

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 294

**APLIKACIJA ZA POHRANU BIOINFORMATIČKIH
PODATAKA U SUSTAVU POSTGRESQL**

Daniel Ranogajec

Zagreb, lipanj 2021.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 294

**APLIKACIJA ZA POHRANU BIOINFORMATIČKIH
PODATAKA U SUSTAVU POSTGRESQL**

Daniel Ranogajec

Zagreb, lipanj 2021.

ZAVRŠNI ZADATAK br. 294

Pristupnik: **Daniel Ranogajec (0036514869)**
Studij: Elektrotehnika i informacijska tehnologija i Računarstvo
Modul: Računarstvo
Mentor: doc. dr. sc. Krešimir Križanović

Zadatak: **Aplikacija za pohranu bioinformatičkih podataka u sustavu PostgreSQL**

Opis zadatka:

S velikim napretkom tehnologija sekvenciranja i bioinformatike općenito u zadnjih dvadeset godina, količina i raznolikost proizvedenih podataka višestruko je narasla. Iako sekvence proizvedene uređajima za sekvenciranje nisu pogodne za pohranu u relacijskoj bazi podataka mnogi od izvedenih podataka jesu. Potrebno je osmisliti relacijski model za pohranu različitih bioinformatičkih podataka te ga implementirati koristeći PostgreSQL bazu podataka. Relacijske podatke povezati sa sekvencama koje treba pohranjivati kao FASTA/FASTQ datoteke na disku. Napisati aplikaciju koja će služiti za upravljanje bazom podataka, te omogućiti unos i pregled podataka. Podatke za testiranje baze i aplikacije preuzeti sa stranica NCBI. Rješenje treba biti napisano kao desktop ili web aplikacija i treba raditi na operacijskom sustavu Linux. Programski kod je potrebno komentirati i pri pisanju pratiti neki od standardnih stilova. Napisati iscrpne upute za instalaciju i izvođenje. Kompletно programsko rješenje postaviti na GitHub pod jednom od OSI-odobrenih licenci.

Rok za predaju rada: 11. lipnja 2021.

SADRŽAJ

1. Uvod	1
2. Aplikacija za pohranu bioinformatičkih podataka	2
2.1. Opis podataka	2
2.2. Opis zadatka	3
2.3. Implementacija zadatka	3
3. Baza podataka	4
3.1. ER Model	4
3.1.1. Opis ER Modela	4
3.1.2. Popis entiteta i njihovih atributa	5
3.1.3. Popis veza	5
3.2. Relacijski model	7
3.3. Kreiranje baze podataka	12
4. Programsko rješenje	13
4.1. Prvo pokretanje i kreiranje baze podataka	13
4.2. Pretraživanje organizama	15
4.3. Dodavanje novog genoma	17
4.4. Dodavanje podataka o genima	19
4.5. Rekurzivno pretraživanje po taksonomskom stablu	21
4.6. Testiranje podataka	23
5. Korišteni alati	24
6. Zaključak	25
Literatura	26
Popis slika	27

1. Uvod

Velik broj znanstvenika koji se bave proučavanjem organizama, njihovih genoma i gena koriste NCBI-evu (Nacionalni centar za biotehnološke informacije) web stranicu. Sam GUI stranice je izrazito kompleksan i korisnici koji prvi put otvaraju stranicu neće se lako snaći.

Zadatak završnog rada bio je napraviti desktop aplikaciju koja služi za pregled podataka organizama, dohvaćanje sekvenci genoma te spremanje istih.

Za tehničku realizaciju aplikacije potrebna je baza podataka u koju će biti spremljeni svi podaci dohvaćeni iz NCBI-eve biblioteke, a sama aplikacija će kroz svoje korisničko sučelje omogućiti korisniku lagano pretraživanje podataka te preuzimanje i spremanje sekvenci genoma i gena. Za realizaciju baze podataka korišten je PostgreSQL [6], a sama aplikacija realizirana je u programskom jeziku Java uz pomoću biblioteke Swing koja omogućuje izradu grafičkog korisničkog sučelja.

2. Aplikacija za pohranu bioinformatičkih podataka

2.1. Opis podataka

Prema Hrvatskoj enciklopediji [4], organizam je svako živo biće. Za funkcioniranje organizama nužne su makromolekule nukleinskih kiselina, proteina i ugljikohidrata. Dvije osnovne nukleinske kiseline su DNA (deoksiribonukleinska kiselina) i RNA (ribonukleinska kiselina). DNA služi kao spremište instrukcija za razvoj i funkcioniranje svih organizama. RNA služi kao prenositelj genetičke informacije u sintezi proteina. Gen je osnovna jedinica nasljeđivanja u svim živućim organizmima.

Gen je dio DNA ili RNA koji kodira informaciju za proizvodnju proteina ili funkcionalnih RNA lanaca. Sve nukleinske kiseline su polimeri nukleotida. Nukleotidi u DNA se sastoje od nukleinskih baza (adenin A, citozin C, gvanin G i timin T) i okosnice koja je građena od naizmjeničnog niza šećera i fosfatne skupine. DNA molekulu čine dva lanca koja se sastoje od nukleotida, a ti su lanci međusobno povezani na način da se adenin spaja s timinom, a gvanin sa citozinom. DNA molekula sastoji se od kromosoma. RNA molekula je također građena od nukleotida, ali za razliku od DNA ima samo jedan lanac te umjesto nukleinske baze timin RNA ima uracil (oznaka U). Geni mogu biti dugi od nekoliko desetaka pa do nekoliko milijuna nukleotida.

Genom je sveukupna nasljedna informacija nekog organizma. Genom sadrži sve gene, a i sve druge dijelove DNA i RNA bez obzira bila njihova funkcija poznata ili ne. Pošto DNA ili RNA lanci sadrže sve genetske informacije, postoje velike koristi od određivanja njihovog slijeda budući da se tako mogu dobiti informacije o mnogim nasljednim svojstvima. Sekvenciranje je postupak određivanja poretka nukleinskih baza unutar DNA ili RNA lanca. Rezultati sekvenciranja se koriste za utvrđivanje genoma, a te iste podatke potrebno je spremati na disku zbog velike količine memorije koje zauzimaju. Sekvence se spremaju u FASTA tekstualnom formatu koji služi za prikazivanje slijedova nukleinskih kiselina ili proteina pri čemu je svaki nukleotid ili aminokiselina

prikazan jednim slovom (A, T, G, C ili U) [5].

2.2. Opis zadatka

Potrebno je izgraditi bazu podataka u koju će se spremati podaci o organizmima, genomima i genima. Potom treba napisati program za popunjavanje iste baze podacima. Nakon što je baza podataka popunjena, potrebno je napraviti aplikaciju koja će ju koristiti. Aplikacija treba omogućiti korisniku pregled svih organizama iz baze podataka, dodavanje sekvenci referentnih genoma organizmima, preuzimanje istih ako su već spremljeni na disku, dodavanje gena i pregled gena ako su već pohranjeni u bazi podataka.

Aplikacija treba omogućiti:

- Prijavu u bazu podataka
- Popunjavanje baze podataka
- Pretraživanje organizama iz baze podataka
- Spremanje referentnih genoma na disku
- Preuzimanje referentnih genoma sa diska
- Pohranjivanje gena u bazu podataka
- Prikaze gena

2.3. Implementacija zadatka

Prva stvar koju je potrebno napraviti je konstruirati ER model baze podataka. Jednom kad se napravi ER model, lagano se može pretvoriti u relacijski model iz kojeg se mogu napraviti tablice baze podataka. Slijedeći korak je popunjavanje istih. Za popunjavanje tablica potrebno je preuzeti dostupne podatke o organizmima iz NCBI-eve biblioteke, napraviti program koji će parsirati iste i automatski ih pohranjivati u bazu podataka. Idući korak je izrada same aplikacije. Ona bi trebala imati tražilicu koja omogućuje pretraživati organizme, a kada korisnik odabere organizam, treba ga preusmjeriti na novi prozor u kojem može vidjeti podatke o organizmu. Aplikacija također treba upravljati sa genima i genomima.

3. Baza podataka

Aplikacija se temelji na bazi podataka u kojoj su spremljeni taksonomski podaci dohvaćeni iz NCBI-eve (Nacionalni centar za biotehnološke informacije) baze podataka za biotehnologiju.

Prema definiciji M. Vetter-a, baza podataka je skup podataka pohranjenih i organiziranih tako da mogu zadovoljiti zahtjeve korisnika. Podatci su pohranjeni u bazu podataka kao entiteti, a svi entiteti posjeduju neke attribute ili svojstva koji ih kategoriziraju [2].

3.1. ER Model

ER model baze podataka je post-relacijski model koji omogućuje eksplicitni prikaz veza koje u sebi sadrže važne semantičke informacije. Kako bi se konstruirao ER model, moraju se prvo definirati entiteti i njihovi atributi te veze između njih.

3.1.1. Opis ER Modela

Za svaki organizam potrebno je spremiti neke osnovne informacije. Sa NCBI-eve stranice moguće je preuzeti podatke o svim poznatim organizmima koji su pohranjeni u 4 .dmp datoteke: nodes.dmp, names.dmp, division.dmp i gencode.dmp. Te datoteke biti će iskorištene kao entiteti za ER model. **Nodes** entitet sadrži strane ključeve na **gencode** i **division** entitete. Pošto jedan organizam može imati više imena (znanstveno ime, common ime, sinonim...), entitet **names** mora imati *tax_id* od **nodes**-a kao strani ključ. Budući da ćemo za organizme spremati i informacije o genima, kao i sekvence genoma, pravimo i entitete vezane uz iste. Entitet **genes** imati će atribut *tax_id* kao strani ključ od entiteta **nodes** pošto jedan organizam može imati i više gena. Entitet **reference_genomes** biti će povezan sa entitetom **nodes** 1:1 vezom te će imati *tax_id* kao strani ključ. Pošto organizam može imati više referentnih genoma, a njihove sekvence se preuzimaju u jednoj datoteci, genomi se međusobno odvajaju entitetom **headers**.

3.1.2. Popis entiteta i njihovih atributa

NODES - tax_id, parent_tax_id, rank, embl_code, inherited_div_flag, inherited_GC_flag, inherited_MGC_flag, genbank_hidden_flag, hidden_subtree_root_flag, comments

NAMES - name_id, name_txt, unique_name, name_class

DIVISION - division_id, division_cde, division_name, comments

GENCODE - genetic_code_id, abbreviation, name, cde, starts

REFERENCE_GENOMES - genome_id, file_location

HEADERS - header_id, header

GENES - gene_id, gene_description, organism, genomic_context, annotation, other_aliases, other_designations, symbol

Napomena: primarni ključevi su podvučeni.

3.1.3. Popis veza

nameOfNode - tax_id, name_id

divOfNode - tax_id, division_id

gencodeOfNode - tax_id, genetic_code_id

mitGencodeOfNode - tax_id, genetic_code_id

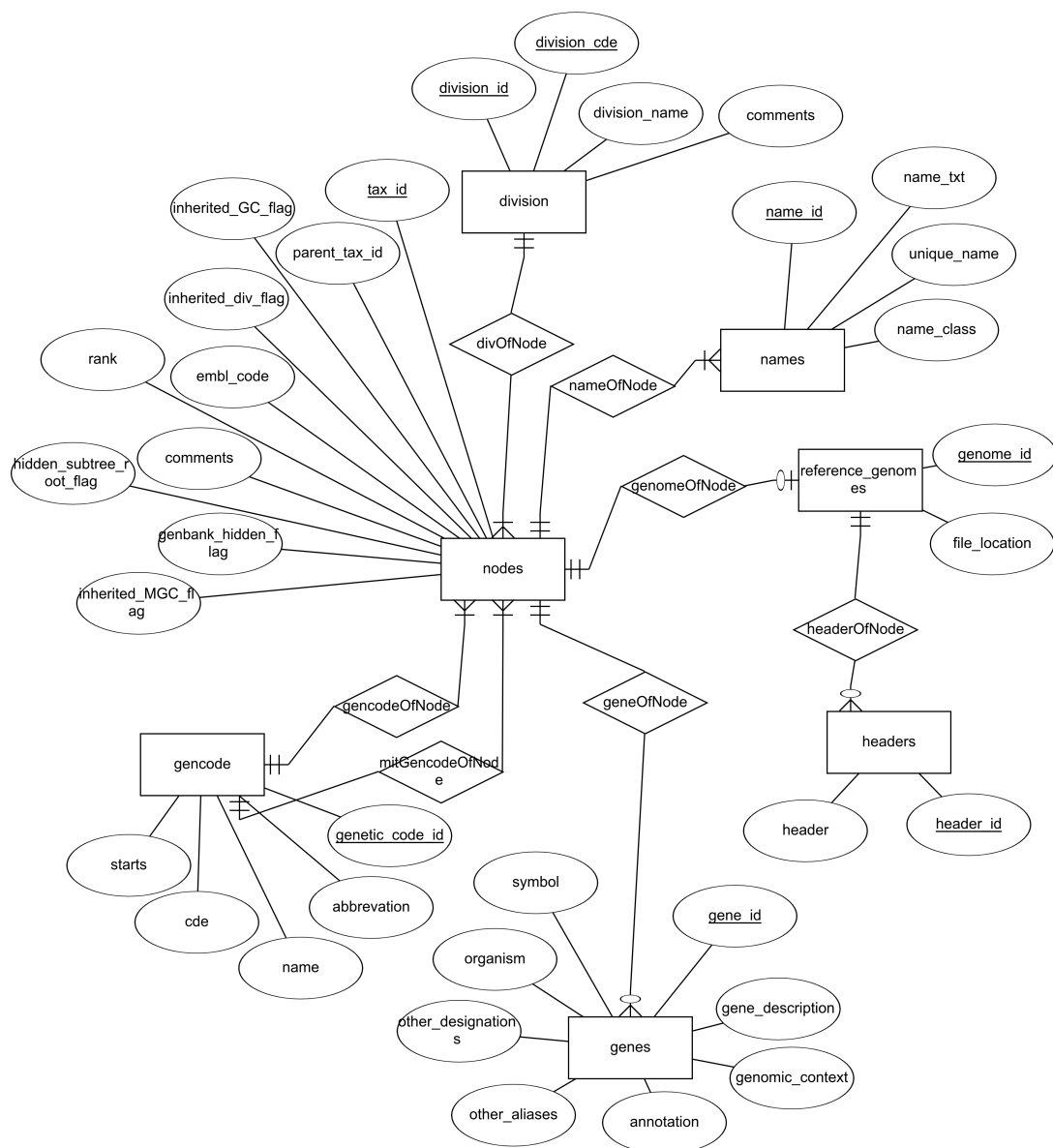
genomeOfNode - tax_id, genome_id

headerOfNode - tax_id, header_id

geneOfNode - tax_id, gene_id

Napomena: podvučeni ključevi su primarni ključevi veza.

Na slici 3.1 prikazan je opisani ER model baze podataka.



Slika 3.1: ER model

3.2. Relacijski model

Jednom kada je napravljen ER model, može ga se lagano pretvoriti u relacijski model koji će se koristiti za kreiranje tablica. Prilikom pretvorbe ER modela u relacijski model, gube se veze te njihovi primarni ključevi postaju strani ključevi kod ostalih entititeta te veze.

ATRIBUT	TIP PODATKA	OPIS
tax_id	INT NOT NULL	ID iz GenBank taksonomije
parent_tax_id	INT NOT NULL	ID roditelja iz GenBank taksonomije
rank	VARCHAR NOT NULL	Sistematska klasifikacija (domena, carstvo, koljeno...)
embl_code	VARCHAR	Lokus ime
division_id	INT NOT NULL	ID taksonomske podjele
inherited_div_flag	INT NOT NULL	1 ako čvor nasljeđuje podjelu od roditelja, inače 0
genetic_code_id	INT NOT NULL	ID taksonomskog genetskog koda
inherited_GC_flag	INT NOT NULL	1 ako čvor nasljeđuje genetski kod od roditelja, inače 0
mitochondrial_genetic_code_id	INT NOT NULL	ID mitohondrijskog genetskog koda
inherited_MGC_flag	INT NOT NULL	1 ako čvor nasljeđuje mitohondrijski genetski kod od roditelja, inače 0
genbank_hidden_flag	INT NOT NULL	1 ako je ime potisnuto u GenBankovom rodu entrya, inače 0
hidden_subtree_root_flag	INT NOT NULL	1 ako ovo podstablo nema sekvencijske podatke, 0 inače
comments	VARCHAR	Komentar

Tablica 3.1: Nodes

ATRIBUT	TIP PODATKA	OPIS
name_id	SERIAL	Serijski broj u tablici
name_txt	VARCHAR NOT NULL	Ime organizma
unique_name	VARCHAR	Jedinstvena varijanta imena u slučaju da ime nije jedinstveno
name_class	VARCHAR NOT NULL	Sinonim, često ime...
tax_id	INT NOT NULL	ID iz GenBank taksonomije

Tablica 3.2: Names

ATRIBUT	TIP PODATKA	OPIS
division_id	INT NOT NULL	ID taksonomske podjele
division_cde	VARCHAR(3) NOT NULL	GenBankov kod podjele
division_name	VARCHAR NOT NULL	Ime podjele
comments	VARCHAR	Komentar

Tablica 3.3: Division

ATRIBUT	TIP PODATKA	OPIS
genetic_code_id	INT NOT NULL	ID taksonomskog genetskog koda
abbreviation	VARCHAR	Skraćenica imena genetskog koda
name	VARCHAR NOT NULL	Ime genetskog koda
cde	VARCHAR	Prijevodna tablica za ovaj genetski kod
starts	VARCHAR	Početni kodoni za ovaj genetski kod

Tablica 3.4: Gencode

ATRIBUT	TIP PODATKA	OPIS
genome_id	SERIAL	Serijski broj u tablici
tax_id	INT NOT NULL	ID iz GenBank taksonomije
file_location	VARCHAR	Lokacija sekvence genoma na disku

Tablica 3.5: Reference_genomes

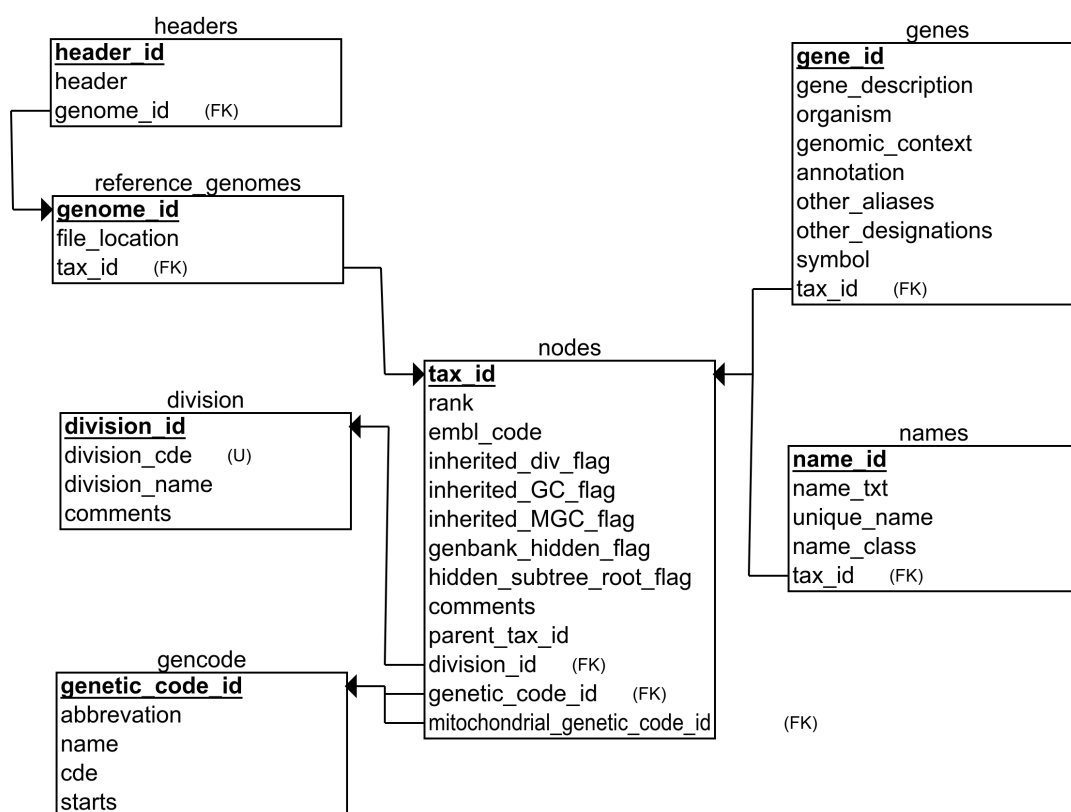
ATRIBUT	TIP PODATKA	OPIS
header_id	SERIAL	Serijski broj u tablici
header	VARCHAR NOT NULL	ID Zaglavlje sekvence
tax_id	INT NOT NULL	ID iz GenBank taksonomije

Tablica 3.6: Headers

ATRIBUT	TIP PODATKA	OPIS
tax_id	INT NOT NULL	ID iz GenBank taksonomije
symbol	VARCHAR NOT NULL	Simbol gena
gene_id	INT NOT NULL	Jedinstveni identifikator za gen
gene_description	VARCHAR	Deskriptivno ime za gen
organism	VARCHAR NOT NULL	Organizam u kojem je gen
genomic_context	VARCHAR	Lokacija gena u organizmu
annotation	VARCHAR	Anotacija gena
other_aliases	VARCHAR	Set alternativnih imena koji su pridodjeljeni genu
other_designations	VARCHAR	Set alternativnih opisa koji su pridodjeljeni genu

Tablica 3.7: Genes

Relacijski model baze podataka prikazan je na slici 3.2.



Slika 3.2: Relacijski model

Baza podataka sada se sastoji se od sedam tablica: names, division, gencode, nodes, referenceq_genomes, headers i genes. Tablica names sadrži informacije o imenima pojedinih organizama. Tablica division služi za organizaciju taksonomske podjele. Tablica gencode služi za dohvaćanje genetskog koda pojedinog organizma. Tablica nodes služi kao čvor koji povezuje te tri tablice. Tablica reference_genomes pohranjuje tax_id koji se može povezati sa tablicom names te lokaciju FASTA datoteke u kojoj je zapisana sekvenca referentnog genoma. Sve FASTA datoteke spremaju se lokalno na disku, a pomoću njihove lokacije koja je zapisana u reference_genomes tablici ih se može pregledavati i preuzimati po potrebi. Tablica headers je pomoćna tablica za tablicu reference_genomes koja pohranjuje sva zaglavlja sekvenci za pojedini referentni genom. Konačno, tablica genes pohranjuje podatke o genima pojedinog organizma. Taksonomija organizama je hijerarhijski strukturirana u odnosu roditelj dijete. Taksonomsko stablo povezuje se tablicom nodes. Svaki čvor iz tablice osim svog tax_id-a koji je ID iz GenBankove taksonomije ima i ID roditelja pa se tako svi čvorovi mogu spojiti u jedno veliko taksonomsko stablo.

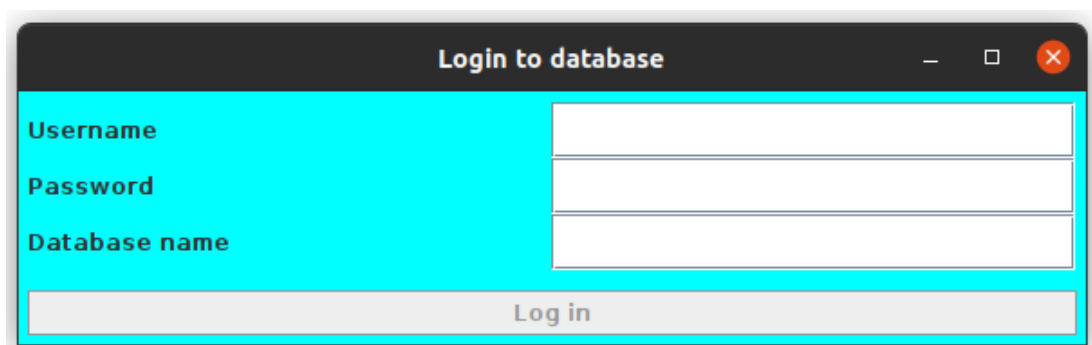
3.3. Kreiranje baze podataka

Ukoliko se aplikacija pokreće prvi puta, korisniku će se prikazati prozor u kojem treba upisati podatke o novoj Postgres bazi podataka (username, lozinku i ime baze podataka). Nakon toga pokreću se `Seed.java` te `SeedReferenceGenomes.java` programi koji idućih nekoliko minuta pune istu bazu podacima. `Seed.java` prvo pokušava uspostaviti konekciju sa bazom podataka, a ako ne uspije javlja korisniku da je upisao krive podatke. Nakon što je uspostavljena konekcija, kreiraju se tablice u bazi podataka. Tablice `division`, `gencode`, `names` i `nodes` se popunjavaju tako što se čitaju `.dmp` datoteke koje se nalaze u `src/resources/taxdump/` direktoriju, zatim se pročitani podaci parsiraju u listu `Stringova` koji se zatim pohranjuju u bazu podataka. Tablica `reference_genomes` i tablica `headers` popuniti će se ako u direktoriju `src/resources/reference_genomes` postoje `FASTA` datoteke sa sekvencama genoma, a ako je direktorij prazan, te sekvence se uvijek mogu dodati ručno ili pokretanjem `SeedReferenceGenomes.java` programa kasnije. Tablica `genes` puni se ručno unutar aplikacije učitavanjem tekstualne datoteke sa sažetkom o poznatim genima pojedinog organizma.

4. Programsko rješenje

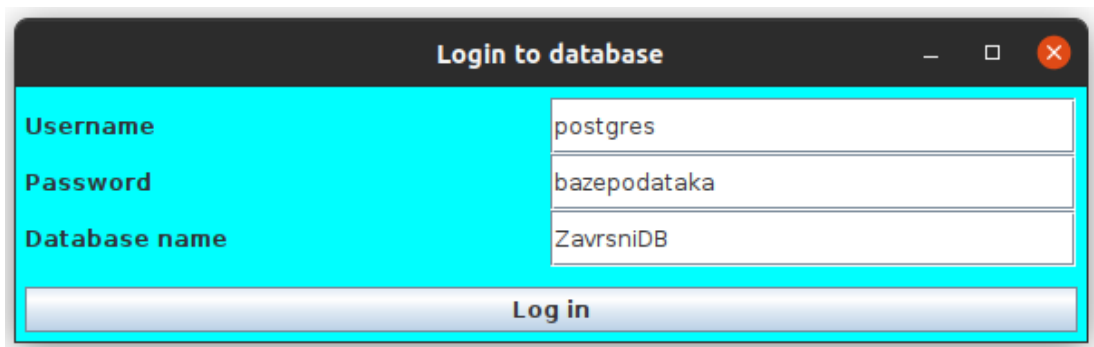
4.1. Prvo pokretanje i kreiranje baze podataka

Korisnicima koji prvi puta koriste aplikaciju prilikom pokretanja otvara se prozor prikazan na slici 4.1. Prije upisivanja podataka korisnik prvo treba napraviti novu PostgreSQL bazu podataka, a privilegije koje on ima kod korištenja iste moraju biti *SELECT*, *INSERT* i *CREATE*. U tražena polja sada treba upisati ime vlasnika baze, lozinku te naziv napravljene baze podataka.



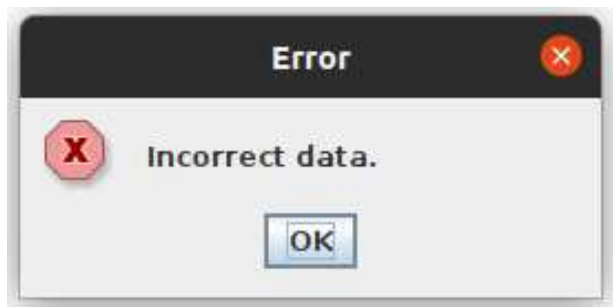
Slika 4.1: Login prozor

Kada korisnik upiše svoje podatke, pritiskom na tipku "Log in" aplikacija se pokušava spojiti na bazu podataka.



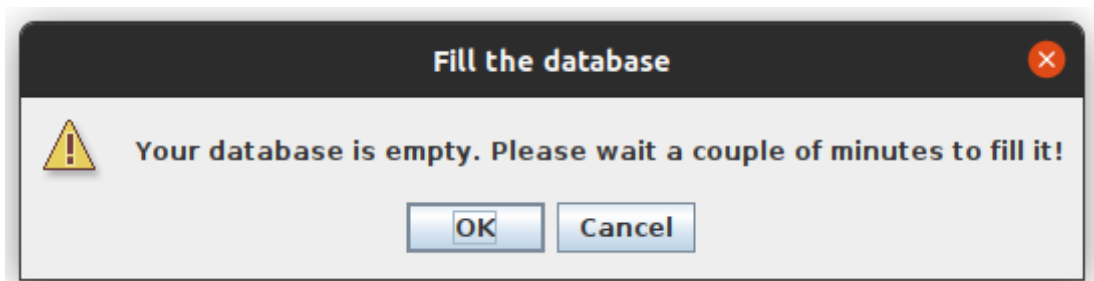
Slika 4.2: Ispravno popunjen login prozor

U slučaju da se aplikacija ne može spojiti na bazu podataka, to najčešće znači da je korisnik upisao krive podatke za prijavu te se pojavljuje skočni prozor koji javlja grešku.



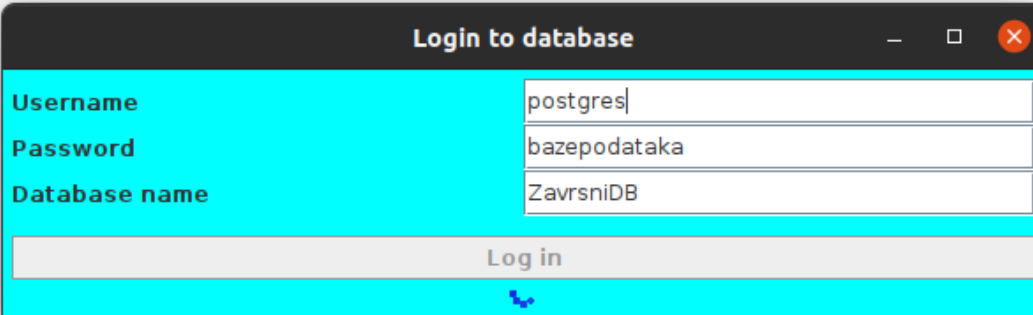
Slika 4.3: Skočni prozor nastao zbog neispravno upisanih podataka

Ako se aplikacija uspješno spoji na bazu podataka, u slučaju da je ona prazna, korisniku se otvara skočni prozor koji mu javlja da će popunjavanje tablica potrajati nekoliko minuta. Ako je baza podataka već popunjena, aplikacija odmah prelazi na idući prozor.



Slika 4.4: Skočni prozor koji obavještava da će krenuti popunjavanje baze podataka

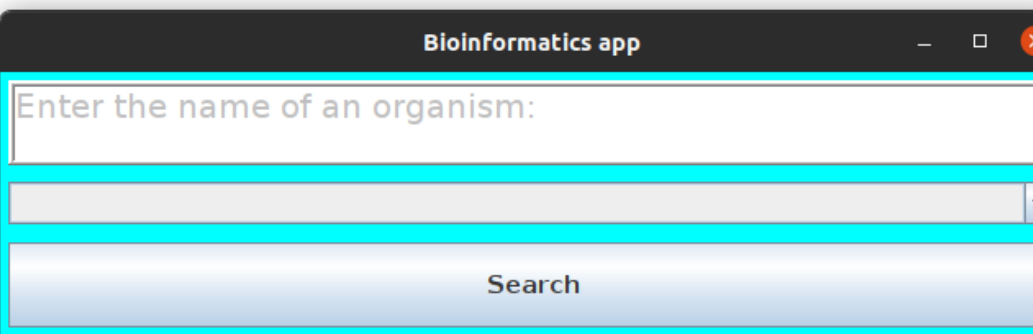
Nakon što korisnik stisne "OK", počinje popunjavanje baze podataka koje može potražiti nekoliko minuta. Za vrijeme popunjavanja cijelo vrijeme se prikazuje gif okretajućeg kruga koji označava učitavanje kako korisnik ne bi mislio da je aplikacija imala neki grešku i prestala raditi.



Slika 4.5: Popunjavanje baze podataka

4.2. Pretraživanje organizama

Nakon popunjavanja baze podataka u aplikaciji se pojavljuje glavni prozor. Taj prozor je ujedno i početni prozor kada se pokreće aplikacija za korisnike koji su već popunili bazu podataka. Ovaj prozor služi kao tražilica u koju korisnik upisuje imena organizama.

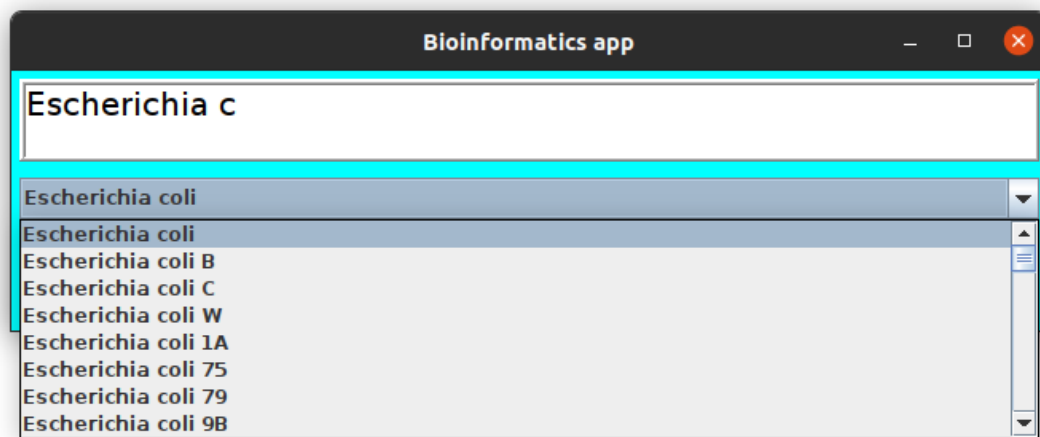


Slika 4.6: Tražilica

Korisnik upisuje ime organizma u tražilicu. Tražilica funkcionira tako da svaki puta kada korisnik ne upiše novo slovo u roku od pola sekunde, pokreće se nova dretva koja koristi *ILIKE* operator da bi pretraživala imena u bazi podataka. U nastavku je prikazan primjer SQL naredbe kojom se traže organizmi sa sličnim imenima.

```
SELECT name_txt FROM names WHERE name_txt ILIKE  
'Escherichia c';
```

Nakon što aplikacija nađe slična imena, ona se pokažu u JComboBox-u te korisnik može odabrati jedno i kliknuti na "Search" gumb.

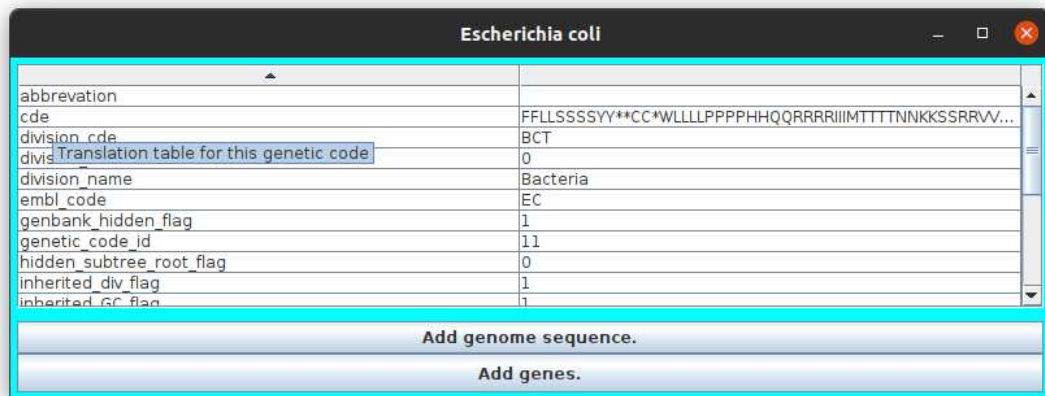


Slika 4.7: Tražilica sa ponuđenim sličnim imenima

Nakon klika na "Search" gumb, pokreće se nova dretva koja traži upisani organizam u bazi podataka. U nastavku je prikazana SQL naredba koja služi za pretraživanje podataka o organizmu.

```
SELECT * FROM names, nodes LEFT OUTER JOIN  
reference_genomes ON nodes.tax_id =  
reference_genomes.tax_id, gencode, division  
WHERE nodes.genetic_code_id = gencode.genetic_code_id  
AND nodes.division_id = division.division_id AND  
nodes.tax_id = names.tax_id AND  
LOWER(name_txt) = TRIM(LOWER('Escherichia coli'));
```

U slučaju da u bazi podataka ne postoji organizam sa upisanim imenom, korisniku se otvara skočni prozor koji javlja grešku. Inače se otvara novi prozor koji prikazuje podatke o odabranom organizmu.



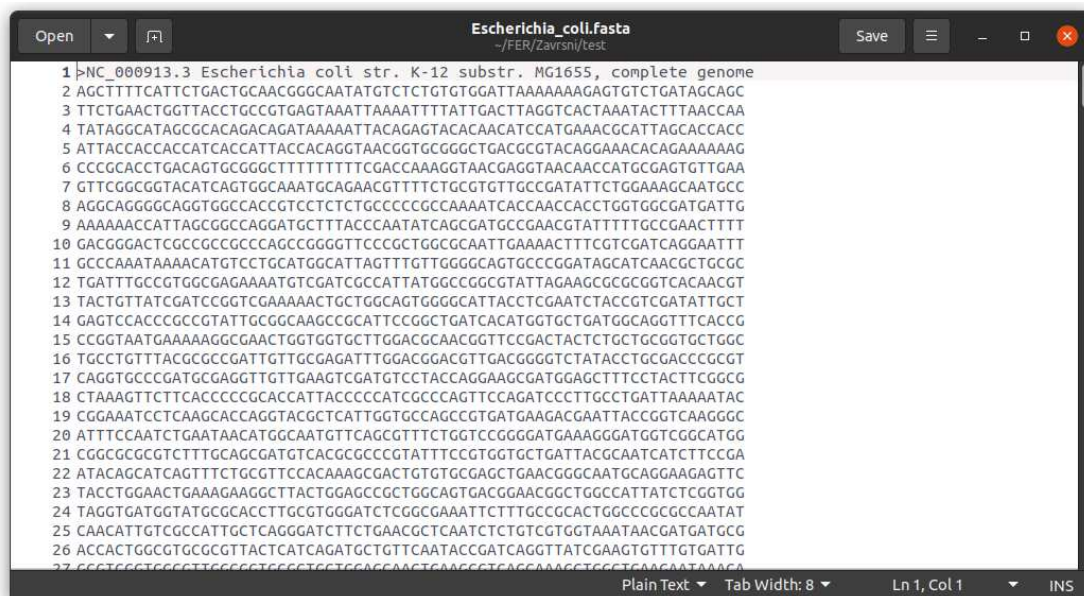
Escherichia coli	
abbreviation	
cde	FFLLSSSSYY**CC*WLLLLPPPHQHQRRIIIMTTTTNNKKSSRRVV...
division_cde	BCT
divis	Translation table for this genetic code
division_name	0
embl_code	Bacteria
genbank_hidden_flag	EC
genetic_code_id	1
hidden_subtree_root_flag	11
inherited_div_flag	0
inherited_GC_flag	1
Add genome sequence.	
Add genes.	

Slika 4.8: Prozor sa informacijama o organizmu

Na slici 4.8 vidi se tooltip koji daje opis svakog podatka kada se mišem pređe preko njega (u ovom primjeru miš se nalazi na podatku "cde").

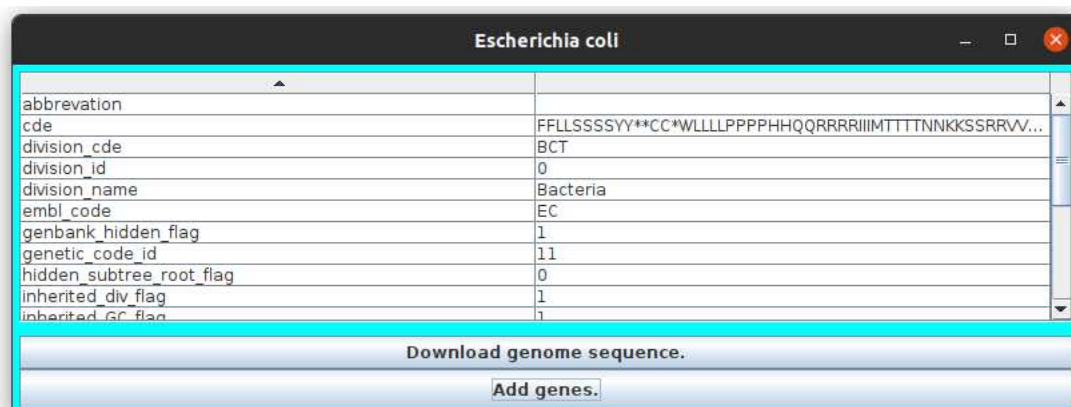
4.3. Dodavanje novog genoma

U slučaju da sekvenca genoma i podaci o genima nisu spremljeni, odnosno da aplikacija ne može naći te podatke u bazi podataka, na prozoru sa informacijama o organizmu biti će aktivni gumbi za dodavanje istih. Klikom na gumb "Add genome sequence." otvara se novi JFileChooser gdje korisnik treba odabrati jednu ili više FASTA datoteka (ako ih je više, spojiti će se u jednu) koje su preuzete iz NCBI-eve biblioteke. Primjer FASTA datoteke je prikazan na slici 4.9.



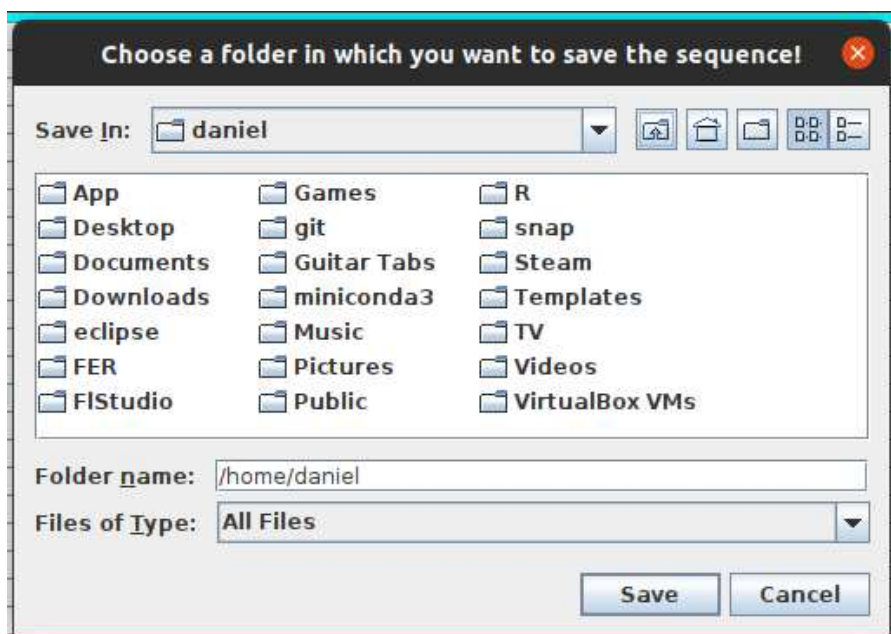
Slika 4.9: Primjer FASTA datoteke

Kada korisnik odabere FASTA datoteku, program ju čita te parsira pročitane podatke na način da mapira zaglavlja svake sekvence kao ključeve te same sekvence kao vrijednosti. U slučaju da korisnik odabere više datoteka, sve će na isti način biti pročitane i mapirane. Nakon što su mapirane, datoteke se spremaju u `src/resources/reference_genomes/` direktorij pod imenom `Naziv_organizma.fasta` (npr. `Escherichia_coli.fasta`), a njihove lokacije zajedno sa svim zaglavljima pohranjuju se u bazu podataka. Nakon što je spremanje genoma gotovo, gumb sa akcijom za dodavanje genoma sada ima novu akciju za preuzimanje istih.



Slika 4.10: Prozor sa informacijama o organizmu za koji je spremljena sekvenca genoma

Klikom na gumb "Download genome sequence." opet se otvara JFileChooser, ali sada on traži direktorij u koji korisnik želi spremiti sekvence genoma. Korisnik tako preuzima sve spremljene sekvence genoma za odabrani organizam.



Slika 4.11: Skočni prozor koji pita korisnika gdje želi spremiti FASTA datoteku sa sekvencom genoma

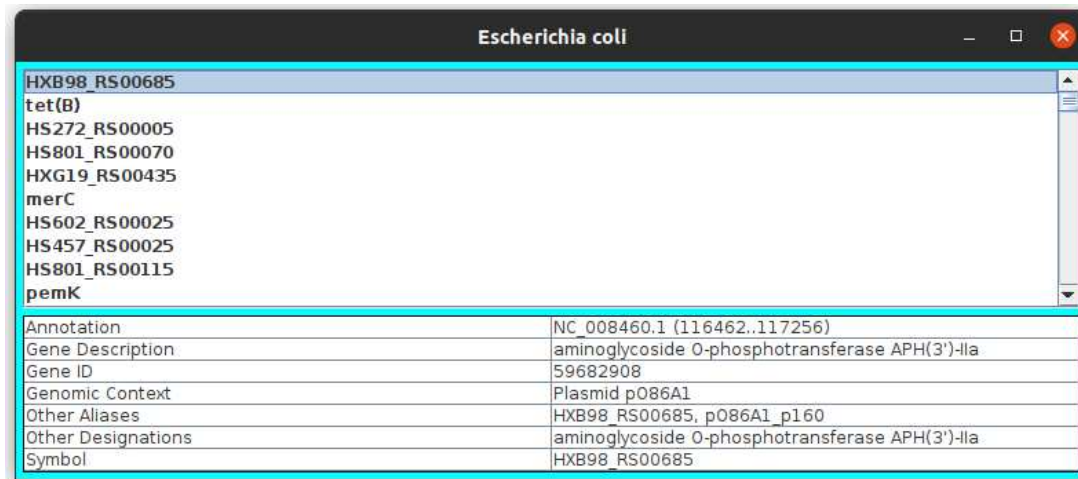
4.4. Dodavanje podataka o genima

Klikom na gumb "Add genes." u prozoru sa podacima o organizmu otvara se JFileChooser te korisnik treba odabrati tekstualnu datoteku koja sadrži sažetak gena organizma koji je preuzet sa NCBI-eve biblioteke. Kada korisnik odabere odgovarajuću datoteku, u bazu podataka se spremaju podaci o genima te se akcija na gumbu mijenja u akciju za prikazivanje gena.



Slika 4.12: Prozor sa informacijama o organizmu za koji su spremljeni geni

Klikom na gumb "Show genes." otvara se novi prozor u kojem su kao lista prikazani svi poznati geni organizma. Klikom na bilo koji gen iz liste pokazuju se informacije o istom.



Slika 4.13: Prozor sa popisom gena i informacijama o njima

4.5. Rekurzivno pretraživanje po taksonomskom stablu

Pojedini organizmi mogu imati puno podvrsta što je prikazano slikom 4.14.

The screenshot shows the NCBI Taxonomy Browser interface. The search bar contains "Salmonella enterica" and the search type is set to "complete name". The results are displayed in a hierarchical list format. The "Lineage" section shows the taxonomic path: cellular organisms; Bacteria; Proteobacteria; Gammaproteobacteria; Enterobacterales; Enterobacteriaceae; Salmonella. The "Salmonella enterica" section lists various subspecies and serovars, each with a "LinkOut" button. The list includes:

- Salmonella enterica subsp. arizonae serovar 11:13,23:g,z51:-
- Salmonella enterica subsp. arizonae serovar 1,13,23:g,z51:-
- Salmonella enterica subsp. arizonae serovar 13,23:g,z51:-
- Salmonella enterica subsp. arizonae serovar 13:g,z51:-
- Salmonella enterica subsp. arizonae serovar 17:z29:-
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:-
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. 91293
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM 32450
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM 32457
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM 43478
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM 43479
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM 43480
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM 43481
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM 43482
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N18383
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N18503
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N18554
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N20028
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N23850
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N25373
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N26624
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N26625
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N26626
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N26928
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N27
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N29354
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N31597
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N4410
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N5
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N6509
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N7307
- Salmonella enterica subsp. arizonae serovar 18:z4,z23:- str. CVM N9135
- Salmonella enterica subsp. arizonae serovar 18:z4,z32:- str. 75773
- Salmonella enterica subsp. arizonae serovar 18:z4,z32:-
- Salmonella enterica subsp. arizonae serovar 35:z4,z32:-
- Salmonella enterica subsp. arizonae serovar 38:z4,z23:-

Slika 4.14: Snimka zaslona kod pretrage za Salmonellu entericu u NCBI-evoj bazi [1]

Iz tog razloga aplikacija nakon klika na gumb "Search" u istoj dretvi u kojoj je tražila organizam sa upisanim imenom u slučaju da za taj organizam nije spremljena sekvenca genoma, rekurzivno pomoću SQL upita traži ima li neki roditeljski organizam spremljenu sekvencu genoma. Traženje roditelja sa spremljenom sekvencom genoma ostvaruje se slijedećom SQL naredbom.

```

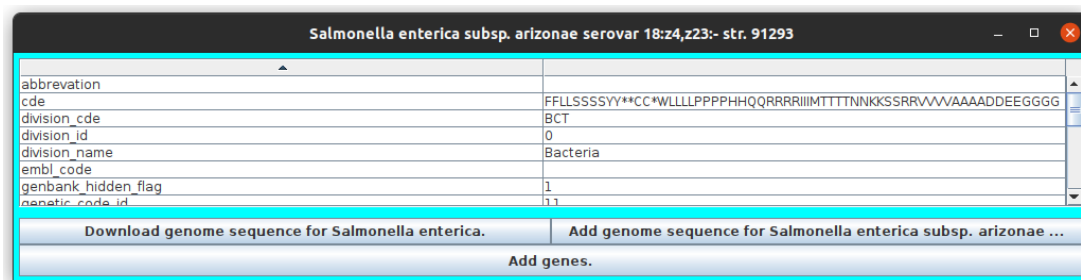
WITH RECURSIVE sub_tree AS (SELECT nodes.tax_id ,
names.name_txt , nodes.parent_tax_id , file_location
FROM names , nodes LEFT OUTER JOIN reference_genomes
ON nodes.tax_id = reference_genomes.tax_id , gencode ,
division WHERE nodes.genetic_code_id =
gencode.genetic_code_id AND nodes.division_id =
division.division_id AND nodes.tax_id = names.tax_id
AND name_txt = 'Salmonella enterica subsp.
arizonae serovar 18:z4,z23:- str. 91293'

UNION ALL

SELECT nod.tax_id , nam.name_txt , nod.parent_tax_id ,
gen.file_location FROM sub_tree st , names nam , nodes
nod LEFT OUTER JOIN reference_genomes gen
ON nod.tax_id = gen.tax_id , gencode gc , division div
WHERE nod.genetic_code_id = gc.genetic_code_id AND
nod.division_id = div.division_id AND nod.tax_id =
nam.tax_id AND st.parent_tax_id = nod.tax_id
AND nam.name_class = 'scientific name'
)
SELECT * FROM sub_tree
LIMIT 5;

```

Prethodna SQL naredba rekurzivno pretražuje po bazi te vraća podatke prvih četiri roditelja organizma. Taj broj je ograničen jer što je veći broj odabran, to će aplikacija sporije provoditi tu naredbu te će sporije otvarati slijedeći prozor. Također, pretpostavlja se da većina organizama neće imati više od pet roditelja. Nakon što SQL naredba vrati podatke, aplikacija provjerava ima li neki od roditelja zapisanu lokaciju sekvence genoma. Ukoliko ima, aplikacija će u idućem prozoru ponuditi dvije opcije: preuzimanje sekvence genoma za glavni organizam ili spremanje nove sekvence genoma za tu podvrstu.



Slika 4.15: Prozor sa informacijama o organizmu za čijeg roditelja je već spremljena sekvenca genoma

4.6. Testiranje podataka

Aplikacija je testirana sa većim brojem podataka kako bi se moglo sa sigurnošću tvrditi da parsiranje istih i spremanje u bazu podataka funkcionira.

Sa NCBI-eve biblioteke jednostavno se mogu preuzeti sažeci podataka o genima i sekvence genoma za pojedine organizme. Testiranje je obavljeno na slijedećim organizmima: *Escherichia coli*, *Salmonella enterica*, *Drosophila melanogaster*, *Klebsiella pneumoniae*, *Saccharomyces cerevisiae*, *Canis lupus*, *Homo sapiens*, *Arabidopsis thaliana*, *Bos taurus* i *Caenorhabditis elegans*. Svi preuzeti podaci su ručno dodani u bazu putem aplikacije te su rezultati bili isti kao što su očekivani.

5. Korišteni alati

Budući da je za programski jezik odabrana Java, kao razvojno okruženje za izradu aplikacije korišten je Eclipse IDE koji znatno olakšava praćenje principa objektno orijentiranog programiranja. Objektno orijentirana programska paradigma je implementirana jer je ona standard u računarstvu te znatno smanjuje kompleksnost razvoja i održavanja programske opreme [3]. Za pregledavanje i upravljanje bazom podataka korišten je pgAdmin koji je najpopularnija Open Source platforma za PostgreSQL. Za pravljenje ER modela i generiranje relacijskog modela te create table naredbi korišten je ERDPlus kao alat za modeliranje baza podataka.

6. Zaključak

Cilj ovog završnog rada bio je organizirati bazu podataka te napraviti desktop aplikaciju za pohranu bioinformatičkih podataka. Programsko rješenje ostvareno je korištenjem jezika Java i PostgreSQL.

Završni rad sastoji se od dva dijela. U prvom dijelu napravljena je baza podataka te su u njoj pohranjeni podaci koji su preuzeti sa NCBI-eve stranice. Drugi dio rada bila je izrada aplikacije pomoću koje se mogu prikazivati spremljeni podaci, ali i dodavati novi.

Samo korisničko sučelje aplikacije je intuitivno za korištenje, dok aplikacija omogućuje sve funkcionalnosti navedene u opisu zadatka. Aplikacija je funkcionalna te bi korisnicima koji se bave bioinformatikom mogla biti od koristi. Što se tiče širine aplikacije i kvantitete podataka koji se spremaju, aplikacija se još može nadograditi. Primjerice, za organizam bi se mogli dodati i podaci o nukleotidima, proteinima, a moglo bi se prikazati i cijelo taksonomsko stablo pojedinog organizma.

Ovaj završni rad bio je mogućnost za dobro upoznavanje sa alatima i jezicima koji su korišteni, a sva nova znanja usvojena prilikom izrade rada zasigurno će biti korisna pri izradi budućih projekata.

LITERATURA

- [1] Ncbi taxonomy, 2020. URL <https://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi>. Pristupano 6.6.2021.
- [2] Nastavni materijali iz kolegija Baze podataka, 2021. URL https://www.fer.unizg.hr/predmet/bazpod_c/materijali. Pristupano 7.6.2021.
- [3] Nastavni materijali iz kolegija Objektno orijentirano programiranje, 2021. URL <https://www.fer.unizg.hr/predmet/oop/predavanja>. Pristupano 6.6.2021.
- [4] Organizam. *Hrvatska enciklopedija, mrežno izdanje*, 2021. URL <http://www.enciklopedija.hr/Natuknica.aspx?ID=45473>. Pristupano 6.6.2021.
- [5] M. Sikic i M. Domazet-Loso. *Skripta iz bioinformatike*. Fakultet elektrotehnike i računarstva, Sveučilište u Zagrebu, 2013. URL https://www.fer.unizg.hr/_download/repository/bioinformatika_skripta_v1.2.pdf.
- [6] *PostgreSQL službena dokumentacija*. The PostgreSQL Global Development Group, 2021. URL <https://www.postgresql.org/docs/>.

POPIS SLIKA

3.1. ER model	6
3.2. Relacijski model	11
4.1. Login prozor	13
4.2. Ispravno popunjen login prozor	14
4.3. Skočni prozor nastao zbog neispravno upisanih podataka	14
4.4. Skočni prozor koji obavještava da će krenuti popunjavanje baze podataka	14
4.5. Popunjavanje baze podataka	15
4.6. Tražilica	15
4.7. Tražilica sa ponuđenim sličnim imenima	16
4.8. Prozor sa informacijama o organizmu	17
4.9. Primjer FASTA datoteke	18
4.10. Prozor sa informacijama o organizmu za koji je spremljena sekvenca genoma	18
4.11. Skočni prozor koji pita korisnika gdje želi spremiti FASTA datoteku sa sekvencom genoma	19
4.12. Prozor sa informacijama o organizmu za koji su spremljeni geni . . .	20
4.13. Prozor sa popisom gena i informacijama o njima	20
4.14. Snimka zaslona kod pretrage za Salmonellu entericu u NCBI-evoj bazi [1]	21
4.15. Prozor sa informacijama o organizmu za čijeg roditelja je već sprem- ljena sekvenca genoma	23

POPIS TABLICA

3.1. Nodes	8
3.2. Names	9
3.3. Division	9
3.4. Gencode	9
3.5. Reference_genomes	10
3.6. Headers	10
3.7. Genes	10

Aplikacija za pohranu bioinformatičkih podataka u sustavu PostgreSQL

Sažetak

U sklopu završnog rada napravljena je aplikacija u kojoj se mogu pretraživati organizmi i pregledavati podaci o istima. Organizmima se dodatno mogu spremati i podaci o genima te sekvence referentnih genoma koje se pohranjuju kao FASTA datoteke na disku. Završni rad temelji se na dva dijela: baza podataka i aplikacija. U bazi podataka pohranjeni su svi podaci o organizmima dohvaćeni iz NCBI-eve (Nacionalni centar za biotehnološke informacije) biblioteke, a putem aplikacije se ti isti podaci prikazuju. Za implementaciju baze podataka korišten je PostgreSQL sustav, a sama aplikacija je programirana u jeziku Java uz pomoć Swing biblioteke.

Ključne riječi: bioinformatika, organizam, gen, genom, FASTA, Java, Swing, PostgreSQL

Application for Storing Bioinformatics Data Using PostgreSQL

Abstract

In this thesis an application was made in which organisms can be searched and its data can be viewed. Organisms can additionally store gene data and sequences of reference genomes which are stored as FASTA files on the disk. This thesis is based on two parts: database and application. The database contains all data on organisms retrieved from the NCBI (National Center for Biotechnology Information) library, and the same data is displayed via the application.

For implementation of database PostgreSQL system was used, and the applicaion itself was programmed in in the Java language with the help of the Swing library.

Keywords: bioinformatics, organism, gene, genome, FASTA, Java, Swing, PostgreSQL