

TARTU ÜLIKOOL
MATEMAATIKA-INFORMAATIKATEADUSKOND
Arvutiteaduse instituut
Infotehnoloogia õppekava

Jaanus Sellin

**Robotika ja programmeerimise
õppematerjalide infosüsteem**

Bakalaureusetöö (6 EAP)

Juhendajad: MA Taavi Duvin
MSc Anne Villems

Tartu 2015

Robootika ja programmeerimise õppematerjalide infosüsteem

Lühikokkuvõte:

Õppematerjalide olemasolu ja kättesaadavus on väga oluline hariduse üldise kvaliteedi parandamiseks, õpetajate töö lihtsustamiseks ja õpilaste iseseisva arengu toetamiseks. Käesoleva bakalaureusetöö raames valmis robootika ja programmeerimise õppematerjalide haldamiseks mõeldud infosüsteem, kuhu on võimalik koondada kõik neid valdkondi puudutavad Eesti üldhariduses ja huviringides kasutatavad õppematerjalid. Töö teise osa moodustab infosüsteemile esitatud nõuete ülevaade ja süsteemi toimimist selgitav dokument. Infosüsteemi arhitektuur on kujundatud vastavalt tellija vajadustele, kusjuures esikohal on lihtsus ja kasutajasõbralikkus.

Võtmesõnad:

robootika, programmeerimine, infosüsteem, andmebaas, õppematerjalid, Nutilabor, üheleheküljeline rakendus, dunaamiline disain, AngularJS, Bootstrap, ASP.NET MVC

Information system for robotics and programming teaching resources

Abstract:

The existence and availability of educational exercises and teaching materials play a crucial role in the improvement of overall quality of education, simplification of teachers' daily work and students' individual development. The aim of this bachelor's thesis is to build an information system allowing comfortable management of infotechnology-related teaching materials. The second part of the thesis provides a written document that supports the system by explaining the requirements and functionality in detail. The architecture of this system is designed based on the requirements from the client that emphasize a friendly user experience.

Keywords:

robotics, programming, information system, database, educational exercises, SmartLabs, single page application, responsive design, AngularJS, Bootstrap, ASP.NET MVC

Sisukord

1. Sissejuhatus	4
2. Õppematerjalide infosüsteem – senised arengud ja vajadused	5
3. Nõuded infosüsteemile	7
4. Süsteemi arhitektuur.....	8
4.1 Üheleheküljeline rakendus	8
4.2 Kasutajate rollid.....	8
4.3 Autentimine ja autoriseerimine	8
4.4 Andmemudel	9
4.5 Rakenduse arendamiseks kasutatud tehnoloogiad.....	11
4.5.1 Serveripoolne süsteem	11
4.5.2 Kasutajapoolne liides	11
5. Süsteemi kasutuslood	13
6. Rakenduse analüüs ja testimine	19
7. Arutelu ja kokkuvõte.....	21
8. Kasutatud kirjandus.....	22
Lisad	23
I. Infosüsteemi lähtekood	23
II. Näiteid AngularJSi moodulite kasutamisest rakenduses	24
III. Litsents	27

1. Sissejuhatus

Infotehnoloogia ja tehnoloogilised seadmed ümbritsevad tänapäeval meid kõiki, kuid praegused noored on esimene põlvkond, kes on kiirelt arenevas infotehnoloogilises keskkonnas üles kasvanud. Kokkupuude infotehnoloogiaga leiab aset nii koduses keskkonnas kui ka koolis, kus see on põimitud haridussüsteemi - mitmed koolid on läbimas nn digipööret, mis seisneb koolidesse nutiseadmete ja teiste tehnoloogiliste lahenduste soetamises ja aktiivses rakendamises erinevate õppeainete raames. Siinkohal on oluline ka õpetajate ja juhendajate koolitamine, kes oskaksid lapsi vastavalt juhendada, toetaksid nende arengut selles valdkonnas maksimaalselt ning panustaksid huvitavate ja õpetlike õppematerjalide loomisesse.

Eestis tegeleb infotehnoloogiaga seotud õppetöö ja huviringide toetamise ning edendamisega Vaata Maailma SA Nutilabori projekt. Õppetöö efektiivseks korraldamiseks vajab Nutilabor infosüsteemi, mis võimaldaks kõiki õppematerjale mugavalt ladustada ja hallata. Käesoleva bakalaureusetöö eesmärk on koostada vastav infosüsteem, lähtudes Nutilabori kui tellija vajadustest. Tellija vajadusi vahendab Tartu Ülikooli robotikaspetsialist Taavi Duvin.

Koostatud töö koosneb kahest osast: esiteks käesolev dokument, mis kirjeldab ja seletab lahti koostatud rakenduse ehituspõhimõtted ja teiseks loodud rakendus. Dokumendis antakse taustaülevaade robotika ja programmeerimise valdkonna kohta Eestis ning selgitatakse vajadust koostada ühtne infosüsteem õppeülesannete haldamiseks. Lisaks kirjeldatakse tellijapoolseid nõudeid, täpseid ootusi infosüsteemile ning süsteemi arhitektuuri vastavalt eelnevalt seatud nõuetele. Samuti on toodud rakenduse kasutamise stsenaariumid. Dokumendi lõpus antakse kokkuvõtte tehtud tööst ning arutletakse võimaluste üle, kuidas infosüsteemi tulevikus täiendada. Bakalaureusetöö oluliseks osaks on loodud infosüsteem "Nutirobot"¹ (Lisa 1).

¹ <http://nutirobot.elasticbeanstalk.com>

2. Õppematerjalide infosüsteem – senised arengud ja vajadused

Nutilabor[1] hõlmab endas IKT-huviringe noortele, mille Vaata Maailma SA käivitas 2012. aastal koostöös kolme partneriga (EMT, Elion ja Microsoft). Nutilabori eesmärk on toetada ja edendada õpilastele mõeldud infotehnoloogiaga (IT) seotud õppetööd ja huviringe. See läbi suureneb teadlikkus IT valdkonna võimaluste osas ja kasvab noorte inimeste arv, kes otsustavad jätkata õpinguid IT erialal. Pikemas perspektiivis võimaldab see suurendada IT spetsialistide arvu Eestis, läbi mille väheneb tööjõupuudus antud valdkonnas. Teiselt poolt vähendab see tööpuudust noorte hulgas, kuna kasvab erialase ettevalmistusega inimeste arv. Nutilaboris tegelevad õpilased peamiselt robootika ja elektroonikaga, programmeerimise ja arvutiehitusega ning loodus-, täppis- ja rakendusteadustega, luues eelduse huvi tekkimiseks antud erialade vastu juba varajases eas.

Nii robootika kui programmeerimise valdkonna tutvustamise ja edendamise üheks suureks probleemiks on asjakohaste õppematerjalide puudumine või raske kättesaadavus. Palju materjale on olemas võõrkeeltes (nt ingliskeelne dokumentatsioon robotite tehniliste omaduste kohta), vähem eesti keeles ning enamasti on need erinevates kohtades laiali. Eestikeelsete materjalide koostamisel on tänuväärse töö teinud Tartu Ülikooli Matemaatika-informaatikaeaduskonna tudengid, kes on oma lõputööde raames koostanud robotite ja nendel rakendavate andurite ülevaateid ning võimaldanud nende sidumise õppematerjalideks (kättesaadavad Tartu Ülikooli elektrooniliste materjalide ehk DSpace keskkonnas[2]). Lisaks tudengitele koostavad õppematerjale ka koolide ja huviringide õpetajad. Pidevalt uute ja huvitavate materjalide koostamine, mis hõlmaks endas kõige uuemate tehnoloogiliste lahenduste kirjeldusi, võtab aga küllaltki palju aega. Näiteks robootika tutvustamisel on hetkel Eestis suures enamuses kasutusel LEGO poolt toodetud Mindstorms NXT robotid (2006. aasta väljalase). Nutilabori loomise ajal olid need kõige uuema põlvkonna legorobotid, nüüdseks on nende tootmine ja müük lõpetatud. 2013. aastal lasti välja järgmise põlvkonna robotid LEGO Mindstorms EV3. Neid on hakatud järk-järgult järjest rohkem kasutusele võtma, kuid võrreldes NXT robotiga on need keerulisemad ja nende tundmaõppimiseks on seda enam vaja tutvustavaid emakeelseid tutvustavaid materjale. Samuti on populaarsust kogumas Arduino mikroprotsessor ja miniarvuti Raspberry Pi.

Kokkuvõttes tuleb nentida, et praeguseks hetkeks on teatud hulk robootikat ja programmeerimist tutvustavaid õppematerjale juba eesti keeles olemas, kuid keerulisem on neid üles leida ja kätte saada, sest puudub ühtne süsteem nende materjalide haldamiseks. On kujunenud olukord, kus ei ole ülevaadet, millised materjalid on olemas, millised teemad vajavad täiendamist ja millises suunas materjalide koostamisega edasi minna. Programmeerimise ja koolirobootika populariseerimiseks ja edasiarendamiseks on tarvis ühtset infosüsteemi, milles kasutajad saaksid enda koostatud materjale jagada, vajadusel teiste poolt loodud materjalidega tutvuda, neid kasutada.

Infosüsteemi vajadust ja õpetajate ootusi sellele on uurinud Marit Saviir oma bakalaureuse-töös “Roboteid omavate Eesti koolide õpetajate ning juhendajate hinnangud koolirobootikaga seonduvale ning ootused robootikateemaliste ülesannete ja juhendite andmebaasile” [3]. Selle töö eesmärgiks oli välja selgitada, milline on robootikateemalisi ülesandeid ning juhendeid koondava andmebaasi vajadus. Töö raames küsitles ta robootikaga tegelevate

koolide õpetajaid ja huviringide juhendajaid. Kokkuvõtvalt leidis ta, et enamus õpetajaid eelistab saada informatsiooni veebi vahendusel ning meelsasti kasutaks selleks ühtset veebikeskkonda.

Hetkel eksisteerib Eestis lahendus nimega „Robootika ülesannete portaal“[4], mille koostamise eesmärgiks oli samuti õppematerjalide koondamine. Kahjuks ei ole see lahendus täitnud oma eesmärki. Tellijad on toonud välja järgmised põhjendused: 1) portaal ei ole piisavalt mugav, huviline peab portaali kasutamiseks looma/registreerima eraldi konto ning pidevalt kontoga seotud andmeid (kasutajatunnus, parool) meeles pidama; 2) portaali sisu on piiratud ainult robootika ülesannete jaoks, mis takistab teistsuguste õppeülesannete (nt programmeerimisülesanded) avaldamist; 3) ülesandeid saab üles laadida koos ülesandepüstituse, vihjete ja lahendusega korraga, mistõttu kaob võimalus probleemülesandeid otse õpilastele esitada, kuna vastused asuvad samal materjalil; 4) nimetatud portaal loodi kolm aastat tagasi ja on visuaalselt ja tehnoloogiliselt aegunud – vähene atraktiivsus on aga üks võimalikest põhjustest, miks see portaal ei saavutanud kasutajate seas menu.

Käesoleva bakalaureusetöö raames on koostatud infosüsteem, lähtudes nii Marit Saviiri bakalaureusetöö tulemustest kui ka varasemalt koostatud ja eelnevalt kirjeldatud robootika ülesannete portaalist. Töö raames loodud portaal võimaldab kokku koondada kõik robootika ja kogu infotehnoloogiaga seotud õppematerjalid. Kasutajad saavad huvipakkuvaid õppematerjale otsida, lisada ning vajadusel redigeerida.

3. Nõuded infosüsteemile

Tellijad esitasid õppematerjalide haldamise infosüsteemile kindlad nõudmised. Õppematerjale peab saama üles laadida, neid hiljem vajadusel uuendada või kustutada. Õppematerjali üleslaadimisel peab olema võimalik defineerida kuus komponenti:

- ülesande pealkiri,
- ülesande sisuline püstitus,
- vihje ülesande lahenduse leidmiseks,
- kogu lahendus ülesandele,
- sildid (ingl k *tags*) ülesande kirjeldamiseks,
- ülesande privaatsusaste, mis defineerib, kas ülesanne on avalikult kõigile näha või ainult kasutajale endale.

Samuti peab kasutajal olema võimalus keskkonda pildifaile üles laadida. Pildifaile üleslaadimiseks peab igal kasutajal olema oma portfoolio, milles on võimalik pilte hallata. Neid pilte on võimalik kasutada ülesannete illustreerimiseks teksti sees.

Õppematerjalide lihtsamaks leidmiseks peab olema võimalik neid siltide järgi filtreerida. Siltide all on mõeldud ülesannet kirjeldavaid ühe- või paarisõnalisi märksõnu. Kasutaja peab enda või teiste poolt üles laaditud ülesandeid saama PDF-vormingus alla laadida või teistele kasutajatele hüperlingina jagada. Nii PDF-vormingus alla laadimiseks kui kasutajatele hüperlingina jagamiseks peab olema võimalik ülesandeid eelnevalt endale sobivaks töödelda ning teatud ülesande sektsioone (nt lahendus) vastavalt vajadusele näidata või hoopis peita.

Tellijad pidasid oluliseks ka võimalust Google'i ja Faceooki kaudu registreerimiseks, et keskkonnaga liituda soovijad ei peaks eraldi kasutajat tegema ning uut kasutajatunnust ja parooli meeles pidama. Kasutajad on vaja jagada erinevatesse rollidesse, et säilitada mõningate ülesannete privaatsus: tavakasutajad ehk registreerimata kasutajad, tavalised registreeritud kasutajad ja täieõiguslikud kasutajad, kelleks on enamjaolt õpetajad, ringijuhid ja õppejõud. Tavakasutajad näevad ainult avalikke materjale. Registreeritud kasutajad näevad avalikke materjale ning saavad ise materjale koostada ja üles laadida, kuid kõik need materjalid on avalikud. Täieõiguslikud kasutajad näevad avalikke materjale, saavad ise materjale koostada, ja neil peab olema võimalus üksteise vahel õppematerjale jagada nii, et ainult nemad neid näeksid ja teistele kasutajagruppidele ei oleks need nähtavad ega kättesaadavad. Täieõiguslikuks kasutajaks saab ainult niimoodi, et pärast registreerimist kuvatakse kõigile juba olemasolevatele täieõiguslikele kasutajatele teade, et uus kasutaja on registreerunud ning palutakse kontrollida tema Google'i või Faceooki kontot, kas ta peaks samuti saama täieõiguslikuks kasutajaks või on ta tavaline kasutaja, näiteks õpilane.

Mittefunktsionaalsete nõueteks on infosüsteemi kaasaegsus ning kasutajamugavus. Raken-duse kasutamine peab olema intuitiivne. Rakendus peab esialgu toetama kahte enimkasutatavat internetibrauserit Google Chrome ja Mozilla Firefox.

4. Süsteemi arhitektuur

Rakenduse ülesehitus lähtub tellijapoolsetest põhinõuetest: 1) rakenduse tänapäevane väljanägemine ja mugav kasutamine, 2) erinevate õigustega kasutajarollid, 3) kasutaja lihtne autentimine ja 4) õppematerjalide lihtne haldamine. Sellest tulenevalt on järgnevatel alampeatükkides kirjeldatud loodud rakenduse arhitektuuri, kasutajarolle ja autentimist. Lähemalt on selgitatud ka rakenduse andmebaasi ülesehitust ning rakenduse loomiseks kasutatud tehnoloogiaid.

4.1 Üheleheküljeline rakendus

Üheleheküljeline rakendus[5] (ingl k *single-page application*; SPA) on rakendus, mille eesmärk on pakkuda kasutajale kiireimat ja sujuvaimat kasutamiskogemust, mis saavutatakse sellega, et rakendus loetakse brauseri poolt tervikuna ja ainult ühe korra. Kui tavapärasel veebirakendusel peab iga alamlehe laadimiseks tegema täieliku lehe värskenduse, siis SPA puhul toimub suhtlus serveriga asünkroonselt. Asünkroonne suhtlus tähendab seda, et suhtlus toimub taustal ning kasutaja jaoks on rakenduse sujuvus võrreldav isegi töölauarakendusega (ingl k *desktop application*). Asünkroonne suhtlus on ka kiirem, sest suhtluse eesmärk pole tervet lehte uuesti alla laadida, vaid üldjuhul on eesmärgiks pärida sisu, mida kasutajale kuvada. Kuna päring kannab vähem infot, siis on ka ooteaeg kasutajale minimaalne.

4.2 Kasutajate rollid

Portaali kasutajatena nähakse üldhariduskoolide õpetajaid, huviringide juhendajaid ja teisi robootikast või programmeerimisest huvitunud inimesi, sealhulgas kindlasti õpilasi. Võttes arvesse erinevate kasutajate vajadusi ning ootusi portaalile, jaguneb kasutajaskond järgmisse kolme gruppi. **Registreerimata kasutajana** nähakse eelkõige õpilasi, aga ka kõiki teisi inimesi, kes on huvitatud programmeerimisest ja robootikaga tegelemisest. Antud rollis olevad kasutajad saavad ligi ainult avalikele materjalidele. **Registreeritud tavakasutajal** on samad õigused nagu registreerimata kasutajal, aga ka võimalus ise avalikke materjale üles ja alla laadida. **Registreeritud täieõiguslikul kasutajal ehk õpetaja-kasutajal** on samad õigused nagu registreeritud tavakasutajal, aga lisaks sellele pääseb ligi materjalidele, mis on mõeldud ainult haridustöötajatele. Selle rolli kasutajaskonna moodustavadki üldjuhul õpetajad, õppejõud, huviringide juhendajad ning teised haridustöötajad. Täieõiguslikuks registreeritud kasutajaks saab siis, kui mõni olemasolev täieõiguslik kasutaja aktsepteerib registreeritud tavakasutaja vastuvõtmise täieõiguslike kasutajate hulka.

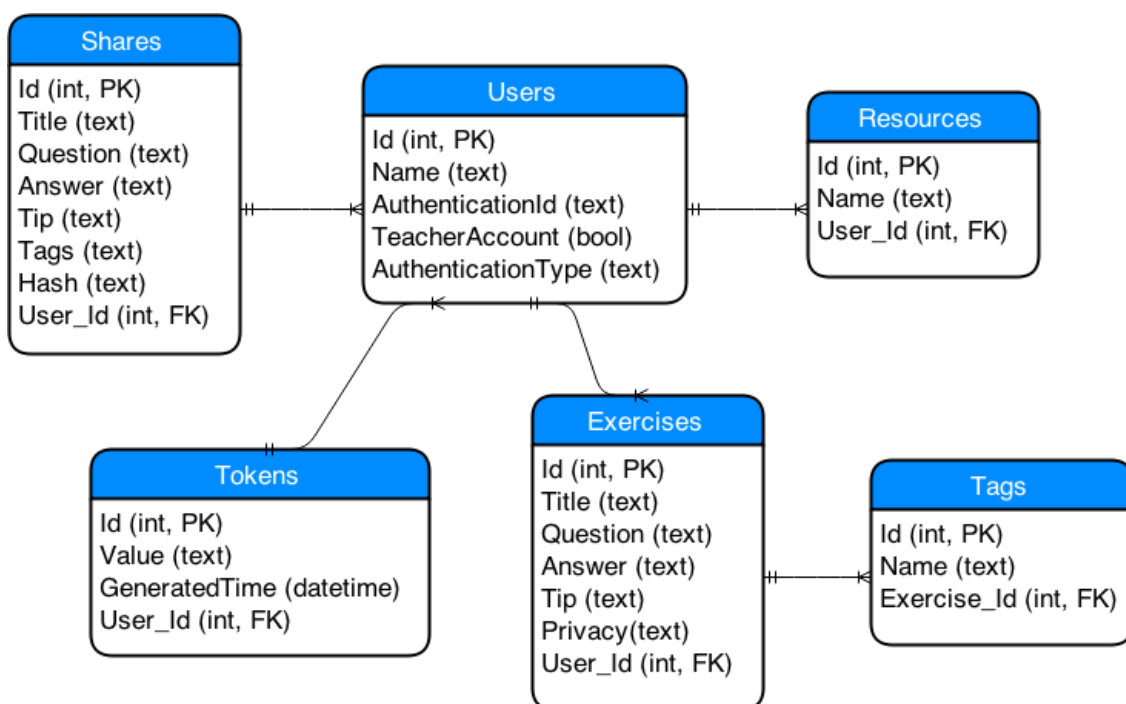
4.3 Autentimine ja autoriseerimine

Rakendusse konto avamine toimub läbi populaarse sotsiaalmeediavõrgustiku Facebook või Google'i konto kaudu. Robootikahuviline, kes tahab registreeruda, peab ühes vastavas süsteemis omama kontot, mis registreerimisel seotakse portaali kontoga. Autentimine toimub

ühe eespool mainitud konto kaudu. Iga kord, kui kasutaja sisse logib, tuleb tal sisestada oma kasutajanimi ja parool vastavasse hüpikaknasse, mille järel tagastatakse kasutajale ajutine identsustõend (ingl k *token*). Kuna ajutine identsustõend aegub üsna kiiresti, siis genereeritakse rakenduse poolt aegumatu identsustõend, mis salvestatakse kasutaja brauserisse. Tänu sellele ei pea kasutaja aegunud identsustõendi korral iga kord Google'i või Facebooki kaudu uuesti sisse logima, vaid kui rakendus serveriga suhtleb, saadetakse rakenduse poolt genereeritud identsustõend päringuga kaasa ja vastavalt sellele toimub kasutaja identiteedi kontroll ning autoriseerimine ehk määratakse see, mis infole kasutajal on õigus ligi pääseda.

4.4 Andmemudel

Rakenduse andmemudel annab ülevaate rakenduse ülesehitusest. Andmebaasis on kuus relatsioonilist tabelit (joonis 1).



Joonis 1. Rakenduse andmebaasi struktuur.

Relatsiooniliste tabelite struktuur ja veergude sisu on järgnevalt lahti kirjutatud.

Tabel **Users** säilitab infot kasutajate kohta, sisaldades järgmisi veerge:

- Id – hoiab tabeli primaarvõtit;
- Name – hoiab registreeritud kasutaja täisnime;
- AuthenticationId – hoiab registreeritud kasutaja Google'i või Facebooki konto identifikaatorit;

- TeacherAccount – defineerib, kas kasutaja konto on saanud registreeritud täieõigusliku kasutaja staatuse;
- AuthenticationType – näitab, läbi millise konto kasutaja registreerus, võimalikeks väärtusteks on kaks eeldefineeritud väärtust: *FACEBOOK*, *GOOGLE*;

Tabel **Exercises** sisaldab infot õppeülesannete kohta, sisaldab järgmisi veerge:

- Id – hoiab tabeli primaarvõtit;
- Title – hoiab ülesande pealkirja;
- Question – sisaldab ülesandepüstitust ja kirjeldust;
- Answer – hoiab ülesande vastust;
- Tip – sisaldab vihjet ülesande lahendamiseks;
- Privacy – defineerib ülesande privaatsusastme, võimalikeks väärtusteks on kaks eeldefineeritud väärtust: *PUBLIC_TO_TEACHERS*, *PUBLIC*;
- User_Id – hoiab välisvõtit, mis vastab tabeli “Users” primaarvõtmele;

Tabel **Tags** hoiab ülesannete külge pandud silte (ingl k *tags*), sisaldades järgmisi veerge:

- Id – hoiab tabeli primaarvõtit;
- Name – sisaldab sildi nime;
- Exercise_Id – hoiab välisvõtit, mis vastab tabeli “Exercises” primaarvõtmele;

Tabel **Shares** hoiab infot ülesannete kohta, mis on kasutajatele välja jagatud. Tabel sisaldab järgimisi veerge:

- Id – hoiab tabeli primaarvõtit;
- Title – hoiab ülesande pealkirja;
- Question – sisaldab ülesandepüstitust ja kirjeldust;
- Answer – hoiab ülesande vastust;
- Tip – sisaldab vihjet ülesande lahendamiseks;
- Tags – hoiab siltide nimekirja, mis kirjeldavad ülesannet;
- Hash – sisaldab unikaalset identifikaatorit, läbi mille on võimalik jagatud ülesandele ligi pääseda;
- User_Id – hoiab välisvõtit, mis vastab tabeli “Users” primaarvõtmele;

Tabel **Tokens** hoiab kasutajatele genereeritud identsustõendeid, sisaldades järgimisi veerge:

- Id – hoiab tabeli primaarvõtit;
- Value – hoiab ülesande pealkirja;
- GeneratedTime – hoiab kuupäeva ja kellaaega, millal identsustõend genereeriti;
- User_Id – hoiab välisvõtit, mis vastab tabeli “Users” primaarvõtmele;

Tabel **Resources** hoiab kasutaja üleslaaditud piltide infot. Tabel sisaldab järgimisi veerge:

- Id – hoiab tabeli primaarvõtit;
- Name – hoiab faili nime;
- User_Id – hoiab välisvõtit, mis vastab tabeli “Users” primaarvõtmele.

4.5 Rakenduse arendamiseks kasutatud tehnoloogiad

Loodud rakendus jaguneb kaheks osaks: kliendipoolne liides ehk eessüsteem (ingl k *front-end*) ja serveripoolne süsteem ehk tagasüsteem (ingl k *back-end*). Kliendipoolne liides jookseb visuaalselt kasutaja brauseris ja kasutaja suhtleb ainult selle rakenduse osaga. Kasutaja ja serveri vahelise suhtluse koordinaatoriks on samuti eessüsteem. Serveripoolne süsteem on rakenduse osa, kus toimub suurem osa rakenduse tööst – info töötlemine, otsimine ja salvestamine. Selline protsessimine jääb tavaliselt kasutajale nähtamatuks ja tehakse välises serveris.

4.5.1 Serveripoolne süsteem

Serveripoolne süsteem on kirjutatud keeles C# ning põhineb Microsoft ASP.NET MVC[6] raamistikul. Raamistik on ASP.NET[7] koodilaiendus, mis jaotab rakenduse kolmeks põhiliseks osaks vastavalt Mudel-Vaade-Kontroller (ingl k *Model-View-Controller*; MVC) kihilisele arhitektuurile:

- **mudel**, kus hoitakse andmeid, mida rakenduse sees kasutatakse;
- **vaade**, mis esitab kasutajale andmed vastavalt mudelile;
- **kontroller** teeb tööd ja leiab mudelile väärtused ning tagastab vaate koos mudeliga.

Kuigi ASP.NET MVC on originaalis mõeldud Mudel-Vaade-Kontroller rakenduste jaoks, sobib see ideaalselt ka üheleheküljelisteks rakendusteks kasutades kõiki häid ASP.NET MVC omadusi, näiteks failide kokkuliitmine (ingl k *bundling*) ja minimeerimine (ingl k *minification*), siis eessüsteemiga suhtlemiseks on käesolevas töös hoopis kasutatud ASP.NET Web API-t. ASP.NET Web API on tuntud Microsofti poolt disainitud raamistik, mida kasutatakse veebiteenuste loomiseks[8]. See on oluline tehnoloogia, mida rakendus kasutab serveriga suhtlemiseks, nii andmete saatmiseks kui pärimiseks.

Andmete hoidmiseks kasutatakse Microsofti andmebaasilahendust Microsoft SQL Server. Suhtluseks andmebaasiga on kasutusel Entity Framework, mis on Microsofti poolt soovitatav standard andmebaasiga suhtlemiseks .NET rakendustel. Entity Framework on ORM (*object-relational mapping*) raamistik, mis aitab objektorienteeritud domeeni mudeleid siduda (*mapping*) traditsiooniliste relatsiooniliste andmebaasidega.

4.5.2 Kasutajapoolne liides

Kasutajapoolne liides on üles ehitatud järgides üheleheküljeliste rakenduste disainireegleid (vt ptk 4.1), mille saavutamiseks on kasutatud AngularJS marsruutimist (ingl k *routing*). Samuti on kasutusel dünaamiline (ingl k *responsive*) veebidisain[9], mis tähendab, et rakendus arvestab, millise ekraaniga kodulehte vaadatakse ning mugandab end jooksvalt vastavalt sellele. Selle jaoks on kasutusel Bootstrap.

Kasutajapoolse liidese tuumana on kasutatud **AngularJS**i[10], mille näol on tegu avatud lähtekoodiga Javascripti veebiraamistikuga, mis võimaldab arendajal ehitada ühelehekülje-

lisi moduleeritud veebirakendusi. Peamine põhjus, miks rakenduse loomiseks otsustati kasutada AngularJSi, on selle arhitektuur, mis lubab arendajal produktiivselt rakendusi arendada. Teiseks oluliseks põhjuseks on selle ülemaailmne populaarsus, kogukonna tugevus ning tänu sellele ka erinevate õpetuste ning kogukonnaliikmete poolt ehitatud moodulite mitmekesisus. Google Trends² järgi (2015. aasta maikuu seisuga) on AngularJS teistest konkureerivatest veebiraamistikest kümneid kordi populaarsem ja ennustab selle kasvu jätkumist ka tulevikus.

AngularJS veebiraamistiku peale saab ise arendada või kasutada juba olemasolevaid erinevaid mooduleid ja komponente. Tavaliselt on need moodulid mõeldud rakenduse integreerimiseks teiste erinevate teenustega või rakenduse töö lihtsustamiseks.

Käesolevas töös on kasutatud järgmisi moodulid (vt Lisa 2):

- *ui-grid* – moodul dünaamiliste tabelite kuvamiseks;
- *ng-storage* – moodul brauseri lokaalse lao (ingl k *local storage*) haldamiseks, mis hoiab kasutaja identsustõendit;
- *text-angular* – moodul veebipõhise tekstiredaktori kasutamiseks;
- *toaster* – moodul rakenduse teadete haldamiseks;
- *satellizer* – moodul erinevate sotsiaalmeediakanalite kaudu autentimiseks;
- *bootstrap-ui* – toob rakendusse dünaamilised veebikomponendid, näiteks navigeerimispaneel;
- *kendo-ui* – toob rakendusse dünaamilised veebikomponendid nagu “*kendo-button*”, “*kendo-window*”, “*kendo-splitter*”;
- *angular-file-upload* – moodul failide üleslaadimiseks;
- *ng-tags-input* – moodul dünaamiliste siltide sisestamiseks;
- *ng-progress* – moodul ooteaja visualiseerimiseks;

Rakenduse dünaamiliseks muutmise eest vastutab raamistik **Bootstrap**[12]. Bootstrap on raamistik, mis võimaldab arendajal kiiresti ja lihtsalt dünaamilisi rakendusi ehitada. Rakenduses kasutatakse Bootstrap'i *grid systemit*, mis lubab HTML elementidele lisada klasse (nt *container-fluid*, *row*, *col-md-1*, *col-md-5*), mis jagavad vastava elemendi kaheteistkümneks dünaamiliseks osaks, mille suurus ja kuju muutuvad vastavalt ekraani suurusele.

² Google Trends baseerub inimeste poolt tehtud Google'i otsingutel[11].

5. Süsteemi kasutuslood

Kasutuslood kirjeldavad süsteemi kasutamise erinevad stsenaariumeid mida kasutajal on vastavalt oma õigustele rakenduses võimalik teha.

Kasutuslugu 1. Registreerimine ja sisselogimine

- **Kirjeldus:** Uus kasutaja registreerib endale konto rakenduses.
- **Tegutseja:** Rakenduse registreeritud kasutajaks saada sooviv isik.
- **Eeldused:** Isikul on rakendus avatud ja näeb sisselogimisnuppe.
- **Põhivoog:**
 - Kasutaja vajutab nupule “Logi sisse Facebookiga” või “Logi sisse Google’iga” (vt Lisa 2 joonis 4).
 - Avaneb aken, kus kasutaja peab sisestama valitud rakenduse kasutajanimi ja parooli.
 - Kasutaja täidab vormi väljad.
 - Kasutaja vajutab vastavalt valitud keskkonnale nupule “*Log in*” (Facebook) või “*Sign in*” (Google) (vt Lisa 2 joonis 5 ja 6).
 - Registreerimisnupu asemele tekib tekst “Tere, //kasutaja nimi//!” ja ilmub nupp “Logi välja”.
- **Tulemus:** Kui kasutajale on see esimene sisselogimine rakendusse, siis registreeritakse talle uus konto, mis on seotud kas Google’i või Facebooki kontoga ning seejärel kuvatakse rakenduse esileht. Kui kasutaja jaoks ei ole see esmane sisselogimine, siis tuntakse ta süsteemis ära ja kuvatakse rakenduse esileht.
- **Alternatiivsed vood:**
 - Sisestatud kasutajanimi või parool on vale.
 - Kasutajale kuvatakse veateada, kui kasutajanimi ja parool ei ühti või ei eksisteeri.

Kasutuslugu 2. Ülesande lisamine

- **Kirjeldus:** Kasutaja sisestab uue ülesande.
- **Tegutseja:** Registreeritud kasutaja.
- **Eeldused:** Kasutaja on sisse logitud ja asub lehel “Lisa ülesanne” (Joonis 2).
- **Põhivoog:**
 - Kasutaja sisestab ülesande pealkirja.
 - Kasutaja sisestab ülesande probleemi tekstina.
 - Kasutaja täidab vihje lahtri (valikuline väli).
 - Kasutaja sisestab ülesande lahenduse.
 - Kasutaja sisestab ülesannet kirjeldavad sildid.
 - Kasutaja vajutab nupule “Laadi üles”.
 - Kasutajale kuvatakse teade “Ülesande salvestamine õnnestus”.
- **Tulemus:** Ülesanne salvestati andmebaasi.
- **Alternatiivsed vood:**
 - Kui tegemist on õpetaja-kasutajaga.

- Kasutaja saab enne üles laadimist valida privaatsuse astme “Avalik ainult õpetajatele” ja “Avalik” vahel.
- Mõni lahter on täitmata.
 - Kasutajale kuvatakse teade “Palun täitke kõik lahtrid”.

Ülesande pealkiri

Ülesande püstitus

H1 H2 H3 B I U ☰ ☱ ☲ ☳ ☴ ☵ ☶ ☷ 

Vihje

H1 H2 H3 B I U ☰ ☱ ☲ ☳ ☴ ☵ ☶ ☷ 

Lahendus

H1 H2 H3 B I U ☰ ☱ ☲ ☳ ☴ ☵ ☶ ☷ 

Sildid

Privaatsus

Avalik Avalik ainult õpetajatele

[Laadi üles](#)

Joonis 2. Ülesande lisamise vorm.

Kasutuslugu 3. Lisatud ülesannete filtreerimine

- **Kirjeldus:** Kasutaja filtreerib sobivad ülesanded välja siltide järgi.
- **Tegutseja:** Tegutsejaks võib olla ükskõik milline rakenduse kasutaja.
- **Eeldused:** Kasutaja asub lehel “Sirvi ülesandeid”.
- **Põhivoog:**
 - Kasutaja sisestab kohustuslikud sildid (valikuline väli) (vt Lisa 2 joonis 7).
 - Kasutaja sisestab keelatud sildid (valikuline väli).
 - Kasutaja vajutab nuppu “Filtreeri”.
- **Tulemus:** Üleslaaditud materjalide nimekiri uuendatakse vastavalt sisestatud filtrile.

Kasutuslugu 4. Ülesande välja jagamine

- **Kirjeldus:** Kasutaja jagab ülesannet.
- **Tegutseja:** Registreeritud kasutaja.

- **Eeldused:** Kasutaja paikneb lehel “Sirvi ülesandeid”.
- **Põhivoog:**
 - Kasutaja vajutab tabelis ülesandele, mida ta välja jagada tahab.
 - Ülesande kohta avaneb detailne vaade.
 - Kasutaja vajutab nuppu “Jaga”.
- **Tulemus:** Avaneb aken, kus on kirjas, mis leheküljel on näha jagatud õppematerjali.
- **Alternatiivsed vood:**
 - Kasutaja tahab enne jagamist ülesannet muuta.
 - Kasutaja märgistab kasti “Redigeeri”.
 - Kasutaja uuendab vormil väljad, mida ta muuta tahab.

Kasutuslugu 5. Ülesande allalaadimine PDF-failina

- **Kirjeldus:** Kasutaja laadib alla ülesande PDF-failina.
- **Tegutseja:** Registreeritud kasutaja.
- **Eeldused:** Kasutaja paikneb lehel “Sirvi ülesandeid”.
- **Põhivoog:**
 - Kasutaja vajutab tabelis ülesandele, mida ta alla laadida soovib.
 - Ülesande kohta avaneb detailne vaade.
 - Kasutaja vajutab nuppu “Laadi alla PDF”.
- **Tulemus:** Kasutajale laaditakse alla PDF-fail ülesandega.
- **Alternatiivsed vood:**
 - Kasutaja tahab enne alla laadimist ülesannet muuta.
 - Kasutaja märgistab kasti “Redigeeri”.
 - Kasutaja uuendab vormil väljad, mida ta muuta soovib.

Kasutuslugu 6. Ülesande redigeerimine ja salvestamine

- **Kirjeldus:** Kasutaja redigeerib ülesannet ja salvestab selle.
- **Tegutseja:** Registreeritud kasutaja.
- **Eeldused:** Kasutaja on sisse logitud, paikneb lehel “Sirvi ülesandeid” ning kasutaja on ülesande autor.
- **Põhivoog:**
 - Kasutaja vajutab tabelis ülesande peale, mida ta redigeerida tahab.
 - Ülesande kohta avaneb detailne vaade.
 - Kasutaja märgistab kasti “Redigeeri”.
 - Kasutaja uuendab vormil väljad, mida ta muuta soovib.
 - Kasutaja vajutab nupule “Muuda”.
 - Kasutajale kuvatakse teade “Ülesande muutmise õnnestus”.
- **Tulemus:** Kasutaja ülesanne uuendatakse vastavalt muudetud väljadele.

Kasutuslugu 7. Ülesande kustutamine

- **Kirjeldus:** Kasutaja kustutab enda poolt üles laaditud ülesande.
- **Tegutseja:** Registreeritud kasutaja.
- **Eeldused:** Kasutaja on sisse logitud, paikneb lehel “Sirvi ülesandeid” ning kasutaja on ülesande autoriks.
- **Põhivoog:**
 - Kasutaja vajutab tabelis ülesande peale, mida ta kustutada tahab.
 - Ülesande kohta avaneb detailne vaade.
 - Kasutaja märgistab ära kasti “Redigeeri”.
 - Kasutaja vajutab nupule “Kustuta”.
 - Kasutajale kuvatakse teade “Ülesande kustutamine õnnestus”.
 - Kustutatud ülesanne eemaldatakse tabelist.
- **Tulemus:** Kasutaja ülesanne kustutatakse andmebaasist.

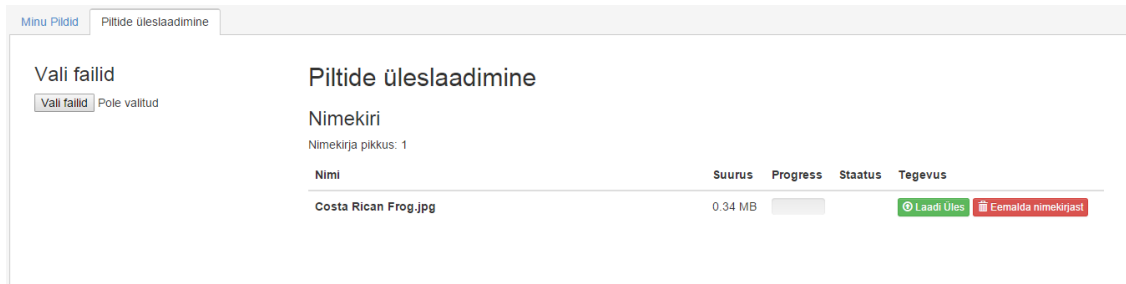
Kasutuslugu 8. Kasutaja konto muutmine õpetaja-kontoks

- **Kirjeldus:** Täieõiguslikuks kasutajaks ehk õpetajastaatuse saamiseks peab mõni juba registreeritud täieõiguslik kasutaja talle õpetajarolli määrama.
- **Tegutseja:** Kasutaja, kes on registreeritud ja on täieõiguslik kasutaja.
- **Eeldused:** Olemas on mõni registreeritud kasutaja ja tema taotlust pole veel üle vaadatud ehk ootab vastuvõtmist või tagasilükkamist.
- **Põhivoog:**
 - Kasutaja vajutab navigeerimise menüül nupule “Vaata õpetajataotlusi”.
 - Avaneb tabel, kus on näha nimekiri kasutajatest, kes ootavad õpetaja staatuse saamist.
 - Kasutaja vajutab nupule “Lisa kasutaja õpetajarolli”.
 - Kasutajale kuvatakse teade “Kasutaja õpetajaks lisamine õnnestus”.
 - Vastu võetud kasutaja kirje kustutatakse tabelist.
- **Tulemus:** Valitud kasutaja lisatakse täieõiguslike kasutajate nimekirja.
- **Alternatiivsed vood:**
 - Kasutaja vajutab nupule “Lükka tagasi”.
 - Kasutajale kuvatakse teade “Kasutaja taotlus lükati tagasi”.
 - Kasutaja kaob tabeli nimekirjast.

Kasutuslugu 9. Piltide üles laadimine

- **Kirjeldus:** Kasutaja laadib üles pildid enda portfooliosse.
- **Tegutseja:** Registreeritud kasutaja.
- **Eeldused:** Kasutaja on sisse logitud, paikneb lehel “Minu pildid” (Joonis 3).
- **Põhivoog:**
 - Kasutaja vajutab nupule “Piltide üleslaadimine”.
 - Avaneb alamleht piltide üles laadimiseks.
 - Kasutaja vajutab nupule “Vali failid”.

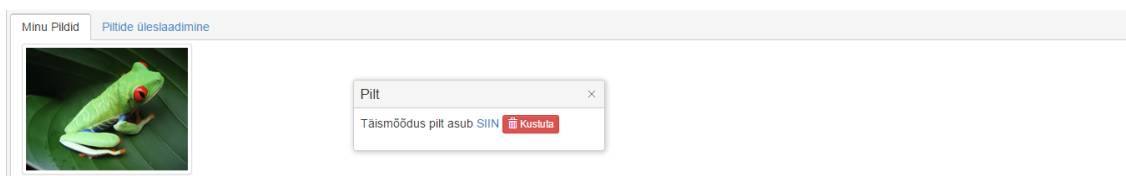
- Kasutaja vajutab nupule “Ava”.
- Valitud pilt või pildid lisatakse piltide üleslaadimise nimekirja.
- Kasutaja vajutab iga pildi juures olevale nupule “Laadi üles”.
- Pildi juures olev progressiriba täitub ja staatuse juurde tekib linnuke.
- **Tulemus:** Valitud pildid laaditi üles kasutaja portfooliosse.
- **Alternatiivsed vood:**
 - Kasutaja vajutab nupule “Eemalda nimekirjast”.
 - Pilt eemaldatakse piltide üleslaadimise nimekirjast.



Joonis 3. Piltide üleslaadimine.

Kasutuslugu 10. Pildi täismõõdus vaatamine

- **Kirjeldus:** Kasutaja soovib pilti vaadata täismõõdus, samuti saab ta kätte pildi internetiaadressi, mida saab kasutada ülesannete üleslaadimisel, kui soovitakse ülesannet pildiga illustreerida.
- **Tegutseja:** Registreeritud kasutaja.
- **Eeldused:** Kasutaja on sisse logitud, paikneb lehel “Minu pildid” (Joonis 4).
- **Põhivoog:**
 - Kasutaja vajutab sobivale pildile.
 - Avaneb aken pildi kohta.
 - Kasutaja vajutab nupule “SIIN”.
 - Avaneb uus aken, kus pilt on esitatud täismõõdus.
- **Tulemus:** Kasutaja näeb pilti täismõõdus.



Joonis 4. Pildi vaatamine ja kustutamine.

Kasutuslugu 11. Pildi kustutamine portfooliost

- **Kirjeldus:** Kasutaja kustutab valitud pildi oma portfooliost.
- **Tegutseja:** Registreeritud kasutaja.
- **Eeldused:** Kasutaja on sisse logitud, paikneb lehel “Minu pildid” (Joonis 4).

- **Põhivoog:**
 - Kasutaja vajutab sobivale pildile.
 - Avaneb aken pildi kohta.
 - Kasutaja vajutab nupule “Kustuta”.
 - Kasutajale kuvatakse teade “Faili kustutamine õnnestus”.
- **Tulemus:** Valitud pilt kustutati portfooliost.

6. Rakenduse analüüs ja testimine

Rakenduse analüüsimisel võeti vaatluse alla süsteemi kasutajasõbralikkus ja kiirus. Töö autor hindas süsteemi kasutajasõbralikkust visuaalselt ja selle käigus käsitleti parameetreid, mis võiksid rakenduse kasutaja seisukohalt olulised olla. Kasutajasõbralikkuse analüüsimisel hinnati, kui mugav ja intuiitiivne on loodud veebirakendust kasutada. Hinnatud parameetrid on toodud tabelis 1.

Tabel 1. Hinnangud süsteemi kasutajasõbralikkusele.

Parameeter	Hinnang
Informatsiooni leidmise lihtsus	Rakenduse sisuliseks informatsiooniks on üleslaaditavad ülesanded. Pealehel on kahest kohast võimalik jõuda ülesannete lehele, kus on toodud üleslaaditud avalike ülesannete nimekiri. Nimekirjast on ühe klikiga võimalik näha terve ülesande detailset sisu.
Rakenduse sisutekstide õigekiri ja arusaadavus	Rakenduses esitatud tekstides õigekirjavigu ei esine, info (sh nuppude tähendused) on selgelt edastatud.
Rakenduses liikumise loogilisus ja arusaadavus	Peaaegu kogu liikumine toimub ühest kohast – vasakust menüüst. Paanide vahel liikumine on kiire.
Kasutaja teadlikkus progressist ja laadimisest	Iga kord, kui kasutaja alamlehtede vahel liigub või serverist infot ootab, kuvatakse talle progressiriba. Samuti annab kasutajale õnnestunud või ebaõnnestunud liigutusest teada ekraani paremasse ülanurka ilmuvad teated.
Informatsiooni ja elementide tihedus	Rakenduses on palju vaba pinda, mida on nt võtmekomponendiks peetud kergelt kasutatavate ja visuaalselt esteetiliste veebilehtede juures[13]. Rakenduses ei ole ebaolulisi elemente. Olemasolevad elemendid on paigutatud kasutajale käepäraselt.
Vähe erinevaid fonte ja ühtne tekstistiil	Rakenduses on kasutusel ainult üks font, teksti suurus teiste elementide suhtes on ühtlane ja loogiline.

Rakenduse kiiruse hindamiseks kasutati veebiteenust Google PageSpeed Insights[14], mis on mõeldud veebirakenduste analüüsimiseks ja optimeerimiseks. PageSpeed Insights analüüsib veebirakenduse ülesehitust, jõudlust ja kiirust ning annab sellele vastavalt rakendusele hinde. Eialgu sai loodud rakendus hindeks 31 punkti sajast. Olgugi, et Javascripti ja CSS-failid olid minimeeritud ja kokkuliidetud, siis peamiseks probleemiks oli ikkagi nende suurus, sest puudus gzip kokkupakkimise võimalus, mis võimaldab andmeside hulka vähendada kuni 70%[15]. Pärast gzip kokkupakkimise rakendamist sai lehekülge kiiruse eest hindeks 85 punkti sajast. Kokkuvõttes on Google PageSpeedi analüüsist näha, et suur enamus hinnatud näitajatest on siiski head: 1) server vastab päringule kiiresti, 2) Javascript, HTML ja CSS on minimeeritud, 3) gzip kokkupakkimine on sisse lülitatud.

Hetkeseisuga on rakenduse tellija poolt esitatud funktsionaalsusnõuded täidetud. Seejuures on süsteemi visuaalne esitus suutnud jääda minimalistlikuks ja kasutajasõbralikuks, mis oli ka üks tellijapoolne nõue. Järgnevate sammudena on oluline, et sihtrühm, kellele antud rakendus mõeldud on, annaks rakendusele oma hinnangu.

7. Arutelu ja kokkuvõte

Käesoleva bakalaureusetöö eesmärgiks oli koostada infosüsteem kooliõpilastele suunatud programmeerimise ja robotika õppematerjalide haldamiseks. Ühtse infosüsteemi vajaduse tingis asjaolu, et õppematerjalid on küll olemas, aga need ei ole seni koondatud ühte keskkonda, kus kõik ülesanded paikneksid koos ja oleksid kõigile avalikult kasutada. Loodud infosüsteemi kaudu saavad selle kasutajad neid ülesandeid vastavalt tehnoloogiate arengule ja uutele ideedele vajadusel uuendada, täiendada ja parandada. Neist vajadustest lähtudes koostati infosüsteem, mis vastaks tellija ja kasutaja soovidele ning nõuetele.

- Kasutaja saab sisse logida olemasoleva Google'i või Facebooki konto kaudu ja ei pea eraldi ennast kasutajaks registreerima.
- Kasutaja saab õppematerjale üles laadida, defineerides eraldi lahtritesse ülesande osad nagu pealkiri, ülesande püstitus, vihje, lahendus ning vajadusel illustreerida asjakohaste joonistega.
- Üleslaaditud õppematerjale saab PDF-vormingus alla laadida ja teistele lingiga jagada. Sealjuures on võimalik originaalülesannet endale käepärasemaks teha ja teatud osad välja jätta või uusi juurde lisada.

Töö koostamise käigus tekkisid ka mõned mõtted, kuidas loodud infosüsteemi võiks tulevikus edasi arendada. Enamus neist on küll mugavusfunktsioonid, aga sellised täiendused tagaksid kasutajale hea kasutajakogemuse ja soovi infosüsteemi aktiivselt kasutada. Hea oleks, kui saaks õppematerjalide juurest avada piltide portfooliot ja sealt otse pilte ülesandele lisada. Hetkel toimub piltide lisamine nii olemasolevast portfooliost kui väliskeskonnast lingi abil. Teiseks võiks tulevikus olla huvilistele nähtav statistika portaali kasutajate ja kasutamise sageduse kohta: kui palju on registreeritud kasutajaid, milliseid õppematerjale ja kui palju on alla laetud, millised on populaarseimad kasutatud sildid jms. Kolmandaks võiks kasutajatel olla võimalus kommenteerida olemasolevaid õppematerjale, anda infot ülesande kohta, kommenteerides, mis töötas ja mis jäi segaseks jne. See aitaks ülesande autoril oma töös tekkinud vigu parandada või mõnda osa selgemalt lahti kirjutada.

8. Kasutatud kirjandus

- [1] Nutilabori kodulehekülg. [Online] <http://www.nutilabor.ee>
- [2] Dspace: repository for all electronic materials of Univeristy of Tartu. [Online] <http://dspace.utlib.ee/dspace/>
- [3] Saviir, M. (2014) Roboteid omavate Eesti koolide õpetajate ning juhendajate hinnangud koolirobootikaga seonduvale ning ootused robootikateemaliste ülesannete ja juhendite andmebaasile. Bakalaureusetöö, Tartu Ülikooli matemaatika-informaatikateaduskond, Tartu. [Online] http://dspace.utlib.ee/dspace/bitstream/handle/10062/41969/saviir_marit.pdf
- [4] Robootika ülesannete portaal. [Online] <http://www.robootika.ee/lego/portaal>
- [5] Single-page application. Wikipedia, the free encyclopedia. [Online] http://en.wikipedia.org/wiki/Single-page_application
- [6] ASP.NET MVC Overview. Microsoft Developer Network. [Online] <https://msdn.microsoft.com/en-us/library/dd381412%28v=vs.108%29.aspx>
- [7] ASP. NET Overview. Microsoft Developer Network. [Online] [https://msdn.microsoft.com/en-us/library/4w3ex9c2\(v=vs.140\).aspx](https://msdn.microsoft.com/en-us/library/4w3ex9c2(v=vs.140).aspx)
- [8] ASP.NET Web API. Microsoft Developer Network. [Online] <https://msdn.microsoft.com/en-us/library/hh833994%28v=vs.108%29.aspx>
- [9] Responsive web design. [Online] http://en.wikipedia.org/wiki/Responsive_web_design
- [10] Developer Guide. Angular JS. [Online] <https://docs.angularjs.org/guide>
- [11] Google Trend Compare. [Online] <https://www.google.com/trends/explore#q=angularjs%2C%20backbone.js%2C%20ember.js%2C%20knockoutjs&date=1%2F2012%2041m&cmpt=q&tz=>
- [12] Bootstrap'i kodulehekülg. [Online] <http://getbootstrap.com/>
- [13] Zen of White Space in Web UI Design. Balance, Contrast, Hierarchy. UXPin Inc. [Online] <http://blog.uxpin.com/6469/free-e-book-zen-white-space-web-ui-design-balance-contrast/>
- [14] PageSpeed Tools. Google Developers. [Online] <https://developers.google.com/speed/pagespeed/>
- [15] Page Speed: Enable gzip compression. GTmetrix. [Online] <http://gtmetrix.com/enable-gzip-compression.html>

Lisad

I. Infosüsteemi lähtekood

Infosüsteemi lähtekood on saadaval aadressil: <https://github.com/sjaanus/nutirobot>.

II. Näiteid AngularJSi moodulite kasutamisest rakenduses

ui-grid

Üleslaaditud materjalid

Tiitel	Kasutaja	Privaatsus	Sildid
Vernier' sensori näidu edastamine ...	Jaanus Sellin	Avalik	lihtne, Vernier, NXT, NXT-G
Alalisvoolu ja alalispinge mõõtmin...	Jaanus Sellin	Avalik	Vernier, NXT, robot, elekter

Sort Ascending

Sort Descending

Remove Sort

Hide Column

Joonis 1. Üleslaaditud ülesannete nimekiri. Näidatud on ka sorteerimisvalikud.

text angular



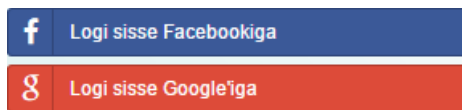
Joonis 2. Tekstiredaktor. Võimalus muuta teksti suurust, stiili ja joondust, sisestada pilte.

toaster

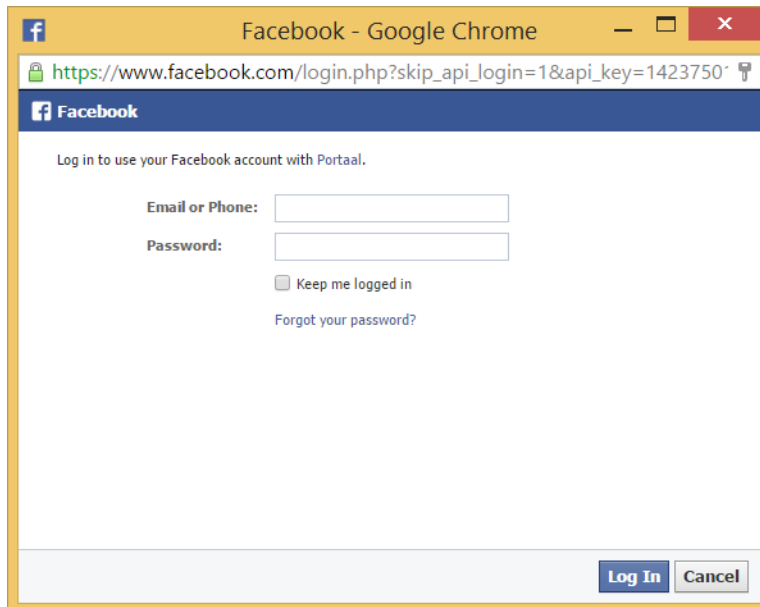


Joonis 3. Veateade ülesande üleslaadimise kohta, teade ülesande üleslaadimise õnnestumise kohta.

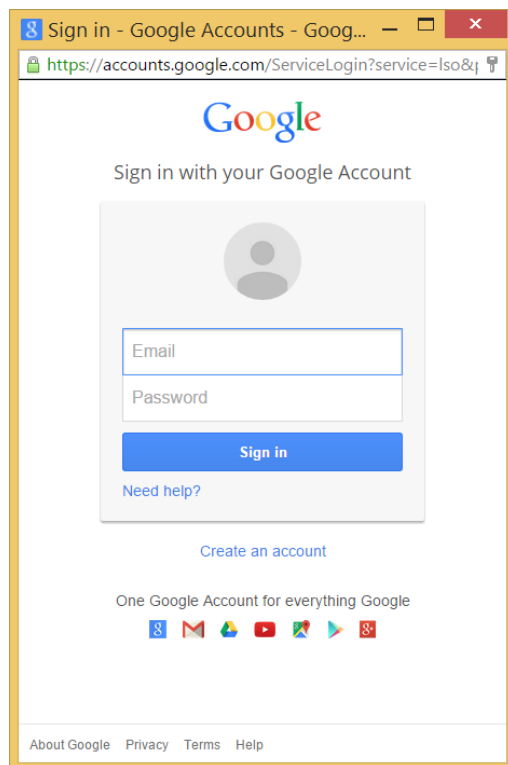
satellizer



Joonis 4. Sisselogimise valikud.



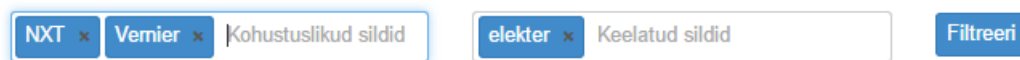
Joonis 5. Sisselogimine Facebooki kaudu.



Joonis 6. Sisselogimine Google'i kaudu.

ng-tags-input

Filtreerimine



The image shows a user interface for filtering search results. It features two input fields and a filter button. The first input field contains the text 'NXT x Vermier x Kohustuslikud sildid'. The second input field contains the text 'elekter x Keelatud sildid'. To the right of the second input field is a blue button labeled 'Filtreeri'.

Joonis 7. Õppematerjalide filtreerimine lubatud ja keelatud siltide abil.

III. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina, **Jaanus Sellin**

(sünnikuupäev: 27.05.1990)

annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose

Robotika ja programmeerimise õppematerjalide infosüsteem,

mille juhendajad on Taavi Duvin ja Anne Villems,

- 1.1 reprodutseerimiseks säilitamise ja üldsusele kättesaadavaks tegemise eesmärgil, sealhulgas digitaalarhiivi DSpace'is lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
- 1.2 üldsusele kättesaadavaks tegemiseks Tartu Ülikooli veebikeskkonna kaudu, sealhulgas digitaalarhiivi DSpace'i kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.
3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tartu, 14.05.2015