

Simulacija modularnog telemetrijskog sustava za praćenje fizioloških indikatora stresnih mentalnih stanja

Novosel, Tomislav

Master's thesis / Diplomski rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Electrical Engineering and Computing / Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva**

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:168:455431>

Rights / Prava: [In copyright](#) / [Zaštićeno autorskim pravom](#).

Download date / Datum preuzimanja: **2025-03-30**



Repository / Repozitorij:

[FER Repository - University of Zagreb Faculty of Electrical Engineering and Computing repository](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 400

**SIMULACIJA MODULARNOG TELEMETRIJSKOG SUSTAVA
ZA PRAĆENJE FIZIOLOŠKIH INDIKATORA STRESNIH
MENTALNIH STANJA**

Tomislav Novosel

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 400

**SIMULACIJA MODULARNOG TELEMETRIJSKOG SUSTAVA
ZA PRAĆENJE FIZIOLOŠKIH INDIKATORA STRESNIH
MENTALNIH STANJA**

Tomislav Novosel

Zagreb, lipanj 2024.

DIPLOMSKI ZADATAK br. 400

Pristupnik: **Tomislav Novosel (0036510401)**
Studij: Računarstvo
Profil: Programsko inženjerstvo i informacijski sustavi
Mentor: izv. prof. dr. sc. Siniša Popović

Zadatak: **Simulacija modularnog telemetrijskog sustava za praćenje fizioloških indikatora stresnih mentalnih stanja**

Opis zadatka:

Proučiti literaturu iz područja kontinuiranog praćenja fizioloških indikatora stresa u kontekstu telepsihijatrijskih intervencija i obavljanja stresnih poslova. Oblikovati koncept simulacije modularnog telemetrijskog sustava za praćenje fizioloških indikatora stresnih mentalnih stanja, neovisnoga o konkretnoj primjeni. Primanje fizioloških podataka sa stvarnih akvizicijskih uređaja zamijeniti u simulaciji s tokovima podataka iz datoteka u kojima se nalaze unaprijed snimljeni stvarni fiziološki signali. Protočna arhitektura simuliranoga sustava treba uključivati komponente za akviziciju fizioloških signala, njihovu obradu i izračun fizioloških indikatora stresnih mentalnih stanja, izračun promatranih mentalnih stanja iz ovih indikatora, vizualizaciju izabranih podataka i sinkroniziranu pohranu svih vremenskih serija koje kolaju u sustavu. Telemetrijske mogućnosti trebaju omogućiti odašiljanje izabranih vremenskih serija podataka prema drugom računalu radi njihove vizualizacije korisnicima na drugoj lokaciji.

Rok za predaju rada: 28. lipnja 2024.

SADRŽAJ

1. Uvod	1
2. Stres i relevantni aspekti ljudske fiziologije	2
2.1. Stres u zrakoplovstvu	3
2.2. Biosignali povezani sa stresom	4
2.2.1. Moždana aktivnost	4
2.2.2. Rad srca	7
2.2.3. Disanje	8
2.2.4. Elektrodermalna aktivnost	8
2.2.5. Ostali fiziološki indikatori	8
3. Izabrana literatura o sustavima za fiziološko praćenje stresa u stvarnom vremenu	10
4. Oblikovanje simulacije telemetrijskog sustava za praćenje fizioloških indikatora stresnih mentalnih stanja	13
4.1. Zahtjevi	13
4.1.1. Funkcionalni zahtjevi	13
4.1.2. Nefunkcionalni zahtjevi	14
4.2. Predložena arhitektura simulacije	14
5. Programski alati za ostvarenje simulacije	18
5.1. LMAX Disruptor	19
5.1.1. Problemi brava i redova	19
5.1.2. Disruptor kao rješenje	20
6. Programsko ostvarenje simulacije	22
6.1. Korišteni skup podataka snimljenih fiziološkim sensorima	22
6.2. Relevantni dijelovi programskog koda	23

6.2.1.	Prikupljanje fizioloških podataka	24
6.2.2.	Računanje fizioloških značajki	25
6.2.3.	Klasifikacija izabranih mentalnih stanja	28
6.2.4.	Spremanje i vizualizacija	28
7.	Rezultati	30
8.	Zaključak	35
	Literatura	36

1. Uvod

Tema rada je izrada simulacije modularnog telemetrijskog sustava za praćenje stresnih mentalnih stanja. Motivacija je bila napraviti sustav koji će brzo i efikasno obrađivati podatke i detektirati stresna mentalna stanja. Razvoj sustava za slične namjene u skladu s načelima paralelizma i istodobnosti (engl. *concurrency*) i njihova evaluacija tema je raznih istraživanja [6][21][20][7]. U ovom radu stavlja se fokus na aspekte ugradnje odgovarajuće razine istodobnosti u okviru oblikovanja i razvoja jednog primjera simulacije ovakvoga sustava. Na ulaznoj strani simulacije, primanje podataka sa stvarnih senzora zamjenjuje se čitanjem unaprijed snimljenih realnih podataka iz datoteka NASA-inoga otvorenog podatkovnog skupa [18] namijenjenog razvoju algoritama za raspoznavanje stresnih i visokorizičnih mentalnih stanja pilota tijekom leta. Na izlaznoj strani simulacije, algoritam za raspoznavanje stresnih mentalnih stanja predstavljen je s nekoliko predefiniраниh pravila koja nemaju posebno teorijsko uporište niti uporište u nekom konkretnom skupu podataka. Glavna biblioteka na kojoj se temelji sustav je LMAX Disruptor koja omogućava efikasnu komunikaciju između dretvi i samim time efikasniju obradu podataka i detekciju stresnih stanja na temelju fizioloških podataka.

2. Stres i relevantni aspekti ljudske fiziologije

Stres je izazvan podražajima koje nazivamo stresori. To je neuroendokrini, autonomni, bihevioralni, psihološki, emocionalni i kognitivni fenomen kao odgovor na stresore [17]. Najlakše ga je opisati s 3 elementa [12]:

- stanje povećane podražljivosti ili uzbuđenja
- iskustvo koje se smatra odvrćućim
- iskustvo koje se doživljava izvan kontrole ili nepredvidljivo.

Stres izaziva različite reakcije ovisno o pojedincu u smislu manifestacije, trajanja i intenziteta. Stresom se pokušava održati homeostaza protiv podražaja, stresora, ali i potaknuti dugoročnu prilagodbu na sami podražaj. [4]

Prema dostupnoj literaturi možemo razlikovati vrste stresora[17]:

1. Fizički: manjak sna, bol, pretjerana fizička aktivnost...
2. Vanjski: buka, vibracija, vrijeme...
3. Emocionalni: smrt bližnje osobe, prekid ljubavne veze...
4. Kognitivni: događaju se zbog prevelikog mentalnog napora oko nekog zadatka
5. Kronični: nezaposlenost, teška financijska situacija, bolest...

Stres je od strane Svjetske zdravstvene organizacije opisan kao epidemija 21. stoljeća, a njegove posljedice na zdravlje ljudi su uveliko dokumentirane. Praćenjem stresa i njegovih posljedica otkriveno je da duža izloženost povećava vjerojatnost razvijanja depresivnih poremećaja, kardiovaskularnih bolesti i bolesti mišićno-koštanog sustava.[24]

Odgovor tijela na stres iskazuje se na dva načina [27]. Prvi je spor i za njega je zadužena osovina hipotalamus-hipofiza-nadbubrežna žlijezda. No za ovaj rad taj način

odgovora nije bitan. Drugi način odgovora tijela na stres je onaj brži za koji je zadužen autonomni živčani sustav. Njegova aktivacija je od koristi do određene granice jer se aktivira simpatički živčani sustav koji pruža pojačanu aktivaciju mišićnog sustava te ubrzava rad srca. U isto vrijeme može se aktivirati parasimpatički živčani sustav te izazvati nesvjesticu ili zastoje rada srca. Odgovori simpatičkog sustava na stres uglavnom se primjenjuju kroz produženu moždinu te tako ubrzavaju rad srca, disanje, podižu tjelesnu temperaturu te potiču znojenje i širenje krvnih žila. Tako tijelo ulazi u stanje "borba ili bijeg" (engl. *fight or flight*).

Korelacija učinkovitosti i stresa iskazuje se Hebbovom ili obrnutom U krivuljom (Slika 2.1) [11]. Učinkovitost raste proporcionalno s razinom stresa do određene razine, a nakon toga kako razina stresa raste, učinkovitost pada. Nedavna istraživanja su pokazala da taj model nije najprecizniji zbog količine i složenosti faktora koji na to utječu. No stres se može podijeliti na pozitivan i negativan. Preniska razina stresa i prevelika razina negativnog stresa negativno utječu na učinkovitost dok je umjerena količina povezana s povećanom učinkovitošću.

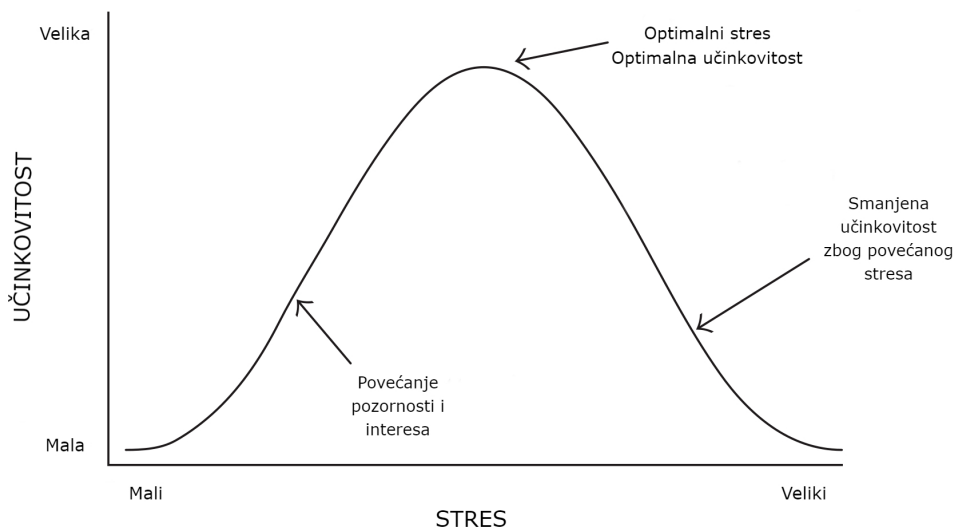
2.1. Stres u zrakoplovstvu

U zrakoplovstvu stres je važan čimbenik jer direktno utječe na izvođenje letova. Nesreće aviona mogu biti posljedica ljudske pogreške koja je pak posljedica stresa, nepažnje, mentalnog opterećenja, umora i odvratanja pažnje. Veza između stresa, anksioznosti i pažnje nije jednostavna i ovisi o mnogo parametara i zato stres nije moguće promatrati kao izolirani parametar. [17]

Brojna istraživanja su pokazala da dugotrajna izloženost stresu može dovesti do ozbiljnih zdravstvenih problema kao što su kardiovaskularne bolesti, mentalni poremećaji i smanjena sposobnost donošenja odluka. Čimbenici koji doprinose stresu uključuju nepravilne radne sate, odgovornost za sigurnost putnika, složene operacije koje moraju biti obavljene pod pritiskom, te česte promjene vremenskih zona [13].

Kod vojnih pilota ti čimbenici su pojačani, a trening za njih posebno stavlja naglasak na reakcije u stresnim situacijama. Od pilota se zahtjeva da u letu, dok rukuju većim brojem zadataka, donose dobre odluke i uvijek budu fizički spremni uz veliki broj stresora koji se pojavljuju, ponekad u isto vrijeme. Razumijevanje opterećenja pod kojim se nalaze piloti, dok lete i upravljaju instrumentima u pilotskoj kabini, važno je za daljnji razvoj novih zrakoplovnih tehnologija. Tehnologija može značajno pomoći pilotima pri rukovanju stresnim situacijama. U vidu poboljšanja prepoznavanja stresnih i mentalno opterećujućih situacija kod pilota imamo 3 pristupa [17]:

- samoprocjena
- procjena uspješnosti
- objektivna procjena



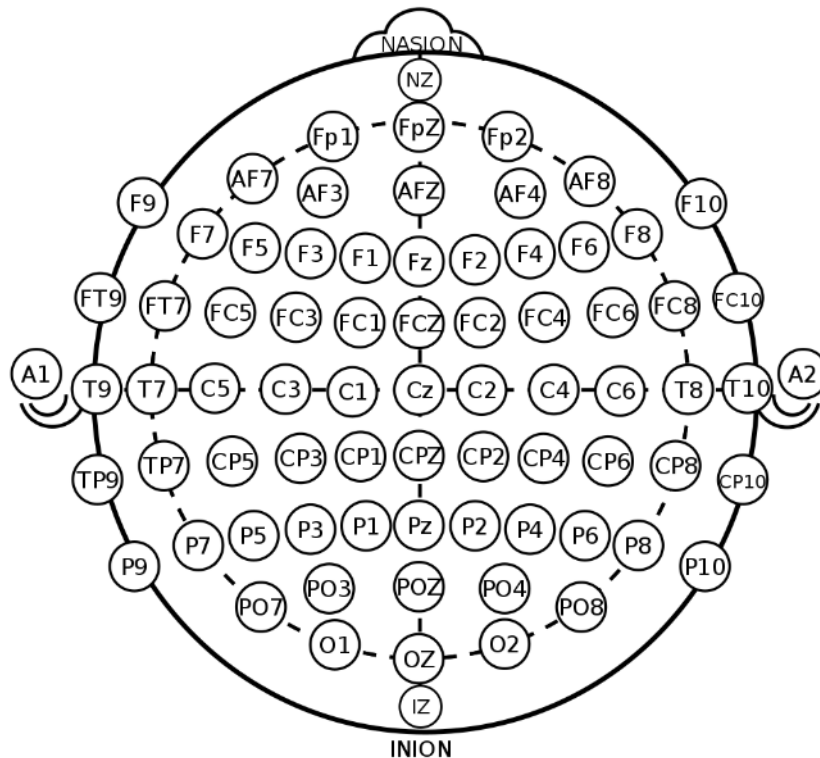
Slika 2.1: Hebbova krivulja

2.2. Biosignali povezani sa stresom

Kada je osoba pod stresom to se vidi na njoj u obliku motoričkih akcija ili opazivih ponašanja kojima posreduje somatski živčani sustav, te se istodobno događaju promjene u njenim centralnim i perifernim fiziološkim signalima, koji odražavaju aktivnost mozga i autonomnog živčanog sustava. Motorički ili bihevioralni signali su promjene tijela, najčešće kao rezultat mišićne aktivnosti. U tu kategoriju spadaju pokreti oka, treptanje, pomicanje tijela i glave te govor i izraz lica. Fiziološki signali povezani su s vitalnim tjelesnim funkcijama kao što su rad srca, krvni tlak, moždana aktivnost, proširivanje zjenica, disanje i znojenje.

2.2.1. Moždana aktivnost

EEG je dijagnostički alat koji bilježi električnu aktivnost mozga. EEG koristi male metalne pločice koje se nazivaju elektrode i one se lijepe na glavu (Slika 2.2) te detektiraju električne impulse mozga. EEG se koristi za dijagnosticiranje raznih bolesti, ali i praćenje moždane aktivnosti kod ljudi u komi.



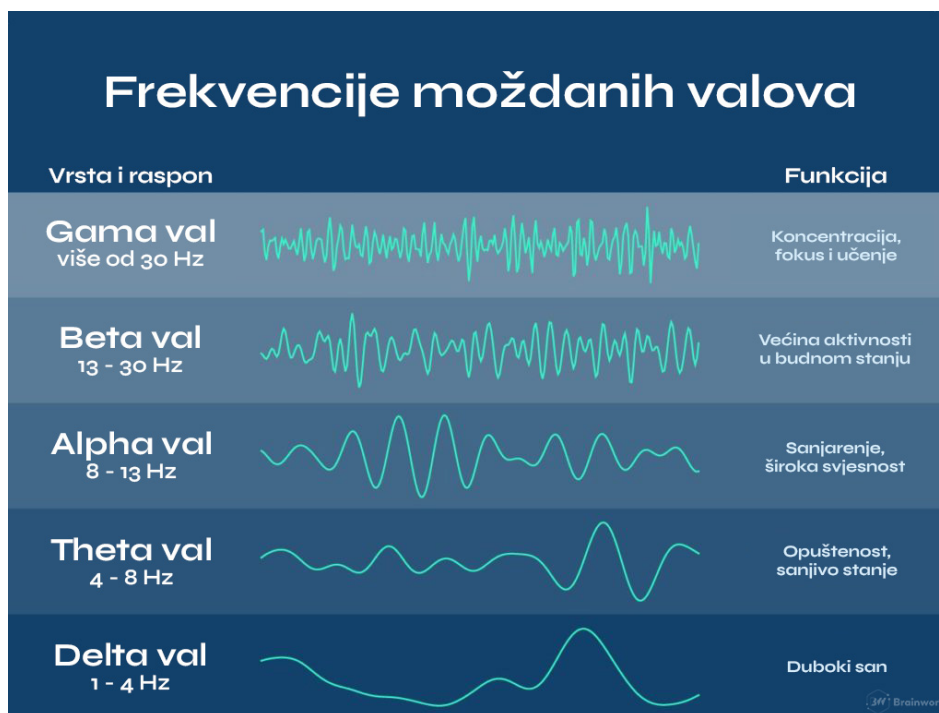
Slika 2.2: EEG elektrode (preuzeto sa [2])

Još jedna značajna prednost EEG tehnologije je njena sposobnost detektiranja promjena u neurofiziološkoj aktivnosti dok osoba izvodi razne zadatke. EEG je izrazito vrijedan alat u psihofiziološkim istraživanjima zbog svoje jedinstvene mogućnosti da precizno zabilježi promjene u moždanoj aktivnosti tijekom različitih psiholoških i emocionalnih stanja, kao i u stresnim situacijama. Ova metoda omogućava dublji uvid u to kako mozak funkcionira pod različitim uvjetima, pružajući tako vrijedne podatke koji mogu biti korisni u širokom spektru znanstvenih i medicinskih primjena [1].

EEG spektar je podijeljen na intervale frekvencija i ima ih 5:

- 0.5-4 Hz - δ
- 4-8 Hz - θ
- 8-13 Hz - α
- 13-30 Hz - β
- 30-70 Hz - γ

Za vrijeme opuštenog stanja i odmaranja dominira α ritam, dok se za vrijeme veće pobuđenosti i stresa može uočiti dominacija β valova [10]. Stres pozitivno korelira s β valovima u prednjem sljepoočnom režnju mozga. Mjera koja daje bolji uvid u to je li



Slika 2.3: EEG valovi (preuzeto sa [3])

osoba pod stresom bila bi

$$\beta/\alpha$$

Uočeno je da što je osoba pod većim stresom to je vrijednost β/α veća.

Lijeva strana mozga povezana je s emocijama koje nas pokreću, kao što su sreća i ljutnja, dok je desna povezana s emocijama koje nas tjeraju na izbjegavanje stvari i tu spadaju tuga i strah [14]. Većina istraživanja pokazuje da je u stanju stresa vidljiva povećana aktivnost α valova na desnoj strani mozga u odnosu na lijevu [1]. Takav fenomen uočen je u stresnim stanjima kao što su pisanje ispita, gledanje sretnih, tužnih i horor filmova. Indeks asimetrije je stoga dobar atribut s pomoću kojeg se može odrediti je li osoba pod stresom. Formula za indeks asimetrije je:

$$\text{Indeks asimetrije} = \ln(\alpha)|_{L_{kanal}} - \ln(\alpha)|_{D_{kanal}}$$

Vrijednosti α valova dobit ćemo s elektroda F3 i F4 koje se nalaze iznad dorzolateralnog prefrontalnog režnja koji je povezan sa stresnim stanjima[15][23].

Povećani γ ritam također može biti jedan od pokazatelja stresa, no povećanje u tom spektru također može biti i pokazatelj mišićne aktivnosti pa to nije najbolji kriterij po kojem možemo određivati stresna stanja [9].

EEG mjera koja daje najbolji uvid u angažiranost ispitanika je indeks angažiranost

(engl. *Engagement index*).

$$\text{Indeks anga\u017eirivosti} = \frac{\beta}{\alpha + \theta}$$

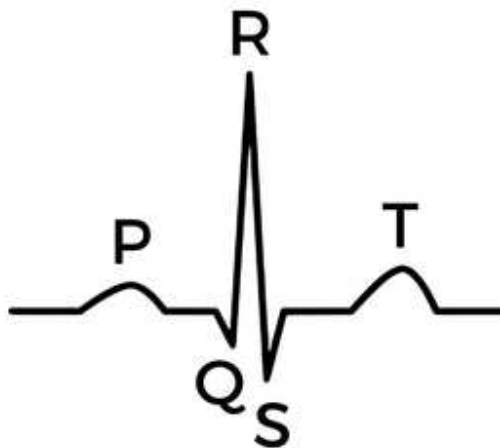
Podaci se prikupljaju s elektroda Pz, Cz, P3 i P4. Indeks anga\u017eirivosti pokazao se kao najosjetljiviji na promjene u stresu i mentalnom optere\u0107enju [22].

2.2.2. Rad srca

Radom srca upravljaju parasimpatički i simpatički \u017div\u0107ani sustav. Stres pokre\u0107e simpatički \u017div\u0107ani sustav koji onda pove\u0107ava broj otkucaja srca i silu kontrakcije sr\u0107anog mi\u0161i\u0107a kako bi krv br\u017ee kolala organizmom s ciljem br\u017ee otklanjanja stresora.

Rad srca bilje\u017ei se ure\u0107ajem koji se zove elektrokardiogram ili skra\u0107eno EKG. EKG ciklus (Slika 2.4) sastoji se od 5 valova:

- P val - depolarizacija pretkljetki i punjenje kljetki krvlju
- QRS val - depolarizacija kljetki, krv odlazi u tijelo iz srca
- T val - polarizacija kljetki i priprema za sljede\u0107i ciklus



Slika 2.4: Jedan ciklus EKG-a (preuzeto s [19])

Najzanimljiviji je RR interval, mjera duljine izme\u0107u dva QRS kompleksa. Na temelju RR intervala mogu\u0107e je dobiti broj otkucaja srca u minuti i varijabilnost sr\u0107ane frekvencije. Otkucaji srca su najrazvijenija mjera za procjenu razine stresa. \u0160to je RR interval manji to je broj otkucaja u minuti ve\u0107i, samim time i osoba je pod ve\u0107im stresom. [9]

2.2.3. Disanje

Disanje se može izmjeriti kao brzina ili volumen kojim se zrak izmjenjuje u plućima. Brzina i dubina disanja su mjere koje se najčešće koriste u promatranju disanja [25]. Isto tako je koristan u detekciji stresa uz druge ranije nabrojane mjere. Tijekom stresa brzina disanja je povećana, srce brže kuca i krv traži više kisika koji će doći do mozga i mišića. Za vrijeme opuštanja brzina disanja je smanjena [26].

Osoba pod stresom nema pravilne obrasce disanja. Te su promjene najčešće vezane uz povećanje minutnog volumena, disanje je brže i pliće, a osoba više ne diše trbuhom nego prsima [9]. Takve promjene u disanju možemo promatrati Hallovim senzorom, senzorom naprezanja te senzorom rastezanja.

2.2.4. Elektrodermalna aktivnost

Elektrodermalna aktivnosti ili galvanski odgovor kože je mjera kojom se mjeri promjena električne aktivnosti u ekrinim znojnim žlijezdama koje kontrolira simpatički živčani sustav. Mjerenje se odvija s pomoću elektroda na dlanovima ili stopalima gdje je najveća koncentracija znojnih žlijezda. Mjera je izuzetno osjetljiva na temperaturu, spol i godine osobe te vlažnost zraka, doba godine i doba dana [5]. Tijekom obavljanja nekog zadatka, dodamo li sekundarni stresor, osoba će se više znojiti pa će i električna provodljivost kože porasti kao posljedica. No povećanjem intenziteta stresora, razina provodljivosti ostaje ista što ukazuje na problem osjetljivosti kod zadataka koji iziskuju veće mentalno opterećenje.

Često korištene mjere elektrodermalne aktivnosti su amplituda, frekvencija i latencija. Uočeno je da aktivnost raste već u samom iščekivanju stresora, prije nego se dogodi, u sličnoj mjeri kao i kada se stresor dogodi. Također, mjera ima visoku osjetljivost kod iznenadnih stresora, a posebnu korelaciju ima s brzinom rada srca kod pilota.

2.2.5. Ostali fiziološki indikatori

Dalje su navedeni još neki indikatori koji nisu korišteni u radu ovog sustava.

Elektromiografija

Elektromiografija se koristi kao tehnika procjene aktivnosti i napetosti mišića s pomoću elektroda na površini kože smještenih iznad mišića. Aktivacija simpatičkog živčanog sustava između ostalog potiče i povećanje mišićne napetosti posebno kod

trapezoidnog mišića na leđima. Aktivnost trapezoidnog mišića povezana je kod većeg i manjeg mentalnog opterećenja [9].

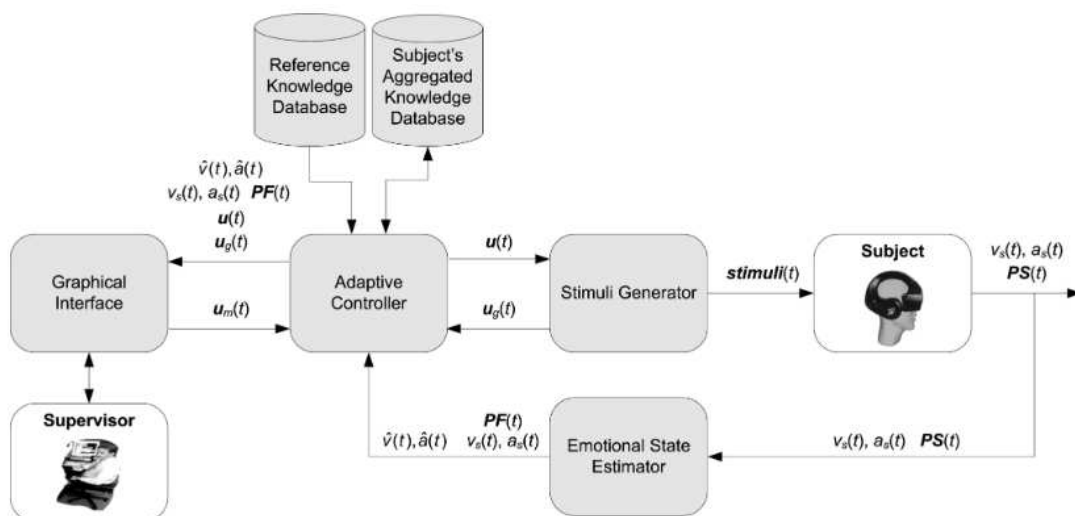
Očna aktivnost

Zbog napretka tehnologije, u posljednje vrijeme povećan je broj istraživanja koje koriste mjere očne aktivnosti kao što su veličina zjenica i količina, brzina i latencija treptanja. Veličina zjenica varira između 0.2 mm i 0.8 mm i to kontrolira grupa mišića koje inerviraju vlakna autonomnog živčanog sustava. Glavna zadaća ove sposobnosti je povećati vidljivost u tamnim okruženjima i povećati mogućnost promjene fokusa. Frekvencija i učestalost treptanja se smanjuju u okolnostima kada je potrebno obraditi više vizualnih informacija, posebno tijekom leta. Također frekvencija treptanja se smanjuje kod zadataka veće kompleksnosti u odnosu na zadatke manje kompleksnosti.

3. Izabrana literatura o sustavima za fiziološko praćenje stresa u stvarnom vremenu

Koncept praćenja fizioloških indikatora u stvarnom vremenu ima mnoge primjene. U kombinaciji s virtualnom stvarnosti moguće je liječenje i prevencija poremećaja uzrokovanih stresom. U područjima afektivnog računarstva mogu se naći primjeri gdje se računalima, uz pomoć fiziologije kao emocionalnih indikatora, daje mogućnost da prepoznaju emocije kod ljudi. Na taj se način postiže da i samo računalo prilagodi svoje djelovanje ovisno o emocijama osobe.

Ćosić et al. [6] u svom radu navode primjer sustava koji koristi podražaje u različitim oblicima koji se prikazuju osobi. To mogu biti slike, videozapisi ili zvukovi. Nakon stimuliranja osobe podražajem, bilježe se fiziološki odgovori osobe, procjenjuje se emocionalno stanje i na temelju tog stanja, u zatvorenoj petlji, generiraju se novi podražaji (Slika 3.1). Sustav se sastoji od adaptivnog upravljača koji generatoru

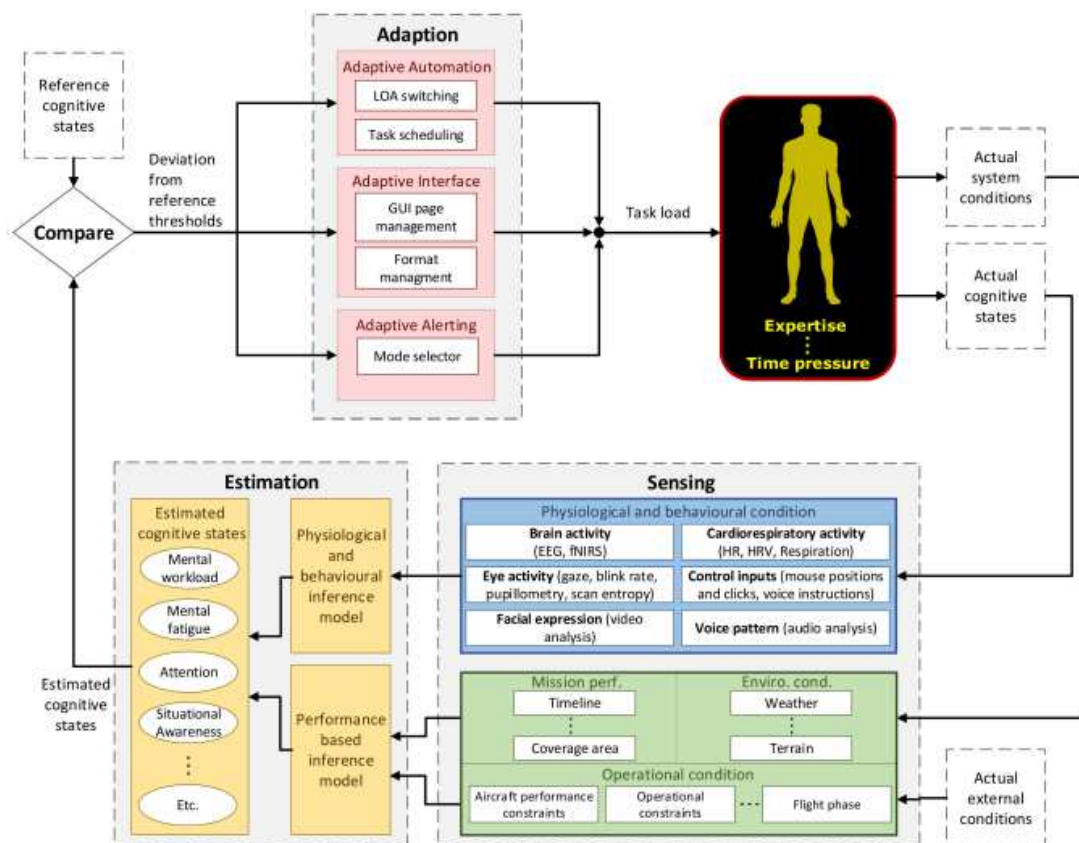


Slika 3.1: Arhitektura adaptivne VR simulacije (preuzeto s [6])

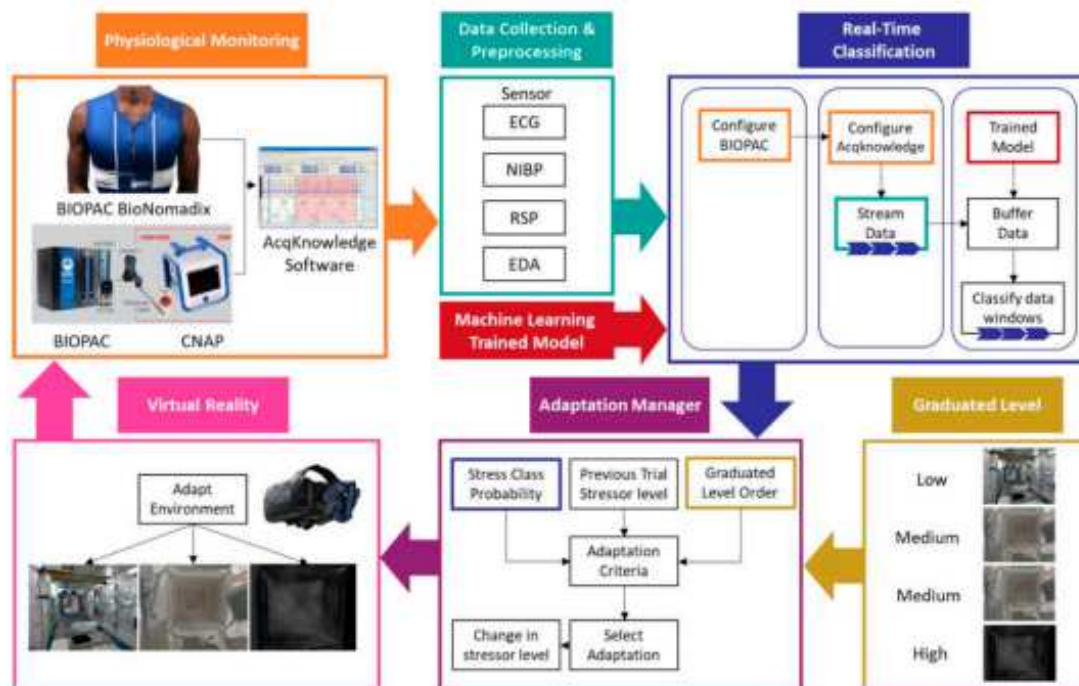
podražaja daje signal kakav bi trebao biti sljedeći podražaj. Nakon svakog podražaja procjenjuje se emocionalno stanje osobe kao reakcija na podražaj, a ta procjena šalje se adaptivnom upravljaču koji onda opet šalje signal generatoru kakav bi trebao biti sljedeći podražaj. Tako nastaje zatvorena petlja u kojoj se generiraju podražaji na temelju procjene emocija kao odgovora na prethodni podražaj. Svaki izlazni rezultat nakon procjene stanja ispitanika izračunat je na temelju fizioloških podataka intervala nekoliko sekundi unatrag od sadašnjeg trenutka. Uzorci prikupljeni tijekom tih nekoliko sekundi koriste se za izračun značajki kao što su srednja vrijednost, standardna devijacija, minimum, maksimum i nagib. Tako se dobivaju brze procjene emocionalnog stanja ispitanika, što bi bilo teško ostvarivo za čovjeka.

Planke et al.[20] u svom radu navode primjer korištenja sustava koji djeluju u stvarnom vremenu kao pomoć razvoju sustava s adaptivnom asistencijom pilotu. Automatizacija je dobra, no može biti i bolja. Neke od avionskih nesreća djelo su ljudske greške koja je rezultat pristranosti automatizacije i smanjenja sposobnosti letačkih vještina. Stoga je cilj razviti antropocentrične sustave koji neće zamijeniti ljudske vještine, nego pomagati ljudima da budu učinkovitiji i djelotvorniji. Takav sustav je prilagodljiv u stvarnom vremenu, mijenja se ovisno o fiziološkim, bihevioralnim i okolišnim podacima (Slika 3.2). Podaci su objedinjeni kako bi pružili uvid u stvarno stanje mentalnog opterećenja, umora i pažnje kod ispitanika. Da bi takav sustav bio ostvariv potrebne su česte i precizne procjene mentalnog opterećenja u stvarnom vremenu. Podaci na temelju kojih rade algoritmi i modeli procjene prikupljaju se sensorima kao što su EKG, EEG, galvanski odgovor kože, praćenje aktivnosti oka te funkcionalna blisko-infracrvena spektroskopija. Korisne značajke koje se mogu dobiti iz ovih podataka su promjene u θ i α valovima EEG-a, broj treptaja oka, promjer zjenica, promjene u pulsu i disanju. Primjenom strojnog učenja radi se procjena nad podacima dobivenim sa senzora, a nakon toga se korisničko sučelje pilota mijenja ovisno o kognitivnom stanju pilota.

Finseth et al.[7] u svome radu predlažu sustave u stvarnom vremenu koji se koriste za poboljšanje treninga astronauta. Trening astronauta često je fokusiran na njihove performanse, a rijetko se uzimaju u obzir stresne situacije u kojima se astronaut može naći. Integrirati stres u trening može biti izazovno zbog različitih reakcija različitih osoba na stres. Osobe različito reagiraju ovisno o spolu, godinama i prijašnjim iskustvima. Također, teško je zadržati standarde treninga i u isto vrijeme prilagoditi ga načinu na koji osoba doživljava stres. Sustav stoga treba u stvarnom vremenu obraditi fiziološke podatke, procijeniti razinu stresa te optimizirati količinu i intenzitet stresora koji se predoče osobi kroz VR tehnologiju (Slika 3.3).



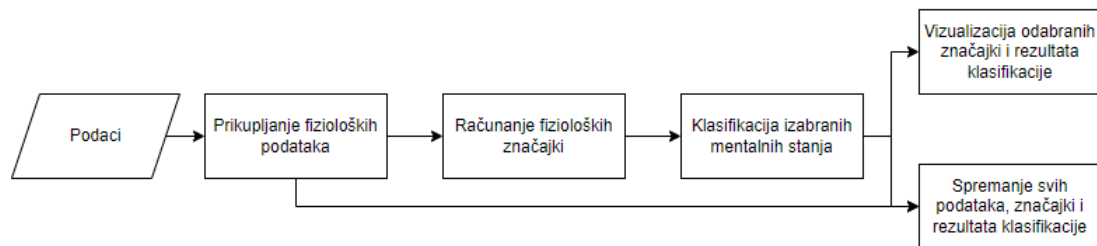
Slika 3.2: Arhitektura sustava sa adaptivnom asistencijom pilotu (preuzeto s [20])



Slika 3.3: Arhitektura adaptivnog sustava za trening astronauta (preuzeto s [7])

4. Oblikovanje simulacije telemetrijskog sustava za praćenje fizioloških indikatora stresnih mentalnih stanja

Način na koji je strukturirana simulacija prikazan je na slici 4.1. Sukladno protočnoj (engl. *pipe-and-filters*) arhitekturi simulacija je sastavljena od različitih dijelova koji mogu raditi u isto vrijeme, a pravokutnici na slici prikazuju logičke dretve sa strelicama koje prikazuju tok podataka. Ova struktura ne samo da poboljšava učinkovitost simulacije omogućavanjem istovremene obrade, već također olakšava modularnost, olakšavajući razumijevanje i modificiranje pojedinačnih komponenti prema potrebi.



Slika 4.1: Idejna konceptualna struktura simuliranog sustava

4.1. Zahtjevi

4.1.1. Funkcionalni zahtjevi

Ovaj sustav mora biti simulacija modularnog telemetrijskog sustava za praćenje fizioloških indikatora stresnih mentalnih stanja. Sustav mora podatke prihvaćati i obrađivati u stvarnom vremenu. Za potrebe ove simulacije stvarni akvizicijski uređaji bit

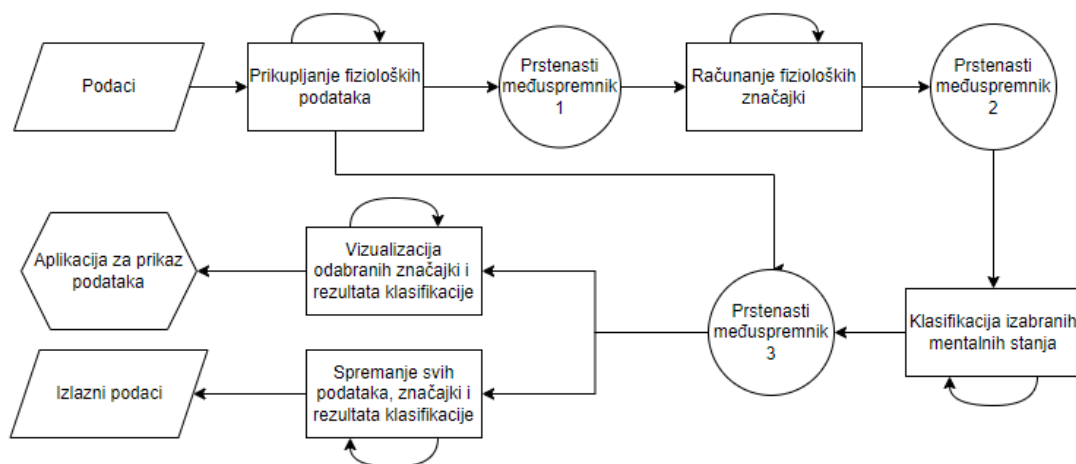
će zamijenjeni s tokovima podataka iz datoteka u kojima se nalaze unaprijed snimljeni stvarni fiziološki signali. Ipak sustav mora biti takav da se datoteke vrlo lako, uz izmjenu nekoliko linija koda, mogu zamijeniti sa stvarnim akvizicijskim uređajima.

Nakon učitavanja podataka sustav treba obraditi podatke i izračunati značajke koje su mu potrebne za klasifikaciju u jedno od 4 kognitivna stanja: normalno, kanalizi-rana pažnja, skrenuta pažnja i prepad/iznenađenje. Cilj nije razviti sustav koji će točno procjenjivati stresno stanje ispitanika, nego je naglasak na brznoj i efikasnoj obradi podataka. Nadalje, sustav treba zatim poslati podatke udaljenom računalu na kojem će se vizualizirati odabrani podaci. Nakon klasifikacije potrebno je i poslati podatke na pohranu u obliku CSV datoteke s vremenskom oznakom.

4.1.2. Nefunkcionalni zahtjevi

Simulirani sustav mora biti sposoban obrađivati podatke u stvarnom vremenu bez značajnih kašnjenja. Rad treba bit podržan za različite količine podataka. Također mora biti otporan na greške te mora točno računati fiziološke značajke koje služe za klasifikaciju. Sustav mora imati sposobnost slanja podataka na udaljenu lokaciju gdje će se podaci vizualizirati. Vizualizacija ne smije biti komplicirana, nego lako čitljiva i prilagođena za sve korisnike neovisno o tehničkoj spretnosti.

4.2. Predložena arhitektura simulacije



Slika 4.2: Arhitektura sustava

Prijenos podataka između klasa odvija se s pomoću prstenastih međuspremnika (engl. *Ring buffer*). Svaka klasa interno pokreće dretvu kroz koju pristupa prstenas-

tom međuspremniku kroz odgovarajuće programersko sučelje (u konkretnom slučaju za ovaj diplomski rad, radi se o sučelju pasivnoga objekta tipa Disruptor) što omogućava brži rad bez većih čekanja. Međuspremnik prima podatke od klase proizvođača i signalizira klasi potrošaču da postoji podatak koji može pročitati. Potrošač može kopirati podatak iz međuspremnika, ali mu može i pristupiti bez nepotrebnog lokalnog kopiranja te izračunati značajke koje su mu potrebne. Također može i pristupiti prijašnjim podacima koji prethode zadnjem podatku koji je dodan u međuspremnik, no u tom slučaju je moguće da je podatak već izgubljen, odnosno preko njega je upisan sljedeći jer je tom podatku ranije pristupljeno.

Podaci se nalaze u CSV datotekama i u sustav ulaze preko klase za prikupljanje podataka. Podaci se pretvaraju u bročano polje kojim se zatim puni "Prstenasti međuspremnik 1" kako bi ga klasa za izračun značajki mogla obraditi. Međuspremnik se puni zapisom u obliku bročanog polja frekvencijom koja odgovara frekvenciji uzorkovanja signala koji se u datoteci nalazi. Time se postiže simulirano prikupljanje podataka koje u smislu vremenske dinamike suštinski odgovara prikupljanju podataka sa stvarnih uređaja za mjerenje fizioloških signala. Čitanje se prvo obavlja nad skupom podataka ranije u datoteci i ti podaci služe kao referentne bazalne vrijednosti koje su potrebne tijekom izračuna značajki u svrhu normalizacije vrlo individualnih fizioloških signala. Nakon toga, sustav nastavlja s normalnim čitanjem i obradom podataka, nad kojima se provodi klasifikacija u izabrana mentalna stanja. Osim što se u ovoj fazi podaci šalju na obradu, u isto ih se vrijeme kroz "Prstenasti međuspremnik 3" šalje na pohranu u CSV datoteku.

Klasa za računanje značajki prima polje podataka iz "Prstenastog međuspremnika 1" te podatke raspoređuje u polja nad kojima se provodi izračun. Naizgled nepotrebno kopiranje podataka u ovoj fazi je neophodno te je ovdje manja korist samog prstenastog međuspremnika. Ovakvo ograničenje nametnuto je bibliotekama koje se koriste za računanje značajki koje očekuju gotovo polje podataka, a takvo polje nije moguće dobiti direktnim pristupom međuspremniku. Da bi se ovo izbjeglo, svaki signal bi trebao imati svoj međuspremnik, a za računanje značajki trebalo bi napisati metode koje efikasnije iteriraju kroz međuspremnik. Polja nad kojima se vrši izračun fiksne su veličine od 2560 vrijednosti i kada se napuni do kraja najstariji podatak se zamjenjuje s novim. Posebno važno je očuvati redosljed podataka kod izračuna značajki EEG-a i EKG-a jer su zapisi u obliku valova pa za ispravnu detekciju redosljed u poljima za izračun treba biti onakav kakav je zapisan u datotekama. Na temelju zastavice, koja je došla uz podatke, prvo se računaju referentne značajke koje se kasnije koriste za normalizaciju fizioloških signala. Nakon toga se vrijednosti u svim poljima za izračun

postavljaju na 0 i polja se pune ispočetka kako bi se dobile nove vrijednosti koje će se koristiti za klasifikaciju. Značajke se računaju svakih pola sekunde, odnosno svakih primljenih 128 podataka. Tako imamo pomični prozor duljine 10 sekundi, odnosno 2560 podataka, koji se pomiče za pola sekunde, odnosno 128 podataka. Nad signalima se računaju značajke na prozoru duljine $W=10$ sekundi:

- F_1^W : srčani ritam izračunat iz EKG-a
- F_2^W : srednja vrijednost elektrodermalne aktivnosti
- F_3^W : standardna devijacija elektrodermalne aktivnosti
- F_4^W : nagib elektrodermalne aktivnosti
- F_5^W : srednja vrijednost disanja
- F_6^W : prosječna snaga vala α iz EEG-a
- F_7^W : prosječna snaga vala β iz EEG-a
- F_8^W : omjer prosječnih snaga valova β i α iz EEG-a
- F_9^W : asimetrija vala α u prefrontalnom režnju iz EEG-a
- F_{10}^W : indeks angažiranosti iz EEG-a

Nakon izračuna značajki puni se novo polje koje se zatim kroz "Prstenasti međuspremnik 2" predaje klasi za klasifikaciju izabranih mentalnih stanja.

Referentnim bazalnim vrijednostima, koje primi klasa za klasifikaciju, puni se polje koje će se koristiti u algoritmu za klasifikaciju izabranih mentalnih stanja. U algoritmu su dva ulazna argumenta. Prvi su referentne vrijednosti označene s F_i^B , a drugi su vrijednosti označene s F_i^W koje predstavljaju prozor od 10 sekundi nad kojima se želi napraviti procjena stresa. Klasifikacija se odvija nakon svakog pristiglog zapisa značajki izračunatih od klase za računanje značajki, to jest svakih pola sekunde. Nadalje se računa *meanRelativeDiff*, parametar koji označava odstupanje od referentnih vrijednosti.

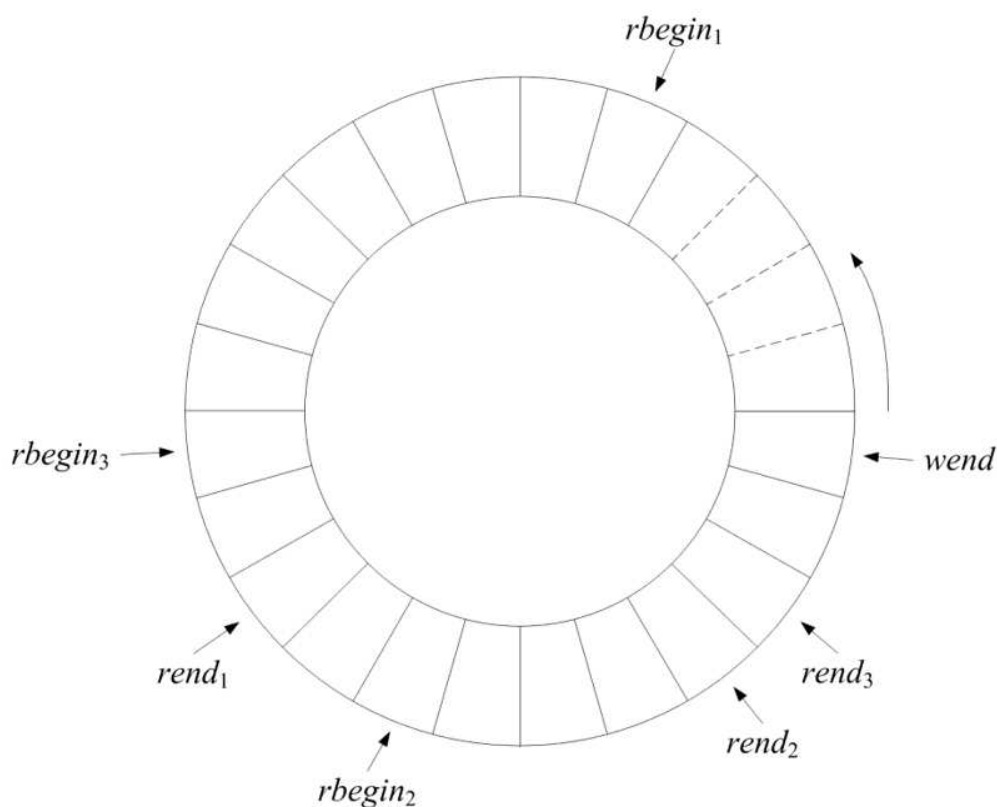
$$\text{meanRelativeDiff} = \frac{1}{n} * \sum_{i=1}^n \frac{F_i^W - F_i^B}{F_i^B} \quad (4.1)$$

Algoritam koji će se koristiti kao ad hoc ilustracija klasifikacijskog algoritma u ovoj simulaciji, odnosi se na četiri izabrana mentalna stanja povezana sa stresom i usmjerenjem pažnje. Algoritam mapira interval s najnižim *meanRelativeDiff* vrijednostima, $< -inf, 0.05]$ uvijek u normalno stanje koje odgovara uvjetima bez stresa i bez podražaja koji induciraju specifične oblike usmjerenja pažnje. Idući interval, $< 0.05, 0.2]$ mapira se u 70% slučajeva u normalno stanje, a preostalih 30% slučajeva ravnopravno

se stohastički mapira na 3 različita stresna mentalna stanja. Idući interval, $< 0.2, 0.35]$, mapira se ravnopravno na 3 stresna mentalna stanja, tj. svako od njih stohastički se pojavljuje u 33% slučajeva. Najviši interval, $[0.35, +inf >$, mapira se u 70% slučajeva na ono mentalno stanje s najvišom razinom stresa, a preostalih 30% slučajeva ravnopravno se podijeli na druga 2 manje stresna stanja koja podrazumijevaju specifične načine usmjeravanja pažnje. Ujedno treba naglasiti da su pravila algoritma u suštini arbitrarna, što je adekvatno za potrebe ovoga rada čiji cilj nije razvoj niti evaluacija klasifikacijskog algoritma koji bi imao nekakav potencijalni praktični značaj.

5. Programski alati za ostvarenje simulacije

Programski kod aplikacije napisan je u Java 17. programskom jeziku. Jedan od ključnih razloga u pozadini izbora ovoga programskog jezika je postojanje pouzdane, ekstenzivno testirane, implementacije specifičnog prstenastog međuspremnik (engl. *ring buffer*), koji se smatra važnom komponentom u protočnoj arhitekturi (engl. *pipe-and-filters architecture*) ovakvih sustava [21] (Slika 5.1).



Slika 5.1: Prstenasti međuspremnik [21]

Ovakav međuspremnik treba omogućiti brzi protok vremenskih serija podataka između jednog proizvođača i više potrošača primjenom:

1. unaprijednog alociranja cjelokupne količine memorije međuspremnika te zabrane dinamičkog rasta ove memorije;
2. izbjegavanja potrebe za kopiranjem podataka od proizvođača u lokalnu memoriju svakoga potrošača prije nego ih potrošač može početi obrađivati na svoj specifični način;
3. minimalnog korištenja mehanizama sinkronizacije proizvođača i potrošača (što bi u idealnom slučaju podrazumijevalo tzv. lock-free implementaciju).

U potrazi za ovakvom strukturom podataka ispostavilo se da ju je implementirala još 2011. godine tvrtka LMAX u programskom jeziku Java za potrebe visokofrekventnog trgovanja te otvorila ovu implementaciju za slobodno korištenje. Implementacija nosi naziv LMAX Disruptor.

5.1. LMAX Disruptor

LMAX grupa je tvrtka osnovana 2010 godina koja se bavi financijskim tehnologijama. Cilj im je biti najbrža platforma za trgovanje. Da bi to postigli morali su izmisliti način na koji bi prenosili podatke uz što manju latenciju i što veću propusnost. Korištenje redova pokazalo se problematičnim zbog visoke latencije pa je kao rezultat istraživanja i testiranja nastao Disruptor.

Disruptor je napisan u Javi za usluge trgovanja, no njegova primjena je moguća u puno više polja i na puno više načina. Tijekom istraživanja i testiranja uočeno je da se puno vremena gubi zbog male priručne memorije procesora te traženja dopuštenja od jezgre operacijskog sustava za pristup određenim dijelovima memorije. Disruptor je stvoren da bolje radi s računalom bez da traži dopuštenja za pristupom od jezgre [16].

5.1.1. Problemi brava i redova

Korištenje brava i zaključavanja pomaže da se osigura da samo jedna osoba radi promjene u isto vrijeme i da se te promjene vide u uređenom redu. Problem nastaje kada dvije osobe žele koristiti istu bravu te se moraju izmjenjivati. Tim izmjenjivanjem upravlja jezgra operacijskog sustava i on blokira radnje dok se brava ne oslobodi. Dok se čeka na bravu moguće je obrađivati druge zadatke, no to može dodatno usporiti rad.

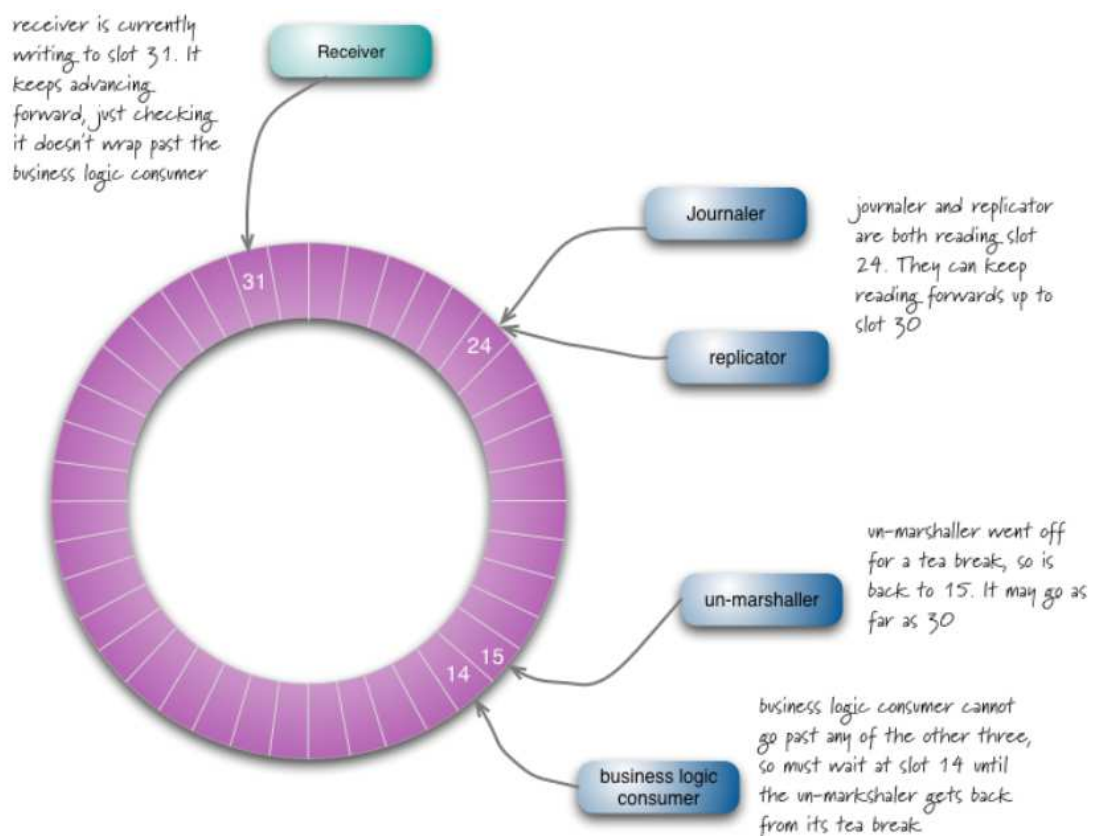
Blokirajući redovi u Javi najčešće koriste povezane liste ili polja za pohranu podataka. U slučaju neograničene memorije ona može neograničeno narasti što dovodi do velikog kvara ako se sva memorija iscrpi, posebno ako je proizvodnja podataka veća

od potrošnje. Kako do toga ne bi došlo potrebno je kontrolirati veličinu reda, a u implementaciji reda čekanja koriste se varijable glava, rep i veličina. Redovi su često ili skroz prazni ili skroz puni zbog neujednačenog rada proizvođača i potrošača. Čak i korištenje mehanizama glave i repa uz brave postoji problem jer oni okupiraju istu priručnu memoriju. Uz sve to u Javi se javlja problem gomilanja smeća [16].

5.1.2. Disruptor kao rješenje

Sva memorija u prstenastom međuspremniku Disruptora je unaprijed dodijeljena, a zbog ograničenja Jave ulazi nisu sami objekti nego pokazivači na objekte. Unaprijed dodijeljenom memorijom riješen je problem smeća jer ulazi će živjeti dok postoji instanca Disruptora. Što je više memorije pridodijeljeno Disruptoru to je veći problem za sakupljača smeća u Javi, no on pridodijeljene ulaze za Disruptor vidi kao besmrtni i tako ga oni ne opterećuju dodatno.

Rad prstenastog međuspremnika prikazan je na slici 5.2, a objašnjen u nastavku. Disruptor ne koristi brave kako bi odredio tko u kojem trenutku smije pristupiti objektu



Slika 5.2: Rad prstenastog međuspremnika (preuzeto sa [8])

u memoriji. Prstenasti međuspremnik je konstruiran tako da proizvođač ako želi upi-

sati nešto u njega mora prije zatražiti slobodnu poziciju. To radi tako da preuzme sekvencijski broj i provjeri jesu li svi potrošači obradili tu poziciju. Barijera na strani proizvođača sprječava da on zapiše u neku poziciju podatak ako ta pozicija još nije obrađena od strane svih potrošača. Na sličan način rade i potrošači. Koristeći barijeru pristupa posljednjoj dostupnoj sekvenci koju je proizvođač proizveo. Barijera osigurava da potrošač ne pristupa sekvenci koju proizvođač nije u potpunosti zapisao. Nakon što se pročita pozicija, potrošač ažurira sekvencijski broj te tako označava da je spreman za čitanje sljedeće pozicije kada ona bude spremna, a barijera označava tu poziciju slobodnom za ponovno prepisivanje od strane proizvođača. Vrijednosti varijable se zapisuju u glavnu memoriju te je tako spriječeno da različite dretve vide različite vrijednosti iste varijable. Na taj se način smanjuje vrijeme i memorija potrebna za obrađivanje zadatka.

Sve ovo dovodi do višestrukog ubrzanja i smanjenja latencije kao što je vidljivo na slici 5.3.

	Array Blocking Queue (ns)	Disruptor (ns)
Min Latency	145	29
Mean Latency	32,757	52
99% observations less than	2,097,152	128
99.99% observations less than	4,194,304	8,192
Max Latency	5,069,086	175,567

Slika 5.3: Usporedba latencija(preuzeto sa [16])

6. Programsko ostvarenje simulacije

6.1. Korišteni skup podataka snimljenih fiziološkim senzorima

Skup podataka koji je korišten preuzet je s NASA-ine stranice [18]. Podaci su prikupljeni u istraživanju koje je provedeno nad parovima pilot/kopilot i za vrijeme leta u simulatoru, ali i u kontroliranim uvjetima u svrhu razvoja modela strojnog učenja za detekciju pažnje pilota. Podaci nevezani uz letačke zadatke su prikupljeni u kontroliranim eksperimentima izvan simulatora leta, dok je testni skup podataka prikupljen kroz cijeli let u simulatoru.

Piloti su bili podvrgnuti smetnjama s ciljem izazivanja jednog od 3 kognitivna stanja:

Kanalizirana pozornost (engl. *Channelized attention*) - fokusiranost samo na jedan zadatak

Skrenuta pozornost (engl. *Diverted attention*) - stanje u kojem je pozornost skrenuta radnjama ili misaonim procesima povezanim s odlukom.

Prepad/iznenađenje (engl. *Startle/Surprise*) - stanje koje se izazove nastupom iznenadnih stresnih situacija tijekom simuliranog leta

Podaci nevezani uz letačke zadatke su se dobili kroz 3 eksperimenta kako bi se dobile vrijednosti za svako od 3 navedena stanja. U svakom eksperimentu piloti su mogli osjetiti samo po jedno stanje. Testni podaci prikupljeni su kroz puni let u simulatoru i piloti su mogli osjetiti bilo koje od 3 stanja, ali nikada više od jednog istovremeno.

Svi senzori su bilježili podatke frekvencijom 256 Hz. Zabilježeno je 20 EEG signala, te signali disanja, znojenja i EKG-a. Podaci EKG-a, disanja i znojenja prikupljeni su NeXuS-10 sustavom.

6.2. Relevantni dijelovi programskog koda

Sustav je zamišljen tako da sadrži 3 prstenasta međuspremnik koji će služiti za komunikaciju 5 klasa koje će odrađivati posao obrade podataka kao što je prikazano na slici 4.2. Klasa za prikupljanje najprije inicijalizira 3 Disruptora i njima pripadajuće prstenaste međuspremnik:

```
Disruptor<EventWrapper> disruptor =  
new Disruptor<>(EventWrapper.EVENT_FACTORY,  
                1024,  
                DaemonThreadFactory.INSTANCE,  
                ProducerType.SINGLE,  
                new BusySpinWaitStrategy());
```

```
RingBuffer<EventWrapper> ringBuffer =  
    disruptor.getRingBuffer();
```

Prva 2 međuspremnik imat će po jednog proizvođača pa se u inicijalizaciji postavlja argument `ProducerType.SINGLE`. "Prstenasti međuspremnik 3" će imati 2 proizvođača i inicijalizira se argumentom `ProducerType.MULTI`. Klasa za prikupljanje podataka će u njega zapisivati ulazne vrijednosti da bi se one mogle spremi, dok će klasa za klasifikaciju zapisivati ranije izračunate značajke te rezultat klasifikacije. Klase za računanje značajki, procjenu stresa, vizualizaciju i spremanje podataka implementiraju sučelje `EventHandler` iz LMAX Disruptor biblioteke. Tako klase koje implementiraju to sučelje moraju implementirati apstraktnu metodu

```
onEvent(EventWrapper event, long sequence, boolean  
        endOfBatch)
```

Metoda se okida svaki put kada u međuspremnik dođe novi podatak. Za svaki od tih `EventHandler`-a postoji pripadajući `Event`, odnosno klasa koja implementira sučelje `Event`:

```
public interface Event {  
    int getType();  
    double[] getMessage();  
    void setMessage(double[] message);  
    Double getTimestamp();  
    void setTimestamp(Double timestamp);  
}
```

Pripadajući Event kreira se kako bi se podaci mogli poslati kroz međuspremnik do željenog EventHandlerera.

Da bi klase mogle međusobno komunicirati potrebno je inicijalizirati *handler-e* koji će prihvaćati i obrađivati podatke koji budu objavljeni u međuspremniku. Pri inicijalizaciji potrebno je i predati argument izlaznog međuspremnika ako je potrebno. Za `VisualizationHandler` i `StorageHandler` to neće biti potrebno. Nakon inicijalizacije potrebno je i svaki *handler* dodijeliti pripadajućem *disruptor-u*. U ovom slučaju nije potrebno inicijalizirati *handler* za klasu koja prikuplja podatke jer ona ne prima podatke iz međuspremnika nego iz CSV datoteka:

```
FeatureComputationHandler featureComputationHandler =
    new FeatureComputationHandler(
        featureComputationBuffer);
ClassificationHandler classificationHandler =
    new ClassificationHandler(classificationBuffer);
VisualizationHandler visualizationHandler =
    new VisualizationHandler();
StorageHandler storageHandler =
    new StorageHandler();

disruptor1.handleEventsWith(featureComputationHandler);
disruptor2.handleEventsWith(classificationHandler);
disruptor3.handleEventsWith(visualizationHandler,
                             storageHandler);
```

6.2.1. Prikupljanje fizioloških podataka

Podatke koje klasa za prikupljanje pročita šalje u `ringBuffer1` koji je zapravo "Prstenasti međuspremnik 1". Podaci se šalju jedan po jedan, odnosno jedan po jedan zapis podataka uz `Thread.sleep(4)` nakon svakog slanja. Prvo se dobije sekvencijski broj međuspremnika, odnosno prva slobodna pozicija u koju je moguće staviti podatak i `EventWarpper` koji pripada toj poziciji:

```
long seqId = ringBuffer1.next();
EventWrapper eventWrapper = ringBuffer1.get(seqId);
```

Nakon toga stvara se `FeatureComputationEvent` željenog tipa. Podatak stavljamo u `event` i postavljamo vremensku oznaku:

```

FeatureComputationEvent event =
    new FeatureComputationEvent ();
event . setMessage ( zapis );
event . setTimestamp ( zapis [ 1 ] );

```

Za kraj `eventWrapper`-u postavljamo `EventType` kako bi ga `EventHandler` kojem šaljemo prepoznao. Ako šaljemo referentne bazne vrijednosti argument će biti `EventType.BASE`, dok za slanje vrijednosti nad kojima se vrši klasifikacija će argument biti `EventType.CLASSIFICATION`. `eventWrapper`-u postavljamo `event` kojeg smo ranije kreirali i objavljujemo ga `ringBuffer1` na `seqId`.

```

eventWrapper . setType ( EventType . BASE . getCode ( ) );
eventWrapper . setEvent ( event );
ringBuffer1 . publish ( seqId );

```

Isto se napravi i za slanje vrijednosti klasi za pohranu podataka, samo što se neće slati kroz `ringBuffer1`, nego `ringBuffer3` koji u kodu predstavlja "Prstenasti međuspremnik 3".

6.2.2. Računanje fizioloških značajki

Nakon što je `event` postavljen u međuspremnik, u klasi za računanje značajki, automatski se poziva metoda `onEvent` koja kao argument prima postavljeni `event` te se preuzima poruka iz njega:

```

event . getEvent ( ) . getMessage ( )

```

Ovdje je ključno da klasa na temelju `EventType`-a prepozna jesu li pristigle vrijednosti referentne ili one nad kojima se vrši klasifikacija. U slučaju kada su to referentne vrijednosti polje veličine 2560 se jednom napuni i nad njime se izračunaju odabrane značajke. Kada klasa za prikupljanje podataka pošalje sve referentne vrijednosti, sljedeći `event` bit će tipa `EventType.CLASSIFICATION` te će se brojač koji označava da je polje puno i spremno za računanje postaviti na 1 kako bi se polje punilo ispočetka. Također uvjetom:

```

if ( halfSecond % 128 == 0 )

```

osigurava se da se značajke računaju svakih pola sekunde, odnosno nakon svakog 128. pristiglog podatka. Njega također treba postaviti na 0 kako se ne bi računale značajke nad nepotpunim poljem vrijednosti.

Koristeći biblioteke treće strane koje su dostupne u Javi se računaju značajke. Za računanje značajki disanja i elektrodermalne aktivnosti koristi se `DescriptiveStatistics` klasa iz Apache Commons Math biblioteke. Izračun pulsa radi se s pomoću biblioteke JELY (Java Ecg LibrarY). JELY koristi QRS detektora za otkrivanje QRS kompleksa u EKG zapisima. Ima mogućnost izravnog učitavanja PhysioNET, binarnih CustoMeds i CSV datoteka. Biblioteka Brainflow koristi se za računanje prosječnih snaga valova, a nakon toga se s tim vrijednostima računaju odabrane značajke. Značajke se računaju na vremenskom prozoru veličine 10 sekundi.

– F_1^W : srčani ritam računa se na EKG signalu.

```
private Ecg ecg = new Ecg(256,
    new LeadConfiguration(EcgLead.UNKNOWN),
    new SubjectInfo(),
    10);
private QrsDetector det =
    new ElgendiFastQrsDetector(ecg);
private HeartbeatDetector detector =
    new HeartbeatDetector(ecg, det);
ecg.addSampleValue(EcgLead.UNKNOWN,
    event.getMessage()[22]);
heartbeat = detector.findHeartbeats().size()*6
```

Prilikom inicijalizacije samog `FeatureComputationHandler`-a odvija se i inicijalizacija varijabli `Ecg` i `QrsDetector` koje detektiraju QRS komplekse i vraćaju broj QRS kompleksa na prozoru od 10 sekundi. Pri inicijalizaciji `Ecg` objekta bitno je predati frekvenciju uzorkovanja kao prvi argument, te veličinu prozora kao četvrti. Vrijednosti se dodaju nakon svakog primljenog *event*-a, a svakih pola sekunde izračuna se broj QRS kompleksa, a zatim se dobivena vrijednost pomnoži sa 6 kako bi se dobio broj otkucaja u minuti.

– F_2^W : srednja vrijednost elektrodermalne aktivnosti.

```
DescriptiveStatistics gsrDS =
    new DescriptiveStatistics(2560);
gsrDS.addValue(event.getMessage()[24]);
srVrZnojenja = gsrDS.getMean();
```

Kod inicijalizacije objekta se predaje veličina prozora od 2560 podataka što odgovara vremenskom prozoru od 10 sekundi. Zatim se dodaje vrijednost za-

primljena iz međuspremnik i na kraju se svakih pola sekunde računa srednja vrijednost metodom iz biblioteke Apache Commons Math3.

- F_3^W : standardna devijacija elektrodermalne aktivnosti- Inicijalizacija i dodavanje vrijednosti radi se na isti način kao i za F_2^W , a rezultat se dobije pozivom metode:

```
stDevDisanja = respirationDS . getStandardDeviation ();
```

- F_4^W : nagib elektrodermalne aktivnosti računa se kao razlika maksimalne i minimalne vrijednosti na prozoru podijeljeno s veličinom prozora.

```
double calculateSlope(double max, double min) {  
    return (max - min) / this.gsrTime;  
}
```

- F_5^W : srednja vrijednost disanja se računa na isti način kao i značajka F_2^W .
- F_6^W : prosječna snaga vala α iz EEG-a računa se sljedećim blokom koda:

```
Pair<double [], double []> parovi =  
    DataFilter . get_avg_band_powers ( signal ,  
        elektrode ,  
        frekvencijaUzorkovanja ,  
        primjeniFiltere );  
double [] valovi = parovi . getLeft ();  
alphaPower = valovi [2];
```

Metoda `get_avg_band_powers` koristi se u izračunu svih značajki, tj. u dobivanju prosječnih snaga valova koje se zatim koriste za računanje traženih značajki. Prvi argument je signal kao polje podataka nad kojim se računa prosječna snaga, a drugi polje popunjeno jedinicama, čija veličina odgovara broju elektroda a kojih se računa tražena značajka. Na primjer, za računanje α vala koristimo signale sa 6 elektroda pa će argument *elektorde* biti polje koje sadrži 6 jedinica. Treći argument je frekvencija uzorkovanja koja je u ovoj simulaciji 256 Hz, a zadnji je *boolean* vrijednost koja označava treba li primijeniti filtere nad signalom nad kojim se računaju prosječne snage, u ovom slučaju ta vrijednost bit će *false*. Metoda vraća parove (srednja vrijednost, standardna devijacija) za svaki val, a Prosječna snaga α vala je u polju *valovi* na indeksu 2.

- F_7^W : prosječna snaga vala β iz EEG-a dobije se iz polja *valovi* s indeksa 4.

- F_8^W : omjer prosječnih snaga valova β i α iz EEG-a:

$$\text{betaAlfaOmjer} = \text{valovi}[4] / \text{valovi}[2];$$
- F_9^W : asimetrija vala α u prefrontalnom režnju iz EEG-a:

$$\text{indeksAsimetrije} = \text{Math.log}(\text{valoviLijevo}[2]) - \text{Math.log}(\text{valoviDesno}[2]);$$
- F_{10}^W : indeks angažiranosti iz EEG-a:

$$\text{indeksAngaziranosti} = \text{valovi}[4] / (\text{valovi}[1] + \text{valovi}[2]);$$

Nakon što se sve značajke izračunaju one se pune u polje, objavljuju *ringBuffer2* te šalju klasi za klasifikaciju na isti način kao na početku odjeljka 6.2.1.

6.2.3. Klasifikacija izabranih mentalnih stanja

Klasa za klasifikaciju prima poruku iz međuspremnika te na temelju algoritma i njemu predanih, ranije izračunatih, značajki procjenjuje u kojem se od 4 stanja nalazi osoba. Klasa prima dva tipa vrijednosti. Najprije primi referentne bazne vrijednosti koje prepoznaje po *event*-u tipa `EventType.BASE` te ih spremi u polje kao F_i^B . Vrijednosti za klasifikaciju pristignu u *event*-u tipa `EventType.CLASSIFICATION` i one predstavljaju F_i^W . Nakon primitka podataka za klasifikaciju formulom 4.1 izračuna se *meanRelativeDiff*. Ovisno o dobivenoj vrijednosti algoritam opisan na kraju odjeljka 4.2 klasificira u jedno od 4 odabrana mentalna stanja.

Rezultat klasifikacije objavljuje se "Prstenastom međuspremniku 3" s podacima nad kojima je napravljena klasifikacija. Na isti način kako je ranije opisano, izlaznom međuspremniku objavi se *event* kojeg zatim primaju `VisualizationHandler` i `StorageHandler`.

6.2.4. Spremanje i vizualizacija

Spremanje se odvija tako da kada podaci pristignu na `StorageHandler` provjeri se vrsta *event*-a. U slučaju da je vrsta *event*-a `EventType.ENTRY` podaci se pohranjuju u `entryValues.csv` s vremenskom oznakom. U slučaju da je vrsta *event*-a `EventType.CLASSIFICATION` podaci se pohranjuju u `result.csv`. Prethodno se pri inicijalizaciji `StorageHandler`a zapiše zaglavlje u kojem se nalaze nazivi značajki, odnosno podataka koji su pročitani iz ulazne CSV datoteke.

Vizualizacija se odvija na udaljenom računalu, a podaci se šalju koristeći RestTemplate iz okruženja Spring.

```
restTemplate . postForObject ( url ,  
                                event . getEvent () . getMessage () ,  
                                String . class );
```

Aplikacija za vizualizaciju napravljena je kao jednostavna jednostranična web aplikacija pomoću Thymeleafa. Važno je napomenuti da VisualizationHandler obrađuje samo *event*-e tipa EventType.CLASSIFICATION jer nema smisla prikazivati ulazne podatke bez konteksta koje klasa za prikupljanje objavi na "Prstenasti međuspremnik 3".

7. Rezultati

Cilj ovog rada nije bio napraviti sustav koji će na temelju podatka s velikom preciznošću procjenjivati razinu stresa nego sustav koji će brzo i efikasno računati značajke na temelju kojih se procjenjuje razina stresa. Sustav kao rezultat ovog rada, uz korištenje prstenastih međuspremnik LMAX Disruptor biblioteke, ima veliku brzinu.

Stranica za vizualizaciju sastoji se od indikatora angažiranosti/stresa i 8 grafova koji pokazuju izračunate značajke. Indikator angažiranosti mijenja se ovisno o izračunu promatranih mentalnih stanja iz indikatora (Slika 7.1).

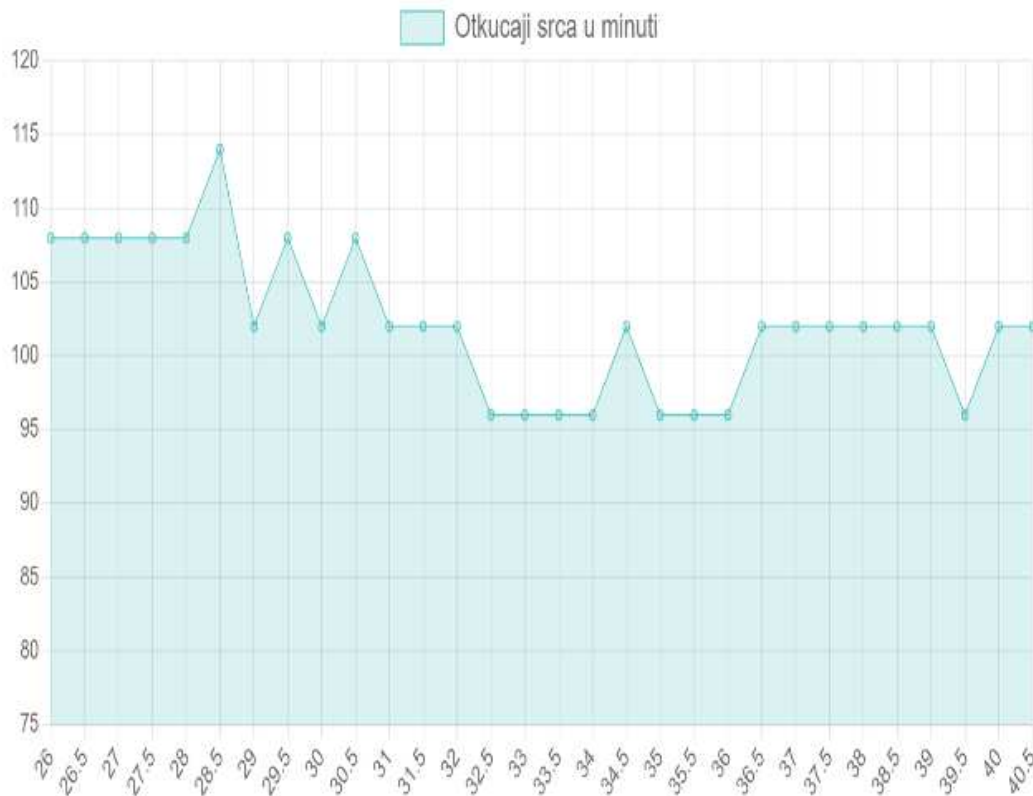
Indikator razine angažiranosti



Slika 7.1: Indikator angažiranosti

Podaci za prikaz na grafovima dolaze svakih pola sekunde. Na x osi nalazi se vrijeme proteklo od početka promatranja, a na y-osi nalaze se vrijednosti koje odgovaraju grafu. Maksimalne i minimalne vrijednosti y-osi mijenjaju se na grafu ovisno o vrijednostima koje aplikacija primi tako da se svi podaci pokažu na grafu uz najbolju moguću vidljivost. Također maksimalni vremenski prozor koji se prikazuje je 15 sekundi, pa se on pomiče kroz vrijeme kako bi prikazao najnovije podatke (Slika 7.2).

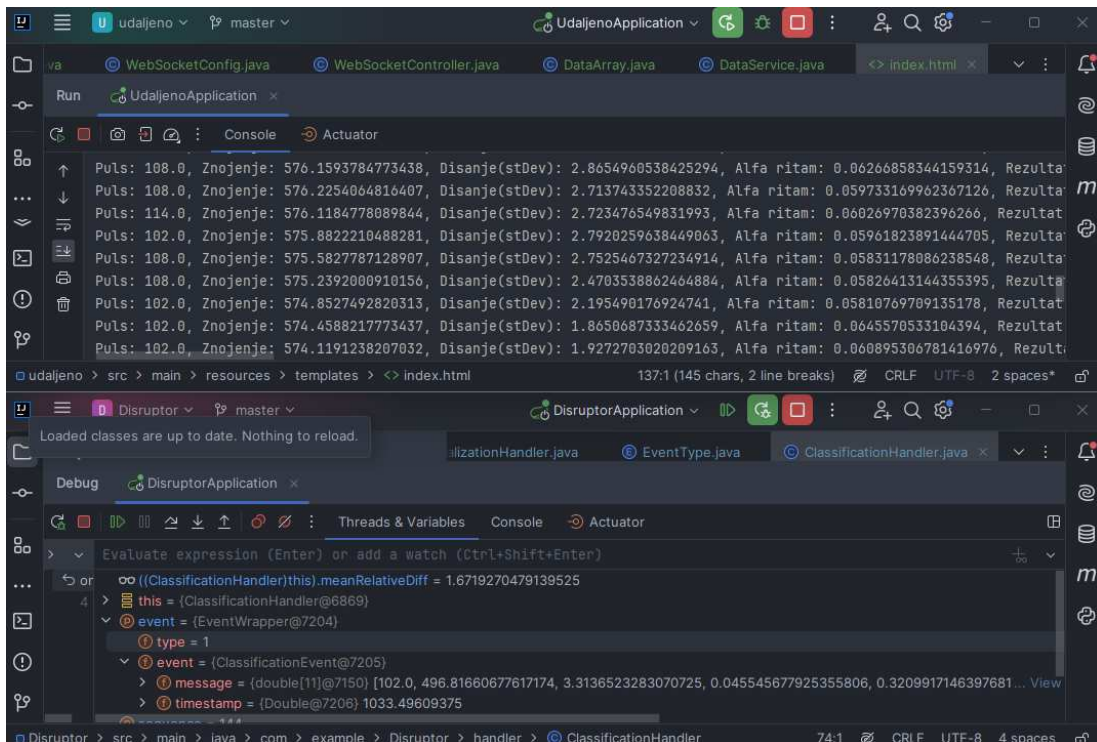
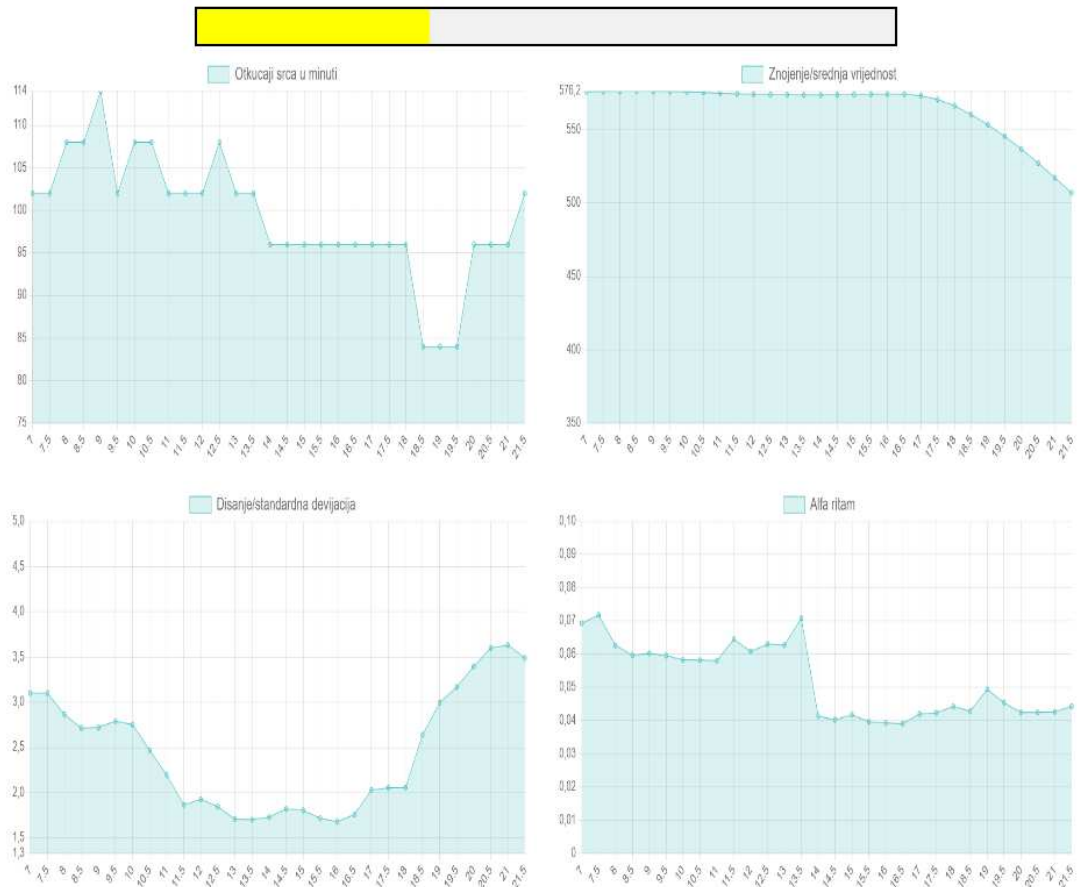
Simulacija je pokretana i testirana na računalu sa AMD Ryzen 7 5800H procesorom, 16 GB radne memorije i NVIDIA GeForce RTX 3060 grafičkom karticom. Kao primjer, izabrana je datoteka iz NASA-inog podatkovnog skupa s podacima tijekom



Slika 7.2: Graf podataka

simulatorske vježbe za ispitanika broj 5. Prozor za izračun referentnih bazalnih vrijednosti postavljen je od 942. sekunde do 1002. sekunde, simulacija rada telemetrijskog sustava u stvarnom vremenu pokrenuta je u 1002. sekundi, a u 1027. sekundi nastupa početak prepada/iznenađenje događaja. Na slici 7.3 može se vidjeti pokrenuta simulacija i njezin prikaz na web aplikaciji. Oko 15. sekunde na grafu, podaci koji ulaze u sustav pripadaju prepadu/iznenađenje događaju. Nekoliko sekundi nakon toga na grafovima se mogu primijetiti promjene. Puls raste kao i standardna devijacija disanja. Alfa ritam je na početku povišen i to označava opušteno stanje, no oko 15. sekunde može se vidjeti pad snage alfa valova označavajući da je osoba izložena prepadu/iznenađenje događaju. Na donjem dijelu iste slike može se vidjeti konzolni ispis podataka koji prolaze kroz sustav, te aplikacija zaustavljena u debug načinu gdje se mogu vidjeti vrijednosti *meanRelativeDiff*, *timestamp* i poruku koja se nalazi u *event*-u.

Indikator razine angažiranosti



Slika 7.3: Pokrenuta simulacija

Na slici 7.4 može se s lijeve strane vidjeti izvadak podataka koji se nalaze u ulaznoj CSV datoteci, prikazana je vremenska oznaka, signal disanja, znojenja i Event kojem pripada trenutni zapis. Na desnoj strani se vide podaci koje je sustav pročitao i spremio u novu CSV datoteku, prikazani su signal disanja i znojenja kože.

TS	R	BSR	Event	R	BSR	
1	1027.0	643.866028	575.471985	6401	643.866028	575.471985
2	1027.00390625	643.866028	575.471985	6402	643.866028	575.471985
3	1027.0078125	643.866028	575.471985	6403	643.866028	575.471985
4	1027.01171875	643.866028	575.471985	6404	643.866028	575.471985
5	1027.015625	643.866028	575.471985	6405	643.866028	575.471985
6	1027.01953125	643.866028	575.471985	6406	643.866028	575.471985
7	1027.0234375	643.866028	575.471985	6407	643.866028	575.471985
8	1027.02734375	643.866028	575.471985	6408	643.866028	575.471985
9	1027.03125	643.866028	575.471985	6409	643.866028	575.471985
10	1027.03515625	643.866028	575.471985	6410	643.866028	575.471985
11	1027.0390625	643.866028	575.471985	6411	643.866028	575.471985
12	1027.04296875	643.866028	575.47699	6412	643.866028	575.47699
13	1027.046875	643.866028	575.47699	6413	643.866028	575.47699
14	1027.05078125	643.8229980000001	575.47699	6414	643.8229980000001	575.47699
15	1027.0546875	643.8229980000001	575.47699	6415	643.8229980000001	575.47699
16	1027.05859375	643.8229980000001	575.47699	6416	643.8229980000001	575.47699
17	1027.0625	643.606995	575.478027	6417	643.606995	575.478027
18	1027.06640625	643.606995	575.478027	6418	643.606995	575.478027
19	1027.0703125	643.606995	575.478027	6419	643.606995	575.478027
20	1027.07421875	643.3319700000002	575.4849849999998	6420	643.3319700000002	575.4849849999998
21	1027.078125	643.3319700000002	575.4849849999998	6421	643.3319700000002	575.4849849999998
22	1027.08203125	643.3319700000002	575.4849849999998	6422	643.3319700000002	575.4849849999998
23	1027.0859375	643.044983	575.489014	6423	643.044983	575.489014
24	1027.08984375	643.044983	575.489014	6424	643.044983	575.489014
25	1027.09375	643.044983	575.487976	6425	643.044983	575.487976
26	1027.09765625	643.044983	575.487976	6426	643.044983	575.487976
27	1027.1015625	643.044983	575.487976	6427	643.044983	575.487976
28	1027.10546875	643.044983	575.487976	6428	643.044983	575.487976
29	1027.109375	643.044983	575.487976	6429	643.044983	575.487976
30	1027.11328125	643.044983	575.487976	6430	643.044983	575.487976
31	1027.1171875	643.044983	575.487976	6431	643.044983	575.487976

Slika 7.4: Ulazni(lijevo) i izlazni(desno) podaci

Za isti vremenski interval u result.csv datoteku zapišu se odabrane značajke te rezultati klasifikacije. Na slici 7.5 mogu se vidjeti izračunate značajke koje su prikazane na grafovima sa slike 7.3. Uz to vide se i rezultati klasifikacije u "Event" stupcu. Rezultati klasifikacije ne odgovaraju ulaznim Eventima jer simulacija koristi algoritam koji ima arbitrarna pravila i cilj ovoga rada nije bio razviti algoritam koji će biti potpuno točan.

Testiranjem popunjenosti međuspremnik došlo se do zaključka da su međuspremnici većinu vremena prazni i slobodni za upisivanje novih podataka. Do tog zaključka se došlo postavljeno spavanje dretve za prikupljanje podataka od 4 milisekunde. To znači da nigdje nema zastajkivanja i da sustav bez problema prima i obrađuje 256 podataka u sekundi. U slučaju kada nema spavanja dretve sustav radi puno brže od stvarnog vremena i učitava puno više od 256 podataka u sekundi, oko 6000. U tom slučaju broj slobodnih pozicija u međuspremnicima se mijenja i nisu cijelo vrijeme sve pozicije slobodne. Kao što je prikazano na slici 7.6 mogući su manji zastoji na me-

đuspremnicima 1 i 3, dok je međuspremnik 2 većinu vremena u potpunosti slobodan. Ovaj ispis dobio se ispisom kapaciteta svakih 6 sekundi.

	vremenska oznaka	puls	znojenjeSrVr	disanjeStDev	alfa	event
31	1026.99609375	96.0	574.1022298757813	1.7172871665439846	0.039751499074600065	1.0
32	1027.49609375	96.0	574.2514424316406	1.6789260200956755	0.03930891670326098	1.0
33	1027.99609375	96.0	574.14101443125	1.7595344663653077	0.03913038122946338	2.0
34	1028.49609375	96.0	573.069231190625	2.029426983447687	0.04201041711704958	1.0
35	1028.99609375	96.0	570.5108454480469	2.0547925561019915	0.04242806426534912	1.0
36	1029.49609375	96.0	566.2341529578125	2.0569736573756487	0.04429795175871374	1.0
37	1029.99609375	84.0	560.3013201335938	2.639089439890193	0.042898318544912994	1.0
38	1030.49609375	84.0	553.2558311593751	2.9947354375193296	0.04947471488170889	2.0
39	1030.99609375	84.0	545.4715163191405	3.1696825476064183	0.04546453159802525	1.0
40	1031.49609375	96.0	536.8121364390626	3.397693604333075	0.042522727730878455	1.0
41	1031.99609375	96.0	527.2845457660156	3.6007296658500283	0.042524091625284914	2.0
42	1032.49609375	96.0	517.2457681703127	3.6324584551281807	0.042677598837233065	1.0
43	1032.99609375	102.0	507.0026844414062	3.491658916465785	0.044359146836765596	1.0
44	1033.49609375	102.0	496.81660677617174	3.3136523283070725	0.045545677925355806	5.0
45	1033.99609375	108.0	486.9268358125026	3.2533209433057855	0.046671897117354845	1.0
46	1034.49609375	114.0	477.19366021015594	3.357552918323025	0.04746411759451164	1.0
47	1034.99609375	108.0	467.36987313046865	3.2668080349791624	0.04553108454187445	1.0
48	1035.49609375	120.0	457.37714127968746	3.0893425301106356	0.05026621229104975	1.0
49	1035.99609375	114.0	447.3077359929686	2.7246822801233237	0.054215908556252114	1.0
50	1036.49609375	114.0	437.4145143148436	2.5350848026872987	0.047668965143559824	1.0
51	1036.99609375	120.0	427.72766161914086	2.2823295235001844	0.04990139957702363	1.0
52	1037.49609375	114.0	418.08033635664066	1.9118190121080423	0.05343536041693444	2.0
53	1037.99609375	114.0	408.63787110507803	1.7016976267480273	0.06405793133468711	1.0
54	1038.49609375	114.0	400.24316207656244	1.657854824007302	0.05850064478638131	5.0
55	1038.99609375	114.0	393.5864748195313	1.4598486291948605	0.04645159677276851	1.0
56	1039.49609375	114.0	388.6344081109375	1.4014562253696747	0.046060932089697304	1.0
57	1039.99609375	126.0	385.46932415624997	1.305091199551346	0.04621338478262162	2.0
58	1040.49609375	132.0	383.70677120859375	1.2543847259105454	0.040169542136640216	5.0

Slika 7.5: Izvadak iz result.csv

```

RingBuffer1 preostali kapacitet: 1024/1024
RingBuffer2 preostali kapacitet: 1024/1024
RingBuffer3 preostali kapacitet: 1024/1024
-----
RingBuffer1 preostali kapacitet: 1024/1024
RingBuffer2 preostali kapacitet: 1024/1024
RingBuffer3 preostali kapacitet: 511/1024
-----
RingBuffer1 preostali kapacitet: 508/1024
RingBuffer2 preostali kapacitet: 1024/1024
RingBuffer3 preostali kapacitet: 73/1024
-----
RingBuffer1 preostali kapacitet: 763/1024
RingBuffer2 preostali kapacitet: 1024/1024
RingBuffer3 preostali kapacitet: 602/1024
-----
RingBuffer1 preostali kapacitet: 250/1024
RingBuffer2 preostali kapacitet: 1024/1024
RingBuffer3 preostali kapacitet: 124/1024

```

Slika 7.6: Promjene slobodnih pozicija međuspremnika

8. Zaključak

U ovom radu predstavljen je modularni pristup simulaciji telemetrijskog sustava za praćenje fizioloških stresnih mentalnih stanja. Modularni pristup potpomognut je LMAX Disruptor bibliotekom koja korištenjem kružnih međuspremnikâ omogućava brzu i efikasnu obradu podataka u realnom vremenu. Cilj simulacije bio je omogućiti sustavu da prikuplja, obrađuje i analizira velike količine podataka, kao što su otkucaji srca, elektrodermalna aktivnost, disanje i moždana aktivnost. Ujedno treba naglasiti da su pravila algoritma za detekciju stresnih mentalnih stanja u suštini arbitrarna, što je adekvatno za potrebe ovoga rada čiji cilj nije razvoj niti evaluacija klasifikacijskog algoritma koji bi imao nekakav potencijalni praktični značaj. Korištenje LMAX Disruptora pokazalo se efikasnim zbog niske latencije i visoke propusnosti u obradi podataka. Zbog modularnog pristupa sustav je fleksibilan, efikasan i skalabilan pa je uz manje izmjene moguće dodati nove senzore kao izvore podataka i nove module za obradu podataka. Rezultati simulacije pokazali su da je sustav sposoban precizno pratiti promjene u fiziološkim stanjima ispitanika, a daljnji razvoj aplikacije mogao bi uključivati razvoj preciznijeg algoritma za detekciju stresnih mentalnih stanja korištenjem strojnog učenja i umjetne inteligencije. Također je moguće dodatno smanjiti nepotrebna kopiranja podataka razvojem metoda za izračun fizioloških značajki koje bolje iskorištavaju arhitekturu kružnog međuspremnikâ u LMAX Disruptoru.

LITERATURA

- [1] U Rajendra Acharya, S Vinitha Sree, Peng Chuan Alvin Ang, Ratna Yanti, i Jajit S Suri. Application of non-linear and wavelet based features for the automated identification of epileptic eeg signals. *International journal of neural systems*, 22 (02):1250002, 2012.
- [2] Rene S. Skukies; Elian E. Jentoft; Olga Asko. University of oslo, 2020. URL www.sv.uio.no/psi/english/research/projects/human-time-data/documents/data-lifecycle/eeg/2.%20electrode-placement/. Pristupljeno: 2024-8-05.
- [3] BrainWork. Neurofeedback, 2023. URL <https://brainwork.hr/neurofeedback/>. Pristupljeno: 2024-8-05.
- [4] Susan R Burchfield. The stress response: A new perspective. *Psychosomatic Medicine*, 41(8):661–672, 1979.
- [5] Rebecca L Charles i Jim Nixon. Measuring mental workload using physiological measures. *Applied Ergonomics*, 2012.
- [6] Krešimir Čosić, Siniša Popović, Davor Kukolja, Marko Horvat, i Branimir Droljić. Physiology-driven adaptive virtual reality stimulation for prevention and treatment of stress related disorders. *CyberPsychology, Behavior, and Social Networking*, 13(1):73–78, 2010.
- [7] Tor Finseth, Michael C Dorneich, Nir Keren, Warren D Franke, i Stephen Varde-man. Virtual reality adaptive training for personalized stress inoculation. *Human Factors*, stranica 00187208241241968, 2024.
- [8] Martin Fowler. The lmax architecture, 2011. URL <https://martinfowler.com/articles/lmax.html>. Pristupljeno: 2024-8-05.

- [9] Giorgos Giannakakis, Dimitris Grigoriadis, Katerina Giannakaki, Olympia Simantiraki, Alexandros Roniotis, i Manolis Tsiknakis. Review on psychological stress detection using biosignals. *IEEE TRANSACTIONS ON AFFECTIVE COMPUTING*, 2022.
- [10] Takuto Hayashi, Eika Okamoto, Haruhiko Nishimura, Yuko Mizuno-Matsumoto, Ryouhei Ishii, i Satoshi Ukai. Beta activities in eeg associated with emotional stress. *International Journal of Intelligent Computing in Medical Sciences & Image Processing*, 3(1):57–68, 2009.
- [11] Jennifer Kavanagh. *Stress and performance: A review of the literature and its applicability to the military*. Rand, 2005.
- [12] Jeansok J Kim i David M Diamond. The stressed hippocampus, synaptic plasticity and lost memories. *Nature Reviews Neuroscience*, 3(6):453–462, 2002.
- [13] Pavithra Kumari i PS Aithal. Stress inducing factors and relevant strategies deployed to overcome stress in the aviation industry sector—a systematic literature review and further research agendas. *International Journal of Management, Technology, and Social Sciences (IJMTS)*, 5(2):347–371, 2020.
- [14] Jeff T Larsen, Gary G Berntson, Kirsten M Poehlmann, Tiffany A Ito, i John T Cacioppo. The psychophysiology of emotion. *Handbook of emotions*, 3:180–195, 2008.
- [15] Richard S Lewis, Nicole Y Weekes, i Tracy H Wang. The effect of a naturalistic stressor on frontal eeg asymmetry, stress, and health. *Biological psychology*, 75(3):239–247, 2007.
- [16] LMAX. Lmax disruptor, 2024. URL <https://lmax-exchange.github.io/disruptor/>. Pristupljeno: 2024-8-05.
- [17] Giulia Masi, Gianluca Amprimo, Claudia Ferraris, i Lorenzo Priano. Stress and workload assessment in aviation—a narrative review. *Sensors*, 23(7):3556, 2023.
- [18] NASA. Flight crew physiological data for crew state monitoring, 2024. URL https://data.nasa.gov/dataset/Flight-Crew-Physiological-Data-for-Crew-State-Moni/8a5d-xhfr/about_data. Pristupljeno: 2024-8-05.

- [19] Labster Theory Pages. Heart rate, 2023. URL https://theory.labster.com/seal_heart_rate/. Pristupljeno: 2024-8-05.
- [20] Lars J Planke, Alessandro Gardi, Roberto Sabatini, Trevor Kistan, i Neta Ezer. Online multimodal inference of mental workload for cognitive human machine systems. *Computers*, 10(6):81, 2021.
- [21] Siniša Popović. *Adaptive stimulation of emotional system based on virtual reality*. Doktorska disertacija, University of Zagreb. Faculty of Electrical Engineering and Computing, 2011.
- [22] Lawrence J Prinzel, Frederick G Freeman, Mark W Scerbo, Peter J Mikulka, i Alan T Pope. A closed-loop system for examining psychophysiological measures for adaptive task allocation. *The International journal of aviation psychology*, 10(4):393–410, 2000.
- [23] Shaozheng Qin, Erno J Hermans, Hein JF Van Marle, Jing Luo, i Guillén Fernández. Acute psychological stress reduces working memory-related activity in the dorsolateral prefrontal cortex. *Biological psychiatry*, 66(1):25–32, 2009.
- [24] Georgina Russell i Stafford Lightman. The human stress response. *Nature reviews endocrinology*, 15(9):525–534, 2019.
- [25] EA Simoes, R Roark, S Berman, L Lm Esler, i J Murphy. Respiratory rate: measurement of variability over time and accuracy at different counting periods. *Archives of disease in childhood*, 66(10):1199–1203, 1991.
- [26] Christiaan H Vinkers, Renske Penning, Juliane Hellhammer, Joris C Verster, John HGM Klaessens, Berend Olivier, i Cor J Kalkman. The effect of stress on core and peripheral body temperature in humans. *Stress*, 16(5):520–530, 2013.
- [27] H Yaribeygi i H Sahraei. Physiological/neurophysiological mechanisms involved in the formation of stress responses. *Neurophysiology*, 50:131–139, 2018.

Simulacija modularnog telemetrijskog sustava za praćenje fizioloških indikatora stresnih mentalnih stanja

Sažetak

Razvoj sustava za praćenje fizioloških indikatora u skladu s načelima paralelizma i istodobnosti (engl. concurrency) i njihova evaluacija tema je raznih istraživanja. Cilj ovog rada bio je izgraditi modularnu simulaciju koja će efikasno obrađivati podatke i pratiti mentalno stanje ispitanika. Sukladno protočnoj (engl. *pipe-and-filters*) arhitekturi simulacija je sastavljena od različitih dijelova koji se mogu izvoditi istodobno: prikupljanje fizioloških signala, računanje fizioloških značajki, klasifikacija izabranih mentalnih stanja, vizualizacija odabranih fizioloških značajki i rezultata klasifikacije, spremanje svih podataka, značajki i rezultata klasifikacije. Prstenasti međuspremnik iz LMAX Disruptor biblioteke u Javi kompatibilan je s predloženom modularnom arhitekturom te osigurava efikasnu razmjenu podataka u programskom ostvarenju simulacije. Izabrane izračunate značajke i mentalna stanja klasificirana primjenom algoritma s ad-hoc pravilima prenose se mrežnim protokolom i vizualiziraju na grafičkom sučelju mrežne aplikacije, simulirajući tako prikaz potencijalno važnih podataka stručnjaku koji se nalazi na udaljenoj lokaciji. Obavljena preliminarna testiranja na javno dostupnim fiziološkim podacima pilota u različitim mentalnim stanjima tijekom vježbi na simulatoru leta pokazuju zadovoljavajući protok podataka kroz ostvarenu arhitekturu na raspoloživom računalnom hardveru, uz postojanje određenih ograničenja i prostora za daljnja unaprjeđenja.

Ključne riječi: Stres, međuspremnik, simulacija, indikatori, značajke, fiziologija, telemetrija, protočna arhitektura.

Simulation of a modular telemetry system for monitoring physiological indicators of stressful mental states

Abstract

The development of physiological indicator monitoring systems in accordance with the principles of parallelism and concurrency and their evaluation are the subject of various research. The aim of this paper was to build a modular simulation that will efficiently process data and monitor the mental state of the subjects. According to the pipe-and-filter architecture, the simulation is composed of different parts that can be performed concurrently: collection of physiological signals, computation of physiological features, classification of selected mental states, visualization of selected physiological features and classification results, storage of all data, features and classification results. The ring buffer from the LMAX Disruptor library in Java is compatible with the proposed modular architecture and ensures efficient data exchange in the software implementation of the simulation. Selected calculated features and mental states classified using algorithm with ad-hoc rules are transmitted through the network protocol and visualized on the web application graphical interface, simulating the presentation of potentially important data to an expert located in a remote location. Preliminary testing on publicly available physiological data of pilots in different mental states during flight simulator exercises shows satisfactory data flow through the implemented architecture on the available computer hardware, with certain limitations and room for further improvements.

Keywords: Stress, buffer, simulation, indicators, features, physiology, telemetry, pipe-and-filter architecture.