

TALLINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

TTÜ IT KOLLEDŽ

Kaarel-Martin Maarand 200815IADB

Veebipõhise keskkonna arendamine driftivõistluse korraldamiseks

[Bakalaureusetöö]

Juhendaja: Toomas Lepikult
PhD

Tallinn 2022

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viitatud.

Autor: Kaarel-Martin Maarand

24.04.2022

Annotatsioon

Lõputöö eesmärk on luua tarkvara, mille abil on võimalik drifti võistlusi efektiivsemalt ja läbipaistvamalt läbi viia. Loodav tarkvara asendab praeguse Microsoft Exceli põhise töökeskkonna. Tarkvara lubab võistlejatel võistlusele registreerida, hindajatel sõite hinnata, pealtvaatajatel reaajas tulemusi jälgida. Kõigepealt selgitab töö autor, mis on drift ja driftimine ning annab ülevaate drifti ürituse hetke korraldusest. Seejärel toob autor välja erinevad arenduseks kasutatavad meetodikad, et neid omavahel võrrelda ning valida välja need mida kasutada antud töö tarkvara arenduseks. Meetodikate valimise järel saab autor kirjeldada arendus protsessi, peale mida selle testimist, kasutust ning tulemuste analüüsi kõige lõpuks.

Lõputöö tulemuseks on ASP.NET Core ja React JS kirjutatud tarkvara mille võõrustamiseks kasutab ta Dockerit ja Azure App Services. Tarkvarale teostab ka autor koormustesti kasutades Apache JMeter tööriista. Tarkvarale saavad ligi kõik kasutades tavalist veebibrauserit. Ürituse läbiviimise funktsionaalsusele saavad ligi vaid kasutajad kes on selleks autoriseeritud.

Töö kontrollimiseks võtab autor eelnevate võistluste kirjed ja protokollid ning neid kasutades kontrollib tarkvara efektiivsust ning analüüsib, kas loodud tarkvara lahendab töö eesmärgid.

Bakalaureuse töö on kirjutatud eesti keeles ning sisaldab teksti 34 leheküljel, 5 peatükki, 20 tabelit.

Abstract

Development of a Web-based Environment for Organizing a Drift Competition

The main goal of this thesis is to create a software, which enables to conduct drift competitions more effectively and more transparently. Developed software will replace current Microsoft Excel based work environment. This software allows competitors to register to drift events, for judges to judge drift runs and spectators to monitor the results in real time. First, the author will explain what is drift and drifting and gives an overview of the current state of drift event organization. Secondly the author will bring out different methods used for development to compare them to each other and choose the ones used for this thesis.

The result of this thesis is a software developed with ASP.NET Core backend and React JS frontend. The software is hosted using Docker and Azure App Services. Load test is conducted with Apache JMeter tool. The software is accessible for everyone through a web browser. Drift event hosting functionality is only accessible to users with proper authentication.

To validate this thesis the author uses results and protocols from previous events. This is used to check the effectiveness of the software and to analyse if the software has achieved all the goals or which ones has failed.

The bachelor's thesis is written in Estonian and contains 34 pages of text, 5 chapters, 20 tables.

Võtmesõnade ja lühendite loetelu

.NET	Microsofti poolt arendatud tarkvara raamistik
ACU	<i>Azure Compute Unit</i> , Azure mõõtühik mis aitab võrrelda protsessori jõudlust erinevate teenuste vahel
Amazon RDS	<i>Relational Database Service</i> , Amazoni Relatsiooniline andmebaasi teenus.
Angular	Komponendi põhine raamistik veebirakenduste loomiseks.
API	<i>Application Programming Interface</i> , Rakendusliides
ASP.NET Core	Populaarne veebiarendus raamistik
Azure App Services	HTTP'1 põhinev teenus veebirakenduste, REST API'de ja mobiilsete taustprogrammide võõrustamiseks
Aurelia2	Eessüsteemi raamistik
Backend	Tagaprogramm
C#	Microsofti poolt arendatud programmeerimis keel
DK	Drifti Komitee
Docker	Tööriist, mis on loodud rakenduste loomise, juurutamise ja käitamise hõlbustamiseks konteinerite abil
Drift	Motospordi ala kus sõitjad kontrollitult juhivad üle-juhitavuses olevat autot
Dünaamiline lõimepuul	<i>Dynamic thread pool</i> , Tarkvara kujundusmuster käskude samaaegseks läbiviimiseks mis ise tegeleb suure koormuse korral uute lõimede loomisega ning sulgeb neid kui lõimed pole kasutuses.
EAL	Eesti Autospordi Liit
EDL	Eesti Drifti Liit
ERD	<i>Entity Relational Diagram</i> , Olemi relatsiooniline diagramm
Frontend	Eessüsteem
GitHub	Veebipõhine versioonihaldus raamistik koos võõrutus teenusega.
HTTP	<i>HyperText Transfer Protocol</i> , Hüperteksti edastusprotokoll
Jakarta EE	Varasemalt tuntud kui Java Platform
Javascript	Skriptikeel mis aitab luua interaktiivseid veebisaita mis on vabalt tüübitud.
JSON	<i>Javascript Object Notation</i> , javascripti objekti tähistus
JSX	Laiend Javascripti keelele mis lubab kasutada süntaksit mis sarnaneb HTML keelele.
JWT	<i>JSON Web Token</i> , javascripti objekti tähistuse veebimärk

LAMP	Tarkvara arendusmeetod koos komponentidega mis võimaldavad hostida ja hallata veebi sisu, Linux, Apache, MySQL, PHP.
Lõimepuul	<i>Thread Pool</i> , Tarkvara kujundusmuster käskude samaaegseks läbiviimiseks.
MariaDB	Avatud lähtekoodiga relatsioonibaasihaldur.
Microsoft Excel	Microsofti poolt arendatud arvutustabeli tarkvara.
MS SQL Server	Microsofti poolt arenduses relatsioonibaasihaldur
MVC	<i>Model View Controller</i> , muster mis eraldab rakenduse 3 ossa - mudel, vaade, kontrollier.
MVP	<i>Minimal Viable Product</i> , Minimaalne elujõuline toode
MySQL	Populaarne avatud lähtekoodiga relatsioonibaasihaldur
NAS	<i>Network Access Server</i> , võrkupääsuserver
OMT	<i>One More Time</i> on sõidu hinne, kus kohtunik ei suutnud otsustada kumb sõitja oli parem.
Patch	Parandus programmis
PostgreSQL	Populaarne avatud lähtekoodiga relatsioonibaasihaldur
RAM	<i>Random Access Memory</i> , muutmälu
RDBMS	<i>Relational database management systems</i> Relatsioonibaasihaldur
React JS	Facebooki poolt arendatud avatud lähtekoodiga Javascripti teek
REST	<i>Representational State Transfer</i> , oleku üleandmis reeglistik mida järgitakse veebirakenduste arendamisel.
TSX	Laiend Typescript keelele mis lubab kasutada süntaksit mis sarnaneb HTML keelele.
Typescript	Javascripti liides mis jõustab tüüpide range kasutuse.
URL	<i>Uniform Resource Locatior</i> , internetiaadress
VPC	<i>Virtual Private Cloud</i>
Vue JS	Progressiivne teek kasutajaliidese loomiseks.

Sisukord

Sissejuhatus.....	10
1 Probleemi püstitus ja ülevaade.....	12
1.1 Ülevaade driftist.....	12
1.2 Ülevaade hetkeseisust drifti ürituse korralduses.....	12
1.3 Ülesande püstitus	13
2 Metoodika	15
2.1 Valikus olevad arendus metoodikad ja tööriistad	15
2.1.1 Tarkvara <i>backend</i> (tagaprogrammi) arenduseks potentsiaalsed meetodid ja nende eripärad	15
2.1.2 Tarkvara <i>frontend</i> (eessüsteemi) arenduseks potentsiaalsed meetodid ja nende eripärad	17
2.1.3 Tarkvara pilve potentsiaalsed valikud ja nende eripärad.....	19
2.1.4 Tarkvara andmebaasi potentsiaalsed valikud ja nende eripärad.....	21
2.2 Tarkvara arenduse metoodikate ja tööriistade valik	25
2.2.1 Tarkvara arenduse taga programmi arenduseks valitud meetod.....	26
2.2.2 Tarkvara arenduse kasutajaliidese arenduseks valitud meetod.....	26
2.2.3 Tarkvara arenduse pilve valitud meetod	26
2.2.4 Tarkvara arenduse andmebaasi valik	26
2.2.5 Tarkvara arenduse funktsionaalsus	27
2.2.6 Tarkvara koormustestimine	27
3 Lahenduskäik ja tulemused.....	28
3.1 Küsitluste ja intervjuude tulemused ja analüüs.....	28
3.1.1 Pealtvaatajate küsitlus.....	28
3.1.2 Korraldusmeeskonna küsitlus	28
3.1.3 Võistlejate küsitlus.....	29
3.2 Veebikeskkonna arendamine	29
3.2.1 Veebikeskkonna funktsionaalsus	29
3.3 Veebikeskkonna testimine	31
3.3.1 Eelnevate ürituste testid	31
3.3.2 API Koormus testid	33
3.4 Veebikeskkonna kasutamine.....	37

3.4.1 Talvedrift 2022 III etapp.....	38
3.4.2 CuFa RC Drifti Liiga 2022 – 1. Etapp.....	39
3.5 Tulemuste analüüs	40
3.5.1 Talve drift III etapp analüüs.....	40
3.5.2 CuFa RC Drifti Liiga 2022 – 1. Etapp analüüs.....	40
3.5.3 Koormustesti analüüs.....	41
4 Edasine arendustöö	42
5 Kokkuvõte.....	43
Kasutatud kirjandus	44
Lisa 1 - Lihtlitsents	46
Lisa 2 – Tarkvara koodifailid.....	47
Lisa 3 – Intervjuud	48
Küsimused Eesti Drifti Liidu sekretärile	48
Küsimused Eesti Drifti Komitee esimehele Kristjan Salmrele.....	48
Küsitlus pealtvaatajatele	49
Küsitlus võistlejatele.....	51
Küsitlus korraldustiimile.....	52
Lisa 4 – ERD skeem	54
Lisa 5 – Olemite semantika.....	55

Tabelite loetelu

Tabel 1. ASP.NET CORE vs Jakarta EE.....	15
Tabel 2. Jakarta EE.	16
Tabel 3. ASP.NET Core.....	16
Tabel 4. Eessüsteemi meetodikad.....	17
Tabel 5. React JS.....	18
Tabel 6. Vue JS.....	18
Tabel 7. Aurelia2	19
Tabel 8. Angular	19
Tabel 9. Amazon RDS	20
Tabel 10. Azure App Services.	20
Tabel 11. NAS.	21
Tabel 12. Andmebaasid.	22
Tabel 13. Andmebaaside toetatud programmeerimis keeled.....	23
Tabel 14. MySQL	24
Tabel 15. MS SQL Server.....	24
Tabel 16. PostgreSQL.....	25
Tabel 17. MariaDB	25
Tabel 18. Koormustestid üksikutele kirjetele.	35
Tabel 19. Koormustestid mitmetele kirjetele.....	36
Tabel 20. Koormustestid kontroll päringutele.	37

Sissejuhatus

Drift on motosportdiala, kus juhid peavad ülejuhitavusse viidud autot kontrollitult juhtima. Driftida saab nii üksinda kui ka paaris, kus järeleõitja peab imiteerima eessõitjat võimalikult täpselt ja lähedalt. Drifti võistlus on üritus, kus saavad võistelda professionaalsed sõitjad kui ka amatöörid tänu erinevatele võistlusklassidele. Võistlus koosneb põhiliselt kvalifikatsioonisõitudest ja kvalifikatsioonisõitude tulemuste põhjal loodud eliminaatortabeli paarissõitudest. Sellele eelneb ka registreerimine ja muud administratiivsed tegevused.

Drifti võistluse läbiviimise veebikeskkonna abil on eeldatavasti võimalik säästa võistluse ooteaegade arvelt kuni 2 tundi. Samuti aitab drifti ürituse läbiviimise veebikeskkond vältida inimlikke vigu, mis tekivad käsitsi andmete sisestamisel. Põhjus, miks aja säästmine väga tähtis korraldajatele on, et kuna tihti on võistlusraja läheduses elamud ning öörahu ajaks peab võistlus lõpetatud olema. Aja säästmine aitab ka ürituse ajakava korrektsele loomisele kaasa, millele pealtvaatajad saavad tugineda. Hetkel on ajakava keskmiselt 1-2 tundi nihkes võistluspäeva kohta seoses erinevate pausidega. Lisaks eelnevale on võimalik pealtvaatajatel ja tiimiliikmetel enda seadmest jälgida reaajas tulemusi ilma kohustuseta käia tulemuste seina juures, kuhu tuleb info viitega paberikujul. Võistlejatel on veebikeskkonnas võimalus ka registreerida üritustele. Viimaseks saab välja tuua potentsiaalse korruptsiooni vältimise hindajate seas. Antud bakalaureusetöö eesmärgiks on valida parimad meetodikad ja arendada tarkvara MVP (*Minimal Viable Product*) mille abil saaks efektiivsemalt läbi viia drifti võistlust, mille tulemusi saaksid reaajas jälgida ka veebikeskkonna kasutajad.

Töö esimeses osa kirjeldatakse käesolevat probleemi ja annab üldise ülevaate, mis on drift ja mis drifti üritus ning hetkel kasutuses olevaid tavasid.

Töö teine osa võrdleb arenduseks valikus olevaid meetodikaid, nende omapärasid ning millised meetodikad autor arenduseks valis.

Töö kolmas osa kirjeldab veebikeskkonna arendust, selle testimist ja kasutust ning analüüsib kasutusest tulenevaid küsimusi ja tulemusi.

Töö neljas osa kirjeldab bakalaureuse töö edasisi plaane ja ambitsioone peale MVP valmimist.

Antud töö raames arendab autor veebikeskkonna tagaprogrammi, eessüsteemi kasutajaliidese ning paigaldab pilvelahenduse.

1 Probleemi püstitus ja ülevaade

Käesolevas peatükis antakse üldine ülevaade driftist, hetke olukorrast drifti ürituste korralduses ning probleemi püstitus.

1.1 Ülevaade driftist

Drift on noor motospordiala mis sai alguse Jaapanist ning on muutunud ülemaailmselt väga populaarseks, eriti noorte seas. Drift on lühidalt üle-juhitava auto kontrollitud juhtimine. Drifti võlu on motospordi kohta madal finantsiline kulu. Drifti üritus koosneb reeglina kolmest osast. Treening, kvalifikatsioon ja paarissõidud. Treening on nagu sõna ise viitab, võimalus teha trenni rajal ja sellega tutvuda. Kvalifikatsioonisõite on 2, milles siis hinnatakse erinevaid aspekte sõidu juures. Paarissõidud on 2 sõitja paarissõidud, mis määratakse kvalifikatsiooni tulemuste põhjal. Iga sõitja teeb ühe ees ja ühe järel sõidu. Eessõitjana pead jälgima kvalifikatsiooni joont ehk kohtunike poolt määratud ideaaltrajektoori parimate sõiduaspektidega. Järelsõitjana pead eessõitjat võimalikult täpselt kopeerima ning seda tegema nii lähedal eesmisele autole kui võimalik.

1.2 Ülevaade hetkeseisust drifti ürituse korralduses

Hetkel toimib Eestis süsteem nii, et kodulehelt ankeedilt saadetakse registreerunute info meilile, kust info ükshaaval käsitsi Exceli tabelisse sisestatakse. Ürituse alguses koostatakse käsitsi erinevad nimekirjad, mis jagatakse paberkujul vajaminevatele korraldustiimi liikmetele laiali. Kui enne registreerimise lõppu on tarvis võistlusjuhil infot, kes on osalejad, siis tehakse eraldi tabel, mis antakse võistlusjuhile. Kui võistlejad on registreerunud, siis vormistatakse lõplik võistlejate nimekiri igale võistlusklassile ning pannakse võistlusnumbri järjekorda. Järgmisena hindajate hindamise tabelisse sisestame võistlejate nimekirja ja prindime selle tabeli välja ning anname hindajatele. Kvalifikatsioonisõitude ajal iga hindaja täidab käsitsi oma hindamise ala käsitsi paberi peal. Kui kvalifikatsioonisõidud on läbi, viiakse täidetud lehed sekretariaati, kus neil lehtedel olevad andmed sisestatakse käsitsi Excelisse. Järgmisena kopeeritakse kvalifikatsiooni hindamise tabelist sõitjate nimekiri eraldi kvalifikatsiooni tulemuste tabelisse, mis automaatselt täidab tulemuste tabeli punktidega. Järgmisena

kopeeritakse kõrvaltabelisse kvalifikatsiooni tulemuste tabelist andmed ja sorteeritakse need paremuse järjekorda ning printitakse sorteeritud kvalifikatsioonitulemuste tabel välja. Sorteeritud kvalifikatsioonitulemuste tabeli järgi koostatakse võistluse eliminaator (rahvakeeli tandem) tabel. Sorteeritud kvalifikatsiooni tulemuste järgi saab kätte juba ka võistluse kvalifikatsioonipunktid, mis lähevad võistluse lõpp-punktisumma arvestusse. Võistluse lõpp-punktisumma summeeritakse samuti käsitsi. Eliminaatortabelit täidetakse jooksvalt käsitsi.

Formula Driftis, Drift Mastersil ja East European Drift Cupil juba kasutatakse olemasolevaid programme. Autor on ise seal olevaid võistlusi jälginud.

Hetkel ei ole Eesti Autospori Liidu Driftikomitee teada Baltikumis kasutuses ühtegi hindamisprogrammi. Läti Drifti liidul on olemas üks hindamisprogramm, kuid kuna see on väga aeglane, siis kasutatakse ikka Microsoft Excel programmi.

Autor on olnud EDL-i (Eesti Drifti Liit) korraldus tiimi liige alates aastast 2012, mistõttu on ta teadlik sellest, kuidas soovitakse vabaneda neist ebavajalikest pausidest. Veebikeskkond nende protsesside läbiviimiseks oleks ka EDL-i esimehe ning teiste korraldustiimi ja komitee liikmete arvates parim lahendus.

Varasemalt on kasutatud drifti sõitudehindamis funktsiooniga programmi, kuid autorile teadaolematutel põhjustel antud koostöö arendajaga katkes ja soov tolle arendajaga edasi töötada puudub, mistõttu antud programmi kahjuks enam kasutada ei saa.

Töö motivatsioon tuleneb sellest, et töö autor on vabatahtlik EDL-i meeskonna liige olnud aastast 2012. Tänu sellele on autor tutvunud enamuse heade ja halbade külgedega drifti spordis ning selle korralduses. Olles driftimise ja üldiselt autospori suur toetaja ja poolehoidja, soovib autor aidata drifti spordile kaasa rohkem kui vaid drifti ürituste boksivanema rollis olles.

1.3 Ülesande püstitus

Eesmärk käesoleval töö on arendada drifti ürituse läbiviimise veebikeskkonna MVP, mis eemaldaks ebavajalikud pausid drifti üritustelt, välistaks inimliku eksimuse eliminaatortabelite koostamisel, annaks võimaluse kõigil jälgida veebis reaajas tulemusi ning võimaldaks

sõitjatel registreerida võistlusele samas veebikeskkonnas. Tulemusi sisestaksid küll hindajad kohtunike tornist programmi käsitsi, kuid kaoks ära vajadus vaheisikule tulemuste sisestamiseks ning sellega ka väheneks tõenäosus teha vigu andmete sisestamisel või edasi andmisel. Kuna iga kohtunik saaks hinnata vaid enda kohustusega kaasnevaid punkte ja mitte teisi punkte, siis see väldib ka tahtlikku hindamisviga enda lemmiku, või vastupidi - mittemeeldiva sõitja sõidule. Tulevikus saab ka seda sama kohtuniku spetsiifilist hindamist kasutada koostöös videotiimiga, kes saab reaalajas videos kuvada *overlay*-na sõitude hindamist.

Taolisi programme on juba arendatud, kuid info põhjal EDL-i esimehelt ja sekretärilt ei ole neist ükski täielikult sobiv EDL-i reeglitega üritusele või ei ole need piisavalt kompetentsed ja kiired. Lätis on arendatud juba üks hindamisprogramm, kuid nad ei kasuta seda, kuna see on liiga aeglane ja keerukas.

Drifti ürituse läbiviimise veebikeskkonna poolt säästetud aeg aitaks kaasa enne öörahu võistluse lõpetamisele, korrektse ajakava koostamisele ja võistlejad saaksid võistelda päevavalguses. Veebikeskkonnas olevate hindamispiirangute abil on võimalik vältida ka korrupsiooni hindajate/kohtunike seas.

2 Metoodika

Käesolevas peatükis tuuakse välja eri metoodikate võrdlused. Selgub, millised metoodikad valitakse ja mis arendustööriistu kasutatakse. Võrreldakse tagaprogrammi, eessüsteemi, pilveteenuste ja andmebaasi võimalusi. Eri metoodikate võrdluste põhjal teostatakse analüüsid, mis metoodikat kasutada.

2.1 Valikus olevad arendus metoodikad ja tööriistad

Erinevate arendus osade jaoks kasutatakse erinevaid metoodikaid ja tööriistu. Järgnevalt toob autor välja milliseid neist võrreldakse.

2.1.1 Tarkvara *backend* (tagaprogrammi) arenduseks potentsiaalsed meetodid ja nende eripärad

Käesolevas peatükis tuuakse välja tabelite kujul erinevused ASP.NET Core ja Jakarta EE vahel.

Tabelis 1 on võrreldud ASP.NET Core ja Jakarta EE omadusi.

Tabel 1. ASP.NET CORE vs Jakarta EE ([1], [2], [3]).

	ASP.NET Core	Jakarta EE
Avatud lähtekood	Jah	Jah
Mitme platvormiline	Windows, Linux, macOS	Iga platvorm ja server
Programmeerimis keel	C#	Java
Kommuuni suurus*	Suur	Suur
Veebiteenused	Sisseehitatud	API-d (<i>Application Program Interface</i>)

Tabelis 2 on välja toodud Jakarta EE plussid ja miinused.

Tabel 2. Jakarta EE ([2], [3]).

Jakarta EE Plussid	Jakarta EE Miinused
Töötab igal platvormil ja serveril	Veebiteenuste standardite sisseehitatud tugi puudub.
Sisaldab mitmeid API-sid millel on funktsioonid nagu Enterprise JavaBeans, Servletid, Java Server Faces ja mitmed veebiteenuste tehnoloogiad mis hõlbustavad veebirakenduste loomist.	Keerukas kasutada kiirete, madalate kuludega ja massturuprojektide jaoks.
Suudab käsitleda keerukaid, suure mahulisi ning tehingu rikkaid rakendusi.	
Suur kommuun, millega kaasneb suur tugi.	
Üle 20 ühilduva teostuse nagu Eclipse GlassFish, Oracle Weblogic Server jt.	

Tabelis 3 on välja toodud ASP.NET Core plussid ja miinused.

Tabel 3. ASP.NET Core ([1]).

ASP.NET Core Plussid	ASP.NET Core Miinused
Avatud lähtekood GitHubis	Arendatud ja toetatud ainult .NET raamistiku poolt.
Mitme platvormiline. Windows, Linux, macOS	
Kerge, suure jõudlusega ja modulaarne HTTP (<i>HyperText Transfer Protocol</i>) päringu konveier.	
MVC (<i>Model View Controller</i>) mustri abil on kerge testida API-sid ja veebirakendusi.	
Võimalik võõrustada Dockeris, Kestrelis, Apaches jt.	
Mudeli sidumine automaatselt kaardistab andmed HTTP päringult tegevus meetodi parameetriteks.	

2.1.2 Tarkvara *frontend* (eessüsteemi) arenduseks potentsiaalsed meetodid ja nende eripärad

Käesolevas peatükis tuuakse välja tabelite kujul React JS, Vue JS, Aurelia 2 ja Angular-i erinevused ja sarnasused ning mis plussid ja miinused neil on.

Tabelis 4 on võrreldud eessüsteemi metoodikate omadusi.

Tabel 4. Eessüsteemi metoodikad ([4], [5], [6], [7], [8], [9], [10], [11], [12], [13]).

	React JS	Vue JS	Aurelia 2	Angular
Avatud lähtekood	Jah	Jah	Jah	Jah
<i>Databinding</i>	<i>One-way</i>	<i>Two-way</i>	<i>Reactive</i>	<i>Two-way</i>
DOM	Virtuaalne	Virtuaalne	Jah	Jah
Suurus	Teek (väike)	Teek (väike)	JS moodulite kollektsioon	Raamistik (suur)
Programmeerimis keel(ed)	Javascript, Typescript, JSX, TSX, HTML, CSS	Javascript, Typescript, HTML, CSS	Javascript, Typescript, HTML, CSS	Javascript, Typescript, HTML, CSS

Tabelis 5 on välja toodud React JS plussid ja miinused.

Tabel 5. React JS ([4], [5], [6]).

React JS Plussid	React JS Miinused
Avatud lähtekoodiga GitHubis	Areneb kiiresti mistõttu tuleb arendajal jooksvalt end kursis hoida uute muutustega.
Palju pistikprogramme	<i>One-way databinding</i>
Suur kommuun millega kaasneb suur tugi.	
Kasutab virtuaalset DOM'i mis muudab React JSi kiiremaks kui tavalist DOM'i kasutavad raamistikud/tegid	
Võimaldab kasutada JSX ja TSX faile.	
Kerge kaasata olemasolevatesse projektidesse kuna tegu on teegi mitte raamistikuga.	
Kerge testida koodi	
Väga paindlik.	
Komponendi põhine.	

Tabelis 6 on välja toodud Vue JS plussid ja miinused.

Tabel 6. Vue JS ([7], [8], [9]).

Vue JS Plussid	Vue JS Miinused
Avatud lähtekoodiga GitHubis	Väike kogukond võrreldes Reacti ja Angulariga mistõttu on ka tugi kesisem
<i>Two-way databinding</i>	Pistikprogrammide kogus on väike võrreldes React JS'ga.
Kasutab virtuaalset DOM'i mis muudab Vue JSi kiiremaks kui tavalist DOM'i kasutavad raamistikud/tegid	
Teek mitte raamistik, ehk suurus on väike.	
Väga paindlik	

Tabelis 7 on välja toodud Aurelia2 plussid ja miinused.

Tabel 7. Aurelia2 ([10], [11]).

Aurelia2 Plussid	Aurelia2 Miinused
Avatud lähtekoodiga GitHubis	Väga väike kommuun ja tugi võrreldes teiste võrdluses olevate teekide ja raamistikega.
Kerge testida koodi	Kerge kaotada järg kirjutades keerukaid veebilehti.
Pärandrakendusi on võimalik järkjärgult täiendada.	
<i>Reactive binding</i> (reaktiivne sidumine) mis seob automaatselt kasutajaliidese ja rakenduse oleku.	
Kiire tänu enda reaktiivsele süsteemile.	

Tabelis 8 on välja toodud Angulari plussid ja miinused.

Tabel 8. Angular ([12], [13]).

Angular Plussid	Angular Miinused
Avatud lähtekoodiga GitHubis	Raske omandada.
Suur kommuun millega kaasneb suur tugi.	Angular on raamistik ehk tunduvat suurem kui teised võrdluses olevad teegid.
<i>Two-way databinding</i> mis teeb koodi kirjutamise lihtsamaks.	
Koodi taaskasutatavus.	

2.1.3 Tarkvara pilve potentsiaalsed valikud ja nende eripärad

Käesolevas peatükis tuuakse tabelite kujul välja Amazon RDS, Azure App Service ja NAS (*Network Access Server*)* plussid ja miinused.

Tabelis 9 on välja toodud Amazon RDS plussid ja miinused.

Tabel 9. Amazon RDS ([14], [15]).

Amazon RDS Plussid	Amazon RDS Miinused
Lihtne kasutada.	Automaatse <i>patchimisega</i> tulevad seisakud.
Automaatsed varukoopiad	Puudub <i>root</i> ligipääs serverile.
Ei pea muretsema füüsilise riistvara pärast.	
Turvaline tänu Amazon VPC-le (<i>Virtual Private Cloud</i>).	

Tabelis 10 on välja toodud Azure App Services plussid ja miinused.

Tabel 10. Azure App Services ([16], [17]).

Azure App Services Plussid	Azure App Services Miinused
Serverita kood ehk maksad vaid selle eest mida kasutate.	Muutub kalliks kui ressursi suurused tõusevad palju.
Microsofti poolt loodud ning aktiivses arenduses.	
Hea ülevaade Azure portaalis enda rakenduse teenusest (app service).	
Töötab .NET, .NET Core, Java, Ruby, Node.js, PHP või Pythoniga.	
Turvaline	
Ei pea muretsema füüsilise riistvara pärast.	
Pidev integreerimine ja tarnimine ilma seisakuteta.	
Toetab Dockerit ja konteinerimist.	

Tabelis 11 on välja toodud NAS* plussid ja miinused.

Tabel 11. NAS ([18]).

NAS* Plussid	NAS* Miinused
Sisesehitatud PHP, phpMyAdmin, MariaDB10 ja Docker paketid.	Aeglane
Võimalik domeenil võõrustada.	Kui riistvara jääb jõudluselt lahjaks või mälu on juurde tarvis siis peab selle jaoks maksma ning server oleks selleks ajaks maas.
Kuna autoril on domeen olemas juba siis oleks see lisa kuludeta lahendus.	

* NAS puhul on tegu Synology DS718+'ga millel on järgnevad näitajad:

Protsessor: Intel Celeron J3455 quad-core 1.5GHz, burst up to 2.3GHz

Mälu: 6GB DDR3L,

Kõvaketas: 2x Seagate ST2000VN004-2E4164 (HDD) Synology Hybrid RAID (SHR) (With data protection for 1-drive fault tolerance)

Operatsiooni süsteem: DSM 7.0.1

2.1.4 Tarkvara andmebaasi potentsiaalsed valikud ja nende eripärad

Käesolevas peatükis tuuakse tabelite kujul välja andmebaaside MariaDB, SQL Server, MySQL ja PostgreSQL erinevused ja sarnasused ning plussid ja miinused.

Tabelis 12 on võrreldud erinevaid andmebaase.

Tabel 12. Andmebaasid ([19]).

	MariaDB	SQL Server	MySQL	PostgreSQL
Põhiline andmebaasi mudel	RDBMS (Relational database management systems Relatsioonbaasihaldur)	RDBMS	RDBMS	RDBMS
Sekundaarne andmebaasi mudel	Document store, Graph DBMS, Spatial DBMS	Document store, Graph DBMS, Spatial DBMS	Document store, Spatial DBMS	Document store, Spatial DBMS
Arendaja	MariaDB Corporation Ab, MariaDB Foundation	Microsoft	Oracle	PostgreSQL Global Development Group
Litsents	Avatud lähtekoodiga, kommerts ettevõttele	Kommerts, piiratud avatud lähtekoodiga	Avatud lähtekoodiga, lisa funktsionaalsusega kommerts võimalus	Avatud lähtekoodiga
Serveri operatsiooni süsteemid	FreeBSD, Linux, Solaris, Windows	Linux, Windows	FreeBSD, Linux, macOS, Solaris, Windows	FreeBSD, HP-UX, Linux, NetBSD, OpenBSD, macOS, Solaris, Unix, Windows
XML tugi	Jah	Jah	Jah	Jah
SQL tugi	Jah	Jah	Jah	Jah
Võõrvõtmed (foreign keys)	Jah	Jah	Jah	Jah
Populaarsus võrreldes üksteisega*	4.	2.	1.	3.

*DB-Engines Ranking järgi seisuga November 2021

Tabelis 13 on võrreldud erinevate andmebaaside toetatud programmeerimis keeli.

Tabel 13. Andmebaaside toetatud programmeerimis keeled ([19]).

	MariaDB	SQL Server	MySQL	PostgreSQL
Toetatud programmeerimis keeled	Ada	C#	Ada	.Net
	C	C++	C	C
	C#	Delphi	C#	C++
	C++	Go	C++	Delphi
	D	Java	D	Java info
	Eiffel	JavaScript (Node.js)	Delphi	JavaScript (Node.js)
	Erlang	PHP	Eiffel	Perl
	Go	Python	Erlang	PHP
	Haskell	R	Haskell	Python
	Java	Ruby	Java	Tcl
	JavaScript (Node.js)	Visual Basic	JavaScript (Node.js)	
	Objective-C		Objective-C	
	OCaml		OCaml	
	Perl		Perl	
	PHP		PHP	
	Python		Python	
	Ruby		Ruby	
	Scheme		Scheme	
	Tcl		Tcl	

Tabelis 14 on välja toodud MySQL plussid ja miinused.

Tabel 14. MySQL ([19], [20], [21], [22]).

MySQL Plussid	MySQL Miinused
Turvaline ja kiire	Oracle omandamisest saati pole MySQL enam täielikult avatud lähtekoodiga ning Oracle ei aktsepteeri kolmanda osapoole arendajate poolt loodud <i>patches</i> (lappe).
Tänu oma lihtsale disainile ja mitmele salvestusmootorile suudab MySQL tagada optimaalse jõudluse ja pideva tööaja	Paljud funktsioonid on vaid <i>Enterprise Edition</i> iga mis on tasuline ja kallis.
Kerge kasutada ja kohandatav	Populaarsus kahaneb ning ka suured ettevõtted on kolunud ümber teistele andmebaasidele.
Disainitud keskendudes veebile, pilvele ja väga suurtele andmetele.	

Tabelis 15 on välja toodud MS SQL Server plussid ja miinused.

Tabel 15. MS SQL Server ([19], [23], [24], [25]).

MS SQL Server Plussid	MS SQL Server Miinused
Lihtne seadistada ja kasutada koos .NET'ga	Ettevõtetele tasuline ja kallis.
Turvaline	
Suur kommuun ja tugi.	
<i>Developer Edition</i> on tasuta.	

Tabelis 16 on välja toodud MS SQL Server plussid ja miinused.

Tabel 16. PostgreSQL ([19], [26], [27], [28]).

PostgreSQL Plussid	PostgreSQL Miinused
Soodne või olenevalt kasutusest tasuta	Kiiruse täiustamine nõuab rohkem tööd seoses PostgreSQL fookusega ühilduvusele.
Avatud lähtekoodiga	Paljudel vabavara rakendustel ei pruugi olla PostgreSQL tuge.
Võimaldab jooksutada dünaamilisi veebirakendusi LAMP pinu valikuna.	Aeglasem kui MySQL
Kõrge tõrgete taluvusega	
Kerge kasutada	
Suur kommuun ja tugi	

Tabelis 17 on välja toodud MariaDB plussid ja miinused.

Tabel 17. MariaDB ([19], [22], [29]).

MariaDB Plussid	MariaDB Miinused
Tagasiühilduv	Väiksem kommuun kui MySQL
Avatud lähtekoodiga	Tasuline tugiteenus (<i>support</i>)
Dünaamiline lõimepuul, tänu millele koos suure lõimepuuliga tagab see funktsiooni märkimisväärselt suurema kiiruse, täiustatud replikatsiooni, kiiremad värskendused ning toetab üle 200 000 ühenduse.	
Turvaline	

2.2 Tarkvara arenduse metoodikate ja tööriistade valik

Drifti ürituse läbiviimise veebikeskkonna MVP koostamiseks kasutab autor ASP.NET Core taustaprogrammide jaoks, React JSi kasutajaliidese jaoks, SQL Server andmebaasi jaoks ning Azure App Service pilveteenuste jaoks. Kasutajaliidese kirjutamisel kasutab autor veel ka HTML, CSS ja TypeScript. Algse vajamineva funktsionaalsuse kohta info sai autor EDL-i komitee liikmetelt, drifti tiimide liikmetelt ja autori enda tähelepanekutest drifti üritustel boksivanema rollis.

2.2.1 Tarkvara arenduse taga programmi arenduseks valitud meetod

Tarkvara taustaprogrammide arendamiseks valis autor ASP.NET Core. Kuna raamistik ASP.NET Core on väga hea arendamiseks ja testimaks veebirakendusi ja selle kiirus ja tugi on paremad kui Jakarta EE'1 siis selle põhjal tegi autor lõpliku valiku. Tagaprogrammi arendamise teostas autor kasutades JetBrains Rider tarkvara.

2.2.2 Tarkvara arenduse kasutajaliidese arenduseks valitud meetod

Kasutajaliidese jaoks oli valikus React JS, Vue JS, Aurelia2 ja Angular, kuna need on turul hetkel üks kõige populaarsemad ja levinumad programmeerimiskeeled kasutajaliidese kirjutamisel tänu millele on neil aktiivne ja tugev tugi. Mitu faktorit mängisid kasutajaliidese programmeerimiskeele valikus rolli. Esiteks React JS rakendab *Virtual Document Object Model*it ja kuigi ta kasutab vaid ühesuunalist andmete sidumist mitte kahesuunalist nagu Angular mis teeb koodi kirjutamise keerukamaks siis tänu sellele on React JS kiirem kui Angular. Teiseks on Angular terve suur raamistik ning React JS on vaid teek mis on tunduvalt väiksem ja võimaldab lihtsamini integreerida React JSi projektidesse. Aurelia 2 kahjuks rääkis selle väike olemasolev kommuun ja tugi. Viimaseks jäid võrdlusesse Vue JS ja React JS mis on enda olemuselt väga sarnased, kuid kuna React JS'il on nii suur tugi siis autori valik langes React JSi kasuks. Eessüsteemi arendamise teostas autor kasutades Visual Studio Code tarkvara.

2.2.3 Tarkvara arenduse pilve valitud meetod

Pilvetehnoloogia valikus Azure App Services, Amazon RDS ja koduse Synology serveri vahel jäi peale Azure App Services. Kodune Synology server ei suuda pakkuda sellist võimsust ja andmete turvalisust nagu Azure App Services ja Amazon RDS ning riistvara uuendamine on kallis ja uuendamise ajal on server maas, mistõttu langes see esimesena võrdlusest. Seoses sellega, et taustprogrammide arenduse valikus on ASP.NET Core, mis on samuti Microsofti poolt loodud ning suudab väga sujuvalt toetada .NET Core ja Dockerit, siis sujuv ja mugav kasutusviis on väga suur pluss Amazon RDS ees.

2.2.4 Tarkvara arenduse andmebaasi valik

Andmebaasi valikus MySQL, MSSQL, PostgreSQL ja MariaDB vahel valis autor MSSQL. Kuigi MSSQL on ettevõtetele tasuline siis arendajatele on ka tasuta versioon olemas. Kuna

MSSQL töötab ideaalselt koos ASP.NET Core backendiga siis valik jäi MSSQL kasuks. PostgreSQL ja MariaDB jäid just selle Microsofti enda tehnoloogiate vahelise harmoonia tõttu valikust välja. MySQL ja MSSQL vahelises võrdluses oli otsustavaks autori varasem kogemus MSSQL-ga.

2.2.5 Tarkvara arenduse funktsionaalsus

Arendamisel järgib autor Eesti Autospordi Liidu Driftikomitee Drifti võistlusmäärust [30] ja DK (Drifti komitee) esimehe ja sekretäri juhendeid. Lisa funktsionaalsuse kohta informatsiooni kogub autor ka küsitluste ja intervjuude põhjal korraldusmeeskonnale, võistlejatele ja pealtvaatajatele.

2.2.6 Tarkvara koormustestimine

Tarkvara koormustestimiseks kasutab autor Apache JMeter [31] tööriista. Koormustestimise eesmärgiks on näha, kui palju päringuid toetab etteantud Azure keskkonnas seadistatud konfiguratsioonis veebikeskkond.

3 Lahenduskäik ja tulemused

Käesolevas peatükis toob autor välja tarkvara arenduse protsessi, selle testimise, selle kasutamise ning vastavalt kasutusele ka analüüsi.

3.1 Küsitluste ja intervjuude tulemused ja analüüs

Käesolevas peatükis tuuakse välja pealtvaatajate, korraldusmeeskonna ja võistlejate ning tiimide küsitluste ja intervjuude tulemused ning analüüsid.

3.1.1 Pealtvaatajate küsitlus

Küsitlusele vastas 32 inimest. Küsitluse tulemused asuvad Lisa 2. Küsitluse tulemuste põhjal saab eeldada, et pealtvaatajad on huvitatud loodavast veebikeskkonnast koos valitud funktsionaalsusega. Hetkel on pealtvaatajad ebarahul reaalajas info edastamisega ja sellest tulenevalt ka pealtvaatajate jaoks arusaamatute pausidega. Avatud tekstiga vastuste eesmärk oli potentsiaalselt pakkuda autorile uusi ideid, kuidas veebikeskkonda veel parendada, kuid autor ei saanud neist uut informatsiooni.

3.1.2 Korraldusmeeskonna küsitlus

Küsitlusele vastas 10 inimest. Küsitluse tulemused asuvad Lisa 2. Küsitluse tulemuste põhjal saab eeldada, et korraldusmeeskonna liikmed on huvitatud loodavast veebikeskkonnast koos valitud funktsionaalsusega. Korraldusmeeskonna liikmete jaoks kõige suuremaks ebamugavuseks hetkel on aeglane info liikumine ja sellest kaasnev segadus. Avatud tekstiga vastuste eesmärk oli potentsiaalselt pakkuda autorile uusi ideid, kuidas veebikeskkonda veel parendada. Uue funktsionaalsuse alla pakuti idee kasutada boksiala 5 minuti taimerit veebikeskkonnas kõigile nähtavalt. See välistaks vajaduse peakohtunikul vajaduse küsida raadiosaatja teel infot järgi jäänud aja ning omaks paremat ülevaadet võistlusest.

3.1.3 Võistlejate küsitlus

Küsitlusele vastas 8 inimest. Küsitluse tulemused asuvad Lisa 2. Küsitluse tulemuste põhjal saab eeldada, et võistlejad on huvitatud loodavast veebikeskkonnast koos valitud funktsionaalsusega. Kõige suuremateks murekohtadeks võistlejatele on ebamugav võistlusele registreerimine, kuna tuleb infot sisestada nii veebis kui ka käsitsi kohapeal. Samuti reaalajas info pole hetkel kättesaadav. Avatud tekstiga vastuste eesmärk oli potentsiaalselt pakkuda autorile uusi ideid, kuidas veebikeskkonda veel parendada, kuid autor ei saanud neist uut informatsiooni.

3.2 Veebikeskkonna arendamine

Algselt loob autor ERD (*Entity Relational Diagram*) skeemi kuidas andmebaas peaks välja nägema. Järgmisena loob ASP.NET Core kasutades relatsioonilise andmebaasi koos äriloogikaga mida võõrustab SQL Serveris. Peale seda loob kasutajaliidese React JS keeles. Kui programm valmis loob Docker *image*, mille laeb DockerHub'i ning juurutab kasutades Azure App Services. Veebikeskkonda arendades kasutab programme JetBrains Rider, Visual Studio Code ja Docker Desktop.

3.2.1 Veebikeskkonna funktsionaalsus

Käesolevas peatükis toob autor välja grupeeritult erinevad funktsionaalsused mida veebikeskkond pakub.

Kasutajate haldus:

- Kasutajaga sisselogimine,
- Kasutaja registreerimine,
- Meili kinnitus,
- Unustatud parooli haldus,
- Rolli haldus,
- Kasutaja parameetrite haldus,
- Võistleja rollis kasutajale sponsorite lisamine (TODO),
- Võistleja rollis kasutajale auto lisamine (TODO),
- Võistleja rollis kasutaja distsiplinaar kirjed,
- Võistleja diskvalifitseerimisel vastava ürituse sarja punktide ja tulemuste eemaldamine (DOUBLE CHECK).

Tiimide haldus:

- Tiimide loomine ja haldus,
- Tiimi liikmete jälgimine (TODO).

Ürituse haldus:

- Ürituse loomine,
- Automaatne võistluste loomine vastavalt valitud võistlus klassidele üritusel.

Võistluse haldus:

- Võimalus valida kvalifikatsiooni sõitude arvu mis on vaikimisi väärtusega 2,
- Võimalus võistelda ainult paarissõitudega jättes vahele kvalifikatsiooni sõidud,
- Võimalus välja lülitada *byerun*-id ehk esimese paarissõitude vooru sõidud kus puudub vastane,
- Jälgida üritusel osavõtjate nimekirja (TODO),
- Võistlusklasside ja võistlussarjade haldus.

Registreerimine:

- Kasutaja üritusele registreerimine,
- Kasutaja sarja/hooaja registreerimine,
- Võistlus numbrite kontroll sarja ja võistlusele registreerimisel vastavalt võistlusnumbrile ja kasutaja rahvusele,
- Registreerimiste põhjal võistluse kvalifikatsiooni tabeli koostamine.

Hindamine:

- Kvalifikatsiooni tabeli automaatne loomine vastavalt kinnitatud üritusele registreerunutele.
- Kvalifikatsiooni sõitjate hindamine hindaja rolli põhise kontrolliga.
- Kvalifikatsiooni tulemuste automaatne sorteerimine paremuse järjestikus vastavalt EDL reeglitele.
- Eliminaatortabeli automaatne loomine vastavalt kvalifikatsiooni tulemustele.
- Paarissõitude hindamine hindaja rolli põhise kontrolliga.
- Paarissõitude põhjal võistluse paremus järjestikuse määramine.
- Kvalifikatsiooni lõpetamisel tulemus kohtade jagamine.
- Paarissõitude lõpetamisel tulemus kohtade jagamine.
- Kvalifikatsiooni lõpetamisel sarja punktide jagamine.
- Paarissõitude lõpetamisel sarja punktide jagamine.

Tulemuste jälgimine:

- Võimalus jälgida kvalifikatsiooni tulemusi nii detailselt kui üldiselt reaalajas.
- Võimalus jälgida paarissõitude tulemusi reaalajas.

Videoülekanne:

- Reaalajas JSON (*Javascript Object Notation*) kirjade esitamine video tiimile, kes kasutavad infot video ülekande parendamiseks (TODO).

3.3 Veebikeskkonna testimine

Valmis olevat veebikeskkonda testib autor kasutades eelmiste ürituste kirjeid. Autor teostab ka koormustesti veebikeskkonnale kasutades Apache JMeter [31] koormustestimise tööriista.

3.3.1 Eelnevate ürituste testid

Käesolevas peatükis tuuakse välja eelnevate ürituste testid. Kõikide testide eelduseks on esialgselt tagaprogrammi lokaalses masinas käivitamine. Esialgselt tuleb luua uus migratsioon käsuga „dotnet ef migrations --project DAL.App.EF --startup-project WebApp add InitialCreate“. Peale seda tuleb seada „appsettings.json“ failis „SeedData“ väärtus tõeseks ning

käivitada tagaprogramm. Testid viib läbi autor iseseisvalt, kasutades erinevates rollides olevaid kasutajaid.

29.01.2022 Talvedrift I etapp:

Talvedrift 2022 I etapil osalesid 34 võistlejat ja võistlus toimus ühe võistlusklassi võistlussarjaga.

Autor lõi alustuseks uue ürituse asukoha, kuna varasemaid kirjeid võistluse asukohaga pole. Järgnevalt lõi autor ürituse kus märkis ära vaid 1 võistlusklassi mis omakorda automaatselt lõi uue võistluskirje. Autor lõi iga võistleja kohta kasutaja ilma võistleja rollita ning siis ükshaaval iga kasutaja alt nõudis võistleja rolli kasutaja seadete alt, mis tuli kinnitada tagaprogrammis seatud meili aadressilt klikkides lingile. Järgmisena lõi autor võistlussarja, kuhu saaksid võistlejad registreerida ning mille põhjal jagatakse sarja punkte. Enne võistluse alustamist tuli registreerida võistlejad sarja, et neile punkte hiljem jagada ning võistlusele, et neid hinnata. Kui kõik võistlejad olid registreeritud hakkas autor iga võistleja kvalifikatsiooni punkte vastava hindaja kontolt sisestama kuni kõik 204 kirjet olid sisestatud. Kvalifikatsiooni lõpetamiseks vajutas autor sekretäri konto alt „Finish Qualification“ nuppu. Kvalifikatsiooni lõpetamisega kaasnes automaatne kvalifikatsiooni kohtade jagamine ning avanes võimalus kohtunikel paarissõite hinnata. Paarissõitude hindamiseks on 2 vaadet. Kui ekraani laius on piisav siis kuvab terve eliminaatortabeli, kui mitte siis kuvab nimekirja võistluspaaridest, kes omavahel võistlevad ning keda hinnata. Kuna autor sisestas andmeid arvutist 27“ QHD monitorilt siis „Top32“ eliminaatortabel ei mahtunud ekraanile ära. Järgmisena sisestas autor paarissõitude tulemused iga kohtuniku alt. Kuna pole saadaval varasemaid individuaalseid otsuseid kohtunike poolt, siis autor proovis erinevaid kombinatsioone, sisestades tulemusi silmas pidades vaid niipalju, et lõplik tulemus peab tulema nagu reaalsel võistlusel oli. Peale kõikide tulemuste sisestamist oli eliminaatortabel nii täisvaates kui ka telefoni vaates korrektne ning sekretäri rollis vajutas autor nuppu „Finish Tandems“, mis seadis paika lõplikult sarja punktid nii kvalifikatsiooni tulemuste kui ka paarissõitude tulemuste põhjal.

Võrreldes veebikeskkonna kirjeid ürituse kirjetega tuli välja, et andmebaasis olid kõik andmed korrektsed, kuid sarja punkte kuvades oli algselt järjekord vale samade punktide saanud võistlejatel. Sarja punktide järjekorras peab asuma eespool võistleja, kes sai parema koha paarissõitudest peale mida järjestatakse kvalifikatsiooni tulemuse põhjal. Autor tegi paranduse kus lisas ka kvalifikatsiooni tulemuste põhjal sorteerimise ning võistluse tulemused olid korrektsed.

02-03.07.2021 „Jump for Drift“ – Laitse:

„Jump for Drift“ üritusel toimusid võistlused Street, Semi-Pro ja Pro klassis. Street klassist võttis osa 30 võistlejat, Semi-Pro klassist 10 võistlejat ning Pro klassist 15 võistlejat. Street klassi ja Semi-Pro klassi võistlused toimusid vaid 1 võistlussarjas, kuid Pro võistlus toimus 3 võistlussarjas. Võistlejad olid pärit 3st erinevast riigist.

Alustuseks lõi autor asukoha, 5 võistlussarja ja ürituse, kus valis kõik 3 võistlusklassi ja 5 võistlussarja. Järgmisena lõi autor kõik 55 kasutajat ning kuna varasemast testist autor oli teadlik, et rollide määramine töötab, andis kohe kasutajatele võistleja rollid. Peale kasutajate loomist registreeris võistlejad vastavatesse sarjadesse, kus võeti osa. 30 street-klassi võistlejat Eesti Drifti Karikavõistlused 2021 Street-klassis sarja, 15 semi-pro võistlejat Eesti Drifti Karikavõistlused 2021 Semi-Pro klassis sarja, 7 pro klassi võistlejat Eesti Drifti Karikavõistlused 2021 Pro klassi sarja, 14 pro klassi võistlejat Livonia Drifti Meistrivõistlused 2021 Pro klassis sarja ja 12 pro klassi võistlejat Baltic Drift Meistrivõistlused 2021 Pro klassis sarja. Peale sarjadesse registreerimist registreeris autor võistlejad vastavatele võistlustele. Kvalifikatsiooni tulemuste sisestamisel sisestas autor punktid vastava rolliga konto alt, kokku 330 kirjet. Kvalifikatsiooni lõpetamiseks vajutas autor sekretäri konto alt „Finish Qualifying“ nuppu, mis seadis paika kvalifikatsiooni tulemused ning avas võimaluse kohtunikel paarissõite hinnata. Kuna individuaalseid hindeid pole võimalik järgi vaadata sisestas autor iga kohtuniku alt hindeid keskenduses, et lõpp-tulemus oleks korrektne erinevate kombinatsioonidega. Peale iga võistlusklassi kvalifikatsioonide ja paarissõitude hindamise lõppu vajutas autor sekretäri rolli alt „Finish Tandems“ nuppu mis genereeris sarja punktid.

Võrreldes kõiki veebikeskkonnas olevaid kirjeid ürituse kirjetega on kõik andmed ühtivad.

3.3.2 API Koormus testid

Tarkvara koormustestimisel, testib autor Apache JMeter abil päringuid kodulehe pihta ning *API controllerite* pihta.

Koormustestimiseks andmed sisestab autor kasutades valmis kirjutatud kirjete genereerimist. Test andmete sisestamiseks muudab autor lokaalses masinas konfiguratsiooni faili parameetreid, et varasemad kirjed kustutataks ja sisestataks test-kirjed. Lokaalsest masinast ligipääsuks lisab autor Azure SQL serveri tulemüüri seadetesse enda IP aadressi ning jooksub tagaprogrammi lokaalses masinas.

Lisatud kirjetesse kuulus kokku 2512 kirjet mis on autori hinnangul ligikaudselt sarnane 2021

Eesti võistlustel tekkinud kirjete arvuga:

- 12 „AppRole“ kirjet
- 22 „AppUser“ kirjet
- 9 „Team“ kirjet
- 15 „Country“ kirjet
- 9 „Location“ kirjet
- 9 „DriftEvent“ kirjet
- 3 „CompetitionClass“ kirjet
- 27 „Competition“ kirjet
- 30 „CompetitionSeries“ kirjet
- 594 „EventRegistration“ kirjet
- 594 „QualificationResult“ kirjet
- 594 „SeriesRegistration“ kirjet
- 594 „TamdemDecision“ kirjet

Koormustestimisel kasutatava arvuti näitajad:

- Arvuti: HP Elitebook 840 G5
- Protsessor: Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz, 1800 Mhz, 4 Tuuma, 8 Loogilist Protsessorit
- RAM (*Random Access Memory*, muutmälu): 32GB
- Operatsiooni süsteem: Windows 11 Pro

Koormustestimise eesmärk on näha, kui palju päringuid toetab veebikeskkond ühe aasta jooksul tekkinud andmete korral. Kuna tavakasutajal pole õigusi teha kirjeid mujale kui enda kasutaja parameetrite muutmiseks, siis keskendub koormustestimine GET meetoditele.

Koormustestid tehakse tagaprogrammi API lõpp-punktide vastu.

Baas url (*Uniform Resource Locatior*): <https://localhost:5001>

Tabelite esimeses reas on harude (*thread*) kogus ja esimeses veerus on API lõpp-punktid.

Tabelite väärtused on põrunud päringute %/läbilaskevõime (päringute kogus sekundis)

Tabelis 18 on välja toodud üksik kirjete päringute tulemused. Päringu *Authorization* päises kasutati „RegularUser“ rolliga kasutaja JWT (*JSON Web Token*) väärtust mis valideerib päringud.

Tabel 18. Koormustestid üksikutele kirjetele.

	1	10	100	250	500
/api/v1/DriftEvents/{id}	0/35,7	0/10,5	0/34,6	0/99,2	0/38,3
/api/v1/DriftEvents/ WithCollections/{id}	0/142,9	0/10,4	0/32,2	0/66,4	0/28,8
/api/v1/Competitions/{id}	0/76,9	0/10,3	0/12,1	4,8/7,4	0,6/12
/api/v1/TandemDecisions/{id}	0/27,8	0/10,5	0/11,9	30,8/6,4	33,6/12
/api/v1/QualificationResults/{id}	0/90,9	0/10,9	0/12,1	1,6/6,2	8,2/11,2
/api/v1/EventRegistrations/{id}	0/58,8	0/11,6	0/12,2	3,6/5,8	8,4/10,8
/api/v1/EventRegistrations/ WithCollections/{id}	0/8,7	0/10,4	0/11,2	6,8/5,7	11/10,1
/api/v1/SeriesRegistrations/{id}	0/111,1	0/18,1	0/12,2	0,8/8,0	13,2/10,6

Tabelis 19 on välja toodud hulgi kirjete päringute tulemused. Päringu *Authorization* päises kasutati „RegularUser“ rolliga kasutaja JWT väärtust mis valideerib päringud.

Tabel 19. Koormustestid mitmetele kirjetele.

	1	10	100	250	500*
/api/v1/EventRegistrations	0/8,3	0/7,9	0/16,1	0/5,0	-
/api/v1/EventRegistrations/ Competition/{competitionId}	0/10,3	0/6,5	0/6,7	6/4,0	-
/api/v1/QualificationResults	0/7,8	0/6,0	0/4,1	4/3,2	-
/api/v1/QualificationResults/ Competition/{competitionId}	0/23,3	0/5,2	0/3,2	4,8/2,7	-
/api/v1/TandemDecisions	0/2,2	0/3,6	0/2,3	0,4/2,1	-
/api/v1/TandemDecisions/ Competition/{competitionId}	0/4,4	0/4,1	0/2,5	1,21/1,9	-
/api/v1/Competitions	0/22,7	0/7,5	0/3,2	0,4/1,9	-
/api/v1/DriftEvents	0/30,3	0/7,6	0/3,3	0/2,0	-
/api/v1/SeriesRegistrations	0/10,8	0/7,4	0/3,4	0/2,0	-
/api/v1/SeriesRegistrations/ CompetitionSeries/ {competitionSeriesId}	0/22,7	0/8,4	0/3,6	0,81/2,0	-

* 500 haruga päring jooksutas Apache JMeter tööriista korduvalt kokku „OutOfMemoryError“ veaga ,mistõttu sealseid väärtuseid ei saa kasutada.

Tabelis 20 on välja toodud kontroll päringute tulemused. Päringu *Authorization* päises kasutati „Competitor“ rolliga kasutaja JWT väärtust mis valideerib päringud.

Tabel 20. Koormustestid kontroll päringutele.

	1	10	100	250	500
/api/v1/EventRegistrations/ CheckForExistingNumber/ {competitionNumber}/ {competitionId}/ {appUserId}	0/1,7	0/9,2	0/19,2	0/20,1	0/16,9
/api/v1/SeriesRegistrations/ CheckForExistingNumber/ {competitionNumber}/ {competitionSeriesId}/ {competitionClassId}/ {appUserId}	0/23,3	0/10,5	0/29,6	0/20,9	0/15,3

Kõikide kirjete päringute ajal mis on kõige nõudvamad päringud kasutas tagaprogramm arvuti protsessorist 50-70% võimsusest kuni protsessori jõudlus oli 100%. Samade päringute ajal kasutas tagaprogramm 5-6GB mälu. Apache JMeter samade päringute ajal kasutas ühtlaselt 4-6% protsessorist ja 750MB-2GB mälu.

3.4 Veebikeskkonna kasutamine

Peale edukat testimist annab autor veebikeskkonna kasutusse reaajas üritusele ning jälgib koos drifti ürituse korraldustiimiga probleemide lahenduvust.

Veebikeskkonda võõrutasid Azure teenused.

- Eessüsteem: Azure App Services Basic B1 seades, millel on 1.75gb mälu ja 100ACU [31] (*Azure Compute Unit*). Eesprogrammi üles laadimiseks kasutati VSC (Visual Studio Code) Azure liidest.
- Tagaprogramm: Azure App Services Basic B1 seades millel on 1.75gb mälu ja 100ACU [31]. Tagaprogramm laeti üles läbi docker hubi.
- Andmebaas ja server: Azure SQL Database General Purpose Serverless maksimaalse 2vCore'ga ning 6gb mäluga mis toetas kuni 100gb andmemahtu.

Kuna Azure keskkonnas omavahel suhtlevad rakendused ei nõua tulemüüri seadete konfigureerimist, siis oli väga mugav ja sujuv kasutada kõiki teenuseid omavahel.

3.4.1 Talvedrift 2022 III etapp

Talvedrift 2022 III etapp toimus 12.03.2022 Võhu jäärajal. Võistluse formaat oli sama mis tavalisel võistlusel erandiga, et võisteldakse vaid ühes üldises võistlus klassis. Veebikeskkonda kasutas autor võistluskorralduse tiimi kõrval samal ajal kui nad kasutasid enda vana süsteemi. Autor võttis stopperiga aega tegevustel mis tekitasid pause ning mida veebikeskkond automatiseeris.

Võistluspäev algas sõitjate võistlusele registreerimisega. Kuna võistlejatel polnud ligipääsu veel veebikeskkonnale tegi autor seda käsitsi samaaegselt. Ajakulu korraldajale sellest ei lisandunud, kuna registreerimine toimus enne võistluse algust. Võistlusele registreerus kokku 24 võistlejat.

Kell 12.00 algas kvalifikatsioon, mille ajal oli autor hindajate juures ning sisestas tulemusi veebikeskkonnas samaaegselt kui hindajad kirjutasid tulemusi paberile. Kvalifikatsiooni sõitude lõppemisel viidi hindamistulemustega paberid sekretariaati ning võistluse sekretär hakkas käsitsi neid tulemusi Microsoft Exceli tabelisse sisestama, peale mida sorteeris tulemused ning lisas võistlejad vastavalt eliminaatortabelisse. Võistluse sekretäril kulus tulemuste sisestamiseks 16 minutit, kvalifikatsiooni punktide põhjal tulemuste järjestamiseks 6 minutit, eliminaatortabeli koostamiseks 19 minutit ning eliminaatortabelite printimise ja võistluskorraldus tiimi liikmete vahel jaotamiseks 6 minutit.

Kell 14.30 algas paarissõitude top32. Jätakuvalt oli autor koos hindajatega ning sisestas tulemusi veebikeskkonda reaajas. Samaaegselt edastati ka paarissõidutulemusi raadiosaatja teel sekretariaati. Paarissõitude lõppemisel viidi järjekordselt tulemustega paberid sekretariaati topelt kontrolliks, peale mida võistluse sekretär arvutas ja seadis paika kvalifikatsiooni punktid mille jaoks kulus 23 minutit ning paarissõitude punktid, mille jaoks kulus 20 minutit.

Peale autasustamist võrdles autor veebikeskkonnas nähtavaid tulemusi ja sarja punkte Microsoft Exceli tabelisse sisestatud andmetega ning kõik tulemused kattusid identselt. DK esimees pöördus autori poole peale ürituse lõppu ning avaldas soovi nii pea kui võimalik võtta veebikeskkond kasutusse mainides, et see aitaks säästa aega nii pauside arvelt kui ka pakuks head ülevaadet ajakava loomisel.

3.4.2 CuFa RC Drifti Liiga 2022 – 1. Etapp

CuFa RC Drift Liiga 2022 1.Etapp toimus 02.04.2022 – 03.04.2022. Suheldes EDL liikmetega tuli soov testida veebikeskkonda ka puldiga juhitava auto drifti võistlusel. Reeglid antud üritusel on samad mis tavalisel drifti võistlusel.

Veebikeskkonna testimine nägi välja kujul, kus kasutati veel nende vana süsteemi, kuid autor testis veebikeskkonda samaaegselt kohtunike kõrval.

Esimene võistluspäev algas koosolekuga, kuna puldiauto drifti võistlusel oli Google Formsi põhine süsteem olemas siis registreerimine oli vaid selle vahendusel, mitte kohapeal.

Võistlusele oli registreerunud 16 võistlejat kuid kohale ilmusid vaid 14.

02.04.2022 kell 19.00 algasid kvalifikatsiooni sõidud mida oli 3 sõitu osaleja kohta tavapärase 2 sõidu asemel. Kuna see oli algselt märkamata jäänud autoril, siis oli tal võimalik reaalajas kaasa hinnata vaid 2 sõitu võistleja kohta ning ülejäänud tulemustest pidi ta pilti tegema hilisemaks andmete sisestuseks. Google Formsi põhisel süsteemil esines puudusi mida hindajad autorile ka välja töid. Nimelt puudus võimalus lisada kommentaare mis oli autori poolt arendatud veebikeskkonnas võimaldatud ning stiili kohtunikul oli võimalik Google Formsi põhises süsteemis lisada üksik üldhinne, mis tähendas, et stiilikohtunik pidi ise arvutama kokku punktid enne kui ta sai need sisestada. Ühel korral lõppes see puudus ka intsidendiga, kus võistleja märkas enda tulemust 72 punkti ning hiljem kui oli hindaja teinud punktidesse paranduse tuli võistleja protestiga, miks tal järsku nüüd 70 punkti on koos fototõestusega varasemast tulemusest 72. Olemasoleva hindamissüsteemi miinuseks oli veel, et sisestatud andmed sünkroniseeris süsteem alles 15 - 30 sekundit peale sisestamist, et neid ekraanil kuvada. Kõikide eelmainitud puuduste tõttu kirjutasid kohtunikud lisaks Google Formsi süsteemi tulemused ja kommentaarid ka paberile käsitsi.

Peale esimest võistluspäeva arendas autor juurde vabama võimaluse valida kvalifikatsiooni sõitude arvu, mis on vaikimisi 2 ning sisestas kolmanda kvalifikatsiooni sõidu tulemused igale võistlejale vastavalt. Kvalifikatsiooni tulemuste sorteerimine ja arvutamine toimus ilma probleemideta ja koheselt ilma ebavajalike pausideta. Võrreldes veebikeskkonna tulemusi Google Formsi ja käsitsi kirjutatud tulemustega, siis kõik kvalifikatsiooni tulemused ja järjestus olid samad. Algselt arusaamatu probleem tekkis aga eliminaatortabeli loomisel. Nimelt kasutati ühepoolset eliminaatortabelit mitte kahepoolset ja kuigi sõitja paarid olid samad oli nende järjekord teine.

Teisel võistluspäeva alguses pöördus autor võistluse korraldaja poole antud avastusega, kuid mõlemad arvasid algselt, et tegu on ühepoolse eliminaatortabeli omapäraga ja jätkasid

võistlusega. Autor sisestas paarissõitude tulemusi veebikeskkonda kohtunike kõrval ning kõik töötas valatult peale sõitude järjekorra. Google Formsi põhisel süsteemil jätkus 15-sekundiline sünkroonimise ajakulu peale igat hinnatud sõitu. Top4 vooru saabudes, aga tuli ilmsiks ka viga, mis põhjustas sõitude vale järjekorra. Nimelt võistluse läbiviija Google Formsi süsteem ei ehitanud eliminaatortabelit vastavalt drifti reeglitele, mida ka võistluse läbiviija tunnistas ise, et tegu on veaga süsteemis. Nimelt olid Google Formsi hindamissüsteemis märgitud eliminaatortabelid sedasi, et top4 vaates läheksid omavahel kokku kvalifikatsiooni 1 ja 3 koht ning 2 ja 4 koht kuigi reaalsuses DK määrustiku järgi peaksid minema kokku kvalifikatsiooni 1 ja 4 koht ning 2 ja 3 koht. Kuna otsustati jätkata üritus lõpuni avastatud veaga siis autori tarkvara testimine sai antud päevaks lõpu.

Suheldes ka võistlejatega nii Eestist kui Leedust oldi ühemeelselt selle poolt, et oleks veebipõhine rakendus kus kõik näeksid reaalajas tulemusi. Samuti soovitakse ühtset süsteemi kõikide võistluste jaoks, et ei peaks otsima tulemusi mööda interneti ringi.

3.5 Tulemuste analüüs

Käesolevas peatükis toob autor välja veebikeskkonna kasutuse põhjal tehtud analüüsid.

3.5.1 Talve drift III etapp analüüs

12.03.2022 toimunud Talvedrift 2022 III etapil võttis osa 24 võistlejat. Võistluse ajal kasutati mõlemat süsteemi - nii vana Microsoft Exceli põhise süsteemi kui ka uut veebikeskkonda. Tulemuste sisestamine ja vastavate tulemuste genereerimine töötas vastavalt reeglitele. Tulemuste kontrollimisel olid veebikeskkonnas tulemused identselt kattuvad Microsoft Excelis olevate andmetega. Kokku oleks veebikeskkond säästnud umbes 90 minutit võistlusklassi kohta, kus oli 24 võistlejat. See teeb keskmiseks ajavõiduks 3,75 minutit võistleja kohta. Kuna põhiline aja kulu tuleneb andmete sisestamisest ja tulemuste võrdlemisest siis ka ajavõit veebikeskkonnas on vastav võistlejate arvule.

3.5.2 CuFa RC Drifti Liiga 2022 – 1. Etapp analüüs

02.03.2022 – 03.03.2022 toimunud CuFa RC Drifti Liiga 2022 1 Etapilt võttis osa 14 võistlejat. Võistluse ajal kasutati mõlemat nii varasemat Google Formsi põhilist süsteemi kui ka uut veebikeskkonda. Võistluse eripäraks oli 3 hinnatavat kvalifikatsiooni sõitu. Kokku toimus 42

hinnatavat kvalifikatsiooni sõitu. Paarissõitusesse pääses edasi 13 sõitjat. Kuna võistlejad kellel polnud vastast pidid ka teostama hinnatava üksik sõidu ning tehti ka 1 OMT (*One More Time*) otsus siis toimus kokku 34 paarissõitu ehk igal kohtunikul tuli esitada 17 otsust. Võttes arvesse sünkroonimisest tulenenud 15-30 sekundit pausi oleks aitanud veebikeskkond säästa ajaliselt 14.75 – 29.5minutit. Samuti puuduks veebikeskkonnaga vajadus kirjutada tulemusi eraldi paberile mis säästaks omakorda aega ning stiili kohtunikul väheneks arvutus vigade tegemise võimalus.

3.5.3 Koormustesti analüüs

Koormustesti tulemustest on näha, mida rohkem kirjeid on andmebaasi tabelis, seda väiksem on läbilaskevõime suurte haru kogustega.

Aasta jooksul kogunenud kirjete arvuga toimib veebikeskkond vastavalt soovidele korraldus meeskonna kasutuse jaoks ja kuni maksimaalselt 250 tavakasutaja jaoks ning ka siis laadimis ajad kasutajatele venisid ebamugavalt pikaks ning pelgaks kasutajad eemale veebikeskkonnast koos osaliste tõrgetega. Kuna ajapikku kirjete kogused andmebaasis suurenevad ja kasutajabaas kasvab, siis antud päringute põhine süsteem ei ole mõistlik.

Autor täheldas ka, et hindamisega ja registreerimisega seotud kirjeid tekib andmebaasi märkimisväärselt rohkem kui teisi kirjeid ja kõikide nende andmete pärimine ja siis nende filtreerimine pole mõistlik lahendus. Päringute andmehulga vähendamiseks muudab autor päringud kaasamaks filtri väärtusi, et edastataks vaid andmed mida kasutaja soovib mis kiirendab märkimisväärselt ka laadimisaegasid.

Suurte haru kogustega läbilaskevõime probleemi lahenduseks võtab autor kasutusele ASP.Net Core SignalR [32], mille abil ei pea tegema kasutaja API päringuid tagaprogrammile igal navigeerimisel ja/või lehe taas laadimisel vaid saadetakse andmebaasi muudatuste puhul uuendatud andmed määratud kasutajatele.

4 Edasine arendustöö

Edasi on plaanis vastavalt tagasisidele veebikeskkonda edasi arendada terviklikuks ning parandada kasutaja liidest ja kogemust. Suur osa veebikeskkonna toimima saamisest ka pealtvaatajatele on ASP.NET SignalR rakendamine. Kui veebikeskkond on korrektselt toimiv siis arendab autor nutiseadme rakenduse varem arendatud veebikeskkonnast, millel oleks tasuline ja tasuta versioon. Plaanis on lisada duplikaat andmebaas, et eristada korraldajate andmebaas teistest kasutajatest eesmärgiga vähendada riski võistluse korralduse probleemidele näiteks ülekoormuse puhul. EAL-i (Eesti Autospori Liit) litsentsi kontrollimis süsteemiga liita võistleja rolli päringud.

Koostöös videotiimiga soovib autor pakkuda veebikeskkonnas reaajas videoülekanne võistlusest. Peale Eestis rakenduse tööle saamist pakuks autor koostöö võimalust ka naaberriikide drifti liitudele. Järgmine plaan on testida tarkvara Lätis 14.05.2022 „Livonia Drift Championship 2022 Round 1“ võistlusel koostöös videotiimiga. DK esimehe nõusolekuga on plaanis 27.05.2022-28.05.2022 toimuva Eesti Meistrivõistluse etapi ajakava loomisel kasutada arendatud veebikeskkonna abi.

5 Kokkuvõte

Selle töö põhieesmärk on luua tarkvara mis võimaldaks drifti üritusi efektiivsemalt läbi viia.

Töös sai autori poolt välja valitud erinevaid meetodikaid võrreldes tagaprogrammi, eessüsteemi, pilve ja andmebaasi võimalusi. Valitud sai ASP.NET Core tagaprogramm, React JS eessüsteem, Azure App Services pilve teenused ning MSSQL Server andmebaasiks.

Eeldus, et võistluse ooteaegade arvelt on võimalik säästa kuni 2 tundi osutus osaliselt korrektseks. Võistluse ooteaegade arvelt säästetud aeg on suhteline võistlusest osavõtjate arvuga. 24 võistlejaga üksikust võistlusklassist säästis veebikeskkond kokku 90 minutit ehk 1,5 tundi.

Veebikeskkond aitab vältida inimlikke vigu, mis tuli ilmsiks CuFa RC Drifti Liiga 2022 1. etapil, kus stiili punktide kohtunik tegi käsitsi punktide kokkuarvutamisel vea, mida veebikeskkonnas poleks tekkinud, kuna autori arendatud veebikeskkonnas on võimalik eraldi sisestada kohtuniku enda poolt hinded ning kohtunik ei pea ise neid kokku arvutama.

Võistlejatel on veebikeskkonnaga võimalus ka registreerida üritustele. Viimaseks saab välja tuua potentsiaalse korrupsiooni vältimise hindajate seas, kuna hindajad ei pea otseselt üksteisega suhtlema ehk puudub võimalus ka teisi mõjutada.

Kasutatud kirjandus

- [1] D. Roth, R. Anderson ja S. Luttin, „Introduction to ASP.NET Core,“ Microsoft, 22 September 2021. [Võrgumaterjal]. Saadaval: <https://docs.microsoft.com/en-gb/aspnet/core/introduction-to-aspnet-core?view=aspnetcore-6.0>. [Kasutatud November 2021].
- [2] Javatpoint, „Java EE,“ [Võrgumaterjal]. Saadaval: <https://www.javatpoint.com/java-ee>. [Kasutatud november 2021].
- [3] „Java EE Compatibility,“ Oracle, [Võrgumaterjal]. Saadaval: <https://www.oracle.com/java/technologies/compatibility-jsp.html>. [Kasutatud november 2021].
- [4] „Design Principles,“ React, [Võrgumaterjal]. Saadaval: <https://reactjs.org/docs/design-principles.html>. [Kasutatud November 2021].
- [5] „React,“ React, [Võrgumaterjal]. Saadaval: <https://reactjs.org/>. [Kasutatud november 2021].
- [6] „facebook/react,“ GitHub, [Võrgumaterjal]. Saadaval: <https://github.com/facebook/react>. [Kasutatud november 2021].
- [7] „Introduction,“ Vue, [Võrgumaterjal]. Saadaval: <https://v3.vuejs.org/guide/introduction.html#what-is-vue-js>. [Kasutatud november 2021].
- [8] „vuejs/vue,“ GitHub, [Võrgumaterjal]. Saadaval: <https://github.com/vuejs/vue>. [Kasutatud november 2021].
- [9] „Pros and Cons of Vue.js Framework Programming,“ DDI Development, detsember 2020. [Võrgumaterjal]. Saadaval: <https://ddi-dev.com/blog/programming/the-good-and-the-bad-of-vue-js-framework-programming/>. [Kasutatud november 2021].
- [10] „Why Aurelia?,“ Aurelia, [Võrgumaterjal]. Saadaval: <https://aurelia.io/>. [Kasutatud november 2021].
- [11] „aurelia/aurelia,“ GitHub, [Võrgumaterjal]. Saadaval: <https://github.com/aurelia/aurelia>. [Kasutatud november 2021].
- [12] „Angular applications: The essentials,“ Angular, [Võrgumaterjal]. Saadaval: <https://angular.io/guide/what-is-angular>. [Kasutatud november 2021].
- [13] „angular/angular,“ GitHub, [Võrgumaterjal]. Saadaval: <https://github.com/angular/angular>. [Kasutatud november 2021].
- [14] „Amazon Relational Database Service (RDS),“ Amazon, [Võrgumaterjal]. Saadaval: <https://aws.amazon.com/rds/>. [Kasutatud november 2021].
- [15] „Amazon RDS Pros and Cons – A detailed overview,“ Saras Analytics, 6 aprill 2020. [Võrgumaterjal]. Saadaval: <https://sarasanalytics.com/blog/amazon-rds-pros-and-cons>. [Kasutatud november 2021].
- [16] „App Service,“ Microsoft, [Võrgumaterjal]. Saadaval: <https://azure.microsoft.com/en-us/services/app-service/#features>. [Kasutatud november 2021].
- [17] „App Service overview,“ Microsoft, 22 oktoober 2021. [Võrgumaterjal]. Saadaval: <https://docs.microsoft.com/en-us/azure/app-service/overview>. [Kasutatud november 2021].

- [18] „DiskStation DS718+,“ Synology, [Võrgumaterjal]. Saadaval: https://global.download.synology.com/download/Document/Hardware/DataSheet/DiskStation/18-year/DS718+/enu/Synology_DS718_Plus_Data_Sheet_enu.pdf. [Kasutatud november 2021].
- [19] „System Properties Comparison MariaDB vs. Microsoft SQL Server vs. MySQL vs. PostgreSQL,“ Solid IT, [Võrgumaterjal]. Saadaval: <https://db-engines.com/en/system/MariaDB%3BMicrosoft+SQL+Server%3BMySQL%3BPostgreSQL>. [Kasutatud november 2021].
- [20] „MySQL,“ Oracle, [Võrgumaterjal]. Saadaval: <https://www.mysql.com/>. [Kasutatud november 2021].
- [21] J. Hurley, „The Pros and Cons of MySQL,“ 3 juuni 2021. [Võrgumaterjal]. Saadaval: <https://www.smartfile.com/blog/the-pros-and-cons-of-mysql/>. [Kasutatud november 2021].
- [22] R. Peterson, „MariaDB vs MySQL: What is the Difference Between MariaDB and MySQL,“ 19 oktoober 2021. [Võrgumaterjal]. Saadaval: <https://www.guru99.com/mariadb-vs-mysql.html>. [Kasutatud november 2021].
- [23] „What you'll love about SQL Server 2019,“ Microsoft, [Võrgumaterjal]. Saadaval: <https://www.microsoft.com/en-us/sql-server/sql-server-2019>. [Kasutatud november 2021].
- [24] „SQL Server 2019 pricing,“ Microsoft, [Võrgumaterjal]. Saadaval: <https://www.microsoft.com/en-us/sql-server/sql-server-2019-pricing?rtc=1>. [Kasutatud november 2021].
- [25] R. Pijacek, „Microsoft SQL Server Pros and Cons,“ 17 jaanuar 2019. [Võrgumaterjal]. Saadaval: <https://learnsql.com/blog/microsoft-sql-server-pros-and-cons/>. [Kasutatud november 2021].
- [26] „About,“ The PostgreSQL Global Development Group, [Võrgumaterjal]. Saadaval: <https://www.postgresql.org/about/>. [Kasutatud november 2021].
- [27] R. Peterson, „What is PostgreSQL? Introduction, Advantages & Disadvantages,“ 7 oktoober 2021. [Võrgumaterjal]. Saadaval: <https://www.guru99.com/introduction-postgresql.html>. [Kasutatud november 2021].
- [28] „LAMP Stack,“ IBM Cloud Educaion, 9 mai 2019. [Võrgumaterjal]. Saadaval: <https://www.ibm.com/cloud/learn/lamp-stack-explained>. [Kasutatud november 2021].
- [29] „About MariaDB Server,“ MariaDB Foundation, [Võrgumaterjal]. Saadaval: <https://mariadb.org/about/>. [Kasutatud november 2021].
- [30] „Drifti Võistlusmäärus,“ Eesti Autospori Liidu Driftikomitee, 3 juuni 2021. [Võrgumaterjal]. Saadaval: <https://www.driftime.ee/wp-content/uploads/2021/06/2021-DRIFTi-vo%CC%83istlusma%CC%88a%CC%88ruskinnitatud-03.06.2021.pdf>. [Kasutatud november 2021].
- [31] „Apache JMeter,“ Apache Software Foundation, [Võrgumaterjal]. Saadaval: <https://jmeter.apache.org/>. [Kasutatud aprill 2022].
- [32] „Azure Compute Unit,“ Microsoft, 11 veebruar 2021. [Võrgumaterjal]. Saadaval: <https://docs.microsoft.com/en-gb/azure/virtual-machines/acu>. [Kasutatud aprill 2022].
- [33] „Overview of ASP.NET Core SignalR,“ Microsoft, 26 märts 2022. [Võrgumaterjal]. Saadaval: https://docs.microsoft.com/en-gb/aspnet/core/signalr/introduction?WT.mc_id=dotnet-35129-website&view=aspnetcore-6.0. [Kasutatud aprill 2022].

Lisa 1 - Lihtlitsents

Mina, Kaarel-Martin Maarand

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose "Veebipõhise Keskkonna Arendamine Driftivõistluse Korraldamiseks" , mille juhendaja on Toomas Lepikult
 - 1.1. reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
 - 1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.
3. Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

Lisa 2 – Tarkvara koodifailid

Arendatud tarkvara eessüsteemi koodifailile link:

<https://gitlab.cs.ttu.ee/kamaar/react-drift>

Arendatud tarkvara tagaprogrammi koodifailile link:

<https://gitlab.cs.ttu.ee/kamaar/icd0009-2020s>

Lingid on ligipääsetavad kasutajatele, kellel on selleks volitatud õigused.

Lisa 3 – Intervjuud

Küsimused Eesti Drifti Liidu sekretärile

1) Kuidas hetkel toimub ürituse läbiviimine?

Hetkel toimib süsteem nii, et kodulehelt ankeedilt saadetakse registreerunute info meilile kust info üks haaval käsitsi Exceli tabelisse sisestatakse. Ürituse alguses koostatakse käsitsi erinevad nimekirjad mis antakse paberkujul vajaminevatele korraldustiimi liikmetele laiali. Kui enne registreerimise lõppu on tarvis võistlusjuhil infot kes on osalejad siis tehakse eraldi tabel mis antakse võistlusjuhile. Kui võistlejad on registreerunud siis vormistatakse lõplik võistlejate nimekiri igale võistlusklassile ning pannakse võistlusnumbri järjekorda. Järgmisena hindajate hindamise tabelisse sisestame võistlejate nimekirja ja prindime selle tabeli välja ning anname hindajatele. Kvalifikatsiooni sõitude ajal iga hindaja täidab käsitsi oma hindamise ala käsitsi paberi peal. Kui kvalifikatsiooni sõidud on läbi viiakse täidetud lehed sekretariaati kus neil lehtedel olevad andmed sisestatakse käsitsi Excelisse. Järgmisena kopeeritakse kvalifikatsiooni hindamise tabelist sõitjate nimekiri eraldi kvalifikatsiooni tulemuste tabelisse mis automaatselt täidab tulemuste tabeli punktidega. Järgmisena kopeerin kõrval tabelisse kvalifikatsiooni tulemuste tabelist andmed ja sorteerin need paremuse järjekorda ning prindin sorteeritud kvalifikatsiooni tulemuste tabeli välja. Sorteeritud kvalifikatsiooni tulemuste tabeli järgi koostan võistluse eliminaatoritabeli. Sorteeritud kvalifikatsiooni tulemuste järgi saab kätte juba ka võistluse kvalifikatsiooni punktid mis lähevad võistluse lõpp punktisumma arvestusse. Võistluse lõpp punkti summa summeeritakse samuti käsitsi. Eliminaator tabelit täidetakse jooksvalt käsitsi.

2) Kes oleksid loodava programmi kasutajad korraldustiimist?

Sekretär, hindajad, võistlusejuht, kommentaatorid, videotiim (meedia)

3) Mis seadmetel peaks programm töötama?

Arvutis ja nutitelefonis

Küsimused Eesti Drifti Komitee esimehele Kristjan Salmrele

1) Miks on tarvis Eesti Drifti Liidul enda programmi ürituse läbiviimiseks?

Et lihtsustada ja kiirendada süsteemi toimimist. Teha registreerimine ja punktisüsteem läbipaistvamakas.

2) Kas ja mis on põhilised erinevused Eesti Drifti Liidu võistlus reeglites võrreldes meie naaber riikidega (5 minuti reegel, pausid, hindamine, kvalifikatsioonid jms)?

Erinevused põhimõtteliselt puuduvad, lätlastel on osad omareeglid autovahetuste kohta.

3) Mis on põhilised erinevused Eestis läbiviidavate ürituste korralduses võrreldes meie naaber riikidega?

Eestis võistlusjuhi töö on oluliselt kaasavam.

4) Kas Eesti naaberriikidel on kasutuses programmid ürituste läbiviimiseks (v.a. Drift Masters)? Kui jah siis miks need programmid meile ei sobi?

Programmid on puudulikud. Soomes on Exceli põhine programm. Läti programm on aeglane ning pole nähtud veel seda korrektselt töötamas, nad üritasid siduda videoga kuid see ei töötanud.

5) Kes oleksid loodava programmi kasutajad?

Kõik kel soovi.

6) Mis seadmetel peab programm töötama?

Arvutis ja nutiseadmetes.

7) Kas drifti hindamisel tulevikus kasutatakse ka kiiruse punkte?

Ei.

Küsitlus pealtvaatajatele

Küsitluses osales 32 inimest. Küsitluse vastuse variantides toob autor välja valik vastuse vastanute protsentuaalse väärtuse. Avatud tekstiga vastuste tulemusi kasutab autor analüüsis jälgides mustreid mis ja miks inimestele meeldib või ei meeldi ning mida nad sooviksid, et tehtaks teisiti.

1) Sugu

a. Mees (84,4%)

b. Naine (15,6%)

2) Kui tihti külastad driftivõistlust aastas?

a. Ei ole veel külastanud (9,4%)

b. 1 kord aastas (9,4%)

c. 2-3 korda aastas (18,8%)

- d. Rohkem kui 3 korda aastas (62,4%)
- 3) Kuidas hoiate end reaalajas tulemustega kursis?
- a. Jälgides üritust / kommentaatorilt (68,8%)
 - b. Paber kandja abil (18,8%)
 - c. Tulemuste seinalt (28,1%)
- 4) Milliseid muutuseid Te tahaksite näha pealtvaatajana mis teeks ürituse meeldivamaks pealtvaatajatele? (Avatud tekstiga vastus)
- 5) Kas ja kuidas te jälgite ürituste tulemusi ja/või üritust ennast, kui Teil pole võimalik kohapeale vaatama minna?
- a. Ei jälgi üritusi eemalt (25%)
 - b. Jälgin tulemusi sotsiaalmeedia vahendusel (43,8%)
 - c. Jälgin tulemusi drifttime.ee leheküljelt (9,4%)
 - d. Jälgin reaalajas videoülekannet (43,8%)
- 6) Kas tutvud enne võistlust sõitjate rivistusega?
- a. Jah (50%)
 - b. Ei (50%)
- 7) Kui vastasid eelmisele küsimusele “Jah”, siis kuidas? Kui “Ei” siis miks? (Avatud tekstiga vastus)
- 8) Milliseid funktsioone sooviksid leida drifti veebikeskkonnas?
- a. Korraldus meeskonna liikmete info (28,1%)
 - b. Otseülekanne (78,1%)
 - c. Reaalajas tulemused (71,9%)
 - d. Sõitjate rivistus konkreetse ürituse kohta (50%)
 - e. Sarja tulemused (53,1%)
 - f. Info tiimide, sõitjate, autode ja sponsorite kohta (53,1%)
 - g. Ürituse üldine info (asukoht, hind, ajakava) (75%)
 - h. Ürituse plaan (wc, toitlustus, boksiala, pealtvaatajate ala, parkla jms.) (65,6%)
- 9) Veel soovitusi/ideid drifti paremini jälgimiseks (Avatud tekstiga vastus)
- a. Kui oleks pileti ostmine appis siis see võiks olla nii et kui inimene ostab endale appist pileti siis tuleb talle näiteks triipkood sinna appi ja kui võistlustele jõuab siis pileti kontrollijad saavad skaneerida selle triipkoodi ja ongi kõik
 - b. Rohkem Web live....Top 32 võiks tasuta olla ja top 16 kuni 5 euri web pileti.
 - c. Kuna driftmasters ja FD on veebis tasuta jälgitavad siis võiks ka Eesti võistlused olla kättesaadavamad kõigile, kui pole võimalik füüsiliselt kohal olla

- d. Võistlejate nimekirjas autode ära mainimine. Lihtsab jälgida uute sõitate tegemisi, kelle nimi pole veel pähe õpitud.
- e. Tunnen puudust vettpidavast ajakavast. Mitte ainult klasside alguse/lõpu piires vaid sooviksin äpi kaudu näha kasvõi mõne minuti täpsusega millal lemmiksõitjaid starti võib oodata, et teada millal on ohutu põiele või sööma minna.
- f. Korraldus võiks parem olla, nii pealtvaataja kui ka osaleja jaoks.
- g. Rohkem reklaami, et teaks aegu sättida ja kohale vaatama sõita. Hind on ka kindlasti määrav ja võiks olla perepilet ja kui üritus on kahel päeval siis ka kahepäevapilet perele.

Küsitlus võistlejatele

Küsitluses osales 8 inimest. Küsitluse vastuse variantides toob autor välja valik vastuse vastanute protsentuaalse väärtuse. Avatud tekstiga vastuste tulemusi kasutab autor analüüsis jälgides mustreid mis ja miks inimestele meeldib või ei meeldi ning mida nad sooviksid, et tehtaks teisiti.

- 1) Kuidas Te hoiate end reaalajas tulemustega kursis?
 - a. Meeskonna liikmetelt küsides (50%)
 - b. Sekretariaadi info seinalt (20%)
 - c. Üritust jälgides ja kommentaatoritelt (75%)
 - d. Tiimiliige jälgib (37,5%)
- 2) Kas Te olete hetkel rahul üritusele registreerimise süsteemiga?
 - a. Jah (62,5%)
 - b. Ei (37,5%)
- 3) Kui vastasid eelmisele küsimusele “EI” siis mida saaks paremini teha? - Kui vastasid “JAH” siis miks? (Avatud tekstiga vastus)
 „JAH“ vastanute seas väidavad võistlejad, et süsteem on piisavalt lihtne või jätavad vastuse tühjaks. „EI“ vastanute seas väidavad kõik, et topelt registreerimis andmete täitmine on ebamugav.
- 4) Kas Te olete hetkel rahul sarja osavõtu registreerimise süsteemiga?
 - a. Jah (100%)
 - b. Ei (0%)

- 5) Kui vastasid eelmisele küsimusele "EI" siis mida saaks paremini teha? - Kui vastasid "JAH" siis miks? (Avatud tekstiga vastus)
Kõik vastanud vastasid „JAH“ ning põhjendasid seda, et on piisavalt lihtne.
- 6) Kuidas te olete rahul sõitude hindamisga? Skaala 1-5 Hinnatavad aspektid:
- Läbinähtavus (1 – 0%, 2 – 10%, 3 – 50%, 4 – 10%, 5 – 10%)
 - Arusaadavus (1 – 0%, 2 – 10%, 3 – 40%, 4 – 10%, 5 – 20%)
 - Kajastamine (1 – 0%, 2 – 10%, 3 – 40%, 4 – 10%, 5 – 20%)
- 7) Milliseid funktsioone kasutaksid uues drifti veebikeskkonnas?
- Korraldus meeskonna liikmete info (50%)
 - Reaalajas tulemused (100%)
 - Ajakava ja/või selle muudatused (100%)
 - Info rajal toimuva kohta (võistluse seisakud / info millal lineup-is olla / kes hetkel sõidab) (100%)
 - Registreerimine (hooaeg / sari / võistlus) (87,5%)
 - Ürituse üldine info (87,5%)
 - Sõitjate rivistus konkreetse ürituse kohta (87,5%)
 - Sarja tulemused (87,5%)
 - Info tiimide, sõitjate, autode ja sponsorite kohta (75%)
 - Ürituse üldine info (asukoht, hind, ajakava) (87,5%)
 - Ürituse plaan (wc, toitlustus, boksiala, pealtvaatajate ala, parkla jms.) (100%)
- 8) Muud ideed ja soovid veebikeskkonnale (Avatud tekstiga vastus)
Ainus esitatud soov oli, et info liikumine võiks olla ajakohane.

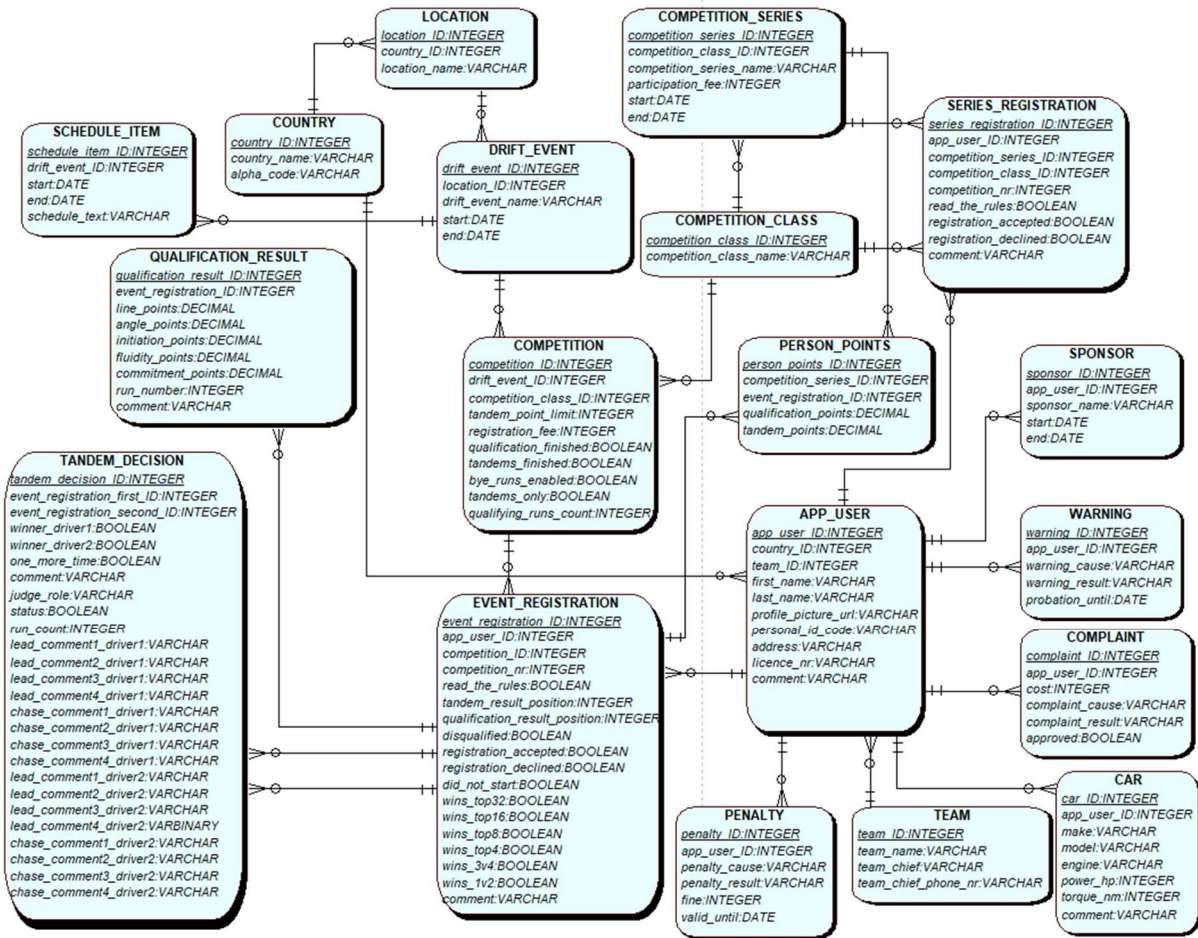
Küsitlus korraldustiimile

Küsitluses osales 10 inimest. Küsitluse vastuse variantides toob autor välja valik vastuse vastanute protsentuaalse väärtuse. Avatud tekstiga vastuste tulemusi kasutab autor analüüsis jälgides mustreid mis ja miks inimestele meeldib või ei meeldi ning mida nad sooviksid, et tehtaks teisiti.

- 1) Kui mugav on ürituse ajal võistlusel toimuvat jälgida? Skaalal 1-5 (1=ebamugav, 5=mugav) Hinnatavad aspektid:
- Tulemused (1 – 0%, 2 - 20%, 3 - 20%, 4 – 40%, 5 – 20%)
 - Ajakava muutused (1 - 10%, 2 – 20%, 3 – 30%, 4 – 40%, 5 – 20%)
 - Info kättesaadavus (1 - 0%, 2 – 0%, 3 – 80%, 4 – 20%, 5 – 0%)

- 2) Mis funktsionaalsus veebikeskkonnas aitaks sinu tööülesandeid mugavamalt ja kiiremini täita? Valiku variandid:
- a. Korraldus meeskonna liikmete info (80%)
 - b. Realajas tulemused (80%)
 - c. Ajakava ja/või selle muutused (0%)
 - d. Info rajal toimuva kohta (võistluse seisakud / info millal lineup-is olla / kes hetkel sõidab) (100%)
 - e. Registreerimine (hooaeg / sari / võistlus) (60%)
 - f. Sõitjate rivistus konkreetse ürituse kohta (60%)
 - g. Info tiimide, sõitjate, autode ja sponsorite kohta (80%)
 - h. Ürituse plaan (wc, toitlustus, boksiala, pealtvaatajate ala, parkla jms.) (90%)
 - i. Ürituse üldine info (asukoht, hind, ajakava) (90%)
 - j. Sarja tulemused (40%)
 - k. Sõitude hindamine (10%)
- 3) Milliseid puudusi esineb hetkel drifti korralduses? (Avatud tekstiga vastus)
- 4) Ettepanekuid parema veebikeskkonna loomiseks (Avatud tekstiga vastus)

Lisa 4 – ERD skeem



Lisa 5 – Olemite semantika

Tabeli nimi	Semantika
APP_USER	Siin tabelis hoitakse kõikide kasutajate andmeid mis olenevad riigist ja tiimist.
EVENT_REGISTRATION	Siin tabelis hoitakse kõike võistluse registreerimis andmeid olenevalt võistlusest, ja kasutajast.
SERIES_REGISTRATION	Siin tabelis hoitakse kõike võistlussarja registreerimis andmeid olenevalt võistlusklassist, võistlussarjast ja kasutajast.
PENALTY	Siin tabelis hoitakse isiku karistus andmeid olenevalt kasutajast.
COMPLAINT	Siin tabelis hoitakse pretensioonide andmeid olenevalt kasutajast.
WARNING	Siin tabelis hoitakse hoiatus andmeid olenevalt kasutajast.
CAR	Siin tabelis hoitakse andmeid auto kohta olenevalt kasutajast.
TEAM	Siin tabelis hoitakse andmeid tiimi kohta.
SPONSOR	Siin tabelis hoitakse sponsorite andmeid olenevalt kasutajast.
DRIFT_EVENT	Siin tabelis hoitakse andmeid drifti ürituse kohta olenevalt asukohas.
COMPETITION_SERIES	Siin tabelis hoitakse kõikide võistlussarjade andmeid olenevalt võistlusklassist.
LOCATION	Siin tabelis hoitakse asukoha andmeid olenevalt riigist.
COUNTRY	Siin tabelis hoitakse kõikide riikide andmeid.
COMPETITION_CLASS	Siin tabelis hoitakse kõikide tasemeklasside andmeid.
PERSON_POINTS	Siin tabelis hoitakse kõikide isikute võistlustelt teenitud punkte olenevalt võistluse registreerimisest ja võistlussarjast.
COMPETITION	Siin tabelis hoitakse kõikide võistluste andmeid olenevalt drifti üritusest ja võistlusklassist.

Tabeli nimi	Semantika
QUALIFICATION_RESULT	Siin tabelis hoitakse kõikide kvalifikatsiooni tulemuste andmeid olenevalt võistluse registreerimisest.
TANDEM_DECISION	Siin tabelis hoitakse kõikide paarissõidu tulemuste andmeid olenevalt kahest võistluse registreerimisest.
SCHEDULE_ITEM	Siin tabelis hoitakse kõikide võistluse ajakava kirjete andmeid olenevalt drifti üritusest.