

# ClassyCraft

## Uvod

ClassyCraft je alat za pravljenje i uređivanje UML klasnih dijagrama. ClassyCraft, dostupan u obliku desktop aplikacije, omogućava korisnicima da vizuelno konstruišu i razumeju kompleksne strukture sistema kroz predstavljanje klasa, njihovih atributa, operacija, kao i odnosa koji postoje između različitih objekata.

## Rečnik

**Objekat:** Objekat je integralna celina podataka koja ima svoje stanje (vrednosti atributa)

**Klasa:** Klasa predstavlja šablon iz kojeg se kreiraju objekti. Ona definiše skup atributa (podataka) i metoda (funkcija ili procedura) koje će objekti formirani iz klase imati

**Interfejs:** Definicija ponašanja koja klasa mora implementirati. Interfejs opisuje kakve akcije su moguće izvršiti s objektom, ali ne određuje kako te akcije treba izvršiti

**Atributi:** Podaci unutar klase koji čuvaju vrednosti vezane za objekte te klase

**Metode:** Operacije kojima možemo da menjamo, ispisujemo, vraćamo vrednosti atributa

**Klasni dijagram:** grafički prikaz klasa, objekata, atributa, operacija, veza između klasa

**Paket:** Organizaciona jedinica koja služi kao okvir za pravljenje i grupisanje klasnih dijagrama

**Projekat:** Centralna struktura koja služi kao osnovni okvir unutar kojeg se prave i organizuju paketi i klasni dijagrami

**Opcije za uvoz/izvoz:** Ove opcije omogućavaju korisnicima da unose (uvoze) ili iznose (izvoze) datoteke, podatke ili projekte iz jedne aplikacije (ili formata) u drugu

**IDE:** Integrisano razvojno okruženje - aplikacija koja pruža usluge programerima za kucanje programa

**Elementi dijagrama:** Klasa, interfejs, veza

**Generisanje koda:** Automatski proces kreiranja programskog koda na osnovu klasnog dijagrama

**Automatska organizacija rasporeda:** Automatski proces pomeranja svih elemenata dijagrama radi bolje preglednosti

## Konkurencija

U sledećoj tabeli prikazane su neke od funkcionalnosti koje ClassyCraft i konkurenti koji se takođe bave kreiranjem UML klasnih dijagrama implementiraju ili ne implementiraju:

Ime\Funkcija	Desktop klijent	Web verzija aplikacije	Generisanje koda	Integracija u IDE	Opcije za uvoz/izvoz	Prilagođavanje stila	Automatska organizacija rasporeda
ClassyCraft	✓	✗	✓	✗	✗	✗	✓
Visual Paradigm	✓	✓	✓	✓	✓	✓	✓
Oracle DataModeler	✓	✗	✓	✓	✓	✗	✗
Microsoft Visio	✓	✓	✗	✗	✓	✓	✓
StarUML	✓	✗	✓	✓	✓	✓	✗
Lucidchart	✗	✓	✗	✗	✓	✓	✓
Enterprise Architect	✓	✗	✓	✓	✓	✓	✓

# Korisnici

Za efikasno korišćenje ClassyCraft-a, korisnicima je potrebno osnovno razumevanje principa UML klasnih dijagrama.

ClassyCraft je razvijen za rad na računaru i zahteva minimalno jednog aktivnog korisnika za ostvarivanje svojih funkcionalnosti.

Osnovna potreba koju ovaj program ispunjava jeste pružanje korisniku sredstava za vizuelno kreiranje klasa, interfejsa, atributa, metoda i različitih veza među njima.

Ciljna grupa za ClassyCraft obuhvata širok spektar ljudi unutar industrije informacionih tehnologija, počevši od studenata, sve do iskusnih stručnjaka.

## Zahtevi

Zahtevi predstavljaju funkcionalnosti koje ClassyCraft treba da ispuni. Uvodimo tri nivoa prioriteta funkcionalnosti:

- Visok – program ne može da funkcioniše bez zadate funkcionalnosti;
- Srednji – poželjno je da program sadrži datu funkcionalnost, ali funkcioniše i bez nje;
- Nizak – program može u potpunosti da radi bez zadate funkcionalnosti;

### 1. Upravljanje strukturom projekta

#### 1.1 Kreiranje novog projekta

Korisnik treba da bude u mogućnosti da napravi novi projekat, koji služi kao glavni ram za pakete i dijagrame koji će biti definisani unutar njega.

Prioritet: Visok

Rizik: Nizak

#### 1.2 Kreiranje novog paketa

Korisnik treba da bude u mogućnosti da napravi novi paket, koji služi kao ram za klasne dijagrame koji će biti definisani unutar njega.

Prioritet: Visok

Rizik: Nizak

#### 1.3 Kreiranje i brisanje klasnog dijagrama

Korisnik treba da bude u mogućnosti da napravi novi klasni dijagram, kao i da ga obrise. Aplikacija treba da traži potvrdu pre brisanja kako bi se izbegli slučajni gubici podataka.

Prioritet: Visok

Rizik: Srednji (postoji rizik slučajnog brisanja klasnog dijagrama)

#### 1.4 Čuvanje i učitavanje projekta

Projekat koji korisnik napravi treba da bude sačuvan negde na računaru tako da kasnije može biti ponovo otvoren.

Prioritet: Srednji

Rizik: Srednji (postoji rizik da korisnikov rad propadne usled lošeg čuvanja projekta)

### 2. Manipulacija elementima dijagrama

#### 2.1 Kreiranje i brisanje klasa

Korisnik treba da bude u mogućnosti da pravi i briše klasu, kao i njene metode i atribute.

Prioritet: Visok

Rizik: Srednji (postoji rizik slučajnog brisanja klase)

#### 2.2 Kreiranje i brisanje interfejsa

Korisnik treba da bude u mogućnosti da pravi i briše interfejs, kao i njegove metode.

Prioritet: Visok

Rizik: Srednji (postoji rizik slučajnog brisanja interfejsa)

#### 2.3 Kreiranje i brisanje veza

Korisnik treba da bude u mogućnosti da pravi i briše veze između klasa i/ili interfejsa. Prilikom pravljenja veze aplikacija treba da pruži korisniku izbor tipa veze koju hoće da koristi. Tipovi veza su: veza zavisnosti, veza asocijacija, veza kompozicija, veza agregacija.

Prioritet: Visok

Rizik: Srednji (postoji rizik slučajnog brisanja veze)

#### 2.4 Uređivanje elemenata dijagrama

Korisnik treba da bude u mogućnosti da detaljno menja strukturu i sadržaj svojih klasa, interfejsa i veza.

- **Sadržaj klase:** Korisnik treba da bude u mogućnosti da menja naziv klase, dodaje, uklanja ili menja atribute i metode.

- **Sadržaj interfejsa:** Korisnik treba da bude u mogućnosti da menja naziv interfejsa i metode koje interfejs sadrži.
- **Sadržaj veze:** Korisnik treba da bude u mogućnosti da definiše i menja karakteristike veza između elemenata, uključujući tip veze (asocijacija, kompozicija, agregacija, zavisnost).

**Prioritet:** Visok

**Rizik:** Srednji (nepravilne promene mogu dovesti do netačnih dijagrama)

## 2.5 Grafički prikaz elemenata dijagrama

Svi elementi dijagrama treba da budu grafički prikazani na radnoj površini. Korisnik treba da bude u mogućnosti da preuređuje elemente dijagrama unutar radne površine po želji.

**Prioritet:** Visok

**Rizik:** Nizak

## 3. Napredne funkcionalnosti

### 3.1 Vizualizacija hijerarhije projekta

Korisnik prilikom rada treba da bude u mogućnosti da jasno vidi hijerarhijsku strukturu svog projekta u formatu stabla, postavljenog na levu stranu.

**Prioritet:** Srednji

**Rizik:** Nizak

### 3.2 Pronalaženje i obaveštenje o greškama

Aplikacija treba automatski da prepozna greške poput neuspeha u čuvanju ili učitavanju projekta, kao i grešaka u manipulaciji sa elementima dijagrama. Kada se pronađe greška, aplikacija treba da obavesti korisnika jasnom i razumljivom porukom o grešci, pružajući informacije o prirodi problema.

**Prioritet:** Srednji

**Rizik:** Srednji (pronalaženje grešaka mora biti precizno, a loše isticanje grešaka može dovesti do problema u razvoju projekta)

### 3.3 Poništiti/ponoviti funkcionalnost

Korisnik treba da bude u mogućnosti da lako poništi nedavne akcije (poništiti) ili ponovo primeni već poništene akcije (ponoviti) tokom rada, zbog efikasnijeg upravljanja promenama i ispravljanja grešaka tokom uređivanja dijagrama.

**Prioritet:** Nizak

**Rizik:** Nizak

### 3.4 Generisanje koda

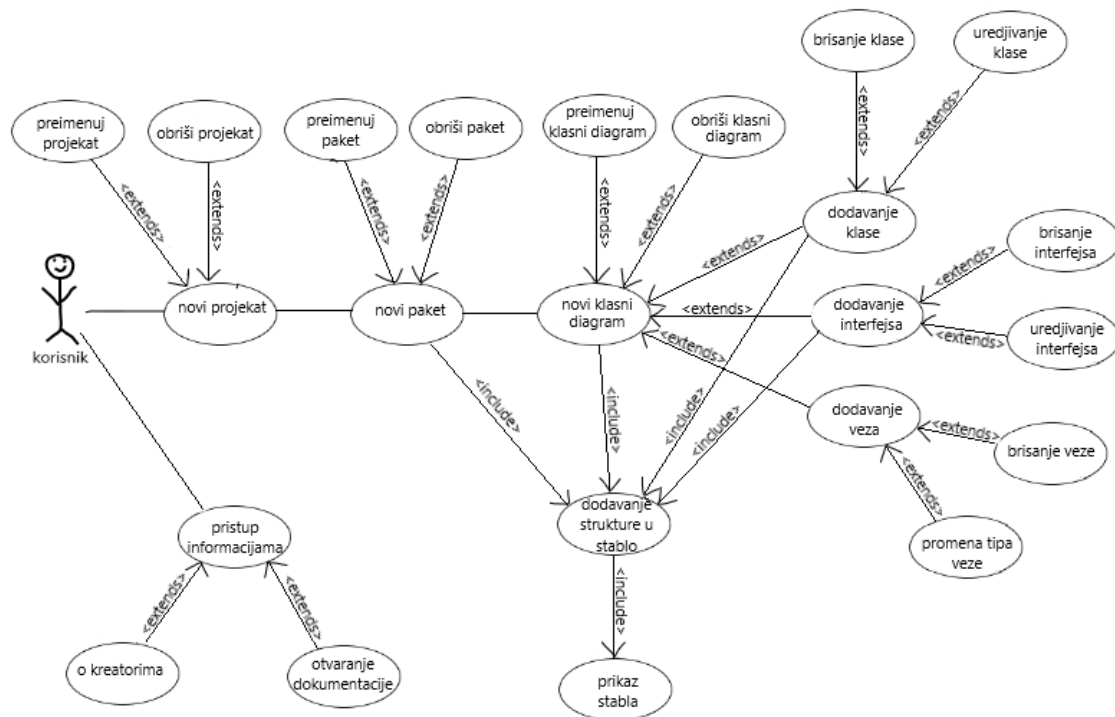
Korisnik treba da bude u mogućnosti da jednostavno pretvori svoj klasni dijagram u programski kod.

**Prioritet:** Srednji

**Rizik:** Srednji (generisanje koda mora biti precizno, a loše generisanje koda može dovesti do problema u kasnijem razvoju projekta)

# Funkcionalni dijagrami

Ispod se nalazi slika UseCase dijagrama aplikacije ClassyCraft



## UseCase

### Novi projekat

Učesnik: Korisnik

Preduslov: Korisnik prethodno mora da otvori aplikaciju ClassyCraft

Opis: Korisnik pravi novi projekat i dodeljuje mu ime

Izuzetak: Korisnik ne može napraviti nikakvu grešku prilikom ove radnje

Stanje: U stablo se automatski dodaje novonastali projekat

### Preimenuj projekat

Učesnik: Korisnik

Preduslov: Korisnik prethodno mora da ima napravljen projekat

Opis: Korisnik menja naziv projekta koji je izabrao

Izuzetak: Korisnik ne može napraviti nikakvu grešku prilikom ove radnje

Stanje: U stablu se menja ime projekta

### Obriši projekat

Učesnik: Korisnik

Preduslov: Korisnik prethodno mora da ima napravljen projekat

Opis: Korisnik briše ceo projekat i sve što se nalazi u njemu  
Izuzetak: Postoji mogućnost da korisnik slučajno obriše projekat koji nije želeo  
Stanje: Briše se sav rad unutar projekta kao i sam projekat iz stabla

## Novi paket

Učesnik: Korisnik  
Preduslov: Korisnik prethodno mora da ima napravljen projekat  
Opis: Korisnik pravi novi paket i dodeljuje mu ime  
Izuzetak: Korisnik ne može napraviti nikakvu grešku prilikom ove radnje  
Stanje: U stablo se automatski dodaje novonastali paket

## Preimenuj paket

Učesnik: Korisnik  
Preduslov: Korisnik prethodno mora da ima napravljen paket  
Opis: Korisnik menja naziv paketa koji je izabrao  
Izuzetak: Korisnik ne može napraviti nikakvu grešku prilikom ove radnje  
Stanje: U stablu se menja ime paketa

## Obriši paket

Učesnik: Korisnik  
Preduslov: Korisnik prethodno mora da ima napravljen paket  
Opis: Korisnik briše ceo paket i sve što se nalazi u njemu  
Izuzetak: Postoji mogućnost da korisnik slučajno obriše paket koji nije želeo  
Stanje: Briše se sav rad unutar paketa kao i sam paket iz stabla

## Novi klasni dijagram

Učesnik: Korisnik  
Preduslov: Korisnik prethodno mora da ima napravljen paket  
Opis: Korisnik pravi novi klasni dijagram i dodeljuje mu ime  
Izuzetak: Korisnik ne može napraviti nikakvu grešku prilikom ove radnje  
Stanje: Prikazuje se prazna radna površina na koju korisnik može dodavati nove elemente i povezivati ih, u stablo se automatski dodaje novonastali klasni dijagram

## Preimenuj klasni dijagram

Učesnik: Korisnik  
Preduslov: Korisnik prethodno mora da ima napravljen klasni dijagram  
Opis: Korisnik menja naziv klasnog dijagrama koji je izabrao  
Izuzetak: Korisnik ne može napraviti nikakvu grešku prilikom ove radnje  
Stanje: U stablu se menja ime klasnog dijagrama

## Obriši klasni dijagram

Učesnik: Korisnik  
Preduslov: Korisnik prethodno mora da ima napravljen klasni dijagram  
Opis: Korisnik briše ceo klasni dijagram i sve elemente koji se nalaze u njemu  
Izuzetak: Postoji mogućnost da korisnik slučajno obriše klasni dijagram koji nije želeo  
Stanje: Briše se sav rad unutar klasnog dijagrama kao i sam klasni dijagram iz stabla

## Dodavanje klase

Učesnik: Korisnik  
Preduslov: Korisnik prethodno mora da ima napravljen klasni dijagram  
Opis: Na radnoj površini se prikazuje nova prazna klasa  
Izuzetak: Korisnik ne može napraviti nikakvu grešku prilikom ove radnje  
Stanje: Klasni dijagram dobija novu praznu klasu koju je korisnik napravio i u stablo se dodaje ista

## Uredjivanje klase

Učesnik: Korisnik  
Preduslov: Korisnik prethodno mora da ima napravljenu klasu

Opis: Dolazi do izmene atributa, metoda ili naziva izabrane klase

Izuzetak: Postoji mogućnost da korisnik pogrešno izmeni neka svojstva atributa ili metoda koja nije želeo da menja

Stanje: Izmenjeni atributi ili metode dobijaju novo svojstvo

## Brisanje klase

Učesnik: Korisnik

Preduslov: Korisnik prethodno mora da ima napravljenu klasu

Opis: Korisnik briše izabranu klasu i sve atribute i metode koje se nalaze u njoj

Izuzetak: Postoji mogućnost da korisnik slučajno obriše klasu koju nije želeo

Stanje: Briše se sav rad unutar klase, vizuelni prikaz na radnoj površini kao i sama klasa iz stabla

## Dodavanje interfejsa

Učesnik: Korisnik

Preduslov: Korisnik prethodno mora da ima napravljen klasni dijagram

Opis: Na radnoj površini se prikazuje novi prazni interfejs

Izuzetak: Korisnik ne može napraviti nikakvu grešku prilikom ove radnje

Stanje: Klasni dijagram dobija novi prazni interfejs koji je korisnik napravio i u stablo se dodaje isti

## Uredjivanje interfejsa

Učesnik: Korisnik

Preduslov: Korisnik prethodno mora da ima napravljen interfejs

Opis: Dolazi do izmene metoda ili naziva izabranog interfejsa

Izuzetak: Postoji mogućnost da korisnik pogrešno izmeni neka svojstva metoda koja nije želeo da menja

Stanje: Izmenjene metode dobijaju novo svojstvo

## Brisanje interfejsa

Učesnik: Korisnik

Preduslov: Korisnik prethodno mora da ima napravljen interfejs

Opis: Korisnik briše izabrani interfejs i sve metode koje se nalazi u njemu

Izuzetak: Postoji mogućnost da korisnik slučajno obriše interfejs koji nije želeo

Stanje: Briše se sav rad unutar interfejsa, vizuelni prikaz na radnoj površini kao i sam interfejs iz stabla

## Dodavanje veze

Učesnik: Korisnik

Preduslov: Korisnik prethodno mora da ima bar dve klase, odnosno interfejsa, na radnoj površini

Opis: Na radnoj površini se prikazuje nova izabrana veza (asocijacija, kompozicija, agregacija, zavisnost) izmedju klasa ili interfejsa

Izuzetak: Postoji mogućnost da korisnik slučajno poveže klase, odnosno interfejse, koje nije želeo

Stanje: Klasni dijagram dobija novu vezu koju je korisnik napravio

## Promena tipa veze

Učesnik: Korisnik

Preduslov: Korisnik prethodno mora da ima napravljenu vezu

Opis: Dolazi do izmene tipa izabrane veze. Ona može biti tipa: asocijacija, kompozicija, agregacija, zavisnosti

Izuzetak: Postoji mogućnost da korisnik slučajno izmeni tip veze u neki koji nije želeo

Stanje: Izabrana veza dobija novi tip veze

## Brisanje veze

Učesnik: Korisnik

Preduslov: Korisnik prethodno mora da ima napravljenu vezu

Opis: Korisnik briše izabranu vezu

Izuzetak: Postoji mogućnost da korisnik slučajno obriše vezu koju nije želeo

Stanje: Klasni dijagram gubi posmatranu vezu

## Dodavanje strukture u stablo

Učesnik: Korisnik

Preduslov: Korisnik prethodno mora da ima napravljen klasni dijagram

Opis: Svaki novi element se automatski dodaje u stablo  
Izuzetak: Korisnik ne može napraviti nikakvu grešku prilikom ove radnje  
Stanje: Novi element se ubacuje u stablo

## Prikaz stabla

Učesnik: Korisnik  
Preduslov: Korisnik prethodno mora da ima napravljen klasni dijagram  
Opis: Stablo sa komponentima se vizuelno prikazuje na odredjenom delu ekrana  
Izuzetak: Korisnik ne može napraviti nikakvu grešku prilikom ove radnje  
Stanje: Korisniku je vidljivo trenutno stanje elemenata sortiranih u obliku stabla

## Pristup informacijama

Učesnik: Korisnik  
Preduslov: Korisnik prethodno mora da otvori aplikaciju ClassyCraft  
Opis: Korisnik pomoću menija otvara listu sa svim informacijama vezanim za aplikaciju ClassyCraft  
Izuzetak: Korisnik ne može napraviti nikakvu grešku prilikom ove radnje  
Stanje: Prikazuje se lista sa svim informacijama vezanim za aplikaciju ClassyCraft

## O kreatorima

Učesnik: Korisnik  
Preduslov: Korisnik prethodno mora da otvori meni sa informacijama o aplikaciji ClassyCraft  
Opis: Korisnik otvara nov prozor u kojem se nalaze informacije o kreatorima aplikacije ClassyCraft  
Izuzetak: Korisnik ne može napraviti nikakvu grešku prilikom ove radnje  
Stanje: Aplikacija otvara nov prozor sa informacijama o kreatorima aplikacije ClassyCraft

## Otvaranje dokumentacije

Učesnik: Korisnik  
Preduslov: Korisnik prethodno mora da otvori aplikaciju ClassyCraft  
Opis: Korisnik otvara nov prozor u kojem se nalazi dokumentacija aplikacije ClassyCraft  
Izuzetak: Korisnik ne može napraviti nikakvu grešku prilikom ove radnje  
Stanje: Aplikacija otvara nov prozor sa dokumentacijom aplikacije ClassyCraft