

TALLINN UNIVERSITY OF TECHNOLOGY  
School of Information Technologies

Igor Stulov 155218 IAPB

# **DEVELOPMENT OF A CONTRACT MANAGEMENT SYSTEM**

Bachelor's thesis

Supervisor: Ago Luberg  
MSc

Tallinn 2019

TALLINNA TEHNIKAÜLIKOOL  
Infotehnoloogia teaduskond

Igor Stulov 155218 IAPB

# **LEPINGUTE HALDAMISE INFOSÜSTEEMI ARENDAMINE**

Bakalaureusetöö

Juhendaja: Ago Luberg  
MSc

Tallinn 2019

## **Author's declaration of originality**

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Igor Stulov

23.05.2019

## **Abstract**

The objective of this thesis is to solve an existing problem of company IT Solutions Group OÜ and to create an administrative zone for contract management for this company. Right now the company creates each contract manually, what is time-consuming and might result in a contract loss. Optimal solution is to create an internal web application as an addition to the company's main website, where the project manager can create new contracts from template by inserting all the important data and download the document to his/her local machine. In addition, the web application will upload a copy to the server preventing data loss. The web application is written in Hypertext Preprocessor (PHP) and JavaScript (JS) and pages are constructed using Bootstrap.

The result of this work is a working web application, where the project manager can create sales, rent and maintenance contracts as well as addendum to those. The templates are provided in Russian, Estonian and English languages, fill on-page located fields with necessary data and download the document without direct. For the recurrent clients, the project manager can select a customer from the list and data will be filled automatically. Also, he/she can add new customers to the customer list, new services to the maintenance or the additional service lists. Furthermore, he/she can monitor the contracts within the application in separate tabs.

The thesis describes the development and the implementation of the web application, outlining how data is being processed and how contracts are created.

This thesis is written in English and is 38 pages long, including 7 chapters and 27 figures.

## **Annotatsioon**

### **Administratiivse Keskkonna loomine**

Selle töö eesmärgiks on lahendada ettevõttes IT Solutions Group OÜ eksisteeriv probleem, luues administratiivse lepingutehaldamise keskkonna lähtuvalt ettevõtte vajadustest. Lepinguid koostatakse ettevõttes käsitsi, mis on aeganõudev ning võib tuua kaasa vigu. Tegevuse optimeerimieks on võetud plaani sisemise lepingutehaldamise keskkonna ehk veebirakenduse loomine, mille abil Projektijuht saab luua uusi lepinguid olemasoleva malli baasil, sisestades selleks vajalikke andmeid ja laadida alla valmis dokumendi. Lisaks, veebirakendus salvestab koopia serverisse, mis tagab andmete säilimise. Veebirakendus on kirjutatud kasutades Hypertext Preprocessorit (PHP) ja JavaScripti (JS) ja leheküljed on ehitatud üles kasutades Bootstrapi.

Töö tulemuseks on töötav veebirakendus, kus Projektijuht saab luua müügi-, rendi- ja hoolduslepinguid ning nende lisaid. Lepingutemallid on kättesaadavad vene, eesti ja inglise keeles. Projektijuht täidab vajalikud lahtrid ja saab alla laadida valmis dokumendi. Püsiklientide puhul saab ta valida kliendi nimekirjast ja süsteem sisestab andmed automaatselt. Samuti saab ta lisada uusi kliente kliendiloendisse, uusi hooldusteenuseid või lisateenuseid teenuseloendisse. Lisaks saab ta hallata kliendilepinguid, avades neid eraldi lehtedel.

Selles töös kirjeldatakse veebirakenduse loomist ja implementeerimist, selgitades lahti, kuidas andmed on töödeldud ja kuidas lepinguid luuakse.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 38 leheküljel, 7 peatükki ja 27 joonist.

## List of abbreviations and terms

Autoincrement	Automatically generated sequential number
Frontend and Backend	Two sections of software, the presentation layer and the data access layer respectively
jQ	jQuery, JavaScript library
JS	JavaScript
MySQL	Relational database, used to store and retrieve data
PHP	Hypertext Preprocessor
PM	Project manager, person who creates contracts
SQL injection	An execution of unwanted action by the creator of the script. Usually, it is used to create superuser or to completely erase database or table.
Wordpress	Content Management System (CMS) used to create and administrate webpages
XML	eXtensible Markup Language

## Table of contents

1 Introduction .....	9
2 Why making a new web application?.....	10
3 Contracts.....	11
3.1 Sales contract.....	11
3.2 Rent contract.....	11
3.3 Maintenance contract.....	12
3.4 Additions to contract .....	12
4 Web application and requirements for it .....	13
4.1 Setting goals .....	13
4.2 Contract templates .....	14
4.3 Contract modification .....	14
5 Web application implementation.....	16
5.1 Early stage and starting page .....	16
5.2 Main working page.....	17
5.2.1 Starting tabs .....	17
5.2.2 Project manager tab, “Make Agreement” sub-tab.....	18
5.2.3 Project manager tab, “Add/remove maintenance service” sub-tab .....	24
5.2.4 Project manager, “Add/remove additional service” sub-tab .....	26
5.2.5 Project manager, “Manage clients” sub-tab .....	26
5.2.6 Project manager, “Show all contracts” sub-tab .....	28
5.3 Code interaction with documents and database.....	29
5.3.1 How a contract template turns into a document ready for signing.....	29
5.3.2 How does the document get its serial number.....	30
5.3.3 How does the document get creation date .....	31
5.3.4 How is a new client added to the database .....	31
5.3.5 How are services added and removed .....	32
6 Web application design .....	34
7 Summary.....	37
References .....	38

## List of figures

Figure 1: Login page .....	16
Figure 2: Administrator's starting tab .....	17
Figure 3: Project manager tab and its sub-tabs.....	18
Figure 4: Make Agreement sub-tab with Sales contract in Russian language selected .	19
Figure 5: Make Agreement sub-tab with sample text example.....	20
Figure 6: Make Agreement sub-tab with selected company example.....	20
Figure 7: Sales contract template in Russian.....	21
Figure 8: Downloaded modified document .....	21
Figure 9: Valid data for example document .....	22
Figure 10: The additional service/services selection.....	23
Figure 11: Document updated with the selected services.....	23
Figure 12: Addition to Sales contract number 62.....	24
Figure 13: Add/remove maintenance service sub-tab .....	25
Figure 14: Manage clients sub-tab.....	26
Figure 15: Red outline when trying to add client without vital data .....	27
Figure 16: Show all contracts sub-tab .....	28
Figure 17: An example of the display of the selected contract within the web application .....	29
Figure 18: An example of document.xml inside .docx file .....	29
Figure 19: Document table data .....	31
Figure 20: Clients table .....	32
Figure 21: Maintenance services table .....	32
Figure 22: Additional services table.....	32
Figure 23: Additional services table after "Dolor sit" service was deleted using the web application .....	33
Figure 24: An example of elements positioning on "Make agreement" sub-tab.....	34
Figure 25: An example of element positioning on "Manage clients" sub-tab.....	35
Figure 26: Web page with sidebar toggle on.....	35
Figure 27: Web page with sidebar toggle off .....	36



# 1 Introduction

Contracts play an important role in every company. They are a connecting link between the client and the service provider marking their business partnership. Moreover, contracts contain vital information about required services, payment and deadlines. Company IT Solutions is no different – it engages in contractual relations with its clients and those client relations need a proper management.

Company IT Solutions provides web services to corporate clients across the Baltics, such as creating online shops and homepages, administration or development of an existing website. Today they are making every contract manually using existing templates. This process is time-consuming and can potentially be erroneous.

The objective of this thesis is to optimize and automate this process as much as possible. The main focus is on two main issues: time and reliability. Introducing a web application for contract creation and management can solve both of these issues: required data is placed in template file automatically, which significantly saves time, and the document is saved not only to a local machine, but also to the server, which prevents data loss. The necessary architecture and structure were designed in close cooperation with the company, tailoring the application to their actual needs.

Web application is developed using combination of PHP programming language and jQuery, a JavaScript programming language library. This combination provides great opportunities for both frontend and backend.

## **2 Why making a new web application?**

The main business of IT Solutions is developing websites using a Wordpress content management system. Company's own webpage is no different and was constructed using Wordpress. It comes with a build-in administrator dashboard, which helps to add new content and further expand the site's functionality. However, this is not enough to build a secure application page, where employees can work. Moreover, Wordpress comes with paid plugins and themes, more importantly, only one theme is allowed on the whole website.

Another solution is to install the premade admin panel. Considering the fact, that Wordpress is highly customizable, it is allowed to place side application into the working dashboard. For example, Forest Admin, it provides great opportunities to manage user database and is highly secure, but alas it does not provide service for contract creation [1]. Given the above, it was decided to develop its own web application, which will be an independent addition to the main site.

Own web application provides greater creative freedom as opposed to acquiring a solution offered by CRM, project management or document management service providers. The application will expand company's opportunities and serve as an administrative addition to the main website. Right now, the main focus is on the project manager's duties and contracts creation, however, this is just the beginning. Further development also includes the client's base handling, advertising and automation of some services offered by the company.

Considering that the creation of contracts is the starting point of every deal, the priority is given to the project manager's duties. IT Solutions works with companies from different countries and the ability to quickly create contracts is a must.

As was stated earlier, the project manager's job also includes the creation of contracts. More information regarding contracts is provided in the next chapter.

## **3 Contracts**

There are four different type of contract: Sales contract, Rent contract, Maintenance contract and Additions to contract, all made in Russian, Estonian and English languages. Each contract must be filled with the client's information such as the company name, the owner's name, the bank account, the company address and the company registration number. Other info varies from contract to contract.

Each document has already IT Solutions' information in it as well as the owner's credentials. Also, there are different terms and conditions for every contract.

Every document must be signed both by IT Solutions' and client's representative in order to take effect.

### **3.1 Sales contract**

A sales contract is the most common contract type. Briefly, this type of contract is for the development and the transfer of the site to the client's final usage. This contract has no price in it nor deadlines, they are marked in the Additions to contract as well as additional services.

Only information this document requires is the client information.

### **3.2 Rent contract**

A rent contract is the rarest contract and also the longest one. This type of contract is for the development and transfer of the site to the client's temporary usage.

This document has two additional client information fields: owner's email and owner's phone number. In addition, domain name, contract fee, monthly rent fee and contract end date must be filled.

### **3.3 Maintenance contract**

A maintenance contract is the shortest one. This type of contract is for maintaining client's site, but only if it was created by IT Solution.

This document must be filled with client information, numbers of addition and document it is attached to, maintenance services needed to be done and their price.

### **3.4 Additions to contract**

An additions to contract is the most complex type of document. Similarly to the Maintenance contract, it only applies to contracts created by IT Solutions.

This document must be filled with client information, numbers of addition and document it is attached to, additional services needed to be done and their price, separated as prepayment made three days after contract is signed and payment two days after the work is complete.

## 4 Web application and requirements for it

### 4.1 Setting goals

To begin with, the web application had to be a place, where different employees could fulfill their duties way faster, transforming a time-consuming routine into a simple select-download mechanism.

In addition to the creation of contract, the project manager must have ability to add clients to the client database, new services to maintenance services list or additional services list and to view all the documents created within the web application.

Alongside the project manager, the web application also must have at least one administrator, who grants and discontinues user roles.

The project manager should be able not only to manually input necessary data by hand, but also have the ability to select it from the list. As an example - client information, by selecting one from the list, all the fields are filled automatically using information from the database. Service selection is slightly different, one cannot write it manually, but only select from the list.

Some information in the document must be filled automatically without employee intervention: contract creation date and contract number. In addition, contracts must be saved with a certain name pattern, which varies from contract to contract.

As an example, a sales contract is saved as a document with the name `contractNumber_CompanyName_year-month-day.docx`, where `contractNumber` is the contract serial number, `CompanyName` is client's company name, `year-month-day` is the document creation date.

## 4.2 Contract templates

Every contract template is a docx document that includes standard terms for service provision, such as normal payment terms and means of communication. This information also includes information about IT Solutions, its representative and typical points of agreement.

In order to create a web application capable of changing docx contents, it is first necessary to make changes to the template files.

## 4.3 Contract modification

To start with, document type docx is in fact a zip archive and it contains XML[2]. This means, that every word is wrapped in XML tags. This makes file slightly difficult to modify. In order to make changes to template document, it is important to add special words, which will be replaced during the code execution.

Those words differ from document to document, but seven of them are the same for all:

- **DATE** – this word is being replaced with current date
- **COMPANY** – this word is being replaced with client's company name
- **REGNUM** – this word is being replaced with client's company registration number
- **ADDRESS** – this word is being replaced with client's company address
- **OWNER** – this word is being replaced with client's full name
- **BANK** – this word is being replaced with client's bank account
- **BREAK, BREAKE** – those words are being replaced with additional newline in order to make document look better inside the web application

Sales contract and Rent contract share additional word:

- **AUTONUM** – this word is being replaced with document's serial number

Sales contract has no additional words to be replaced.

Rent contract has six additional words:

- **EMAIL** – this word is being replaced with client's email
- **PHONE** – this word is being replaced with client's phone number
- **DOMAIN** – this word is being replaced with selected by client domain name
- **COST** – this word is being replaced with contract price
- **PAYMENT** – this word is being replaced with monthly rental fee
- **ENDTIME** – this word is being replaced with contract expiration date

Maintenance contract and Addition to contract document share two additional words:

- **numOfAddition** – this word is being replaced with the contract supplement number.
- **numOfAgreement** – this word is being replaced with the number of the document this contract is applied to

Maintenance contract does not have **ADDRESS** word, it however has two additional fields:

- **SERVICE** – this word is being replaced with the selected maintenance services
- **COST** – this word is being replaced with the maintenance contract price

Addition to contract has five additional words:

- **FULLPRICE** – this word is being replaced with the price of services
- **HALF** – this word is being replaced with **FULLPRICE** divided by two

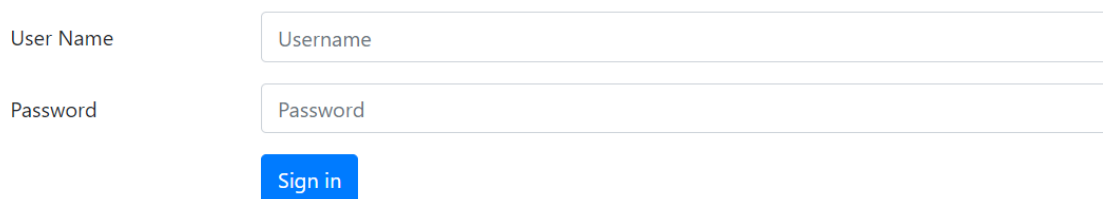
- **PERIOD** – this word is being replaced with the client deadline for providing information
- **ENDTIME** – this word is being replaced with the contract expiration date
- **SERVICE** – this word is being replaced with the selected additional services

## 5 Web application implementation

### 5.1 Early stage and starting page

To begin with, the web application must have an authentication feature. Each user, no matter whether it is an administrator or a project manager, must first log in to the application in order to start working. Every worker's credentials are stored in the database, so the logging in process is a simple check whether input data matches the one in the database. The application uses the MySQL database.

Login page contains only username and password fields as seen in Figure 1.



The image shows a simple login form. On the left, there are two labels: 'User Name' and 'Password'. To the right of 'User Name' is a text input field containing the placeholder text 'Username'. To the right of 'Password' is a text input field containing the placeholder text 'Password'. Below these two input fields is a blue rectangular button with the text 'Sign in' in white.

Figure 1: Login page

To make the web application more secure, it is important to avoid SQL injection. Both input fields may be the targets of that unwanted action. In order to prevent SQL injection, input values need to be escaped. Fortunately, PHP provides two functions to select from.

Considering the fact, that the application uses a MySQL database, the required function is `mysqli_real_escape_string()` [3]. This function escapes special characters in a string



and this prevents SQL injection. It escapes seven special characters, but application really needs to escape only two: single quote and double quote.

During the creation of the prototype, it was necessary to create users with different roles. Those roles are an administrator and a user. The administrator has an ability to change other user roles and has full access to the web application. The user does not have administrative functions, however has access to the project manager panel. During this stage of development other roles are not needed.

## 5.2 Main working page

### 5.2.1 Starting tabs

To begin with, the main working page is divided into tabs and starting tab varies from the user role. As was already mentioned, the administrator has additional abilities and requires its own control tab. The administrator's tab is called "User's rights" and allows him/her to change user roles as is shown in Figure 2.

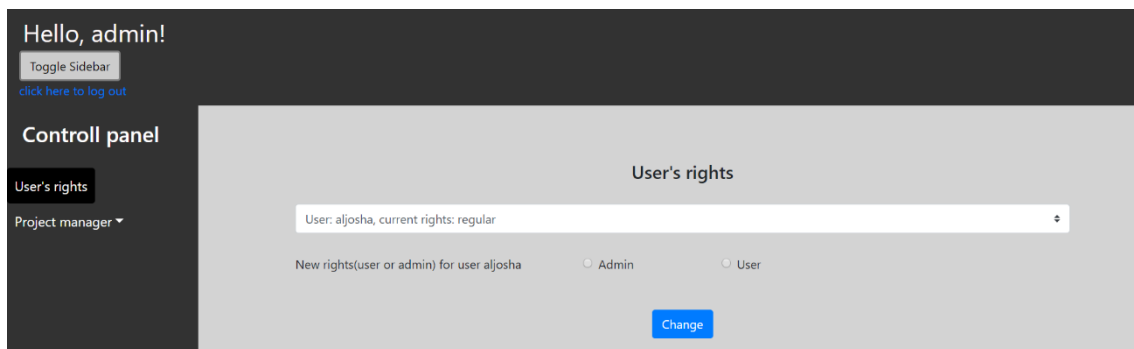


Figure 2: Administrator's starting tab

At this panel the administrator can select any user except for him-/herself from the employee table in the database. After that administrator must select new rights for the selected person and press the "Change" button. This will lead to the change of the user rights.

A regular user starts directly from the "Project manager" tab.

### 5.2.2 Project manager tab, “Make Agreement” sub-tab

The project manager tab is divided into more sub-tabs. Those sub-tabs are “Make Agreement”, “Add/remove maintenance service”, “Add/remove additional service”, “Manage clients” and “Show all contracts” as is shown in Figure 3.

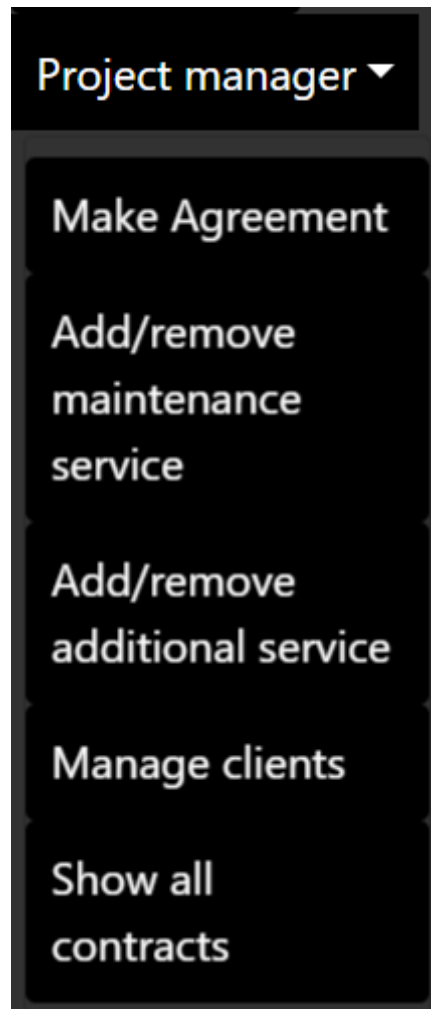


Figure 3: Project manager tab and its sub-tabs

As was already mentioned, regular users start from the “Project manager” tab, to be more precise from the “Make Agreement” sub-tab. This sub-tab is the most important one and right here every contract is created. “Make Agreement” sub-tab is shown in Figure 4.

Figure 4: Make Agreement sub-tab with Sales contract in Russian language selected

Once again, this sub-tab is the most important one. Here, the user can input all client's data required for the contract and can preview document on the right side of the page. The user can select one of the companies from the list. The company list is stored in the database in the table called clients. This will automatically place data in the input fields and change the displayed document.

An action is bound to the input fields and the displayed document is updating constantly. This ensures that with every typed character the document will change in the real time. Same applies also to the company selector, as soon as company is selected the preview document will be updated.

As an example, sample text is inserted into those text fields. Outcome is shown in Figure 5.

### Select type of contract

Sale contract
Rent contract
Maintenance contract
Additions to contract

In Russian
In Estonian
In English

Select company <span style="float: right;">▾</span> Company name <input type="text" value="Sample"/> Owner <input type="text" value="Sample"/> Bank account <input type="text" value="Sample"/> Address <input type="text" value="Sample"/> Registration number <input type="text" value="Sample"/>	ДОГОВОР ОКАЗАНИЯ УСЛУГ № 62 г. Таллин  6.5.2019 IT Solutions Group OÜ, Регистрационный код: 14049412, расположенный по адресу: Katuserapi 4 Кабинет С111, Таллинн, Ласнамяэ, 11412, в лице члена правления Pavel Zahharov (39207110815) (далее будет называться как: Исполнитель) а также Sample, Регистрационный код: Sample расположенный по адресу Sample в лице члена правления Sample (далее будет называться как: Заказчик) <b>ПРЕДМЕТ ДОГОВОРА</b> 1.1. На условиях настоящего Договора Исполнитель обязуется выполнить следующие виды услуг: Разработки Сайта - на трёх языках. Заказчик обязуется оплатить эти услуги в соответствии с п.3 настоящего Договора. Web-Сайт Заказчика, именуемый в
---	---

Download document

Figure 5: Make Agreement sub-tab with sample text example

As is seen from Figure 5, the document is updated with the data from the input fields. An outcome with the selected company is shown in Figure 6.

### Select type of contract

Sale contract
Rent contract
Maintenance contract
Additions to contract

In Russian
In Estonian
In English

User: Fairy IT <span style="float: right;">▾</span> Company name <input type="text" value="Fairy IT"/> Owner <input type="text" value="Heart Dragneel"/> Bank account <input type="text" value="EE2244556677"/> Address <input type="text" value="Eesti 20-0"/> Registration number <input type="text" value="14525"/>	ДОГОВОР ОКАЗАНИЯ УСЛУГ № 62 г. Таллин  6.5.2019 IT Solutions Group OÜ, Регистрационный код: 14049412, расположенный по адресу: Katuserapi 4 Кабинет С111, Таллинн, Ласнамяэ, 11412, в лице члена правления Pavel Zahharov (39207110815) (далее будет называться как: Исполнитель) а также Fairy IT, Регистрационный код: 14525 расположенный по адресу Eesti 20-0 в лице члена правления Heart Dragneel (далее будет называться как: Заказчик) <b>ПРЕДМЕТ ДОГОВОРА</b> 1.1. На условиях настоящего Договора Исполнитель обязуется выполнить следующие виды услуг: Разработки Сайта - на трёх языках. Заказчик обязуется оплатить эти услуги в соответствии с п.3 настоящего Договора. Web-Сайт Заказчика, именуемый в
--	--

Download document

Figure 6: Make Agreement sub-tab with selected company example

Once the user clicks on the “Download document” button, the web application will initiate download with the modified document to the user’s personal computer. This document will be also saved on the server. The template document is shown in Figure 7 and the downloaded document content is shown in Figure 8.

**ДОГОВОР ОКАЗАНИЯ УСЛУГ № AUTONUM**

г. Таллин

**DATE**

**IT Solutions Group OÜ**, Регистрационный код: 14049412, расположенный по адресу: Kätusepapi 4 Кабинет С111, Таллинн, Ласнамяэ, 11412, в лице члена правления **Pavel Zahharov** (39207110815) (далее будет называться как: **Исполнитель**)

а также

**COMPANY**, Регистрационный код: REGNUM расположенный по адресу ADDRESS в лице члена правления OWNER (далее будет называться как: **Заказчик**)

Figure 7: Sales contract template in Russian

**ДОГОВОР ОКАЗАНИЯ УСЛУГ № 62**

г. Таллин

**06.05.2019**

**IT Solutions Group OÜ**, Регистрационный код: 14049412, расположенный по адресу: Kätusepapi 4 Кабинет С111, Таллинн, Ласнамяэ, 11412, в лице члена правления **Pavel Zahharov** (39207110815) (далее будет называться как: **Исполнитель**)

а также

**Fairy IT**, Регистрационный код: 14525 расположенный по адресу Eesti 20-0 в лице члена правления Heart Dragneel (далее будет называться как: **Заказчик**)

Figure 8: Downloaded modified document

As is shown in Figure 7 the optimal outcome is achieved and document is ready for signing. This document was saved as 62\_FairyIT\_2019-05-06.docx, where 62 is contract serial number, FairyIT is client’s company name and 2019-05-06 is contract creation and download date.

As was already mentioned, a Sales contract does not include price or additional services. In order to fill the gap, an Additions to contract document must be created. To do so, the user clicks on the “Additions to contract” navigation button and is transferred to another tab.

At this point, the user must input all the corresponding fields. Most important part is to input client data and number of agreement correctly. As a continuation of previous example, correct data for document 62\_FairyIT\_2019-05-06.docx is shown in Figure 9.

The screenshot shows a web application interface for creating an "Additions to contract" document. At the top, there are four navigation buttons: "Sale contract", "Rent contract", "Maintenance contract", and "Additions to contract". Below these are three language selection buttons: "In Russian", "In Estonian", and "In English". The main form contains several input fields: "Number of addition" (value: 1), "Number of contract" (value: 62), "Services" (dropdown menu), "Data waiting period (In days)", "Contract end date (In weeks)", "Price", "User" (value: Fairy IT), and "Company name" (value: Fairy IT). To the right of the form is a preview area showing the document content in Russian, including the title "Приложение 1 к Договору оказания услуг № 62" and a list of services and terms.

Figure 9: Valid data for example document

In addition to the valid company data, additional services, their price, data waiting period and contract end date must be filled. As for the services, they are selected from the table stored in the database. That list can be expanded in “Add/remove additional services” sub-tab. Service selection and document modification with selection of services is shown in Figure 10 and Figure 11 respectively.

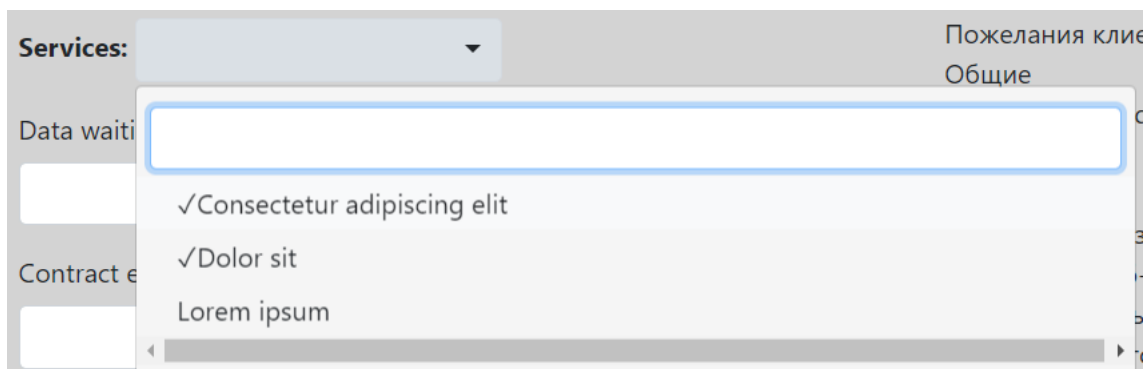


Figure 10: The additional service/services selection

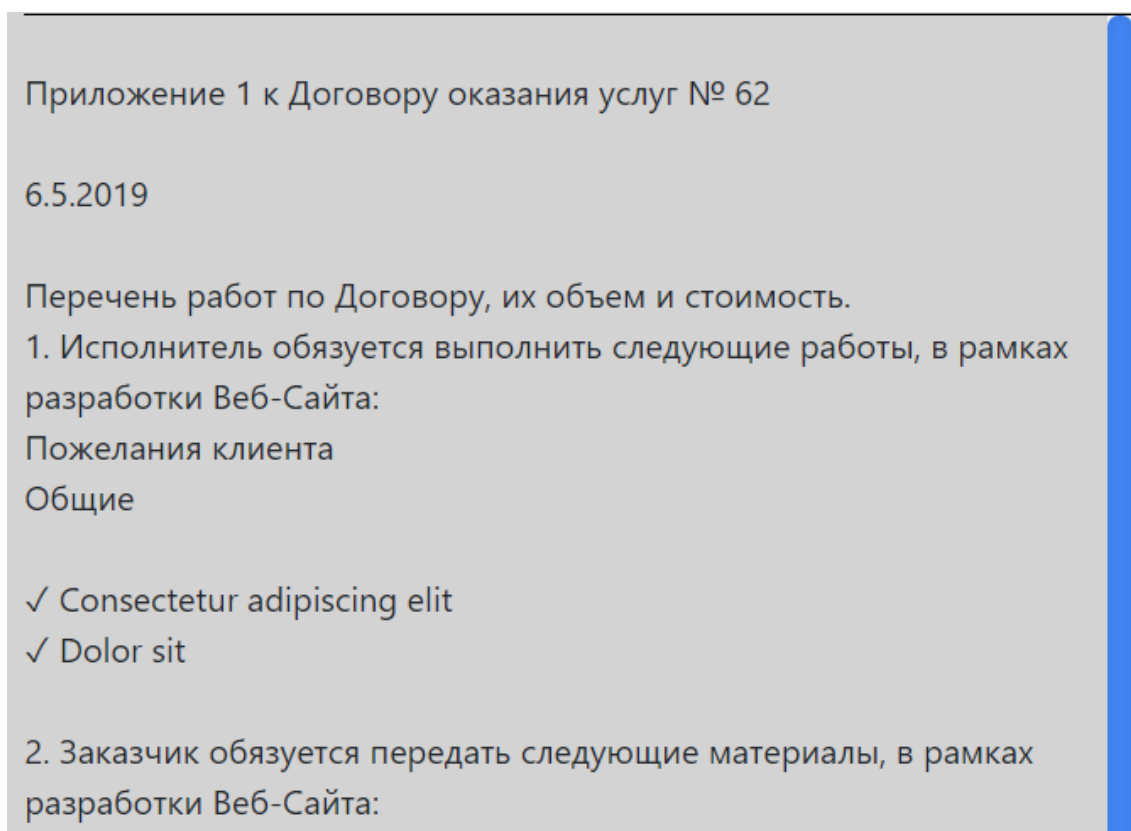


Figure 11: Document updated with the selected services

After filling other inputs the document is ready for download and can be seen in Figure 12.

06.05.2019

Перечень работ по Договору, их объем и стоимость.

1. Исполнитель обязуется выполнить следующие работы, в рамках разработки Веб-Сайта:

Пожелания клиента

Общие

- ✓ Consectetur adipiscing elit
- ✓ Dolor sit

Figure 12: Addition to Sales contract number 62

Example of addition contract is downloaded with the name AdditionNr1ToContractNr62\_2019-05-06.docx, which contains the number of addition and the document number it is applied to along with the date of contract creation.

### 5.2.3 Project manager tab, “Add/remove maintenance service” sub-tab

The add/remove maintenance service tab contains all the information regarding maintenance services. Here the maintenance service list is displayed and the user can enlarge or shorten this list by clicking the corresponding buttons. This sub-tab is shown in Figure 13.



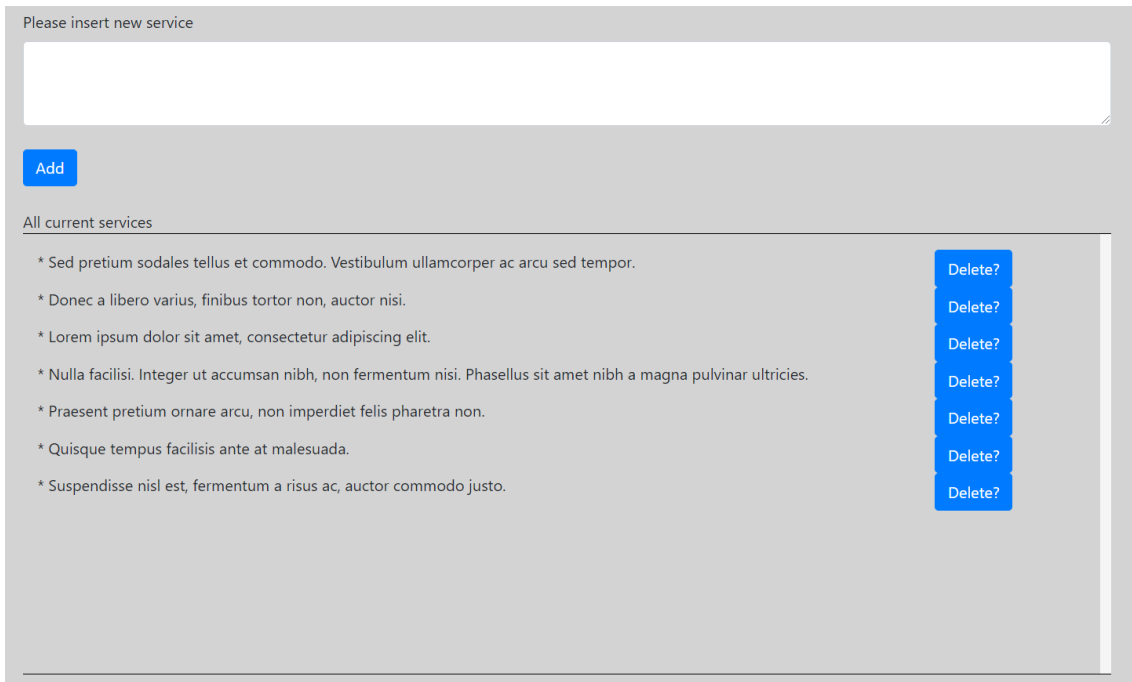


Figure 13: Add/remove maintenance service sub-tab

In order to add a new service, the user must input the service title into the text area at the beginning of the tab and click the “Add” button. Once this button is clicked, the application checks whether the service is already in the database. If so, this service will not be added once again, and user will be notified about it. Otherwise, a new service will be added to the maintenance service list, the user will be notified about that and the new service will appear at the “All current services” sector alongside with the “Delete?” button.

In order to delete an existing service, the user must press the “Delete?” button. Once it is pressed, the web application will notify the user about a successful removal and the service will be removed both from the database and “All current services” sector.

Both the addition and the deletion update the current maintenance service selector automatically and the user does not need to reload the page in order to do it manually. Same applies to the deletion, if the user selects any maintenance service at the “Make Agreement” sub-tab and then deletes it at the “Add/remove maintenance service” sub-tab. The service will be not only removed from the database and selector, but also from the preview document.

### 5.2.4 Project manager, “Add/remove additional service” sub-tab

The add/remove additional service tab contains all the information regarding additional services. It is absolutely the same as “Add/remove maintenance service” sub-tab with an exception that here the user adds/removes services to/from additional service list. In other words, the user is modifying another table in database.

### 5.2.5 Project manager, “Manage clients” sub-tab

The manage clients tab contains all the information regarding clients. Page is divided into three parts: Add new client, All current clients and Client info. This sub-tab is shown in Figure 14.

**Clients**

**Add new client**

Company name

Owner

Bank account

Address

Registration number

**All current clients**

- \* Adrey IT
- \* Adrey2 IT
- \* Blackout OU
- \* Fairy IT
- \* IT Revolution OU

**Client info**

Company Name: **Fairy IT**  
Owner: **Heart Dagneel**  
Bank Account: **EE2244556677**  
Address: **Eesti 20-0**  
Registration number: **14525**

Figure 14: Manage clients sub-tab

The first part of the sub-tab is “Add new client”. In order to add a new client, the user must fill every input field regarding the client’s company. Those fields are company name, owner’s name, owner’s bank account, company address and company registration number. In case the user forgets to fill any field, it will be highlighted with a red outline. An example can be seen in Figure 15.

**Add new client**

Company name

Owner

Bank account

Address

Registration number

Please fill every input field

Figure 15: Red outline when trying to add client without vital data

If every field is filled and the “Add” button is pressed, the web application will verify whether the client is already in the database. In case it is, the application will not add the client once again and will notify the user about that. Otherwise the client will be added to the database and to “All current clients” part.

The second part of the sub-tab holds the client list and the button “More”, upon the click on which detailed information about this client appears.

The third part of the sub-tab holds the detailed information of the client. This information includes company name, owner's name, owner's bank account, company address and company registration number. All the information is shown in bold.

### 5.2.6 Project manager, “Show all contracts” sub-tab

The show all contracts sub-tab contains a list of all the stored documents in the server, created using the web application. This tab is divided into two parts: a document list and a scrollable preview window. This tab is shown in Figure 16.

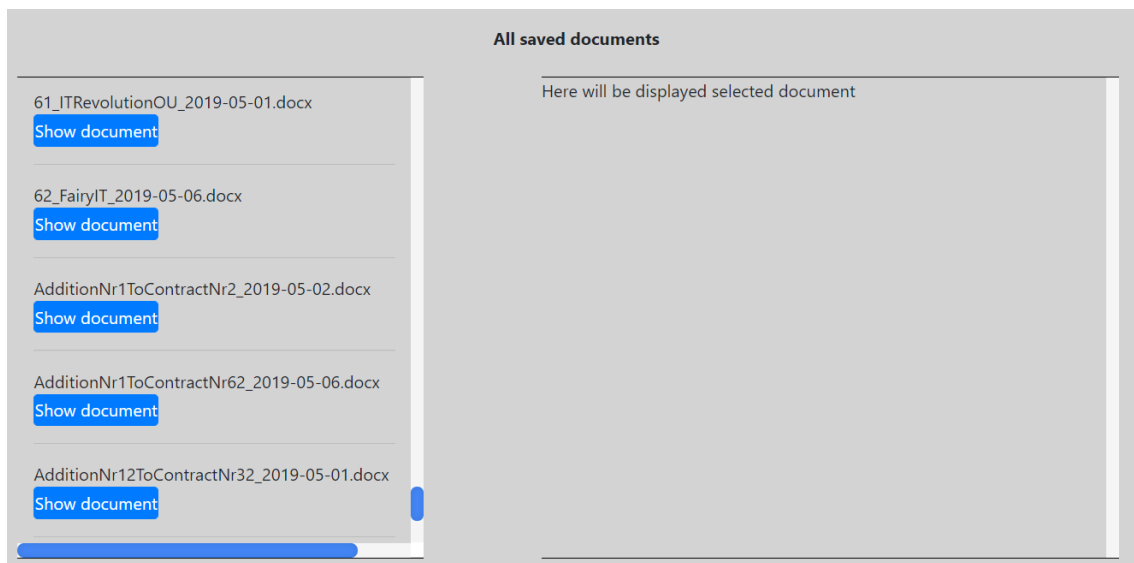


Figure 16: Show all contracts sub-tab

Here the user can select any of the stored contract and press a “Show document” button. After that, the selected document's content will be displayed on the second part of the page. As an example shown in Figure 17 is selected document `62_FairyIT_2019-05-06.docx`.

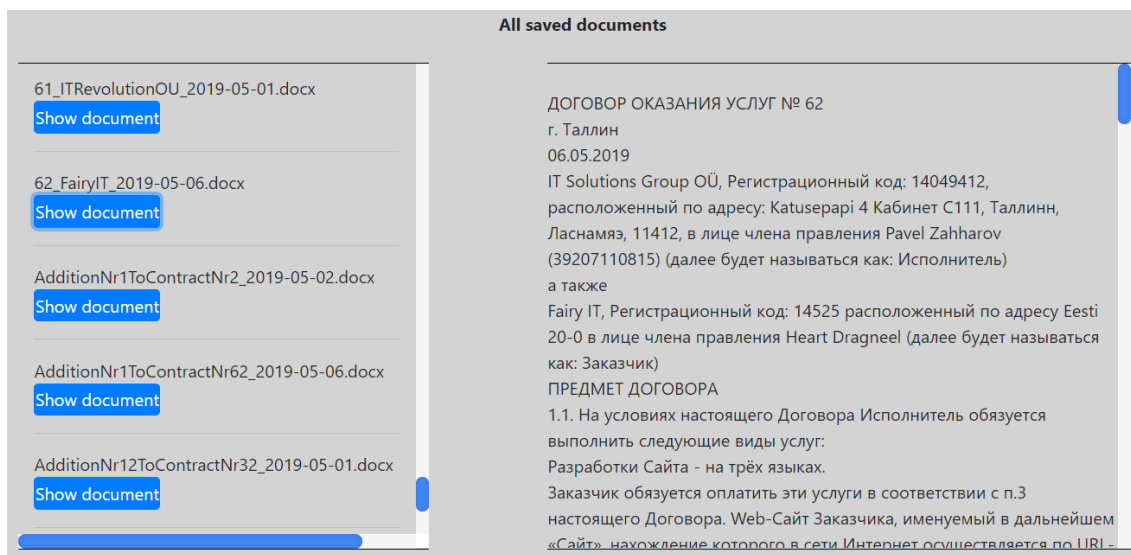


Figure 17: An example of the display of the selected contract within the web application

## 5.3 Code interaction with documents and database

### 5.3.1 How a contract template turns into a document ready for signing

To begin with, the web application has template files stored in the server. After the user enters the main working page, template files are read by the script. Due to the fact, that docx file is a zip archive with xml files in it, it is required to work with main document content with the name document.xml [4]. Every word is wrapped in XML tags as is shown in Figure 18.

```

<w:t xml:space="preserve"> к Договору оказания услуг № </w:t>
</w:r>
<w:proofErr w:type="spellStart"/>
- <w:r w:rsidR="00A34B09">
  - <w:rPr>
    <w:b/>
    <w:lang w:val="et-EE"/>
  </w:rPr>
  <w:t>numOfAgreement</w:t>
</w:r>
<w:proofErr w:type="spellEnd"/>
</w:p>

```

Figure 18: An example of document.xml inside .docx file

In order to show the document to the user in ordinary view, the web application first has to strip the document of the XML tags. To achieve this, PHP function `nl2br` is required [3].

The next step is inserting HTML line breaks before all the newlines. Here the PHP function *strip\_tags* is needed [3]. After those manipulations the document transforms into a plain string and is ready to be displayed inside the web application.

After the data is input and the “Download” button is pressed, the application once again opens the corresponding template, creates a copy with required name and modifies its document.xml using *str\_replace* function [3]. This function searches for every special word described in 4.3 and replaces them with input text. After that, the document is saved to the server and application initiates upload of the document to the user’s device.

### **5.3.2 How does the document get its serial number**

After the document is saved, its name is also added to the database. Table containing documents has two columns: document number and document name. Document number is autoincrement and is set automatically. During the document modification, the application takes last document number from the table and increases it by one for further usage both in the document preview and document modification. A database table is shown in Figure 19.

	123 docid	ABC doc
43	43	FairyIT_2019-04-30
44	44	ITRevolutionOU_2019-04-30
45	45	ITRevolutionOU_2019-04-30
46	46	FairyIT_2019-04-30
47	47	ITRevolutionOU_2019-05-01
48	48	ITRevolutionOU_2019-05-01
49	49	ITRevolutionOU_2019-05-01
50	50	ITRevolutionOU_2019-05-01
51	51	ITRevolutionOU_2019-05-01
52	52	ITRevolutionOU_2019-05-01
53	53	FairyIT_2019-05-01
54	54	Adrey2IT_2019-05-01
55	55	ITRevolutionOU_2019-05-01
56	56	ITRevolutionOU_2019-05-01
57	57	ITRevolutionOU_2019-05-01
58	58	ITRevolutionOU_2019-05-01
59	59	ITRevolutionOU_2019-05-01
60	60	ITRevolutionOU_2019-05-01
61	61	ITRevolutionOU_2019-05-01
62	62	FairyIT_2019-05-06

Figure 19: Document table data

### 5.3.3 How does the document get creation date

Preview document's date is updated every time the user inserts new values into input fields. This is done to prevent date discrepancy if document is created at midnight. Preview document's date updates using JavaScript object Date and its functions *getDate*, *getMonth* and *getFullYear* [5]. Once the user hits "Download" button, creation date is updated once again using PHP built-in function [3].

### 5.3.4 How is a new client added to the database

As user fills every input field and presses the "Add" button at "Manage Clients" sub-tab, the web application checks whether the client is already in the database. If not, application adds client to the clients table. This table is shown in Figure 20.

	ABC company	ABC owner	ABC ownersAccount	ABC address	ABC regnum
1	Adrey IT	Andrei	US2222222222	Gerr Strasse 5	14789
2	Blackout OU	Khaled Jahani	UZ123456789	Night 1000-1	12548
3	Fairy IT	Heart Dragneel	EE2244556677	Eesti 20-0	14525
4	IT Revolution OU	Vladimir Lenin	EE228228228228	Voidujooksu 13-3	13625
5	Tuul	Tarvo Tuul	EE111111111111	Sopruse 12	14789

Figure 20: Clients table

### 5.3.5 How are services added and removed

After the user fills the text area at the “Add/remove maintenance service” or “Add/remove additional service” and presses the “Add” button, the web application checks whether such service already exist in the database. If not, the service is added to the corresponding table. Maintenance services table is shown in Figure 21 and Additional services table is shown in Figure 22.

	ABC service
1	Sed pretium sodales tellus et commodo. Vestibulum ullamcorper ac arcu sed tempor.
2	Donec a libero varius, finibus tortor non, auctor nisi.
3	Lorem ipsum dolor sit amet, consectetur adipiscing elit.
4	Nulla facilisi. Integer ut accumsan nibh, non fermentum nisi. Phasellus sit amet nibh a magna pulvinar ultricies.
5	Praesent pretium ornare arcu, non imperdiet felis pharetra non.
6	Quisque tempus facilisis ante at malesuada.

Figure 21: Maintenance services table

	ABC service
1	Consectetur adipiscing elit
2	Dolor sit
3	Lorem ipsum

Figure 22: Additional services table

In case the user wants to remove service from the table and presses the “Delete?” button, web application erases this service from the database. As an example, erasion of the “Dolor sit” service from “Additional services” table is shown in Figure 23.




	ABC service 
1	Consectetur adipiscing elit
2	Lorem ipsum

Figure 23: Additional services table after "Dolor sit" service was deleted using the web application

## 6 Web application design

Design of the web application is created using Bootstrap. That HTML, CSS and JavaScript library allows to build pages using a grid system. Each tab is divided into the 12 parts, which allows to separate page's functional into different blocks. As an example, the "Make agreement" sub-tab shown in Figure 24. Here, the main block is divided into two blocks with equal size, both having 6 parts.

The screenshot shows a web application interface titled "Select type of contract". At the top, there are four blue buttons: "Sale contract", "Rent contract", "Maintenance contract", and "Additions to contract". Below these are three more blue buttons: "In Russian", "In Estonian", and "In English". The main content area is divided into two columns. The left column contains a form with the following fields: "User: Blackout OU" (with a dropdown arrow), "Company name" (with the value "Blackout OU"), "Owner" (with the value "Khaled Jahani"), "Bank account" (with the value "UZ123456789"), "Address" (with the value "Night 1000-1"), "Registration number" (with the value "12548"), and "Owner's email". The right column contains text: "Website Usage Agreement NO. 63", "This website usage agreement is concluded on in Tallinn 7.5.2019 .", "(1)IT Solutions Group OÜ, registry code 14049412, based in Katusepapi 4, office C111, Tallinn, Lasnamäe, 13816, represented by board member Pavel Zahharov (hereinafter: Provider),", "and", "(2)Blackout OU, registry code 12548 , based in Night 1000-1, represented by CEO Khaled Jahani (hereinafter: User), hereinafter referred to as a Party or jointly as Parties agree to the following:", "Object of the agreement", "Provider develops a custom website (hereinafter the Website) and set the domain to DOMAIN (hereinafter referred to as Domain).", "The Website description and accompanying services are set in Appendix 1."

Figure 24: An example of elements positioning on "Make agreement" sub-tab

The "Manage clients" sub-tab's main block shown in Figure 25 is divided into three blocks, "Add new client" block has three parts, "All current clients" has six parts and "Client info" has three parts.

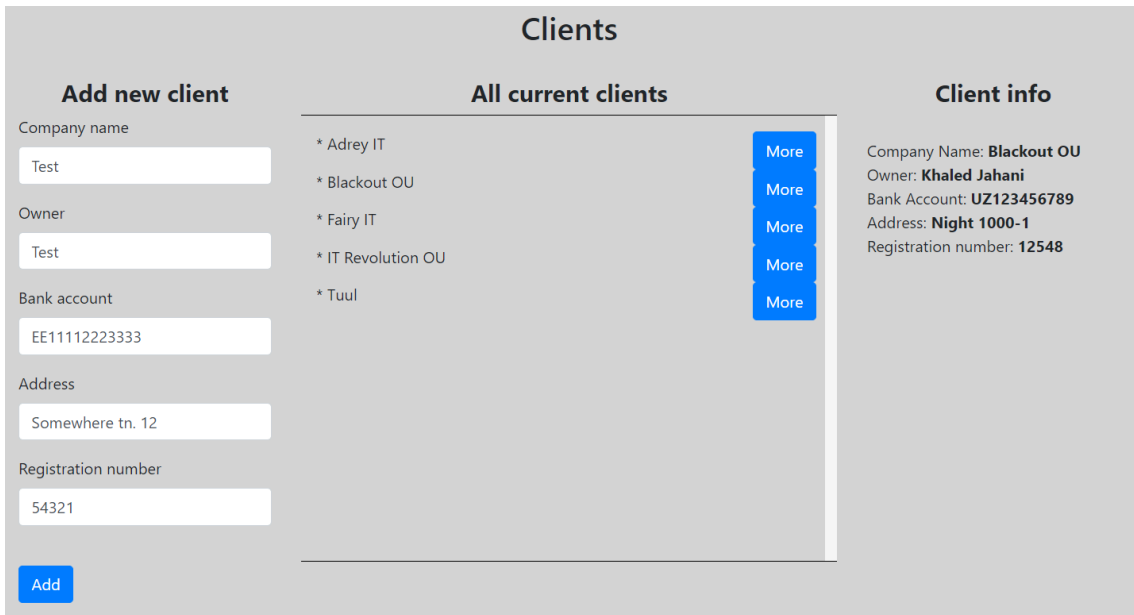


Figure 25: An example of element positioning on "Manage clients" sub-tab

In addition, navigation on the left side of the web page is toggleable, meaning that the user can collapse the navigation in order to increase the workspace as is shown in Figure 26 and Figure 27.

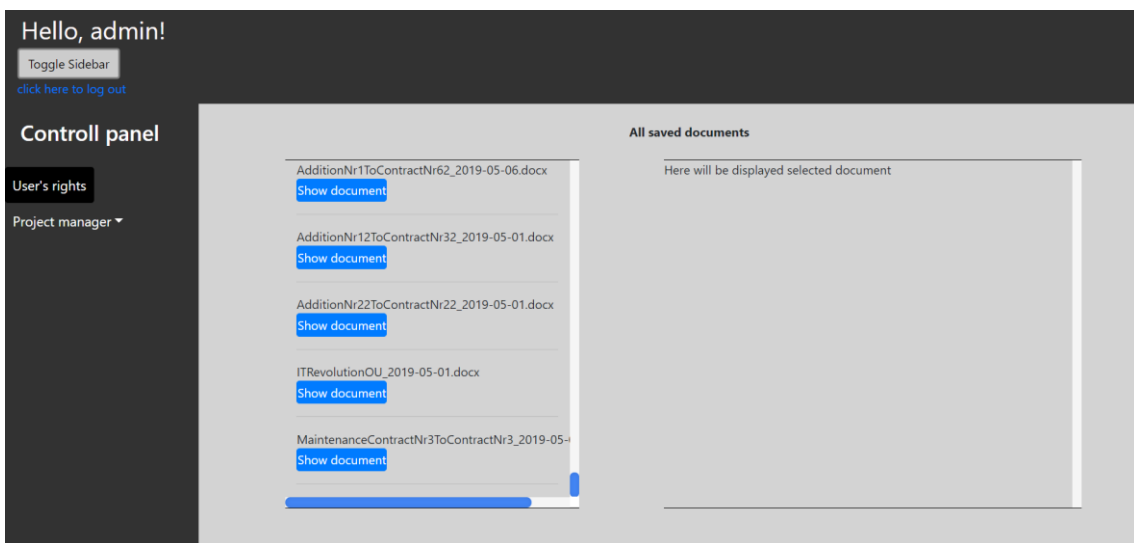


Figure 26: Web page with sidebar toggle on

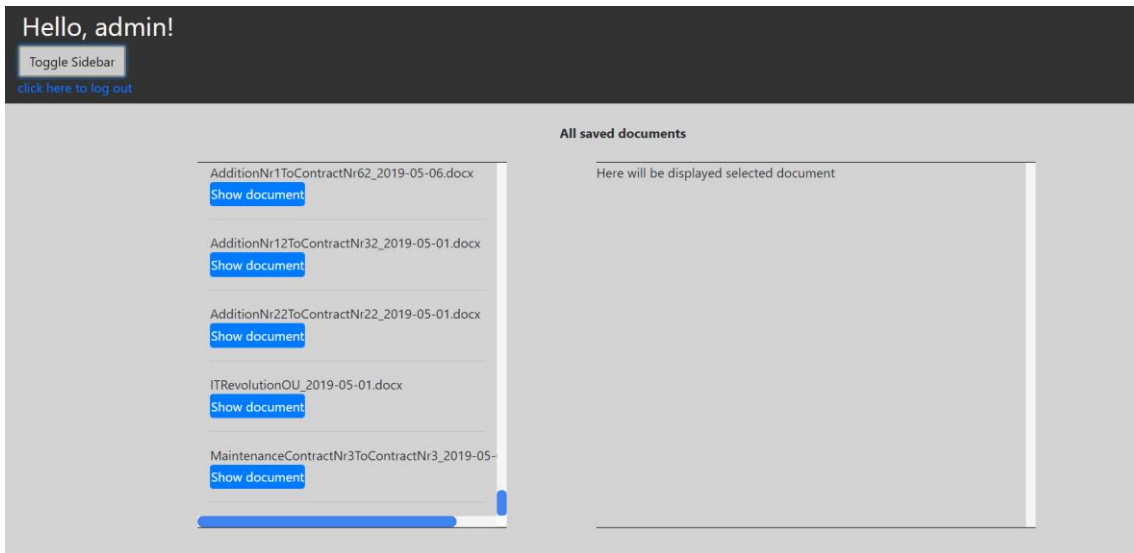


Figure 27: Web page with sidebar toggle off

After the job is done and the user wants to sign off, the user must press the “click here to log out” text, located below the “Toggle Sidebar” button. After that, the session will be cleared and user will be redirected back to the login page.

## 7 Summary

The aim of this work was to create an administrative zone where project manager could create contract documents spending less time and effort than creating documents manually. The result is a web application where a project manager can create documents, manage clients and expand services list. The main topics covered were document creation, rights update, new client addition to the clients base, new service addition to the services list.

The main goal was to create a document modification service, where the user could use stored information, such as client data and services, to furthermore speed up the process. The implemented solution is fast and allows user to bypass minor inconveniences such as finding the serial number of the last created document and the necessity to input same client's information to the different documents. The result was a reliable document modification system where the copy of the documents are stored in the server.

The above mentioned goal were implemented into a web application. While the project manager's duties can be fulfilled, the web application is far from completion. In the future it will include other features such as automated audit, call center, where user can send new service offers to the clients and much more. Overall, the prototype is a good first step and marks the beginning of a great workspace.

## References

- [1] Forest Admin [WWW] <https://www.forestadmin.com/> (15.05.2019)
- [2] Word Document (DOCX) [WWW]  
[https://www.forensicswiki.org/wiki/Word\\_Document\\_\(DOCX\)](https://www.forensicswiki.org/wiki/Word_Document_(DOCX)) (05.05.2019)
- [3] PHP Manual [WWW] <https://www.php.net/manual/en/> (06.05.2019)
- [4] An informal introduction to DOCX [WWW] <https://www.toptal.com/xml/an-informal-introduction-to-docx> (07.05.2019)
- [5] JavaScript Get Date Methods [WWW]  
[https://www.w3schools.com/js/js\\_date\\_methods.asp](https://www.w3schools.com/js/js_date_methods.asp) (07.05.2019)