnost i kontrola nad administrativnim funkcionalnostima.

Administrator sistema ima pristup posebnom delu informacionog sistema (“admin.php”), gde mu se nude različite funkcionalnosti za efikasno upravljanje sistemom. Ove funkcionalnosti uključuju mogućnost kreiranja novih izložbi, dodavanja eksponata, premeštanja eksponata između lokacija, kreiranja događaja, pregleda rezervacija i brisanja rezervacija.

Kreiranje izložbi omogućava administratoru da unese osnovne informacije o izložbi, uključujući naziv, datum, opis, sliku, lokaciju i izložbu. Ove informacije se čuvaju u sistemu i postaju dostupne korisnicima za rezervaciju.

Dodavanje eksponata omogućava administratoru da unese nove eksponate u bazu podataka sistema. Prilikom dodavanja eksponata, unosimo detalje kao što su naziv eksponata, slike, opise i druge relevantne informacije. Administrator takođe može povezati svaki eksponat sa odgovarajućom izložbom.

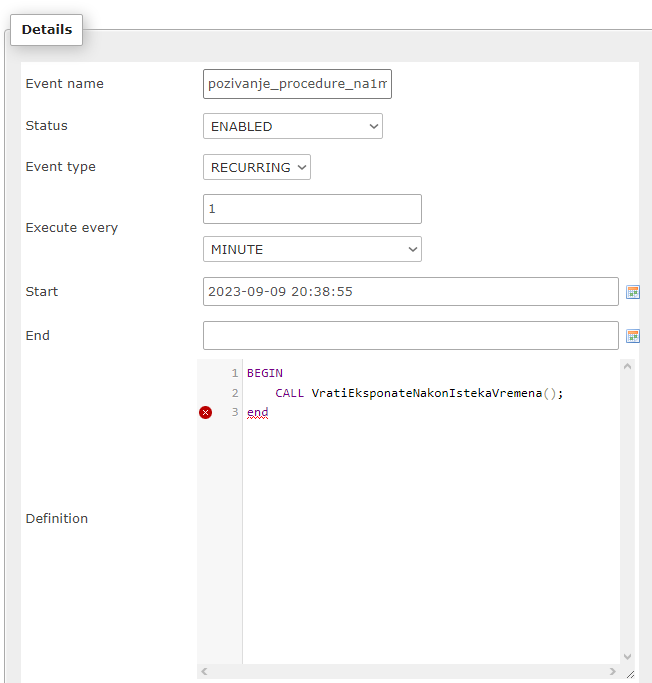
Premeštanje eksponata sa jedne lokacije na drugu daje administratoru kontrolu nad organizacijom izložbi i njihovim rasporedom na različitim lokacijama.

Kreiranje događaja omogućava administratoru da planira posebne događaje povezane sa izložbama. Svaki događaj ima svoj datum i vreme održavanja, kao i datum i vreme završetka. Događaj je specifičan zato što može biti na lokaciji nezavisnoj od lokacija koje se nalaze u bazi podataka. Kreiranjem događaja vrednosti eksponata (“id\_eksponata i id\_izlozbe”) smeštamo u tabelu prethodne vrednosti (Slika 5.2), nakon što dogadjaj istekne putem eventa (Slika 5.11) pozivamo funkciju za vraćanje eksponata u predhodne izložbe (Slika 5.12).

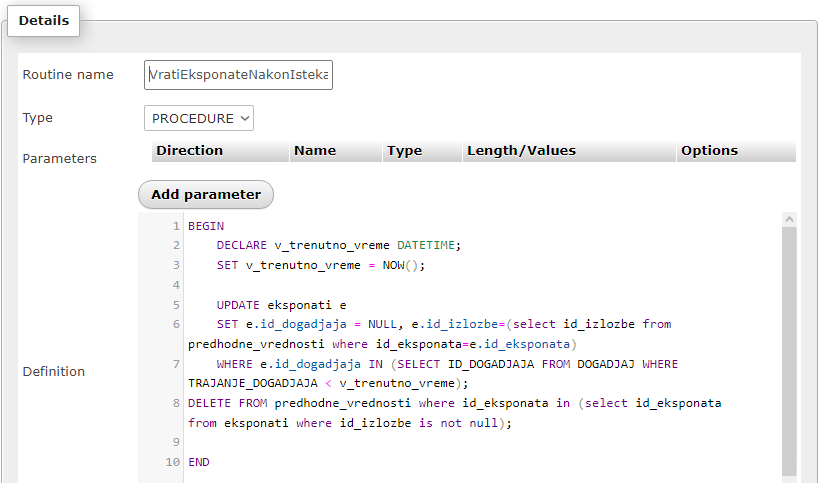
Pregled rezervacija daje administratoru uvid u sve rezervacije koje su napravili korisnici. Ovo omogućava praćenje popunjenosti izložbi i događaja.

Brisanje rezervacija omogućava administratoru da obriše postojeće rezervacije po potrebi ili na zahtev korisnika, pružajući fleksibilnost u upravljanju rezervacijama.

Sve ove funkcionalnosti su ključne za efikasno upravljanje sistemom za rezervaciju izložbi i omogućavaju administratorima da održavaju aktuelne informacije i pružaju podršku korisnicima.



*Slika 5.11 - Event koji poziva proceduru.*

**

*Slika 5.12 – Procedura za vraćanje predhodnih vrednosti.*

# 6. Testiranje

Testiranje našeg informacionog sistema za rezervaciju izložbi izvedeno je sa dva ključna aspekta: korisničke i administratorske strane.

Na korisničkoj strani, fokusirali smo se na temeljno testiranje svih osnovnih funkcionalnosti koje su dostupne korisnicima. Ovo uključuje procese pretrage izložbi, eksponata i događaja. Proverili smo kako korisnici efikasno pretražuju i pregledaju sadržaj, rezervišu karte za izložbe, kao i kako uređuju ili otkazuju svoje rezervacije. Takođe smo analizirali interakciju sa sistemom za registraciju i prijavu, proveravajući da li korisnici mogu bez problema da kreiraju naloge, prijave se i upravljaju svojim profilima.

Testiranje sa korisničke strane takođe je uključivalo proveru korisničkog iskustva (UX). Analizirali smo da li je korisnički interfejs intuitivan i lak za korišćenje. Takođe smo testirali odzivnost dizajna na različitim uređajima, uključujući računare, tablete i mobilne telefone, kako bismo osigurali da korisnici imaju dosledno iskustvo bez obzira na uređaj koji koriste.

Na administratorskoj strani, fokusirali smo se na testiranje svih funkcionalnosti koje su dostupne administratorima. Ovo uključuje kreiranje izložbi, dodavanje eksponata, premještanje eksponata između lokacija, kreiranje događaja, pregled rezervacija i brisanje rezervacija.

## 6.1 Testiranje korisničkih zahteva

Početna tačka svake interakcije između korisnika i aplikacije je stranica index.php, koja predstavlja osnovni korisnički interfejs. Ova stranica sadrži navigacioni meni (Slika 6.1.1) koji omogućava korisnicima brz pristup različitim delovima aplikacije. Navigacioni meni obuhvata ključne sekcije kao što su “Početna”, “Izložbe”, “Događaji”, “Moje rezervacije”i link ka “Admin” delu.

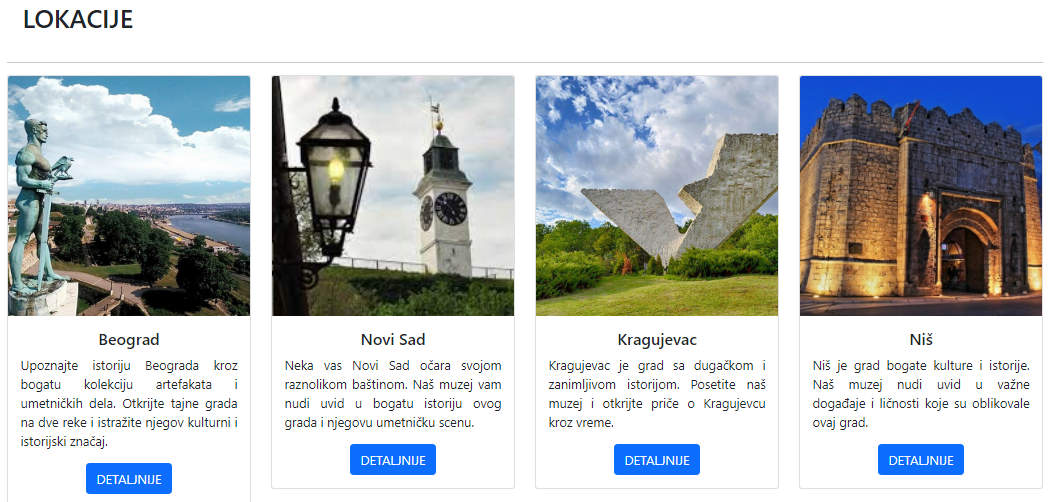
Stranica “Početna” predstavlja informativnu tačku gde korisnici mogu pronaći osnovne informacije o aplikaciji, muzejima, izložbama i događajima. Ova stranica služi kao početna tačka za istraživanje i upoznavanje sa ponudom aplikacije.

“Izložbe” i “Događaji” su sekcije gde korisnici mogu pronaći detaljne informacije o aktuelnim izložbama i događajima. Koristeći navigacioni meni, korisnici mogu brzo pregledati dostupne opcije i pronaći ono što ih interesuje. Svaka od ovih stranica omogućava korisnicima da saznaju više o pojedinostima, datumima i vremenima, kao i da rezervišu svoje mesto za posetu.



*Slika 6.1.1 – Navigacioni meni aplikacije.*

Na početnoj stranici aplikacije ističu se dostupne lokacije muzeja (Slika 6.1.2), svaka praćena sažetim opisom. Ovaj prikaz omogućava korisnicima da brzo pregledaju različite lokacije i steknu uvid u njihovu privlačnost. Klikom na opciju "Detaljnije", korisnici dobijaju pristup obimnijim informacijama o izložbama, eksponatima i događajima koji se nalaze na odabranoj lokaciji. Ova funkcionalnost omogućava korisnicima dublje istraživanje muzejskih sadržaja i olakšava im planiranje poseta specifičnim lokacijama koje ih interesuju.

**

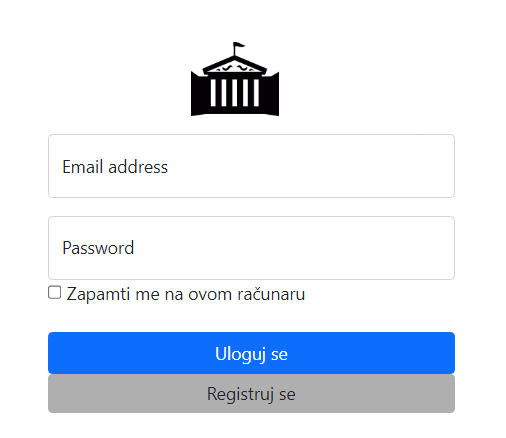
*Slika 6.1.2 – Lokacije muzeja.*

Korisnik ima mogućnost da pristupi stranici izložbi, gde mu se otvaraju različite opcije. Može da rezerviše posetu nekoj od izložbi ili da detaljnije (Slika 6.1.3) istraži informacije o izložbi, uključujući opis i kompletan pregled eksponata koji se nalaze unutar nje.



*Slika 6.1.3 – Primer izložbe.*

Ukoliko korisnik nema registrovani nalog, postoji mogućnost da ga kreira na stranici za prijavu (Slika 6.1.4), što mu omogućava da iskoristi sve prednosti sistema rezervacije i istraživanja muzejskih sadržaja. Ova pristupna tačka omogućava korisnicima personalizovano iskustvo u aplikaciji, čineći je pristupačnom i funkcionalnom.



*Slika 6.1.4 – Login aplikacije.*

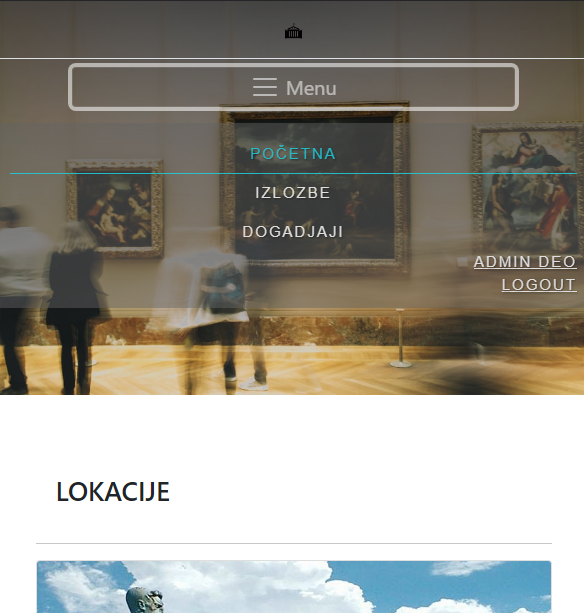
Sistem omogućava korisniku da jednostavno rezerviše svoju odabranu izložbu putem intuitivnog interfejsa. Klikom na link "Rezerviši" (vidljiv na Slici 6.1.3) otvaraju se opcije za rezervaciju. Korisnik ima mogućnost da pažljivo odabere datum i vreme posete, prilagođavajući svoj plan u skladu sa svojim preferencama i slobodnim vremenom. Nakon što izvrši rezervaciju, korisnik je uspešno zakazao svoju posetu muzeju za odabranu izložbu.

Takođe, korisnik može da pristupi svojim rezervacijama klikom na stranicu “Moje rezervacije” na navigacionom meniju (Slika 6.1.1) gde može da upravlja svojim rezervacijam (Slika 6.1.5).



*Slika 6.1.5 – Prikaz naših rezervacija.*

Dizajn informacionog sistema je usmeren na pružanje optimalnog korisničkog iskustva, a jedan od ključnih elemenata tog dizajna je njegova responsivnost (Slika 6.1.6). Ova karakteristika omogućava da se informacioni sistem prilagodi različitim vrstama uređaja, kao što su mobilni telefoni, tableti i računari.



*Slika 6.1.6 – Respozivnost menija.*

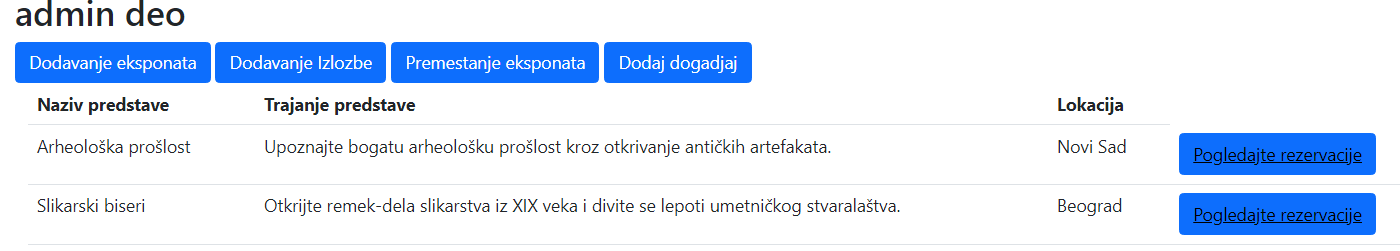
## 6.2 Testiranje administrativnih zahteva

Administrator ima specifičan pristup administrativnom delu sistema, čime mu se omogućava upravljanje ključnim aspektima aplikacije. Da bi pristupio ovom delu, administrator koristi poseban link u navigacionom meniju, koji je označen kao "ADMIN DEO". Ovaj link se jasno ističe na početnoj stranici sistema (Slika 6.1.1), i klikom na njega, administratora se preusmerava na stranicu poznatu kao admin.php.

Na stranici admin.php (Slika 6.2.1), administratoru su dostupne različite funkcionalnosti i alatke koje su dizajnirane za efikasno upravljanje sistemom. Ove funkcionalnosti uključuju kreiranje izložbi, dodavanje eksponata, premestanje eksponata između lokacija, kreiranje događaja, pregled rezervacija i brisanje rezervacija. Administrator ima potpunu kontrolu nad ovim operacijama i može ih izvršavati prema potrebama i zahtevima muzeja.

Ovaj odvojeni pristup za administratore je ključan za očuvanje sigurnosti i integriteta sistema. Administratorske funkcionalnosti su obično zaštićene jakim lozinkama i drugim sigurnosnim mehanizmima kako bi se sprečio neovlašćeni pristup i eventualni rizici za sistem. Osim toga, administratorima se pruža poseban nivo obuke i podrške kako bi se osiguralo da pravilno koriste ove alatke.

Pristup administrativnom delu (Slika 6.2.1) omogućava administratorima da efikasno upravljaju sadržajem sistema, prate rezervacije, i obezbede kvalitetno korisničko iskustvo za sve posetioce muzeja. S obzirom na njihovu ključnu ulogu u vođenju sistema, administratori su ključni akteri u održavanju i unapređenju aplikacije za rezervaciju izložbi.



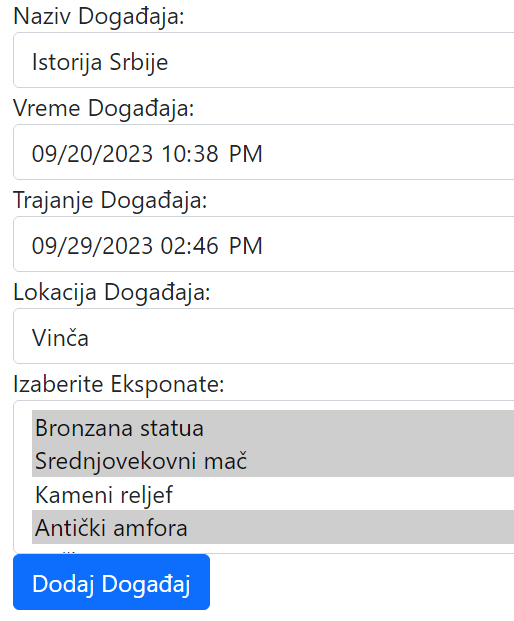
*Slika 6.2.1 – Administrativni deo*

Sistem za rezervaciju izložbi omogućava korisnicima i administratorima pregled i upravljanje rezervacijama. Kroz opciju “Pogledaj rezervacije”, korisnici mogu proveriti svoje rezervacije za određenu izložbu, uključujući datum, vreme i broj karata. Takođe, mogu obrisati rezervacije po potrebi, što olakšava prilagođavanje planova.

Dodavanje eksponata, kreiranje izložbi i premeštanje eksponata su ključne funkcionalnosti u administrativnom delu sistema. Ove funkcionalnosti se obavljaju kroz odgovarajuće obrasce sa određenim validacijama kako bi se obezbedila tačnost i doslednost unosa podataka.

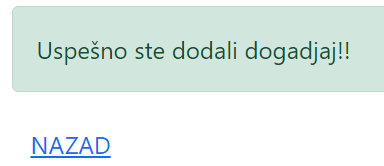
Na primeru dodavanja događaja (Slika 6.2.1), ovaj proces obuhvata više koraka. Administrator unosi osnovne informacije o događaju, kao što su naziv događaja, lokacija, datum i vreme, cena ulaznice i druge relevantne informacije. Ovi podaci se validiraju kako bi se sprečilo unošenje netačnih ili nedostajućih informacija.

Nakon unosa podataka i validacije, sistem čuva informacije o događaju u bazi podataka, a korisnici će moći da vide ovaj događaj na javnom delu sistema.



*Slika 6.2.2 – Forma za dodavanje događaja*

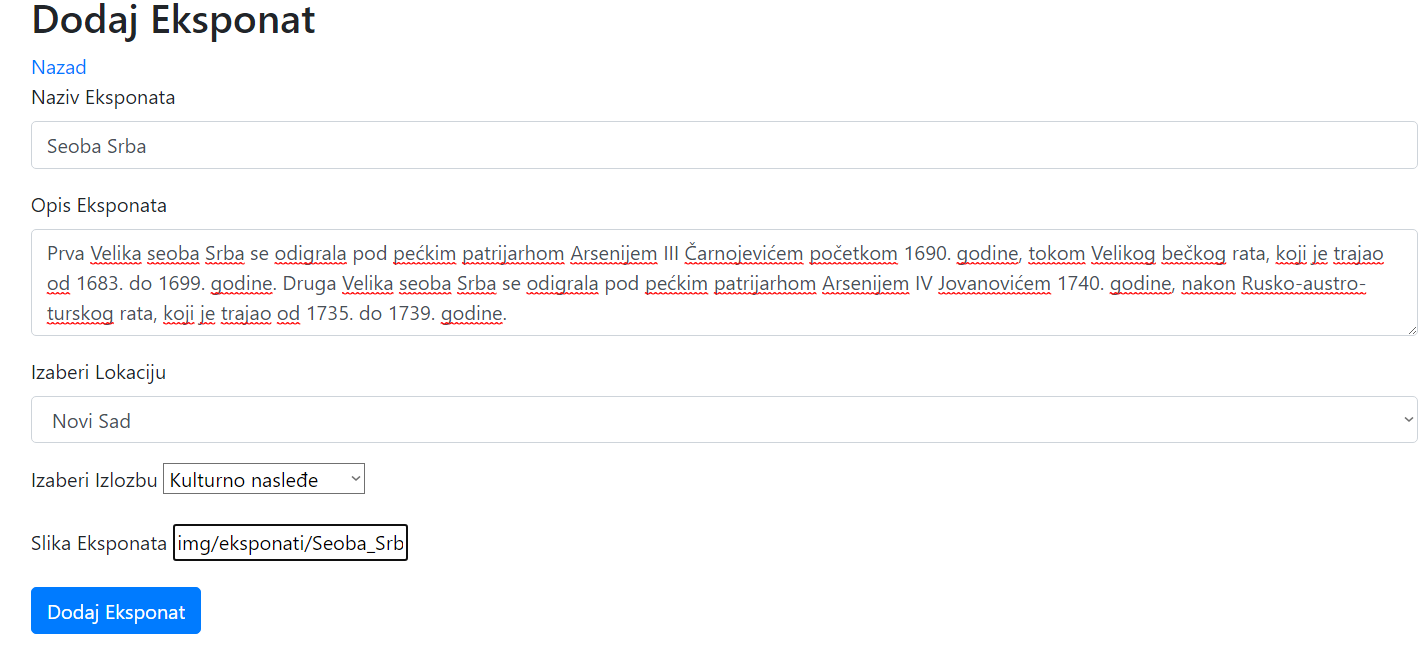
Kada su sva polja pravilno popunjena i uslovi za dodavanje događaja ispunjeni, sistem pruža povratnu informaciju o uspešnoj operaciji dodavanja događaja (Slika 6.2.3).



*Slika 6.2.3 – Potvrda o uspešnom dodavanju događaja.*

Slično, proces dodavanja eksponata i kreiranja izložbi takođe uključuje unos podataka, validaciju i čuvanje tih informacija u bazi podataka.

Proces dodavanja eksponata u sistem zahteva precizno popunjavanje svih relevantnih informacija o eksponatu, uključujući naziv, opis, kategoriju, i druge detalje. Administrator zatim odabira lokaciju i izložbu gde će eksponat biti smešten, čime se osigurava tačno mesto za izlaganje. Nakon unosa svih podataka i izbora opcija, administrator potvrđuje dodavanje eksponata, a sistem pruža povratnu informaciju o uspešnom dodavanju (Slika 6.2.5, slika 6.2.6).



*Slika 6.2.4 – Forma za dodavanje eksponata*



*Slika 6.2.5 – Potvrda o uspešno dodatom eksponatu.*

# 7. Zaključak

Ovaj projekat tj. završni rad predstavlja svojevrsno putovanje kroz različite faze, od konceptualnog planiranja do implementacije i testiranja, koristeći različite tehnologije.

U okviru ovog rada, ostvareni su značajni koraci u razvoju sistema. Analizirane su i primenjene tehnologije koje su omogućile stvaranje responzivnog i funkcionalnog infromacionog sistema. XAMPP kao alat za lokalni razvoj omogućio je simulaciju servera, PHP je poslužio kao jezik za serversku logiku, dok su JavaScript, Ajax i jQuery doprineli interaktivnosti korisničkog interfejsa. HTML i CSS su bili ključni za dizajn, dok je Bootstrap unapredio korisničko iskustvo. SQL je koršćen za upravljanje bazom podataka, što je bilo od suštinskog značaja za čuvanje podataka o izložbama, eksponatima, korisnicima i rezervacijama.

Modelovanje sistema kroz Use Case dijagrame, dijagrame sekvenci i dijagrame aktivnosti omogućilo je jasno definisanje korisničkih zahteva i procesa unutar sistema. Ovi modeli olakšali su razumevanje funkcionalnosti sistema, što je bilo od suštinskog značaja za uspešnu implementaciju.

Kroz implementaciju, razvijen je funkcionalan sistem koji omogućava korisnicima da pregledaju izložbe, rezervišu posete i prate događaje u muzeju. Administratori imaju pristup dodavanju eksponata, kreiranju izložbi, upravljanju rezervacijama i drugim ključnim funkcionalnostima.

Testiranje sistema bilo je od suštinskog značaja kako bismo osigurali da informacioni sistem radi ispravno i da ispunjava korisničke zahteve. Testiranje korisničkih zahteva obuhvatilo je proveru funkcionalnosti sistema, od pregleda izložbi do rezervacija. Takođe smo testirali i administrativne funkcionalnosti kako bismo osigurali da su one dostupne samo administratorima i da rade kako je predviđeno.

Zaključujemo da ovaj rad predstavlja korak napred u razvoju informacionih sistema za muzeje. Korišćene tehnologije i alati omogućili su stvaranje sistema koji je responzivan, interaktivan i lak za korišćenje. Međutim, u budućnosti bi bilo korisno razmotriti dodatne funkcionalnosti kao što su statistički izveštaji o rezervacijama i analiza poseta izložbama. Takođe, možemo razmisliti o unapređenju korisničkog interfejsa kako bismo ga učinili još intuitivnijim. U svakom slučaju, ovaj rad predstavlja osnovu za dalji razvoj informacionog sistema za muzeje i pruža osnovu za poboljšanje korisničkog iskustva posetilaca muzeja.

# 8. Indeks pojmova

**A.**

Ajax 4, 6

Admnistrator 5, 7 , 10, 14, 15, 16, 28, 31, 34, 35, 36

**B.**

Baza 5, 6, 7, 18, 20, 22, 23

**I.**

Inforamcioni sistem 5, 10, 16, 18, 20, 33, 37

**K.**

Kod 7, 22, 23, 25

Korisnik 5, 9, 10, 12, 13, 14, 16, 18, 19, 25, 26, 27, 28, 31, 32, 33

**M.**

Model 5, 8, 9, 12, 18, 19, 20, 21, 22, 23

Mysql 6, 7, 22, 23, 26

**P.**

PHP 4, 6, 22, 23, 24, 26, 27, 28, 31, 34

# 9. Literatura

[1] Gabrijela Dimić, Boško Bogojević, *Programiranje veb aplikacija*, *Visoka škola elektrotehnike i računarsva*, Beograd, 2022.

[2] Dr. Pavle Kaluđeričić, Dr. Slobodan Obradović, *Projektovanje infromacionih sistema relacione baze podataka*, *Visoka škola elektrotehnike i računarsva* Beograd, 2012

Preuzeto sa: <https://www.viser.edu.rs/uploads/2018/09/24.pdf>

Preuzeto avgust.2023

[2] Visual Paradigam:

<https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/>

[3] Stack Overflow: <https://stackoverflow.com/>

[4] Php Documentation: <https://www.php.net/docs.php>

[5] JavaScript: <https://www.w3schools.com/js/>, <https://www.tutorialspoint.com/javascript/index.htm>

# 10. IZJAVA O AKADEMSKOJ ČESTITOSTI

|  |  |
| --- | --- |
| **Student (ime, ime jednog roditelja i prezime):** | **Danilo Milanović** |
| **Broj indeksa:** | **is-21/20** |

Pod punom moralnom, materijalnom, disciplinskom i krivičnom odgovornošću izjavljujem da je završni rad, pod naslovom:

**Impelementacija informacionog sistema na primeru muzeja**

1. rezultat sopstvenog istraživačkog rada;

2. da ovaj rad, ni u celini, niti u delovima, nisam prijavljivao/la na drugim visokoškolskim ustanovama;

3. da nisam povredio/la autorska prava, niti zloupotrebio/la intelektualnu svojinu drugih lica;

4. da sam rad i mišljenja drugih autora koje sam koristio/la u ovom radu naznačio/la ili citirao/la u skladu sa Uputstvom;

5. da su svi radovi i mišljenja drugih autora navedeni u spisku literature/referenci koji je sastavni deo ovog rada, popisani u skladu sa Uputstvom;

6. da sam svestan/svesna da je plagijat korišćenje tuđih radova u bilo kom obliku (kao citata, parafraza, slika, tabela, dijagrama, dizajna, planova, fotografija, filma, muzike, formula, veb sajtova, računarskih programa i sl.) bez navođenja autora ili predstavljanja tuđih autorskih dela kao svojih, kažnjivo po zakonu (Zakon o autorskom i srodnim pravima), kao i drugih zakona i odgovarajućih akata Visoke škole elektrotehnike i računarstva strukovnih studija u Beogradu;

7. da je elektronska verzija ovog rada identična štampanom primerku ovog rada i da pristajem na njegovo objavljivanje pod uslovima propisanim aktima Visoke škole elektrotehnike i računarstva strukovnih studija u Beogradu;

8. da sam svestan/svesna posledica ukoliko se dokaže da je ovaj rad plagijat.

U Beogradu, \_\_. \_\_. 201\_. godine

Svojeručni potpis studenta

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_