

## Kratki opis korištenja aplikacije:

Nakon što neregistrovani korisnik pokrene aplikaciju, otvara se prozor sa listom predstava za trenutni mjesec. U gornjem lijevog uglu postoji opcija prijave admina, dok kraj svake predstave u listi postoji opcija „rezerviši“. Izabравši tu opciju, neregistrovani korisnik postaje registrovani korisnik (kupac/premium kupac) nakon što ispuni obrazac za registraciju ili prijavu (ukoliko posjeduje račun od ranije). Ukoliko je registracija/prijava uspješna, korisniku se prikazuje slika sa sjedištima i ima mogućnost rezervacije, koju poslije može i otkazati.

## Klase:

### Korisnik

Opis:	Apstraktna klasa
Atributi:	<ul style="list-style-type: none"><li>• ime: String</li><li>• prezime: String</li><li>• username: String</li><li>• password: int</li><li>• e-mail: String</li></ul>
Metode:	<ul style="list-style-type: none"><li>• <b>login</b>(in username:String, in password:int)</li><li>• logout()</li><li>• <b>changePassword</b>(in oldPassword:String, in newPassword:String)</li><li>• <b>changeUsername</b>(in oldUsername:String, in newUsername:String)</li><li>• ostali geteri i seteri</li></ul>

### NeregistrovaniKorisnik

Opis:	Klasa koja implementira IPregledPredstava interfejs
Atributi:	/
Metode:	<ul style="list-style-type: none"><li>• <b>NeregistrovaniKorisnik()</b></li><li>• <b>registracija</b>(in kupacID:int, in ime:String, in prezime:String, in e-mail:String, in password:int)</li><li>• <b>login</b>(in username:String, in password:int)</li></ul>

### Kupac

Opis:	Klasa naslijeđena iz Korisnik klase, implementira IPregledPredstava interfejs
-------	---

Atributi:	<ul style="list-style-type: none"> <li>• ID: int (statički atribut)</li> <li>• kupacID: int = ID</li> </ul>
Metode:	<ul style="list-style-type: none"> <li>• <b>Kupac</b>(in ime:String, in prezime:String, in e-mail:String, in password:int)</li> <li>• getKupacID(): int</li> <li>• <b>rezervisiKarte</b>(in predstava:Predstava, in rezervacija:Rezervacija, in sjedista:List&lt;Sjediste&gt;)</li> <li>• <b>rezervisiKartu</b>(in predstava:Predstava, in rezervacija:Rezervacija, in sjediste:Sjediste)</li> <li>• <b>otkaziRezervaciju</b>(in rezervacija:Rezervacija)</li> <li>• <b>promijeniStatusUPremium</b>(in vrstaKartice:Kartica)</li> </ul>

### PremiumKupac

Opis:	Klasa naslijeđena iz Kupac klase
Atributi:	<ul style="list-style-type: none"> <li>• karticaID: int</li> <li>• vrstaKartice: Kartica</li> </ul>
Metode:	<ul style="list-style-type: none"> <li>• Konstruktor i geteri</li> </ul>

### Admin

Opis:	Klasa naslijeđena iz Korisnik klase, implementira IPregledPredstava interfejs
Atributi:	<ul style="list-style-type: none"> <li>• ID: int (statički atribut)</li> <li>• adminID: int = ID</li> </ul>
Metode:	<ul style="list-style-type: none"> <li>• <b>-Admin()</b></li> <li>• <b>getAdminId(): int</b></li> </ul>

### AdminController

Opis:	Klasa pomoćna za Admin klasu
Atributi:	<ul style="list-style-type: none"> <li>• admin: Admin</li> </ul>
Metode:	<ul style="list-style-type: none"> <li>• <b>AdminController</b>(in admin:Admin)</li> <li>• <b>prikaziPredstave</b>(in pozoriste:Pozoriste)</li> <li>• <b>prikaziKupce</b>(in pozoriste:Pozoriste)</li> <li>• <b>prikaziDogadjaje</b>(in pozoriste:Pozoriste)</li> </ul>

	<ul style="list-style-type: none"> <li>• <b>prikaziRezervacije</b>(in pozoriste:Pozoriste)</li> <li>• <b>otkaziRezervaciju</b>(in pozoriste:Pozoriste, in rezervacija:Rezervacija)</li> <li>• <b>azurirajPredstavu</b>(in pozoriste:Pozoriste, in predstava:Predstava, in datum:DateTime, in cijenaSjedista:double)</li> <li>• <b>azurirajDogadjaj</b>(in pozoriste:Pozoriste, in dogadjaj:Dogadjaj, in naziv:String, in opis:String, in listaGlumaca:List&lt;String&gt;, in slika:String)</li> </ul>
--	---

### Predstava

Opis:	Klasa koja predstavlja predstavu sa tačnim datumom i vremenom izvođenja
Atributi:	<ul style="list-style-type: none"> <li>• predstavaID: int = ID</li> <li>• dogadjaj: Dogadjaj</li> <li>• datum: DateTime</li> <li>• cijenaSjedista: double</li> <li>• listaPredstava: List&lt;Predstava&gt; (statički atribut)</li> <li>• ID: int = 0 (statički atribut)</li> </ul>
Metode:	Konstruktor, geteri, seteri

### Dogadjaj

Opis:	Klasa koja predstavlja predstavu dostupnu za izvođenje, tj. sastavni dio predstava (jedne ili više njih)
Atributi:	<ul style="list-style-type: none"> <li>• dogadjajID: int = ID</li> <li>• naziv: String</li> <li>• opis: String</li> <li>• listaGlumaca: List&lt;String&gt;</li> <li>• slika: String</li> <li>• listaDogadjaja: List&lt;Dogadjaj&gt; (statički atribut)</li> <li>• ID: int = 0 (statički atribut)</li> </ul>
Metode:	Konstruktor, geteri, seteri

### Sjedište

Opis:	Klasa koja implementira interfejs IRacunCijene
Atributi:	<ul style="list-style-type: none"><li>• sjedisteID: int = ID</li><li>• redSjedista: char</li><li>• brojSjedista: int</li><li>• predstava: Predstava</li><li>• rezervacija: Rezervacija = null</li><li>• slobodno: boolean = true</li><li>• popust: int = 0</li><li>• ukupanBrojSjedista: int (statički atribut)</li><li>• ID: int = 0 (statički atribut)</li></ul>
Metode:	Konstruktor, geteri, seteri

### Rezervacija

Opis:	Klasa
Atributi:	<ul style="list-style-type: none"><li>• rezervacijaID: int = ID</li><li>• kupac: Kupac</li><li>• predstava: Predstava</li><li>• ukupniTrosak: double = 0</li><li>• rezervisanaSjedista: List&lt;Sjediste&gt;</li><li>• ID: int = 0 (statički atribut)</li></ul>
Metode:	<ul style="list-style-type: none"><li>• Konstruktor, destruktork, geteri, seteri</li><li>• <b>addToRezervisanaSjedista</b>(in sjediste:Sjediste): void</li></ul>

### Kartica

Opis:	Enumeration (pobrojani tip)
Atributi:	<ul style="list-style-type: none"><li>• zlatna</li><li>• srebrna</li></ul>
Metode:	/

### Pozorište

Opis:	Singleton klasa koja implementira IBaza interfejs
Atributi:	instanca: Pozoriste predstave: List<Predstava> dogadjaji: List<Dogadjaj> rezervacije: List<Rezervacija> korisnici: List<Korisnik>

Metode:	<ul style="list-style-type: none"> <li>• <b>Pozoriste()</b></li> <li>• <b>getInstanca():</b> Pozoriste</li> <li>• <b>dajRezervacije():</b> List&lt;Rezervacija&gt;</li> <li>• <b>dajKorisnici():</b> List&lt;Korisnik&gt;</li> <li>• <b>dajPredstave():</b> List&lt;Predstava&gt;</li> <li>• <b>dajDogadjaje():</b> List&lt;Dogadjaj&gt;</li> <li>• <b>dodajPredstavu</b>(in dogadjaj:Dogadjaj, in datum:DateTime, in cijenaSjedista:double)</li> <li>• <b>izbrisiPredstavu</b>(in predstava:Predstava)</li> <li>• <b>urediPredstavu</b>(in predstava:Predstava, in datum:DateTime, in cijenaSjedista:double)</li> <li>• <b>dodajDogadjaj</b>(in naziv:String, in opis:String, in listaGlumaca:List&lt;String&gt;, in slika:String)</li> <li>• <b>izbrisiDogadjaj</b>(in dogadjaj:Dogadjaj)</li> <li>• <b>urediDogadjaj</b>(in dogadjaj:Dogadjaj, in naziv:String, in opis:String, in listaGlumaca:List&lt;String&gt;, in slika:String)</li> <li>• <b>izbrisiRezervaciju</b>(in rezervacija:Rezervacija)</li> </ul>
---------	---

### IRacunCijene

Opis:	interfejs
Atributi:	/
Metode:	<ul style="list-style-type: none"> <li>• <b>uracujPopustUCijenu():</b> void</li> <li>• <b>setUkupniTrosak():</b> void</li> </ul>

### IBaza

Opis:	interfejs
Atributi:	/

Metode:	<ul style="list-style-type: none"> <li>• <b>ažurirajBazu(): void</b></li> </ul>
---------	---

### IPregledPredstava

Opis:	interfejs
Atributi:	/
Metode:	<ul style="list-style-type: none"> <li>• <b>prikaziPredstave(): void</b></li> </ul>

### *Povezanost klasa:*

Iz apstraktne klase **Korisnik** naslijeđene su klase **Admin** i **Kupac**, dok je iz klase **Kupac** naslijeđena klasa **PremiumKupac**. **NeregistrovaniKorisnik** je klasa preko čijih metoda se inicijalizira objekat tipa **Korisnik**. **AdminController** klasa ima za atribut objekat tipa **Admin**. **Predstava** je klasa koja ima za atribut objekat tipa **Dogadjaj**, s obzirom da može pripadati samo jednom opisu predstave. **Dogadjaj**, s druge strane, može da pripada jednoj ili više različitih objekata tipa **Predstava**. **Sjedište** klasa ima za attribute objekat tipa **Predstava** i objekat tipa **Rezervacija** (koji može biti null, u slučaju da sjedište nije rezervisano). **Predstava** objekat može pripadati 1 ili više (maksimalno do ukupnog broja sjedišta) sjedišta. **Sjedište** objekat može imati samo 1 **Predstava** objekat i samo 1 **Rezervacija** objekat (ukoliko je zauzeto, inače je null). **Rezervacija** klasa ima za atribut objekat tipa **Kupac** (koji je izvršio rezervaciju karata), objekat tipa **Pozorište** (za koju je rezervacija namijenjena) i objekat tipa **List<Sjedište>**, odnosno lista u kojoj se čuvaju rezervisana sjedišta kupca za određenu predstavu. Bez ove tri klase, klasa **Rezervacija** ne bi imala smisla. **Pozorište** klasa je klasa koja kao svoje attribute ima **List<Predstava>**, **List<Dogadjaj>**, **List<Rezervacija>** i **List<Korisnik>**.