
Dokumentacija projekta Informacioni sistem studentske službe

Sadržaj

1. UVOD.....	3
2. MODEL	3
2.1 Student.....	3
2.2 Profesor	4
2.3 Predmet	5
2.4 Ocena	5
2.5 BazaStudenata	6
2.6 BazaProfesora	7
2.7 BazaPredmeta	8
3. GUI	9
3.1 MainFrame	9
3.2 MenuBar	10
3.3 Toolbar	11
3.4 StatusBar	11
3.5 MyApp.....	12
4. VIEW.....	12
4.1 AbstractTableModelStudenti	12
4.2 StudentiJTable.....	12
4.3 ATMProfesori	13
4.4 ProfesorTable	13
4.5 ATMPredmeti.....	14
4.6 PredmetiTable.....	14
4.7 Dodaj studenta.....	14
4.8 DodajProfesora	15
4.9 DodajPredmet	16

4.10 IzmenaStudenta	16
4.11 IzmjenaProfesora	17
4.12 IzmenaPredmeta	18
4.13 ATMNepolozeni	19
4.14 NepolozeniTable	19
4.15 ATMPolozeni	20
4.16 PolozeniTable	20
4.17 ATMProfPredmeti	21
4.18 ProfPredmetiTable	21
4.19 DodajPredmetProfesoru	21
4.20 UpisOcjene	22
5. CONTROLLER	23
5.1 StudentiController	23
5.2 ProfesoriController	23
5.3 PredmetiController	24
6. ABSTRACTACTION	25
6.1 CreateEntityAction	25
6.2 EditEntityAction	25
6.3 DeleteEntityAction	25
6.4 SearchEntityAction	25

1. UVOD

Aplikacija modeluje i implementira informacioni sistem studentske službe, namijenjen za upotrebu od strane referenta službe. Za razvoj aplikacije korišćeno je Eclipse IDE 2020-09 okruženje, upotrebom Java programskog jezika (Java SE Development Kit 8 Update 271 64-bit), u skladu sa MVC šablonom.

Sadržaj projekta

- Paketi:
 - abstractAction
 - controller
 - gui
 - model
 - view
- Folderi:
 - icons – korišćene ikonice
 - lib - eksterne biblioteke korišćene u projektu
- Tekstualni fajlovi za serijalizaciju:
 - predmeti.ser
 - profesori.ser
 - studenti.ser

2. MODEL

Klase ovog paketa su:

2.1 Student

Klasa koja modeluje entitet Student u informacionom sistemu. Atributi entiteta Student predstavljeni su privatnim poljima ove klase:

- String ime - ime studenta
- String prezime – prezime studenta
- String brojIndeksa
- Date datumRodjenja
- String adresaStanovanja
- String kontaktTelefon
- String mailAdresa

- int trenutnaGodinaStudija
- String status – status studenta (budžetski/samofinansirajući student)
- ArrayList<Predmet> spisakPolozenihPredmeta
- ArrayList<Ocena> spisakPolozenihIspita
- ArrayList<Predmet> spisakNepolozenihIspita
- double prosek

Klasa sadrži dva konstruktora:

- public Student() – prazan konstruktor
- public Student (String ime, String prezime, String brojIndeksa, Date datumRodjenja, String adresaStanovanja, String kontaktTelefon, String mailAdresa, int trenutnaGodinaStudija, String status, double prosek, ArrayList<Ocena> polozeni, ArrayList<Predmet> nepolozeni) - konstruktor sa parametrima koji kreira objekat klase Student i inicijalizuje polja na proslijeđene vrijednosti

Metode klase su:

- geteri i seteri za sva polja klase
- public void removePolozeni(Ocena o) – uklanja proslijeđenu ocjenu iz liste položenih ispita
- public void removeNepolozeni(Predmet p) – uklanja proslijeđeni predmet iz liste nepoloženih ispita

2.2 Profesor

Klasa koja modeluje entitet Profesor u informacionom sistemu. Atributi entiteta Profesor predstavljeni su privatnim poljima ove klase:

- String prezime
- String ime
- Date datumRodjenja
- String adresaStanovanja
- String telefon
- String email
- String adresaKancelarije
- String brojLicneKarte
- Titula titula
- Zvanje zvanje
- List<Predmet> predmeti – lista predmeta koje profesor predaje

Klasa sadrži konstruktore:

- public Profesor(String prezime, String ime, Date datumRodjenja, String adresaStanovanja, String telefon, String email, String adresaKancelarije, String brojLicneKarte, Titula titula, Zvanje zvanje, List<Predmet> predmeti) – konstruktor sa parametrima koji kreira objekat klase Profesor i inicijalizuje polja na proslijeđene vrijednosti

Metode klase su:

- geteri i seteri za sva polja klase

2.3 Predmet

Klasa koja modeluje entitet Predmet u informacionom sistemu. Atributi entiteta Predmet predstavljeni su privatnim poljima ove klase:

- String sifraPredmeta
- String nazivPredmeta
- Semestar semestar
- int godinaStudija
- Profesor predmetniProfesor – profesor koji predaje predmet (svaki predmet predaje najviše jedan profesor)
- int espb
- List <Student> polozili – lista studenata koji su položili predmet
- List <Student> nisuPolozili – lista studenata koji nisu položili predmet

Klasa sadrži konstruktore:

- public Predmet(String sifraPredmeta, String nazivPredmeta, Semestar semestar, int godinaStudija, Profesor predmetniProfesor, int espb, List<Student> polozili, List<Student> nisuPolozili) – konstruktor sa parametrima koji kreira objekat klase Predmet i inicijalizuje polja na proslijeđene vrijednosti

Metode klase su:

- geteri i seteri za sva polja klase

2.4 Ocena

Klasa koja modeluje entitet Ocena u informacionom sistemu. Atributi entiteta Ocena predstavljeni su privatnim poljima ove klase:

- int ocena
- Date datumPolaganja

- Student student
- Predmet predmet

Klasa sadrži konstruktore:

- public Ocena(int ocena, Date datumPolaganja, Student student, Predmet predmet) – konstruktor sa parametrima koji kreira objekat klase Ocena i inicijalizuje polja na proslijeđene vrijednosti

Metode klase su:

- geteri i seteri za polja klase

2.5 BazaStudenata

Klasa predstavlja bazu podataka za entitet Student u sistemu. Implementirana je kao singleton.

Sadrži privatna polja:

- List<Student> studenti – lista studenata (objekata klase Student) koji se nalaze u sistemu
- List<String> kolone – lista stringova koji predstavljaju nazive kolona tabele za prikaz podataka o studentima iz baze

Konstruktori klase:

private BazaStudenata() – prazan konstruktor koji kreira objekat klase i postavlja polja klase na inicijalne vrijednosti

Metode klase:

- public List<Student> getStudenti() - geter za polje "studenti"
- public Student getStudent(int index) - vraća objekat klase Student iz liste studenti sa proslijeđenog indeksa
- public void setStudenti(List<Student> studenti) – seter za polje „studenti“
- public int getColumnCount() – vraća broj kolona tabele
- public String getColumnName(int index) – vraća naziv kolone koja je selektovana (indeks iz tabele)
- public Student getRow(int rowIndex) - vraća objekat klase Student za proslijeđeni indeks vrste iz tabele
- public String getValueAt(int row, int column) – vraća vrijednost iz polja tabele na presjeku proslijeđenog indeksa reda i kolone
- public void dodajStudenta(String ime, String prezime, String brojIndeksa, Date i, String adresaStanovanja, String kontaktTelefon, String mailAdresa, int

- trenutnaGodinaStudija, String status, double prosek) - kreira objekat sa proslijeđenim atributima i dodaje ga u bazu studenata
- public void izmeniStudenta (String ime, String prezime, String brojIndeksa, Date i, String adresaStanovanja, String kontaktTelefon, String mailAdresa, String j, int trenutnaGodinaStudija, String status, double prosek, List<Ocena> polozeni, List<Predmet> nepolozeni) – pronalazi studenta sa proslijeđenom vrijednošću polja brojIndeksa i ažurira njegova polja proslijeđenim vrijednostima
 - public void izbrisiStudenta(String id) - uklanja studenta sa proslijeđenim brojem indeksa iz baze studenata
 - public void dodajPredmetStudentu(Student student, String sifra) – dodaje predmet sa proslijeđenom šifrom studentu iz baze studenata
 - public void uklonilzNepolozenih(Predmet predmet) – uklanja obrisani predmet iz liste nepoloženih predmeta studenata
 - public void uklonilzPolozenih(Predmet predmet) - uklanja obrisani predmet iz liste položenih predmeta studenata
 - public void serializeStudent() -
 - public void deserijalizacijaStudenta()

2.6 BazaProfesora

Klasa predstavlja bazu podataka za entitet Profesor u sistemu. Implementirana je kao singleton.

Sadrži privatna polja:

- List<Profesor> profesori– lista profesora (objekata klase Profesor) koji se nalaze u sistemu
- List<String> kolone – lista stringova koji predstavljaju nazive kolona tabele za prikaz podataka o profesorima iz baze

Konstruktori klase:

- private BazaProfesora () – prazan konstruktor koji kreira objekat klase i postavlja polja klase na inicijalne vrijednosti

Metode klase:

- public List<Profesor> getProfesori() – getter za polje „profesori“
- public void setProfesori(List<Profesor> profesori) – setter za polje „profesori“
- public Profesor getProfesor(String blk) - vraća objekat klase Profesor iz liste profesori na osnovu proslijeđene vrijednosti broja lične karte
- public int getColumnCount() – vraća broj kolona tabele
- public String getColumnName(int index) - vraća naziv kolone koja je selektovana

- `public Profesor getRow(int rowIndex)` – vraća objekat klase Profesor za proslijeđeni indeks vrste iz tabele
- `public String getValueAt(int row, int column)` – vraća vrijednost iz polja tabele na presjeku proslijeđenog indeksa vrste i kolone
- `public void dodajProfesora(Profesor p)` – proslijeđenog profesora dodaje u listu profesora
- `public void izbrisiProfesora(String id)` – uklanja profesora iz liste profesori na osnovu proslijeđenog broja lične karte
- `public void izmijeniProfesora(String prezime, String ime, Date datRodj, String adresa, String telefon, String email, String adresaKanc, String blk, Titula titula, Zvanje zvanje, List<Predmet> predmeti)` – vrši izmjenu profesora sa proslijeđenim brojem lične karte - ažurira odgovarajuća polja koja predstavljaju parametre ove metode
- `public void dodajPredmetProfesoru(Profesor profesor, String nazivPredmeta)` - dodaje proslijeđeni predmet u listu predmeta proslijeđenog profesora
- `public void ukloniPredmet(Predmet predmet)` - uklanja obrisani predmet iz liste predmeta svakog profesora koji je taj predmet predavao
- `public void serializeProfesor()` – vrši serijalizaciju podataka (pri zatvaranju aplikacije dolazi do kreiranja datoteke (ukoliko ne postoji), tj. ažuriranja datoteke (ukoliko postoji) „profesori.ser“ čime se stanje sistema trajno čuva na hard disku)
- `public void deserijalizacijaProfesora()` – vrši deserijalizaciju podataka (pri pokretanju aplikacije dolazi do učitavanja podataka iz datoteke „profesori.ser“)
- `public void ukloniPredmetSaProf(Profesor profesor, Predmet predmet)` – vrši uklanjanje proslijeđenog predmeta iz liste predmeta proslijeđenog profesora

2.7 BazaPredmeta

Klasa predstavlja bazu podataka za entitet Predmet u sistemu. Implementirana je kao singleton.

Sadrži privatna polja:

- `List< Predmet > predmeti`– lista predmeta (objekata klase Predmet) koji se nalaze u sistemu
- `List<String> kolone` – lista stringova koji predstavljaju nazive kolona tabele za prikaz podataka o profesorima iz baze

Konstruktori klase:

- `private BazaPredmeta()` – prazan konstruktor koji kreira objekat klase i postavlja polja klase na inicijalne vrijednosti

Metode klase:

- `public List<Predmet> getPredmeti()` – getter za polje „predmeti“

- `public void setPredmeti(List<Predmet> predmeti)` – seter za polje „predmeti“
- `public Predmet getPredmetBySifra(String sifra)` - vraća objekat klase `Predmet` iz liste `predmeti` na osnovu proslijeđene vrijednosti šifre predmeta
- `public int getColumnCount()` – vraća broj kolona tabele
- `public String getColumnName(int index)` - vraća naziv kolone koja je selektovana
- `public Profesor getRow(int rowIndex)` – vraća objekat klase `Predmet` za proslijeđeni indeks vrste iz tabele
- `public String getValueAt(int row, int column)` – vraća vrijednost iz polja tabele na presjeku proslijeđenog indeksa vrste i kolone
- `public void dodajPredmet(Predmet p)` - proslijeđeni predmet dodaje u listu `predmeta`
- `public void ukloniPredmetIzTabele(Predmet predmet)` - proslijeđeni predmet uklanja iz liste `predmeta` u bazi `predmeta`
- `public void izmeniPredmet(String sifra, String naziv, int espb, int godinaStudija, Semestar semestar)` - vrši izmjenu predmeta sa proslijeđenom šifrom - ažurira odgovarajuća polja koja predstavljaju parametre ove metode
- `public Predmet findByName(String name)` - vraća predmet na osnovu proslijeđenog naziva predmeta
- `public void serializePredmet()` - vrši serijalizaciju podataka (pri zatvaranju aplikacije dolazi do kreiranja datoteke (ukoliko ne postoji), tj. ažuriranja datoteke (ukoliko postoji) „predmeti.ser“, čime se stanje sistema trajno čuva na hard disku)
- `public void deserijalizacijaPredmeta()` - vrši deserijalizaciju podataka (pri pokretanju aplikacije dolazi do učitavanja podataka iz datoteke „predmeti.ser“)

3. GUI

3.1 MainFrame

Klasa predstavlja glavni prozor koji se prikazuje pri pokretanju aplikacije. Nasljednica je `Jframe` klase. Inicijalno zauzima $\frac{3}{4}$ dužine i širine ekrana i centriran je u odnosu na ekran.

Implementirana je kao singleton.

Polja klase su:

- `MenuBar` menu
- `ToolBar` toolbar
- `StatusBar` statusbar
- `JtabbedPane` `tabbedPane` – tabovi koji prikazuju tabele s entitetima
- `ProfesoriTable` `profTable` – tabela profesora
- `StudentiJTable` `tabelaStudenata`

- PredmetiTable predTable – tabela predmeta

Konstruktor klase:

- public MainFrame() – kreira glavni prozor zadatih dimenzija, instancira menu, toolbar i status bar, te poziva metodu za kreiranje tabova. Unutar konstruktora se takođe nalazi window listener čije se metode *public void windowDeactivated(WindowEvent arg0)* i *public void windowOpened(WindowEvent arg0)* koriste za pozivanje metoda za serijalizaciju i deserijalizaciju, respektivno.

Metode klase:

- private void createTabbedPane() - kreira tabove i u njih dodaje tabele entiteta
- private JScrollPane showStudentiTable() - vraća tabelu studenata koja se prikazuje u tabu Student
- public JScrollPane showProfesoriTable() - vraća tabelu profesora koja se prikazuje u tabu Profesor
- private JScrollPane showPredmetiTable() - vraća tabelu predmeta koja se prikazuje u tabu Predmet
- public void azurirajPrikaz(String akcija, int vrednost) - vrši ažuriranje prikaza u tabeli studenata
- public void updateViewProf() - vrši ažuriranje prikaza u tabeli profesora
- public void updateViewPred() - vrši ažuriranje prikaza u tabeli predmeta
- public int selectedTab() - vraća indeks selektovanog taba
- public int selectedProf() - vraća indeks selektovanog profesora
- public int selectedPred() - vraća indeks selektovanog predmeta
- public int selectedStud() - vraća indeks selektovanog studenta
- public boolean indexCheckStud() - vraća true ukoliko je selektovan (bilo koji) student, false ako nije
- public boolean indexCheckProf() - vraća true ukoliko je selektovan (bilo koji) profesor, false ako nije
- public boolean indexCheckPred() - vraća true ukoliko je selektovan (bilo koji) predmet, false ako nije

3.2 MenuBar

Klasa predstavlja menu bar aplikacije. Menu bar posjeduje menije File (sa podmenijima new i close), Edit(podmeniji edit i delete) i Help (podmeniji help i about).

Sve stavke menija imaju odgovarajuću ikonicu, accelerator i mnemonik.

Polja klase:

- JMenu file – file meni
- JMenuItem miNew – dodavanje novog entiteta
- JMenuItem miClose – zatvaranje aplikacije
- JMenu edit – edit meni
- JMenuItem miEdit – izmjena postojećeg entiteta
- JMenuItem miDelete – brisanje postojećeg entiteta
- JMenu help – help meni
- JMenuItem miHelp – opis načina korišćenja aplikacije
- JMenuItem miAbout – prikaz i opis verzije aplikacije; biografije autora

Konstruktor klase:

- public MenuBar(final JFrame parent) – kreira menu bar i sve menije, kao i njihove stavke

3.3 Toolbar

Klasa predstavlja toolbar aplikacije. Posjeduje funkcionalnosti za otvaranje dijaloga za kreiranje entiteta, izmjenu i brisanje entiteta.

Sve stavke Toolbar-a imaju ikonice, tooltip-ove i vezane akceleratora.

Polja klase:

- public static JTextField textField – polje za unos kriterijuma pretrage
- JButton btnCreate – dugme za otvaranje dijaloga za kreiranje entiteta informacionog sistema (student, profesor, predmet)
- JButton btnEdit – dugme za otvaranje dijaloga za izmjenu označenog entiteta
- JButton btnDelete – dugme za brisanje označenog entiteta
- JButton btnSearch – dugme za pretragu tabele aktivnog taba na osnovu kriterijuma pretrage iz textField polja

Konstruktor klase:

- public Toolbar(final JFrame parent) – kreira toolbar aplikacije, instancira klase iz paketa abstractAction za postavljanje tooltip-ova, ikonica i acceleratora, te action listenera na odgovarajuću dugmad toolbar-a

3.4 StatusBar

Klasa predstavlja status bar aplikacije, koji se postavlja na dno glavnog prozora. Sadrži naziv aplikacije, trenutni datum i vrijeme.

Polja klase:

- JLabel labela – naziv aplikacije

Konstruktor klase:

- public StatusBar() – kreira status bar i inicijalizuje njegove komponente

3.5 MyApp

Klasa iz koje se vrši inicijalno pokretanje aplikacije. Sadrži samo jednu metodu: *public static void main()* koja dobavlja instancu MainFrame klase i omogućava njenu vidljivost.

4. VIEW

4.1 AbstractTableModelStudenti

Klasa predstavlja model tabele studenata. Nasljeđuje AbstractTableModel, te vrši override odgovarajućih metoda roditeljske klase.

Konstruktor klase:

- public AbstractTableModelStudenti()

Metode klase:

- public int getRowCount() - vraća broj redova tabele
- public int getColumnCount() - vraća broj kolona tabele
- public String getColumnName(int column) - vraća naziv označene kolone
- public Object getValueAt(int rowIndex, int columnIndex) - vraća vrijednost iz tabele na odgovarajućim indeksima vrste i kolone

4.2 StudentiJTable

Klasa predstavlja tabelu studenata koja se postavlja u tab Studenti. Nasljeđuje klasu JTable.

Konstruktor klase:

- public StudentiJTable() - prazan konstruktor koji omogućava selektovanje redova i kolona i postavlja odgovarajući model tabele

Metode klase:

- `public Component prepareRenderer(TableCellRenderer renderer, int row, int column)` - metoda za iscrtavanje ćelija tabele; omogućava da selektovani red ima drugačiju boju

4.3 ATMProfesori

Klasa predstavlja model tabele profesora. Nasljeđuje `AbstractTableModel`, te implementira odgovarajuće metode roditeljske klase.

Metode klase:

- `public int getRowCount()` - vraća broj redova tabele
- `public int getColumnCount()` - vraća broj kolona tabele
- `public String getColumnName(int column)` - vraća naziv označene kolone
- `public Object getValueAt(int rowIndex, int columnIndex)` - vraća vrijednost iz tabele na odgovarajućim indeksima vrste i kolone

4.4 ProfesoriTable

Klasa predstavlja tabelu profesora koja se postavlja u tab `Profesori`. Nasljeđuje klasu `JTable`. Implementirana je kao singleton.

Polja klase:

- `private static TableRowSorter<ATMProfesori> sorter`

Konstruktor klase:

- `public ProfesoriTable()` - prazan konstruktor koji omogućava selektovanje redova i kolona i postavlja odgovarajući model tabele, kao i automatsko sortiranje po svim kolonama

Metode klase:

- `public Component prepareRenderer(TableCellRenderer renderer, int row, int column)` - metoda za iscrtavanje ćelija tabele; omogućava da selektovani red ima drugačiju boju
- `public static void newFilter(String arg, int column)` – metoda koja omogućava pretragu po proslijeđenoj vrijednosti *arg* za proslijeđenu kolonu *column*

4.5 ATMPredmeti

Klasa predstavlja model tabele predmeta. Nasljeđuje AbstractTableModel, te implementira odgovarajuće metode roditeljske klase.

Metode klase:

- public int getRowCount() - vraća broj redova tabele
- public int getColumnCount() - vraća broj kolona tabele
- public String getColumnName(int column) - vraća naziv označene kolone
- public Object getValueAt(int rowIndex, int columnIndex) - vraća vrijednost iz tabele na odgovarajućim indeksima vrste i kolone

4.6 PredmetiTable

Klasa predstavlja tabelu predmeta koja se postavlja u tab Predmeti. Nasljeđuje klasu JTable. Implementirana je kao singleton.

Polja klase:

- private static TableRowSorter<ATMPredmeti> sorter

Konstruktor klase:

- public PredmetiTable() - prazan konstruktor koji omogućava selektovanje redova i kolona i postavlja odgovarajući model tabele, kao i automatsko sortiranje po svim kolonama

Metode klase:

- public Component prepareRenderer(TableCellRenderer renderer, int row, int column) - metoda za iscrtavanje ćelija tabele; omogućava da selektovani red ima drugačiju boju
- public static void newFilter(String arg) – metoda koja omogućava pretragu po proslijeđenoj vrijednosti *arg* (naziv predmeta)

4.7 Dodaj studenta

Klasa predstavlja dijalog za dodavanje novog studenta u sistem. Ukoliko referent u tabu Studenti odabere neku od opcija za dodavanje novog entiteta (dugme toolbar-a, stavka menija, odgovarajući akcelerator ili mnemonik) otvara se dijalog koji je modalan i centriran u odnosu na glavni prozor. Prikazuje se forma za unos informacija o studentu, kao i dva dugmeta za potvrdu i odustanak. Pritisak na dugme "Potvrđi" je moguće samo ako su sva polja validno popunjena.

Nakon dodavanja novog studenta u sistem, prikaz tabele sa studentima se osvježava i sadržaj se ažurira.

Polja klase:

- private JTextField txtIme
- private JTextField txtPrezime
- private JTextField txtMail
- private JTextField txtBrojTelefona
- private JTextField txtAdresaStanovanja
- private JTextField txtGodUpisa
- private JTextField txtIndeks
- private JTextField txtDatumRodjenja

Konstruktor klase:

- public DodajStudenta() – kreira dijalog za dodavanje entiteta, inicijalizuje komponente i postavlja odgovarajuće osluškivače događaja

4.8 DodajProfesora

Klasa predstavlja dijalog za dodavanje novog profesora u sistem. Ukoliko referent u tabu Profesori odabere neku od opcija za dodavanje novog entiteta (dugme toolbar-a, stavka menija, odgovarajući akcelerator ili mnemonik) otvara se dijalog koji je modalan i centriran u odnosu na glavni prozor. Prikazuje se forma za unos informacija o profesoru, kao i dva dugmeta za potvrdu i odustanak. Pritisak na dugme "Potvrdi" je moguće samo ako su sva polja validno popunjena. Nakon dodavanja novog profesora u sistem, prikaz tabele sa profesorima se osvježava i sadržaj se ažurira.

Polja klase:

- private JTextField tfPrezime - polje za unos prezimena
- private JTextField tfIme - polje za unos imena
- private JTextField tfDatRodj - polje za unos datuma rođenja
- private JTextField tfAdresa - polje za unos adrese stanovanja
- private JTextField tfEmail - polje za unos mejl adrese
- private JTextField tfBrLK - polje za unos broja lične karte
- private JComboBox<Titula> cbTitula - titula profesora
- private JComboBox<Zvanje> cbZvanje - zvanje profesora
- private JTextField tfAdresaKanc - polje za unos adrese kancelarije
- private JTextField tfTelefon - polje za unos telefona
- private JButton btnOK - dugme "Potvrdi"
- private JButton btnCancel - dugme "Odustani"

- private SimpleDateFormat sdf - za parsiranje datuma

Konstruktor klase:

- public DodajProfesora() – kreira dijalog i poziva metodu initComponents()

Metode klase:

- initComponents() - metoda koja vrši inicijalizaciju komponenti, postavlja odgovarajuće osluškivače događaja

4.9 DodajPredmet

Klasa predstavlja dijalog za dodavanje novog predmeta u sistem. Ukoliko referent u tabu Predmeti odabere neku od opcija za dodavanje novog entiteta (dugme toolbar-a, stavka menija, odgovarajući akcelerator ili mnemonik) otvara se dijalog koji je modalan i centriran u odnosu na glavni prozor. Prikazuje se forma za unos informacija o predmetu, kao i dva dugmeta za potvrdu i odustanak. Pritisak na dugme "Potvrdi" je moguće samo ako su sva polja validno popunjena. Nakon dodavanja novog predmeta u sistem, prikaz tabele sa predmetima se osvježava i sadržaj se ažurira.

Polja klase:

- private JTextField txtEspb
- private JTextField txtNaziv
- private JTextField txtSifra

Konstruktor klase:

- public DodajPredmet() - kreira dijalog za dodavanje entiteta, inicijalizuje komponente i postavlja odgovarajuće osluškivače događaja

4.10 IzmenaStudenta

Klasa predstavlja dijalog za izmjenu postojećeg studenta u sistemu. Ukoliko referent u tabu Studenti odabere neku od opcija za izmjenu entiteta (dugme toolbar-a, stavka menija, odgovarajući akcelerator ili mnemonik) otvara se dijalog koji je modalan i centriran u odnosu na glavni prozor. Dijalog sadrži tabove:

1. Informacije - sa formom za izmjenu označenog studenta i dugmadima za potvrdu i odustanak; dugme "Potvrdi" je aktivno ukoliko su svi izmijenjeni podaci u formi validni; nakon uspješne izmjene podataka, ažurira se sadržaj tabele sa studentima

2. Položeni - sa spiskom položenih predmeta označenog studenta
3. Nepoloženi - sa spiskom nepoloženih predmeta označenog studenta

Polja klase:

- public static PolozeniJTable polozeniTable
- public static NepolozeniTable nepolozeniTable
- private JTextField txtIme
- private JTextField txtPrezime
- private JTextField txtMail
- private JTextField txtBrojTelefona
- private JTextField txtAdresaStanovanja
- private JTextField txtGodUpisa
- private JTextField txtIndeks
- private JTextField txtDatumRodjenja

Konstruktor klase:

- public IzmeniStudenta(Student student) - kreira dijalog, tabove i vrši inicijalizaciju komponenti i dodavanje odgovarajućih osluškivača događaja i akcija

Metode klase:

- private JScrollPane showPolozeni(Student student) - prikaz tabele položenih predmeta proslijeđenog studenta
- public void azuriraj(String akcija, int vrednost) - ažuriranje prikaza tabele položenih predmeta
- private JScrollPane showNepolozeniTable(Student student) - prikaz tabele nepoloženih predmeta proslijeđenog studenta
- public void updateNepol() - ažuriranje prikaza tabele nepoloženih predmeta

4.11 IzmjenaProfesora

Klasa predstavlja dijalog za izmjenu postojećeg profesora u sistemu. Ukoliko referent u tabu Profesori odabere neku od opcija za izmjenu entiteta (dugme toolbar-a, stavka menija, odgovarajući akcelerator ili mnemonik) otvara se dijalog koji je modalan i centriran u odnosu na glavni prozor. Dijalog sadrži dva taba:

1. Informacije - sa formom za izmjenu označenog studenta i dugmadima za potvrdu i odustanak; dugme "Potvrdi" je aktivno ukoliko su svi izmijenjeni podaci u formi validni; nakon uspješne izmjene podataka, ažurira se sadržaj tabele sa studentima

2. Predmeti - prikazuje tabelu sa spiskom predmeta koje profesor predaje, iznad koje su dva dugmeta - "Dodaj predmet" i "Ukloni predmet".

Polja klase:

- private JTextField tfPrezime - polje za unos prezimena
- private JTextField tfIme - polje za unos imena
- private JTextField tfDatRodj - polje za unos datuma rođenja
- private JTextField tfAdresa - polje za unos adrese stanovanja
- private JTextField tfEmail - polje za unos mejl adrese
- private JTextField tfBrLK - polje za unos broja lične karte
- private JComboBox<Titula> cbTitula - titula profesora
- private JComboBox<Zvanje> cbZvanje - zvanje profesora
- private JTextField tfAdresaKanc - polje za unos adrese kancelarije
- private JTextField tfTelefon - polje za unos telefona
- private JButton btnOK - dugme "Potvrđi"
- private JButton btnCancel - dugme "Odustani"
- private SimpleDateFormat sdf - za parsiranje datuma

Konstruktor klase:

- public IzmjenaProfesora(Profesor profesor) - kreira dijalog i poziva metodu za kreiranje tabova

Metode klase:

- private void createTabbedPane(Profesor profesor) - kreira tabove i poziva metode za kreiranje forme i prikaz tabele predmeta profesora
- public JPanel showInfo(Profesor profesor) - inicijalizuje komponente za prikaz forme za izmjenu označenog profesora
- public JPanel showPredmeti(Profesor profesor) - vrši prikaz predmeta koje profesor predaje, implementira odgovarajuće akcije za gore pomenutu dugmad
- public void updatePred() - ažurira prikaz predmeta profesora

4.12 IzmjenaPredmeta

Klasa predstavlja dijalog za izmjenu postojećeg predmeta u sistemu. Ukoliko referent u tabu Predmeti odabere neku od opcija za izmjenu entiteta (dugme toolbar-a, stavka menija, odgovarajući akcelerator ili mnemonik) otvara se dijalog koji je modalan i centriran u odnosu na glavni prozor. On sadrži formu kojom je moguće izmijeniti podatke o označenom predmetu i dugmad za potvrdu i odustanak; dugme "Potvrđi" je aktivno ukoliko su svi izmijenjeni podaci u formi validni; nakon uspješne izmjene podataka, ažurira se sadržaj tabele sa predmetima. Na

dnu forme se nalazi polje koje prikazuje profesora koji predaje izabrani predmet. Pored polja nalaze se dva dugmeta - jedno za dodavanje, drugo za uklanjanje profesora sa predmeta. Jedan predmet može da predaje najviše jedan profesor.

Polja klase:

- private JTextField txtEspb
- private JTextField txtNaziv
- private JTextField txtSifra

Konstruktor klase:

- public IzmenaPredmeta(Predmet predmet) - kreira dijalog za izmjenu entiteta, inicijalizuje komponente i postavlja odgovarajuće osluškivače događaja i implementira akcije

4.13 ATMNepolozeni

Klasa predstavlja model tabele nepoloženih predmeta studenta. Nasljeđuje AbstractTableModel, te implementira odgovarajuće metode roditeljske klase.

Polja klase:

- private static List<Predmet> predmeti - lista nepoloženih predmeta studenta
- private List<String> kolone - lista stringova koji predstavljaju nazive kolona tabele

Konstruktor klase:

- public ATMNepolozeni(List<Predmet> predmeti) - kreira objekat i postavlja polja klase na inicijalne vrijednosti

Metode klase:

- public int getRowCount() - vraća broj redova tabele
- public int getColumnCount() - vraća broj kolona tabele
- public String getColumnName(int column) - vraća naziv označene kolone
- public Object getValueAt(int rowIndex, int columnIndex) - vraća vrijednost iz tabele na odgovarajućim indeksima vrste i kolone

4.14 NepolozeniTable

Klasa predstavlja tabelu nepoloženih predmeta koje student sluša.

Konstruktor klase:

- `public NepolozeniTable(Student student)` - omogućava selektovanje redova i kolona i postavlja odgovarajući model tabele

Metode klase:

- `public Component prepareRenderer(TableCellRenderer renderer, int row, int column)` metoda za iscrtavanje ćelija tabele; omogućava da selektovani red ima drugačiju boju

4.15 ATMPolozeni

Klasa predstavlja model tabele položenih predmeta studenta. Nasljeđuje `AbstractTableModel`, te implementira odgovarajuće metode roditeljske klase.

Polja klase:

- `private static List<Ocena> polozeni` – lista položenih predmeta studenta
- `private List<String> kolone` - lista stringova koji predstavljaju nazive kolona tabele

Konstruktor klase:

- `public Polozeni(ArrayList<Ocena> polozeni)` - kreira objekat i postavlja polja klase na inicijalne vrijednosti

Metode klase:

- `public int getRowCount()` - vraća broj redova tabele
- `public int getColumnCount()` - vraća broj kolona tabele
- `public String getColumnName(int column)` - vraća naziv označene kolone
- `public Object getValueAt(int rowIndex, int columnIndex)` - vraća vrijednost iz tabele na odgovarajućim indeksima vrste i kolone

4.16 PolozeniTable

Klasa predstavlja tabelu položenih predmeta koje student sluša.

Konstruktor klase:

- `public PolozeniTable(Student student)` - konstruktor koji omogućava selektovanje redova i kolona i postavlja odgovarajući model tabele

Metode klase:

- `public Component prepareRenderer(TableCellRenderer renderer, int row, int column)` metoda za iscrtavanje ćelija tabele; omogućava da selektovani red ima drugačiju boju

4.17 ATMProfPredmeti

Klasa predstavlja model tabele predmeta koje profesor predaje. Nasljeđuje `AbstractTableModel`, te implementira odgovarajuće metode roditeljske klase.

Polja klase:

- `private static List<Predmet> predmeti` - lista predmeta koje profesor predaje
- `private List<String> kolone` - lista stringova koji predstavljaju nazive kolona tabele

Konstruktor klase:

- `public ATMProfPredmeti(List<Predmet> predmeti)` - kreira objekat i postavlja polja klase na inicijalne vrijednosti

Metode klase:

- `public int getRowCount()` - vraća broj redova tabele
- `public int getColumnCount()` - vraća broj kolona tabele
- `public String getColumnName(int column)` - vraća naziv označene kolone
- `public Object getValueAt(int rowIndex, int columnIndex)` - vraća vrijednost iz tabele na odgovarajućim indeksima vrste i kolone

4.18 ProfPredmetiTable

Klasa predstavlja tabelu predmeta koje profesor predaje.

Konstruktor klase:

- `public ProfPredmetiTable(Profesor profesor)` - omogućava selektovanje redova i kolona i postavlja odgovarajući model tabele

Metode klase:

- `public Component prepareRenderer(TableCellRenderer renderer, int row, int column)` metoda za iscrtavanje ćelija tabele; omogućava da selektovani red ima drugačiju boju

4.19 DodajPredmetProfesoru

Klasa predstavlja dijalog koji se prikazuje klikom na dugme "Dodaj predmet" u tabu *Predmeti* dijaloga *Izmjena profesora*. Dijalog je centriran u odnosu na dijalog *Izmjena profesora* i sadrži listu predmeta koje nijedan profesor ne predaje. Ispod liste nalaze se dva dugmeta za potvrdu dodavanja profesora na predmet (Potvrdi), odnosno odustanak od pomenute akcije (Odustani). Nakon uspješno odrađene akcije tabela predmeta profesora se ažurira.

Polja klase:

- private DefaultListModel<String> model - model liste
- private JList<String> list - lista predmeta profesora
- private String row
- private JButton btnOK
- private JButton btnCancel

Konstruktor klase:

- public DodajPredmetProfesoru(Profesor profesor) - kreira dijalog i postavlja odgovarajuće osluškivače događaja i akcije

4.20 UpisOcjene

Klasa predstavlja dijalog koji se prikazuje klikom na dugme "Polaganje" u tabu Nepoloženi dijaloga Izmjena studenta. Dijalog je centriran u odnosu na dijalog sa spiskom nepoloženih predmeta i sadrži formu za upis ocjene na označeni nepoloženi predmet. Na dnu su dugmad za potvrdu, odnosno odustanak od upisa ocjene. Dugme za potvrdu je omogućeno samo ako su sva polja validno popunjena. Nakon uspješno izvršene akcije, predmet se briše iz tabele nepoloženih i prebacuje u tabelu položenih predmeta studenta.

Polja klase:

- private JTextField tfSifra - polje sa šifrom predmeta (nije moguće mijenjati)
- private JTextField tfNaziv - polje sa nazivom predmeta (nije moguće mijenjati)
- private JTextField tfOcjena - polje za unos ocjene (dom(ocjena) = {6, 7, 8, 9, 10})
- private JTextField tfDatum - polje za unos datuma polaganja
- private JButton btnOK
- private JButton btnCancel
- private SimpleDateFormat sdf - format za parsiranje datuma

Konstruktor klase:

- public UpisOcjene(Predmet p, Student s) - kreira dijalog , inicijalizuje komponente i postavlja odgovarajuće osluškivače događaja i akcije

5. CONTROLLER

Klase iz ovog paketa koordinišu odgovarajuće klase iz view i model paketa.

5.1 StudentiController

Klasa koja rukuje događajima koji se odnose na entitet Student. Implementirana je kao singleton klasa.

Konstruktor klase:

- private StudentiController()

Metode klase:

- public void dodajStudenta(String ime, String prezime, String brojIndeksa, Date i, String adresaStanovanja, String kontaktTelefon, String mailAdresa, int trenutnaGodinaStudija, String status, double prosek) - poziva funkciju iz baze studenata za dodavanje studenta i vrši ažuriranje prikaza tabele studenata u glavnom prozoru
- public void izmeniStudenta(String ime, String prezime, String brojIndeksa, Date i, String adresaStanovanja, String kontaktTelefon, String mailAdresa, int trenutnaGodinaStudija, String status, double prosek, ArrayList<Ocena> polozeni, ArrayList<Predmet> nepolozeni) - poziva funkciju iz baze studenata za izmjenu studenta i vrši ažuriranje prikaza tabele studenata u glavnom prozoru
- public void izbrisiStudenta(int rowSelectedIndex) - poziva funkciju iz baze studenata za brisanje studenta i vrši ažuriranje prikaza tabele studenata u glavnom prozoru
- public void dodajPredmetStudentu(Student student, String nazivPred) - poziva funkciju iz baze studenata za dodavanje predmeta studentu
- public void uklonilzNepolozenih(Predmet predmet) - poziva funkciju iz baze studenata za uklanjanje iz liste nepoloženih predmeta i vrši ažuriranje prikaza tabele studenata u glavnom prozoru
- public void uklonilzPolozenih(Predmet predmet) - poziva funkciju iz baze studenata za uklanjanje iz liste položenih predmeta i vrši ažuriranje prikaza tabele studenata u glavnom prozoru

5.2 ProfesoriController

Klasa koja rukuje događajima koji se odnose na entitet Profesor. Implementirana je kao singleton klasa.

Konstruktor klase:

- private ProfesoriController()

Metode klase:

- public void dodajProfesora(String prezime, String ime, Date datRodj, String adresa, String telefon, String email, String adresaKanc, String blk, Titula titula, Zvanje zvanje, ArrayList<Predmet> predmeti) - poziva funkciju iz baze profesora za dodavanje novog profesora i vrši ažuriranje prikaza tabele profesora u glavnom prozoru
- public void izmijeniProfesora(String prezime, String ime, Date datRodj, String adresa, String telefon, String email, String adresaKanc, String blk, Titula titula, Zvanje zvanje, ArrayList<Predmet> predmeti) - poziva funkciju iz baze profesora za izmjenu profesora i vrši ažuriranje prikaza tabele profesora u glavnom prozoru
- public void izbrisiProfesora(int rowSelectedIndex) - poziva funkciju iz baze profesora za brisanje profesora i vrši ažuriranje prikaza tabele profesora u glavnom prozoru
- public void dodajPredmetProfesoru(Profesor profesor, String nazivPred) - poziva funkciju iz baze profesora za dodavanje predmeta profesoru i vrši ažuriranje prikaza tabele predmeta koje profesor predaje
- public void ukloniSaProfesora(Predmet predmet) - poziva funkciju iz baze profesora za uklanjanje predmeta iz liste predmeta profesora koji taj predmet predaje (referencijalne zavisnosti)
- public void ukloniPredmetSaProf(Profesor profesor, Predmet predmet) - poziva funkciju iz baze profesora za uklanjanje prosljeđenog predmeta iz liste predmeta koje prosljeđeni profesor predaje

5.3 PredmetiController

Klasa koja rukuje događajima koji se odnose na entitet Predmet. Implementirana je kao singleton klasa.

Konstruktor klase:

- private PredmetiController()

Metode klase:

- public void dodajPredmet(String sifraPredmeta, String nazivPredmeta, Semestar semestar, int godinaStudija, Profesor profesor, int espb, List<Student> polozili, List<Student> nisuPolozili) - poziva funkciju iz baze predmeta za dodavanje novog predmeta i vrši ažuriranje prikaza tabele predmeta
- public void izmeniPredmet(String sifra, String naziv, int espb, int godinaStudija, Semestar semestar) - poziva funkciju iz baze predmeta za izmjenu predmeta i vrši ažuriranje prikaza tabele predmeta
- public void ukloniPredmetIzTabele(Predmet predmet) - poziva funkciju iz baze predmeta za uklanjanje predmeta i vrši ažuriranje prikaza tabele predmeta

6. ABSTRACTACTION

Instance klase ovog paketa proslijeđene su u konstruktorima dugmadi toolbara.

6.1 CreateEntityAction

Konstruktor klase postavlja tooltip, ikonicu i akcelerator za dodavanje novog entiteta u sistem. Klasa ima jednu metodu, `public void actionPerformed(ActionEvent arg)`, koja zavisno od aktivnog taba kreira objekte klase za dodavanje označenog entiteta.

6.2 EditEntityAction

Konstruktor klase postavlja tooltip, ikonicu i akcelerator za dodavanje novog entiteta u sistem. Klasa ima jednu metodu, `public void actionPerformed(ActionEvent arg)`, koja zavisno od aktivnog taba kreira objekte klase za izmjenu označenog entiteta.

6.3 DeleteEntityAction

Konstruktor klase postavlja tooltip, ikonicu i akcelerator za dodavanje novog entiteta u sistem. Klasa ima jednu metodu, `public void actionPerformed(ActionEvent arg)`, koja zavisno od aktivnog taba prikazuje prozor za potvrdu/odustanak od brisanja označenog entiteta.

6.4 SearchEntityAction

Konstruktor klase postavlja tooltip, ikonicu i akcelerator za dodavanje novog entiteta u sistem. Klasa ima jednu metodu, `public void actionPerformed(ActionEvent arg)`, koja zavisno od aktivnog taba poziva metodu za pretragu iz klase `JTable` odgovarajućeg entiteta.