

УНИВЕРЗИТЕТ У БЕОГРАДУ
МАТЕМАТИЧКИ ФАКУЛТЕТ



Григории И. Рогов

РАЗВОЈ ВЕБ АЛАТА ЗА ДЕТЕКЦИЈУ И
АНАЛИЗУ ГЕНОМСКИХ ВАРИЈАЦИЈА

мастер рад

Београд, 2026.

Ментор:

проф. др Јована КОВАЧЕВИЋ, ванредни професор
Универзитет у Београду, Математички факултет

Чланови комисије:

др Мирјана МАЉКОВИЋ РУЖИЧИЋ, доцент
Универзитет у Београду, Математички факултет

Невена ТИРИЋ, асистент
Универзитет у Београду, Математички факултет

Датум одбране: _____

*Захваљујем Србији и њеном народу на
постојанству и прилици да у овим тешким
временима мирно живим, радим и студирам.*

*Искрено захваљујем својој менторки, проф. др
Јовани Ковачевић, на помоћи, издвојеном времену и
грађоценим саветима.*

Овај рад посвећујем својој супрузи Анастасији.

Наслов мастер рада: Развој веб алата за детекцију и анализу геномских варијација

Резиме: Савремена геномска анализа често подразумева употребу већег броја специјализованих алата за припрему података, поравнање секвенци, откривање генетичких варијација и анализу еволуционих односа. Њихово комбиновање захтева ручни пренос података између различитих формата и окружења, као и додатно управљање међурезултатима. У овом мастер раду анализирани су постојећи приступи и софтверски алати за обраду и поређење геномских секвенци, са циљем утврђивања могућности за обједињавање најважнијих корака анализе у јединствен систем.

Као резултат рада развијен је софтверски систем *JELICA* (*Juxtaposing Evolutionary Lineages in Comparative Analysis*), намењен аутоматизацији упоредне анализе целих или приближно целих геномских секвенци узимајући у обзир њихову повезаност. Систем обухвата проверу и припрему улазних података, израчунавање нумеричких карактеристика секвенци, вишеструко поравнање секвенци, упоредну анализу, формирање матрице генетичких растојања, реконструкцију филогенетског стабла и издвајање група сродних секвенци. Задачи у оквиру тражене геномске анализе могу се покретати и пратити преко интерфејса командне линије, веб-платформе и десктоп апликације, док се резултати могу приказивати интерактивно и извозити у облику извештаја. За њихово чување и размену уведен је преносиви формат `.jelica`, који обједињује улазне податке, међурезултате и пратеће метаподатке. Оваква организација омогућава да се поступак геномске анализе извршава на доследан и репродуцибилан начин, уз мању потребу за ручним повезивањем независних алата и обрадом њихових резултата.

Кључне речи: Биоинформатика; геномска анализа; упоредна анализа секвенци; вишеструко поравнање секвенци; филогенетска анализа; генетичко растојање; систем *JELICA*

Садржај

1	Увод	1
2	Анализа алата за геномску анализу	2
2.1	Избор алата	2
2.1.1	Критеријуми избора	3
2.1.2	Примена критеријума избора	3
2.2	Методика анализе алата	5
2.3	Скуп података за тестирање	6
2.4	Анализа изабраних алата	7
2.4.1	Анализа алата <i>BLAST</i>	7
2.4.2	Анализа алата <i>Nextclade</i>	8
2.4.3	Анализа алата <i>MAFFT</i>	10
2.4.4	Анализа алата <i>Clustal Omega</i>	12
2.4.5	Анализа алата <i>Bowtie</i>	13
2.4.6	Анализа алата <i>QUAST</i>	15
3	Развој система <i>JELICA</i>: пројектовање и имплементација	18
3.1	Намена и општи концепт алата	18
3.1.1	Намена алата	18
3.1.2	Основни сценарији употребе	18
3.1.3	Границе функционалности и ограничења	20
3.2	Спецификација система <i>JELICA</i>	20
3.2.1	Функционална спецификација	21
3.2.2	Спецификација критеријума квалитета система	26
3.2.3	Спецификација формата улазних и излазних података	28
3.3	Пројектовање система	28
3.3.1	Избор архитектуре система	28

3.3.2	Општа архитектура система	29
3.3.3	Пројектовање тока аналитичког задатка	32
3.3.4	Конфигурације и модел резултата	33
3.3.5	Видови коришћења система	37
3.4	Избор технологија и развојних алата	37
3.4.1	Аналитичко језгро	38
3.4.2	Серверски део и <i>API</i>	38
3.4.3	Веб-интерфејс	38
3.4.4	Десктоп апликација и <i>CLI</i>	39
3.4.5	Изградња, тестирање и испорука система	39
3.4.6	Развојно окружење и примена вештачке интелигенције .	40
3.5	Имплементација система <i>JELICA</i>	40
3.5.1	Организација имплементације система	41
3.5.2	Модел аналитичког задатка и управљање извршавањем .	41
3.5.3	Имплементација тока аналитичког задатка	44
3.5.4	Чување стања, међурезултата и резултата аналитичког задатка	46
3.5.5	Поузданост, опоравак и праћење тока извршавања	47
3.5.6	Кориснички интерфејси	49
3.5.7	Инфраструктура веб-платформе	54
3.5.8	Корисничка документација	55
4	Закључак	56
	Библиографија	57

Глава 1

Увод

Упоредна анализа геномских секвенци има значајну улогу у истраживању генетичке разноврсности и еволуционих веза између организама. Она омогућава откривање нуклеотидних супституција, инсерција, делеција, као и идентификовање варијанти карактеристичних за појединачне еволуционе групе. Упркос постојању великог броја софтверских решења, многа од њих су уско специјализована, намењена појединачним фазама анализе или одређеним типовима организама. Узастопна примена више таквих алата повећава утрошак времена, а од корисника захтева и разумевање специфичности рада сваког од њих. Наведене чињенице указују на оправданост развоја интегрисаног алата за геномску анализу са интуитивним корисничким интерфејсом.

Циљ овог рада је пројектовање и развој алата *JELICA* — *Juxtaposing Evolutionary Lineages in Comparative Analysis* (поређење еволуционих линија у оквиру упоредне анализе), намењеног анализи више геномских секвенци. Алат обједињује проверу улазних података, поравнање секвенци, откривање разлика међу секвенцама (укључујући *SNP*, *indel*, позиције са празнинама у поравнању (енгл. *gap*) и неодређене позиције (нпр. означене симболом *N*)) и представљање резултата, са нагласком на анализу еволуционих односа између секвенци и прелиминарно сврставање секвенци у филогенетске групе.

У поглављу 2 анализирани су постојећи алати и изведени закључци значајни за пројектовање система. У поглављу 3 описани су пројектовање и имплементација система *JELICA*. Рад се закључује поглављем 4, у којем су сумирани резултати и наведени правци даљег развоја.

Изворни текст рада у формату ЛАТ_ΕX и пратећи материјали доступни су у репозиторијуму на адреси <https://github.com/RogovGreg/master-thesis>, а изворни код алата *JELICA* на <https://github.com/RogovGreg/jelica>.

Глава 2

Анализа алата за геномску анализу

У данашње време постоји велики број програмских алата намењених геномској анализи, поравнавању секвенци, откривању геномских варијација, контроли квалитета и визуелизацији резултата. Ови алати се разликују према намени, подржаним типовима података, примењеним методама анализе, начинима представљања резултата и другим карактеристикама.

Пре развоја сопственог алата сврсисходно је размотрити неколико популарних решења и анализирати њихове основне могућности. Таква анализа омогућиће препознавање успешних приступа које би нови алат могао да користи при представљању резултата и интеракцији са корисником, а може помоћи и приликом уочавања ограничења постојећих алата и формулисања захтева за систем који се развија. Изведени закључци могу се користити при избору функционалности, структуре корисничког интерфејса и општег сценарија рада алата.

2.1 Избор алата

У овој секцији описан је поступак избора алата за даљу анализу. Најпре су дефинисани обавезни критеријуми, након чега је приказана њихова примена при формирању коначног списка алата.

2.1.1 Критеријуми избора

Као почетни извор коришћен је регистар *bio.tools*, који садржи податке о програмским алатима и базама података у области биоинформатике. Сваки запис обухвата намену алата, подржане формате, тематске категорије, услове приступа, публикације и друге карактеристике. Пошто већину поменутих поља у запису попуњавају аутори алата и она нису обавезна, подаци могу бити непотпуни или застарели. Због тога је избор спроведен у две фазе: аутоматизована прелиминарна филтрација посредством *API*-ја сервиса *bio.tools* и накнадна ручна провера резултата.

Обавезни критеријуми избора били су:

- **Доступност.** Алат мора бити јавно доступан, без потребе за коришћењем затворене инфраструктуре или добијањем посебне дозволе.
- **Цена.** Основна анализа мора бити бесплатна, док се бесплатна регистрација корисника сматрала прихватљивом.
- **Начин коришћења.** Мора бити доступан веб-интерфејс и/или званична верзија за *Windows*, *MacOS* или *Linux*.
- **Формат улазних података.** Алат мора прихватати нуклеотидне секвенце у формату *FASTA*.
- **Релевантна функционалност.** Алат мора подржавати најмање један од разматраних задатака: поравнавање секвенци, откривање геномских варијанти у односу на референтну секвенцу, поређење више секвенци или израчунавање основних нумеричких карактеристика дате секвенце.
- **Пратећи материјали.** Морају бити доступни документација, упутство, материјали за обуку или примери улазних и излазних података.
- **Научне публикације.** Алат мора бити повезан са најмање једном научном публикацијом.

2.1.2 Примена критеријума избора

За прелиминарну филтрацију развијена је *Python* скрипта `fetch_biotoools.py`, која је додата у репозиторијум рада. Део резултата добијених извршавањем

Done. Total unique tools with publications: 163

#	Name	biotoolsID	Citation score
1	MAFFT	MAFFT	46292.50
2	BLAST	blast	35406.80
3	SILVA rRNA database	silva	34890.60
4	MUSCLE	muscle	28501.25
5	Clustal 2 (Clustal W, Clustal X)	clustal2	27123.05
6	make_nr	make_nr	16737.20
7	Clustal Omega	clustalo	12631.05
8	Salmon	salmon	11269.35
9	Bowtie	bowtie	11043.30
10	ClustalW (BioLib)	clustalw_biolib	10344.00
11	QUAST	quast	8907.95
12	eggNOG	eggnoG	7479.25
13	Jalview	Jalview	6461.80
14	JASPAR	jaspar	6432.10
15	MMseqs2	MMseqs2	6233.50
16	Jellyfish	jellyfish	3762.35
17	CHOPCHOP	chopchop	3599.35
18	nf-core-viralrecon	nf-core-viralrecon	3572.95
19	nf-core-chipseq	nf-core-chipseq	3572.95
20	nf-core-rnaseq	nf-core-rnaseq	3572.95
21	nf-core-smrnaseq	nf-core-smrnaseq	3572.95
22	nf-core-cageseq	nf-core-cageseq	3572.95
23	nf-core-methylseq	nf-core-methylseq	3572.95
24	nf-core-hic	nf-core-hic	3572.95
25	nf-core-atacseq	nf-core-atacseq	3572.95
26	Meryl	meryl	3147.50

Слика 2.1: Резултат извршавања скрипте `fetch_biotools.py`: првих 26 од 163 алата добијених прелиминарном филтрацијом података из регистра *bio.tools* и ранжираних према израчунатој оцени цитираности.

скрипте приказан је на слици 2.1. Из *EDAM* категорија¹ изабране су области *Genomics*, *Comparative genomics*, *Sequence analysis*, *Genetic variation*, *Structural variation*, *Alignment*, *Gene structure* и *Whole genome sequencing*. За сваку категорију су посредством *API*-ја сервиса *bio.tools* преузимани бесплатно доступни алати који прихватају формат *FASTA* и којима је у овом сервису додељен статус *Mature*². Добијени записи су обједињени, а дубликати су уклоњени на основу идентификатора *biotoolsID*.

За рангирање алата израчунавана је оцена цитираности њихових публи-

¹**EDAM** (енгл. *EMBRACE Data and Methods*) представља онтологију за стандардизовани опис и класификацију биоинформатичких ресурса према областима примене, операцијама, типовима података и форматима. Назив потиче од европског пројекта **EMBRACE** (енгл. *A European Model for Bioinformatics Research and Community Education* — Европски модел за биоинформатичка истраживања и образовање заједнице), који је био усмерен на интеграцију биоинформатичких алата, ресурса и истраживачких група.

²Термин **Mature** представља вредност параметра *maturity*, који означава степен развијености софтверског производа; у сервису *bio.tools* разликују се вредности *Emerging* (у развоју), *Mature* (стабилан и добро развијен) и *Legacy* (застарео или више се не одржава).

кација на основу података сервиса *OpenAlex*³. У обзир су узимани цитати из периода 2017–2026. године: за текућу годину примењиван је коефицијент 1, док се за сваку претходну годину смањивао за 0,05, до вредности 0,55 за 2017. годину. Оцене свих публикација једног алата су сабирале. Алати без наведених публикација били су искључени, док је у случају немогућности преузимања података о публикацији њен допринос оцени износио нула. Након тога је списак сортиран према укупној оцени цитираности.

Коначни избор извршен је ручно, при чему је ранг-листа коришћена као помоћни, а не као једини критеријум избора. Из списка су искључене базе података, уско специјализовани алати, решења која су слабо повезана са темом рада и алати сличне намене. Међу алатима сличне намене предност је давана цитиранијим и боље документованим решењима. Као резултат описаног поступка, за даљу анализу одабрано је шест алата: *BLAST*, *Nextclade*, *MAFFT*, *Clustal Omega*, *Bowtie* и *QUAST*. Они обухватају различите задатке, укључујући претрагу сличности, мапирање на референтну секвенцу, вишеструко поравнавање, процену квалитета склопљеног генома и анализу вирусних секвенци.

2.2 Методика анализе алата

Пошто разматрани алати служе за обраду различитих класа задатака, њихово поређење се не спроводи као директно поређење тачности или перформанси. Анализа се изводи према јединственој шеми, која омогућава да се упореде не сами алгоритми, већ подржани кориснички сценарији, типови улазних и излазних података, начини представљања резултата и ограничења значајна за касније пројектовање.

За сваки алат разматрају се његова основна намена, подржани типови улазних података, могућност рада са референтном секвенцом, подршка за анализу више секвенци, постојање функција поравнања, претраге геномских варијанти или процене квалитета, као и начини визуелизације и извоза резултата. Посебна пажња посвећује се корисничком интерфејсу: постојању веб-верзије, *CLI*-интерфејса, сложености подешавања параметара и једно-

³ *OpenAlex* је отворена база података о научним публикацијама и њиховој цитираности, доступна и преко *API*-ја.

ставности тумачења резултата за корисника који није стручњак за конкретан алат.

Практична провера извршена је коришћењем јединственог скупа података за тестирање који садржи секвенце *SARS-CoV-2*, описаног у поглављу 2.3. То омогућава употребу геномских секвенци истог организма при анализи различитих алата, уз истовремено уважавање специфичности сваког од њих. За алате вишеструког поравнања коришћен је обједињени *FASTA*-скуп, док су у *Nextclade* учитаване само анализиране секвенце, без посебног задавања референтне секвенце, јер је она унапред дефинисана изабраним скупом података самог алата. За алате усмерене на друге типове задатака, на пример *Bowtie* и *QUAST*, провера и опис изведени су у складу са њиховом основном наменом. На основу анализе, за сваки алат формулишу се предности, ограничења и закључци значајни за пројектовање сопственог решења.

2.3 Скуп података за тестирање

За практичну проверу разматраних алата припремљен је јединствен скуп нуклеотидних секвенци *SARS-CoV-2* у формату *FASTA*. Састав скупа података за тестирање приказан је у табели 2.1. Као референтна секвенца коришћен је запис *NC_045512.2*, који одговара изолату *Wuhan-Hu-1*. Избор *SARS-CoV-2* заснован је на релативно малој величини генома, широкој заступљености података у отвореним базама и подршци овом организму у више специјализованих алата, укључујући *Nextclade*.

У скуп је укључено шест додатних записа из базе *NCBI GenBank* [15]. Они су изабрани тако да обухвате како секвенце блиске референтној, тако и секвенце каснијих варијанти вируса. Секвенца *Sample 5 (OZ074724.1)* остављена је у скупу као пример нижег квалитета, са уочљивим бројем непознатих симбола. То омогућава процену тога колико јасно алати указују на проблеме у улазним подацима.

Уз рад је приложена архива са изворним материјалима: посебним *FASTA* датотекама за референтну и сваку анализирану секвенцу, обједињеним *multi-FASTA* датотекама за заједничко учитавање секвенци, као и табелом са основним карактеристикама записа, укључујући *accession ID*, дужину секвенце и број непознатих симбола. Такав састав архиве обезбеђује репродуцибилност анализе и омогућава употребу истог скупа података за тестирање при разматрању различитих класа алата.

Табела 2.1: Састав скупа података за тестирање

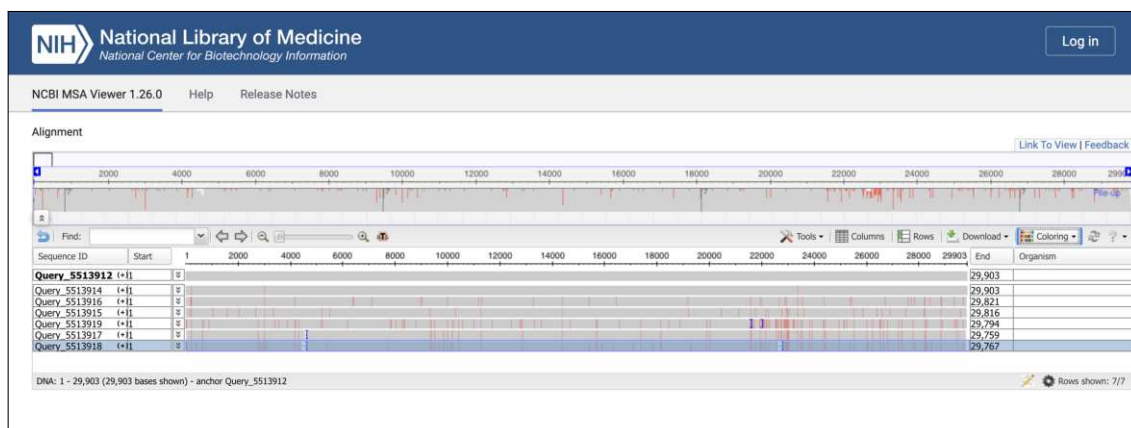
Назив секвенце	<i>Accession</i>	Кратак опис
<i>Reference</i>	<i>NC_045512.2</i>	<i>Wuhan-Hu-1</i> , референтна секвенца <i>SARS-CoV-2</i>
<i>Sample 1</i>	<i>PX622476.1</i>	Кина, 2020; рана секвенца блиска референтној секвенци
<i>Sample 2</i>	<i>OR936774.1</i>	ЈАР, 2021; линија <i>B.1.351</i> , варијанта <i>Beta</i> према класификацији <i>WHO</i>
<i>Sample 3</i>	<i>OR939692.1</i>	ЈАР, 2021; линија <i>B.1.617.2</i> , варијанта <i>Delta</i> према класификацији <i>WHO</i>
<i>Sample 4</i>	<i>PQ726080.1</i>	Шпанија, 2022; цела геномска секвенца <i>SARS-CoV-2</i>
<i>Sample 5</i>	<i>OZ074724.1</i>	Место и година узорковања нису наведени у <i>FASTA</i> заглављу. Секвенца нижег квалитета са неодређеним симболима
<i>Sample 6</i>	<i>PZ368081.1</i>	САД, 2026; каснија секвенца која проширује временски опсег скупа података

2.4 Анализа изабраних алата

2.4.1 Анализа алата *BLAST*

BLAST (*Basic Local Alignment Search Tool*) је један од најпознатијих алата за проналажење сличности између биолошких секвенци. Прва верзија алгорита представљена је 1990. године [2]. За нуклеотидне секвенце користи се варијанта *BLASTN*. Основни тип анализе који *BLAST* покрива јесте проналажење локалне сличности између улазне секвенце и секвенци из базе података или задате референтне секвенце. Алат такође омогућава поређење две задате секвенце, коришћење локалних база података и подешавање формата излаза, али није намењен за вишеструко поравнавање нити за анализу геномских варијанти.

Као улазне податке *BLAST* прихвата секвенце у *FASTA* формату: једну или више упитних секвенци, као и базу података или посебну референтну секвенцу. За припремљени скуп података *SARS-CoV-2* природан сценарио је поређење *sample* секвенци са референтном секвенцом *NC_045512.2*. Резултат рада представљају локална поравнања: идентификатори секвенци са којима су пронађена поклапања, координате поравнатих региона у упитној и циљној секвенци, проценат идентичности, дужина поравнања, број непклапања, празнине, *E-value* и *bit score*. Пример добијених поравнања у



Слика 2.2: Приказ резултата вишеструког поравнавања *SARS-CoV-2* секвенци добијеног *BLAST*-ом у *NCBI MSA Viewer*-у са укљученим режимом обележавања разлика бојом у односу на референтну секвенцу.

NCBI MSA Viewer-у [16], са укљученим режимом обележавања разлика бојом, приказан је на слици 2.2. Корисник може радити са *BLAST*-ом преко веб-интерфејса *NCBI*, који је погодан за ручну проверу, или преко локалних програма *BLAST+*, који су погодни за репродуцибилну и пакетну анализу.

Предности *BLAST*-а су велика брзина, универзалност, широка распрострањеност, подршка за веб-интерфејс и командну линију, као и могућност да се брзо провери да ли је непозната секвенца слична секвенци очекиваног организма или референтној секвенци. Главна ограничења повезана су са тиме што *BLAST* даје локална поклапања, а не комплетну табелу варијанти у односу на референтну секвенцу. Он не формира списак *SNP* и *indel* варијанти, не агрегира варијанте по више секвенци, не врши контролу квалитета улазних података и не повезује пронађене варијанте са генима или функционалним ефектима. За пројектовани алат из овога следи да вреди преузети идеју брзог поређења са референтном секвенцом, адекватне мере сличности и могућност извоза резултата, али је потребно побољшати интерпретацију: аутоматски издвајати геномске варијанте, узимати у обзир неодређене симболе, приказивати квалитет секвенци, агрегирати издвојене варијанте по скупу секвенци и представљати резултате у облику погоднијем за корисника.

2.4.2 Анализа алата *Nextclade*

Nextclade је специјализовани алат за анализу већ састављених вирусних нуклеотидних секвенци. Прва стабилна верзија *Nextclade* 1.0.0 објављена је 2021.

године [1]. Основна намена *Nextclade*-а је упоређивање улазних секвенци са унапред припремљеним референтним скупом података, откривање геномских варијанти у односу на референтну секвенцу, контрола квалитета, сврставање секвенци у познате филогенетске групе, односно кладе или линије. Дакле, овај алат не покрива универзалну анализу произвољних секвенци, већ специјализовани сценарио стандардизоване интерпретације вирусних секвенци.

Као улазне податке *Nextclade* прихвата *FASTA* датотеку са једном или више секвенци и користи изабрани референтни скуп података, који обухвата референтну секвенцу, анотацију генома са положајима гена и кодирајућих региона, филогенетско стабло и параметре анализе. Као резултат се формира табела за сваку секвенцу: списак нуклеотидних замена, промене на нивоу аминокиселина, делеције, инсерције, региони са недостајућим или неодређеним нуклеотидима у односу на референтну секвенцу, мере квалитета анализиране секвенце (мере које указују на потпуност секвенце и могуће проблеме са њеним квалитетом), упозорења, као и додељене кладе или линије. Алат омогућава извоз резултата у више формата. Корисник може радити преко веб-интерфејса, који је погодан за мање скупове података, или преко *CLI* верзије, која је погоднија за обраду већих количина података и репродуцибилна покретања.

Пример резултата добијених анализом секвенци *SARS-CoV-2* из скупа података за тестирање у веб-интерфејсу *Nextclade* приказан је на слици 2.3.

Снага *Nextclade*-а је у томе што обједињује учитавање података, поређење са референтном секвенцом, контролу квалитета и формирање прегледног извештаја у оквиру једног корисничког сценарија. Из ове анализе за сопствени алат корисно је преузети идеју компактне табеле резултата, аутоматске провере квалитета, јасног приказа непознатих и пропуштених позиција, извоза података и прегледног извештаја за сваку секвенцу. При томе не треба понављати снажну зависност од унапред припремљених скупова података: пројектовани алат треба боље да подржи рад са референтним секвенцама које задаје корисник и да раздвоји универзално поређење секвенци од специјализоване интерпретације, која је могућа само уз постојање анотација и спољних база података.

#	i	Sequence name	QC	Clade	Pango lineage (Nextclade)	WHO name	Mut.	non-ACGTN	Ns	Cov.	Gaps	Ins.	FS	SC	Genetic feature
0	0	reference_wuhan_hu_1_NC_045512.2 acc	N M P C F S	19A	B		0	0	0	100.0%	0	0	0	0	
1	2	sample_5_unknown_lineage_OZ074724 acc	N M P C F S	23A	XBB.1.5		92	3	378	98.7%	56	0	0	0 (1)	
2	1	sample_6_2026_usa_PZ368081.1 acc	N M P C F S	25C	XFG.3		150	0	1	99.8%	68	12	0	0	
3	3	sample_4_2022_spain_PQ726080.1 acc	N M P C F S	21L	BA.2	Omicron	66	0	86	99.3%	18	0	0	0	
4	4	sample_3_delta_B.1.617.2_OR939692.1 acc	N M P C F S	21J	B.1.617.2	Delta	36	0	0	99.8%	13	0	0	0	
5	5	sample_2_beta_B.1.351_OR936774.1 acc	N M P C F S	20H	B.1.351	Beta	35	0	19	99.7%	18	0	0	0	
6	6	sample_1_early_2020_PX622476.1 acc	N M P C F S	20A	B.1		3	0	0	100.0%	0	0	0	0	

Слика 2.3: Резултати анализе *SARS-CoV-2* секвенци у веб-интерфејсу *Nextclade*: табела са пронађеним геномским варијантама у односу на референтну секвенцу, мерама квалитета и додељеним кладама.

2.4.3 Анализа алата *MAFFT*

MAFFT је алат за вишеструко поравнавање нуклеотидних и аминокиселинских секвенци. Прва верзија је објављена 2002. године [10]. У оквиру ове анализе *MAFFT* се разматра као представник класе *MSA*⁴ алата: он не претражује секвенце у бази података и не спроводи потпуну анализу геномских варијанти, већ гради вишеструко поравнавање више хомологих секвенци. Поред тога, алат подржава различите алгоритамске режиме поравнавања (нпр. *FFT-NS-1*, *FFT-NS-2*, *L-INS-i*, *G-INS-i* и *E-INS-i*)⁵, локалну и веб-верзију, као и употребу у својству међузорака у сложенијим аналитичким процесима.

Као улаз *MAFFT* прихвата секвенце у *FASTA* формату; у веб-верзији је практичније радити са једним унапред припремљеним *multi-FASTA* фајлом, што представља ограничење када за сваку секвенцу постоји засебан фајл. Као излаз алат даје вишеструко поравнавање, у коме су секвенце доведене на исту дужину додавањем празнина. За скуп података *SARS-CoV-2* са рефе-

⁴*MSA* — *Multiple Sequence Alignment*, вишеструко поравнавање секвенци.

⁵*FFT-NS-1* и *FFT-NS-2* су бржи режими засновани на прогресивном поравнавању. *L-INS-i* је намењен секвенцама са локално сличним регионима, *G-INS-i* секвенцама које су сличне целом дужином, док је *E-INS-i* погодан за секвенце које садрже више сличних региона раздвојених областима које се тешко поравнавају. Ова три режима користе итеративно побољшавање поравнавања.

```
greg@MacBook-Pro-Greg combined % mafft --6merpair --keeplength --addfragments sars_cov2_samples_on1y.fasta ../reference/reference_wuhan_hu_1_NC_045512.2.fasta > mafft_ref_based.fasta
nadd = 6
ppenalty_ex = -10
nthread = 0
blosum 62 / kimura 200
sueff_global = 0.100000
norg = 1
njobc = 2
generating a scoring matrix for nucleotide (dist=200) ... done

Making a distance matrix ..

There are 487 ambiguous characters
  1 / 1
done.

fTEP 5 / 6

Combining ..
done.

done.

addsingle (nuc) Version 7.526
alg=A, model=DNA200 (2), 1.53 (4.59), -0.00 (-0.00), noshift, amax=0.0
0 thread(s)

To keep the alignment length, 12 letters were DELETED.
To know the positions of deleted letters, rerun the same command with the --mapout option.

Strategy:
FFT-NS-fragment (Not tested.)
?

If unsure which option to use, try 'mafft --auto input > output'.
For more information, see 'mafft --help', 'mafft --man' and the mafft page.

The default gap scoring scheme has been changed in version 7.110 (2013 Oct).
It tends to insert more gaps into gap-rich regions than previous versions.
To disable this change, add the --leavegapppyregion option.

greg@MacBook-Pro-Greg combined %
```

Слика 2.4: Излаз унутар терминала након успешног локалног извршавања *MAFFT* команде за референтно поравнавање шест секвенци *SARS-CoV-2* у односу на секвенцу *NC_045512.2*.

реентном секвенцом *NC_045512.2* и више секвенци такав резултат представља вишеструко поравнање секвенци и омогућава њихово поређење по позицијама, као и даље откривање супституција, инсерција, делеција и непознатих региона. Ипак, *MAFFT* не формира комплетну табелу *SNP/indel* варијанти, не повезује аутоматски откривене варијанте са генима на основу анотације референтне секвенце, не оцењује квалитет секвенци у облику посебног извештаја и не даје биолошку интерпретацију пронађених варијанти.

Предности *MAFFT*-а су подршка за стандардни *FASTA* формат, могућност подешавања режима поравнања (нпр. брзи режим *FFT-NS-2*, прецизнији режими *L-INS-i* и *G-INS-i*, као и аутоматски избор режима) и могућност добијања поравнатих секвенци у стандардним форматима погодним за

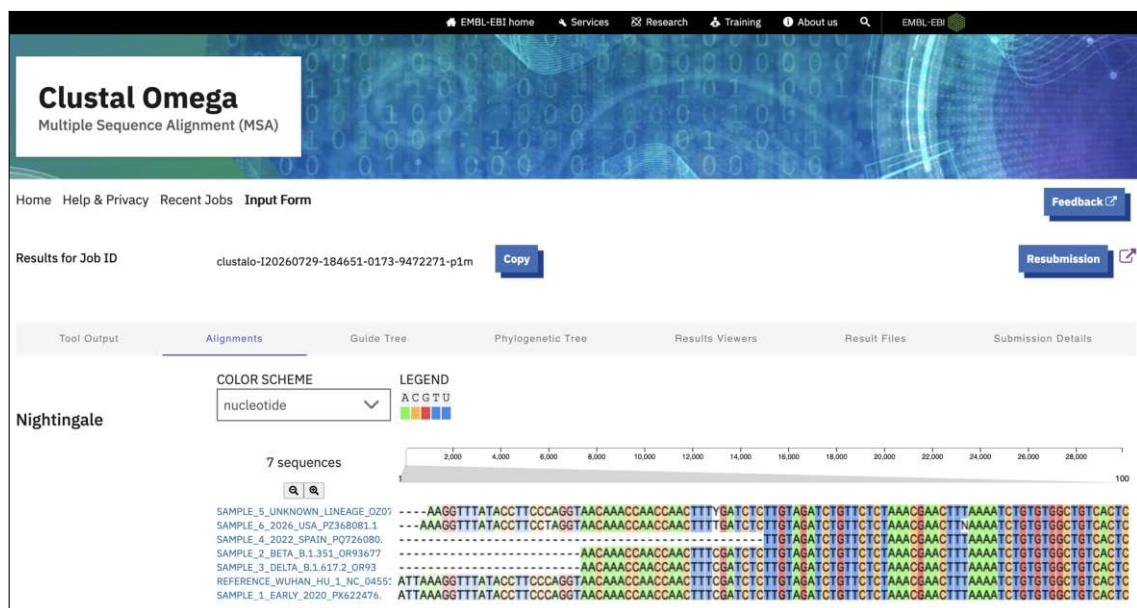
даљу обраду у другим алатима. Главна ограничења нису толико повезана са квалитетом самог поравнавања, већ са корисничким сценаријем: потребно је унапред објединити засебне *FASTA* фајлове, преглед дугих поравнања у веб-интерфејсу и *MAFFT MSA Viewer*-у [13] није довољно удобан, недостаје практична навигација по позицијама геномских варијанти, као и комплетан извештај о геномским варијантама у односу на референтну секвенцу. Из овога за пројектовани алат следи да је корисно преузети идеју стандардног улаза у *FASTA* формату и поравнања секвенци, али је потребно унапредити кориснички интерфејс: омогућити учитавање више засебних фајлова, аутоматско издвајање геномских варијанти у односу на референтну секвенцу, чување координата варијанти у односу на референтну секвенцу, приказивање *QC* (енгл. *Quality Control*, контрола квалитета) упозорења и представљање резултата не само у облику поравнања, већ и у облику табеларног приказа, статистике и прегледне визуализације.

На слици 2.4 приказан је пример успешног локалног извршавања *MAFFT* команде преко терминала у оперативном систему *macOS*.

2.4.4 Анализа алата *Clustal Omega*

Clustal Omega је алат за вишеструко поравнавање секвенци, објављен 2011. године [21]. Он припада класи *MSA* алата и намењен је за поређење три или више хомологних нуклеотидних или протеинских секвенци. У контексту геномске анализе, *Clustal Omega* покрива задатак изградње заједничког поравнања, на основу којег се могу уочити конзервирани региони, супституције, инсерције и делеције. Додатно, алат може да прикаже једноставно стабло које приказује сличност између секвенци и матрицу сличности за сваки пар секвенци, али не врши специјализовану анотацију геномских варијанти.

Као улазне податке *Clustal Omega* прихвата скуп секвенци, најчешће у *FASTA* формату. Референтна секвенца може бити додата у заједнички фајл заједно са осталим секвенцама, али нема посебан статус: алат све секвенце посматра као равноправне. Као излаз корисник добија вишеструко поравнање, а у веб-верзији [7] може да погледа и стабло и матрицу сличности. Пример резултата вишеструког поравнања у веб-интерфејсу приказан је на слици 2.5. *Clustal Omega* је доступан и преко веб-интерфејса и као *CLI* апликација, што олакшава репродукцију анализе података.



Слика 2.5: Приказ резултата вишеструког поравнања у веб-интерфејсу алата *Clustal Omega* за референтну секвенцу *SARS-CoV-2* и шест анализираних секвенци

Главне предности *Clustal Omega* су једноставност употребе, подршка за стандардне формате, као и постојање веб и локалне верзије. Међутим, за задатак ове мастер тезе његов резултат представља само међукорак. *Clustal Omega* не формира табелу *SNP* и *indel* варијанти у односу на изабрану референтну секвенцу, не повезује аутоматски откривене варијанте са генима на основу анотације референтне секвенце, не оцењује квалитет секвенци и не тумачи биолошко значење пронађених варијанти. Из ове анализе се за пројектовани алат може преузети идеја једноставног улаза преко *FASTA* формата и прегледног приказа поравнања, али је неопходно унапредити *reference-based* сценарио: јасно издвојити референтну секвенцу, аутоматски формирати извештај о геномским варијантама, узимати у обзир неодређене симболе и не ограничити се само на визуелно поравнање.

2.4.5 Анализа алата *Bowtie*

Bowtie је алат за брзо мапирање кратких читавања нуклеотидних секвенци на референтни геном. Прва верзија објављена је 2009. године [12]; касније је развијена проширена верзија *Bowtie 2* [11]. У оквиру ове анализе *Bowtie* се разматра као представник класе алата за мапирање читавања на

```
Created 397 pseudo-reads: bowtie_run/reads/sample_1_early_2020_PX622476.1.fastq
397 reads; of these:
  397 (100.00%) were unpaired; of these:
    0 (0.00%) aligned 0 times
    397 (100.00%) aligned exactly 1 time
    0 (0.00%) aligned >1 times
100.00% overall alignment rate
```

Слика 2.6: Приказ покретања алата *Bowtie 2* у терминалу: кратка вештачки припремљена читавања једне секвенце *SARS-CoV-2* мапирана су на референтну секвенцу *NC_045512.2*.

референтну секвенцу (енгл. *reference-based read mapping*), односно за мапирање великог броја кратких читавања са познатом референтном секвенцом. Његов основни сценарио је рана фаза анализе *NGS* података⁶, после које се резултати могу користити за процену покривености, откривање геномских варијанти, изградњу консензусне секвенце или визуализацију поравнања.

Као улазне податке *Bowtie* користи референтну секвенцу и скуп кратких читавања, најчешће у формату *FASTQ*⁷. На излазу се формира датотека са резултатима мапирања, најчешће у формату *SAM* (енгл. *Sequence Alignment/Map*, текстуални формат за запис резултата мапирања читавања на референтну секвенцу), која се затим може претворити у формат *BAM* (бинарна и компримована верзија формата *SAM*) и користити у другим програмима. *Bowtie* алат се користи из командне линије: то није веб-апликација нити интерактивно окружење за анализу, већ биоинформатички алат ниског нивоа. За припремљени скуп података *SARS-CoV-2* применљивост *Bowtie* алата је ограничена, јер скуп садржи већ састављене консензусне *FASTA* секвенце. Због тога је за демонстрационо покретање једна секвенца представљена у облику вештачки припремљених кратких читавања, која су мапирана на референтну секвенцу *NC_045512.2*. Такво покретање приказује типичан кориснички сценарио *Bowtie* алата, али не представља потпуну анализу изворних података добијених секвенцирањем.

Пример извршавања овог сценарија у терминалу приказан је на слици 2.6. У приказаном примеру сва вештачки припремљена читавања изабране секвенце успешно су мапирана на референтну секвенцу. Овај резултат треба

⁶Подаци добијени у поступку секвенцирања нове генерације (енгл. *Next-Generation Sequencing, NGS*).

⁷**FASTQ** је формат за запис читавања нуклеотидних секвенци који, за разлику од *FASTA* формата, поред саме нуклеотидне секвенце чува и вредности квалитета за сваки прочитани симбол.

тумачити као меру успешности мапирања читавања, а не као доказ потпуне идентичности анализиране и референтне секвенце. Чак и у присуству појединачних супституција или других разлика, кратки фрагменти могу успешно да пронађу своју позицију на референтном геному. Због тога резултат *Bowtie* алата сам по себи не даје кориснику комплетну табелу варијанти, не приказује проблематичне позиције и не објашњава биолошко значење пронађених варијанти.

Главне предности *Bowtie* алата су велика брзина, економична употреба меморије и подршка за стандардне формате. Његова ограничења повезана су са уском специјализацијом: алат је оријентисан на рад са кратким читавањима, даје резултат мапирања ниског нивоа, за добијање интерпретабилног резултата захтева комбиновање са другим алатима и не формира комплетан извештај о геномским варијантама у односу на референтну секвенцу. Из ове анализе за пројектовани алат може се преузети идеја приступа заснованог на референтној секвенци и подршка за стандардне формате, али је потребно унапредити кориснички сценарио: прихватити геномске секвенце у форматама *FASTA* и *GenBank*, јасно издвојити референтну секвенцу, аутоматски формирати табелу *SNP* и *indel* варијанти, узимати у обзир неодређене симболе и представљати резултате не само као међуфајлове, већ и као разумљив извештај за корисника.

2.4.6 Анализа алата *QUAST*

QUAST (*Quality Assessment Tool*) је алат за процену квалитета састављања генома, први пут објављен 2013. године у раду *Gurevich et al* [9]. У оквиру ове анализе он се разматра као представник класе алата за контролу квалитета састављања генома (енгл. *genome assembly*). Његова основна намена је процена већ добијеног резултата састављања генома: његове дужине, фрагментираности, потпуности, усклађености са референтном секвенцом и могућих грешака при састављању генома. Додатно, алат формира графиконе, извештаје у различитим форматама и визуелизацију положаја контига⁸ помоћу алата *Icarus*.

Као улазне податке *QUAST* прихвата један или више резултата састављања генома у *FASTA* формату. Референтна секвенца може бити задата

⁸контига (енгл. *contig*) је континуирани фрагмент ДНК секвенце добијен спајањем преклапајућих читавања нуклеотидних секвенци у процесу састављања генома.

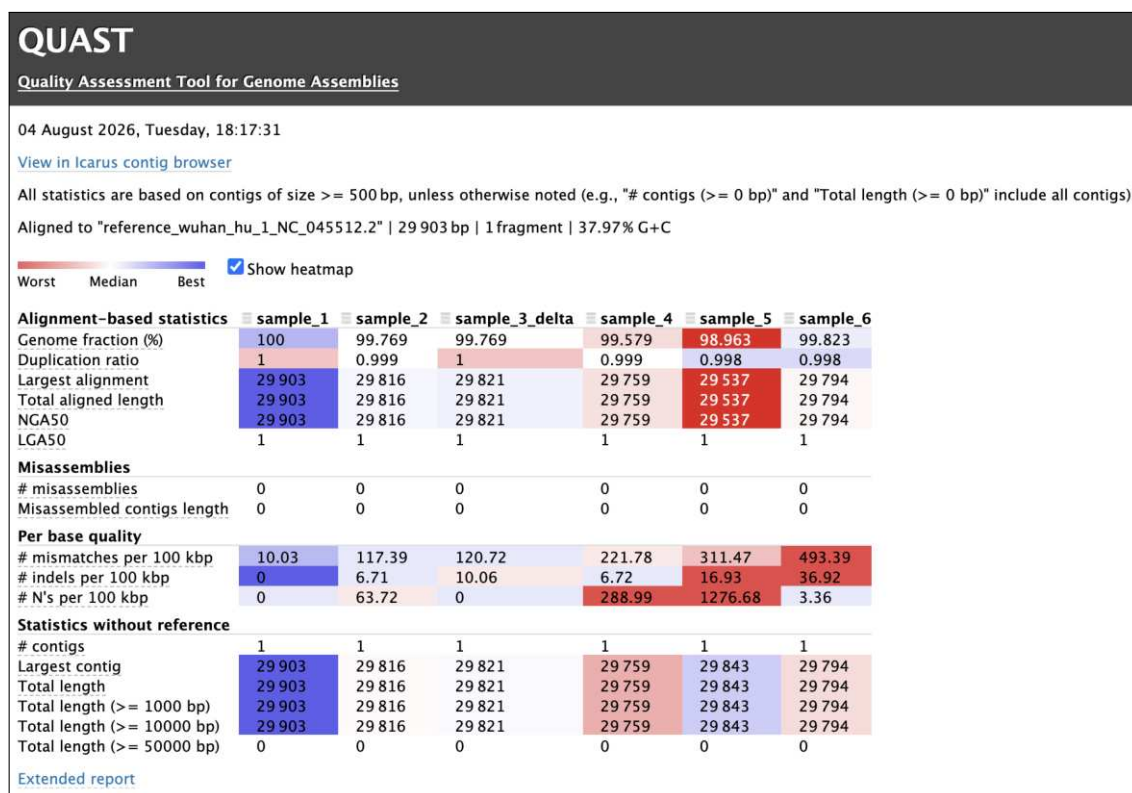
додатно, али није обавезна. Без референце алат израчунава основне статистике за улазне резултате састављања генома: број контига, укупну дужину, дужину највеће контиге, $N50^9$, $L50^{10}$, GC -састав и број неодређених симбола. При постојању референтне секвенце додатно се израчунавају метрике поравнања: удео покривеног генома, дужина поравнатих региона, број непоклапања, инсерција и делеција, број непоравнатих контига и могуће грешке при састављању генома. Резултати се чувају у облику директоријума са текстуалним (.txt), табеларним (.csv), *HTML* и *PDF* извештајима, као и додатним датотекама за визуелизацију.

За припремљени скуп података *SARS-CoV-2 QUAST* је покренут у *reference-based* режиму: шест секвенци у *FASTA* формату предато је као засебни улазни подаци, док је секвенца *NC_045512.2* коришћена као референтна. Пример кратког *HTML* извештаја приказан је на слици 2.7. У њему су улазне секвенце представљене као посебни резултати састављања генома, а табела садржи како метрике израчунате у односу на референцу, тако и метрике без коришћења референтне секвенце.

Главна предност *QUAST*-а је у томе што обједињује покретање анализе, израчунавање QC метрика, формирање графикона и креирање погодног упоредног извештаја за више улазних датотека. Међутим, његова ограничења повезана су са истом том специјализацијом: алат је оријентисан на процену квалитета резултата састављања генома, а не на детаљно поређење нуклеотидних секвенци. *QUAST* не формира комплетну табелу *SNP* и *indel* варијанти у односу на изабрану референтну секвенцу, не повезује варијанте са генима, не агрегира их по више секвенци и не даје интерпретацију пронађених варијанти. За пројектовани алат из ове анализе вреди преузети идеју посебног QC блока, јединствене упоредне табеле, *HTML* извештаја и извоза резултата у машински читљивим форматима. При томе не треба понављати оријентацију *QUAST*-а на метрике на нивоу контиге као главни резултат: за овај рад основни објект анализе треба да буду конкретне разлике између скупа секвенци и референтне секвенце, укључујући *SNP*, *indel*, позиције са празнинама у поравнању и неодређене позиције, њихове координате и збирну статистику за скуп података.

⁹ ***N50*** је метрика фрагментираниости резултата састављања генома: дужина такве контиге да се најмање 50% укупне дужине резултата састављања генома налази у контигама те дужине или дужим.

¹⁰ ***L50*** је број најдужих контига који је потребан да би се покрило најмање 50% укупне дужине резултата састављања генома.



Слика 2.7: Кратак *HTML* извештај алата *QUAST* након демонстрационог покретања на шест припремљених *SARS-CoV-2 FASTA* секвенци у односу на референтну секвенцу *NC_045512.2*: приказане су упоредне метрике поравнања, контроле квалитета и основне статистике секвенци.

Глава 3

Развој система *JELICA*: пројектовање и имплементација

3.1 Намена и општи концепт алата

3.1.1 Намена алата

Софтверски алат *JELICA*, који је развијен у оквиру овог рада, намењен је упоредној анализи секвенци целих или приближно целих генома уз узимање у обзир њихове еволуционе повезаности. Његова основна намена је обједињавање провере и припреме улазних података, израчунавања нумеричких карактеристика, и упоредне анализе, откривања геномских варијација и филогенетске анализе у јединствени софтвер.

Систем омогућава процену сличности и разлика између анализираних секвенци, издвајање претпостављених група еволутивно повезаних секвенци и формирање структурираних резултата погодних за даље истраживање и обраду. Конкретни сценарији употребе алата описани су у наредном потпоглављу.

3.1.2 Основни сценарији употребе

Програмски алат *JELICA* намењен је анализи једне или више целих геномских секвенци. Скуп операција које се могу извршавати зависи од броја учитаних секвенци, постојања референтне секвенце и додатних података које је корисник навео. Основни сценарији употребе алата обухватају:

- **Проверу и анализу нумеричких карактеристика појединачне геномске секвенце**, која укључује проверу њеног формата, целовитости и квалитета, одређивање дужине, нуклеотидног састава и других основних карактеристика.
- **Упоредну анализу више геномских секвенци**, која обухвата њихово вишеструко поравнање, откривање геномских варијанти, укључујући нуклеотидне супституције, инсерције и делеције, као и израчунавање мера сличности и генетичких растојања између секвенци.
- **Анализу секвенци у односу на референтну секвенцу**, коју корисник задаје међу улазним секвенцама, при чему се улазне секвенце поравнавају са заједничком референтном секвенцом, а откривене разлике приказују у оквиру јединственог координатног система.
- **Испитивање скупа без унапред задате референтне секвенце**, при чему систем може да обрађује већ поравнате секвенце и извршава оне фазе анализе за које су испуњени неопходни предуслови. Уколико улазне секвенце нису унапред поравнате, систем у овом режиму не извршава поравнање, па се не покрећу ни фазе које захтевају поравнате секвенце.
- **Одређивање претпостављених еволутивних веза између секвенци** на основу израчунатих растојања између датих секвенци и конструисаног филогенетског стабла, као и аутоматско издвајање претпостављених група еволутивно повезаних секвенци на основу матрице растојања.
- **Анализу група секвенци које је задао корисник**, као што су групе засноване на линијама, кладама, сојевима, географском пореклу или времену, уз утврђивање геномских варијанти карактеристичних за њих и поређење кориснички задате поделе са групама издвојеним на основу генетичких растојања.
- **Проверу квалитета и биолошке упоредивости улазних података**, која омогућава откривање оштећених и парцијалних секвенци, као и оних које се значајно разликују од осталих.
- **Преглед и испитивање резултата анализе**, укључујући нумеричке карактеристике секвенци, резултате поравнања, детаље о појединачним

позицијама у поравнању, откривене геномске варијанте, анализиране секвенце и групе издвојене на основу генетичких растојања.

- **Извоз резултата** у форматима погодним за даљу обраду, визуелизацију и употребу у истраживачком раду.

3.1.3 Границе функционалности и ограничења

JELICA ради са припремљеним геномским секвенцама које представљају целе или приближно целе геноме. Обрада сирових читавања секвенцирања, *de novo* састављање генома¹, аутоматска анотација непознатих генома и исправљање грешака у примарним подацима нису подржани тренутном верзијом алата.

Резултати анализе коју *JELICA* производи имају истраживачки карактер. Откривене разлике не тумаче се аутоматски као функционално значајне, патогене или раније непознате варијанте. Прелиминарне еволуционе групе које систем издвоји представљају претпостављене линије или кладе у оквиру анализираног скупа и не морају одговарати званичним класификацијама.

Поузданост резултата одређена је квалитетом, потпуношћу и упоредивошћу улазних података. Аутоматска валидација омогућава смањење ризика од некоректне анализе, али не може у потпуности заменити стручну процену. Строга ограничења обима података који се обрађују нису постављена, али стварне перформансе система зависе од доступних рачунарских ресурса, због чега повећање броја и величине секвенци може значајно продужити трајање анализе.

3.2 Спецификација система *JELICA*

Спецификација система *JELICA* представљена је кроз три међусобно повезане групе. **Функционална спецификација** описује предвиђене могућности система, **спецификација критеријума квалитета** одређује карактеристике квалитета и услове рада, док **спецификација формата улазних и излазних података** утврђује правила представљања, обраде и формирања података.

¹Састављање генома без употребе референтне секвенце.

3.2.1 Функционална спецификација

Функционална спецификација одређује операције које програмски алат *JELICA* треба да подржава и разрађује функционалности неопходне за имплементацију основних сценарија употребе описаних у потпоглављу 3.1.2. Предвиђене функције обухватају учитавање и валидацију геномских података, спровођење анализе нумеричких карактеристика и упоредне анализе, филогенетску анализу, визуелизацију и извоз добијених резултата.

Начини коришћења система

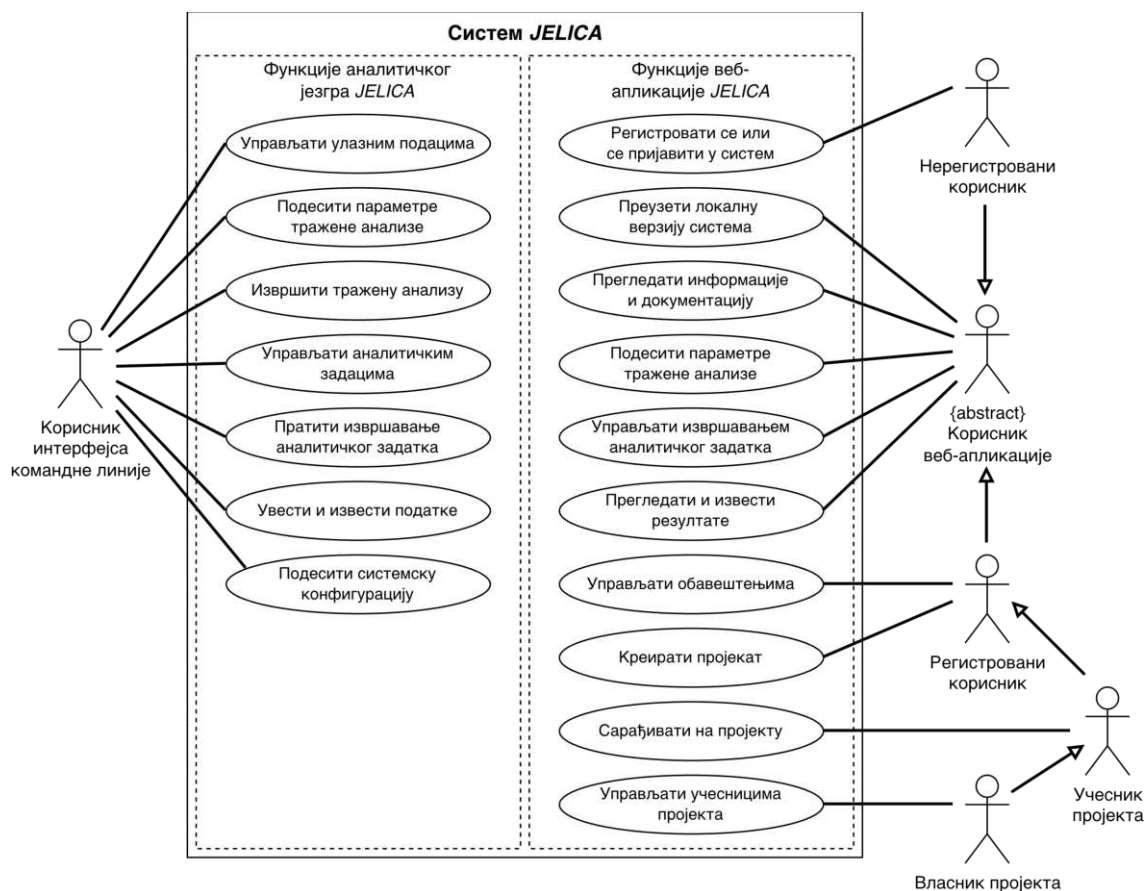
Систем треба да буде доступан у локалној и веб-верзији. Локална верзија треба да се извршава коришћењем рачунарских ресурса корисника и да подразумевано пружа интерфејс командне линије за управљање учитавањем података, покретањем анализе и чувањем резултата. Корисник такође треба да има могућност да засебно инсталира графички интерфејс који користи функционалности локалног аналитичког језгра (описаног у потпоглављу 3.3.1). Веб-верзија треба да представља апликацију у облаку, у којој се чување пројеката и извршавање тражених анализа обављају на серверској инфраструктури система.

Основне функције система *JELICA* приказане су на слици 3.1.

Учитавање и валидација података

Систем треба да омогући учитавање улазних података у форматима дефинисаним у потпоглављу 3.2.3. Након учитавања, систем треба да издвоји основне информације о свакој секвенци, прикаже списак додатих секвенци и омогући кориснику да пре покретања анализе провери које су секвенце укључене у скуп података. Корисник треба да има могућност да уклони грешком учитану секвенцу, преименује је, дода или исправи метаподатке и, по потреби, одреди секвенцу која ће се користити као референтна.

Валидација сваке датотеке треба да буде започета аутоматски након њеног учитавања и да се извршава у позадини. Током њеног извршавања корисник треба да има могућност да настави са учитавањем других датотека и да ради са већ провереним секвенцама. За сваку датотеку систем треба да прикаже тренутно стање обраде, резултат провере и откривене проблеме. Грешка при-



Слика 3.1: UML дијаграм случајева коришћења који приказује основне функције система JELICA.

ликом обраде једне датотеке не треба да онемогући учитавање и валидацију осталих датотека у скупу.

Пре извршавања основне анализе систем треба да спроведе вишеслојну валидацију улазних података. Она треба да обухвати проверу исправности структуре датотека, постојања секвенци у њима и коректности употребљених симбола. За сваку секвенцу треба израчунати основне нумеричке карактеристике, укључујући дужину, нуклеотидни састав, GC-садржај, број и удео непознатих позиција (позиција означених симболом N), као и број или учесталост појављивања k -мера које је задао корисник. При учитавању више секвенци систем такође треба да процени њихову упоредивост, препозна секвенце које се значајно разликују и могуће дупликате. Откривени проблеми треба да буду праћени порукама о грешкама или упозорењима.

Спровођење геномске анализе

Геномска анализа у систему *JELICA* извршава се у оквиру **аналитичког задатка**, који представља конфигурисано покретање анализе над улазним скупом секвенци. Обим и параметри анализе одређују се конфигурацијом аналитичког задатка, која је детаљније описана у потпоглављу 3.3.4.

Ако је учитана једна секвенца, систем треба да израчуна њене нумеричке карактеристике и формира њихов преглед. Функције за које је неопходно постојање више секвенци у том случају не треба да буду покренуте.

Ако постоји више упоредивих секвенци и конфигурацијом је предвиђено формирање поравнања у односу на задату референтну секвенцу, систем треба да изврши њихово вишеструко поравнање. Систем такође треба да подржи анализу већ поравнатих секвенци. Ако поравнање није доступно нити се формира у оквиру задатка, фазе које зависе од њега не треба покретати. Корисник треба да има могућност да поравнање покрене засебно или да оно буде аутоматски извршено у оквиру потпуне упоредне анализе. Резултат поравнања треба да се користи као основа за даље откривање разлика, израчунавање мере сличности и филогенетску анализу. Систем треба да очува везу између позиција у изворним и поравнатим секвенцама, као и да приликом тумачења резултата узме у обзир празнине и неодређене нуклеотиде.

На основу поравнања систем треба да детектује нуклеотидне супституције, инсерције и делеције. За сваку откривену варијанту треба одредити њен положај, тип, референтно и алтернативно стање, као и секвенце у којима је присутна. Систем треба да израчуна учесталост варијанти у анализираном скупу и да разликује заједничке, јединствене и варијанте карактеристичне за поједине групе. Резултати треба да буду представљени у облику структуриране табеле која омогућава преглед, сортирање и филтрирање откривених варијанти према њиховим основним карактеристикама.

Систем треба да израчунава меру сличности и генетичка растојања између свих парова секвенци, формира матрицу растојања и одређује најсличније секвенце и секвенце који се највише разликују. На основу резултата поравнања или израчунатих растојања треба изградити филогенетско стабло које приказује претпостављене еволутивне односе између анализираних секвенци. Корисник треба да има могућност да прегледа структуру стабла и да кроз повезане приказе истражује појединачне секвенце и издвојене групе, укључујући откривене геномске варијанте повезане са њима.

Поред тога, систем треба да подржава аутоматско издвајање прелиминарних еволуционих група секвенци. За сваку такву групу треба приказати њен састав, положај на филогенетском стаблу и карактеристичне геномске варијанте. Аутоматски издвојене еволуционе групе треба означити као претпостављене линије, кладе или кластере и не треба их представљати као званично утврђене таксономске категорије. Ако корисник зада сопствену поделу секвенци на групе, систем треба да омогући њено поређење са израчунатим генетичким растојањима и структуром стабла.

Управљање аналитичким задацима

Дуготрајне рачунске операције треба да се извршавају у облику засебних аналитичких задатака. Систем треба да приказује њихово тренутно стање и фазу извршавања, као и да омогући кориснику да прегледа резултате већ завршених међуфаза. Резултати парсирања датотека, валидације, поравнања и других ресурсно захтевних операција треба аутоматски да се чувају након завршетка одговарајуће фазе. У случају прекида аналитичког задатка систем треба да омогући његов наставак од последње успешно завршене фазе, без поновног учитавања података, поновног подешавања параметара и поновног извршавања већ завршених прорачуна.

У веб-верзији корисник треба да покреће анализу слањем захтева за креирање аналитичког задатка. Након слања захтева, задатак треба да буде постављен у ред чекања и покренут када се ослободе рачунарски ресурси сервера. Систем треба да приказује стање задатка, укључујући чекање у реду, извршавање, паузирање, успешно завршавање, завршавање са грешком и отказивање. За задатак који чека треба приказати његов тренутни положај у реду. Ако постоји довољно података, систем може додатно да прикаже процењено време почетка извршавања, при чему таква процена треба јасно да буде означена као приближна.

Корисник треба да има могућност да паузира, настави или откаже сопствени аналитички задатак. Након паузирања систем треба током унапред утврђеног временског периода да омогући наставак задатка без поновног проласка кроз општи ред чекања. Ако корисник не настави задатак до истека тог периода, његови међурезултати треба да буду сачувани, али задатак при каснијем наставку треба поново да буде постављен у ред. У случају прекида услед техничког квара, систем треба да покуша да аутоматски настави из-

вршавање од последње сачуване фазе. Ако тренутни наставак није могућ, задатак треба поново поставити у ред без губитка већ добијених међурезултата.

Приказ резултата и могућности веб-верзије

Резултати аналитичког задатка, чији је састав дефинисан у потпоглављу 3.2.3, треба да буду представљени кроз више међусобно повезаних типова визуализација, као што су табеле, графици и дијаграми, чији састав и параметре корисник може да прилагођава. Промена типа визуализације пружа могућност кориснику да поступно истражује добијене резултате.

Веб-верзија система треба додатно да обезбеди кориснику лични радни простор за чување пројеката и резултата извршених аналитичких задатака. Корисник треба да има могућност да поново отвори раније креиране пројекте и другим корисницима омогући контролисан приступ изабраним пројектима или резултатима. Корисници треба да имају могућност да остављају коментаре и напомене у пројектима и анализама којима имају приступ, у складу са додељеним правима.

Јавни део веб-апликације треба да садржи документацију за инсталацију локалне верзије, упутство за коришћење система, опис подржаних метода и формата података, као и опште информације о пројекту и његовом развоју.

Обавештења

Систем треба да обавештава корисника о догађајима који не захтевају стално праћење тока извршавања аналитичког задатка. У локалној верзији корисник треба да добије обавештење на уређају након успешног завршетка аналитичког задатка или његовог прекида услед грешке. Поред тога, коначно стање операције треба да буде приказано у интерфејсу који се користи, а при раду преко интерфејса командне линије треба да буде исписано у терминалу.

Веб-верзија треба да подржава обавештења унутар апликације, обавештења прегледача, електронску пошту и слање порука путем подржаних апликација за размену порука, укључујући *WhatsApp*, *Viber* и *Telegram*. Обавештења прегледача и спољна обавештења треба слати тек након што корисник одобри њихово примање и подеси изабрани комуникациони канал.

Обавештења у веб-верзији треба да обухвате не само промене стања аналитичких задатака већ и догађаје повезане са заједничким радом, укључујући

позив за сарадњу у пројекту, доделу или измену права приступа, додавање коментара или напомене од стране другог корисника и друге радње повезане са пројектима којима корисник има приступ. Корисник треба да има могућност да засебно подеси канале и категорије обавештења, као и да искључи необавезна обавештења. Обавештења унутар апликације треба да буду сачувана у списку доступном кориснику, уз могућност њиховог прегледа и означавања као прочитаних.

Увоз и извоз података

Систем треба да омогући извоз резултата анализе. У зависности од врсте података, корисник треба да има могућност да сачува нумеричке карактеристике скупа секвенци, табелу откривених геномских варијанти, матрицу растојања, поравнате секвенце, филогенетско стабло и финални извештај. Извоз треба да садржи и податке неопходне за накнадно тумачење резултата и репродуковање спроведене анализе, у складу са спецификацијом излазних података из потпоглавља 3.2.3.

Систем такође треба да омогући извоз и поновни увоз података неопходних за наставак рада са резултатима спроведене анализе, у складу са структуром извезених података дефинисаном у потпоглављу 3.2.3. На тај начин треба омогућити поновну визуелизацију резултата и њихово даље истраживање без обавезног поновног извршавања већ завршених рачунских фаза.

3.2.2 Спецификација критеријума квалитета система

Спецификација критеријума квалитета система обухвата квалитативне карактеристике система *JELICA* и услове извршавања предвиђених функција. Основни критеријуми квалитета система обухватају перформансе, поузданост, репродуцибилност резултата, употребљивост, безбедност, преносивост и проширивост.

Систем треба рационално да користи доступне рачунарске ресурсе, док дуготрајне операције не смеју да блокирају графички интерфејс. Локална верзија треба да узима у обзир карактеристике локалног уређаја, а веб-верзија да распоређује аналитичке задатке у складу са доступним серверским ресурсима. Граничне величине улазних фајлова и дозвољени број секвенци треба да буду одређени на основу резултата тестирања система.

Грешка при обради појединачне датотеке или аналитичког задатка не сме да доведе до губитка других података или нарушавања рада система у целини. Резултати завршених фаза аналитичког задатка треба да се чувају како би се, након отказа или поновног покретања система, аналитички задатак могао наставити од последње успешно завршене фазе. Основне операције, упозорења и грешке треба евидентирати у дневнику рада, уз навођење времена, фазе аналитичког задатка и одговарајућег аналитичког задатка.

Поновно извршавање анализе са истим улазним подацима, параметрима и верзијама коришћених алата треба да даје исте резултате. Уз резултате је потребно чувати параметре аналитичког задатка, податке о изабраној референтној секвенци и верзије коришћених алгоритама. Исправност основних израчунавања треба проверавати помоћу аутоматизованих тестова и контролних скупова података.

Графички интерфејси локалне и веб-верзије треба да имају сличну структуру, пружају информативне поруке о грешкама и приказују стање дуготрајних операција. Интерфејс командне линије треба да користи стабилан машински читљив формат, једнозначне кодове завршетка и структуриране поруке о току извршавања.

Веб-верзија треба да обезбеди заштићен пренос података, аутентификацију корисника, различита права приступа за различите кориснике и изолацију аналитичких задатака између корисника, тако да један корисник не може неовлашћено да приступи задацима другог корисника. Геномски подаци не смеју се прослеђивати спољним сервисима без знања корисника. Локална верзија треба да омогући потпуно извршавање анализе на локалном уређају.

Систем треба да има модуларну структуру која раздваја аналитичко језгро, интерфејс командне линије, графичке апликације и помоћне сервисе. Архитектура треба да омогући додавање нових формата података, метода за поједине фазе анализе и начина представљања резултата без поновне имплементације система. Изворни код система треба да буде објављен у јавно доступном репозиторијуму под отвореном лиценцом, уз примену јединствених правила форматирања и аутоматске провере кода. Јавни програмски интерфејси, основни модули и нетривијални делови имплементације треба да буду праћени ажурном документацијом и детаљним коментарима.

3.2.3 Спецификација формата улазних и излазних података

Систем треба да прихвата једну или више целих или готово целих геномских секвенци у подржаним форматима, укључујући *FASTA*, *GenBank* и једноставан текстуални формат. Улазни подаци могу бити учитани у облику локалних датотека или преузети из базе *GenBank* на основу наведене везе или идентификатора записа. Свакој секвенци треба доделити јединствени идентификатор, наведен у улазним подацима, задат од стране корисника или формиран на основу назива датотеке или *GenBank* идентификатора. Метаподаци секвенци нису обавезни; уколико су доступни, систем треба да их сачува ради приказивања и извоза.

Излазни подаци треба да обухватају резултате валидације, основне нумеричке карактеристике, вишеструко поравнање, табелу уочених разлика у поравнању, матрицу генетичких растојања између секвенци, филогенетско стабло и резултате груписања еволутивно повезаних секвенци, уколико су одговарајуће фазе применљиве на анализирани скуп. Резултати треба да буду представљени у облику разумљивом кориснику, као и у структурираним машински читљивим форматима погодним за даљу обраду и извоз.

За чување и пренос резултата аналитичког задатка систем треба да користи специјализовани формат *.jelica*. Датотека овог формата треба да садржи улазне податке, доступне метаподатке, конфигурацију аналитичког задатка, податке о коришћеним методама и верзијама система, као и добијене резултате неопходне за њихов накнадни преглед, даљу обраду и репродукцију аналитичког задатка без поновног извршавања већ завршених фаза.

3.3 Пројектовање система

3.3.1 Избор архитектуре система

При пројектовању система *JELICA*, основни захтев била је независност аналитичког језгра од корисничког интерфејса, као и могућност његовог коришћења у локалном и серверском окружењу. Алат треба да подржава локални и серверски режим рада, при чему обе варијанте употребе треба да буду засноване на истом аналитичком језгру система.

У складу с тим, изабрана је модуларна архитектура са издвојеним самосталним аналитичким језгром *JELICA Core*. Спољне компоненте комуницирају са њим преко јединственог интерфејса, што омогућава употребу исте имплементације аналитичке функционалности у интерфејсу командне линије, десктоп и веб-апликацији, а уједно поједностављује проширивање система и његову интеграцију са другим алатима и корисничким скриптовима.

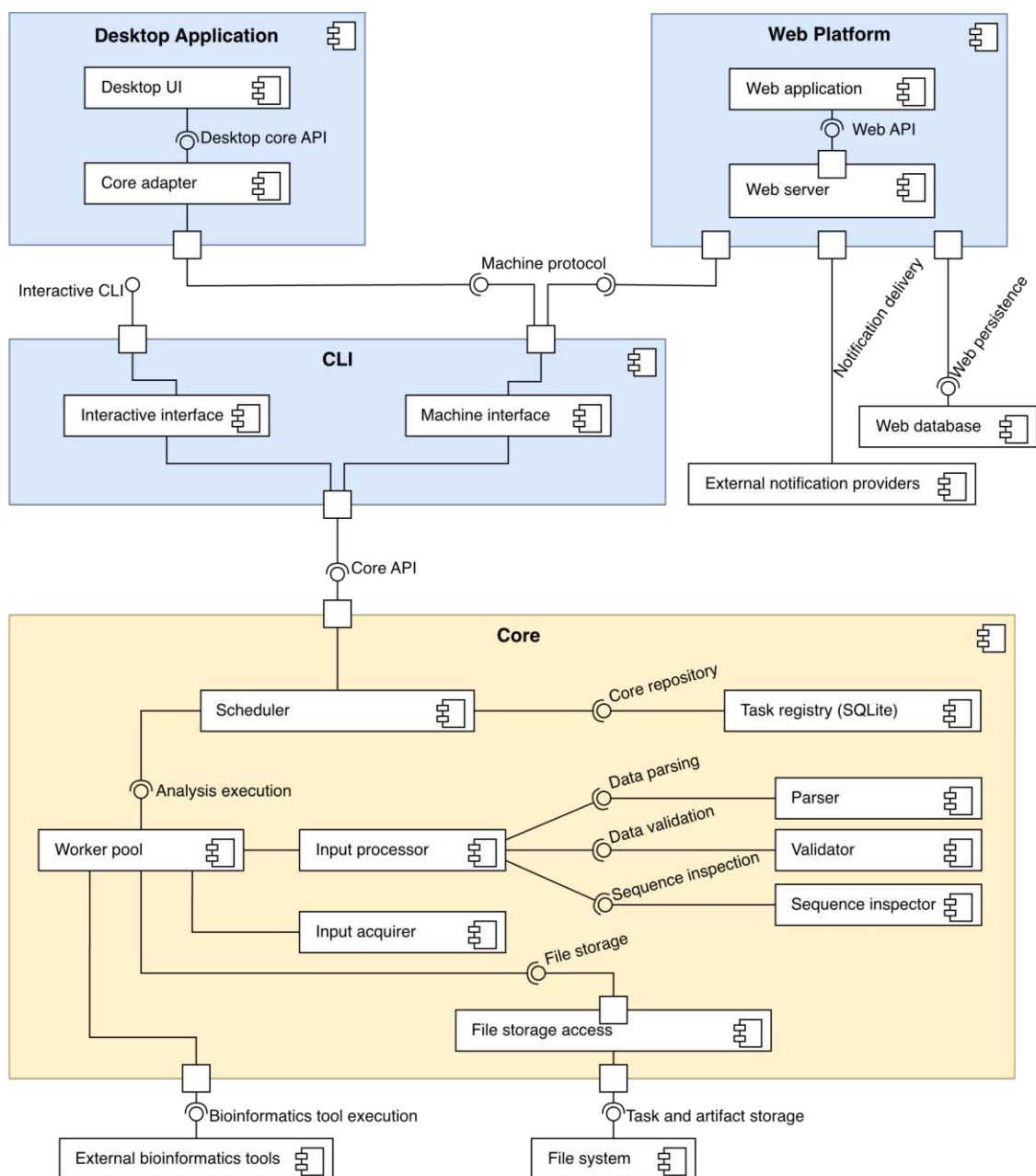
За организацију изворног кода изабран је монорепозиторијум, који поједностављује заједнички развој компоненти, управљање верзијама и дељење модула и компоненти између различитих делова система. Јасна подела система на модуле омогућава даље проширивање система без значајних измена његове опште структуре.

3.3.2 Општа архитектура система

Општа архитектура система *JELICA* обухвата аналитичко језгро (модул *Core*), интерфејс командне линије (модул *CLI*), десктоп апликацију (модул *Desktop Application*) и веб-платформу (модул *Web Platform*). *Core* обједињује интерне модуле задужене за припрему података, управљање задацима, извршавање аналитичког задатка и чување резултата, док спољашње компоненте обезбеђују различите начине интеракције са системом. Општа архитектура програмског алата *JELICA* и међусобни односи његових главних компоненти приказани су на *UML* дијаграму компоненти 3.2.

Управљање распоређивањем аналитичких задатака у оквиру модула *Core* обавља компонента *Scheduler*. Она организује ред задатака, примењује правила за одређивање приоритета и додељује задатке расположивим радним извршиоцима, који су у наставку детаљније описани у оквиру компоненте *Worker pool*. Редоследом аналитичких фаза у оквиру конкретног покретања задатка управља оркестратор (*Orchestrator*), који је детаљније описан у потпоглављу 3.5.2. Модул *Core* пружа јединствени програмски интерфејс преко којег спољашњи интерфејси користе функције аналитичког језгра.

Пријем и материјализацију улазних података обавља компонента *Input acquirer*. Под материјализацијом се подразумева претварање улазних података из подржаних формата у јединствени облик за даљу обраду, при чему се свака секвенца издваја у засебну датотеку у формату *FASTA* и смешта у радни простор аналитичког задатка. Компонента препознаје различите типове улазних података, преузима податке са веб локација и обрађује локалне



Слика 3.2: Дијаграм компоненти програмског система JELICA

датотеке и директоријуме. Након тога компонента ***Input processor*** координише парсирање, инспекцију и валидацију секвенци. Компонента ***Parser*** претвара записе из подржаних формата у нормализовану интерну репрезентацију. Под **нормализованом репрезентацијом секвенце** подразумева се јединствен облик за даљу обраду, независан од начина уноса и изворног формата. Његову основу чини засебан запис секвенце у формату *FASTA*, којем се могу придружити изворни идентификатори и пратећи метаподаци када су доступни. Компонента ***Sequence inspector*** обавља инспекцију тако што у једном пролазу анализира нормализоване секвенце и прикупља њихове основне нумеричке карактеристике. За разлику од инспекције, валидација се односи на проверу исправности података: компонента ***Validator*** на основу резултата парсирања и инспекције проверава исправност појединачних записа и погодност целог скупа за даљу анализу. ***Input processor*** такође управља дедупликацијом секвенци, одређивањем референтне секвенце и формирањем међурезултата за наредне фазе обраде.

Извршавање аналитичких задатака обавља скуп радних извршилаца (компонента ***Worker pool***), којима планер додељује задатке за извршавање. Сваки радни извршилац обрађује додељени аналитички задатак проласком кроз његове дефинисане аналитичке фазе, користећи како алгоритме реализоване у оквиру система *JELICA*, тако и специјализоване спољашње биоинформатичке алате (нпр. *MAFFT*). Ако су на располагању довољни рачунарски ресурси, више радних извршилаца може паралелно извршавати различите задатке, чиме се повећава укупна ефикасност система.

Core обухвата регистар аналитичких задатака заснован на бази *SQLite* и компоненту за приступ фајл систему. У бази података чувају се структурирани подаци о задацима, њиховим стањима, приоритетима, времену извршавања, грешкама и путањама до припадајућих конфигурација и резултата. Ревизије конфигурација, материјализовани улазни подаци, **манифести** фаза (описне датотеке које бележе статус извршавања фазе, настале датотеке и контролне суме), резултати поравнања, формирани резултати и други међурезултати чувају се у фајл систему.

Интеракција са аналитичким језгром остварује се преко интерфејса командне линије (***CLI***), који представља главну приступну тачку за *Core*. *CLI* је пре свега намењен машинској интеракцији и обезбеђује стабилан програмски интерфејс за спољашње апликације, али подржава и непосредан

рад корисника у интерактивном режиму. Десктоп апликација (*Desktop Application*) представља графичку надоградњу над *CLI*-јем. Веб-апликација (*Web Application*) такође комуницира са аналитичким језгром преко *CLI*-ја, али се та интеракција одвија на серверској страни: серверска инфраструктура претвара захтеве веб-интерфејса у команде машинског режима *CLI*-ја и прослеђује добијене резултате назад клијентској апликацији. На тај начин су све аналитичке функционалности имплементиране у оквиру *Core* и не дуплирају се у различитим варијантама испоруке система.

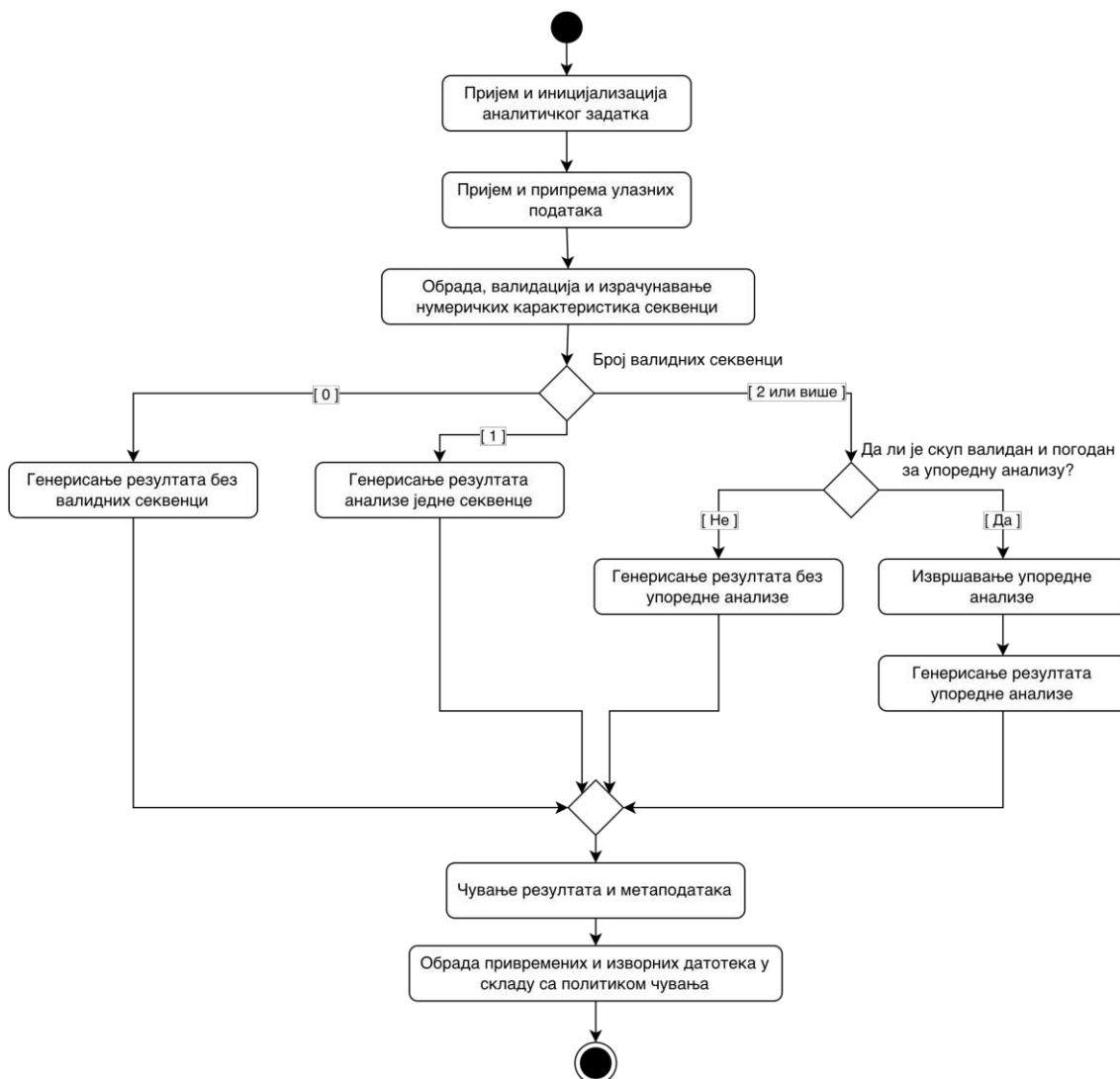
3.3.3 Пројектовање тока аналитичког задатка

Ток аналитичког задатка у систему *JELICA* представља низ фаза обраде података које се извршавају под управљањем оркестратора. Његов састав се одређује на основу карактеристика улазних података и изабраних параметара анализе, чиме се омогућава аутоматско изостављање фаза које немају смисла за конкретан скуп података. Основне фазе аналитичког задатка и избор одговарајућег сценарија обраде у зависности од броја валидних секвенци и резултата валидације приказани су на слици 3.3.

Након пријема задатка формира се његов радни простор, а улазни извори се препознају и материјализују за даљу обраду. Припрема улазних података затим се извршава посредством компоненти описаних у потпоглављу 3.3.2, кроз парсирање, инспекцију и валидацију секвенци. На основу броја валидних секвенци, резултата валидације и конфигурације задатка настављају се само фазе применљиве на конкретан скуп података. На пример, за једну валидну секвенцу задржавају се резултати валидације и анализе нумеричких карактеристика, док се фазе упоредне анализе не покрећу.

Аналитичке задатке извршавају радни извршиоци компоненте *Worker pool*, којима задатке додељује компонента *Scheduler*, описана у потпоглављу 3.3.2. У оквиру конкретног покретања оркестратор управља редоследом фаза аналитичког задатка, прати њихово стање, прослеђује информације о напретку корисничким интерфејсима и обрађује управљачке наредбе за паузирање, настављање и отказивање задатка.

Након завршетка тока аналитичког задатка чувају се добијени резултати и припадајући метаподаци, а завршни резултати обједињују се у преносиву датотеку формата *.jelica*, чија је структура описана у потпоглављу 3.3.4. Привремене и изворне датотеке обрађују се у складу са изабраном политиком чувања.



Слика 3.3: Дијаграм извршавања аналитичког задатка у систему JELICA.

3.3.4 Конфигурације и модел резултата

Ради обезбеђивања уједначене размене података између компоненти система дефинисане су стандардизоване конфигурације и модел резултата. Њима се описују подешавања система, параметри аналитичких задатака и структура завршног извештаја. Независно од корисничког интерфејса преко којег се креира аналитички задатак, улазни подаци и параметри преводе се у јединствену интерну репрезентацију коју користи аналитичко језгро.

Системска конфигурација

Системска конфигурација обухвата скуп глобалних параметара који одређују окружење у коме аналитичко језгро и остале компоненте система функционишу. За разлику од конфигурације аналитичког задатка, која описује параметре једног конкретног покретања аналитичког задатка, системска конфигурација садржи подешавања која важе за све аналитичке задатке док их корисник не измени.

У системској конфигурацији чувају се параметри који одређују локације радних директоријума, подразумеване вредности параметара који се користе при извршавању аналитичких задатака када нису експлицитно задате у конфигурацији појединачног аналитичког задатка, ограничења коришћења системских ресурса, као и друга глобална подешавања неопходна за рад система. Системска конфигурација такође садржи глобалне техничке параметре који, заједно са нормализованом конфигурацијом аналитичког задатка, учествују у формирању модела резултата. Централизовано чување ових параметара омогућава њихово доследно коришћење у свим компонентама система и поједностављује администрацију и измену конфигурације.

За чување системске конфигурације изабран је формат *TOML*, јер омогућава једноставно организовање глобалних параметара у прегледну хијерархијску структуру, лак је за ручно уређивање и добро је подржан у програмском језику *Python*. Оваква организација поједностављује одржавање конфигурације и њено коришћење у свим компонентама система.

При покретању система проверава се постојање системске конфигурационе датотеке, која је неопходна за рад основних функционалности система. Уколико конфигурација није пронађена, систем захтева њено експлицитно навођење како би се омогућило даље извршавање. Системска конфигурација може бити иницирана и мењана путем групе команди `jelica config`, или ручно, директним уређивањем конфигурационе датотеке.

Конфигурација аналитичког задатка

Конфигурација аналитичког задатка представља потпун опис једног аналитичког задатка и дефинише све параметре неопходне за његово извршавање. За разлику од системске конфигурације, која садржи глобална подешавања окружења за извршавање, конфигурација аналитичког задатка одре-

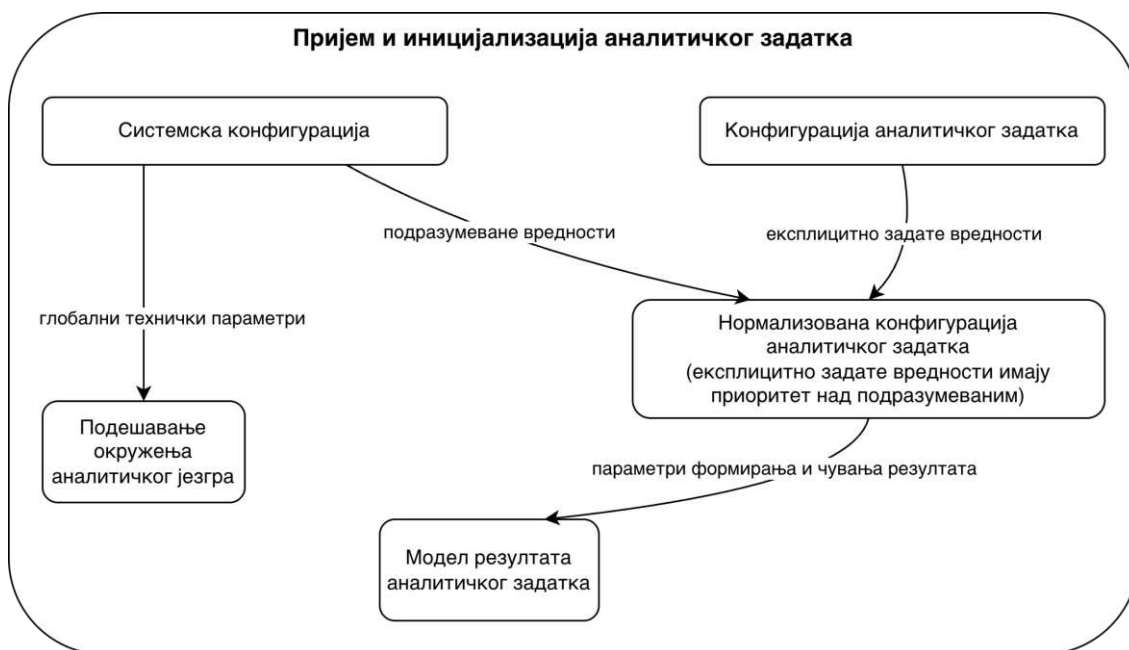
ђује који подаци треба да буду анализирани, које кораке аналитичког задатка је потребно извршити, са којим параметрима их треба покренути, као и на који начин треба формирати завршни извештај, укључујући његову структуру, садржај приказаних резултата и формат приказа. На тај начин, једна конфигурација у потпуности одређује ток извршавања аналитичког задатка, од учитавања улазних података до формирања завршног извештаја.

Конфигурација аналитичког задатка садржи информације о изворима улазних података, списак анализираних секвенци, референтну секвенцу (уколико се користи), параметре за валидацију и претпроцесирање података, параметре за израчунавање нумеричких карактеристика, упоредну и филогенетску анализу, као и параметре за формирање завршног извештаја. Оваква организација омогућава да се све информације неопходне за репродукцију анализе чувају у једној конфигурационој датотеци, независно од начина покретања система. Конфигурација аналитичког задатка може бити формирана ручним уређивањем конфигурационе датотеке или аутоматски, коришћењем једног од корисничких интерфејса система: интерфејса командне линије, веб-апликације или десктоп апликације.

За чување конфигурације аналитичког задатка користи се формат *JSON*. Његов избор је условљен широком подршком у савременим програмским језицима, веб-технологијама и библиотекама за серијализацију података. Захваљујући томе, иста конфигурација може бити коришћена од стране интерфејса командне линије, десктоп апликације и веб-интерфејса, док аналитичко језгро увек добија јединствену интерну репрезентацију аналитичког задатка, независно од начина његовог креирања.

Коришћење јединствене конфигурације аналитичког задатка омогућава репродуковање спроведене анализе, поједностављује размену конфигурација аналитичких задатака између различитих компоненти система и омогућава поновно извршавање претходно припремљених аналитичких сценарија без потребе за поновним подешавањем параметара.

Однос системске конфигурације и конфигурације аналитичког задатка при формирању нормализоване конфигурације аналитичког задатка приказан је на слици 3.4. Приказани поступак представља детаљнију разраду пријема и иницијализације аналитичког задатка, приказаног као почетна фаза на дијаграму извршавања аналитичког задатка на слици 3.3.



Слика 3.4: Детаљни приказ пријема и иницијализације аналитичког задатка: формирање нормализоване конфигурације аналитичког задатка и модела резултата.

Модел резултата и формат .jelica

Модел резултата представља структуриран машински читљив приказ стања и резултата извршавања аналитичког задатка, укључујући грешке и упозорења откривена током валидације. Овај модел не зависи од конкретног начина приказа и могу га користити аналитичко језгро, интерфејс командне линије, веб-интерфејс и десктоп апликација. На основу њега могу се формирати извештаји у форматима *PDF* и *HTML*, као и интерактивни прикази резултата у верзијама система са графичким корисничким интерфејсом.

За комплетно чување резултата једног успешно завршеног аналитичког задатка предвиђен је специјализовани преносиви формат са екстензијом *.jelica*. Он обједињује модел резултата са улазним подацима, конфигурацијом аналитичког задатка и међурезултатима аналитичких фаза неопходним за накнадни преглед и даљу обраду резултата. Посебна екстензија омогућава једнозначно препознавање датотека система *JELICA*, док дефинисана структура формата омогућава њихову униформну обраду у различитим компонентама система.

Модел резултата садржи идентификатор аналитичког задатка, верзију

формата резултата, податке о верзији система, опис анализираних секвенци и референтне секвенце, параметре анализе примењене у конкретном аналитичком задатку, резултате провере улазних података, укључујући откривене грешке и упозорења, резултате анализе нумеричких карактеристика, упоредне и филогенетске анализе, као и податке о формираним излазним датотекама. Структура и садржај модела резултата одређују се системском конфигурацијом, нормализованом конфигурацијом аналитичког задатка, прослеђеним улазним подацима и корацима анализе извршеним у конкретном задатку.

Датотеке `.jelica` могу се поново учитати у систем ради провере, прегледа и формирања извештаја у траженом формату без поновног извршавања аналитичког задатка. Оваква организација раздваја чување резултата аналитичког задатка од њиховог приказа и омогућава коришћење истог модела резултата у различитим корисничким интерфејсима и форматима за извоз.

3.3.5 Видови коришћења система

Архитектура система *JELICA* предвиђа више видова коришћења који се ослањају на заједничко аналитичко језгро *JELICA Core*.

Основну варијанту представља локално коришћење преко интерфејса командне линије, намењеног пре свега аутоматизованом покретању, пакетној обради података (енгл. *batch processing*) и интеграцији у спољне радне токове. За интерактиван рад корисника обезбеђена је десктоп апликација са графичким интерфејсом, која представља надградњу над локалним компонентама система.

Поред тога, *JELICA* је доступна у облику веб-верзије у облаку са клијент-сервер архитектуром. Аналитички задаци се у њој извршавају на серверској страни, док се интеракција са системом одвија преко веб-прегледача.

3.4 Избор технологија и развојних алата

Технологије за развој система *JELICA* изабране су узимајући у обзир потребу за поновном употребом функционалности аналитичког језгра у различитим варијантама испоруке, подршку за више оперативних система и могућност постепеног проширивања система. Додатни критеријуми били су постојање развијеног екосистема за биоинформатичка израчунавања, погод-

ност за аутоматизацију, једноставност тестирања и могућност брзе припреме функционалне верзије алата.

3.4.1 Аналитичко језгро

Аналитичко језгро *JELICA Core* реализује се на програмском језику *Python*, који се широко примењује у биоинформатици и располаже великим бројем библиотека за обраду биолошких секвенци и научна израчунавања. За учитавање и трансформацију секвенци користи се библиотека *Biopython* [3], док се специјализоване операције могу извршавати посредством интеграције са спољним алатима. Аналитичко језгро организовано је као самосталан скуп *Python* модула и не зависи од корисничког интерфејса нити од начина инсталације система.

3.4.2 Серверски део и API

Серверски део верзије у облаку развијен је на програмском језику *Python* уз коришћење радног оквира *FastAPI* [8]. Он пружа могућност за креирање *HTTP API*-ја, проверу улазних података и аутоматско генерисање документације у формату *OpenAPI*. Опис и валидација структура података обављају се помоћу *Pydantic* [18] модела.

Серверска апликација одговорна је за пријем захтева, управљање аналитичким задацима и достављање резултата, док се дуготрајне аналитичке операције извршавају у засебним радним извршиоцима који користе *JELICA Core*. Захваљујући томе, извршавање аналитичког задатка не блокира обраду осталих захтева. За чување корисника, параметара задатака, стања и метаподатака резултата користи се *PostgreSQL* [17], док се велике улазне и излазне датотеке смештају у складиште сервера.

3.4.3 Веб-интерфејс

Веб-интерфејс реализује се на језику *TypeScript* [14] уз коришћење библиотеке *React* [20] и радног оквира *Next.js* [23]. *TypeScript* омогућава статичку проверу типова и смањује вероватноћу грешака при раду са сложеним структурама резултата аналитичког задатка. *React* се користи за изградњу интерактивних елемената интерфејса, док *Next.js* обезбеђује организацију апликације, рутирање, изградњу и припрему верзије у облаку за инсталацију. Поред

тога, избор технологија *React* и *TypeScript* поједностављује каснији развој десктоп апликације засноване на оквиру *Electron* [6], захваљујући могућности поновне употребе значајног дела клијентског кода.

Веб-апликација комуницира са серверским делом преко *HTTP API*-ја и *WebSocket* веза и не имплементира сопствене аналитичке функционалности. Њени задаци ограничени су на учитавање података, подешавање параметара, приказивање статуса извршавања у реалном времену и визуелизацију добијених резултата.

3.4.4 Десктоп апликација и *CLI*

Интерфејс командне линије реализује се на програмском језику *Python* уз коришћење библиотек *Typer* [22]. Његов основни сценарио употребе представља коришћење из командне линије: покретање анализе из скрипти, аутоматизованих процеса и других програмских компоненти. Истовремено, *CLI* пружа основне команде за непосредну интеракцију корисника са локалном верзијом алата.

Десктоп апликација развија се уз коришћење технологија *TypeScript*, *React* и *Electron*. Примена *Electron*-а омогућава подршку за главне десктоп оперативне системе и поновну употребу технологија и дела компоненти веб-интерфејса. Десктоп апликација представља графичку надоградњу локалног *CLI*-ја и компоненте *JELICA Core*, због чега се алгоритми већ имплементирани у аналитичком језгру у њој не имплементирају поново. За чување корисничких подешавања интерфејса и других мањих локалних података користи се уграђена база података *SQLite*.

3.4.5 Изградња, тестирање и испорука система

Python зависностима и конфигурацијом пакета управља се помоћу алата *uv* и датотеке *pyproject.toml*, док се зависностима *TypeScript* компоненти управља помоћу алата *npm*.

За тестирање *Python* компоненти, како појединачно тако и интегрално, користи се *pytest* [19]. Компоненте корисничког интерфејса тестирају се помоћу алата *Vitest* [24], док се основни кориснички сценарији веб-апликације тестирају помоћу алата *Cypress* [4]. Аутоматизоване провере, тестирање и изградња извршавају се у оквиру *GitHub Actions*-а.

Серверске компоненте и неопходна инфраструктура испоручују се у *Docker* [5] контејнерима, чиме се обезбеђује репродуцибилна инсталација у окружењу у облаку. Десктоп апликација пакује се у инсталационе пакете за подржане оперативне системе, док се *CLI* може дистрибуирати као *Python* пакет или као самосталан извршни модул.

3.4.6 Развојно окружење и примена вештачке интелигенције

Систем *JELICA* развијан је у окружењу *VS Code*, уз коришћење екстензија за примењене програмске језике и технологије. За управљање верзијама изворног кода користи се *Git*, док је јавни репозиторијум пројекта постављен на платформи *GitHub*. Ручна провера и отклањање грешака у корисничким интерфејсима спроводе се у прегледачима *Chrome*, *Safari*, *Firefox*, *Edge* и *Opera*, а за проверу рада система у различитим оперативним окружењима користи се *Parallels Desktop*.

Као помоћно средство при развоју користи се *GitHub Copilot*, интегрисан у *VS Code*, са моделима из породица *Claude* и *GPT-Codex*. Агенти засновани на вештачкој интелигенцији примењују се за убрзавање писања и измене програмског кода, проналажење грешака и предлагање варијанти имплементације. Сви добијени резултати накнадно се ручно проверавају, покрећу и тестирају, док је агентима онемогућен приступ *Git* операцијама и самостално евидентирање измена. *GitHub Copilot* се такође користи за аутоматизовани прелиминарни преглед измена на платформи *GitHub*, а веб-верзија алата *ChatGPT* као помоћ при пројектовању и провери архитектонских решења. Коначни избор решења и одговорност за њихову исправност остају на аутору система.

3.5 Имплементација система JELICA

У овом потпоглављу описана је имплементација архитектонских решења представљених у потпоглављу 3.3, од организације изворног кода и управљања аналитичким задацима до имплементације фаза аналитичког задатка, механизма за чување резултата и корисничких интерфејса.

3.5.1 Организација имплементације система

Имплементација система обухвата аналитичко језгро, интерфејс командне линије, веб-платформу и десктоп апликацију, као и заједничке моделе података, конфигурацију радног простора и аутоматизоване тестове. Компоненте написане на програмском језику *Python* обједињене су у заједнички развојни радни простор, док су веб-платформа и десктоп апликација организационо раздвојене од аналитичке имплементације.

Централни део имплементације представља пакет *jelica-core*. Његове јавне операције користе интерфејс командне линије, серверски део веб-платформе и интеграциони слој десктоп апликације. Сваки интерфејс самостално претвара кориснички унос у одговарајући захтев, а добијене резултате анализе у облик погодан за даљу употребу у одговарајућем интерфејсу.

За размену података између компоненти користе се заједнички модели који дефинишу структуру конфигурација, аналитичких задатака и резултата, описани у потпоглављу 3.3.4.

Коректност имплементације проверава се аутоматизованим тестовима који обухватају кључне модуле система, као и њихово заједничко функционисање.

3.5.2 Модел аналитичког задатка и управљање извршавањем

У зависности од конфигурације, аналитички задаци могу да обухвате анализу нумеричких карактеристика једне секвенце, али и потпуну упоредну анализу скупа секвенци, као што је описано у потпоглављу 3.2.1.

Модел извршавања аналитичког задатка, чије су основне компоненте и њихове улоге описане у потпоглављу 3.3.2, имплементиран је у пакету *jelica-core*. Језгро пружа јединствен скуп операција за креирање, покретање, паузирање, настављање и отказивање задатка, као и за добијање његовог тренутног стања.

Приликом креирања аналитичког задатка конфигурација описана у потпоглављу 3.3.4 проверава се и обједињује са параметрима системске конфигурације применљивим за овај задатак, након чега језгро формира нормализовану конфигурацију аналитичког задатка са параметрима који се стварно користе. Након успешне провере вредности параметара и формирања нормализоване конфигурације, задатку се додељује јединствени идентификатор, креира се

засебан радни директоријум, а конфигурација се у њему чува као датотека у формату *JSON*. Уколико иницијализација задатка не може да буде завршена, подаци настали током њеног извршавања се уклањају, чиме се спречава креирање непотпуних аналитичких задатака.

Аналитички задатак представља трајни ентитет који повезује конфигурацију захтеваног задатка, његово стање према моделу из табеле 3.1 и податке о извршавању. Интерни метаподаци о аналитичким задацима, као што су идентификатор задатка, тренутно стање, путања до радног директоријума и подаци о извршавању, чувају се у регистру задатака, у бази података креираној помоћу система за управљање базама података *SQLite*. На тај начин омогућавају се очување стања и управљање задацима након поновног покретања система.

Табела 3.1: Стања аналитичког задатка

Стање	Тип	Значење
<code>waiting</code>	Основно	Задатак је креиран и чека покретање.
<code>queued</code>	Основно	Задатак је у реду и чека доступног радног извршиоца.
<code>running</code>	Основно	Задатак се активно извршава.
<code>pause_requested</code>	Прелазно	Примљен је захтев за контролисано паузирање.
<code>preemption_requested</code>	Прелазно	Задатак треба да уступи ресурс задатку вишег приоритета.
<code>paused</code>	Основно	Извршавање је паузирано и може бити настављено.
<code>cancel_requested</code>	Прелазно	Примљен је захтев за контролисано отказивање.
<code>completed</code>	Завршно	Задатак је успешно завршен.
<code>failed</code>	Завршно	Извршавање је завршено због грешке.
<code>cancelled</code>	Завршно	Извршавање је отказано пре завршетка.
<code>deletion_requested</code>	Прелазно	Примљен је захтев за контролисано брисање задатка.

Креирани задаци смештају се у ред за извршавање. Планер (*Scheduler*) бира задатке који чекају на основу њиховог приоритета, редоследа пристизања и броја расположивих радних извршилаца компоненте *Worker pool*. Пре покретања задатка, планер у оквиру једне трансакције мења стање изабраног задатка, чиме се спречава да исти аналитички задатак буде истовремено додељен већем броју радних извршилаца. Када су сви расположиви радни

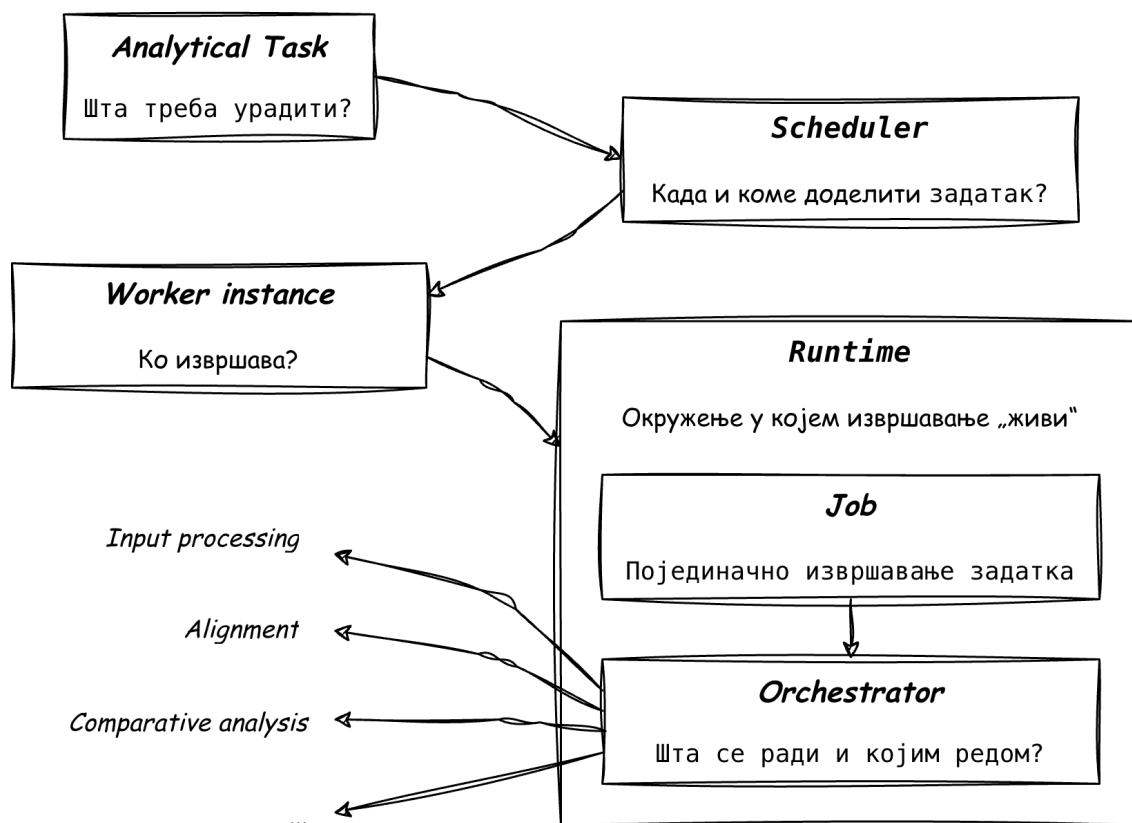
извршиоци заузети, задатак остаје у реду све док се не ослободи неки од њих или док, у складу са правилима приоритета, активни задатак нижег приоритета контролисано не уступи свог радног извршиоца.

За свако покретање или настављање аналитичког задатка формира се засебно окружење за извршавање (*Runtime*). Ово окружење постоји само током текуће обраде и повезује податке који се трајно чувају у регистру задатака и радном директоријуму задатка са привременим подацима неопходним за извршавање. У њега спадају, на пример, учитана нормализована конфигурација, подаци о текућој фази аналитичког задатка, информације о напретку, привремени подаци који настају током појединачних фаза и ознаке команди за управљање извршавањем, којима се тражи паузирање, настављање или отказивање задатка. Приликом настављања раније паузираног задатка или поновног покретања задатка који је претходно отказан или завршен грешком формира се ново окружење за извршавање, али трајни идентификатор аналитичког задатка остаје непромењен.

Ток извршавања аналитичког задатка координише оркестратор (*Orchestrator*). Приликом покретања конкретног аналитичког задатка формира се објекат текућег извршавања (*Job*), који планер додељује једном од расположивих радних извршилаца компоненте *Worker pool* (*Worker instance*). Додељени радни извршилац затим редом извршава фазе аналитичког задатка које су предвиђене његовом конфигурацијом и применљиве на улазни скуп. При настављању аналитичког задатка оркестратор проверава међурезултате и манифесте претходних фаза у радном директоријуму задатка и не понавља фазе чији су резултати већ успешно формиран и евидентирани. Однос између основних елемената који учествују у извршавању аналитичког задатка приказан је на слици 3.5.

Команде за управљање извршавањем аналитичког задатка, као што су паузирање, настављање и отказивање, обрађују се централизовано, док се прелази између стања задатка спроводе тек након провере његовог тренутног стања. На тај начин спречавају се недозвољене операције над активним или завршеним задацима.

При паузирању, отказивању или техничком прекиду аналитичког задатка чувају се резултати свих у потпуности завршених фаза. То омогућава наставак аналитичког задатка без понављања рачунски захтевних операција које су већ успешно извршене.



Слика 3.5: Модел извршавања аналитичког задатка у систему JELICA

3.5.3 Имплементација тока аналитичког задатка

Аналитички ток пројектован у потпоглављу 3.3.3 имплементиран је као скуп међусобно повезаних фаза са дефинисаним улазним подацима, конфигурацијом и резултатима које формира свака фаза. Оркестратор на основу нормализоване конфигурације аналитичког задатка одређује које се фазе извршавају у конкретном покретању.

Прва фаза, *input_acquisition*, задужена је за преузимање улазних података. Извори могу бити локалне датотеке и директоријуми, као и записи из базе *NCBI GenBank* који су задати приступним бројем или *URL* адресом. Након тога фаза *input_processing* проверава формат и валидност секвенци и преводи их у нормализовану интерну репрезентацију (како је објашњено у потпоглављу 3.2.2), која се користи у наставку обраде. У истој фази израчунавају се основне нумеричке карактеристике секвенци, укључујући дужину секвенце, садржај неодређених нуклеотида, расподелу симбола, *GC* садржај и, када је задато конфигурацијом, резултате претраге *k*-мера. Примери изра-

чунатих нумеричких карактеристика и резултата претраге k -мера приказани су на слици 3.6.

```

570 {
571   "facts": {
572     "ambiguous_count": 381,
573     "base_counts": {
574       "definite": {
575         "A": 8820,
576         "C": 5392,
577         "G": 5768,
578         "T": 9482,
579         "U": 0
580       },
581       "potential": {
582         "A": 9198,
583         "C": 5773,
584         "G": 6146,
585         "T": 9863,
586         "U": 381
587       }
588     },
589     "canonical_count": 29462,
590     "expected_gc_content": 0.3803404483463459,
591     "expected_gc_count": 11350.5,
592     "gap_count": 0,
593     "gc_content_total": 0.37395704185236067,
594     "gc_count": 11160,
595     "invalid_positions": [],
596     "invalid_positions_truncated": false,
597     "invalid_symbol_count": 0,
598     "invalid_symbol_counts": {},
599     "kmer_summaries": [
600       {
601         "definite_match_count": 131,
602         "hits_path": "input_processing/kmer_hits/
603           d121c577270129df1d5742ed3b1e876957ed42a424b543f0
604           d5761d2b71d089f5.json",
605         "possible_match_count": 368,
606         "query": "ATTC",
607         "strand": "forward"
608       }
609     ]
610   }
611 }

```

```

> input_processing > kmer_hits > {} d121c577270129df1d5742ed3b1e876957e
1 {
2   "alignment_mode": "compute",
3   "coordinate_system": "zero_based_end_exclusive",
4   "queries": [
5     {
6       "coordinate_system": "zero_based_end_exclusive",
7       "query": "ATTC",
8       "strand": "forward",
9       "hits": [
10        {
11          "alignment_range": null,
12          "coordinate_system": "zero_based_end_exclusive",
13          "match_kind": "definite",
14          "sequence_range": {
15            "end": 547,
16            "start": 543
17          },
18          "strand": "+"
19        },
20        {
21          "alignment_range": null,
22          "coordinate_system": "zero_based_end_exclusive",
23          "match_kind": "definite",
24          "sequence_range": {
25            "end": 1097,
26            "start": 1093
27          },
28          "strand": "+"
29        },
30        {
31          "alignment_range": null,
32          "coordinate_system": "zero_based_end_exclusive",
33          "match_kind": "definite",
34          "sequence_range": {
35            "end": 1114,
36            "start": 1110
37          },
38          "strand": "+"
39        }
40      ]
41    }
42  ]
43 }

```

(а) Фрагмент нумеричких карактеристика секвенце (б) Фрагмент резултата претраге k -мера

Слика 3.6: Пример резултата фазе `input_processing`

Начин обраде поравнања одређен је конфигурацијом аналитичког задатка. У режиму *compute* систем формира вишеструко поравнање коришћењем алата *MAFFT*, при чему се задата референтна секвенца укључује у поравнање и користи као основа за одређивање позиција у наредним фазама анализе, тако да се њихове координате изражавају у односу на референтну секвенцу. У режиму *prealigned* користе се већ поравнате улазне секвенце, чија се компатибилност претходно проверава; уколико је референтна секвенца задата, она се и у овом режиму користи у фазама које захтевају референтни координатни систем. У режиму *none* фаза поравнања се не извршава, па се изостављају и наредне аналитичке фазе којима су поравнате секвенце неопходне као улаз.

Над формираним поравнањем извршава се фаза `comparative_analysis`, која обухвата израчунавање нумеричких карактеристика и упоредну анализу

изабраних парова секвенци. Поређења се могу извршавати за цео скуп, унутар група секвенци задатих од стране корисника или за експлицитно задате парове. Добијени резултати користе се за визуелизацију и формирање извештаја.

Фаза `distance_matrix` из поравнатих секвенци формира квадратну матрицу генетичких растојања. Добијене вредности представљају основу за процену међусобне сличности секвенци и за каснију реконструкцију њихових еволуционих односа. Матрица се чува у машински читљивим форматима, што омогућава њен непосредан приказ, извоз или даљу обраду без поновног израчунавања претходних фаза.

На основу матрице растојања фаза `phylogenetic_tree` формира филогенетско стабло применом методе *Neighbor Joining*. Резултат се чува у стандардном формату *Newick*², као и у структурираном облику погодном за визуелизацију у графичким интерфејсима. Имплементација обезбеђује детерминистичко извршавање аналитичког задатка, тако да се при истим улазним подацима и конфигурацији добија исти резултат.

Након формирања матрице растојања могуће је извршити и фазу `clade_detection`, у којој се секвенце групишу према задатом максималном растојању унутар групе. При томе се користе вредности из матрице генетичких растојања, у којој се растојање између две секвенце рачуна као однос броја позиција на којима се поравнате секвенце разликују и укупног броја позиција које се у том поређењу узимају у обзир.

3.5.4 Чување стања, међурезултата и резултата аналитичког задатка

Чување података у систему *JELICA* реализовано је применом хибридног модела. Метаподаци о аналитичким задацима чувају се у бази података *SQLite*, док се конфигурације, улазни подаци и резултати њиховог извршавања смештају у фајл систем. За сваки задатак формира се засебан радни директоријум који садржи његову нормализовану конфигурацију, податке и резултате појединачних аналитичких фаза. Пример фрагмента нормализоване конфигурације аналитичког задатка приказан је на слици 3.7а.

²*Newick* је текстуални формат за представљање структуре филогенетског стабла помоћу угњеждених заграда, назива чворова и, по потреби, дужина грана.

Резултати завршених фаза објављују се тек након успешног формирања свих обавезних међурезултата и манифеста. На тај начин међурезултати остају раздвојени од завршних и систем може да настави прекинути аналитички задатак од последње успешно завршене фазе.

Завршни резултати аналитичког задатка обједињују се у датотеку формата `.jelica`, интерно реализовану као компримована архивска структура. У наставку се ова датотека означава као **пакет са резултатима аналитичког задатка**, јер поред самих резултата садржи и пратеће податке неопходне за њихово поновно учитавање и приказ. Пакет садржи нормализоване улазне секвенце, нормализовану конфигурацију задатка, резултате аналитичких фаза, њихове манифесте, дијагностичке податке и контролне суме за проверу интегритета. Као једини кориснички изменљив додатак дозвољена је датотека `./NOTES.txt`. Пример структуре садржаја формираног пакета приказан је на слици 3.7b. Формирани и увезени пакети са резултатима аналитичког задатка организују се у директоријуму `result_packages`.

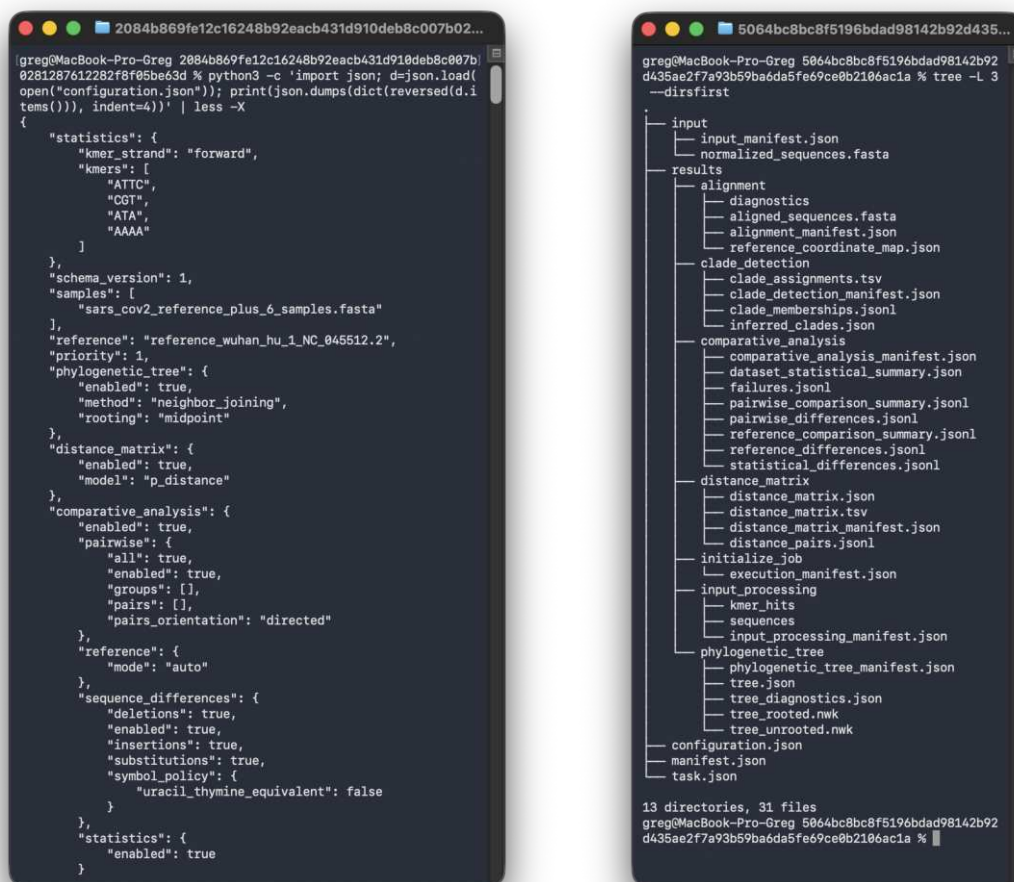
На основу структурираног садржаја пакета формирају се *HTML* и *PDF* извештаји, интерактивне визуелизације и други формати датотека за извоз резултата, без измене садржаја самог пакета. Исти пакет се зато може користити у интерфејсу командне линије, веб-платформи и десктоп апликацији.

3.5.5 Поузданост, опоравак и праћење тока извршавања

Захтеви за поузданост, опоравак и праћење тока извршавања дефинисани у потпоглављу 3.2.2 имплементирани су кроз проверу конфигурације и улазних података, контролу дозвољених прелаза између стања аналитичких задатака и трансакционо ажурирање регистра. Проверавају се и резултати појединачних аналитичких фаза, тако да наредна фаза не користи непотпуне или неисправно формиране податке.

При неуспешном креирању задатка привремени подаци се уклањају, док при прекиду започетог аналитичког задатка резултати успешно завршених фаза остају сачувани и могу се користити при настављању. Грешке су изоловане у оквиру појединачног задатка и не прекидају извршавање других задатака.

Грешке, упозорења и други значајни догађаји бележе се у структурираном



(a) Фрагмент нормализоване конфигурације (b) Структура садржаја пакета резултата аналитичког задатка формата .jelica

Слика 3.7: Чување конфигурације аналитичког задатка и структура пакета резултата

облику са временом настанка, нивоом критичности, бројчаним кодом, извором и контекстом. За различите компоненте система резервисани су засебни опсежи кодова, приказани у табели 3.2, док се догађаји аналитичког задатка повезују са његовим идентификатором. Интерни изузеци претварају се у структуриране грешке погодне за приказ у корисничким интерфејсима, без откривања интерних техничких детаља. Структурирани дневници омогућавају реконструкцију тока извршавања и утврђивање фазе у којој је проблем настао.

Табела 3.2: Опсези кодова догађаја у систему JELICA

Компонента	Опсег кодова
Резервисано за будуће системске догађаје (SYSTEM_*)	1000–1999
Аналитичко језгро (CORE_*)	2000–2999
Интерфејс командне линије (CLI_*)	3000–3999
Серверска компонента веб-платформе (SERVER_*)	4000–4999
Веб-платформа (WEB_*)	5000–5999
Десктоп апликација (DESKTOP_*)	6000–6999
Резервисано	7000–9999

3.5.6 Кориснички интерфејси

Систем JELICA доступан је преко **интерфејса командне линије, веб-платформе и десктоп апликације**. Основни облици њиховог коришћења и однос према заједничком аналитичком језгру описани су у потпоглављу 3.3.5. Веб-платформа и десктоп апликација подржавају локализацију корисничког интерфејса на енглески, руски и српски језик, при чему је српски доступан на ћириличком и латиничном писму. Интерфејс командне линије користи енглески језик, док се извештаји могу генерисати на сва четири подржана облика.

При имплементацији корисничких интерфејса посебна пажња посвећена је приступачности. Веб-платформа развијена је са циљним нивоом усаглашености AA према смерницама *Web Content Accessibility Guidelines 2.2* (WCAG 2.2)³, док се применљиви принципи истог стандарда користе и приликом пројектовања десктоп апликације. То обухвата, између осталог, рад помоћу тастатуре, видљив фокус активног елемента, довољан контраст, семантички структуриран садржај и избегавање ослањања искључиво на боју за преношење значења.

Приступачност интерфејса командне линије примењује се у складу са ограничењима окружења терминала. Боје и декоративни симболи нису неопходни

³ *Web Content Accessibility Guidelines* (WCAG) представља међународни стандард приступачности веб-садржаја који развија *World Wide Web Consortium* (W3C) у оквиру иницијативе *Web Accessibility Initiative* (WAI). Стандард дефинише смернице и проверљиве критеријуме намењене побољшању приступачности веб-страница и веб-апликација особама са визуелним, слухним, моторичким, говорним, когнитивним и другим облицима инвалидитета. Смернице су организоване око четири основна принципа: садржај треба да буде опажљив, употребљив, разумљив и робустан. Критеријуми усаглашености подељени су на нивое A, AA и AAA; усаглашеност са нивоом AA подразумева испуњавање свих критеријума нивоа A и AA. Верзија WCAG 2.2 објављена је као препорука W3C-а 5. октобра 2023. године и представља актуелну верзију стандарда WCAG 2 у време израде овог рада. [25]

за разумевање резултата и могу се искључити, док су значајне поруке увек доступне и у текстуалном облику. Графички интерфејси прилагођавају распоред различитим величинама екрана и омогућавају извршавање основних операција помоћу тастатуре.

Приликом избора технологија и имплементације интерфејса тежи се и компатибилности са различитим рачунарима, оперативним системима и веб-прегледачима. Интерфејси не претпостављају употребу високоперформансног локалног уређаја, а зависност од функција доступних искључиво у најновијим верзијама платформи избегава се када за то не постоји функционална потреба. На тај начин се подржава широк спектар актуелних и старијих система, у мери у којој то не нарушава исправност рада апликације.

Интерфејс командне линије

Интерфејс командне линије реализован је као засебна апликација `jelica-cli`, која након инсталације омогућава коришћење команде `jelica` из било ког директоријума на рачунару на којем је инсталирана локална верзија система. Он представља непосредну приступну тачку локалној верзији аналитичког језгра и не имплементира сопствене аналитичке функционалности. Његов задатак је да корисничке команде и параметре претвори у структуриране захтеве намењене аналитичком језгру и да добијене резултате прикаже у облику погодном за интерактивну или машинску употребу. Интерфејс командне линије омогућава подешавање локалног окружења, креирање, покретање, праћење, паузирање, настављање и отказивање аналитичких задатака, њихово клонирање и приступ сачуваним пакетима резултата.

Већина појединачних команди интерфејса командне линије извршава засебне операције над аналитичким задацима, што је погодно за интеграцију са другим програмима и аутоматизацију. За најчешћи интерактивни сценарио употребе уведена је команда `jelica analyze`, која у једном позиву обједињује креирање задатка (`jelica tasks create`), његово покретање (`jelica tasks start`) и праћење извршавања (`jelica tasks watch`). Извори улазних података задају се позиционим аргументима (нпр. путањама до датотека или директоријума и *NCBI* идентификаторима), док се конфигурација може проследити у *JSON* датотеци, параметрима командне линије или комбинацијом оба начина. За угнежђене параметре користи се тачкаста нотација, при чему вредности задате командом имају приоритет над вредностима из конфигу-

рационе датотеке. Структура команде `jelica analyze` и начини задавања њених параметара приказани су на слици 3.8a, док је пример њеног покретања и извршавања аналитичког задатка приказан на слици 3.8b.

Током извршавања *CLI* приказује идентификатор задатка, тренутно стање, активну фазу и напредак, као и значајне дијагностичке и друге поруке. Недозвољени параметри, грешке у конфигурацији и недопуштени прелази између стања приказују се без откривања интерних техничких детаља извршавања.

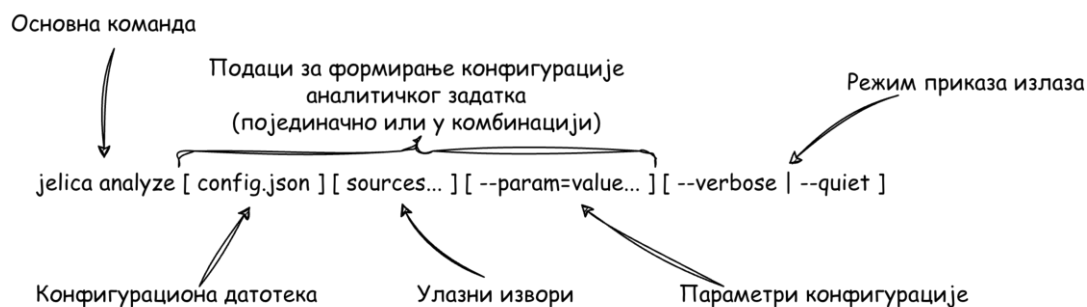
За аутоматизовану употребу команде подржавају машински читљив излаз и једнозначне излазне кодове. Захваљујући томе *JELICA* се може укључити у скрипте, *wrapper*-е програма и друге аутоматизоване процесе без анализирања текста намењеног човеку. Исти скуп основних команди доступан је у оперативним системима *macOS*, *Windows* и *Linux*.

Веб-платформа

Веб-платформа пружа графички приступ функцијама система без потребе за локалним радом у терминалу. Кориснички интерфејс омогућава организовање аналитичких задатака у оквиру пројеката, додавање улазних података и управљање њима, подешавање параметара аналитичког задатка, праћење његовог извршавања и интерактиван преглед добијених резултата.

Пројекат представља организациони ниво специфичан за веб-платформу и не мења модел аналитичког задатка у језгру. Један пројекат може да обједини повезане улазне податке, више аналитичких задатака и њихове резултате, али и да омогући заједнички рад више корисника над истим анализама. Корисници са одговарајућим правима могу у оквиру пројекта креирати и мењати задатке, покретати, паузирати и настављати њихово извршавање, као и остављати коментаре другим учесницима. На тај начин пројекти истовремено служе за организацију истраживачких података и основну сарадњу корисника у оквиру веб-платформе.

При покретању аналитичког задатка веб-интерфејс формира конфигурацију аналитичког задатка у истом формату као и остали интерфејси система и прослеђује је серверској компоненти. Архитектура комуникације веб-платформе са аналитичким језгром описана је у потпоглављу 3.3.2. Кориснику се током извршавања приказују тренутно стање аналитичког задатка и напредак његовог извршавања. Корисник може прилагодити обавештења о значајним догађајима у систему и начин њиховог достављања — у самом



(a) Шематски приказ структуре команде `jelica analyze`

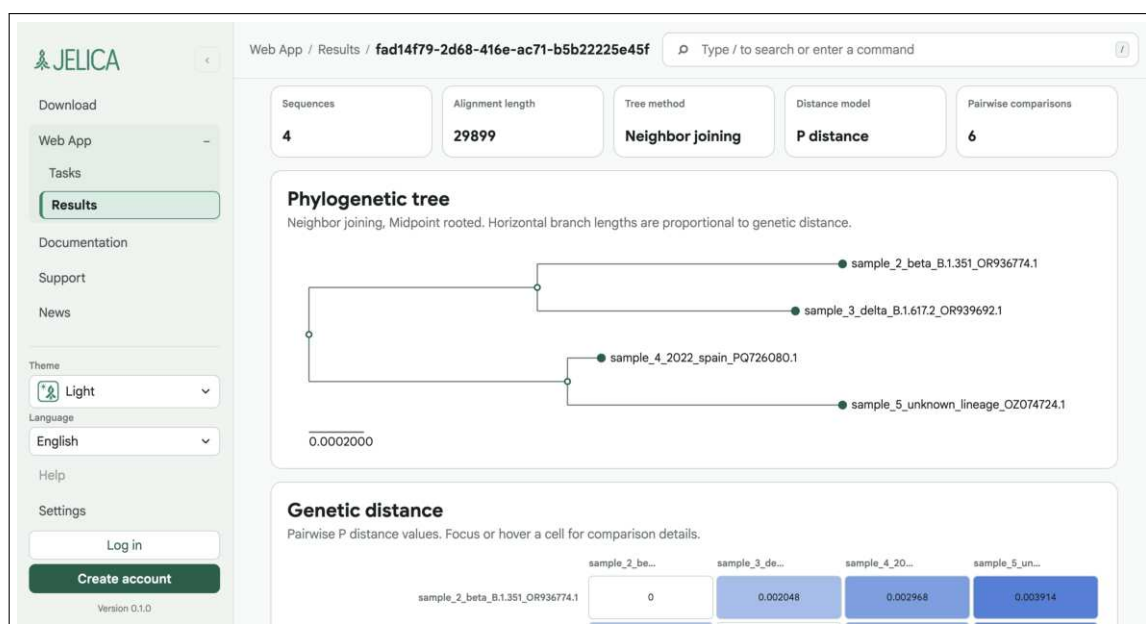
```
greg@MacBook-Pro-Greg combined % jelica analyze sars_cov2_reference_plus_6_sample
s.fasta --reference=reference_wuhan_hu_1_NC_045512.2 --statistics.kmers='["ATTC",
"CGT", "ATA", "AAAA"]' --comparative_analysis.pairwise.enabled=true --distance_m
atrix.enabled=true \
[ --phylogenetic_tree.enabled=true \
  --clade_detection.enabled=true \
  --clade_detection.max_within_clade_distance=0.01
Analysis task ccbde351-7547-4048-b101-4d4da7be3ba6 was created and started.
Stage started: initialize_job
Stage completed: initialize_job
Stage started: input_acquisition
Input acquisition completed: 1 input file was materialized.
Stage completed: input_acquisition
Stage started: input_processing
Input processing started: 1 files.
Input processing completed: 7 valid, 0 invalid.
Stage completed: input_processing
Stage started: alignment
Stage completed: alignment
Alignment result published.
Alignment completed.
Stage started: comparative_analysis
Comparative-analysis result published.
Comparative analysis completed.
Stage started: distance_matrix
Stage completed: distance_matrix
Stage started: phylogenetic_tree
Stage completed: phylogenetic_tree
Stage started: clade_detection
Stage completed: clade_detection
Stage started: result_package
Stage completed: result_package
Task ccbde351-7547-4048-b101-4d4da7be3ba6 completed.
Task ID                               State   Stage   Progress   Warnings
ccbde351-7547-4048-b101-4d4da7be3ba6 completed -      100%      0
greg@MacBook-Pro-Greg combined %
```

(b) Пример покретања и извршавања аналитичког задатка командом `jelica analyze`

Слика 3.8: Структура и пример употребе команде `jelica analyze`

интерфејсу, путем електронске поште или преко месинџера *Telegram*. Подршка за *WhatsApp*, која је била предвиђена функционалном спецификацијом система, није укључена у тренутну верзију због додатних организационих и инфраструктурних услова за коришћење званичног програмског интерфејса овог сервиса, али архитектура система омогућава њено накнадно додавање као додатног канала за достављање обавештења. Иако је подршка за *Viber* такође била предвиђена функционалном спецификацијом система у потпоглављу 3.2.1, током имплементације се од ње одустало због комерцијалних услова коришћења програмског интерфејса овог сервиса и фиксних трошкова одржавања бота.

Резултати се у веб-интерфејсу приказују комбинацијом табеларних података, сажетака и интерактивних визуелизација, као што је приказано на слици 3.9. На основу пакета резултата могуће је формирати и преузети извештаје и извозне датотеке, као и преузети саму датотеку `.jelica` ради локалног чувања или отварања у десктоп апликацији.



Слика 3.9: Приказ резултата анализе у веб-интерфејсу *JELICA*: сажетак, филогенетско стабло и матрица генетичких растојања

Поред аналитичких функција, веб-платформа служи и као централно место за преузимање локалних верзија система са *CLI* и десктоп интерфејсом, приступ документацији и информацијама о пројекту, као и за подршку и слање повратних информација.

Десктоп апликација

Десктоп апликација обезбеђује графички рад са локалном верзијом система и кориснику поједностављује креирање аналитичких задатака, подешавање њихових параметара и управљање редом извршавања. Она користи исти визуелни систем и основне компоненте интерфејса као веб-платформа, чиме се обезбеђује уједначен начин навигације и рада са резултатима. Комуникација са аналитичким језгром остварује се преко машинског интерфејса *CLI*-ја, који служи као посреднички слој између десктоп апликације и модула *Core*.

Апликација омогућава и приказ структурираних резултата, филогенетских стабала, нумеричких карактеристика секвенци и извештаја. Формат *.jelica* може бити повезан са десктоп апликацијом, тако да се пакет резултата отвара непосредно као документ система *JELICA*, без ручног приступа његовој унутрашњој структури.

3.5.7 Инфраструктура веб-платформе

За разлику од локалних варијанти система, веб-платформа обухвата додатни инфраструктурни слој за мрежни приступ, истовремени рад више корисника и трајно чување њихових података. Јавно доступна верзија система користи протокол *HTTPS*, док су кориснички интерфејс, серверски слој и аналитичко извршавање логички раздвојени. За јавно постављање веб-платформе регистрован је наменски домен <https://jelica.bio>, преко којег је омогућен приступ јавној верзији система.

Комуникација серверског слоја са аналитичким језгром организована је у складу са архитектуром описаном у потпоглављу 3.3.2. Аналитички задатак се извршава независно од животног циклуса појединачног *HTTP* захтева, док се ажурирања стања и напретка активних задатака веб-клијенту прослеђују преко *WebSocket* везе.

Аналитички задаци и пакети резултата повезују се са одговарајућим пројектима и корисничким налозима, док серверска апликација контролише приступ подацима и ограничења расположивог простора за њихово чување. Раздвајање веб-инфраструктуре од аналитичког језгра омогућава независну инсталацију и скалирање веб-платформе без измене већ имплементираних алгоритама и формата резултата.

3.5.8 Корисничка документација

Корисничка документација обухвата опис техничке организације система и практично упутство за његово коришћење. Доступна је у оквиру веб-платформе и десктоп апликације, са могућностима брзе навигације и пре-траге, и локализована је на исте језичке варијанте као графички кориснички интерфејси. При њеној имплементације примењују се исти принципи приступности као и у осталим деловима корисничког интерфејса.

Све варијанте документације формирају се из заједничког извора садржаја, чиме се избегава паралелно одржавање више независних верзија. За локални рад преко интерфејса командне линије аутоматски се формира *PDF* верзија истог садржаја, која се испоручује уз локалну инсталацију и доступна је за преузимање са веб-платформе.

Глава 4

Закључак

У оквиру овог рада развијен је софтверски систем *JELICA*, који обједињује различите кораке упоредне анализе геномских секвенци у јединствен аналитички ток. Аутоматизација припреме улазних података, поравнања, поређења и филогенетске анализе смањује потребу за ручним преношењем података између независних алата и обезбеђује доследан начин извршавања и чувања резултата. Преносиви формат *.jelica* омогућава чување, размену и накнадну употребу целокупног резултата спроведене анализе.

Систем је доступан преко интерфејса командне линије, веб-платформе и десктоп апликације, чиме је прилагођен и аутоматизованој и интерактивној употреби. Веб-верзија омогућава рад без локалне инсталације, док локалне варијанте омогућавају обраду података на сопственом рачунару. Изворни код је јавно доступан и, у складу са лиценцом, може се користити, мењати, проширивати и укључивати у друге системе уз очување података о оригиналном пројекту и ауторству, док су ограничени препродаја самог система и комерцијална употреба заснована превасходно на њему.

Даљи развој може обухватити подршку за додатне методе и алате за поравнање секвенци, развој сопственог алгоритма поравнања и замену рачунарски захтевних операција заснованих на библиотеци *Biopython* ефикаснијим сопственим имплементацијама. Планирано је и проширење метода за израчунавање генетичких растојања и филогенетску анализу, унапређење паралелног и дистрибуираног извршавања, додавање нових локализација и програмски приступ јавној веб-верзији путем јавног *API*-ја.

Библиографија

- [1] Ivan Aksamentov, Cornelius Roemer, Emma B. Hodcroft, and Richard A. Neher. Nextclade: clade assignment, mutation calling and quality control for viral genomes. *Journal of Open Source Software*, 6(67):3773, 2021.
- [2] Stephen F. Altschul, Warren Gish, Webb Miller, Eugene W. Myers, and David J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410, 1990.
- [3] Biopython Project. Biopython. <https://biopython.org/>.
- [4] Cypress. Cypress. <https://www.cypress.io/>.
- [5] Docker. Docker. <https://www.docker.com/>.
- [6] Electron. Electron. <https://www.electronjs.org/>.
- [7] EMBL-EBI. Clustal Omega. <https://www.ebi.ac.uk/jdispatcher/msa/clustalo>.
- [8] FastAPI. FastAPI. <https://fastapi.tiangolo.com/>.
- [9] Alexey Gurevich, Vladislav Saveliev, Nikolay Vyahhi, and Glenn Tesler. QUASt: quality assessment tool for genome assemblies. *Bioinformatics*, 29(8):1072–1075, 2013.
- [10] Kazutaka Katoh, Kazuharu Misawa, Kei-ichi Kuma, and Takashi Miyata. Mafft: a novel method for rapid multiple sequence alignment based on fast fourier transform. *Nucleic Acids Research*, 30(14):3059–3066, 2002.
- [11] Ben Langmead and Steven L. Salzberg. Fast gapped-read alignment with bowtie 2. *Nature Methods*, 9(4):357–359, 2012.

- [12] Ben Langmead, Cole Trapnell, Mihai Pop, and Steven L. Salzberg. Ultrafast and memory-efficient alignment of short dna sequences to the human genome. *Genome Biology*, 10(3):R25, 2009.
- [13] MAFFT. MAFFT MSA Viewer. <https://mafft.cbrc.jp/msviewer/>.
- [14] Microsoft. TypeScript. <https://www.typescriptlang.org/>.
- [15] National Center for Biotechnology Information. GenBank. <https://www.ncbi.nlm.nih.gov/genbank/>.
- [16] National Center for Biotechnology Information. Multiple Sequence Alignment Viewer. <https://www.ncbi.nlm.nih.gov/tools/msviewer/>.
- [17] PostgreSQL Global Development Group. PostgreSQL. <https://www.postgresql.org/>.
- [18] Pydantic. Pydantic. <https://pydantic.dev/>.
- [19] pytest. pytest. <https://pytest.org/>.
- [20] React. React. <https://react.dev/>.
- [21] Fabian Sievers, Andreas Wilm, David Dineen, Toby J. Gibson, Kevin Karplus, Weizhong Li, Rodrigo Lopez, Hamish McWilliam, Michael Remmert, Johannes Söding, Julie D. Thompson, and Desmond G. Higgins. Fast, scalable generation of high-quality protein multiple sequence alignments using clustal omega. *Molecular Systems Biology*, 7(1):539, 2011.
- [22] Typer. Typer. <https://typer.tiangolo.com/>.
- [23] Vercel. Next.js. <https://nextjs.org/>.
- [24] Vitest. Vitest. <https://vitest.dev/>.
- [25] World Wide Web Consortium. Web content accessibility guidelines (WCAG) 2.2. <https://www.w3.org/TR/WCAG22/>, 2024. W3C Recommendation, 12 December 2024.

Биографија аутора

Григории Рогов рођен је 12. августа 1998. године у Москви, у Русији. Средњу школу са појачаном наставом из биологије и хемије завршио је у Москви 2016. године. Самостално је стекао знања из програмирања, пре свега у програмском језику *JavaScript*, а професионалну каријеру започео је 2018. године у компанији *STREAM LABS*, која се бави развојем хардверских и софтверских решења за телевизијску и радијску опрему, као и системе видео-надзора. Исте године уписао је Московски технички универзитет веза и информатике, на студијски програм „Информациони системи и технологије“, који је завршио 2023. године са одличним успехом. Тема дипломског рада била је „Развијање модела гранања за систем контроле верзија *Git*“.

Године 2022. преселио се са супругом у Београд, где је наставио професионални рад као самостални предузетник, пружајући услуге развоја софтвера. Током професионалног рада бавио се пре свега развојем веб-апликација и корисничких интерфејса. Мастер академске студије на студијском програму Информатика Математичког факултета Универзитета у Београду уписао је 2024. године.