

TALLINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

Tarkvarateaduse instituut

Taaniel Tüür 135045IAPB

**EESTI AERUTAMISFÖDERATSIOONI
HOOAJA HALDAMISE INFOSÜSTEEM**

bakalaureusetöö

Juhendaja: Rein Paluoja

PhD

Tallinn 2018

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 viidatud.

Autor: Taaniel Tüür

20.05.2018

Annotatsioon

Antud lõputöö eesmärk on luua hooaja haldamise infosüsteem Eesti Aerutamisföderatsioonile, mis on eelkõige mõeldud kõigile tema liikmetele võistluste korraldamiseks, sportlaste ülesandmiseks ning tulemuste sisestamiseks. Eeldatavalt hakkab süsteem hooaja jooksul kümnekonna võistluse ning võistlusel osaleva ligikaudu 150mne sportlase ja paatkonna haldamise pealt kokku hoidma ajakulu, välistama vea tekkimise võimalusi ning vähendama vajaminevat inimressurssi võistluse korraldaja klubil võrreldes senise Microsoft Excelil põhineva haldusega. Süsteemi loomisel on järgitud MVC arhitektuuri põhimõtteid ning kogu klient-serveri vaheline suhtlus käib RESTful teenuse kaudu. Töö tulemusena valmib veebirakendus, mis koosneb kasutajaliidesest, andmebaasist ja põhiprogrammist. Rakenduse põhiülesanded on:

- Kasutaja ehk treeneri autentimine ja valideerimine
- Sportlaste ja paatkonna deklareerimine
- Võistluste loomine
- Võistleja esitamine võistlusele
- Sõitude loosimine
- Tulemuste sisestamine
- Tulemas oleva võistluse emaili teavituste väljastamine
- Finiši- ja stardiprotokolli väljastamine Exceli formaati
- Finišiprotokollide saatmine kõikidele treeneritele

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 30 leheküljel, 4 peatükki, 28 joonist, 7 tabelit.

Abstract

Estonian Canoeing Federation season management information system

The aim of the current thesis is to develop season management information system for Estonian Canoeing Federation. Presumably the system will spare time, exclude possibilities for errors and reduce necessity for human resource when it comes to organizing, applying athletes to competitions and adding results, compared to the old Microsoft Excel based management. The system will follow the MVC architectural pattern and all the client-server communication uses RESTful services. The software contains userinterface, database and main program logic. The main components of the system are:

- User authentication
- Declaring athletes and teams
- Creating competitions
- Submitting athletes and teams to competition
- Generating heats, semi-finals and finals
- Inputting results
- Inform coaches about upcoming competitions by email
- Exporting start list and final entries to xlsx
- Sending start list to coaches by email

The thesis is in Estonian and contains 30 pages of text, 4 chapters, 28 figures, 7 tables.

Lühendite ja mõistete sõnastik

EAF	Eest Aerutamisföderatsioon
ICF	International Canoeing Federation
URL	<i>Uniform Resource Locator</i> – Universaalne ressursilokaator
JavaScript	Kõrgtasemeline, dünaamiline ja nõrgalt tüpiseeritud skriptimiskeel, mis on standardiseeritud ECMAScript-i poolt
CSS	<i>Cascading Style Sheets</i> – programmeerimiskeel, millega kirjeldatakse, kuidas veebileht peaks välja nägema
MVC	<i>Model-view-controller</i> – arhitektuurimuster, mis jagab rakenduse mudeliks, vaadeteks ja kontrolleriiks
IDE	<i>Integrated development environment</i> – integreeritud programmeerimiskeskond
REST	<i>Representational State Transfer</i> – on tarkvaraarhitektuuri stiil, mida kasutatakse veebiteenuste arendamisel ning mille eesmärk pakkuda ligipääsu ressurssidele ühtse liidestuse kaudu.
PHP	<i>Hypertext Preprocessor</i> on skriptimiskeel serveripoolsete rakenduste loomiseks
Labengine	PHP raamistik, mis on omandi tarkvara loodud Codelab Solutions OÜ poolt.

Sisukord

1 Sissejuhatus.....	11
2 Senine võistluste korraldamise protsess.....	12
2.1 Võistluste erinevad tüübid	12
2.2 Sõitude loosimine.....	13
2.3 Tulemuste sisestamine	13
2.4 Nõuded.....	14
2.4.1 Süsteemi administraator.....	14
2.4.2 Treener	15
2.4.3 Võistluse loomine	15
2.4.4 Sportlaste esitamine	16
2.4.5 Sõitude loosimine	17
2.4.6 Protokollide printimine ja saatmine.....	19
2.4.7 Tulemuste sisestamine	19
2.5 Vahendid	20
3 Analüüs ja arendus.....	22
3.1 Kasutusjuhtude diagramm	22
3.2 Olemi-suhte diagrammid	23
3.3 Failistruktuur.....	25
3.4 Operatsioonid.....	25
3.5 Kontrollerid.....	26

3.6 Tõlked	26
3.7 Objektid.....	27
3.8 Mallid ja staatilised failid.....	28
4 Kasutuslood võistluse korraldaja vaatest.....	29
4.1 Sisse logimine	29
4.2 Sportlaste sisestamine andmebaasi	30
4.3 Võistluste loomine	32
4.4 Võistlusele registreerimine	35
4.5 Tulemuste sisestamine ja loosimine.....	38
4.6 Toimunud võistlused, tulemuste järele vaatamine.....	40
Kokkuvõte.....	41

Jooniste loetelu

Joonis 1. Kasutusjuhtude diagramm	22
Joonis 2. Võistluse olemi-suhte diagramm	23
Joonis 3. Sõidu olemi-suhte diagramm	23
Joonis 4. Sportlase ja võistleja olemi-suhte diagramm.....	24
Joonis 5. Kasutaja olemi-suhte diagramm'	24
Joonis 6. Labenginist tulenev failistruktuur näidatuna PhpStormis.....	25
Joonis 7. Klubi vaadetega seotud klassid.....	26
Joonis 8. Võistluste nimekirja vaate kontrollid.....	26
Joonis 9. Eesti keelse tõlke märgend	26
Joonis 10. Kasutaja rolli kontrollimise meetod.....	27
Joonis 11. Sisse loginud kasutaja põhivaade	28
Joonis 12. Sisse logimise vaade	29
Joonis 13. Sisse loginud Pärnu Aerutamisklubi liikme vaade	30
Joonis 14. Sportlaste sisestamise vaade Pärnu Aerutamisklubi näitel.....	30
Joonis 15. Sportlase lisamise hüpinkaken.....	31
Joonis 16. Paatkonna lisamise esimene hüpinkaken.....	31
Joonis 17. Paatkonna lisamise teine hüpinkaken	32
Joonis 18. Võistluste nimekirja vaade	32

Joonis 19. Võistluse loomise vaade	33
Joonis 20. Võistluse vaade.....	33
Joonis 21. Uue distantsi loomine võistlusel.....	34
Joonis 22. Võistluse vaade, kui distantsid loodud ja klubide ülesandmised tehtud.....	35
Joonis 23. Sportlaste lisamise vanusegruppide hüplikaken	36
Joonis 24. Täiskasvanud sportlaste ja paatkondade esitamine võistlusele	36
Joonis 25. Loositud 200m sõidud süsteemi vaates	37
Joonis 26. Sportlase või paatkonna lisamine sõitu.....	38
Joonis 27. Aegade sisestamise vaade.....	38
Joonis 28. Sõidu algusaja muutmise vaade.....	39

Tabelite loetelu

Tabel 1. Stardiprotokolli näidis.....	13
Tabel 2. Finišiprotokolli näidis.....	14
Tabel 3. Kohandatud üheksa raja valemite tabel	18
Tabel 4. Kohandatud kuue raja valemite tabel.....	18
Tabel 5. 200m PE vanuseklassi stardiprotokoll xlsx faili kujul.....	37
Tabel 6. Finišiprotokoll PE vanuseklassi eelsõitude põhjal xlsx faili kujul	39
Tabel 7. 200m PE vanuseklassi loositud poolfinaalid xlsx faili kujul.....	40

1 Sissejuhatus

Eesti Aerutamisföderatsioon koos tema alla kuuluvate spordiklubidega korraldavad hooaja jooksul mitmeid võistlusi, mille korral esitatakse korraldaja klubile nimekiri seal osalevatest sportlastest. Nimekiri koosneb sportlase nimest, vanuseklassist, paadiklassist ning distantsidest milles sportlane kavatseb osaleda. Probleemide olemus seisneb selles, et tihti peale on esitatav nimekiri sellises formaadis nagu esitajale parasjagu mugav koostada oli – Exceli tabel, pilt sellest või paber kandjal, mille tulemusena on trükivead lihtsad tulema. Lisaks on korraldajale suur töö vahetult enne võistlust ning võistluse ajal teha manuaalselt erinevate sõitude loosimist ning tulemuste sisestamist.

Antud töö eesmärgiks saab olema luua multifunktsionaalne tööriist aerutamisevõistlusi korraldavatele treeneritele, mis hakkab asendama siiani kasutatavat MS Exceli programmi. Platvorm hakkab pakkuma võimalusi, kuidas lihtsalt võistlusi luua, sportlasi/paatkondi lisada ja esitada võistlustele, automaatset sõitude loosimist ning tulemuste sisestamise ja eksportimise võimalusi. Süsteem peab autonoomselt haldama sportlaste vanusegrupe, mis muutuvad igal aastal ning saatma kõikidele treeneritele informatiivseid emaile uue võistluse tulekul. Töö teeb keerukaks asjaolu, et Eesti aerutajate kogukond on suhteliselt väike ning selle pärast tuleb süsteemi kirjutada väga palju erandeid.

2 Senine võistluste korraldamise protsess

Iga hooaja alguses pannakse paika võistluskalender, mida kõik Eesti Aerutamise föderatsiooni liikmed on kohustatud järgima. Hetkel tegutseb on Eesti Aerutamise föderatsioonil 13 liiget, milleks on Pirita Aerutamisklubi, Sõudmise ja Aerutamise klubi "Tartu", TÜASK, Pärnu Kalevi Aerutajate Klubi, Paikuse Aerutamisklubi, Spordiühing Ekstreempark, Spordiklubi Narva Viikingid, Eesti Spordiselts Põhjakotkas SK, Aidu Spordiklubi, Emajõe Aeruklubi, Eesti Draakonaerutamise Ühendus, Viljandi Aerutamisklubi, Tartu Ülikooli Akadeemiline Spordiklubi ja Emajõe Aeruklubi ja Pärnu Spordiselts Kalev [1], kus iga klubiga seotud linnas toimub aasta jooksul vähemalt üks võistlus ning mille korraldajaks on see sama klubi. Võistluse algus kuupäeva saabudes saadavad teised klubid korraldajale enda sportlaste nimekirja, kes kavatsevad võistlusel osaleda. Nimekiri koosneb sportlase ees- ja perekonnanimest, vanuseklassist, paadiklassist ning distantsidest. Üldiselt päev enne võistlust pannakse nimekiri lukku ning korraldaja klubi hakkab sõite loosima.

2.1 Võistluste erinevad tüübid

Eestis sõidetavateks vanuseklassideks on vanuselises järjekorras M, N, A, B, C, D, ja E vanuseklassid, mis omakorda jagunevad sportlase soo järgi: poisid E on PE ja tüdrukud A on TA. Sellisel kujul on M – mehed, N – naised ning näiteks A on juuniorid, mis on 2018 aasta arvestuse järgi kõik noored, kes on sündinud vahemikus 2000 – 2001 aastal ning teise näitena E klass, kuhu kuuluvad kõik sportlased, kes on sündinud peale 2007ndat aastat. Lisaks peab korraldaja arvestama paadiklassidega, milleks on K-1 (ühene süst), C-1 (ühene kanuu), K-2 (kahene süst), C-2 (kahene kanuu) ja K-4 (neljane süst) ning sportlase sooga.

Üldiselt korraldatakse Eesti rajaaerutamise võistlusi distantsidel, kus on kuus või üheksa rada. Kui esitatud sportlasi on sama palju või vähem, siis tuleb kohe finaali. Kui on rohkem kui ühe sõidu ja vähem kui kahe sõidu jagu sportlasi, siis tuleb 2 poolfinaali.

Kolme sõidu puhul muster kordub, kuid nelja korral tehakse Eestis juba neli eelsõitu, kust paremad saavad edasi poolfinaali.

2.2 Sõitude loosimine

Võistluse koostamisel ja haldamisel on siiani olnud peamiseks töövahendiks Microsoft Excel, milles luuakse alamtabelid iga sõidu kohta. Iga tabeliblokk peab koosnema sõidu numbriga, kella ajaga, distantsi pikkusega, loositud paadi numbriga, paadiklassiga, vanusegrupi ja sportlase informatsiooniga, mis printitakse välja ja jagatakse võistluse päeval treeneritele laiali.

Sõitude loosimisel arvestatakse ka sportlase taset. Kuna Eesti aerutajate kogukond on suhteliselt väike ja tase on väga kõikumine, siis on kombeks eelsõitude loosides panna paremad sportlased keskmistele radadele. Järgnevate sõitude stardinimekirjade järjestus sõltub aga eelmiste tulemustest.

Selles peatükis kirjeldatud sõitude loosimine kehtib ainult sprindidistantside kohta, sest intervall ja ühisstardi kohta käiv loogika on oluliselt lihtsam.

1	12:00	N K-1	1	Nimi1	Perenimi1	1999	Põhjakotkas
		1000	2	Nimi2	Perenimi2	1990	Pirita
		Heat 1	3	Nimi3	Perenimi3	2002	Pärnu
			4	Nimi4	Perenimi4	1984	Narva
			5	Nimi5	Perenimi5	2001	Põhjakotkas

Tabel 1. Stardiprotokolli näidis

2.3 Tulemuste sisestamine

Võistlustulemuste selgudes lisatakse stardiprotokollist tulenenud tabelisse võistlejate ajad ning järjestatakse aja kasvavas järjekorras. Tabelisse tuleb kanda ka lisainformatsioon, milliste reeglite alusel edasi pääsetakse. 3F + 3BT tähendab, et eelsõidust finaali saavad kolm esimest ning lisaks 3 parimat aega (Tabel 2).

Sõitude loosimisel kasutatakse ICF-i reegleid mida iga klubi vastavalt sportlaste arvule ja vajadustele natukene kohandab [2]. Näiteks kuni 18-kümne sportlase osavõtul tuleks

kõigepealt sõita eelsõidud ning alles siis poolfinaalid aga Eestis on kombeks aja kokkuhoiu mõttes sõita kohe poolfinaalid.

Finaali puhul arvutatakse käsitsi iga sportlase ja distantsi kohta teenitud karikapunktid kokku ning kantakse üldisesse Exceli tabelisse, mille alusel hooaja lõpus saavad parimad omale karika. Karikapunktide saamise loogika on järgmine: esimene koht saab 12 punkti, teine koht 9 punkti, kolmas koht 7 punkti, neljas koht 6 punkti jne [3].

1	09:00	M K-1	1	Nimi1	Perenimi1	1990	SAK Tartu	1:54,8
		500	2	Nimi2	Perenimi2	1998	Põhjakotkas	1:56,2
		Heat 1	3	Nimi3	Perenimi3	1982	TÜASK	2:01,0
		3 F + 3 BT	4	Nimi4	Perenimi4	1991	TÜASK	2:02,0
			5	Nimi5	Perenimi5	1997	Põhjakotkas	2:09,7
			6	Nimi6	Perenimi6	1999	Pirita	2:13,2

Tabel 2. Finišiprotokolli näidis

2.4 Nõuded

Võttes arvesse kõik selles peatükis kirjeldatu, paneb autor kokku nõuded loodava süsteemi jaoks. Kuna sportlasi tunneb kõige paremini treener ning tema on ka see, kes võistlusi korraldab, siis süsteemi peamiseks kasutajaks saab olema just tema. Lisaks on vaja ka administraatori õigustega kasutajaid, kes saavad omakorda hallata kõikide klubide treenereid ning muid andmeid, millele ligipääsu treeneritel igapäevaselt vaja pole nagu näiteks klubide, vanuseklasside ja paadiklasside redigeerimist.

2.4.1 Süsteemi administraator

Süsteemi administraator peab rakenduse kasutusele võtu alguses looma vanusegrupid, paadiklassid, klubid, kasutajad ning deklareerima distantsid. Iga vanusegrupp peab olema defineeritud soo, sünniaastate vahemiku ning vanusegrupi nimega, kus iga sportlane saab kuuluda ainult ühte vanusegruppi, mis ajas muutub. Järgmisena tuleks defineerida paadiklassid, kus iga väli koosneb paadiklassi nimest, sportlaste arvust paadis ning tüübist: kanuu või süst. Viimasena peaks looma distantsid, kus iga distants koosneb raja pikkusest ning tüübist: intervall, ühisstart või sprint. Kõik need väärtused peaksid olema redigeeritavad ja mitte süsteemi staatiliselt kirjutatud, sest ajajooksul võivad vajadused muutuda. Näiteks paadiklassi saaks lisada draakonaerutamise, kus on

kaheksa meeskonnaliiget. Distantside puhul vastupidise näitena saab tuua, et need kattuvad üldjuhul aastate lõikes ehk neid ei ole mõtet iga kord uuesti defineerida aga samas vastavalt vajadusele saab selle võistluse jaoks luua.

2.4.2 Treener

Iga treener peab kuuluma ainult ühte klubisse, kuid igal klubil saab olla mitu treenerit. Antud süsteemi puhul on klubi kõiges klassifikaator, mis on mõeldud treenerite grupeerimiseks ning selleks, et keegi ei saaks redigeerida teise klubi sportlasi. Kui klubi on loodud ja sinna alla treenerid lisatud, saavad nemad hakata deklareerima enda klubi sportlasi ning paatkondi.

Sportlase lisamisel peab määrama tema soo, sünniaasta, ees- ja perekonnanime ning kuna aerutamises on üheseid paadiklasse kahte tüüpi: kanuu ja süst, siis tuleb ka määrata kas ta sõidab ühte või teist või mõlemat. Mõlema valik on oluline, sest mõni noorem sportlane tahab ühel võistlusel sõita kanuud, kuid järgmisel valib ta süsta. Sportlase nimi peab olema unikaalne väärtus. Paatkondasid saab olema nelja tüüpi: K-2, C-2, C-4 ja K-4. Üldjuhul sportlased sõidavad hooajalõikes kõikidel võistlustel samades meeskondades aga juhul, kui soovitakse meeskonnaliikmeid vahetada siis saab luua paatkondi, kus üks sportlane paikneb mitmes meeskonnas korraga. Võistluse esitades peab süsteem kontrollima, et ega see sportlane juba ei sõida seda distantsi kellegi teisega koos.

2.4.3 Võistluse loomine

Iga klubi treener saab luua võistlusi, kus süsteem määrab automaatselt võistluse korraldajaks treeneri küljes oleva klubi. Lisaks peab korraldaja määrama võistluse ja peakohtuniku nime, algus- ja lõppkuupäeva ning asukoha. Peakohtuniku ning asukoha nimi on sõne tüüpi väljad, sest kohtunikuks võib olla kes iganes ning võistluse asukoht ei pruugi olla linnas sees. Võistlust ning sellega seotud andmeid saab redigeerida ainult võistluse korraldaja klubiga seotud treenerid kuni võistluse lõpuni. Pärast seda saab ainult võistluse tulemusi vaadata.

Järgmisena tuleb korraldajal luua distantsid, mis võistlusel olema hakkavad. Distanti loomisel tuleb rangelt ära määrata millisel päeval distantsi sõidetakse, sõidu algus kellaeg, distantsi pikkus, kas tegemist on karikavõistlusel arvestatava distantsiga ning

milline on vanusegruppide vaheline ajavahe. Sprindi distantside puhul peab määrama ka mitu paati igas sõidus on: kas kuus või üheksa. Lisaks tuleb eraldi ära määrata kõik vanusegrupid iga paadiklassi kohta, mis sellel distantsil osalema hakkavad, sest need ei pruugi kattuda teiste võistlusel toimuvate distantsidega.

Võistluse nimekirja ning üksikasjalik vaade peavad kasutajale kuvama selle põhiandmeid, milleks on võistluse nimi, korraldaja, asukoht, toimumise kuupäevad ning päevi jäänud võistluseni. Detailvaade peab ka kuvama peakohtuniku nime. Lisaks on detailvaates kõikide klubide esitatud sportlaste ja paatkondade arvud koos vastava klubi nimega, mis annab mugava ülevaate nii korraldaja kui ka osalevale klubile.

Võistluse loomist saab sisuliselt teostada kolmes osas. Esimene osa on võistluse enda kirje loomine, teine osa on distantside loomine ning kolmas on sõitude genereerimine. Selline järjekord nõuab rangelt eelmise käigu teostust, kus näiteks distantse ei ole võimalik lisada, kui võistluse kirje pole loodud. Selline lähenemine on oluliselt lihtsam autorile, kus ta ei pea looma keerulist võistluse lisamise vaadet koos distantside ja vanusegruppide valikutega ning mugavam ka treenerile, kes tuleb võistlejaid üles andma valides distantside kaupa milliseid sportlasi sinna üles anda.

2.4.4 Sportlaste esitamine

Sportlasi peab olema treenerile mugav üles anda võistlusele, kus kasutaja lihtsalt valib vastava distantsi ja vanusegrupi juurest õiged sportlased või paatkonnad. Süsteem peab selleks tema jaoks klubi nimekirjast ära grupeerima kõik esitatavad sportlased, kes pole juba selle võistluse nimekirjas. Selline viis on küll kasutajale lihtne ja kiire, kuid ei ole väga paindlik. Teine kord näiteks soovib treener panna sportlase vanemasse vanusegruppi sõitma või üldse esitada enda klubi alt mõne teise klubi sportlane. Selleks tuleb luua teine võimalus, kus kasutaja saab ükshaaval valida sportlasi ja vanusegruppe.

Kui kõik sportlased on esitatud siis vahetult enne võistlust vajutab korraldaja nuppu ning rakendus automaatselt loosib sõidud. Süsteem ei tee genereerimist ning stardiprotokollide laiali saatmist automaatselt, sest alati on vaja loositud sõidud enne avalikustamist korraldaja pilguga üle vaadata. Peamine, mis võib muutuda on sõitude kellaaeg ning intervall, mis suuresti sõltub sportlaste arvust ning see on kõige selgem vahetult enne võistlust. Tihti peale avaldub õhtu enne starti mõne sportlase haigus või

siis teise tervenemine ning siis on vaja muudatused sisse viia. Muudatusi saavad teha kõik klubid enda sportlaste kohta kuni võistluse päeva eelneva õhtu kella kaheksani ning peale seda saab seda teha ainult korraldaja klubi.

2.4.5 Sõitude loosimine

Sõitude loosimisel peab süsteem arvestama paadiklasse, vanusegruppe, distantsi tüüpi ning sportlaste arvu. Genereeritud tulemus peab tulema iga distantsi tüübi järgi erinev: intervall tüüpi sõidud peavad olema grupeeritud vanusegruppide ning paadiklasside järgi ning järjestatud stardi algusaja järgi kasvavalt. Tavaliselt on intervalliks üks minut. Ühisstartide puhul on grupeeritud ainult vanusegruppi ning paadiklassi järgi ning “sprint” tüüpi sõidud on neist kõige keerulisemad.

Rajaaerutamise distantsid, milleks on 200m, 500m ja 1000m ehk “sprint” distantside puhul tuleb loosimisel arvesse võtta sportlaste ning radade arvu. Siiani oli kombeks iga võistluse korraldajal ise otsustada, milliseid valemiteid kasutada sõitude loosimisel. Aluseks on alati võetud ICF-i reegleid, kuid mida on kohandatud vastavalt sportlaste arvule ning korraldaja vajadustele. Kuna autori loodud süsteem ei saa olla nii paindlik, siis tuleb luua konkreetsed raamid, mida iga võistluse puhul kasutama hakatakse.

Sellega seoses võttis autor ühendust Eesti Aerutamisföderatsiooni peakohtuniku Kristo Sepp’ga, kes on ka rahvusvahelise aerutamisföderatsiooni (ICF) litsentseeritud rajaaerutamise kohtunik. Ühise arutelu käigus sai koostatud valemileht nii üheksa (Tabel 3) rajaga sõitude kui ka kuue (Tabel 4) rajaga sõitude kohta. Valemite koostamisel sai arvestatud üldist Eesti aerutamismaastikul harrastatavat tavapära, et finaali jõudmiseks tuleks sportlasel läbida võimalikult vähe sõite. Seda põhjusel, et pikkasid distantse ei jõua väikesed mitu korda sõita ja et võistluse päev ei läheks ülearu pikaks. Lisaks peab mainima et osavõtjate arv on just suurim nooremate seas, milleks on C- ja D-klass.

COMPETITORS

HEATS

SEMI-FINALS

FINALS

1 / 9	-	-	A: 1 X 9
10 / 18	-	2 x 9 1 / 3 to Final 3 next BT to Final	A: 1 X 9
19 / 27	-	3 X 9 1/3 to Final A 4/6 to Final B Rest out	A: 1 X 9
28 / 36	4 x 9 1 / 6 to SF + 3next BT	3 X 9 1/3 to Final A 4/6 to Final B	A: 1 X 9

Tabel 3. Kohandatud üheksa raja valemite tabel

COMPETITORS	HEATS	SEMI-FINALS	FINALS
1 / 6	-	-	A: 1 X 6
7 / 12	2 x 6 1 / 2 to Final A 3 / 5 to SF	1 X 6 1 / 2 to Final A	A: 1 X 6
13 / 18	3 X 6 1 / 3 to SF + 3next BT	2 X 6 1 / 3 to Final A 4 / 5 to Final B	A: 1 X 6 B: 1 X 6
19 / 24	4 X 6 1 / 4 to SF + 2next BT Rest out	3 X 6 1 / 2 to Final A 3 / 4 to Final B Rest out	A: 1 X 6 B: 1 X 6
25 / 30	5 X 6 1 / 3 to SF + 3next BT Rest out	3 X 6 1 / 2 to Final A 3 / 4 to Final B Rest out	A: 1 X 6 B: 1 X 6

Tabel 4. Kohandatud kuue raja valemite tabel

Põhjus, miks üheksa radadega sõitudel on ainult A finaaliid ning kuue radadega sõitudel ka B finaaliid, on selle pärast, et karikapunkte jagatakse esimesele üheksale sportlasele. Ühe poolfinaali puhul saavad siis järgmised parimad sportlased, kes finaali ei pääsenud, 3 viimast punkti.

Võistlejate loosimine esimestesse sõitutesse peab käima sportlase taseme järgi. Sprindi distantside puhul lähevad tugevamad sportlased keskmistele radadele ning

intervallsõitude puhul viimasteks startijateks. Edasiste sõitude positsioon radadel sõltub sportlase eelmistest tulemustest.

2.4.6 Protokollide printimine ja saatmine

Kui kõik sõidud on loositud, siis saab kasutaja hakata nende ajakava sättima. Kuna võistlust luuest valib kasutaja konkreetse sõidu toimumise päeva, selle algus kellaaja ning sõitude vahelise intervalli, siis võib juhtuda, et mõnda sõitu ei tule osalejate vähesuse tõttu täis ning kasutaja peab ajakava veidike ümber tegema. Selleks tuleb süsteemi luua mugav võimalus, kuidas käsitsi sõitude kellaegasid muuta.

Sõitude mugavaks haldamiseks on vajalik, et rakendus kuvab kasutajale kohe keskkonnas kõik vajaliku informatsiooni iga sõidu kohta, kuid endiselt võistluse korraldamise ajal on vaja andmed andmebaasist kätte saada ning paber kandja kujul need kõigile nähtavale stendile üles panna. Seega peab süsteem võimaldama eksportida andmed *xlsx* [4] formaati, et seda saaks Microsoft Exceliga lugeda, vajadusel töödelda ning printida paber kandjale.

Protokolle, mida eksporditakse peab olema kahte tüüpi: finiši- ja stardiprotokollid ning need saavad saadavaks kohe peale sõitude genereerimist. Neid saavad alla tõmmata kõik, kellel on ligipääs sellele süsteemile igal ajal, ka peale võistlusi.

Kui sõidud on loositud ja ajakava paigas, peab korraldaja kõikidele klubidele välja saatma esmase stardiprotokollid, mis on oluline järgmisel päeval osalevatele sportlastele ning treeneritele. Selleks tuleb luua kasutajale nupp, mille vajutamisel süsteem saadab välja emailid. Stardiprotokollid küll ei pruugi olla lõplik, sest võistluspäeva hommikul võib osalevate sportlaste puhul sisse tulla muudatusi.

2.4.7 Tulemuste sisestamine

Võistlustulemusi saab jällegi sisestada ainult võistluse korraldaja ja võistluse toimumise ajal, kuid mitte hiljem. Loositud sõitudele aegade sisestamisel peab süsteem jälgima, kas kõik selle tüübi sõidud, näiteks poolfinaalid, on sõidetud ning ajad sisestatud ja sellest tulenevalt loosima automaatselt järgmised sõidud, milleks antud juhul on finaalid A ja B. ICF-i valemite lehel [2], lehekülgedel 63-69 on lisaks veel kirjeldatud täpselt millistest poolfinaalidest ja eelsõitudest millistele radadele sportlased loositakse ning

sega järgib autori süsteem väga täpselt välja arvatud kuueste radade puhul, sest ametlikult sellist sõitu pole olemas.

Võistlustulemusi peab saama sisestada formaadis 00:00:000, milleks minutid, sekundid ja sajandikud. Kui tegemist on finaali, intervall või ühisstardi tüüpi sõiduga ning korraldaja on märkinud, et seda distantsi arvestatakse karikavõistluse sarjas, siis rakendus peab automaatselt parematele välja jagama ka karikapunktid. Karikavõistluse punktide väljajagamisel kasutab autor EAF-i poolt koostatud ja ametlikult kinnitatud reegleid [3].

2.5 Vahendid

Antud süsteemi puhul on oluline selle kättesaadavus kõigile kasutajatele olenemata kasutatavast riistvarast või tema asukohast. Seega peab rakendusele ligipääs oleme üle veebi ning see välistab rakendustarkvara tegemise.

Kuna süsteem saab olema tsentraalne, siis kasutab autor selle loomiseks klassikalist klient-server arhitektuuri [5]. Veebiserveri loomiseks valis ta PHP programmeerimiskeele, kuna sellega on autoril kõige rohkem kogemusi ning raamistik, mida antud süsteemi loomisel kasutatakse, on samuti kirjutatud selle keelega.

Codelab Solutions OÜ poolt loodud raamistiku kasutamisega on autoril olnud tööalaseid kogemusi juba antud töö loomise hetkeks üle aasta ning pärast kasutamise õiguste saamist, langes otsus selle kasuks kiirelt. Raamistik nimega *labengine* põhineb klassikalisel MVC arhitektuuril [6].

Antud raamistik on ideaalne tagaruumi tüüpi veebirakenduste loomiseks, sest üldjuhul on nende kasutamisele õigused ainult valitud kasutajatel ning seega ei ole unikaalsete välimuste ja loogikaga kasutajaliidesed ja kasutajakogemused väga olulised. Sellest tulenevalt on raamistikus väga täpselt defineeritud kolm põhilist tüüpi vaadet: nimekirja-, detaili- ning redigeerimis ja lisamis vaated. Juhul, kui autor oleks valinud mõne muu PHP raamistiku, näiteks *Symphony* [7], oleks ta nullist pidanud need samad vaated defineerima. Sama kehtib ka sisse logimis vaadete ja logifailide pidamise kohta – loogika on alati sama ning CSS-i kaudu saab lihtsalt välimust muuta omanäolisemaks.

Kasutajaliidese kujunduse loomiseks on kasutatud populaarset CSS-i teeki nimega Bootstrap [8]. Arendamisel kasutab autor PhpStormi [9]. Versioonikontrolli tarkvaraks valis autor git-i [10], veebipõhiseks koodi hoidlaks nimega GitHub-i [11] ning veebiserver jookseb Apache HTTP peal [12].

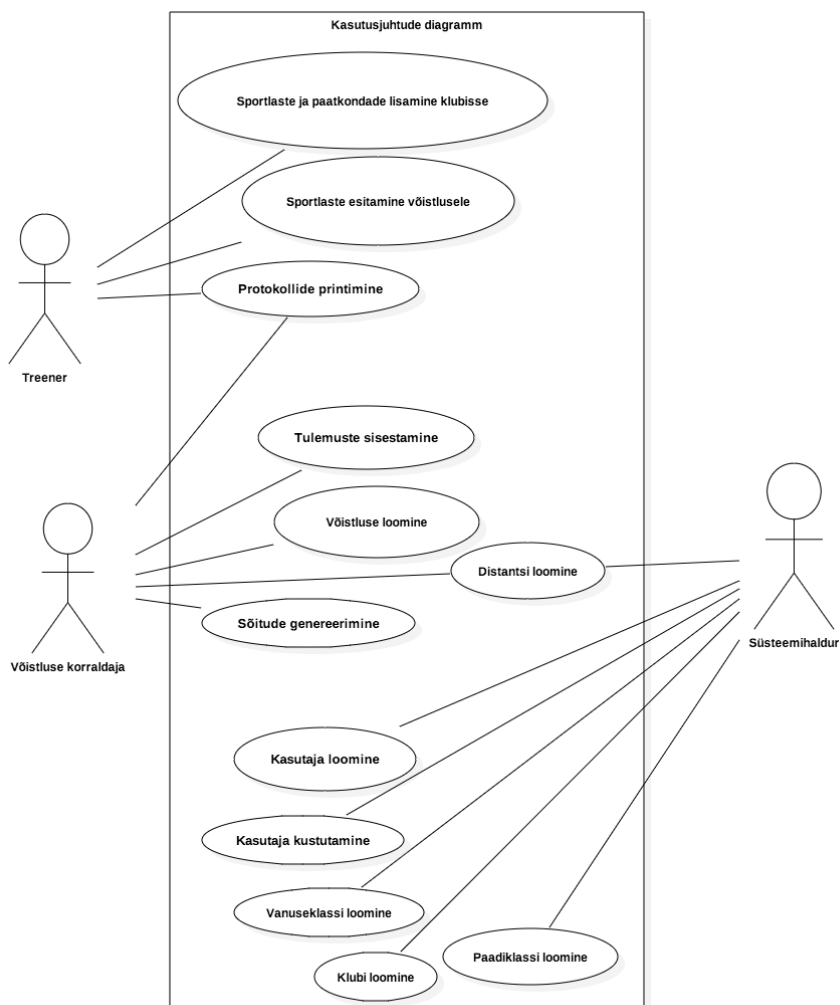
Raamistikul on vaikumise kasutatud Javascripti teeki nimega jQuery, millega on loodud kõiksugused hüpikakende kuvamised, ajax päringud, ning muud kliendi poolset andmete manipulatsiooni. jQuery kasutamise põhjuseks on lihtsustada JavaScripti kirjutamist, kuid antud süsteemi puhul sportlaste ja paatkondade lisamise vaate jaoks valis kasutaja React js [13] nimelise JavaScripti teegi. Põhjuseks on see, et keerukamate vaadete puhul läheb kasutajaliidese seisundi haldamine jQueryga keeruliseks ning antud vaade just seda on. React js-iga on väga lihtne kapseldada erinevaid komponente ning ka tuvastada selle komponendi olekut ükskõik millises ajahetkes.

3 Analüüs ja arendus

Käesolev peatükk kirjeldab loodavat süsteemi tehnilisest vaatepunktist: tuuakse välja süsteemi kasutusjuhtude ja olemi-suhte diagramm ning ülejäänud osa kirjeldatakse reaalse süsteemi loomist.

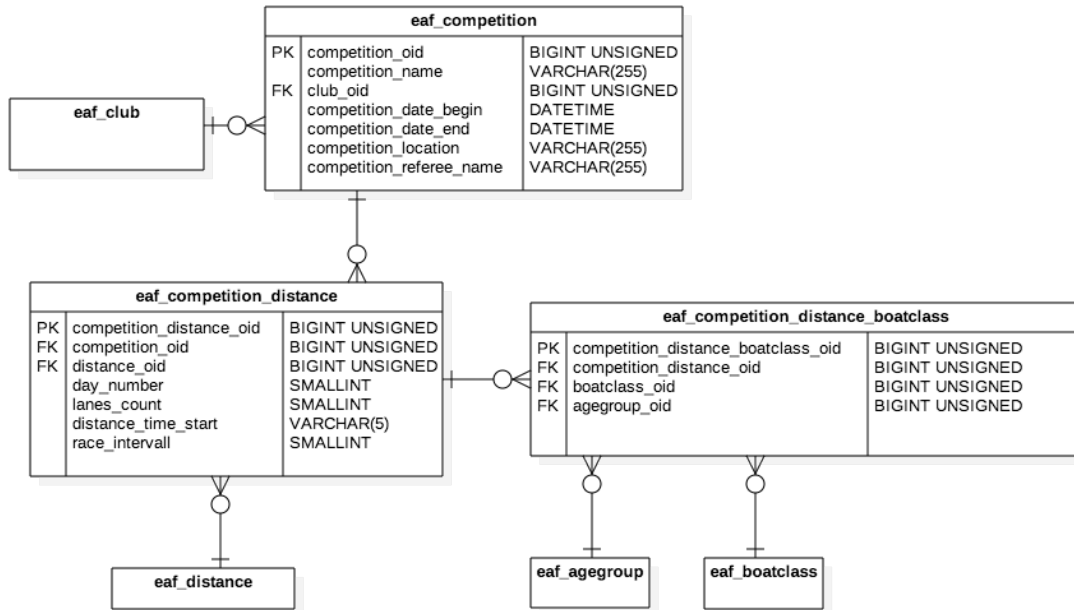
3.1 Kasutusjuhtude diagramm

Järgneval joonisel on kirjeldatud kasutusjuhtude diagramm. Süsteemi seisukohast on reaalselt kahte tüüpi kasutajaid: treener ja süsteemihaldur, kuid võistluse loomisel muutub ühe klubiga seotud kasutajad võistluse korraldajaks, kus teistel treeneritel võistluse administreerimisel õigusi pole (Joonis 1).

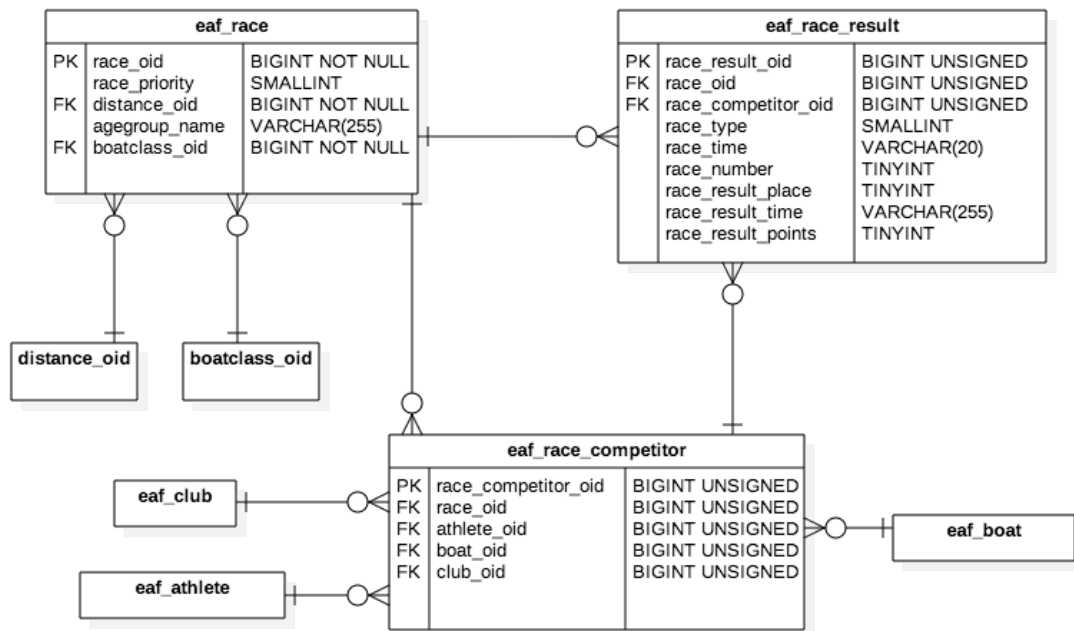


Joonis 1. Kasutusjuhtude diagramm

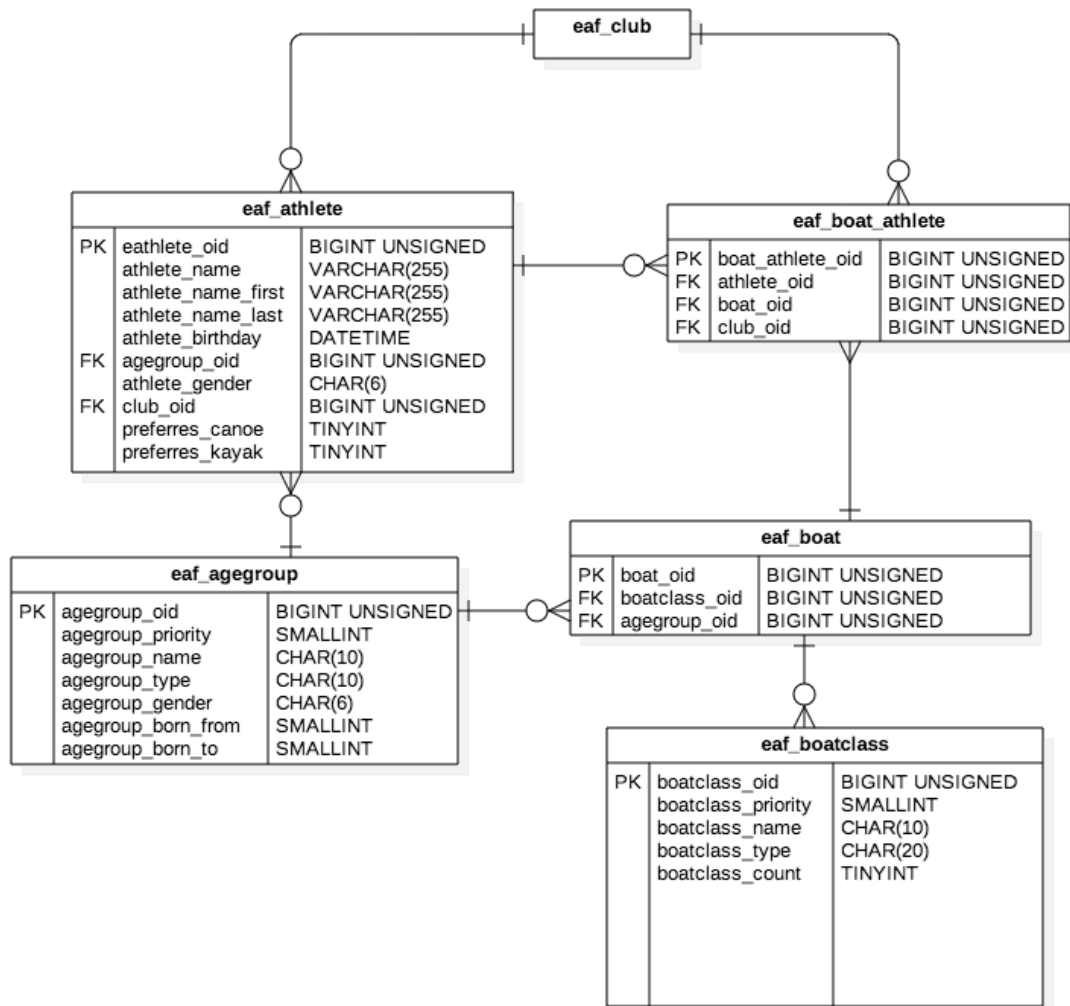
3.2 Olemi-suhte diagrammid



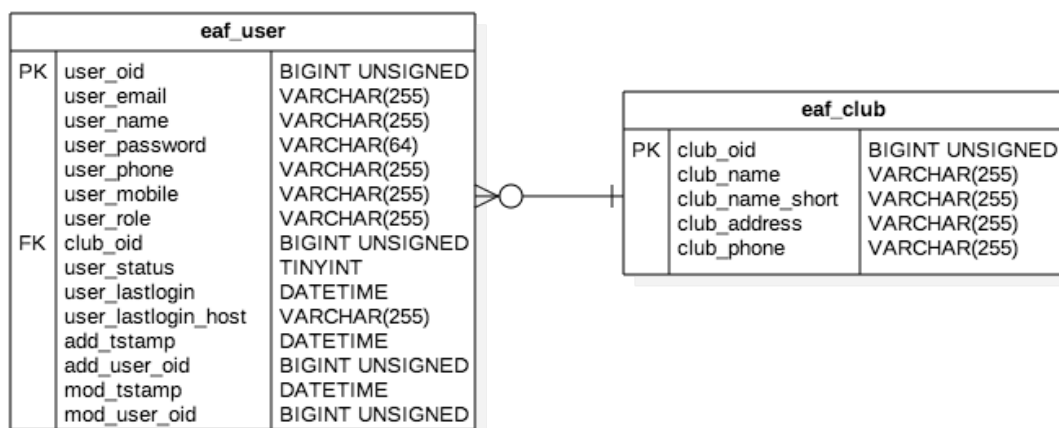
Joonis 2. Võistluse olemi-suhte diagramm



Joonis 3. Sõidu olemi-suhte diagramm



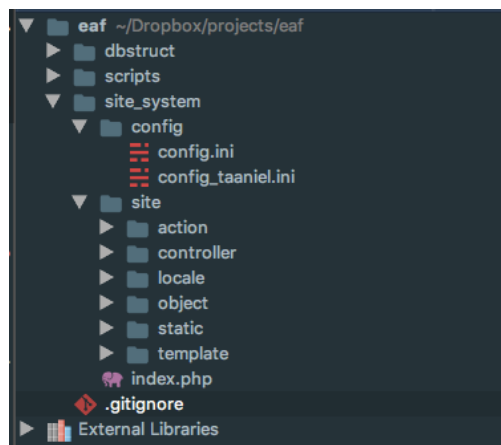
Joonis 4. Sportlase ja võistleja olemi-suhte diagramm



Joonis 5. Kasutaja olemi-suhte diagramm?

3.3 Failistruktuur

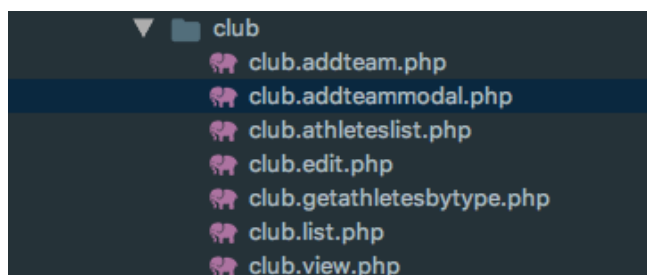
Käesoleva töö autor alustas projekti loomist andmebaasi, andmebaasitabelite, failistruktuuri ja andmeobjektide defineerimisest. Et eristada projekti klasse ja vaateid raamistiku omadest siis defineeriti need nimega “eaf”, mis on akronüüm Eesti Aerutamisföderatsioonist. Antud raamistiku kasutamine nõuab ranget failistruktuuri jälgimist (Joonis 2). Kõik andmebaasi tabelite loomise ja muutmisega seotud päringud käivad “dbstruct” nimelisse kausta. Erinevad skriptid, mille kutsub välja cron [14] või mis käivitatakse manuaalselt näiteks andmete muutmiseks andmebaasis, lisatakse kausta nimega “scripts”. Kogu rakenduse loogika on paigutatud kausta nimega “site_system”, kus asuvad erinevad konfiguratsiooni failid, indeksfail ning MVC arhitektuuril põhinev süsteemi ülesehitus.



Joonis 6. Labenginist tulenev failistruktuur näidatuna PhpStormis

3.4 Operatsioonid

Kaust nimega “action” koosneb vaadete klassidest, mis on omakorda jaotatud tüüpideks. Näiteks “club” kausta lähevad kõik klubi kuvamisega seotud failid: nimekirja ja üksikasjalik vaade ning temaga seotud sportlaste pärimise, lisamisega, kustutamise ja redigeerimisega seotud vaated (Joonis 7).



Joonis 7. Klubi vaadetega seotud klassid

3.5 Kontrollerid

Kontrollerid jaotatakse sama moodi tüübi järgi failidesse ning igas kontrolleris kirjeldatakse päringud, mida see konkreetne kontroller kuulama hakkab. Näiteks kui soovitakse saada vaadet kõikidest võistlustest siis URL peaks välja nägema selline: /competition/list (Joonis 8).

```
$a=$COMPETITION->addAction('list');  
$a->setInclude('site/action/competition/competition.list.php');  
$a->setClass('COMPETITION_LIST');
```

Joonis 8. Võistluste nimekirja vaate kontroller

3.6 Tõlked

Kaustas *locale* hoitakse erinevate keelte tõlkefaile. Kuna hetkel on plaanis süsteemi kasutada ainult Eestis ning erinevad treenerid kinnitasid autorile, et ka vene rahvusest inimesed saavad eesti keelse süsteemi kasutamise hästi hakkama, siis esialgu jääb keeleks ainult eesti keel. Juhul, kui kunagi on vajadus üle minna mitmekeelsele süsteemile, siis tuleb ainult teha vastava keele laiendiga fail ning sõna märgendi konstandile lisada vastav tõlge (Joonis 9).

EAF.COMPETITION.Name

Nimi

Joonis 9. Eesti keelse tõlke märgend

3.7 Objektid

Raamistikuga tuleb kaasa mugav andmeobjektide loogika, kus andmebaasis unikaalsete identifikaatoritega read vastendatakse temale vastava klassiga määratud mallile, tänu millele selle konkreetse objektiga kaasnevad nii andmed kui ka meetodid. Enamikest meetoditest, mida autor oma lõputöös andmeobjektidesse kirjutab, on juhtumisi staatilised: näiteks sportlaste nimekirja pärimise meetod peab olema välja kutsutav igalt poolt ning ei ole objekti spetsiifiline. Seevastu kasutaja rolli kontrollimine on seotud konkreetse objektiga ning selle pärast ei saa see olla staatiline. Meetodile antakse kaasa sõne või nimekiri sõnedest mida võrreldakse andmebaasis oleva kasutaja küljes oleva rolliga (Joonis 10).

```
public function checkRole( $role )
{
    if (!$this->isLoggedIn()) return FALSE;
    $roleList=str_array($role);
    foreach ($roleList as $role)
    {
        if ($role==$this->user_role) return true;
    }
    return FALSE;
}
```

Joonis 10. Kasutaja rolli kontrollimise meetod

3.8 Mallid ja staatilised failid

Ülejäänud kaustad on mallide ja staatiliste failide jaoks. Malle on antud projekti puhul kaks tükki: autentimise mall ning tavaline. Mallide ülesanne on kaasata HTML-i päisesse kõik vajalikud Javascript-i ja CSS-i laiendused ja defineerida ära üldine süsteemi välimus. Iga vaate puhul jääb sisse loginud kasutajale visuaalselt samaks rakenduse päis ja jalus (Joonis 11).



Joonis 11. Sisse loginud kasutaja põhivaade

4 Kasutuslood võistluse korraldaja vaatest

Antud peatükk kirjeldab piltide ja kirjelduste näol reaalset süsteemi kasutamist läbi kasutajaliidese võistluse korraldaja vaatepunktist. Kuna kõik ekraanitõmmised on tehtud realselt toimivast süsteemist, siis autor on isikute varjamiseks piltidelt eemaldanud nende perekonnanimed.

Kasutuslood hakkavad pihta alates seisundist, kui kõikidele treeneritele on kasutajad, distantid, paadiklassid ja vanusegrupid loodud ning klubid defineeritud. Järgmisena hakkab kasutaja lisama enda klubi alla sportlasi ja paatkondi ning seejärel suundub võistluse loomise vaatesse, et korraldada uut võistlust.

4.1 Sisse logimine

Kõigepealt tuleb treeneril süsteemi sisse logida, kasutades süsteemi administraatori poolt loodud kasutajanime, mis on samas ka e-maili aadress ning parooli (Joonis 12).



Aerutajate infosüsteem

E-mailiaadress:

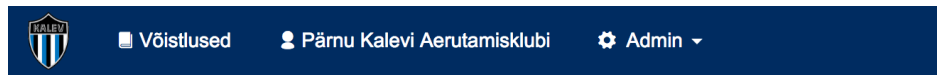
Salasõna:

Logi sisse

Joonis 12. Sisse logimise vaade

Kui kasutajanimi ja parool on sisestatud ning õiged, saab treener süsteemi sisse.

Kuna sisse loginud kasutaja, antud juhul autor ise, kes on ka süsteemi halduri õigustega ning kuulub Pärnu Aerutamisklubi liikmete alla, siis temale kuvatakse menüü ribal Pärnu klubi nime (Joonis 13).



- Uute kasutajate lisamiseks valige ülevalt "Admin" → "Kasutajad".
- Klubi sportlaste/paatkondade lisamiseks valige ülevalt enda klubi.
- Võistlustele sportlaste ülesandmiseks valige ülevalt "Võistlused" → vastav võistlus.

Joonis 13. Sisse loginud Pärnu Aerutamisklubi liikme vaade

4.2 Sportlaste sisestamine andmebaasi

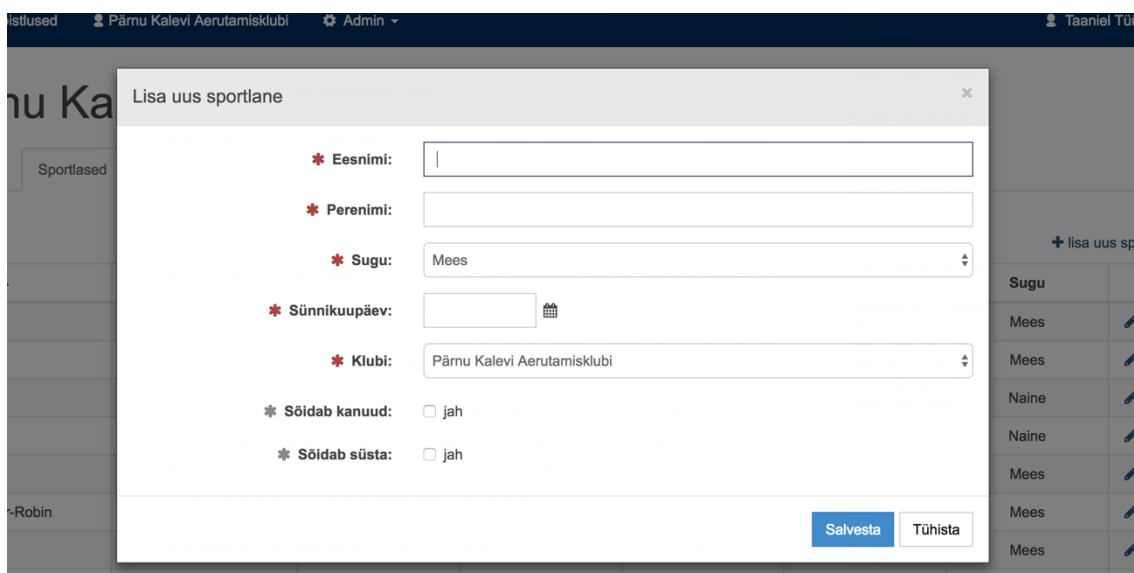
Esimest korda sisse logides tuleks tal esmalt navigeerida sportlaste lisamise vaatesse, kuhu saab läbi menüüribal oleva klubi nime ning kus tal tuleks hakata sisestama sportlasi ning paatkondi (Joonis 14).

Eesnimi	Perenimi	Vanusegrupp	Sünnikuupäev	Sõidab süsta	Sõidab kanuud	Sugu	
Alex	<input type="text"/>	PC	30.09.2004	✓		Mees	/ /
Andrei	<input type="text"/>	PD	12.10.2007	✓		Mees	/ /
Anette	<input type="text"/>	N	24.03.1998	✓	✓	Naine	/ /

Joonis 14. Sportlaste sisestamise vaade Pärnu Aerutamisklubi näitel

Uut sportlast sisestades avane tal vaade hüpickmenüüst, kus punase tärniga veerund on kohustuslikud ning halli tärniga väljadel on spetsiifilisem loogika. Antud juhul peab

vähemalt üks neist valitud olema. Vormi pole võimalik salvestada, kui kohustuslikud väljad pole täidetud. Klubi väli on autorile nähtav ainult selle pärast, et ta on süsteemi administraator - treeneri õigustega kasutajatel seda välja pole ning sportlase külge läheb sama klubi, mis on treenerilgi (Joonis 15).



The screenshot shows a web application interface for adding a new athlete. The form is titled "Lisa uus sportlane" and contains the following fields and options:

- Eesnimi:** Text input field.
- Prenimi:** Text input field.
- Sugu:** Dropdown menu with "Mees" selected.
- Sünnikuupäev:** Date picker field.
- Klubi:** Dropdown menu with "Pärnu Kalevi Aerutamisklubi" selected.
- Sõidab kanuud:** Radio button with "jah" selected.
- Sõidab süsta:** Radio button with "jah" selected.

Buttons for "Salvesta" (Save) and "Tühista" (Cancel) are located at the bottom right of the form.

Joonis 15. Sportlase lisamise hüplikaken

Paatkondade lisamise puhul tuleb kõigepealt ära määrata, millist paadiklassi lisada tahetakse ning siis rakendus genereerib kasutaja jaoks vastavalt sportlaste arvule paadis väljad (Joonis 16).



The screenshot shows a web application interface for selecting a boat class. The dropdown menu is titled "Valige paadiklass" and contains the following options:

- ✓ --- vali siit ---
- C-2
- K-2
- C-4
- K-4

Joonis 16. Paatkonna lisamise esimene hüplikaken

Joonis 17. Paatkonna lisamise teine hüplikaken

Kui paatkonna kombineerimise vaade on avanenud (Joonis 17), siis iga tähe vajutamisel peale peab rakendus kasutajale kuvama kõikide võimalike sportlaste nimesid, mis nendest tähekombinatsioonidest on võimalik tuletada. Lisainformatsioonina kuvatakse talle ka sportlase vanuseklass ja klubi et vähendada eksimise tõenäosust.

4.3 Võistluste loomine

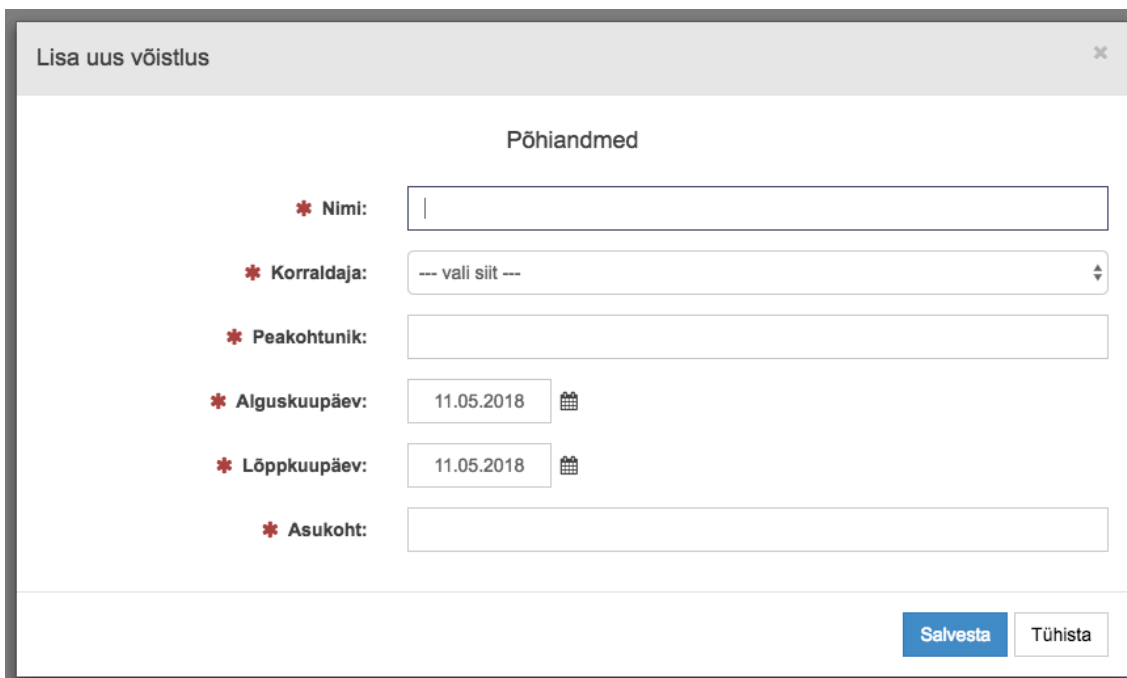
Kui kõik sportlased ja paatkonnad on süsteemi deklareeritud, siis järgmine kasutusjuht näeb ette võistluse loomist. Selleks navigeerib kasutaja menüüribal oleva *Võistlused* valiku kaudu võistluste nimekirja vaatesse, kus on näha toimunud, käesolevaid ning tulemas olevaid võistlusi (Joonis 18).

Nimi	Staatus	Korraldaja ▲	Alguskuupäev	Lõppkuupäev	Asukoht
EKV II ja III etapp	Lõppenud	Spordiklubi Narva Vikingid	19.05.2018	20.05.2018	Narva
EKV I etapp	Lõppenud	SAK Tartu	05.05.2018	05.05.2018	Tartu

Kokku 2 kirje(t)

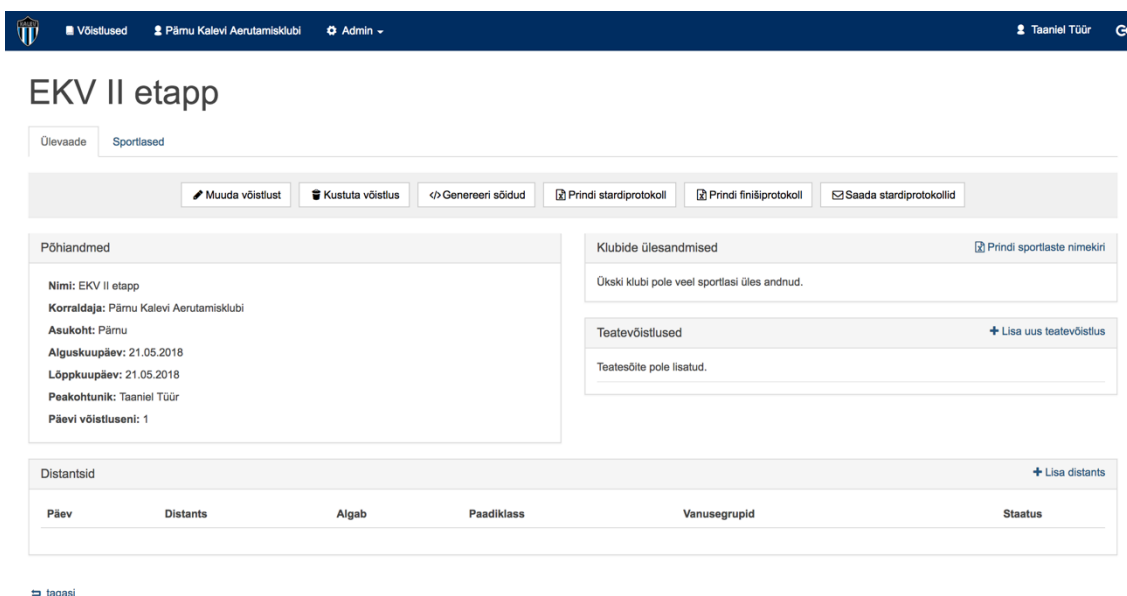
Joonis 18. Võistluste nimekirja vaade

Uue võistluse loomiseks valib kasutaja paremalt valiku *Lisa uus võistlus* ning tal avaneb vaade, kus kõik väljad on kohustuslikud (Joonis 19).



Joonis 19. Võistluse loomise vaade

Kui kasutajal on kõik väljad täidetud ning vajutatud *Salvesta*, siis ta suunatakse automaatselt võistluse vaatesse, kus tal järgmisena tuleks hakata lisama võistlusele distantse (Joonis 20).



Joonis 20. Võistluse vaade

Valides paremalt alt *Lisa distants*, avaneb kasutajal hüplikaken, kus tal tuleb määrata sõidu toimumise päev, distantsi algus kellaaeg, distantsi pikkus, intervall sõitude vahel minutites ning kas tegemist on karikavõistlusega. Sprindi distantside puhul tuleks tal ka valida paatide arvu sõidus. Lisaks peab korraldaja määrama vanusegrupid iga paadiklassi kohta ning vajutades *Salvesta* on esimene distants loodud ning kasutajad saavad sinna hakata üles andma (Joonis 21).

The screenshot shows a web application interface for adding a new distance. The form is titled "Lisa uus distants" and contains the following fields and options:

- Sõidu toimumise päev:** A dropdown menu with the text "-- vali siit --".
- Sõidu kellaaeg:** A text input field containing "12:00".
- Distants:** A dropdown menu with the text "-- vali siit --".
- Karikavõistlus:** A checkbox labeled "jah" (yes).
- Paadiklassid - Vanusegrupid:** A section for selecting age groups for each boat class. Under the heading "C-1:", there is a list of checkboxes for classes M, N, PA, TA, PB, TB, PC, TC, PD, TD, and PF.

At the bottom right of the form, there are two buttons: "Salvesta" (Save) and "Tühista" (Cancel).

Joonis 21. Uue distantsi loomine võistlusel

4.4 Võistlusele registreerimine

Kui distantsid loodud, siis saavad kasutajad hakata tegema ülesandmisi. Näitena alustab autor olukorrast, kus kõik klubid, välja arvatud Pärnu Aerutamisklubi, on oma ülesandmised teinud ning just seda viimast hakkab ta nüüd teostama (Joonis 22).

Põhiandmed

Nimi: EKV II etapp
Korraldaja: Pärnu Kalevi Aerutamisklubi
Asukoht: Pärnu
Alguskuupäev: 21.05.2018
Lõppkuupäev: 21.05.2018
Peakohtunik: Taaniel Tüür
Päevi võistluseni: 1

Klubide ülesandmised

Printi sportlaste nimekiri

Klubi	Sportlased	Paatkonnad
Pirita Aerutamisklubi	35	1
Spordiklubi Narva Vikingid	36	7
Tartu Ülikooli Akadeemiline Spordiklubi	16	7
SAK Tartu	28	6
Emajõe Aeruklubi	27	2
Spordiselts Põhjakotkas Spordikool	73	24
Kokku	215	47

Teatevõistlused

+ Lisa uus teatevõistlus

Teatesõite pole lisatud.

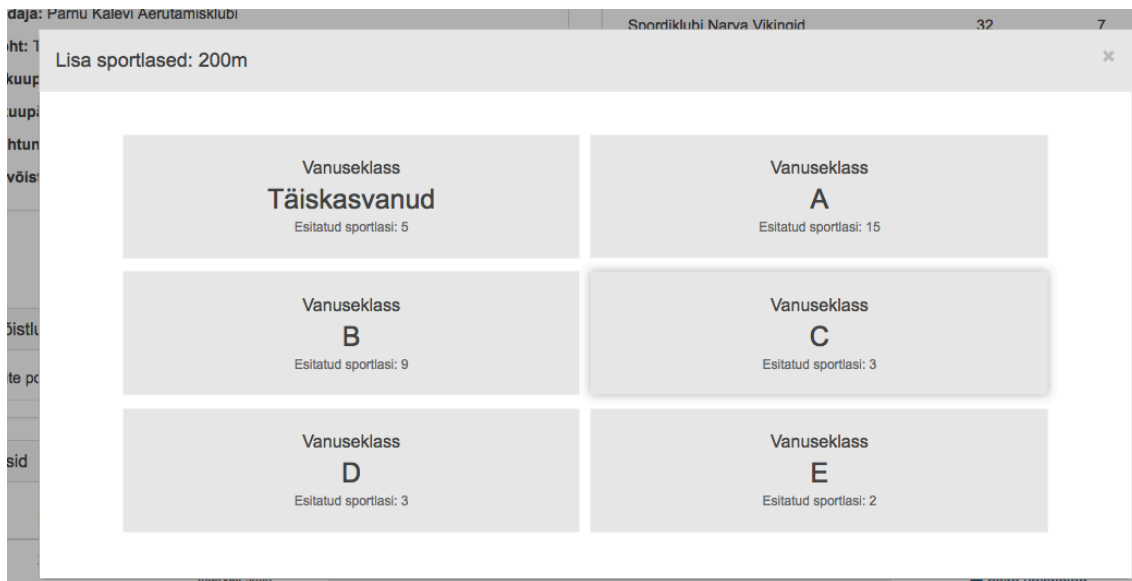
Distantsid

+ Lisa distants

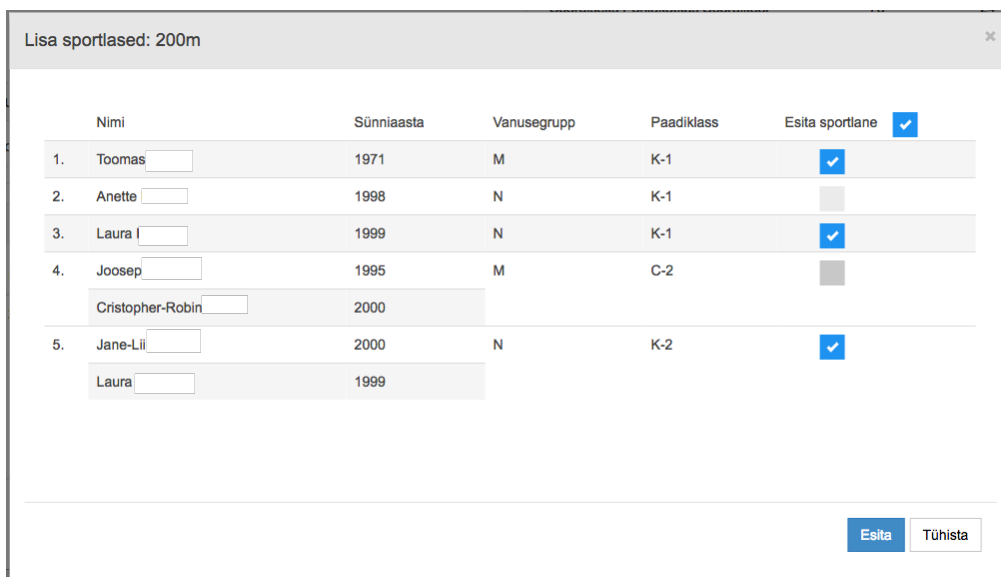
Päev	Distants	Algab	Paadiklass	Vanusegrupid	Staat
1.	200m (Sprint)	09:00 intervall: 5min paastide arv: 6	C-1	M PA PB PD PE PC	Ülesandmine tegemata + esita võistlejad
			K-1	M N PA TA PB TB PD TD PE TE PC TC	
			C-2	M PA PB PD PE PC	
			K-2	M N PA TA PB TB PD TD PE TE PC TC	

Joonis 22. Võistluse vaade, kui distantsid loodud ja klubide ülesandmised tehtud

Et hakata klubi sportlasi lisama, vajutab treener distantsi küljes olevat *esita võistlejad* nuppu ning avaneb hüpikaken, kus tal tuleb kõigepealt valida vanusegrupp (Joonis 23) ning alles siis tekib nähtavale nimekiri kõigist selles vanuses olevates sportlastest (Joonis 24). Vanusegruppide nimistu tuleneb võistluse korraldaja loodud distantsi poolsest vanusegruppide valikust.



Joonis 23. Sportlaste lisamise vanusegruppide hüpikaken



Joonis 24. Täiskasvanud sportlaste ja paatkondade esitamine võistlusele

Kui sportlased on lisatud, siis saab võistluse korraldaja genereerida sõidud. Selleks valib ta võistluse vaatest ülevalt võistlusemenüüst *Genereeri sõidud*, mille tulemusena tekib sinna vaatesse uus vahekaart, mille pealkirjaks on genereeritud distantsti nimi. Antud juhul on selleks 200m (Joonis 25).

Sõidu nimi

PE K-1 09:00 Heat 1					1/3 to SF + 3next BT				
1	Aksel	<input type="text"/>	2008	Viljandi	<input type="button" value="x"/>				
2	Sergei	<input type="text"/>	2008	Põhjakotkas	<input type="button" value="x"/>				
3	Andrei	<input type="text"/>	2008	Narva	<input type="button" value="x"/>				
4	Roman	<input type="text"/>	2008	Pirita	<input type="button" value="x"/>				
5	Grigori	<input type="text"/>	2008	Põhjakotkas	<input type="button" value="x"/>				

M C-1 11:55 Heat 2					1/2 to Final A, 3/5 to SF				
1	Roman	<input type="text"/>	1985	Narva	<input type="button" value="x"/>				
2	Vitali	<input type="text"/>	1987	Narva	<input type="button" value="x"/>				
3	Mihkel	<input type="text"/>	1991	TÜASK	<input type="button" value="x"/>				
4	Deniss	<input type="text"/>	1999	Põhjakotkas	<input type="button" value="x"/>				
5	Andreas	<input type="text"/>	1995	TÜASK	<input type="button" value="x"/>				
6	Taimo	<input type="text"/>	1997	TÜASK	<input type="button" value="x"/>				

PE K-1 09:05 Heat 2					1/3 to SF + 3next BT				
1	Ilja	<input type="text"/>	2008	Pirita	<input type="button" value="x"/>				

M K-1 12:00 Heat 1					1/4 to SF + 2next BT Rest out				
--------------------	--	--	--	--	-------------------------------	--	--	--	--

Joonis 25. Loositud 200m sõidud süsteemi vaates

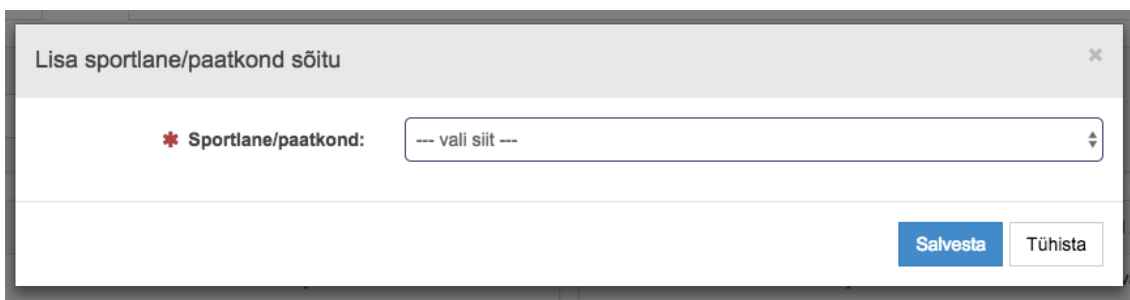
Kui korraldaja on veendunud, et kõik sõidud on paigas, siis valib ta võistluse vaatest ülevalt menüüribalt *Saada stardiprotokollid*, mille tulemusena saavad kõik võistlusel osalevate klubide treenerid emaili peale stardiprotokollid järgmisel päeval toimuvast võistlusest *xlsx* faili kujul (Tabel 5).

EKV II etapp						
21.05.2018						
21.05.2018						
Pärnu						
Korraldaja: Pärnu Kalevi Aerutamisklubi						
1	09:00	PE K-1	1/3 to SF + 3next BT			
		200	1	Aleksei	2009	Põhjakotkas
		Heat 1	2	Nikita	2010	Narva
			3	Leonid	2008	Põhjakotkas
			4	Grigori	2008	Põhjakotkas
			5	Roman	2008	Pirita
2	09:05	PE K-1	1/3 to SF + 3next BT			
		200	1	Maksim	2010	Narva
		Heat 2	2	Veiko	2009	Pärnu
			3	Denis	2010	Põhjakotkas
			4	Sergei	2008	Põhjakotkas
			5	Andrei	2008	Narva
3	09:10	PE K-1	1/3 to SF + 3next BT			
		200	1	Artemi	2008	Põhjakotkas
		Heat 3	2	Saveli	2008	Narva
			3	Rico	2009	SAK Tartu
			4	Ilja	2008	Pirita
			5	Fred	2008	Pärnu

Tabel 5. 200m PE vanuseklassi stardiprotokoll *xlsx* faili kujul

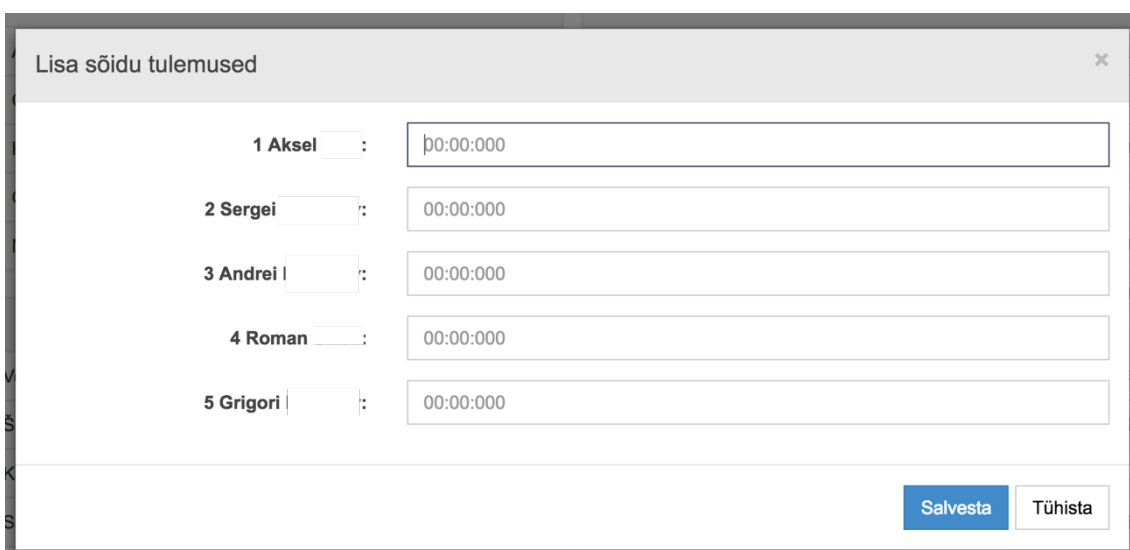
4.5 Tulemuste sisestamine ja loosimine

Järgmised tegevused toimuvad juba võistluse toimumise ajal, kus süsteemis on iga loositud sõidu kohal neli olulist tööriista: valides jooniselt 25 plussiga tähistatud uue võistleja lisamine konkreetsesse sõitu (Joonis 26), hammasratas on sportlaste aegade sisestamiseks (Joonis 27), pliiats on konkreetse sõidu kellaaja muutmiseks (Joonis 28) ning prügikast on terve sõidu kustutamiseks.



The screenshot shows a dialog box titled "Lisa sportlane/paatkond sõitu". It contains a label "Sportlane/paatkond:" followed by a dropdown menu with the text "-- vali siit --". At the bottom right, there are two buttons: "Salvesta" (Save) and "Tühista" (Cancel).

Joonis 26. Sportlase või paatkonna lisamine sõitu



The screenshot shows a dialog box titled "Lisa sõidu tulemused". It contains a list of five rows, each with a name and a time input field:

1 Aksel	:	<input type="text" value="00:00:00"/>
2 Sergei	:	<input type="text" value="00:00:00"/>
3 Andrei	:	<input type="text" value="00:00:00"/>
4 Roman	:	<input type="text" value="00:00:00"/>
5 Grigori	:	<input type="text" value="00:00:00"/>

At the bottom right, there are two buttons: "Salvesta" (Save) and "Tühista" (Cancel).

Joonis 27. Aegade sisestamise vaade

Joonis 28. Sõidu algusaja muutmise vaade

Kui tulemused on sisestatud, siis saab korraldaja välja printida finišiprotokoll. Järgmises näites on autor sisestanud kolme eelsõidu ajad (Tabel 6) ning kolmandat salvestades loosis süsteem eelsõitude tulemuste põhjal välja poolfinaalid (Tabel 7).

EKV II etapp							
21.05.2018							
21.05.2018							
Pärnu							
Korraldaja: Pärnu Kalevi Aerutamisklubi							
1	09:00	PE K-1	1/3 to SF + 3next BT				
		200	1	Roman	2008	Pirita	00:34:123
		Heat 1	2	Aleksei	2009	Põhjakotkas	00:54:123
			3	Leonid	2008	Põhjakotkas	01:00:123
			4	Nikita	2010	Narva	01:14:123
			5	Grigori	2008	Põhjakotkas	DNF
2	09:05	PE K-1	1/3 to SF + 3next BT				
		200	1	Andrei	2008	Narva	00:34:123
		Heat 2	2	Denis	2010	Põhjakotkas	00:54:220
			3	Veiko	2009	Pärnu	01:01:0
			4	Maksim	2010	Narva	01:10:123
			5	Sergei	2008	Põhjakotkas	01:34:232
3	09:10	PE K-1	1/3 to SF + 3next BT				
		200	1	Artemi	2008	Põhjakotkas	00:34:123
		Heat 3	2	Saveli	2008	Narva	00:34:123
			3	Rico	2009	SAK Tartu	00:34:123
			4	Ilja	2008	Pirita	01:34:242
			5	Fred	2008	Pärnu	01:34:332

Tabel 6. Finišiprotokoll PE vanuseklassi eelsõitude põhjal xlsx faili kujul

39	12:15	PE K-1	1/3 to Final A, 4/6 to Final B Rest out				
		200	1	Rico		2009	SAK Tartu
		SF 1	2	Denis		2010	Põhjakotkas
			3	Roman		2008	Pirita
			4	Artemi		2008	Põhjakotkas
			5	Leonid		2008	Põhjakotkas
			6	Nikita		2010	Narva
40	12:20	PE K-1	1/3 to Final A, 4/6 to Final B Rest out				
		200	1	Maksim		2010	Narva
		SF 2	2	Saveli		2008	Narva
			3	Andrei		2008	Narva
			4	Aleksei		2009	Põhjakotkas
			5	Veiko		2009	Pärnu
			6	Sergei		2008	Põhjakotkas

Tabel 7. 200m PE vanuseklassi loositud poolfinaalid xlsx faili kujul

4.6 Toimunud võistlused, tulemuste järele vaatamine

Kui võistlus on läbi, siis kogu senine salvestatud seis jääb andmebaasi alles ning võistluste nimekirja vaatest (Joonis 18) saab valida konkreetse võistluse, mis kasutajale kõige enam huvi pakub. Erinevus, mis on tulemas olevate ja toimunud võistlustel, on see, et võistluse redigeerimist toatavad nupud on peale võistlust kõigile nähtamatuks muudetud. Sellegi poolest saab kasutaja vaadata kasutajaliidese kaudu võistlustulemusi ning välja printida nii stardi- kui ka finišiprotokollid.

Kokkuvõte

Käesoleva projekti prototüüp sai kõigepealt avalikustatud ning testitud aasta aega selle lõputöö vormistamist mõningatel võistlustel: Pärnu KV, Eesti Meistrivõistlused täiskasvanutel, noortel ja juunioritel, Pirita KV, EKV I etapp Tartus ning EKV II ja III etapp Narvas. Kuna prototüüp olid edukas, siis Eesti Aerutamisföderatsioon koos kõikide liikmetega võtsid vastu otsuse käesoleval hooajal (2018) seda süsteemi kõikide võistlustel kasutada.

Antud projekt on leidnud palju positiivset tagasisidet, sest see rahuldab täpselt Eesti aerutamise seotud võistluse korraldamise ning treenerite vajadusi. Kuna meie aerutamise miljöö on võrreldes suuremate riikidega nii erinev, siis selle pärast ükski senine testitud rahvusvaheline süsteem meie vajadusi ei rahulda. Sellele väitele tugineb autor EAF-i poolt saadud impulsist pärast Leedus kasutatava süsteemi katsetamist, mille ostmine oleks olnud kulukas, kasutajaliides keeruline ning mis poleks vastanud EAF-i nõuetele.

Kuigi antud veebirakendus sai spetsiaalselt valmistatud aerutajatele, siis autor on arendamisel ka silmas pidanud võimalikkust seda süsteemi üle kanda sõudjatele ning mõningate muudatuste tulemusena saab välja lasta uue platvormi just nende vajadusi silmas pidades.

Järgmisena hakkab autor uurima süsteemi võimalikku integreerimist automaatse ajavõtusüsteemiga, mis kaotaks ära senise klassikalise stopper-meetodi ning vähendaks võistlusel vajaminevat inimressurssi veelgi.

Kasutatud kirjandus

- [1] „EAF,“ [Võrgumaterjal]. Available: <http://www.aerutaja.ee/aerutamisföderatsioon/#liikmed>.
- [2] International Canoeing Federation, [Võrgumaterjal]. Available: https://www.canoeicf.com/sites/default/files/icf_canoe_sprint_for_print_2017.pdf.
- [3] Eesti Aerutamisföderatsioon, „Karikavõistluste üldjuhend,“ 26 10 2016. [Võrgumaterjal]. Available: <http://www.aerutaja.ee/wp-content/uploads/2015/04/kvuldjuhend2017.doc>. [Kasutatud 14 4 2018].
- [4] „Wikipedia,“ [Võrgumaterjal]. Available: https://en.wikipedia.org/wiki/Office_Open_XML.
- [5] „Wikipedia,“ [Võrgumaterjal]. Available: https://en.wikipedia.org/wiki/Client%E2%80%93server_model.
- [6] „Wikipedia,“ [Võrgumaterjal]. Available: <https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller>.
- [7] „Symphon,“ [Võrgumaterjal]. Available: <https://symfony.com/at-a-glance>.
- [8] O. M. Introduction.. [Võrgumaterjal]. Available: <https://getbootstrap.com/docs/3.3/getting-started/>.
- [9] JetBrains SRO, „JetBrains,“ [Võrgumaterjal]. Available: <https://www.jetbrains.com/phpstorm/>.
- [10] „Getting started – About Version Control,“ [Võrgumaterjal]. Available: <https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control>.
- [11] „What is github?,“ [Võrgumaterjal]. Available: <https://guides.github.com/activities/hello-world/#what>.
- [12] „Apache,“ [Võrgumaterjal]. Available: http://httpd.apache.org/ABOUT_APACHE.html.
- [13] „React js,“ [Võrgumaterjal]. Available: <https://reactjs.org/tutorial/tutorial.html#what-is-react>.
- [14] „Cron,“ [Võrgumaterjal]. Available: <https://en.wikipedia.org/wiki/Cron>.