



TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

# Joomla veebilehtede turvalisuse tagamine

Diplomitöö

Juhendaja/õppejõud: Meelis Antoi

Üliõpilane Raido Martinson

Üliõpilase meiliaadress [raido.martinson@gmail.com](mailto:raido.martinson@gmail.com)

Õppekava nimetus Süsteemiadministraator

## **Autorideklaratsioon**

Deklareerin, et käesolev diplomitöö, mis on minu iseseisva töö tulemus, on esitatud IT Kolledž Tallinna Tehnikaülikool lõpudiplomi taotlemiseks IT süsteemide administreerimise erialal. Diplomitöö alusel ei ole varem eriala lõpudiplomit taotletud.

Autor R. MARTINSON.....

(allkiri ja kuupäev)

Töö vastab kehtivatele nõuetele

Juhendaja M. Antoi.....

(allkiri ja kuupäev)

## **Annotatsioon**

Antud lõputöö teemaks on "Joomla veebilehtede turvalisuse tagamine". Teema ajendiks olid autori leitud probleemid, nagu näiteks tõsiasi, et ilma uuendusteta võivad Joomla! veebilehed muutuda ebaturvaliseks. Ebaturvalised veebilehed võivad sattuda rünnaku ohvriks, mis võib kaasa tuua majanduslikku kahju nii teenust pakkuvale, kui tellivale ettevõttele.

Lõputöö raames loodud süsteemi JSite Manager abil on võimalik pärida veebilehtedelt informatsiooni ning vajadel viitab süsteem veebilehti uuendada. Lisaks on võimalik veebikeskkonna kaudu veebilehti varundada, eeldusel, et veebilehele on paigaldatud Akeeba Backup lisaprogramm. Testides loodud süsteemi, leidis autor, et viie veebilehe varundamise ning uuendamise ajakulu vähenes poole võrra. Samuti võib üheks tulemuseks lugeda seda, et JSite Manager on tänaseks päevaks võetud Reaktiiv Grupp OÜ's kasutusele.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 31 leheküljel, 5 peatükki, 11 joonist, 0 tabelit.

## **Abstract**

### **Ensuring the Security of Joomla Web Pages**

The theme of this thesis is “Ensuring the Security of Joomla Web Pages”. The incentive of the graduation thesis was the fact that without any new updates Joomla! websites can become unsafe very easily. Vulnerability makes websites an easy target for attackers. That can even become an economical damage to service provider and also to the client.

JSite Manager, a system created in the course of this thesis, allows to query information from websites and if needed, then also to refer them to update. In addition to it, it is also possible to back up different websites through the JSite Manager. But, in that case, there should be an Akeeba Backup program installed to the computer.

As the author was testing the system, it was found, that using the JSite Manager the time spent on backing up and updating five different web sites was reduced by half. As it was tested without the JSite Manager, the time result of backing up and updating was 14 minutes 21 seconds. Nextly, the testing was done with the help of the system and the same work time result was 7 minutes 56 seconds. As it seems, with the help of the JSite Manager, a user is able to do more work with the same amount of time.

Also, one of the positive consequence is also the fact, that the system in being used in the company named Reaktiiv Grupp OÜ, where JSite Manager will be tested and then upgraded, to be more comfortable and effective.

This work consists of both- theoretical and practical part. On the theoretical part, the author introduces Joomla! content management system. Also it is shown, which data the developer should monitor and query the web server to keep Joomla! websites as secured as possible. On the second part, the author describes the created environment- JSite Manager, which asks the information about the web pages that are inserted into the system. It also helps to update them if needed.

This thesis is written in Estonian and it contains 31 text pages, 5 chapters, 11 diagrams and 0 tables.

## Lühendid

<b>Administrator (Joomla!)</b>	(eesti k. administraatori liides) Administreerimise programm, mida nimetatakse veel <i>back-end</i> , <i>Admin Panel</i> või <i>Control panel</i> . Tegu on kasutajaliidesega, kus administraatorid ja teised ligipääsu õigusega kasutajad saaksid kergesti hallata veebilehte. Administraatorliideses on võimalik hallata veebilehe sisu ja laiendeid. [3]
<b>Alias</b>	Alternatiivne nimi, mille abil on võimalik muuta veebiaadress veebilehitsjatele kergesti loetavaks. Näiteks asendatakse alternatiivsel nimel täpitähed, tühikute asemel kasutatakse sidekriipsu jne. Oletame, et tegu on veebiaadressiga, mille nimi on: Jõulupeo üritus (veebiaadress: “ <a href="https://www.example.ee/Jõulupeo_üritus">https://www.example.ee/Jõulupeo üritus</a> ”, siis pealkirjast loodud alias oleks “joulupeo-uritus” ja veebiaadress: <a href="https://www.example.ee/joulupeo-uritus">https://www.example.ee/joulupeo-uritus</a> . [13]
<b>Apache</b>	Üks enimlevinuid veebiserveri tarkvarasid. [15]
<b>Client side</b>	(eesti k. kliendipoolne) Nimetatakse veel <i>front-end</i> . Veebilehe kliendi poolne osa. [10]
<b>Components</b>	(eesti k. komponendid) Kõige suurema mahuga sisuhaldussüsteemi laiendused. Enamus komponentidel on kaks osa: kliendipoolne ja administraatorliidese poolne. Kasutatakse näiteks veebilehe sisu kuvamiseks. [5] Graafiline selgitus on näha Lisas 1.
<b>Content management system (CMS)</b>	(eesti k. sisuhaldussüsteem) Tarkvara programm, mis võimaldab kasutajatel hallata veebilehe sisu. [1]

<b>Extensions</b>	(eesti k. laiendused) Süsteemi laiendused. Joomla! Sisuhaldussüsteemil on kasutusel viite erinevat laienduse varianti: <i>Components</i> (eesti k. komponendid), <i>modules</i> (eesti k. moodulid), <i>plugins</i> (eesti k. pistikprogramm), <i>templates</i> (eesti k. mallid), <i>languages</i> (eesti k. keeled). [4]  Graafiline selgitus on näha Lisas 1.
<b>Global Configuration</b>	Joomla! kasutajaliideses asuv sisuhaldussüsteemi seadingu muutmiseks loodud sektsioon. [14]
<b>ISS 7</b>	Veebiserveri tarkvara. [16]
<b>L.A.M.P.</b>	Vabavaraline veebiarendus platvorm, mis kasutab Linuxi operatsioonisüsteemi, Apache veebiserverit, MySQL andmebaasi haldamise süsteemi ja PHP skriptimiskeelt. [31]
<b>Languages (Joomla!)</b>	(eesti k. keeled) Kõige põhilisem laiendus. Laiend koosneb keelevaliku muutujatest ning väärtusest, mille abil tõlgitakse Joomla! lähtekoodi. Keele pakid mõjutavad nii veebilehe kliendipoolset ja administraatori poolset osa. [9]
<b>Modules</b>	(eesti k. moodul) Väiksemad ja paindlikumad sisuhaldussüsteemi laiendused. Mooduleid tuntakse, kui “bokse”, mis paigutatakse ümber veebilehe sisu. Näiteks: sisselogimise vormi moodul. [6]  Graafiline selgitus on näha Lisas 1.
<b>MySQL</b>	Üks populaarseim tasuta kasutatavaid SQL andmebaasi haldamise süsteeme. [32]
<b>Plugin</b>	(eesti k. pistikprogramm) Laiendused, mis annavad põhiprogrammile lisavõimalusi. [7]  Graafiline selgitus on näha Lisas 1.
<b>Script</b>	(eesti k. skript) Arvutikeel, millega on võimalik faili abil käsklusi edastada. [2]

<b>Server side</b>	(eesti k. serveripoolne) Nimetatakse veel <i>back-end</i> . Serveripoolsed osad, mis käivitatakse enne HTML laadimist. [11]
<b>Template</b>	(eesti k. mall) Veebilehe kesta jaoks mõeldud failide kogum. Koosneb failidest, mis määravad veebilehe välimuse. [8] Graafiline selgitus on näha Lisas 1.
<b>URL</b>	Lühend sõnast <i>Uniform Resource Locator</i> . Tegu on aadressiga, mis sisestamisel kuvab kindlat veebiserveris olevat ressursi. Ressurssiks võib olla näiteks HTML veebileht, CSS dokument, pilt jne. [12]



# Sisukord

<b>Autorideklaratsioon</b>	<b>2</b>
<b>Annotatsioon</b>	<b>3</b>
<b>Abstract</b>	<b>4</b>
<b>Ensuring the Security of Joomla Web Pages</b>	<b>4</b>
<b>Lühendid</b>	<b>6</b>
<b>Jooniste loetelu</b>	<b>11</b>
<b>Sissejuhatus</b>	<b>12</b>
<b>1. Joomla sisuhaldussüsteemi eelistuse põhjus</b>	<b>14</b>
<b>2. Joomla sisuhaldussüsteem</b>	<b>15</b>
2.1 Joomla sisuhaldussüsteemist üldiselt	15
2.2 Joomla sisuhaldussüsteemi kasutamise näiteks	15
<b>3. Joomla sisuhaldussüsteemist päritav informatsioon</b>	<b>16</b>
3.1 Veebilehe administraatorliidese ligipääsu peitmine	16
3.2 Laiendused/Lisaprogrammid	16
3.3 Mallid	17
3.4 Joomla sisuhaldussüsteemi versioon	17
3.5 Search Engine Friendly	17
3.6 Kasutajate kontroll	18
3.7 Andmebaasis olevate tabelid	18
3.8 Veebiserveri poolsed kontrollid	18
3.8.1 SSL	18
3.8.2 Expose_php	19
3.8.3 Display_errors	19
3.8.4 Allow_url_fopen	19
3.8.5 Allow_url_include	19
3.8.6 File_uploads	19
3.8.7 PHP versioon	19
3.8.8 Upload_tmp_dir	19
3.9 Varundamine	20
<b>4. Lõputöö raames väljas arendatud veebikeskkond JSite Manager</b>	<b>21</b>
4.1 JSite Manager kirjeldus	21
4.2 JSite Manager veebikeskkonna ülesseehitus	21
4.2.1 JSite Manager komponent	22
4.3 Veebikeskkonnas kasutatud vahendid	29
4.3.1 HTML	29
4.3.2 Javascript	29

4.3.3 CSS	29
4.3.4 PHP	29
4.4 Kasutajade haldamine JSite Manager veebikeskkonnas	29
4.5 JSite Manager veebikeskkonna kasutamine	32
4.5.1 Kasutaja registreerimine	32
4.5.2 Listid	32
4.5.3 Listi detailvaade	34
4.6 JSite Manager edaspidi planeeritavad tööd	38
4.6.1 Üldine	38
4.6.2 Listi vaade	39
4.6.3 Listi detailvaade	39
4.6.4 Veebilehe lisamine	39
<b>5. Majanduslik efekt</b>	<b>40</b>
<b>Kokkuvõte</b>	<b>41</b>
<b>Kasutatud kirjandus</b>	<b>43</b>
<b>Lisa 1 - Joomla veebilehe laienduste plaan</b>	<b>47</b>
<b>Lisa 2 - JSite Manager veebikeskkonnas hallatavate veebilehtede pistikprogrammi faili jsitemanager.php skript.</b>	<b>48</b>
<b>Lisa 3 - JSite Manager veebikeskkonnas hallatavate veebilehtede pistikprogrammi jsitemanager.xml skript.</b>	<b>55</b>
<b>Lisa 4 - PHP fail, mis pärib JSite Manager keskkonda sisestatud veebilehelt informatsiooni.</b>	<b>56</b>

## Jooniste loetelu

Joonis 1. JSite Manageri veebikeskkonnas uue kasutaja registreerimine.	30
Joonis 2. JSite Manager veebikeskkonnas kasutaja parooli taastamine.	31
Joonis 3. JSite Manager veebikeskkonna listi vaade. Liste ei ole tekitatud.	32
Joonis 4. JSite Manager veebikeskkonna listi vaade. Kasutaja on sisesetatud süsteemi 3 listi.	33
Joonis 5. Uue listi loomine.	34
Joonis 6. Listi vaade. Veebilehti ei ole lisatud.	35
Joonis 7. Listi vaade. Veebileht on lisatud listi.	35
Joonis 8. Tuleks aktiveerida <i>front-end</i> varundamine.	36
Joonis 9. Tuleb sisestada <i>Secret word</i> väärtus, mis peab olema identne JSite Manageri sisestatud veebilehe Akeeba Key väärtusega.	36
Joonis 10. Uue veebilehe lisamine.	37
Joonis 11. PHP funktsioon, mis kutsub esile informatsiooni pärimise veebilehe pistikprogrammist. a) \$site - veebilehe nimi. Parameeter 1, mille alusel kontrollitakse, et päritud informatsioon salvestatakse andmebaasi tabelisse # <i>__jsite_manager_site_data</i> uuendatud informatsiooni; b) \$secret - JSite Manageris veebilehele väärtustatud salasõna, mis on võrdne pistikprogrammi " <i>Secret</i> " väärtusega; c) \$whitelist - veebilehe IP aadress, et veenduda, kas päring tehakse õigele veebiserverile; d) \$id - veebilehe id. Parameeter 2, mille alusel kontrollitakse, et päritud informatsioon salvestatakse andmebaasi tabelisse # <i>__jsite_manager_site_data</i> õige veebilehe informatsiooniks. e) \$returnIP - JSite Manageri veebiserveri IP aadress, veendumiseks, et saadatud informatsioon tuleb õigest veebiserverist.	38

## Sissejuhatus

Administraatori esmane ülesanne on tagada klientide veebilehtede turvalisus. Ettevõttes, kus autor töötab, on autori ülesandeks kliendi lahenduste haldamine. Sellest tuli lõputöö teema “Joomla veebilehtede turvalisuse tagamine” ning see ajendas looma veebikeskkonda, mille abil on kergem hallata mitmeid Joomla! sisuhaldussüsteemil töötavaid veebilehti.

Joomla! kohta on levinud mitmeid erinevaid eelarvamusi. Seda on väga kerge veebiserverisse paigaldada, samal ajal on nõudmised kasutatavale keskkonnale ja kasutajale suhteliselt madalad. Seetõttu on Joomla! kasutatavuselt üks maailma kõige populaarseim sisuhaldussüsteeme. Statistika põhjal on see maailmas kasutatavuse poolest esikolmikus. Lisaks Joomla!le on esikolmikus veel Drupal ja Wordpress. [17]

Tihti võib tekkida olukord, kus kasutaja paigaldab Joomla! ebaturvalisse veebiserverisse, kasutades samal ajal nõrkasid paroole ning aegunud lisaprogramme. See omakorda võib tekitada olukorra, kus Joomla! sisuhaldussüsteemil töötav veebileht võib sattuda turvaaukude tõttu rünnaku ohvriks. Kui hallata on mitmeid veebilehti, on raske kõiki lehti õigeaegselt uuendada, seega võidakse tihti unustada veebilehtede sisuhaldussüsteem uuendamata. Uuendamata veebilehtedel võivad ilmned turvaaugud, mis muudavad veebilehe ebaturvaliseks.

Antud probleemi lahendaks veebikeskkond, mis pärib reaajas informatsiooni süsteemi sisestatud veebilehtedelt. Reaktiiv Grupp OÜ'1 on tänase päeva seisuga valminud 200+ Joomla! sisuhaldussüsteemile loodud veebilehte. Lisaks tegeleb ettevõtte uute arenduste tegemise kõrvalt ka olemasolevate veebilehtede haldamise ja hooldusega. Haldamise tööd lihtsustaks ühtne süsteem, kuhu päritakse Joomla! sisuhaldussüsteemil töötavatest veebilehtedest informatsiooni. Selline veebikeskkond võimaldaks töötajal monitoorida Joomla! veebilehti. Lisaks annaks süsteem töötajale märku, kui saadaval on Joomla! sisuhaldussüsteemi uuendus ning milline versioon veebilehel on kasutuse. Arendajal on

võimalik pärast arenduse töö lõppu kontrollida, kas Joomla! veebileht on õigesti konfigureeritud.

Käesoleva lõputöö probleemi lahendus on rakendusliku iseloomuga ning loodud süsteem võetakse ettevõttes Reaktiiv Grupp OÜ kasutusele. Tänu sellele muutub veebilehete haldamine lihtsamaks ja kiiremaks, mis omakorda tõstab süsteemi sisestatud veebilehete turvalisust.

Antud töö koosneb nii teoreetilisest kui praktilisest osast. Teoreetilisest osast tutvustab autor Joomla! sisuhaldussüsteemi, toob välja, milliseid andmeid tuleks arendajal veebilehe serveri poolt jälgida ja pärida, et hoida Joomla! veebilehed turvatuna. Praktilisest osast loob autor veebikeskkonna JSite Manager. Veebikeskkonna loomiseks oli autorile antud ette L.A.M.P. platvorm, testimised ning meetodid töötas välja autor. Lõputöö teises pooles kirjeldab autor loodud keskkonda, mis pärib süsteemi sisestatud veebilehete kohta informatsiooni.

## 1. Joomla! sisuhaldussüsteemi eelistuse põhjus

On teada, et lisaks Joomla! sisuhaldussüsteemile on maailmas tänase päeva seisuga kasutuvuselt esikolmikus veel Drupal ja Wordpress. [17] Miks Reaktiiv Grupp OÜ eelistab Joomla! sisuhaldussüsteemi?

2017. aastal on võrreldes Drupali ja Wordpressiga, Joomla! ainuke sisuhaldussüsteem, millele on mitmekeelsuse tugi sisse ehitatud. Drupalil peab kasutaja mitmekeelsuse aktiveerimiseks paigaldada laienduse nimega “*Internationalization*” ning Wordpressil on samuti vaja paigaldada laiendus mitmekeelsuse aktiveerimiseks. Joomla! sisuhaldussüsteemis on vaja kõigest administraatorliideses aktiveerida mitmekeelsuse tugi ning seejärel saabki hakata konfigureerima veebilehe mitmekeelsust. [18]

Ettevõtte kompetentsus Joomla! arendamiseks on suurem, kui Wordpressi või Drupali puhul. Ettevõtte on teinud +200 veebilehte Joomla! sisuhaldussüsteemile. Ettevõttel on Joomla! arendamiseks loodud koodibaas, mis sisaldab nii erinevaid malle, komponente ning moduleid. Lisaks on ettevõttel Joomla! sisuhaldussüsteemi haldamiseks loodud õppevideote keskkond. Õppevideote keskkond on kasuks klientidele ja uutele töötajatele.

## **2. Joomla sisuhaldussüsteem**

### **2.1 Joomla sisuhaldussüsteemist üldiselt**

Joomla! sisuhaldussüsteem loodi 17. augustil 2005. Joomla! arenes välja veebiarendus projektist Mambo. [19] Tänapäevaks on Joomla! mitmeid auhindu võitnud sisuhaldussüsteem, mis võimaldab lihtsamalt ehitada veebilehti ning veebirakendusi. Kerge haldamise ja painduvuse tõttu on Joomla! üks maailma kõige populaarsematest sisuhaldussüsteemidest. [18]

### **2.2 Joomla sisuhaldussüsteemi kasutamise näiteks**

Joomla! sisuhaldussüsteemi kasutatakse erinevate veebilehtede arenduseks, olenemata veebilehe suurusest või ülesehitusest. Siin mõned näited:

- Ettevõtete veebilehed.
- Ettevõtete sisevõrgud.
- Online-ajakirjad (ajalehtede veebilehed jm).
- E-kaubanduse ja veebipõhised broneerimissüsteemid.
- Valitsustele loodud rakendused.
- Mittetulundusühingute veebilehed.
- Koolide, lasteaedade ja kirikute veebilehed.
- Personaalsed või perekonna veebilehed. [20]

### 3. Joomla sisuhaldussüsteemist päritav informatsioon

Käesolevas punktis autor kirjeldab, millist informatsiooni JSite Manager veebikeskkond hakkab veebilehtedelt pärima. Veebikeskkond on loodud vastavalt ettevõtte vajadustele, veebilehtedelt päritava informatsiooni list on koostatud ettevõtte arendajate poolt.

#### 3.1 Veebilehe administraatorliidese ligipääsu peitmine

Joomla! sisuhaldussüsteemil töötava veebilehe administreerimiseks on eraldi administraatorliidese keskkond, kus on võimalik veebilehte hallata. Joomla! sisuhaldussüsteemi installeerides on vaikimisi administraatorliidese aadressiks: "<https://www.example.com/administrator>" ehk administraatorliidesse pääsemiks on vaja veebiaadressi lõppu kirjutada "*administrator*" ning seejärel on võimalik pääseda ligi sisselogimise paneelini. Turvariski vähendamiseks oleks vaja administraatorliidese veebiaadress ära muuta. Joomla! versioon 3 puhul soovitab autor kasutada administraatorliidese peitmiseks näiteks AdminExile laiendust (<https://extensions.joomla.org/extension/adminexile/>). Laiendus võimaldab administraatorliidese veebiaadressile lisada salasõna aadressi lõppu. Näiteks lisades salasõnaks "detsember", ei ole administraatorliides veebiaadress enam "<http://www.example.com/administrator>", vaid hoopis "<http://www.example.com/administrator?detsember>".

#### 3.2 Laiendused/Lisaprogrammid

Joomla! veebileht koosneb erinevatest laiendustest: komponentidest, moodulitest ja pistikprogrammidest. Tihti võib tekkida olukord, kus varem lehel kasutusel olnud laiendusi enam ei kasutata. Mitte aktiivsed laiendused tuleks võimalusel alati veebilehelt eemaldada. Siiski jätavad töötajad vanad mitte aktiivsed laiendused veebilehele alles. Taoliselt laiendustel võivad aja jooksul ilmned turvaaukud. Seega tuleks veebilehtedelt eemaldada vanad ja ebaaktiivsed laiendused. Vastasel juhul võib veebilehel suurendada turvarisk.



### 3.3 Mallid

Joomla! kliendipoolsel lehel on kasutusel mallid, millega määratakse veebilehe stiilid ja ülesehitus. Vaikimisi on paigaldatud Joomla! sisuhaldussüsteemi *beez3* ja *protostar* mallid. Tavaliselt genereeritakse veebilehte luues vastavalt disainile uus mall. Lisaks võib tekkida olukord, kus aastate jooksul veebilehe välimus mitmeid kordi muutub. Sellisel juhul ei arendata uut veebilehte, vaid paigaldatakse sisuhaldussüsteemi uus mall. Vanad mallid, aga jäetakse veebilehele alles. Tark oleks sisuhaldussüsteemist eemaldada kõik mitte aktiivsed mallid (ka Joomla! sisuhaldussüsteemi poolt vaikimisi paigaldatud mallid).

Pärast igat Joomla! uuendust paigaldatakse vaikimisi paigaldatud mallid tagasi, seega tuleks pärast igat uuendust *beez3* ja *protostar* mallid ära kustutada.

### 3.4 Joomla sisuhaldussüsteemi versioon

Tähtis on hoida Joomla! sisuhaldussüsteemi uuendatud. Aasta jooksul väljastatakse mitmeid Joomla! uuendusi. Uuendusi tuleks teha regulaarselt. Lisaks sisuhaldussüsteemi uuendamisele tuleks uuendada regulaarselt ka Joomla! lisaprogramme.

### 3.5 Search Engine Friendly

*Search Engine Friendly* (eesti k. otsingumootori sõbralik), lühend SEF. Määrab, kas veebileht on inimese ja otsingumootorite jaoks loetav. Alates versioon 1.5 on Joomla! suutlik teisendama veebiaadressi erinevatesse formaatidesse. Kasutaja saab tänu “*alias*” väljale määrata veebiaadressi kuju. [21]

Toon näite “*Welcome to Joomla!*” aadressi genereerimist erinevate seadetega:

1) Kui SEF ei ole aktiveeritud:

“[http://www.example.com/index.php?option=com\\_content&view=article&id=1:welcome-to-joomla&catid=1:latest-news&Itemid=50](http://www.example.com/index.php?option=com_content&view=article&id=1:welcome-to-joomla&catid=1:latest-news&Itemid=50)”

2) Kui SEF on aktiveeritud:

“<http://www.example.com/the-news/1-latest-news/1-welcome-to-joomla>”

SEF on võimalik aktiveerida *Global Configuration* sektsiooni alt. Alates Joomla! versioonist 1.6 on SEF vaikimisi aktiveeritud. Kui SEF on aktiveeritud, siis otsingumootorid indekseerivad aadressis olevad sõnad ning veebileht on tulevikus otsingumootorites paremini leitav. [21]

### **3.6 Kasutajate kontroll**

Joomla! administraatorliideses on kasutajate haldamiseks eraldi sektsioon. Kasutajaid on võimalik määrata erinevatesse kasutajagruppidesse, lisaks on võimalik kasutajate informatsiooni muuta. Aeg-ajalt oleks kasulik kontrollida veebilehele loodud kasutajaid ning vajadusel eemaldada kasutajad, mis ei ole enam mõnda aega kasutusel olnud.

### **3.7 Andmebaasis olevate tabelid**

Paigaldades serverisse Joomla! sisuhaldussüsteemi, tekitab Joomla! veebiserveriga seotud andmebaasi sisuhaldussüsteemi toimimiseks vajalikud andmebaasi tabelid. Juurde paigaldatavate laiendustega võidakse andmebaasi tekitada uusi tabeleid, mis hiljem laienduste eemaldamisel jäävad andmebaasi alles. Seetõttu oleks kasulik aeg-ajalt kontrollida veebilehega seotud andmebaasi tabeleid ning vajadusel eemaldada tabelid, mis ei ole enam kasutusel.

### **3.8 Veebiserveri poolsed kontrollid**

#### **3.8.1 SSL**

SSL sertifikaat tagab külastajale ja veebilehele turvalise ühenduse. Lisaks kontrollib, et külastatav veebileht oleks see, kuhu külastaja soovis jõuda. Turvaline ühendus muudab veebilehe ja külastaja vahelised päringud SSL sertifikaadi abil krüpteerituks. Sisuliselt on nii veebilehe kuvamine kui kõik seal tehtavad tegevused päringud. Näiteks järgmisele sisulehele klõpsamine ei ole midagi muud kui veebilehelt info pärimine. Kui ühendus veebilehe ja selle külastaja vahel ei ole krüpteeritud, tehakse kõik päringud

avalikult ning veebilehe ründajad saavad tahtmise korral seda suhtlust pealt kuulata.  
[22]

### **3.8.2 Expose\_php**

(eesti k. paljasta\_php) väärtus määrab, kas PHP informatsioon on avalikustatud serverist saadetava vastuse päise informatsioonis või mitte. [23]

### **3.8.3 Display\_errors**

(eesti k. kuva\_vigu) määrab, kas veebilehega seotud viga kuvatakse kliendipoolsel veebilehel. [23]

### **3.8.4 Allow\_url\_fopen**

PHP parameeter. "Url\_fopen" määrab, kas süsteem lubab PHP abil kaasata faile süsteemivälistest allikatest. Kui "allow\_url\_fopen" on serveris aktiveeritud, toob see kaasa turvariski kõikidele domeenidele, mis on majutatud antud serveris. Seetõttu tuleks kontrollida, et "allow\_url\_fopen" oleks veebiserveris deaktiveeritud. [24]

### **3.8.5 Allow\_url\_include**

Määrab, kas veebiserveris on võimaldatud failides teadlikult teiste failide kaasamine samast serverist, kasutades järgmisi PHP funktsioone: include, include\_one, require, require\_once. [25]

### **3.8.6 File\_uploads**

Määrab, kas veebiserveris on failide üleslaadimine võimaldatud läbi veebikeskkonna.  
[23]

### **3.8.7 PHP versioon**

Serverisse paigaldatud PHP versioon. PHP versiooni tuleks veebiserveris samuti aeg-ajalt vastavalt vajadusele uuendada.

### **3.8.8 Upload\_tmp\_dir**

Määrab asukoha, kuhu veebikeskkonnas üleslaetud failid salvestatakse. [26]

### **3.9 Varundamine**

Joomla! sisuhaldussüsteemi varundamiseks on laiendus nimega Akeeba Backup (<https://www.akeebabackup.com/>). Veebilehti tuleks varundada regulaarselt. Lisaks regulaarsele varundamisele tuleks veebilehti varundada enne sisuhaldussüsteemi versiooni uuendamist või muudatuste tegemist.

## 4. Lõputöö raames väljas arendatud veebikeskkond JSite Manager

### 4.1 JSite Manager kirjeldus

Lõputöö raames loodi veebikeskkond JSite Manager, millega on võimalik Joomla! sisuhaldussüsteemil töötavaid veebilehti kontrollida, varundada. Lisaks viitab süsteem haldajat võimalusel uuendama veebilehte. Veebilehtesid on võimalik süsteemis varundada, eeldusel, et veebileht on korrektselt süsteemi konfigureeritud ja veebilehele on paigaldatud Akeeba Backup lisaprogramm. JSite Manager veebikeskkonna kaudu teostatud varundamisel salvestatakse varundamisega loodud failid varundatud veebilehe serverisse.

### 4.2 JSite Manager veebikeskkonna ülesseehitus

Veebiserverisse paigaldati Joomla! sisuhaldussüsteem versioon 3.8.2. Seejärel autor arendas ja paigaldas süsteemi uue malli nimega “jsitemanager”. Mallides asuvad järgmised kaustad/failid:

- **Classes (kaust)**

Kaustas olevatesse failidesse kirjutatakse PHP funktsioonid, mis on kasutusel malli struktuuri ülesehitusel.

- **CSS (kaust)**

Kausta on tekitatud CSS failid. Failidesse on kirjutatud CSS reeglid, mis on JSite Manager veebikeskkonnas kasutusel.

- **Fonts (kaust)**

Kaustas hoitakse kõiki JSite Manageriga seotud fontide faile.

- **HTML (kaust)**

Kaust on tekitatud Joomla! sisuhaldussüsteemi poolt loodud failide ümberkirjutamise jaoks. Kui muuta Joomla! sisuhaldussüsteemi tuuma faile, siis pärast sisuhaldussüsteemi uuendamist kirjutatakse tuuma failid igakord üle ning arendaja poolt tehtud muudatused lähevad kaduma. Seetõttu on tekitatud HTML kaust, kuhu lisatakse kõik muudetud failid ning seda kausta Joomla uuendusega

üle ei kirjutata. Joomla! sisuhaldussüsteem kontrollib, kas HTML kaustas on ümber kirjutatud faile või mitte. Kui ümber kirjutatud fail on olemas, siis kasutab süsteem ümber kirjutatud faili. Kui süsteem ei tuvasta ümber kirjutatud faili, siis kasutatakse tuuma faili.

- **Images (kaust)**

Kausta lisatakse kõik pildid, mis on veebilehe välimusega seotud.

- **Js (kaust)**

Kaustas asuvad Javascripti failid. Failidesse on kirjutatud Javascripti reeglid, mis on kasutusel malli struktuuris.

- **Favicon.ico (fail)**

Otsetee ikoon, tavaliselt 16x16 või 32x32 piksli suurune ikoon, mis on seotud konkreetse veebilehe disainiga.

- **Index.php (fail)**

Pannakse paika veebilehe üldine struktuur: üldine HTML loogika, kus hakkavad paiknema templateDetails.xml väärtustatud moodulite positsioonid, kuhu tuleb veebilehe sisu.

- **templateDetails.xml (fail)**

Määratakse malliga seotud parameetrid ning tekitatakse moodulite tarvis positsioonide nimed.

- **Vars.php (fail)**

Index.php aitamiseks loodud fail, kus määratakse vajalikud muutujad ning imporditakse vajalikud CSS ja Javascripti failid. Faili abil on võimalik hoida index.php puhta ja lühikesena. Vars.php fail imporditakse index.php failis. Seega kõik väärtustatud muutujad ning imporditud failid on kasutusel index.php failis.

#### 4.2.1 JSite Manager komponent

JSite Manageri andmebaaside tabelite ja vajalike vaadete genereerimiseks kasutati Component Creator (eesti k. komponendi generaator) veebikeskkonda: <https://www.component-creator.com/en/>. Component Creator võimaldab kergesti Joomla! sisuhaldussüsteemile genereerida komponente.

Component Creator abil loodud andmebaasi struktuur:

1. #**\_\_jsite\_manager\_contact**

Süsteemi sisestatud veebilehele on võimalik määrata seotud kontaktisik. Uusi kontakte saab lisada vaid sisseloginud kasutaja.

Andmebaasi tabeli struktuur:

- a. **ID** - unikaalne väli, mis annab igale reale unikaalse väärtuse.
- b. **STATE** - andmebaasis oleva rea staatus.
- c. **MODIFIED\_BY** - viimati muudatuse teinud kasutaja ID.  
Kasutaja ID võetakse automaatselt vastavalt sellele, mis kasutajaga on sisse logitud ja muudatust tehakse.  
Kasutaja informatsiooni hoitakse #**\_\_contact\_details** andmebaasi tabelis ehk **MODIFIED\_BY** määratakse kasutaja ID #**\_\_contact\_details** andmebaasi tabelist.
- d. **CREATED\_BY** - kontakti loonud kasutaja ID.  
Kasutaja ID võetakse automaatselt vastavalt sellele, mis kasutajaga on sisse logitud ja muudatust tehakse.  
Kasutaja informatsiooni hoitakse #**\_\_contact\_details** andmebaasi tabelis ehk **CREATED\_BY** määratakse kasutaja ID #**\_\_contact\_details** andmebaasi tabelist.
- e. **LIST\_ID** - kontakti luues tuleb määrata, mis listidega kontakt seotud hakkab olema. Kasutaja saab kontaktile lisada ainult nendesse listidesse, millele tal endal on ligipääs. Listi ID määratakse vastavalt selle, mis list momendil valitud on, kui vajutatakse *site* lisamise nuppu ehk **LIST\_ID**'ks määratakse valitud listi ID #**\_\_jsite\_manager\_list** andmebaasi tabelist.
- f. **NAME** - sisestatud kontakti nimi (kohustuslik väli).
- g. **EMAIL** - sisestatud kontakti e-mail (kohustuslik väli).
- h. **PHONE** - sisestatud kontakti telefoni number (kohustuslik väli).
- i. **SKYPE** - sisestatud kontakti Skype.

j. **COMPANY\_NAME** - sisestatud kasutaja ettevõtte nimi.

## 2. #\_jsite\_manager\_developer

Süsteemi sisestatud veebilehele on võimalik sarnaselt kontaktiga võimalik määrata veebilehega seotud arendaja. Uusi arendajaid saab lisada vaid sisseloginud kasutaja.

Kasutusel olev andmebaasi tabeli struktuur:

a. **ID** - unikaalne väli, mis annab igale reale unikaalse väärtuse.

b. **STATE** - andmebaasis oleva rea staatus.

c. **MODIFIED\_BY** - viimati muudatuse teinud kasutaja ID.

Kasutaja ID võetakse automaatselt vastavalt sellele, mis kasutajaga on sisse logitud ja muudatust tehakse.

Kasutaja informatsiooni hoitakse #\_contact\_details andmebaasi tabelis ehk MODIFIED\_BY määratakse kasutaja ID #\_contact\_details andmebaasi tabelist.

d. **CREATED\_BY** - kontakti loonud kasutaja ID.

Kasutaja ID võetakse automaatselt vastavalt sellele, mis kasutajaga on sisse logitud ja muudatust tehakse.

Kasutaja informatsiooni hoitakse #\_contact\_details andmebaasi tabelis ehk CREATED\_BY määratakse kasutaja ID #\_contact\_details andmebaasi tabelist.

e. **LIST\_ID** - arendajat luues tuleb määrata, mis listidega arendaja seotud hakkab olema. Kasutaja saab arendajaid lisada ainult nendesse listidesse, millele tal endal on ligipääs. Listi ID määratakse vastavalt selle, mis list momendil valitud on, kui vajutatakse site lisamise nuppu ehk **LIST\_ID**'ks määratakse valitud listi ID #\_jsite\_manager\_list andmebaasi tabelist.

f. **NAME** - sisestatud arendaja nimi (kohustuslik väli).

g. **EMAIL** - sisestatud arendaja e-mail (kohustuslik väli).

h. **PHONE** - sisestatud arendaja telefoni number (kohustuslik väli).

i. **SKYPE** - sisestatud arendaja Skype.



- j. **DEVELOPER\_STATUS** - sisestatud arendaja staatus - töötab ettevõttes, ei tööta ettevõttes.

### 3. #\_\_jsite\_manager\_list

Kasutaja poolt loodud listid, kuhu on võimalik lisada hallatavaid Joomla! veebilehti. Erinevad listid tekitavad võimaluse hallata erinevate klientidega seotud veebilehti eraldi.

Kasutusel olev andmebaasi tabeli struktuur:

- a. **ID** - unikaalne väli, mis annab igale reale unikaalse väärtuse.
- b. **STATE** - andmebaasis oleva rea staatus.
- c. **USER\_ID** - listi loonud kasutaja ID.

Kasutaja ID võetakse automaatselt vastavalt sellele, millise kasutajaga on sisse logitud ning muudatust tehakse.

Kasutaja informatsiooni hoitakse #\_\_*contact\_details* andmebaasi tabelis ehk **USER\_ID** määratakse kasutaja ID #\_\_*contact\_details* andmebaasi tabelist.

- d. **NAME** - loodud listi nimi (kohustuslik).
- e. **DESCRIPTION** - loodud listi kirjeldus.

### 4. #\_\_jsite\_manager\_site

Kasutaja poolt süsteemi sisestatud Joomla! sisuhaldussüsteemil töötavad veebilehed. Veebileht peab kuuluma listi alla.

Kasutusel olev andmebaasi tabeli struktuur:

- a. **ID** - unikaalne väli, mis annab igale reale unikaalse väärtuse.
- b. **STATE** - andmebaasis oleva rea staatus.
- c. **LIST\_ID** - listi ID, millega sisestatud veebileht seotud.

Listi ID määratakse vastavalt sellele, mis list momendil valitud on, kui vajutatakse *site* lisamise nuppu ehk **LIST\_ID**'ks määratakse valitud listi ID #\_\_*jsite\_manager\_list* andmebaasi tabelist.

- d. **URL** - sisestatud veebilehe aadress (kohustuslik).

- e. **DEVELOPMENT\_DATE** - sisestatud veebilehe arenduse kuupäev.
- f. **CONTACT\_ID** - veebilehega seotud kontakti ID.  
Valitud kontakti ID võetakse vastavalt valikule **#\_jsite\_manager\_contact** andmebaasi tabelist ehk **CONTACT\_ID**'ks määratakse valitud kontakti ID **#\_jsite\_manager\_contact** andmebaasi tabelist.
- g. **DEVELOPER\_ID** - veebilehega seotud arendaja ID.  
Valitud arendaja ID võetakse vastavalt valikule **#\_jsite\_manager\_developer** andmebaasi tabelist ehk **DEVELOPER\_ID**'ks määratakse valitud arendaja ID **#\_jsite\_manager\_developer** andmebaasi tabelist.
- h. **SITE\_NAME** - sisestatud veebilehe nimi (kohustuslik)
- i. **MOBILE** - hoitakse väärtust - kas mobiiliversiooni tugi on olemas või mitte.
- j. **AKEEBA** - hoitakse väärtust - kas veebilehele on paigaldatud Akeeba Backup komponent.
- k. **AKEEBA\_KEY** - hoitakse Akeeba Backup komponendist genereeritud **akeeba\_key** väärtust, mille abil on võimalik JSite Manageri abil teostada veebilehe varundamist.
- l. **ADMIN\_URL** - aadress, millega on võimalik veebilehe kasutajaliidesesse sisse logida. Tänu sellele tekib süsteemi link, millele vajutades avatakse brauseri uues aknas veebilehe kasutajaliidesesse logimise link.
- m. **LAST\_BACKUP** - hoitakse kuupäeva, millal on viimast korda tehtud JSite Manageri kaudu veebilehele Akeeba Backupiga varundamine.
- n. **SECRET** - veebilehega seotud salasõna.  
Salasõna peab kattuma sisestatud veebilehele paigaldatud JSite Manageri pluginas. Tänu sellele on võimalik pärida veebilehe kohta informatsiooni **#\_jsite\_manager\_site\_data** andmebaasi tabelisse.
- o. **WHITLIST\_IP** - hoitakse sisestatud veebilehe IP'd. Kontrollib, kas sisestatud veebilehe JSite Manageri saadetud informatsioon tuleb õigest

kohast (päritakse õigelt veebilehelt).

#### 5. #\_\_jsite\_manager\_site\_data

Süsteemi sisestatud veebilehele genereeritakse automaatselt rida #\_\_jsite\_manager\_site\_data andmebaasi tabelisse. Sellesse tabelisse hakatakse pärima informatsiooni sisestatud veebilehe serveri ja Joomla! konfiguratsiooni kohta. Päritava informatsiooni kokkuvõte on leitav peatükis 3.

Kasutusel olev andmebaasi tabeli struktuur:

- a. **ID** - unikaalne väli, mis annab igale reale unikaalse väärtuse.
- b. **ADMIN\_FOLDER** - väärtus, kas administraatori kaust on salasõna abil peidetud.
- c. **SITE\_ID** - site ID, millega päritud site informatsioon seotud.  
Site ID määratakse vastavalt selle, mis site kohta informatsiooni päritakse ehk **SITE\_ID**'ks määratakse valitud listi ID #\_\_jsite\_manager\_site andmebaasi tabelist.
- d. **COMPONENTS\_JSON** - veebilehele paigaldatud moodulite ülevaade, JSON formaadis.
- e. **MODULES\_JSON** - veebilehele paigaldatud moodulite ülevaade, JSON formaadis.
- f. **PACKAGE\_JSON** - veebilehele paigaldatud pakside ülevaade, JSON formaadis.
- g. **PLUGINS\_JSON** - veebilehele paigaldatud pluginate ülevaade, JSON formaadis.
- h. **TEMPLATES\_JSON** - veebilehele paigaldatud mallide ülevaade, JSON formaadis.
- i. **JOOMLA\_VERSION** - veebilehele paigaldatud Joomla versioon.
- j. **SEARCH\_ENGINE\_FRIENDLY** - väärtus, kas SEF on aktiveeritud või mitte.
- k. **USERS\_JSON** - veebilehele lisatud kasutate informatsioon, JSON formaadis.
- l. **SSL** - kas lehele on paigaldatud SSL sertifikaat.

- m. **DATABASE\_JSON** - veebilehe andmebaasi kokkuvõte, JSON formaadis.
- n. **EXPOSE\_PHP** - väärtus, kas sisse või välja lülitatud.
- o. **DISPLAY\_ERRORS** - väärtus, kas sisse või välja lülitatud.
- p. **ALLOW\_URL\_FOPEN** - väärtus, kas sisse või välja lülitatud.
- q. **ALLOW\_URL\_INCLUDE** - väärtus, kas sisse või välja lülitatud.
- r. **FILE\_UPLOADS** - väärtus, kas sisse või välja lülitatud.
- s. **PHP\_VERSION** - lehele paigaldatud PHP Versioon.
- t. **UPLOAD\_TMP\_DIR** - veebilehe tmp kausta asukoht.

Komponendiga loodi veebiserveris olevasse Joomla! “components” kausta uus alamkaust nimega “com\_site\_manager”, milles asuvad järgmised failid/kaustad:

- **Assets (kaust)**  
Hoitakse komponendiga seotud CSS ja Javascript faile.
- **Controllers (kaust)**  
Vaadete meetodfunktsioonid.
- **Files (kaust)**  
Pisiprogrammi failid, mille peab paigaldama JSite Manageri sisestatud veebilehele. Tänu paigaldatud pistikprogrammile saab JSite Manager pärida informatsiooni kodulehtede kohta. Paigaldatav pistikprogramm koosneb kahest failist: jsitemanager.php (skript lisas 2) ja jsitemanager.xml (skript lisas 3).
- **Helpers (kaust)**  
Kaustas olevatesse failidesse kirjutatakse PHP funktsioonid, mis on komponendi töö jaoks vajalikud.
- **Modules (kaust)**  
Päritakse vaadete sisud ning tehakse vajalikud andmebaasi päringud.
- **Views (kaust)**  
Pannakse kokku vajalikud vaated. Vaadetesse tulev sisu päritakse modules kaustast vastavalt vaatele.
- **Controller.php (fail)**  
Meetod vaadete näitamiseks.

- **Jsite\_manager.php (fail)**  
Fail, mis käivitab kontrollereid.
- **Router.php (fail)**  
Paneb kokku komponendiga seotud veebiaadressid.

## 4.3 Veebikeskkonnas kasutatud vahendid

### 4.3.1 HTML

Lühend sõnast Hypertext Markup Language, eesti keeles võib tõlkida kui Hüpertexti Märkimise Keel. Aitab määrata veebilehe üldise struktuuri, võimaldab ehitada HTML lehe blokkidena.

Lisaks on võimalik HTML abil luua pealkirjasid, paragraafe, tabelleid jne. [27]

### 4.3.2 Javascript

Tihti kasutatakse lühendina JS. Kasutatakse peamiselt veebilehtedel skriptimiseks. Saab kasutada lihtsamate efektide tegemiseks veebilehel. Kasutatakse *client-side* skriptimiskeelena. [28]

### 4.3.3 CSS

Lühend sõnast Cascading Style Sheets. CSS kirjeldab, kuidas HTML elemendid ekraanil paigutatakse. [29]

### 4.3.4 PHP

Lühend sõnast PHP: *Hypertext Preprocessor*. Kasutatakse *server-side* skriptimiskeelena. PHP skriptide tööks on vaja PHP toega serverit. [30]

## 4.4 Kasutajade haldamine JSite Manager veebikeskkonnas

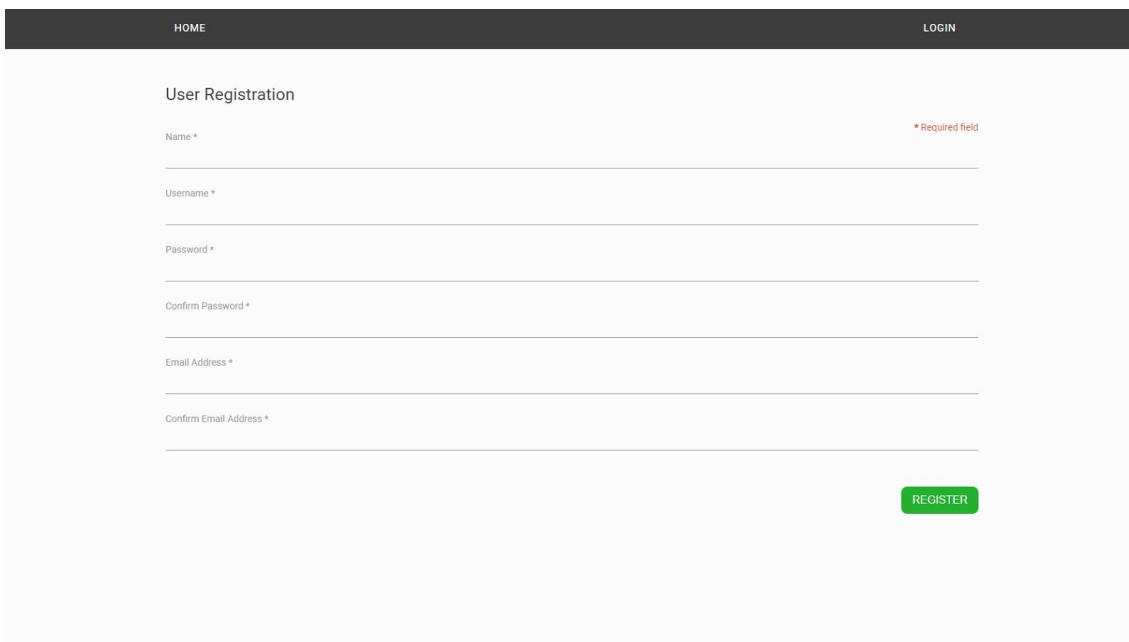
Kasutajate jaoks kasutatakse Joomla! sisuhaldusega vaikimisi paigaldatud Joomla! sisuhaldussüsteemi kasutajate sektsiooni. See võimaldab kergesti tekitada front-end'is registreerimise ja sisselogimise vaated.

Joomla! kasutajad liigitatakse gruppidesse. JSite Manageri veebikeskkonna jaoks tekitati kaks kasutaja gruppi: 1) Super Users; 2) Manager.

- 1) **Super Users** - kõik õigused. On võimalik logida *back-endi*, hallata kõiki kasutajaid ning seksioone.
- 2) **Manager** - piiratud õigused. Õigus kasutada ainult JSite Manageri *front-end* võimalusi.

JSite Manager registreerides peab uus kasutaja täitma järgmised väljad:

- **Name** - kasutaja päris nimi (kohustuslik).
- **Username** - kasutajanimi, millega hakkab JSite Manageri süsteemi sisse logima (kohustuslik).
- **Password** - parool, millega hakkab JSite Manageri süsteemi sisse logima (kohustuslik).
- **E-maili address** - e-maili aadress, millega loodud kasutaja seotud on - kinnitus e-mail tuleb sisestatud e-maili aadressile. Lisaks on võimalik e-maili abil unustatud salasõna taastada. E-maili sisestamine on kohustuslik.



The screenshot shows a web interface for user registration. At the top, there is a dark navigation bar with 'HOME' on the left and 'LOGIN' on the right. Below this, the main content area is titled 'User Registration'. It contains a form with the following fields, each with a red asterisk indicating it is required: 'Name \*', 'Username \*', 'Password \*', 'Confirm Password \*', 'Email Address \*', and 'Confirm Email Address \*'. A green 'REGISTER' button is located at the bottom right of the form.

Joonis 1. JSite Manageri veebikeskkonnas uue kasutaja registreerimine.

Please enter the email address for your account. A verification code will be sent to you. Once you have received the verification code, you will be able to choose a new password for your account.

Email Address \*

---

SUBMIT

Joonis 2. JSite Manager veebikeskkonnas kasutaja parooli taastamine.

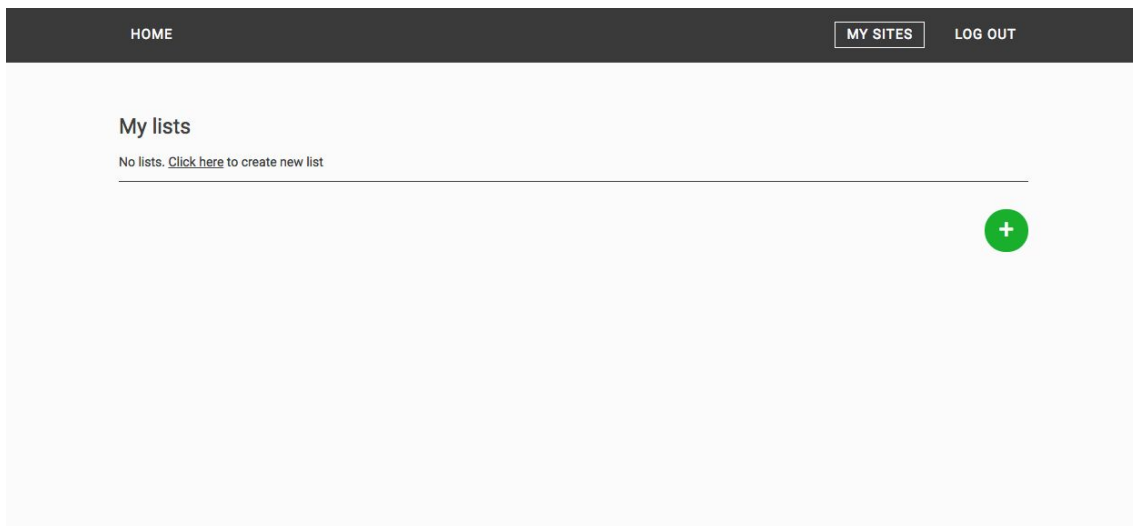
## 4.5 JSite Manager veebikeskkonna kasutamine

### 4.5.1 Kasutaja registreerimine

JSite Manageri kasutades tuleb luua kasutaja. Seejärel tuleb kontrollida registreerimisel sisestatud e-maili sissetulnud kirju ning aktiveerida e-mailile saadetud aadressi abil loodud kasutaja. Pärast konto aktiveerimist on võimalik sisse logida JSite Manager kasutajasse. Pärast sisselogimist suunatakse kasutaja automaatselt “*My lists*” vaatesse.

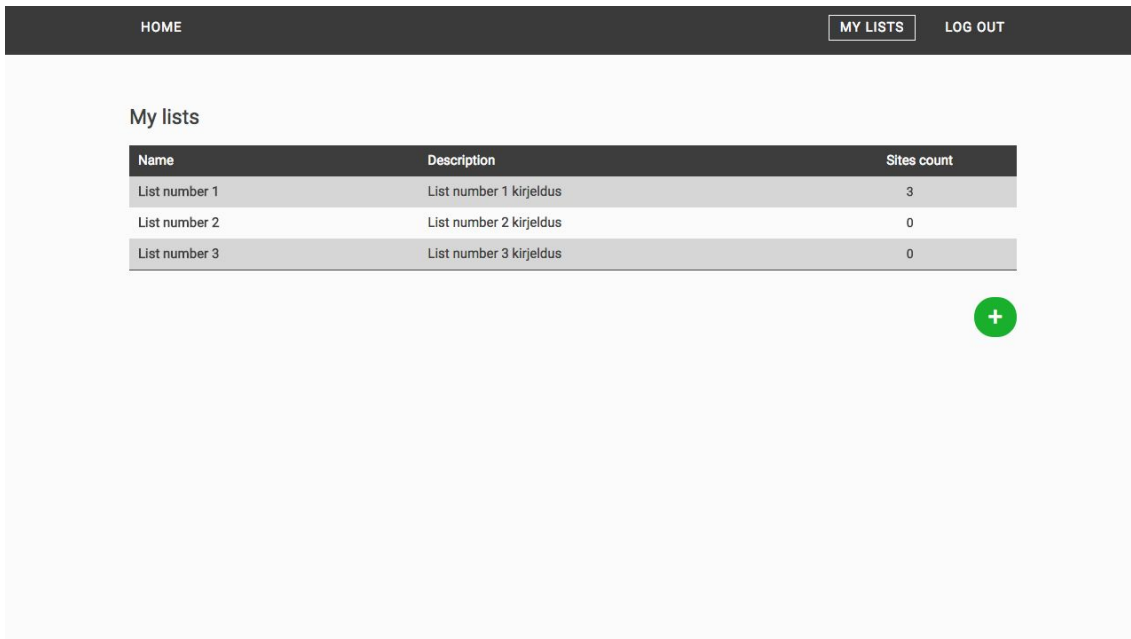
### 4.5.2 Listid

Kui loodud liste ei ole, siis on vaatesse kuvatud tekst: *No lists. Click here to create new list.* Listide loomine on võimalik, kui vajutada antud tekstile. Lisaks on uue listi loomiseks paremal all roheline nupp, mis võimaldab samuti lisada uut listi.



Joonis 3. JSite Manager veebikeskkonna listi vaade. Liste ei ole tekitatud.





Joonis 4. JSite Manager veebikeskkonna listi vaade. Kasutaja on sisesatatud süsteemi 3 listi.

Listi vaate jaoks kasutatakse Component Creatori poolt genereeritud vaadet nimega managerlist. Vaate sisu jaoks tehakse andmebaasi päring. `#__jsite_manager_list` tabelist päritakse kõik andmebaasitabeli read, mille `status = 1` (staatus on aktiivne) ja `user_id` (listi loonud kasutaja ID) on võrdne sisselõigatud kasutaja ID'ga.

Listide tabelis on 3 veergu: a) **Name** - listi nimi; b) **Description** - listi kirjeldus; c) **Sites count** - näitab, mitu veebilehte on listi lisatud. Kogu tabeli rida on klikitav ning vajutades tabeli reale suunatakse kasutaja listi vaatesse.

HOME MY SITES LOG OUT

## Add new list item

Name \*

Name

Description

Description

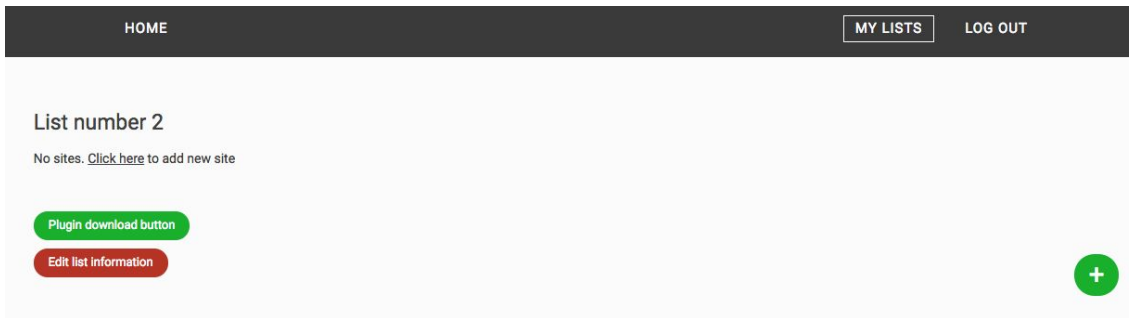
SUBMIT

Joonis 5. Uue listi loomine.

Uue listi loomisel kasutatakse Component Creatori poolt genereeritud vaadet nimega: managerlistform. Vaade koosneb HTML vormist, kuhu saab sisestada listi nime ja kirjelduse.

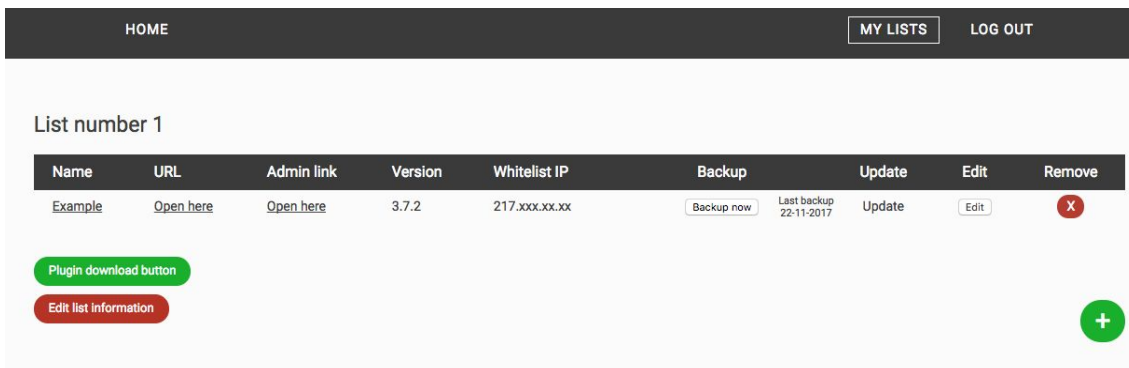
#### 4.5.3 Listi detailvaade

Listivaates on välja toodud pealkiri, nupp “*Plugin download button*”, mida kasutaja saab paigaldada listi lisatud veebilehtede sisuhaldussüsteemi. Veel on nupp “*Edit list information*”, millele vajutades avaneb listi muutmise vaade. Listi vaate sisu kuvamiseks kasutatakse Component Creatori poolt genereeritud vaadet nimega managerlist. Vaate sisu jaoks tehakse andmebaasi päring. #*\_jsite\_manager\_site* tabelist päritakse kõik andmebaasitabeli read, mille *status = 1* (staatus on aktiivne) ja *list\_id* (listi ID, kuhu veebileht on lisatud) on võrdne lahti oleva listiga. Enne listi avamist kontrollib süsteem, kas sisseloginud kasutajal on õigused listi avada või mitte. Kui kasutajal puuduvad õigused, suunatakse kasutaja automaatselt avalehele.



Joonis 6. Listi vaade. Veebilehti ei ole lisatud.

Uue veebilehe lisamiseks on võimalik vajutada “*Click here to add new site*” või kasutada all paremal rohelist nuppu.



Joonis 7. Listi vaade. Veebileht on lisatud listi.

Veebilehtede tabelis on 9 veergu: a) **Name** - veebilehe nimi, millele vajutades suunatakse kasutaja veebilehe informatsiooni detailvaatesse; b) **URL** - veebilehe aadress, millele vajutades avatakse sisestatud veebileht brauseri uues aknas; c) **Admin link** - veebilehe administraatorliidese link, millele vajutades avatakse administraatorliidese link brauseris uues aknas; d) **Version** - veebilehe praegune versioon; e) **Whitelist IP** - sisestatud veebilehe IP aadress, mille abil päritakse veebilehelt informatsiooni (turvalisuse eesmärgil); f) **Backup** - Akeeba Backup laiendusega koostöös töötav nupp. Vajutades *backup* nupule, tehakse veebilehe varundamine. Varundatud fail säilitatakse veebilehe serverisse, mitte JSite Manageri serverisse. Kui JSite Manageri kaudu on tehtud varundamine, siis säilitakse tabelisse kuupäev, millal viimane varundamine veebilehele tehtud sai; g) **Update** - kui veebilehe ei ole paigaldatud viimane Joomla! sisuhaldussüsteemi versioon, siis tekib tabelisse

nupp “*update*” (eesti k. uuenda). Vajutades “*update*” nuppu, avatakse hüpikaknas kodulehe veebilehe uuendamise alamleht, kus kõigepealt tuleb veebilehe *back-end* kasutajaga sisse logida ning seejärel hüpikaknas vajutada uuendamise nuppu; h) **Edit** - veebilehe informatsiooni muutmise nupp; i) **Remove** - veebilehe eemaldamine listist.

### Uue veebilehe lisamine listi

Uue veebilehe lisamisel tuleb täita järgmised väljad:

- **Site Name** - veebilehe nimi (kohustuslik).
- **URL** - veebilehe aadress (kohustuslik).
- **Administrator URL** - veebilehe administraatorliidese aadress (kohustuslik).
- **Development Date** - veebilehe arenduse kuupäev.
- **Contact** - veebilehega seotud kontakt.
- **Developer** - veebilehega seotud arendaja.
- **Warranty** - näitab, kas veebilehel on garantii (kui on valitud “*yes*”, ilmub kuupäeva lisamine, millal garantii algab ja millal lõpeb).
- **Mobile** - näitab, kas mobiiliversioon on veebilehest välja arendatud.
- **Akeeba** - näitab, kas veebileht toetab Akeeba Backup laiendust (kui on valitud “*yes*”, ilmub Akeeba Key lisamise võimalus. Akeeba Key peab olema identne väärtusega, mis lisatakse sisestatava veebilehe Akeeba Backup laienduse seadete all “*Secret word*” väärtuseks.

“*Secret word*” saab määrata: *Components* -> *Akeeba Backup* -> *Options* -> *Front-end backup*.

Enable front-end and remote backup

 No  Yes

Joonis 8. Tuleks aktiveerida *front-end* varundamine.

Secret word

Joonis 9. Tuleb sisestada *Secret word* väärtus, mis peab olema identne JSite Manageri sisestatud veebilehe Akeeba Key väärtusega.

- **Whitelist IP** - sisestatava veebilehe IP aadress, mille abil JSite Manager kontrollib, kas andmete päring tehakse õige veebilehele. Veebilehe IP aadressi saamiseks võib kasutada näiteks:  
<https://www.site24x7.com/find-ip-address-of-web-site.html>

The screenshot shows a web application interface with a dark header containing 'HOME', 'MY LISTS', and 'LOG OUT'. The main content area is titled 'Add' and contains a form with the following fields:

- Site Name \* (text input)
- Url \* (text input)
- Administrator URL \* (text input)
- Development Date (text input with value '2017-01-11' and a calendar icon)
- Contact (dropdown menu with 'Select an option')
- Developer (dropdown menu with 'Select an option')
- Warranty (dropdown menu with 'No')
- Mobile (dropdown menu with 'No')
- Akeeba (dropdown menu with 'No')
- Whitelist IP (text input)

A green 'SUBMIT' button is located at the bottom right of the form.

Joonis 10. Uue veebilehe lisamine.

Kui kasutaja siseneb listi, siis süsteem pärib automaatselt kõikide veebilehtede informatsiooni eeldusel, et pistikprogramm on veebilehtedele lisatud ning veebileht korrektselt JSite Manageri lisatud. Listi vaatesse sisenedes käivitatakse funktsioon, mis käivitab PHP faili, mille abil saadetakse päring veebilehtedele paigaldatud pistikprogrammi pihta.

```
getDataHelper::getData($site, $secret, $whitelist, $id, $returnIP  
$_SERVER['REMOTE_ADDR']);
```

Joonis 11. PHP funktsioon, mis kutsub esile informatsiooni pärimise veebilehe pistikprogrammist. a) \$site - veebilehe nimi. Parameeter 1, mille alusel kontrollitakse, et päritud informatsioon salvestatakse andmebaasi tabelisse `#__jsite_manager_site_data` uuendatud informatsiooni; b) \$secret - JSite Manageris veebilehele väärtustatud salasõna, mis on võrdne pistikprogrammi “*Secret*” väärtusega; c) \$whitelist - veebilehe IP aadress, et veenduda, kas päring tehakse õigele veebiserverile; d) \$id - veebilehe id. Parameeter 2, mille alusel kontrollitakse, et päritud informatsioon salvestatakse andmebaasi tabelisse `#__jsite_manager_site_data` õige veebilehe informatsiooniks. e) \$returnIP - JSite Manageri veebiserveri IP aadress, veendumiseks, et saadetud informatsioon tuleb õigest veebiserverist.

Seejärel käivitatakse PHP skript, mis teeb päringu listis oleva/olevate veebilehtede paigaldatud pistikprogrammi pihta (Lisas 4).

Peale andmete pärimist töödeldakse need ning kontrollitakse, et andmed oleksid õiges formaadis ning seejärel salvestatakse `#__jsite_manager_site_data` andmebaasi tabelisse õige veebilehe ID’ga.

Pistikprogrammi paigaldamine veebilehele: *Extensions* -> *Install* seejärel tuleb navigeerida *Extensions* -> *Plugins* ning otsida välja pistikprogramm nimega JSite Manager. Plugin tuleb aktiveerida ning *Secret* väljale tuleb lisada sama salasõna, mis momendil on kasutusel Akeeba Key’na.

## 4.6 JSite Manager edaspidi planeeritavad tööd

### 4.6.1 Üldine

1. Luua UX prototüüp ning vastavalt kinnitatud prototüübile luua veebikeskkonnale disain.
2. Testida JSite Manageri Reaktiiv Grupp OÜ erinevate töötajatega ning otsida võimalikud ohukohad. Vajadusel ohukohad kõrvaldada.
3. Testida kiirust. Täna sel päeval haldab Reaktiiv Grupp OÜ umbkaudu 10 veebilehte, kuid tulevikus tuleb neid juurde. Tuleb tagada, et JSite Manager ei muutuks veebilehtede informatsiooni pärides aeglaseks.
4. Arendada välja mobiilne versiooni.

#### 4.6.2 Listi vaade

1. Tabeli sorteerimise võimalus.
2. Listide eemaldamise nupp. Antud momendil saab liste eemaldada ainult JSite Manager peakasutaja *back-endist*.

#### 4.6.3 Listi detailvaade

1. Tabeli sorteerimise võimalus.
2. Veebilehtede haldamiseks filter.
3. Iga veebilehe tarvis eraldi pistikprogrammi allalaadimise nupp. Pistikprogrammis oleks vaikimisi *Secret* vastavalt sisestatule väärtustatud.

#### 4.6.4 Veebilehe lisamine

1. Lehte lisades automaatne kontroll, kas sisestatud URL ja Administrator URL eksisteerivad.
2. Tekitada uue kontakti tekitamiseks funktsionaalsus. Momendil saab uusi kontakte lisada ainult JSite Manageri peakasutaja *back-endist*.
3. Tekitada uue arendaja tekitamiseks funktsionaalsus. Momendil saab uusi arendajaid lisada ainult JSite Manager peakasutaja *back-endist*.
4. Akeeba Key väärtuse lisamiseks tekitada automaatse salasõna genereerimise nupp.

## 5. Majanduslik efekt

Ettevõtte poolt vaadates seisneb majanduslik efekt selles, et tänud veebikeskkonnale saab töötaja sama ajaga rohkem tööd tehtud. Seega on võimalik kasutajatoega tegelevale töötajale planeerida rohkem tööd.

JSite Manageri sisestatud veebilehtede staatuse põhjal on võimalik hinnata, kas veebilehel oleks vaja uuendada süsteemi, eemaldada ebavajalikud lisaprogrammid või muuta veebilehe konfiguratsiooni. Tänu sellele väheneb turvarisk. Tänu turvariski vähenemisele, väheneb oht kaotada kliente veebilehe rünnakute tõttu. Rünnakute tõttu võib lekkida klientide veebilehtedelt salajast informatsiooni, tekitades nii finantsilist kahju. Taolisel juhul oleks tegu IT kuriteoga, mis on Eesti seadusandluses võrdsustatud majandusliku kuriteoga. Tänu turvalisematele veebilehtedele saab ettevõtte ja kliendid tunda ennast kindlamalt.



## Kokkuvõte

Käesoleva lõputööga seadis autor eesmärgiks luua veebikeskkond, mille abil saab Reaktiiv Grupp OÜ vähendada hallatavate veebilehtede turvariski. Mida rohkem süsteemi monitoorida, seda kiiremini on ohukohad leitavad, seega väheneb JSite Manageri kasutades veebilehtede turvarisk. Tänu sellele võivad ettevõtted tunda end turvalisemalt. JSite Manager veebikeskkona abil on töötajal kergem hallata Joomla! sisuhaldussüsteemil töötavaid veebilehti, vajadusel neid varundada ning viidata veebilehti end uuendama.

Töö raames välja arendatud JSite Manager vähendab ajaliselt Reaktiiv Grupp OÜ poolt hallatavate veebilehtede varundamise ja uuendamise töö protsessi. Lisaks töötaja poolt tehtavate varundamise, uuendamise ja haldamise töödele aitab JSite Manager veenduda arendajal, kas veebiserveri ja Joomla! sisuhaldussüsteemi seaded on korrektselt seadistatud. Esialgu oli plaan luua veebilehtedele paigaldatav laiendus (komponent), millega on võimalik pärida/kontrollida veebilehe parameetreid. Kuna ettevõttel on palju lehti hallata, siis jõuti järeldusele, et lõputöö raames töötatakse välja ühtne süsteemi, kuhu päritakse veebilehega seotud parameetreid. Lisaks tekkis arenduse jooksul idee lisada JSite Manageri sisestatud veebilehtede varundamise võimalus koostöös Akeeba Backup komponendiga.

Sooritati ajakulu testid viie erineva veebilehega eeldusel, et kasutajatoega tegelev töötaja ei tea peast administraatorliidese aadressi ega sisuhaldussüsteemi kasutajaandmeid. Varundamiseks kasutati Akeeba Backup laiendust. Ilma JSite Manageri veebikeskkonna kasutamisele kulus viie veebilehe varundamise ja sisuhaldussüsteemi uuendamisele aega 14 minutit 41 sekundit. JSite Manageri veebikeskkonnaga läks eelnevalt süsteemi sisestatud veebilehtede varundamisele ja sisuhaldussüsteemi uuendamisele aega 7 minutit ja 56 sekundit. Ajakulu vähenes tänu sellele, et administraatorliidese veebiaadressi ei pidanud käsitsi kirjutama veebilehitseja aadressiribale ning veebilehe varundamiseks oli vaja vajutada ainult ühte nuppu. Kokkuhoitav ajakulu kindlasti tulevikus suureneb, kui veebilehti süsteemi juurde lisandub ning veebikeskkonda paremaks ning mugavamaks arendatakse. 2018 aasta

jaanuari algusest alustab Reaktiiv Grupp OÜ uusi Joomla! sisuhaldussüsteemile tehtud veebilehti sisestama JSite Manager programmi, samal ajal täiustades seda vastavalt punktis 4.6 välja toodud nimekirjale.

Autori hinnangul oleks võinud kasutajamugavust JSite Manager keskkonnas lõputöö raames parandada. Siiski on autor rahul, et JSite Manager aitab hoida kokku aega samade tööde teostamisel.

## Kasutatud kirjandus

1. *Content management system*. Computer Hope (2017) [WWW]  
[<https://www.computerhope.com/jargon/c/cms.htm>] (15.11.2017)
2. *Script*. Computer Hope (2017) [WWW]  
[<https://www.computerhope.com/jargon/s/script.htm>] (15.11.2017)
3. *(Joomla!) Administrator*. Joomla! Documentation (2015) [WWW]  
[[https://docs.joomla.org/Administrator\\_\(Application\)](https://docs.joomla.org/Administrator_(Application))] (15.11.2017)
4. *Joomla Extensions*. Joomla! Documentation (2015) [WWW]  
[[https://docs.joomla.org/Extension\\_types\\_\(general\\_definitions\)](https://docs.joomla.org/Extension_types_(general_definitions))] (15.11.2017)
5. *Joomla Components*. Joomla! Documentation (2015) [WWW]  
[[https://docs.joomla.org/Extension\\_types\\_\(general\\_definitions\)](https://docs.joomla.org/Extension_types_(general_definitions))] (15.11.2017)
6. *Joomla Modules*. Joomla! Documentation (2015) [WWW]  
[[https://docs.joomla.org/Extension\\_types\\_\(general\\_definitions\)](https://docs.joomla.org/Extension_types_(general_definitions))] (15.11.2017)
7. *Joomla Plugins*. Joomla! Documentation (2015) [WWW]  
[[https://docs.joomla.org/Extension\\_types\\_\(general\\_definitions\)](https://docs.joomla.org/Extension_types_(general_definitions))] (15.11.2017)
8. *Joomla Template*. Joomla! Documentation (2015) [WWW]  
[[https://docs.joomla.org/Extension\\_types\\_\(general\\_definitions\)](https://docs.joomla.org/Extension_types_(general_definitions))] (15.11.2017)
9. *Joomla Language*. Joomla! Documentation (2015) [WWW]  
[[https://docs.joomla.org/Extension\\_types\\_\(general\\_definitions\)](https://docs.joomla.org/Extension_types_(general_definitions))] (15.11.2017)
10. *Front-end (client side)*. Code Conquest [WWW]  
[<http://www.codeconquest.com/website/client-side-vs-server-side/>] (17.11.2017)

11. *Back-end (server side)*. Code Conquest [WWW]  
[<http://www.codeconquest.com/website/client-side-vs-server-side/>] (17.11.2017)
12. *URL*. Developer Mozilla [WWW]  
[[https://developer.mozilla.org/en-US/docs/Learn/Common\\_questions/What\\_is\\_a\\_URL](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/What_is_a_URL)] (17.11.2017)
13. *Alias*. Computer Hope (2017) [WWW]  
[<https://www.computerhope.com/jargon/a/alias.htm>] (17.11.2017)
14. *Global Configuration*. Joomla! Documentation (2015) [WWW]  
[[https://docs.joomla.org/J3.x:Global\\_configuration](https://docs.joomla.org/J3.x:Global_configuration)] (18.11.2017)
15. *Apache*. The Apache Software Foundation [WWW]  
[<https://www.apache.org/foundation/>] (18.11.2017)
16. *ISS 7*. Microsoft Documentation [WWW]  
[<https://docs.microsoft.com/et-ee/iis/get-started/whats-new-in-iis-10-version-1709/new-features-introduced-in-iis-10-1709>] (19.11.2017)
17. *Popular CMS by Market Share*. R.Mening. Website Setup. (2017) [WWW]  
[<https://websitesetup.org/popular-cms/>] (25.11.2017)
18. *CMS Comparision: Wordpress vs. Joomla vs. Drupal. Which is Best?* C. Newcomer. Wpmajor (2017) [WWW]  
[<https://wpmajor.com/cms-comparison-wordpress-vs-joomla-vs-drupal/>]  
(2.12.2017)
19. *Essay: History of Joomla*. Joomfreak (2016) [WWW]  
[<https://www.joomfreak.com/blog/essay-history-of-joomla.html>] (2.12.2017)

20. *About Joomla*. Joomla! [WWW] [<https://www.joomla.org/about-joomla.html>]  
(2.12.2017)
21. *Search Engine Friendly URLs*. Joomla! Documentation (2016) [WWW]  
[[https://docs.joomla.org/Search\\_Engine\\_Friendly\\_URLs](https://docs.joomla.org/Search_Engine_Friendly_URLs)] (10.12.2017)
22. *What is SSL?* SSL.COM (2016) [WWW]  
[<http://info.ssl.com/article.aspx?id=10241>] (10.12.2017)
23. *Description of core php.ini directives*. PHP.NET [WWW]  
[<http://php.net/manual/en/ini.core.php>] (10.12.2017)
24. *What is allow\_url\_fopen and what must it be set to?* Hetzner [WWW]  
[[https://hetzner.co.za/help-centre/website/what-is-allow\\_url\\_fopen-and-what-must-it-be-set-to/](https://hetzner.co.za/help-centre/website/what-is-allow_url_fopen-and-what-must-it-be-set-to/)] (10.12.2017)
25. *Runtime Configuration*. PHP.NET [WWW]  
[<http://php.net/manual/en/filesystem.configuration.php>] (10.12.2017)
26. *How do i set the upload\_tmp\_dir*. Media Temple [WWW]  
[[https://mediatemple.net/community/products/dv/204644200/how-do-i-set-the-upload\\_tmp\\_dir](https://mediatemple.net/community/products/dv/204644200/how-do-i-set-the-upload_tmp_dir)] (10.12.2017)
27. *HTML introduction*. W3Schools.com [WWW]  
[[https://www.w3schools.com/html/html\\_intro.asp](https://www.w3schools.com/html/html_intro.asp)] (26.12.2017)
28. *What is Javascript*. MDN web docs [WWW]  
[[https://developer.mozilla.org/en-US/docs/Web/JavaScript/About\\_JavaScript](https://developer.mozilla.org/en-US/docs/Web/JavaScript/About_JavaScript)]  
(26.12.2017)

29. *CSS introduction*. W3Schools.com [WWW]  
[[https://www.w3schools.com/css/css\\_intro.asp](https://www.w3schools.com/css/css_intro.asp)] (26.12.2017)
30. *What is PHP?* PHP.NET [WWW] [<http://php.net/manual/en/intro-what-is.php>]  
(26.12.2017)
31. LAMP (Linux, Apache, MySQL, PHP). Whatis.com [WWW]  
[<http://whatis.techtarget.com/definition/LAMP-Linux-Apache-MySQL-PHP>]  
(11.01.2018)
32. *What is MySQL?* MySQL [WWW]  
[<https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>] (11.01.2018)

# Lisa 1 - Joomla veebilehe laienduste plaan

The screenshot displays the Joomla! 3.0 installation template interface. At the top, the Joomla! logo is on the left, the word 'TEMPLATE' is centered in yellow, and a search bar is on the right. Below the logo are navigation links for 'Home', 'Sample Sites', and 'Joomla.org'. A large red abstract image serves as a background for the main content area.

The main content area features a 'Joomla! COMPONENT' section with the following text:

**Joomla! COMPONENT**  
Congratulations! You have a Joomla site! Joomla makes it easy to build a website just the way you want it and keep it simple to update and maintain.

Joomla is a flexible and powerful platform, whether you are building a small site for yourself or a huge site with hundreds of thousands of visitors. Joomla is open source, which means you can make it work just the way you want it to.

The content in this installation of Joomla has been designed to give you an in depth tour of Joomla's features.

Below this text are three columns for user levels: **Beginners**, **Upgraders**, and **Professionals**. Each column contains introductory text and a 'Read more' link. A blue arrow points from the 'Read more: Beginners' link to the word 'PLUGIN'.

On the right side, a sidebar contains several modules:

- About Joomla!** with links: Getting Started, Using Joomla!, The Joomla! Project, The Joomla! Community.
- MODULES** (header)
- This Site** with links: Home, Site Map, Login, Sample Sites, Site Administrator, Example Pages.
- Login Form** with fields for User Name and Password, a Remember Me checkbox, a Log in button, and links for Create an account, Forgot your username?, and Forgot your password?.

At the bottom left, there is a 'Home' link with a dropdown arrow.

## Lisa 2 - JSite Manager veebikeskkonnas hallatavate veebilehtede pistikprogrammi faili jsitemanager.php skript.

```
<?php

// no direct access
defined('_JEXEC') or die;

jimport('joomla.plugin.plugin');
jimport('joomla.filesystem.folder');
jimport(JURI::root().'/libraries/cms/version/version.php');

class plgAjaxjsitemanager extends JPlugin{
    protected $admin_folder_status;
    protected $all_components;
    protected $all_modules;
    protected $all_packages;
    protected $all_plugins;
    protected $all_templates;
    protected $version_full;
    protected $sef;
    protected $sef_rewrite;
    protected $users;
    protected $ssl;
    protected $db_tables;
    protected $expose_php;
    protected $open_basedir;
    protected $display_errors;
    protected $allow_url_fopen;
    protected $allow_url_include;
    protected $file_uploads;
    protected $php_version;
    protected $upload_tmp_dir;

    function onAjaxjsitemanager() {

        $version = new JVersion;

        $secret = JFactory::getApplication()->input->get('s');
        $secretControll = md5(date('Ymd') .
$this->params->get('secret'));
```



```

        if ($secret != md5(date('Ymd') .
$this->params->get('secret') . $_SERVER['REMOTE_ADDR'])) {
            die;
        }

$config = JFactory::getConfig();
$phpinfo = plgAjaxjsitemanager::getPhpInfo();

// Controll "admin" folder
$this->admin_folder_status =
plgAjaxjsitemanager::adminFolder();

// Get all components
$this->all_components =
plgAjaxjsitemanager::getExtensions('component');

// Get all modules
$this->all_modules =
plgAjaxjsitemanager::getExtensions('module');

// Get all packages
$this->all_packages =
plgAjaxjsitemanager::getExtensions('package');

// Get all plugins
$this->all_plugins =
plgAjaxjsitemanager::getExtensions('plugin');

// Get all templates
$this->all_templates =
plgAjaxjsitemanager::getExtensions('template');

// Joomla version
$this->version_full = $version->RELEASE . '.' .
$version->DEV_LEVEL;

// Search Engine Friendly URL's
$this->sef = $config->get('sef');

// Use URL Rewriting
$this->sef_rewrite = $config->get('sef_rewrite');

// Get all users
$this->users = plgAjaxjsitemanager::getUsers();

// SSL
$this->ssl = plgAjaxjsitemanager::ssl();

```

```

// Database tables
$this->db_tables = plgAjaxjsitemanager::allTables();

// Expose_php
$this->expose_php = $phpinfo['expose_php'][0];

// Open_basedir
$this->open_basedir = $phpinfo['open_basedir'][0];

// Display Errors
$this->display_errors = $phpinfo['display_errors'][0];

// Allow url fopen
$this->allow_url_fopen = $phpinfo['allow_url_fopen'][0];

// Allow url include
$this->allow_url_include =
$phpinfo['allow_url_include'][0];

// File uploads
$this->file_uploads = $phpinfo['file_uploads'][0];

// PHP Version
$this->php_version = $phpinfo['PHP Version'];

// Upload tmp dir
$this->upload_tmp_dir = $phpinfo['upload_tmp_dir'][0];

$data = array(
    'admin_folder_status' =>
$this->admin_folder_status,
    'all_components' => $this->all_components,
    'all_modules' => $this->all_modules,
    'all_packages' => $this->all_packages,
    'all_plugins' => $this->all_plugins,
    'all_templates' => $this->all_templates,
    'version_full' => $this->version_full,
    'sef' => $this->sef,
    'sef_rewrite' =>
$this->sef_rewrite,
    'users' => $this->users,
    'ssl' => $this->ssl,
    'db_tables' =>
$this->db_tables,
    'expose_php' =>
$this->expose_php,

```

```

        'open_basedir'           =>
$this->open_basedir,
        'display_errors'       =>  $this->display_errors,
        'allow_url_fopen'      =>
$this->allow_url_fopen,
        'allow_url_include'    =>
$this->allow_url_include,
        'file_uploads'         =>
$this->file_uploads,
        'php_version'          =>
$this->php_version,
        'upload_tmp_dir'       =>  $this->upload_tmp_dir
    );

    return $data;
}

/* Controll "admin" folder function */
private static function adminFolder(){
    $response = get_headers(JURI::base() . 'administrator',
1);

    $return = null;

    if($response[0] == 'HTTP/1.1 404 Not Found'){
        $return = '0';
    } else {
        $return = '1';
    }

    return $return;
}

// Get list of extensions (WHERE TYPE)
private static function getExtensions($type = 'component'){
    $db = JFactory::getDbo();
    $query = $db->getQuery(true);
    $query->select('extension_id, name, element, enabled,
manifest_cache');
    $query->from('#__extensions');
    $query->where('type = "' . $type . '"');
    $db->setQuery($query);
    $results = $db->loadObjectList();

    if($results){
        return $results;
    } else {

```

```

        return false;
    }
}

// Get all users
private static function getUsers(){
    $db = JFactory::getDbo();
    $query = $db->getQuery(true);
    $query->select('id, name, username, email, block,
lastvisitDate, registerDate');
    $query->from('#__users');
    $db->setQuery($query);
    $results = $db->loadObjectList();

    if($results){
        return $results;
    } else {
        return false;
    }
}

// Controll SSL status
private static function ssl(){
    if (!empty($_SERVER['HTTPS']) && $_SERVER['HTTPS'] !=
'off') {
        $ssl = 'Yes';
    } else {
        $ssl = 'No';
    }
    return $ssl;
}

// Get all tables
private static function allTables(){
    $config = JFactory::getConfig();
    $db = JFactory::getDbo();
    $query = $db->getQuery(true);
    $query->select('TABLE_NAME');
    $query->from('INFORMATION_SCHEMA.TABLES');
    $query->where('table_schema = "' . $config->get('db') .
''');
    $query->where('TABLE_NAME LIKE "' .
$config->get('dbprefix') . '%"');
    $db->setQuery($query);
    $results = $db->loadObjectList();

    if($results){

```

```

        return $results;
    } else {
        return false;
    }
}

// Get php info
private static function getPhpInfo(){
    ob_start();
    phpinfo();
    $pinfo = ob_get_contents();
    ob_end_clean();
    preg_match_all('#<body[^\>]*>(.*?)</body>#siU', $pinfo,
$output);
    $output = preg_replace('#<table[^\>]*>#', '<table
class="table table-striped adminlist">', $output[1][0]);
    $output = preg_replace('#(\w),(\w)#', '\1, \2', $output);
    $output = preg_replace('#<hr />#', '', $output);
    $output = str_replace('<div class="center">', '',
$output);
    $output = preg_replace('#<tr class="h">(.*?)</tr>#',
'<thead><tr class="h">$1</tr></thead><tbody>', $output);
    $output = str_replace('</table>', '</tbody></table>',
$output);
    $output = str_replace('</div>', '', $output);

    $html = strip_tags($output, '<h2><th><td>');
    $html = preg_replace('/<th[^\>]*>([^\<]+)</th>/',
'<info>\1</info>', $html);
    $html = preg_replace('/<td[^\>]*>([^\<]+)</td>/',
'<info>\1</info>', $html);
    $t = preg_split('/(<h2[^\>]*>[^\<]+</h2>)/', $html, -1,
PREG_SPLIT_DELIM_CAPTURE);
    $r = array();
    $count = count($t);
    $p1 = '<info>([^\<]+)</info>';
    $p2 = '/' . $p1 . '\s*' . $p1 . '\s*' . $p1 . '/';
    $p3 = '/' . $p1 . '\s*' . $p1 . '/';

    for ($i = 1; $i < $count; $i++){
        if (preg_match('/<h2[^\>]*>([^\<]+)</h2>/', $t[$i],
$matches)){
            $name = trim($matches[1]);
            $vals = explode("\n", $t[$i + 1]);

            foreach ($vals AS $val){
                // 3cols

```

```

        if (preg_match($p2, $val, $matches)){
            $r[$name][trim($matches[1])] =
array(trim($matches[2]), trim($matches[3]),);
        }
        // 2cols
        elseif (preg_match($p3, $val, $matches)){
            $r[$name][trim($matches[1])] =
trim($matches[2]);
        }
    }
}
return $r['Core'];
}
}

```

## Lisa 3 - JSite Manager veebikeskkonnas hallatavate veebilehtede pistikprogrammi jsitemanager.xml skript.

```
<?xml version="1.0" encoding="utf-8"?>

<extension version="3.1" type="plugin" group="ajax" method="upgrade">

    <name>JSite Manager</name>
    <version>0.2</version>
    <creationDate>14-11-2017</creationDate>
    <author>Raido Martinson</author>
    <authorUrl>jsitemanager.ee</authorUrl>
    <description>Joomla Ajax Plugin to determine the Joomla
information.</description>

    <files>

        <filename plugin="jsitemanager">jsitemanager.php</filename>

    </files>

    <config>

        <fields name="params">

            <fieldset name="basic">

                <field name="secret" type="text" default="passphrase"
label="Secret" description="This is the secret for your api. Not
everyone should know your joomla version." />

            </fieldset>

        </fields>

    </config>

</extension>
```

## Lisa 4 - PHP fail, mis pärib JSite Manager keskkonda sisestatud veebilehelt informatsiooni.

Suhtleb veebilehele paigaldatud pistikprogrammiga.

```
<?php

defined('_JEXEC') or die;

class dataHelper{
    public $url = "";
    public $secret = "";
    public $whitelist = "";
    public $id = "";
    public $returnIP = "";

    private function getCMSVersionWeb($url, $secret, $whitelist) {
        if ( substr($url, -1) != "/" ) {
            $url = $url . "/";
        }

        $fullurl = $url .
'/index.php?option=com_ajax&plugin=jsitemanager&format=json';
        $ch = curl_init();
        curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
        curl_setopt($ch, CURLOPT_URL, $fullurl);
        curl_setopt($ch, CURLOPT_RETURNTRANSFER, true );
        curl_setopt($ch, CURLOPT_POSTFIELDS, "s=" . md5(date('Ymd') .
$secret . $whitelist));
        curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
        curl_setopt($ch, CURLOPT_COOKIEFILE,
realpath(dirname(__FILE__) . '/cookies.txt'));
        curl_setopt($ch, CURLOPT_COOKIEJAR, realpath(dirname(__FILE__)
. '/cookies.txt'));
        //curl_setopt($ch, CURLOPT_HTTPHEADER, $this->header);
        $result = curl_exec($ch);
        curl_close($ch);
        $item = json_decode($result);

        dataHelper::addThisItem($item);
    }

    // Public functions
    public function __construct($url, $secret, $whitelist, $id,
$returnIP)
    {
        $this->url = $url;
```



```

$this->secret = $secret;
$this->whitelist = $whitelist;
$this->id = $id;
$this->returnIP = $returnIP;
if (strlen($this->url) > 0) {
    //$this->getcms();
    $this->getCMSVersionWeb($this->url, $this->secret,
$this->whitelist);
} else {
    return false;
}
}

```

```

private function addThisItem($item){
    if($item->data[0]->admin_folder_status){
        $this->admin_folder_status =
$item->data[0]->admin_folder_status;
    }
    if($item->data[0]->all_components){
        $this->all_components = $item->data[0]->all_components;
    }
    if($item->data[0]->all_modules){
        $this->all_modules = $item->data[0]->all_modules;
    }
    if($item->data[0]->all_packages){
        $this->all_packages = $item->data[0]->all_packages;
    }
    if($item->data[0]->all_plugins){
        $this->all_plugins = $item->data[0]->all_plugins;
    }
    if($item->data[0]->all_templates){
        $this->all_templates = $item->data[0]->all_templates;
    }
    if($item->data[0]->version_full){
        $this->version_full = $item->data[0]->version_full;
    }
    if($item->data[0]->sef){
        $this->sef = $item->data[0]->sef;
    }
    if($item->data[0]->sef_rewrite){
        $this->sef_rewrite = $item->data[0]->sef_rewrite;
    }
    if($item->data[0]->users){
        $this->users = $item->data[0]->users;
    }
    if($item->data[0]->ssl){

```

```

        $this->ssl = $item->data[0]->ssl;
    }
    if($item->data[0]->db_tables){
        $this->db_tables = $item->data[0]->db_tables;
    }
    if($item->data[0]->expose_php){
        $this->expose_php = $item->data[0]->expose_php;
    }
    if($item->data[0]->open_basedir){
        $this->open_basedir = $item->data[0]->open_basedir;
    }
    if($item->data[0]->display_errors){
        $this->display_errors = $item->data[0]->display_errors;
    }
    if($item->data[0]->allow_url_fopen){
        $this->allow_url_fopen = $item->data[0]->allow_url_fopen;
    }
    if($item->data[0]->allow_url_include){
        $this->allow_url_include =
$item->data[0]->allow_url_include;
    }
    if($item->data[0]->file_uploads){
        $this->file_uploads = $item->data[0]->file_uploads;
    }
    if($item->data[0]->php_version){
        $this->php_version = $item->data[0]->php_version;
    }
    if($item->data[0]->upload_tmp_dir){
        $this->upload_tmp_dir = $item->data[0]->upload_tmp_dir;
    }
}
}
}

```