

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Иркутский национальный исследовательский технический университет»

Кафедра автоматизированных систем
(наименование кафедры)

О Т Ч Ё Т

о прохождении производственной практики
в экспертно-криминалистическом отделе Следственного управления
Следственного комитета по Иркутской области
(наименование предприятия)

6.000.00.09

Студента Костылева Дениса Алексеевича
гр. АСУб-12-1
(ФИО, группа, подпись)

Руководитель практики от кафедры
Бучнев О. С., доцент
(ФИО, должность, подпись)

Руководитель практики от предприятия
Портнягин Д. Г., эксперт
(ФИО, должность, подпись)

Допущен к защите _____
(ФИО, должность, подпись)

Оценка по практике _____
(неуд., удовл., хор., отл.)

(ФИО, подпись, дата)
Содержание отчета на 24 стр.
Приложение к отчету на 32 стр.

Иркутск 2015 г.

Содержание

Введение	3
1 Описание предприятия	4
1.1 Организационная структура предприятия	4
1.2 Основные направления деятельности	5
1.3 Функциональные обязанности сотрудников отдела	6
1.4 Основные задачи	7
2 Индивидуальное задание	9
2.1 Постановка задачи	9
2.2 Используемые инструменты	9
2.2.1 Javascript, Node.js, node-webkit, JSON	9
2.2.2 HTML5, CSS3, SCSS	10
2.2.3 Python 3.4.3	10
2.2.4 Git	10
2.3 Описание работы программы	11
2.3.1 Загрузка базы DEF-кодов	18
2.3.2 Реализация кросс-платформенности	21
Заключение	22
Список использованных источников	23
Приложение А: Примеры базы в форматах html и json	25
Приложение Б: Листинг <code>main.js</code>	28
Приложение В: Листинг <code>main.scss</code>	50

ВВЕДЕНИЕ

Производственная практика была пройдена в период с 01.07.2015 по 10.07.2015 в экспертно-криминалистическом отделе Следственного управления Следственного комитета России по Иркутской области. Руководитель — эксперт Дмитрий Геннадьевич Портнягин.

В состав задач на производственную практику входили следующие:

- изучить организационную структуру предприятия;
- ознакомиться с функциональной структурой предприятия;
- принять непосредственное участие в выполнении задач отдела, а именно в автоматизации некоторых из них.

Основными аргументами в пользу актуальности выбранной практики являются:

- наглядность использования прогрессивных методов автоматизации работы предприятия;
- практическое применение навыков, полученных в процессе обучения;
- место предприятия в городе и области и его масштабность, способствующая приобретению современных знаний, необходимых для работы в данной сфере.

Необходимыми условиями успешного прохождения производственной практики являются: подробное ознакомление с принципами работы выбранного отдела и предприятия в целом; понимание структуры используемого оборудования и ПО и готовность к применению полученных знаний на практике; совершенствование собственных знаний, касающихся выбранной специальности.

1 Описание предприятия

Следственный комитет Российской Федерации — федеральный государственный орган в России, осуществляющий полномочия в сфере уголовного судопроизводства. Является государственной военизированной организацией. Наделен правом ведения предварительного следствия и дознания.

Экспертно-криминалистический отдел является структурным подразделением Следственного управления следственного комитета России по Иркутской области и производит судебные экспертизы (исследования) в соответствии с требованиями и положениями действующего законодательства и экспертных методик.

СК РФ образован на базе Следственного комитета при прокуратуре Российской Федерации[1]. Начал свою деятельность 15 января 2011 года. Началом деятельности экспертно-криминалистического отдела является 20 октября 2014 г.[2]

1.1 Организационная структура предприятия

Следственный комитет России находится в непосредственном подчинении Президента РФ. Руководитель — Бастрыкин Александр Иванович. Главой Следственного управления СК РФ по Иркутской области является Бунёв Андрей Юрьевич. Экспертно-криминалистический отдел возглавляет Себякин Алексей Геннадьевич.



Рисунок 1.1 — Организационная структура Следственного комитета РФ

На рисунке выделена группа подразделений, к которой относится

Следственное управление Следственного комитета РФ по Иркутской области в состав которого входит экспертно-криминалистический отдел.

В состав отдела входят 1 заместитель, 3 старших эксперта, 5 экспертов. Распределение специализаций между старшими экспертами и экспертами отдела и их взаимозаменяемость осуществляются в соответствии с распоряжением руководителя Следственного управления №29-246р.[3]

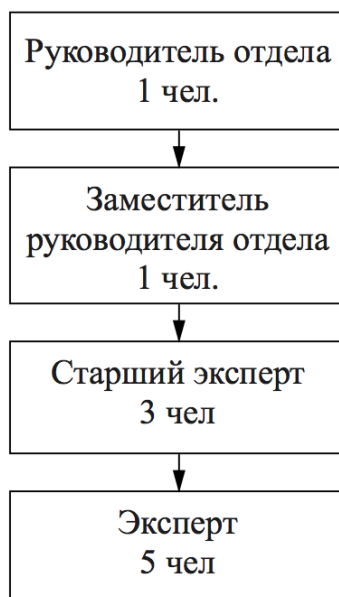


Рисунок 1.2 — Функциональная структура экспертно-криминалистического отдела Следственного управления Следственного комитета РФ по Иркутской области

1.2 Основные направления деятельности

Экспертно-криминалистический отдел Следственного управления Следственного комитета РФ по Иркутской области занимается широким спектром исследований. Главными можно выделить следующие виды экспертиз:

1. компьютерно-техническая;
2. информационно-аналитическая;
3. видеотехническая;
4. лингвистическая;
5. фоноскопическая;
6. экономическая;
7. психофизиологическая экспертиза с использованием полиграфа;
8. иные экспертизы, в соответствии с организационно-распорядительными документами СК России и Следственного управления.

1.3 Функциональные обязанности сотрудников отдела

За каждым сотрудником экспертно-криминалистического отдела закреплена одна или несколько обязанностей, из представленных ниже:

1. Проведение компьютерно-технических экспертиз и исследований. При производстве экспертиз и исследований решаются следующие задачи:
 - диагностика аппаратных средств и их составных частей в целях установления их технического состояния;
 - исследование носителей с целью поиска заданной информации;
 - поиск латентной и закодированной информации на носителях информации;
 - исследование фрагментов информации в целях ее реконструкции;
 - исследование компьютерных систем для установления возможности решения конкретных преступных задач;
 - исследование программ для ЭВМ и баз данных для определения их предназначения, работоспособности и потребительских свойств.
2. Проведение информационно-аналитических экспертиз и исследований. При производстве экспертиз и исследований решаются следующие задачи:
 - анализ детализаций телефонных соединений абонентских номеров;
 - анализ массивов данных систем распознавания государственных регистрационных знаков автотранспортных средств;
 - анализ массивов данных систем обслуживания магнитных пластиковых карт.
3. Проведение видео-технических экспертиз и исследований. При производстве экспертиз и исследований решаются следующие задачи:
 - улучшение качества видеозаписи;
 - идентификация средств видеозаписи;
 - установление признаков внесения изменений в видеозапись.
4. Проведение лингвистических экспертиз и исследований. При производстве экспертиз и исследований решаются следующие задачи:
 - семантический анализ текстов;
 - лингвистический анализ продуктов речевой деятельности.
5. Проведение фоноскопических экспертиз и исследований. При производстве экспертиз и исследований решаются следующие задачи:
 - идентификация личности по голосу и устной речи;
 - идентификация средств звукозаписи, идентификация источников звуков;

- установление дословного содержания разговора;
 - определение количества участников разговора;
 - установление признаков внесения изменений в аудиозапись.
6. Проведение экономических экспертиз и исследований. При производстве экспертиз и исследований решаются следующие задачи:
- исследование отражения в первичных учетных, иных первичных документах, регистрах учета (бухгалтерского и налогового), отчетности (бухгалтерской и налоговой) фактов финансово-хозяйственной деятельности, имущества и обязательств исследуемого лица;
 - исследование соответствия порядка отражения фактов финансово-хозяйственной деятельности, имущества и обязательств в первичных учетных, иных первичных документах, регистрах учета (бухгалтерского и налогового), отчетности (бухгалтерской и налоговой), реализованного хозяйствующим субъектом, правилам бухгалтерского и налогового учета;
 - исследование финансового состояния хозяйствующего субъекта;
 - исследование соблюдения принципов кредитования.
7. Проведение психофизиологических экспертиз и исследований. Целью проведения психофизиологических экспертиз и исследований является проверка достоверности информации, сообщаемой обследуемым лицом, при решении задач:
- уголовного судопроизводства;
 - отбора кадров при приеме граждан на службу (работу) в СК России;
 - служебной деятельности при проведении служебных проверок в системе СК России.

1.4 Основные задачи

К задачам отдела относятся:

- производство судебных экспертиз и экспертных исследований в соответствии с постановлениями и поручениями должностных лиц Следственного управления, наделенных правом назначения судебных экспертиз в соответствии с действующим законодательством Российской Федерации;
- проведение научно-исследовательских работ по теоретическим и методическим проблемам в области судебной экспертизы;
- создание и совершенствование методик экспертного исследования, а также внедрение новых методов и методик в судебно-экспертную практику;

- осуществление работы, направленной на повышение квалификации и подготовку экспертных кадров;
- оказание методической помощи сотрудникам правоохранительных органов и судам по вопросам назначения и производства судебных экспертиз и экспертных исследований, проводимых в ЭКО;
- производство судебных экспертиз и экспертных исследований по заданиям должностных лиц правоохранительных органов и судов в случаях подбора необходимых материалов для проведения обобщения экспертной практики, научно-исследовательской и методической работы.

2 Индивидуальное задание

Основной задачей производственной практики являлась разработка программы для получения списка операторов с соответствующими им DEF-кодами и диапазонами номеров и выводе его в удобном для дальнейшего использования экспертом формате.

2.1 Постановка задачи

Задачу можно разделить на несколько подзадач:

1. сформировать запрос с необходимой информацией и получить полный список с url `http://mtt.ru/defcodes`;
2. выбрать из полученного списка необходимые область, оператора и def-код;
3. преобразовать диапазон в заданный шаблон;
4. Результат работы программы вывести на экран, в текстовые файлы или файл электронных таблиц.

Результат работы программы должен быть выведен в формате, представленном ниже.

- Текстовые файлы: В каталоге с датой и временем вывода¹ создать каталоги с названием области, в каждом из которых сформировать .txt-файл с названием оператора, с вложенными шаблонами диапазонов для конкретных области и оператора.
- Файл таблицы: В каталоге с датой и временем вывода создать файл электронной таблицы с названиями листов, соответствующим областям, с вложенными шаблонами диапазонов для всех операторов конкретной области.

2.2 Используемые инструменты

При разработке данной программы использовалось несколько языков программирования, фреймворков и библиотек.

2.2.1 Javascript, Node.js, node-webkit, JSON

Основная часть программы, объединяющая все модули приложения и обеспечивающая взаимодействие с пользователем разработана с использованием языка программирования Javascript.

¹Уникальное имя каталога необходимо для предотвращения конфликтов при выводе

Выбор данной технологии был обусловлен возможностью разработки кросс-платформенных приложений. Данное свойство приложения обеспечивается фреймворком node-webkit[4]. Он обеспечивает среду выполнения для HTML+Javascript приложений. Преимуществом данной технологии является отсутствие требований к установленному программному обеспечению пользователя. Все необходимые компоненты присутствуют в дистрибутиве программы.

К недостаткам данного подхода можно отнести большой размер распространяемой программы (115 Мб для Mac OS X и 95 Мб для Windows).

Для взаимодействия с операционной системой (использование файлов, выполнение сторонних приложений, и т. п.) используется фреймворк Node.js[5].

Для хранения различных данных на диске, а также для обмена информацией между частями программы был выбран формат JSON[6]. Выбор был обусловлен тем, что несмотря на тот факт, что стандарт основан на языке Javascript, он получил широкое распространение и, как следствие, поддержку большим числом языков программирования.

2.2.2 HTML5, CSS3, SCSS

Для верстки графического интерфейса пользователя использовался язык разметки HTML5.

Стили были разработаны с использованием препроцессора SCSS[7], которые в свою очередь были скомпилированы в CSS3 и подключены к HTML-файлу. В дополнение к стилям программы в SCSS-файл были интегрированы стили сторонних библиотек.

2.2.3 Python 3.4.3

В связи с политикой безопасности web-стандартов[8], в языке Javascript существует ограничение на доступ к данным, находящимся на другом url. Так как загрузка базы DEF-кодов осуществлялась с домена, доступа к которому осуществлялся исключительно через API провайдера, для решения данной задачи был создан Python-скрипт[9].

2.2.4 Git

В качестве системы управления версиями использовался интернет-сервис GitHub[10].

Данный сервис позволяет хранить все версии приложения, а также распространять открытый исходный код.

2.3 Описание работы программы

Запуск программы осуществляется через исполняемый файл `DEFcodes.app` в операционной системе Mac OS X, `DEFcodes.exe` в ОС Windows. Этот исполняемый файл запускает среду выполнения `node-webkit`, которая выполняет первоначальную настройку согласно файлу `package.json`² (Листинг 2.1).

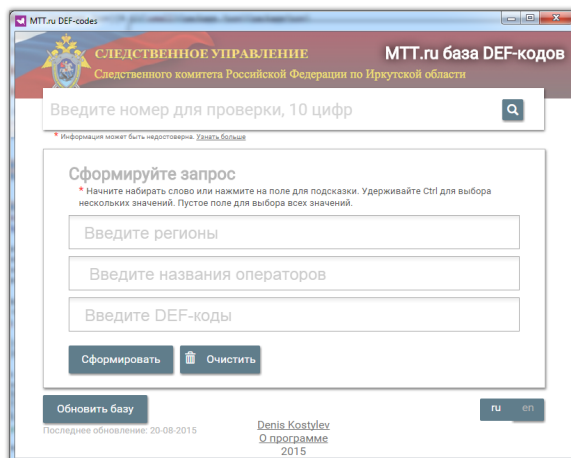
Листинг 2.1— `package.json`

```

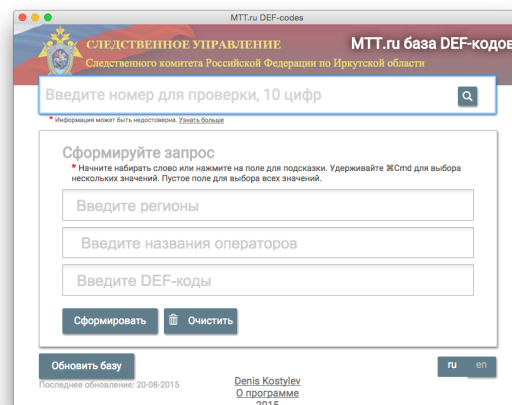
1 {
2   "name": "defbase",
3   "version": "0.4.4",
4   "description": "DEF codes parser",
5   "main": "index.html",
6   "lang": "ru",
7   "repository": {
8     "type": "git",
9     "url": "git+https://github.com/Kostydenis/practice2015.2
10  git"
11  },
12  "keywords": [
13    "def",
14    "html",
15    "parse"
16  ],
17  "author": "Denis Kostylev",
18  "license": "GPL-2.0",
19  "bugs": {
20    "url": "https://github.2
21    com/Kostydenis/practice2015/issues"
22  },
23  "homepage": "https://github.2
24  com/Kostydenis/practice2015#readme",
25  "window": {
26    "title": "MTT.ru DEF-codes",
27    "toolbar": false,
28    "frame": true,
29    "icon": "_pics/icon.png",
30    "width": 800,
31    "height": 500,
32    "position": "mouse",
33    "min_width": 600,
34    "min_height": 200
35  },
36  "dependencies": {
37    "normalize.css": "^3.0.3"
38  }
39 }
```

Точкой входа в программу является файл `index.html` (Листинг 2.2). Базовый шаблон, в который входит основная верстка интерфейса: контейнеры, заголовки, и т. п., инициализируется с помощью Javascript-функции `initBaseTemplate()` (Листинг 2.3).

²Файл `package.json` един для различных ОС



(a) Windows 7



(б) Mac OS X 10.10

Рисунок 2.3 — Окно программы при запуске программы

На рисунке 2.3 дизайн программы был спроектирован в соответствии с концепцией Material Design[11].

Были использованы иконки из набора Twitter Bootstrap и FontAwesome. Иконка приложения — логотип сайта <http://mtt.ru>.

Листинг 2.2— index.html

```

1 <html>
2 <head>
3   <link rel="stylesheet" href="_css/normalize.css">
4   <link rel="stylesheet" href="_css/main.min.css">
5   <link rel="stylesheet" href="_css/nanoscroll.css">
6   <meta charset="UTF-8">
7   <script src="_js/jquery.min.js"></script>
8   <script src="_js/jquery.mask.min.js"></script>
9   <script src="_js/main.min.js"></script>
10  <script src="_js/jquery.nanoscroll.min.js"></script>
11  <script src="_js/chosen.jquery.min.js"></script>
12 </head>
13 <body>
14   <!-- there's nothing here -->
15   <!-- Content generates by script -->
16 </body>
17 </html>

```

Листинг 2.3— initBaseTemplate

```

1 function initBaseTemplate() {
2
3   var tip = "";
4   if (os.platform() === 'darwin') {
5     tip = '<p class="input_tip">' + lang.input_tip_mac + '</>';
6     /p>';
7   } else {
8     tip = '<p class="input_tip">' + lang.input_tip_win + '</>';
9     /p>';
10  }

```

```

11     var langbuttons;
12     if (chosenLang === 'ru') {
13         langbuttons =
14             '<div id="btn_en" class="switch_lang_btn 2
15             not_pressed">en</div>' +
16             '<div id="btn_ru" class="switch_lang_btn">ru<2
17             /div>'
18     } else {
19         langbuttons =
20             '<div id="btn_en" class="switch_lang_btn">en<2
21             /div>' +
22             '<div id="btn_ru" class="switch_lang_btn 2
23             not_pressed">ru</div>'
24     }
25
26     $('body').empty();
27     $('body').append(
28         '<div id="modal_wrapper" class="modal_wrapper
hide"><2
29         /div>' +
30         '<div id="modal" class="modal preinit"></div>' +
31         '' +
33         '<header>' +
34             '' +
35             '<h1></h1>' +
36         '</header>' +
37
38         '<div class="content">' +
39             '<div id="error" class="hide">' +
40             '</div>' +
41
42             '<div class="block_number_check preinit">' +
43                 '<input type="text" id="input_number_check"
44                 />' +
45                 '<button id="button_number_check"><span 2
46                 class="glyphicon glyphicon-search" aria-2
47                 hidden="true"></span></button>' +
48                 '<p class="NB" id="note_mnp"></p>' +
49             '</div>' +
50
51             '<div class="form_request preinit">' +
52                 '<h2 id="header_form_request"></h2>' +
53                 tip +
54                 '<select id="input_locations" class="chosen-
55                 select" multiple >' +
56                 '</select>' +
57                 '<select id="input_providers" class="chosen-
58                 select" multiple >' +
59                 '</select>' +
60                 '<select id="input_defs" class="chosen-2
61                 select" multiple >' +
62                 '</select>' +
63                 '<button id="btn_form_request"></button>' +
64                 '<button id="btn_clear_form_request"><2
65                 /button>' +
66             '</div>' +
67
68             '<div class="update">' +
69                 '<button id="btn_update" ></button>' +
70                 '<span class="last_update">2015-08-3</span>'
71     )

```

```

71         +
72         '</div>' +
73         '<div class="switch_lang">' +
74         langbuttons +
75         '</div>' +
76         '</div>' +
77         '</div>' +
78         '<footer>' +
79         '<p><a id="btn_author"></a></p>' +
80         '<p><a id="btn_about"></a></p>' +
81         '<p id="lbl_year">2015</p>' +
82         '</footer>'
83     );
84 }
85 }
86 }

```

В данной программе реализована мультиязычность, это значит, что все используемые в программе тексты загружаются из внешних файлов. В настоящий момент используются два языка: русский (основной, загружается при запуске) и английский. Языковые файлы находятся в каталоге ./lang — ru.json и en.json для русского и английского языка соответственно. Файл русского языка приведен ниже (Листинг 2.4)

Листинг 2.4— Файл перевода

```

1  {
2  "title": "МТТ.ru база DEF-кодов",
3  "find": "Найти номер",
4  "note_mnp": "Информация может быть 2
5  недостоверна.",
6  "learnmore": "Узнать больше",
7  "placeholder_input_number_check": "Введите 2
8  номер для проверки, 10 цифр",
9  "header_form_request": "Сформируйте 2
10 запрос",
11 "placeholder_input_locations": "Введите 2
12 регионы",
13 "placeholder_input_providers": "Введите 2
14 названия операторов",
15 "placeholder_input_defs": "Введите DEF-коды",
16 "btn_form_request": "Сформировать",
17 "btn_clear_form_request": "Очистить",
18 "btn_update": "Обновить базу",
19 "error_number_not_found": "Номер не найден",
20 "export_to_txt": "В текстовый файл",
21 "export_to_excel": "В excel-файл",
22 "successful_export": "Успешно 2
23 экспортировано",
24 "unsuccessful_export": "Ошибка при 2
25 экспорте",
26 "input_tip_win": "Начните набирать 2
27 слово или нажмите на поле для 2
28 подсказки. Удерживайте Ctrl для 2
29 выбора нескольких значений. 2
30 Пустое поле для выбора всех 2
31 значений.",
32 "input_tip_mac": "Начните набирать 2
33 слово или нажмите на поле для 2
34 подсказки. Удерживайте Cmd для 2
35 выбора нескольких значений. 2
36 Пустое поле для выбора всех 2
37 значений.",
38 "base_last_update": "Последнее 2
39 обновление: ",

```

```

40 "youreoffline": "Подключение к 2
41 интернету отсутствует. 2
42 Проверьте подключение и 2
43 повторите попытку.",
44 "successful_update": "Успешно обновлено",
45 "unsuccessful_update": "Ошибка при 2
46 обновлении",
47 "needupdate": "База не обновлялась 2
48 больше месяца. ",
49 "gonnaupdate": "Обновить сейчас",
50 "currPath": "Сохраняю сюда: ",
51 "change_path": "Изменить",
52
53 "about_page": "Заметили ошибку? 2
54 Напишите об этом: ",
55 "about_manual": "Руководство 2
56 пользователя и другую полезную 2
57 информацию вы можете найти на 2
58 странице проекта: ",
59 "open": "Открыть",
60
61 "btn_about": "О программе"
62 }

```

После инициализации базового шаблона, программа формирует расширенный шаблон — добавляет надписи, кнопки и другие элементы управления, а также присваивает их событиям определенные функции. После чего начинается инициализация базы (Листинг 2.5), данные из JSON-файла преобразовываются в объекты и загружаются в оперативную память.

Также в этом методе осуществляется проверка даты обновления базы. При отсутствии обновлений базы в течение месяца, программа выводит соответствующее уведомление.

Листинг 2.5— base_init()

```

1 var base_init = function () {
2   log('Starting initialization of base');
3   $.getJSON('base/base.json', function (response) {
4
5     $(' .last_update' ).empty();
6     var baseModifyDate = fs.statSync('base/base.json').2
7     mtime;
8     $(' .last_update' ).append(lang.base_last_update+2
9     baseModifyDate.toLocaleDateString().replace(/[:]|,
10    |[/g, '-'));
11    if (dateDiffInDays(baseModifyDate, new Date()) > 30)
12    {
13      setError();
14    }
15    base = response;
16    locations = getLocations();
17    initAutocomplete();
18
19    <...>
20
21    $(' #input_locations' ).chosen();
22    $(' #input_providers' ).chosen();
23    $(' #input_defs' ).chosen();
24
25    });
26 }

```

Затем формируется список автодополнения и инициализируется плагин командой `<selector>.chosen()`.

Листинг 2.6— `initAutocomplete`

```

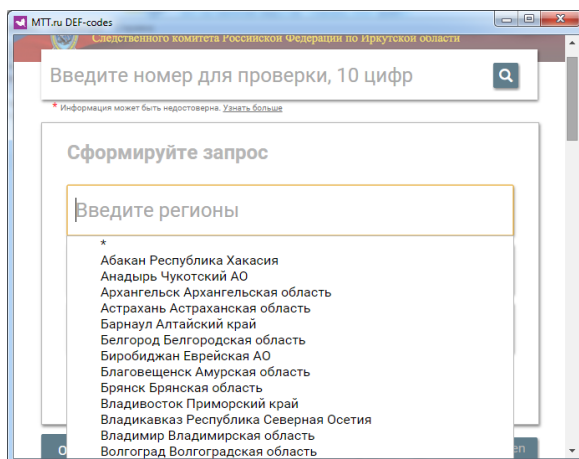
1 function initAutocomplete() {
2
3     locations = getLocations();
4     providers = [];
5     defs = [];
6     $('#input_locations').empty();
7     $('#input_providers').empty();
8     $('#input_defs').empty();
9
10    var idProv = 1;
11    var idDef = 1;
12
13    for (loc in locations) {
14        $('#input_locations').append('<option>' + locations[loc] + '</option>');
15
16        $('#input_providers').append('<optgroup label="' + locations[loc] + '" id="provider_optgroup' + idProv
17        + '">');
18        var tmpProviders = getProviders(locations[loc]);
19        for (prov in tmpProviders) {
20            providers.push(tmpProviders[prov]);
21
22            $('#input_providers #provider_optgroup' + idProv).append('<option>' + tmpProviders[prov] + '</option>');
23
24            var tmpDefs = getDefs(locations[loc], tmpProviders[prov]);
25            $('#input_defs').append('<optgroup label="' + tmpProviders[prov].replace(/"([\^"]+)/g, '<1>') + '" id="def_optgroup' + idDef + '">');
26            for (def in tmpDefs) {
27                defs.push(tmpDefs[def]);
28                $('#input_defs #def_optgroup' + idDef).append('<option>' + tmpDefs[def] + '</option>');
29            }
30            idDef++;
31        }
32        idProv++;
33    }
34
35    $('#input_locations').trigger('chosen:updated');
36    $('#input_providers').trigger('chosen:updated');
37    $('#input_defs').trigger('chosen:updated');
38 }

```

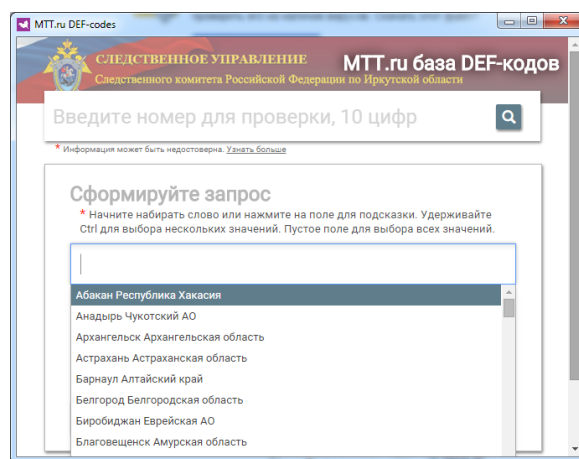
Для реализации фильтрации вывода сначала был использован стандартный элемент HTML для ввода текстовой информации — `<input>` с подключенным расширением jQueryUI `autocomplete`[12]. Однако, руководитель от предприятия внес соответствующие пожелания к использованию данного интерфейса. Выбранный модуль не позволял выбирать несколько значений из списка, не открывая его заново для каждого значения.

Впоследствии этот элемент был заменен на стандартный HTML-элемент `<select>`, с подключенной библиотекой `chosen`[13]. Данный

модуль решил вышеуказанную проблему — для выбора нескольких значений достаточно удерживать клавишу-модификатор (Ctrl для ОС Windows и Cmd для Mac OS).



(a) Windows 7



(б) Mac OS X 10.10

Рисунок 2.4 — Сравнение модулей для фильтрации данных на примере выбора области в ОС Windows

Также в приложении предусмотрен режим отладки. Если переменная `logging` равна `True`, то в каталоге `./logs` создается текстовый файл с именем даты и время запуска программы, в который записывается информация о работе приложения: свойства компьютера, на котором запущена программа; и ход работы приложения (пример файла журнала приведен ниже). При этом в стандартный поток вывода (в фреймворке `node-webkit` также встроена Javascript консоль, но она скрыта от пользователя в финальной версии программы) отправляются данные о длительности выполнения той или иной функции.

```
27-08-2015-16-59-57: Application is launched
27-08-2015-16-59-57: OS Platform: win32
27-08-2015-16-59-57: OS Type: Windows_NT
27-08-2015-16-59-57: OS Release: 6.1.7601
27-08-2015-16-59-57: OS Arch: ia32
27-08-2015-16-59-57: OS Hostname: Denis-PC
27-08-2015-16-59-57: Begin initialization
27-08-2015-16-59-57: Chosen language file is ru
27-08-2015-16-59-57: Extended template is initialized
27-08-2015-16-59-57: Starting initialization of base
27-08-2015-16-59-57: Begin parsing lang-JSON
27-08-2015-16-59-57: Basic template is initialized
27-08-2015-16-59-57: Starting initialization of basic template
27-08-2015-16-59-57: starting initialization of extended template
27-08-2015-16-59-57: base is opened
27-08-2015-16-59-57: locations is formed
27-08-2015-17-01-10: Export modal is opened
27-08-2015-17-01-18: Starting export to txt
27-08-2015-17-01-18: Folder 27-08-2015-17-01-18 is created
27-08-2015-17-01-20: Starting export to excel
```

```

27-08-2015-17-01-20: Excel file is created
27-08-2015-17-01-20: Sheet Абакан Республика Хакасия is created
27-08-2015-17-01-20: Sheet Анадырь Чукотский АО is created
<...>

```

Все файлы исходных кодов были минимизированы и к каждому был создан .tar-файл, что ускорило работу программы, но в то же время оставило возможность видеть исходный код в консоли программы.[14]

2.3.1 Загрузка базы DEF-кодов

Для загрузки базы с удаленного адреса была создана программа на языке программирования Python (Листинг 2.8).

В процессе анализа работы сервиса был выявлен формат запроса для получения базы с url `http://mtt.ru/defcodes`. Он имеет следующий вид (Листинг 2.7)

Листинг 2.7— Формат запроса

```

1 data = {
2     'g': 'mtt',
3     'm': 'def_codes',
4     'a': 'processRequest',
5     'def': "*",
6     'def_number': "",
7     'area': "*",
8     'operator': "*",
9     'standard': "*",
10    'date': dwnldate
11 }

```

Отправление запроса происходит методом `post`, который содержится в модуле `requests`. Данный оператор принимает два аргумента: адрес и параметры запроса, указанные выше.

Ответ от сервера приходит в формате JSON, который содержит два поля:

1. `'status'` — состояние ответа (`success` — данные успешно получены; `error` — возникла ошибка; `captcha` — слишком много запросов с IP-адреса и неверно указан проверочный код);
2. `'resultHTML'` — база DEF-кодов в формате HTML-таблицы.

После получения успешного ответа от сервера, программа сохраняет локальную копию в файл `base.html` и начинает анализ таблицы. Для данной операции используется модуль `BeautifulSoup`, которая позволяет работать с сущностями таблицы как с объектами. Проанализированные данные программа сохраняет в JSON-файл `base.json`, с которым в последствии работает основное приложение.

Листинг 2.8— Python-скрипт для загрузки базы DEF-кодов

```

1 import sys
2 from datetime import datetime
3 import requests
4 import json
5 from bs4 import BeautifulSoup
6
7 # if script started with date-argument then download base
8 for that date
9 if len(sys.argv) > 1:
10     dwnldate = sys.argv[1]
11 else:
12     dwnldate = datetime.strftime(datetime.now(), "%d.%m.%Y")
13
14 # check os and set right path to base
15 # NB! basePath is path without extension
16 if sys.platform == 'win32':
17     basePath = '../base/base'
18 elif sys.platform == 'darwin':
19     basePath = '../.../.../.../base/base'
20 else:
21     print('You are running on unsupported OS.')
22     print('Or cygwin under the Windows (run in cmd)')
23     raise SystemExit(1) # exit with code 1 (error)
24
25 # collecting data from mtt.ru
26
27 # form request
28 data = {
29     'g': 'mtt',
30     'm': 'def_codes',
31     'a': 'processRequest',
32     'def': "*",
33     'def_number': "",
34     'area': "*",
35     'operator': "*",
36     'standard': "*",
37     'date': dwnldate
38 }
39
40 print('Starting downloading')
41
42 # send request
43 r = requests.post("http://www.mtt.ru/?callback=?",
44 params=data)
45
46 # parse from json
47 r = r.json()
48
49 if r.get('status') == 'success':
50     print('Download was successful')
51 elif r.get('status') == 'captcha':
52     print('Too many requests from your IP. Try again later.')
53     raise SystemExit(1) # exit with code 1 (error)
54 else:
55     print('Something went wrong. Download was unsuccessful')
56     raise SystemExit(1) # exit with code 1 (error)
57
58 # take html-table with data
59 r = r.get('resultHTML')
60 # r = "<table>\r\n" + r + "\r\n</table>"
61
62 # write html-table into file
63 with open(basePath+".html", "w", encoding='utf-8') as f:
64     f.write(r)
65 print('Starting parsing')
66 base = dict()
67

```

```

68 # creating object for html-parser
69 html = BeautifulSoup(r, 'html.parser')
70
71 # for each row of table
72 # get each cell of row
73 # if location or provider of def-code doesnt exist, create
74 dict-entity in base-dictionary
75 for row in html.findAll('tr'):
76     if row.findAll('th'): # save header of table
77         headers = row.findAll('th')
78     else:
79         cell = row.findAll('td')
80
81         intvl_head = headers[1].text
82
83         code = cell[0].text
84         intvl = cell[1].text
85         loc = cell[2].text
86         prov = cell[3].strong.text
87
88         if loc not in base:
89             base[loc] = dict()
90         if prov not in base[loc]:
91             base[loc][prov] = dict()
92         if code not in base[loc][prov]:
93             base[loc][prov][code] = dict()
94
95         base[loc][prov][code].update({intvl: {}})
96
97 print('Parsing was successful')
98 print('Exporting to JSON')
99
100 # export to JSON-format
101 with open(basePath+'.json', 'w', encoding='utf-8') as f:
102     outfile:
103         json.dump(base, outfile, sort_keys=True, indent=4,
104                 ensure_ascii=False)
105
106 print('Exporting was successful')
107 print('Exiting...')
108
109 # exit with code 0 (with no errors)
110 raise SystemExit(0)

```

Готовый скрипт упакован с помощью утилит "py2app" для Mac OS X и "py2exe" для Windows. Пример сценария для сборки Mac OS X версии представлен ниже (Листинг 2.9)

Листинг 2.9— Python-скрипт для сборки Mac OS X версии

```

1 from setuptools import setup
2
3 setup(
4     name = "dwnldData",
5     app=["dwnldData.py"],
6     version = "0.4.4",
7     setup_requires=["py2app"],
8     options={'py2app': {}},
9 )

```

2.3.2 Реализация кросс-платформенности

Весь код приложения является кросс-платформенным, имея различия лишь в дистрибутиве контейнера node-webkit.

Некоторые особенности операционной системы учтены с помощью условных конструкций проверки свойства `os.platform()` (`sys.platform` в Python): для Mac OS X функция возвращает строку "darwin", для Windows — "win32". Например, в листинге 2.10 представлена реализация в js-программе — функция обновления базы. Для выполнения данной функции в различных средах необходимо запускать разные команды, `open` для Mac OS X и `start` для Windows.

Листинг 2.10— `updateBase().js`

```

1 function updateBase() {
2     <...>
3     if (os.platform() === 'darwin') {
4         // mac
5         exec('open _py/dwnldData.app', function() {
6             <...>
7         });
8     } else {
9         // win
10        exec('start _py/dwnldData.exe', function() {
11            <...>
12        });
13    }
14 }

```

В листинге 2.11 представлена реализация в python-скрипте. Эта часть кода выбирает путь до базы исходя из операционной системы, в которой запущен скрипт, что связано с особенностями упаковки сценария.

Листинг 2.11— Выбор пути до файла базы в python-скрипте

```

1 # check os and set right path to base
2 # NB! basePath is path without extension
3 if sys.platform == 'win32':
4     basePath = '../base/base'
5 elif sys.platform == 'darwin':
6     basePath = '../../../../base/base'
7 else:
8     print('You are running on unsupported OS.')
9     print('Or cygwin under the Windows (run in cmd)')
10    raise SystemExit(1) # exit with code 1 (error)

```

ЗАКЛЮЧЕНИЕ

В ходе прохождения производственной практики в период с 01.07.2015 по 10.07.2015 в экспертно-криминалистическом отделе Следственного комитета Иркутской области были получены следующие основные результаты:

1. была изучена организационная и функциональная структуры предприятия;
2. улучшены профессиональные навыки;
3. принято участие в автоматизации задачи отдела.

Также были изучены и рассмотрены основные принципы работы выбранного отдела, использовано в практическом применении установленное в отделе оборудование и ПО, увеличен багаж знаний по выбранной специальности, что в дальнейшем может являться информационной базой будущих исследований.

Производственная практика являлась ценным дополнением учебного процесса, а также способствовала лучшему освоению материала.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Регламент «Следственного управления Следственного комитета Российской Федерации по Иркутской области». — И. : Следственный комитет РФ, 2015.
- [2] Положение «Об экспертно-криминалистическом отделе Следственного управления СК России по Иркутской области». — И. : Следственный комитет РФ, 2014.
- [3] Распоряжение №29-246р «О распределении обязанностей между сотрудниками экспертно-криминалистического отдела Следственного управления СК России по Иркутской области». — И. : Следственный комитет РФ, 2015.
- [4] NW.js is an app runtime based on Chromium and node.js. — URL: <http://nwjs.io/> (дата обращения: 21.08.2015).
- [5] Node.js is platform for build applications. — URL: <https://nodejs.org> (дата обращения: 20.08.2015).
- [6] JSON official site. — URL: <http://json.org/> (дата обращения: 08.07.2015).
- [7] CSS-preprocessor. — URL: <http://sass-lang.com/> (дата обращения: 21.08.2015).
- [8] Same-origin policy. — URL: https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin_policy (дата обращения: 25.08.2015).
- [9] Python official site. — URL: <https://www.python.org/> (дата обращения: 10.07.2015).
- [10] practice2015 repository on GitHub. — URL: <https://github.com/kostydenis/practice2015> (дата обращения: 25.08.2015).
- [11] Material Design.Guidelines. — Google. — URL: <https://www.google.com/design/spec/material-design> (дата обращения: 28.05.2015).
- [12] Autocomplete plugin for jQuery. — URL: <https://jqueryui.com/autocomplete/> (дата обращения: 17.07.2015).
- [13] Filler Patrick. jQuery plugin for html-select. — URL: <https://harvesthq.github.io/chosen/> (дата обращения: 20.08.2015).

- [14] Введение в Javascript Source Maps / Habrahabr. — URL: <http://habrahabr.ru/post/148098/> (дата обращения: 23.08.2015).

Приложение А: Примеры базы в форматах html и json

Листинг А.1— base.html

```

1  <tr>
2  <th>Код DEF</th>
3  <th width="20%">Диапазон номеров</th>
4  <th>Область</th>
5  <th>Оператор/Действительно</th>
6  </tr>
7  <tr>
8  <td>900</td>
9  <td>0000000-0099999</td>
10 <td>Краснодар Краснодарский край<\/td>
11 <td><strong>000 &quot;Телеком Евразия&quot;<\/strong><br />
12 <span class="date-from-to">26.10.2012-01.01.2038<\/span>
13 <\/td>
14 <\/tr>
15 <tr>
16 <td>900<\/td>
17 <td>0100000-0199999<\/td>
18 <td>Тверь Тверская область<\/td>
19 <td><strong>ЗАО &quot;Смоленская <\/strong>
20 <strong>Сотовая Связь&quot;<\/strong><br />
21 <span class="date-from-to">16.11.2012-01.01.2038<\/span>
22 <\/td>
23 <\/tr>
24 <tr>
25 <td>900<\/td>
26 <td>0200000-0299999<\/td>
27 <td>Челябинск Челябинская <\/td>
28 <td><strong>000 &quot;Челябинская <\/strong>
29 <strong>Сотовая связь&quot;<\/strong><br />
30 <span class="date-from-to">02.02.2013-01.01.2038<\/span>
31 <\/td>
32 <\/tr>
33 <tr>
34 <td>900<\/td>
35 <td>0300000-0499999<\/td>
36 <td>Екатеринбург Свердловская <\/td>
37 <td><strong>000 &quot;Екатеринбург-2000&quot;<\/strong><br />
38 <span class="date-from-to">21.10.2013-01.01.2038<\/span>
39 <\/td>
40 <\/tr>
41 <tr>
42 <td>900<\/td>
43 <td>0500000-0599999<\/td>
44 <td>Кемерово Кемеровская область<\/td>
45 <td><strong>ЗАО &quot;Кемеровская <\/strong>
46 <strong>Мобильная Связь&quot;<\/strong><br />
47 <span class="date-from-to">27.02.2015-01.01.2038<\/span>
48 <\/td>
49 <\/tr>
50 <tr>
51 <td>900<\/td>
52 <td>0600000-0699999<\/td>
53 <td>Челябинск Челябинская <\/td>
54 <td><strong>000 &quot;Челябинская <\/strong>
55 <strong>Сотовая связь&quot;<\/strong><br />
56 <span class="date-from-to">03.04.2015-01.01.2038<\/span>
57 <\/td>
58 <\/tr>

```

```

65 </tr>
66 <...>

```

Листинг А.2— base.json

```

1  {
2    <...>
3    "Иркутск Иркутская область": {
4      "ЗАО \"Байкалвестком\"": {
5        "901": {
6          "6300000-6344999": { }},
7          "6400000-6419999": { }},
8          "6500000-6549999": { }},
9          "6550000-6749999": { }}
10     },
11     "902": {
12       "1700000-1799999": { }},
13       "5100000-5169999": { }},
14       "5190000-5199999": { }},
15       "5400000-5499999": { }},
16       "5600000-5619999": { }},
17       "5660000-5699999": { }},
18       "5760000-5799999": { }},
19       "7600000-7699999": { }}
20     },
21     "904": {
22       "1100000-1599999": { }}
23     },
24     "908": {
25       "6400000-6699999": { }},
26       "7700000-7799999": { }}
27     },
28     "950": {
29       "0500000-1499999": { }}
30     },
31     "952": {
32       "6100000-6399999": { }}
33     }
34   },
35   "ОАО \"Вымпел-Коммуникации\"": {
36     "964": {
37       "1000000-1299999": { }},
38       "2100000-2299999": { }},
39       "2300000-2309999": { }},
40       "2600000-2899999": { }},
41       "3500000-3599999": { }},
42       "5400000-5499999": { }},
43       "6500000-6599999": { }},
44       "7300000-7599999": { }},
45       "8000000-8249999": { }}
46     }
47   },
48   "ОАО \"Мегафон\" 2
49   3G_Дальневосточный филиал": {
50     "934": {
51       "4310000-4319999": { }}
52     }
53   },
54   "ОАО \"Мегафон\" 2

```

```

55     Дальневосточный филиал": {
56         "924": {
57             "2900000-2949999": { } ,
58             "5300000-5399999": { } ,
59             "5400000-5499999": { } ,
60             "6000000-6399999": { } ,
61             "7000000-7199999": { } ,
62             "8200000-8299999": { } ,
63             "8300000-8399999": { }
64         } ,
65         "929": {
66             "4300000-4399999": { }
67         } ,
68         "999": {
69             "2303000-2312999": { } ,
70             "2510000-2549999": { }
71         }
72     } ,
73     "ОАО \"Мобильные
74     Телесистемы\"": {
75         "914": {
76             "0000000-0149999": { } ,
77             "8700000-9599999": { }
78         } ,
79         "983": {
80             "2400000-2499999": { } ,
81             "4000000-4199999": { } ,
82             "4400000-4499999": { } ,
83             "4600000-4699999": { } ,
84             "6900000-6999999": { }
85         } ,
86         "984": {
87             "2700000-2799999": { }
88         }
89     }
90 } ,
91 <...>
92 }

```

Приложение Б: Листинг main.js

Листинг Б.1— main.js

```

1
2 var os = require('os');
3 var gui = require('nw.gui');
4 var fs = require('fs');
5 var exec = require("child_process").exec;
6 var spawn = require("child_process").spawn;
7 var excelbuilder = require('msexcel-builder');
8
9 var logging = false;
10
11 var lang;
12 var chosenLang = 'ru';
13 var base;
14 var locations = [];
15 var providers = [];
16 var defs = [];
17
18 var chosenLocations = [];
19 var chosenProviders = [];
20 var chosenDefs = [];
21
22 var saveTo;
23 if (os.platform() === 'darwin') {
24     saveTo = window.location.pathname.slice(0, -10);
25 } else {
26     saveTo = window.location.pathname.slice(1, -10);
27 }
28
29 var logFileDate = new Date();
30 logFileDate = logFileDate.toLocaleString().replace(/[:]|,|[/g, '-');
31
32 var logFilePath = 'logs/' + logFileDate + '.txt';
33
34 function log(msg) {
35     if (logging) {
36         var date = new Date();
37         date = date.toLocaleTimeString().replace(/[:]|,|[/g, '-');
38         msg = date + ': ' + msg + '\r\n';
39         fs.appendFileSync(logFilePath, msg);
40     }
41 }
42
43 log('Application is launched');
44 log('OS Platform: ' + os.platform());
45 log('OS Type: ' + os.type());
46 log('OS Release: ' + os.release());
47 log('OS Arch: ' + os.arch());
48 log('OS Hostname: ' + os.hostname());
49
50
51
52 function init() {
53     log('Begin initialization')
54     var langfile;
55
56
57     if (chosenLang == 'ru') {
58         langfile = 'lang/ru.json'
59     }
60
61     if (chosenLang == 'en') {
62         langfile = 'lang/en.json'
63     }
64     log('Chosen language file is ' + chosenLang);
65     $.getJSON(langfile, function (response) {
66         log('Begin parsing lang-JSON');
67         jQuery(document).ready(function () {
68             log('Basic template is initialized');

```

```

69     lang = response;
70     log('Starting initialization of basic template');
71     initBaseTemplate();
72     log('starting initialization of extended 2
73     template');
74     $('header h1').empty();
75     $('header h1').append(lang.title);
76
77     $('#note_mnp').empty();
78     $('#note_mnp').append(lang.note_mnp + ' ' + '<a 2
79     id="learnmore_mnp">' + lang.learnmore + '</a>');
80
81     $('#learnmore_mnp').click(function () {
82         gui.Shell.openExternal('http://pravo.gov.2
83         ru/proxy/ips/?docbody=&link_id=2&nd=1021622542
84         ');
85     });
86
87
88     $('.block_number_check').removeClass('preinit');
89     $('#input_number_check').focus();
90     $('#input_number_check').attr('placeholder', 2
91     lang.placeholder_input_number_check);
92
93     $('#button_number_check').click(function () {
94         log('#button_number_check is pressed');
95         var provider = findProvider(strToCode($('#2
96         #input_number_check').val()).def, strToCode(2
97         $('#input_number_check').val()).number);
98         if (provider.error != undefined) {
99             openModal(provider.error);
100         } else {
101             openModal(provider.loc + '<br />' + 2
102             provider.prov);
103         }
104     });
105     $('#input_number_check').keypress(function (e) {
106         if (e.keyCode == 13) {
107             log('#input_number_check is enter-2
108             pressed');
109             if ($('#modal').attr('class').indexOf(2
110             'preinit') != -1) {
111                 var provider = findProvider(2
112                 strToCode($('#input_number_check').2
113                 val()).def, strToCode($('#2
114                 #input_number_check').val()).number)2
115                 ;
116                 if (provider.error != undefined) {
117                     openModal(provider.error);
118                 } else {
119                     openModal(provider.loc + '<br />2
120                     ' + provider.prov);
121                 }
122             } else {
123                 closeModal();
124             }
125         }
126     });
127
128     $('#input').keypress(function (e) {
129         if ((e.ctrlKey == true) && (e.keyCode == 65) 2
130         ||
131             (e.metaKey == true) && (e.keyCode == 2
132             65)) {
133             this.select();
134             log('Whole text in input is selected');
135         }
136     };
137
138     });

```

```

139 document.onkeydown = function (e) {
140     if ((e.ctrlKey == true) && (e.keyCode == 82) )
141     ||
142         (e.metaKey == true) && (e.keyCode == 2
143             82)) {
144         log('Application is reset');
145         location.reload();
146     }
147     if ((e.ctrlKey == true) && (e.keyCode == 81) )
148     ||
149         (e.metaKey == true) && (e.keyCode == 2
150             81)) {
151         log('Closing application');
152         gui.Window.get().close();
153     }
154 }
155
156 $('#btn_about').empty();
157 $('#btn_about').append(lang.btn_about);
158
159 $('#btn_about').click(openAboutModal);
160
161 $('#btn_author').empty();
162 $('#btn_author').append(gui.App.manifest.author);
163
164 $('#btn_author').click(function () {
165     gui.Shell.openExternal('http://kostydenis.2
166     me')
167 });
168
169
170 $('.form_request').removeClass('preinit');
171
172 $('#header_form_request').empty();
173 $('#header_form_request').append(lang.2
174     header_form_request);
175
176 $('#input_locations').attr('data-placeholder', 2
177     lang.placeholder_input_locations);
178 $('#input_providers').attr('data-placeholder', 2
179     lang.placeholder_input_providers);
180 $('#input_defs').attr('data-placeholder', lang.2
181     placeholder_input_defs);
182
183
184 $('#btn_form_request').empty();
185 $('#btn_form_request').append(lang.2
186     btn_form_request);
187
188 $('#btn_form_request').click(function () {
189     openExportModal()
190 });
191
192 $('#btn_clear_form_request').empty();
193 $('#btn_clear_form_request').append('<span 2
194     class="glyphicon glyphicon-trash" aria-2
195     hidden="true">&nbsp;</span>' + lang.2
196     btn_clear_form_request);
197
198 $('#btn_clear_form_request').click(function () {
199     log('Clear button is pressed');
200     location.reload();
201 });
202
203 $('#btn_update').empty();
204 $('#btn_update').append(lang.btn_update);
205 $('#btn_update').click(function () {
206     if (navigator.onLine == false) {
207         openModal(lang.youreoffline);
208     } else {
209         updateBase();
210     }
211 });

```

```

212
213     $('#input_number_check').mask('+7 (000) 000-2
214     0000');
215
216     $('#btn_ru').click(function () {
217         log('Russian language is chosen');
218         $('#btn_ru').removeClass('not_pressed');
219         $('#btn_en').addClass('not_pressed');
220         chosenLang = 'ru';
221         init();
222     });
223     $('#btn_en').click(function () {
224         log('English language is chosen');
225         $('#btn_en').removeClass('not_pressed');
226         $('#btn_ru').addClass('not_pressed');
227         chosenLang = 'en';
228         init();
229     });
230
231     $('#export_to_txt').empty();
232     $('#export_to_txt').append(lang.export_to_txt);
233     $('#export_to_txt').click(function () {
234         log('Exporting to txt');
235         exportToTxt();
236     });
237
238     $('#export_to_excel').empty();
239     $('#export_to_excel').append(lang.
240     export_to_excel);
241     $('#export_to_excel').click(function () {
242         log('Exporting to excel');
243         exportToExcel();
244     });
245     });
246 })
247 log('Extended template is initialized');
248 base_init();
249
250 };
251
252 var base_init = function () {
253     log('Staring initialization of base');
254     $.getJSON('base/base.json', function (response) {
255
256         $('#.last_update').empty();
257         var baseModifyDate = fs.statSync('base/base.json').
258         mtime;
259         $('#.last_update').append(lang.base_last_update+
260         baseModifyDate.toLocaleDateString().replace(/[:]|,|[/g, '-')));
261         if (dateDiffInDays(baseModifyDate, new Date()) > 30) {
262             {
263                 setError();
264             }
265         }
266         base = response;
267         log('base is opened');
268         locations = getLocations();
269         log('locations is formed');
270         initAutocomplete();
271
272         $('#input_locations').chosen().change(function(event,
273         params) {
274             $('#input_providers').empty();
275             $('#input_defs').empty();
276
277             locations = [];
278             providers = [];
279
280             defs = [];
281             if ($('#input_locations').val() != null) {
282                 //locations is not empty

```

```

285         locations = $('#input_locations').val();
286
287         var idProv = 1;
288         var idDef = 1;
289
290         for (loc in locations) {
291             $('#input_providers').append('<optgroup >
292             label="'+locations[loc]+'"' >
293             id="provider_optgroup'+ idProv +' ">');
294             var tmpProviders = getProviders(>
295             locations[loc]);
296
297             for (prov in tmpProviders){
298                 providers.push(tmpProviders[prov]);
299                 $('#input_providers >
300                 #provider_optgroup'+idProv).append('<>
301                 option>'+tmpProviders[prov]+'<>
302                 /option>');
303
304                 var tmpDefs = getDefs(locations[loc],>
305                 tmpProviders[prov]);
306                 $('#input_defs').append('<optgroup >
307                 label="'+tmpProviders[prov].replace(>
308                 /"([^\"]+)/g, '«$1»')+ "' >
309                 id="def_optgroup'+ idDef +' ">');
310                 for (def in tmpDefs) {
311                     defs.push(tmpDefs[def]);
312                     $('#input_defs #def_optgroup'+>
313                     idDef).append('<option>'+tmpDefs[>
314                     def+'</option>');
315                 }
316                 idDef++;
317             }
318             idProv++;
319         }
320     } else { //locations is empty
321         initAutocomplete();
322     }
323
324     $('#input_providers').trigger('chosen:updated');
325     $('#input_defs').trigger('chosen:updated');
326 });
327
328 $('#input_providers').chosen().change(function(event,>
329 params) {
330
331     $('#input_defs').empty();
332     providers = [];
333     defs = [];
334
335     if ($('#input_providers').val() != null) { >
336         //providers is not empty
337         if ($('#input_locations').val() == null) { >
338             //providers is not empty, locations is >
339             empty
340             $('#input_locations').empty();
341             $('#input_locations').attr('disabled', >
342             true);
343             locations = [];
344
345             var idDef = 1;
346             providers = $('#input_providers').val();
347             var tmpLocations = getLocations();
348             for (loc in tmpLocations) {
349                 for (prov in providers){
350                     if ($.inArray(providers[prov], >
351                     Object.keys(base[tmpLocations[>
352                     loc]])) != -1) {
353                         var tmpDefs = getDefs(>
354                         tmpLocations[loc], providers[>
355

```



```

357         prov]);
358         $('#input_defs').append('<
359         optgroup label="'+
360         tmpLocations[loc]+' "
361         id="def_optgroup'+ idDef +' ">
362         ');
363         for (def in tmpDefs) {
364             defs.push(tmpDefs[def]);
365             $('#input_defs
366             #def_optgroup'+idDef).
367             append('<option>'+
368             tmpDefs[def]+'</option>')
369             ;
370         }
371         idDef++;
372     }
373 }
374 }
375 } else { //providers is not empty, locations
376 is not empty
377     var idDef = 1;
378     providers = $('#input_providers').val();
379     for (loc in locations) {
380         for (prov in providers) {
381             if ($.inArray(providers[prov],
382             Object.keys(base[locations[loc]]))
383             ) != -1) {
384
385                 var tmpDefs = getDefs(
386                 locations[loc], providers[
387                 prov]);
388                 $('#input_defs').append('<
389                 optgroup label="'+providers[
390                 prov].replace(/"([\^"]+)/g,
391                 '<`$1>'+' "
392                 id="def_optgroup'+ idDef +' ">
393                 ');
394                 for (def in tmpDefs) {
395                     defs.push(tmpDefs[def]);
396                     $('#input_defs
397                     #def_optgroup'+idDef).
398                     append('<option>'+
399                     tmpDefs[def]+'</option>')
400                     ;
401                 }
402                 idDef++;
403             }
404         }
405     }
406 }
407 } else { //providers is empty
408     $('#input_locations').attr('disabled', false);
409     ;
410     if ( $('#input_locations').val() == null ) {
411         //providers is empty, locations is empty
412         initAutocomplete();
413     } else { //providers is empty, locations is
414     not empty
415         locations = getLocations();
416
417         var idDef = 1;
418         for (loc in locations) {
419             var tmpProviders = getProviders(
420             locations[loc]);
421             for (prov in tmpProviders) {
422                 providers.push(tmpProviders[prov]);
423             };
424
425             var tmpDefs = getDefs(locations[
426             loc], tmpProviders[prov]);

```

```

427         $('#input_defs').append('<
428         optgroup label="'+tmpProviders[2
429         prov].replace(/"([^\"]+)/g, <
430         '«$1»')+'" id="def_optgroup'+ 2
431         idDef +' ">');
432         for (def in tmpDefs) {
433             defs.push(tmpDefs[def]);
434             $('#input_defs 2
435             #def_optgroup'+idDef).append(2
436             '<option>'+tmpDefs[def]+'<
437             /option>');
438         }
439         idDef++;
440     }
441 }
442 }
443 }
444
445 var startTime = Date.now();
446 $('#input_locations').trigger('chosen:updated');
447 $('#input_defs').trigger('chosen:updated');
448 var endTime = Date.now();
449 console.log('trigger: '+(endTime - startTime));
450 });
451 $('#input_defs').chosen().change(function(event, 2
452 params) {
453     if ($('#input_defs').val() != null) {
454         if ( ($('#input_locations').val() == null) 2
455         && ($('#input_providers').val() == null) ) {
456             $('#input_locations').attr('disabled', 2
457             true);
458             $('#input_providers').attr('disabled', 2
459             true);
460         }
461     } else {
462         if ( ($('#input_locations').val() == null) 2
463         && ($('#input_providers').val() == null) ) {
464             $('#input_locations').attr('disabled', 2
465             false);
466             $('#input_providers').attr('disabled', 2
467             false);
468             initAutocomplete();
469         }
470     }
471     $('#input_locations').trigger('chosen:updated');
472     $('#input_providers').trigger('chosen:updated');
473 });
474
475
476
477 $('#input_locations').chosen();
478 $('#input_providers').chosen();
479 $('#input_defs').chosen();
480
481 });
482 }
483
484 function getLocations() {
485     var locations = [];
486     for (prop in base) {
487         locations.push(prop);
488     }
489     return locations;
490 };
491 function getProviders(loc) {
492     var providers = [];
493     for (prop in base[loc]) {
494         providers.push(prop);
495     }

```

```

496     return providers;
497 };
498 function getDefs(loc, prov) {
499     var defs = [];
500     for (prop in base[loc][prov]) {
501         defs.push(prop);
502     }
503     return defs;
504 };
505
506
507 function dateDiffInDays(a, b) {
508     var _MS_PER_DAY = 1000 * 60 * 60 * 24;
509     // Discard the time and time-zone information.
510     var utc1 = Date.UTC(a.getFullYear(), a.getMonth(), a.getDate());
511     var utc2 = Date.UTC(b.getFullYear(), b.getMonth(), b.getDate());
512     return Math.floor((utc2 - utc1) / _MS_PER_DAY);
513 }
514
515 function updateBase() {
516     showLoading();
517     if (os.platform() === 'darwin') {
518         // mac
519         exec('open _py/dwnldData.app', function() {
520             $(''.last_update').empty();
521             var baseModifyDate = fs.statSync('base/base.json').mtime;
522             $(''.last_update').append(lang.base_last_update+baseModifyDate.toLocaleDateString().replace(/[:]/g, '|') + '/' + baseModifyDate.toLocaleDateString().replace(/[:]/g, '|') + '-' + baseModifyDate.toLocaleDateString().replace(/[:]/g, '|'));
523             hideLoading();
524             openModal(lang.successful_update);
525         });
526     } else {
527         // win
528         exec('start _py/dwnldData.exe', function() {
529             $(''.last_update').empty();
530             var baseModifyDate = fs.statSync('base/base.json').mtime;
531             $(''.last_update').append(lang.base_last_update+baseModifyDate.toLocaleDateString().replace(/[:]/g, '|') + '/' + baseModifyDate.toLocaleDateString().replace(/[:]/g, '|') + '-' + baseModifyDate.toLocaleDateString().replace(/[:]/g, '|'));
532             hideLoading();
533             openModal(lang.successful_update);
534         });
535     }
536 }
537
538 function initAutocomplete() {
539     var startTime = Date.now();
540     locations = getLocations();
541     providers = [];
542     defs = [];
543     $('#input_locations').empty();
544     $('#input_providers').empty();
545     $('#input_defs').empty();
546
547     var idProv = 1;
548     var idDef = 1;
549
550     for (loc in locations) {
551         $('#input_locations').append('<option>' + locations[loc] + '</option>');
552
553         $('#input_providers').append('<optgroup label="' + locations[loc] + '" id="provider_optgroup' + idProv + '">');
554     }
555 }

```

```

568         ');
569
570     var tmpProviders = getProviders(locations[loc]);
571     for (prov in tmpProviders) {
572         providers.push(tmpProviders[prov]);
573
574         $('#input_providers #provider_optgroup'+idProv).append('<option>'+tmpProviders[prov]+'</option>');
575     }
576
577     var tmpDefs = getDefs(locations[loc], tmpProviders[prov]);
578     $('#input_defs').append('<optgroup label="'+tmpProviders[prov].replace(/"([\^"]+)/g, '&quot;$1&quot;')+ " id="def_optgroup'+ idDef +' ">');
579     for (def in tmpDefs) {
580         defs.push(tmpDefs[def]);
581         $('#input_defs #def_optgroup'+idDef).append('<option>'+tmpDefs[def]+'</option>');
582     }
583     idDef++;
584 }
585 idProv++;
586 }
587
588 var endTime = Date.now();
589 console.log('init: '+(endTime - startTime));
590
591 $('#input_locations').trigger('chosen:updated');
592 $('#input_providers').trigger('chosen:updated');
593 $('#input_defs').trigger('chosen:updated');
594 }
595
596 //building chosen...
597 function buildChoose() {
598     var startTime = Date.now();
599     chosenLocations = [];
600     chosenProviders = [];
601     chosenDefs = [];
602
603     if ( $('#input_locations').val() != null ) {
604         chosenLocations = $('#input_locations').val();
605     } else {
606         chosenLocations = getLocations();
607     }
608
609     if ( $('#input_providers').val() != null ) {
610         chosenProviders = $('#input_providers').val();
611     } else {
612         for (loc in chosenLocations) {
613             var tmpProviders = getProviders(chosenLocations[loc]);
614             for (prov in tmpProviders) {
615                 chosenProviders.push(tmpProviders[prov]);
616             }
617         }
618     }
619
620     if ( $('#input_defs').val() != null ) {
621         chosenDefs = $('#input_defs').val();
622     } else {
623         if ( $('#input_providers').val() == null ) {
624             for (loc in chosenLocations) {
625                 var tmpProviders = getProviders(chosenLocations[loc]);
626                 for (prov in tmpProviders) {
627                     var tmpDefs = getDefs(chosenLocations[loc], tmpProviders[prov]);

```

```

640         for (def in tmpDefs) {
641             chosenDefs.push(tmpDefs[def]);
642         }
643     }
644 }
645 } else {
646     for (loc in chosenLocations) {
647         for (prov in chosenProviders) {
648             if ( $.inArray(chosenProviders[prov], 2
649                 getProviders(chosenLocations[loc])) != -2
650                 1 ) {
651                 var tmpDefs = getDefs(2
652                     chosenLocations[loc], 2
653                     chosenProviders[prov]);
654                 for (def in tmpDefs) {
655                     chosenDefs.push(tmpDefs[def]);
656                 }
657             }
658         }
659     }
660 }
661 }
662 }
663
664 var endTime = Date.now();
665 console.log('buildChoose() push: '+(endTime - startTime));
666 ;
667
668 }
669
670
671 function getInterval(loc, prov, def) {
672     var interval = [];
673     for (prop in base[loc][prov][def]) {
674         interval.push(prop);
675     }
676     return interval;
677 }
678 ;
679 function isInInterval(from, nbr, to) {
680     if ((nbr >= from) && (nbr <= to)) {
681         return true;
682     } else {
683         return false;
684     }
685 }
686 function strToCode(str) {
687     var output = {};
688     output.def = str.substring(4, 7);
689     output.number = str.substring(9).replace('-', '');
690     return output;
691 }
692 function findEqualDigits(first, second) {
693     var count = 0;
694     for (var i = 0; i < first.length; i++) {
695         if (first[i] === second[i]) {
696             count++;
697         } else {
698             return count;
699         }
700     }
701     return count;
702 }
703 function findEqualDigitsInArray(array) {
704     var count = 0;
705     for (var i = 0; i < array[0].length; i++) {
706         if (array[0][i] === array[1][i]) {
707             count++;

```

```

708         } else {
709             return count;
710         }
711     }
712     return count;
713 }
714 function findZeroQty(number) {
715     var count = 0;
716     var tmpNumber = number;
717     while (tmpNumber[tmpNumber.length-1] === '0') {
718         tmpNumber = tmpNumber.slice(0,-1);
719         count++;
720     }
721     return count;
722 }
723 function findNineQty(number) {
724     var count = 0;
725     var tmpNumber = number;
726     while (tmpNumber[tmpNumber.length-1] === '9') {
727         tmpNumber = tmpNumber.slice(0,-1);
728         count++;
729     }
730     return count;
731 }
732
733 function intervalToTemplate(partsOfInterval) {
734     var output = [];
735     var tmpQty;
736     var nines = findNineQty(partsOfInterval[1]);
737     var zeros = findZeroQty(partsOfInterval[0]);
738     if (nines < zeros) {
739         tmpQty = nines;
740     } else {
741         tmpQty = zeros;
742     }
743     partsOfInterval[0] = partsOfInterval[0].slice(0, -2
744     1*tmpQty); //cut off that characters from the end
745     partsOfInterval[1] = partsOfInterval[1].slice(0, -2
746     1*tmpQty); //for both numbers
747     tmpQty = findEqualDigitsInArray(partsOfInterval); //find 2
748     quantity of the same starting characters
749     var equalDigits = partsOfInterval[0].slice(0,tmpQty); 2
750     //remember it, need to put before mutable part
751     partsOfInterval[0] = partsOfInterval[0].slice(tmpQty); 2
752     //and cut it off
753     partsOfInterval[1] = partsOfInterval[1].slice(tmpQty); 2
754     //for both numbers
755     var diff = partsOfInterval[1] - partsOfInterval[0]; 2
756     //calculating difference between interval
757     if (diff == 0) {
758         output.push(equalDigits+'%');
759     } else {
760         for (var i = parseInt(partsOfInterval[0]); i<=
761         =parseInt(partsOfInterval[0])+diff; i++) { //form 2
762             template increasing first number of interval for 2
763             ^diff times
764             output.push(equalDigits + i + '%');
765         }
766     }
767     return output;
768 }
769
770 function exportToTXT() {
771     log('Starting export to txt');
772     var date = new Date();
773     date = date.toLocaleString().replace(/[:]/g, '|./g, '-');
774     fs.mkdirSync(saveTo+date);
775     log('Folder '+date+' is created');
776
777     if ($('#input_providers').val() == null) {

```

```

778     if ($('#input_defs').val() == null) {
779         for (loc in chosenLocations) {
780             fs.mkdirSync(saveTo+date+'/' + chosenLocations[loc]);
781         };
782         var tmpProviders = getProviders(chosenLocations[loc]);
783         for (prov in tmpProviders) {
784             var file = fs.openSync(saveTo+date + '/' + chosenLocations[loc] + '/' + tmpProviders[prov].replace(/"([\^"]+)/g, '«$1»') + '.txt', 'w');
785             var tmpDefs = getDefs(chosenLocations[loc], tmpProviders[prov]);
786             for (def in tmpDefs) {
787                 var tmpInterval = getInterval(chosenLocations[loc], tmpProviders[prov], tmpDefs[def]);
788                 for (intvl in tmpInterval) {
789                     var tmpParts = intervalToTemplate(tmpInterval[intvl].split('-'));
790                     for (tmpl in tmpParts) {
791                         fs.writeSync(file, tmpDefs[def] + tmpParts[tmpl] + '\r\n');
792                     }
793                 }
794             }
795         }
796         fs.close(file);
797     }
798 } else {
799     for (loc in chosenLocations) {
800         fs.mkdirSync(saveTo+date+'/' + chosenLocations[loc]);
801         log('File ' + chosenLocations[loc] + ' is created');
802         var tmpProviders = getProviders(chosenLocations[loc]);
803         for (prov in tmpProviders) {
804             var file = fs.openSync(saveTo+date + '/' + chosenLocations[loc] + '/' + tmpProviders[prov].replace(/"([\^"]+)/g, '«$1»') + '.txt', 'w');
805             for (def in chosenDefs) {
806                 if ( $.isArray(chosenDefs[def], getDefs(chosenLocations[loc], tmpProviders[prov])) != -1 ) {
807                     var tmpInterval = getInterval(chosenLocations[loc], tmpProviders[prov], chosenDefs[def]);
808                     for (intvl in tmpInterval) {
809                         var tmpParts = intervalToTemplate(tmpInterval[intvl].split('-'));
810                         for (tmpl in tmpParts) {
811                             fs.writeSync(file, chosenDefs[def] + tmpParts[tmpl] + '\r\n');
812                         }
813                     }
814                 }
815             }
816         }
817         fs.close(file);
818     }
819 }

```

```

847     }
848   } else {
849     if ($('#input_defs').val() == null) {
850       for (loc in chosenLocations) {
851         fs.mkdirSync(saveTo+date+'/'+chosenLocations[loc]);
852         log('File '+chosenLocations[loc]+' is
853         created');
854         for (prov in chosenProviders) {
855           var file = fs.openSync(saveTo+date + '/' +
856           + chosenLocations[loc]+'/' +
857           chosenProviders[prov].replace(/"([^\"]+)"
858           "/g, '«$1»') + '.txt', 'w');
859           if ( $.inArray(chosenProviders[prov],
860           getProviders(chosenLocations[loc])) != -
861           1 ) {
862             var tmpDefs = getDefs(
863             chosenLocations[loc],
864             chosenProviders[prov]);
865             for (def in tmpDefs) {
866               var tmpInterval = getInterval(
867               chosenLocations[loc],
868               chosenProviders[prov], tmpDefs[
869               def]);
870               for (intvl in tmpInterval) {
871                 var tmpParts =
872                 intervalToTemplate(
873                 tmpInterval[intvl].split('-')
874                 );
875                 for (tmpl in tmpParts) {
876                   fs.writeFileSync(file,
877                   tmpDefs[def] + tmpParts[
878                   tmpl] + '\r\n');
879                 }
880               }
881             }
882           }
883         }
884         fs.close(file);
885       }
886     }
887   } else {
888     for (loc in chosenLocations) {
889       fs.mkdirSync(saveTo+date+'/'+chosenLocations[loc]);
890       log('File '+chosenLocations[loc]+' is
891       created');
892       for (prov in chosenProviders) {
893         var file = fs.openSync(saveTo+date + '/' +
894         + chosenLocations[loc] + '/' +
895         chosenProviders[prov].replace(/"([^\"]+)"
896         "/g, '«$1»')+'.txt', 'w');
897         if ( $.inArray(chosenProviders[prov],
898         getProviders(chosenLocations[loc])) != -
899         1 ) {
900           for (def in chosenDefs) {
901             if ( $.inArray(chosenDefs[def],
902             getDefs(chosenLocations[loc],
903             chosenProviders[prov])) != -1 ) {
904               var tmpInterval =
905               getInterval(chosenLocations[loc],
906               chosenProviders[prov],
907               chosenDefs[def]);
908               for (intvl in tmpInterval) {
909                 var tmpParts =
910                 intervalToTemplate(
911                 tmpInterval[intvl].split(
912                 '-' ));
913                 for (tmpl in tmpParts) {
914                   fs.writeFileSync(file,
915

```



```

916             chosenDefs[def] + 2
917             tmpParts[tmp1] + 2
918             '\r\n');
919         }
920     }
921 }
922 }
923 }
924 fs.close(file);
925 }
926 }
927 }
928 }
929 }
930 }
931 function exportToExcel() {
932     showLoading();
933     log('Starting export to excel');
934     var date = new Date();
935     date = date.toLocaleString().replace(/[:|,| |.]/g, '-');
936     fs.mkdirSync(saveTo+date);
937     var wb = excelbuilder.createWorkbook(saveTo+date+'/', 2
938     'defs.xlsx');
939     log('Excel file is created');
940
941     if ($('#input_providers').val() == null) {
942         if ($('#input_defs').val() == null) {
943             for (loc in chosenLocations) {
944                 var sheet = wb.createSheet(chosenLocations[2
945                 loc], 1, 3000);
946                 log('Sheet '+chosenLocations[loc]+' is 2
947                 created');
948                 var line = 1;
949
950                 var tmpProviders = getProviders(2
951                 chosenLocations[loc]);
952                 for (prov in tmpProviders) {
953                     var tmpDefs = getDefs(chosenLocations[2
954                     loc], tmpProviders[prov]);
955                     for (def in tmpDefs) {
956                         var tmpInterval = getInterval(2
957                         chosenLocations[loc], tmpProviders[2
958                         prov], tmpDefs[def]);
959                         for (intvl in tmpInterval) {
960                             var tmpParts = 2
961                             intervalToTemplate(tmpInterval[2
962                             intvl].split('-'));
963                             for (tmp1 in tmpParts) {
964                                 sheet.set(1, line++, tmpDefs[2
965                                 def] + tmpParts[tmp1]);
966                             }
967                         }
968                     }
969                 }
970             }
971         }
972     }
973     } else {
974         for (loc in chosenLocations) {
975             var sheet = wb.createSheet(chosenLocations[2
976             loc], 1, 3000);
977             log('Sheet '+chosenLocations[loc]+' is 2
978             created');
979             var line = 1;
980
981             var tmpProviders = getProviders(2
982             chosenLocations[loc]);
983             for (prov in tmpProviders) {
984                 for (def in chosenDefs) {

```

```

986         if ( $.inArray(chosenDefs[def], 2
987         getDefs(chosenLocations[loc], 2
988         tmpProviders[prov])) != -1 ) {
989             var tmpInterval = getInterval(2
990             chosenLocations[loc], 2
991             tmpProviders[prov], chosenDefs[2
992             def]);
993             for (intvl in tmpInterval) {
994                 var tmpParts = 2
995                 intervalToTemplate(2
996                 tmpInterval[intvl].split('-')2
997                 );
998                 for (tmp1 in tmpParts) {
999                     sheet.set(1, line++, 2
1000                     chosenDefs[def] + 2
1001                     tmpParts[tmp1]);
1002                 }
1003             }
1004         }
1005     }
1006 }
1007 }
1008 }
1009 } else {
1010     if ($('#input_defs').val() == null) {
1011         for (loc in chosenLocations) {
1012             var sheet = wb.createSheet(chosenLocations[2
1013             loc], 1, 3000);
1014             log('Sheet ' + chosenLocations[loc] + ' is 2
1015             created');
1016             var line = 1;
1017
1018             for (prov in chosenProviders) {
1019                 if ( $.inArray(chosenProviders[prov], 2
1020                 getProviders(chosenLocations[loc])) != -2
1021                 1 ) {
1022                     var tmpDefs = getDefs(2
1023                     chosenLocations[loc], 2
1024                     chosenProviders[prov]);
1025                     for (def in tmpDefs) {
1026                         var tmpInterval = getInterval(2
1027                         chosenLocations[loc], 2
1028                         chosenProviders[prov], tmpDefs[2
1029                         def]);
1030                         for (intvl in tmpInterval) {
1031                             var tmpParts = 2
1032                             intervalToTemplate(2
1033                             tmpInterval[intvl].split('-')2
1034                             );
1035                             for (tmp1 in tmpParts) {
1036                                 sheet.set(1, line++, 2
1037                                 tmpDefs[def] + tmpParts[2
1038                                 tmp1]);
1039                             }
1040                         }
1041                     }
1042                 }
1043             }
1044         }
1045     } else {
1046         for (loc in chosenLocations) {
1047             var sheet = wb.createSheet(chosenLocations[2
1048             loc], 1, 3000);
1049             log('Sheet ' + chosenLocations[loc] + ' is 2
1050             created');
1051             var line = 1;
1052
1053             for (prov in chosenProviders) {
1054                 if ( $.inArray(chosenProviders[prov], 2

```

```

1055         getProviders(chosenLocations[loc])) != -2
1056     1 ) {
1057         for (def in chosenDefs) {
1058             if ( $.inArray(chosenDefs[def], 2
1059                 getDefs(chosenLocations[loc], 2
1060                 chosenProviders[prov])) != -1 ) {
1061                 var tmpInterval = 2
1062                 getInterval(chosenLocations[2
1063                     loc], chosenProviders[prov], 2
1064                     chosenDefs[def]);
1065                 for (intvl in tmpInterval) {
1066                     var tmpParts = 2
1067                     intervalToTemplate(2
1068                         tmpInterval[intvl].split(2
1069                             '-' ));
1070                     for (tmpl in tmpParts) {
1071                         sheet.set(1, line++, 2
1072                             chosenDefs[def] + 2
1073                             tmpParts[tmpl]);
1074                     }
1075                 }
1076             }
1077         }
1078     }
1079 }
1080 }
1081 }
1082 }
1083 wb.save(function (ok) {hideLoadingWOWrapper(); alert(2
1084     lang.successful_export)});
1085 }
1086 }
1087 function exportToScreen() {
1088     if ($('#input_providers').val() == null) {
1089         if ($('#input_defs').val() == null) {
1090             for (loc in chosenLocations) {
1091                 $('#modal-content').append('<h3>' + 2
1092                     chosenLocations[loc] + '</h3>');
1093                 var tmpProviders = getProviders(2
1094                     chosenLocations[loc]);
1095                 for (prov in tmpProviders) {
1096                     $('#modal-content').append('<h4>' + 2
1097                         tmpProviders[prov] + '</h4>');
1098                     var tmpDefs = getDefs(chosenLocations[2
1099                         loc], tmpProviders[prov]);
1100                     for (def in tmpDefs) {
1101                         var tmpInterval = getInterval(2
1102                             chosenLocations[loc], tmpProviders[2
1103                                 prov], tmpDefs[def]);
1104                         for (intvl in tmpInterval) {
1105                             var tmpParts = 2
1106                             intervalToTemplate(tmpInterval[2
1107                                 intvl].split('-'));
1108                             for (tmpl in tmpParts) {
1109                                 $('#modal-content').append('<2
1110                                     p>' + tmpDefs[def] + 2
1111                                     tmpParts[tmpl] + '</p>');
1112                             }
1113                         }
1114                     }
1115                 }
1116             }
1117         }
1118     } else {
1119         for (loc in chosenLocations) {
1120             $('#modal-content').append('<h3>' + 2

```

```

1121     chosenLocations[loc] + '</h3>');
1122     var tmpProviders = getProviders(
1123     chosenLocations[loc]);
1124     for (prov in tmpProviders) {
1125         $('#modal-content').append('<h4>' +
1126         tmpProviders[prov] + '</h4>');
1127         for (def in chosenDefs) {
1128             if ( $.inArray(chosenDefs[def],
1129             getDefs(chosenLocations[loc],
1130             tmpProviders[prov])) != -1 ) {
1131                 var tmpInterval = getInterval(
1132                 chosenLocations[loc],
1133                 tmpProviders[prov], chosenDefs[
1134                 def]);
1135                 for (intvl in tmpInterval) {
1136                     var tmpParts =
1137                     intervalToTemplate(
1138                     tmpInterval[intvl].split('-')
1139                     );
1140                     for (tmpl in tmpParts) {
1141                         $('#modal-content').
1142                         append('<p>' + chosenDefs[
1143                         def] + tmpParts[tmpl] + '<
1144                         /p>');
1145                     }
1146                 }
1147             }
1148         }
1149     }
1150 }
1151 }
1152 } else {
1153     if ($('#input_defs').val() == null) {
1154         for (loc in chosenLocations) {
1155             $('#modal-content').append('<h3>' +
1156             chosenLocations[loc] + '</h3>');
1157             for (prov in chosenProviders) {
1158                 $('#modal-content').append('<h4>' +
1159                 chosenProviders[prov] + '</h4>');
1160                 if ( $.inArray(chosenProviders[prov],
1161                 getProviders(chosenLocations[loc])) != -
1162                 1 ) {
1163                     var tmpDefs = getDefs(
1164                     chosenLocations[loc],
1165                     chosenProviders[prov]);
1166                     for (def in tmpDefs) {
1167                         var tmpInterval = getInterval(
1168                         chosenLocations[loc],
1169                         chosenProviders[prov], tmpDefs[
1170                         def]);
1171                         for (intvl in tmpInterval) {
1172                             var tmpParts =
1173                             intervalToTemplate(
1174                             tmpInterval[intvl].split('-')
1175                             );
1176                             for (tmpl in tmpParts) {
1177                                 $('#modal-content').
1178                                 append('<p>' + tmpDefs[
1179                                 def] + tmpParts[tmpl] +
1180                                 '</p>');
1181                             }
1182                         }
1183                     }
1184                 }
1185             }
1186         }

```

```

1187     } else {
1188         for (loc in chosenLocations) {
1189             $('#modal-content').append('<h3>' + 2
1190             chosenLocations[loc] + '</h3>');
1191             for (prov in chosenProviders) {
1192                 $('#modal-content').append('<h4>' + 2
1193                 chosenProviders[prov] + '</h4>');
1194                 if ( $.inArray(chosenProviders[prov], 2
1195                 getProviders(chosenLocations[loc])) != -2
1196                 1 ) {
1197                     for (def in chosenDefs) {
1198                         if ( $.inArray(chosenDefs[def], 2
1199                         getDefs(chosenLocations[loc], 2
1200                         chosenProviders[prov])) != -1 ) {
1201                             var tmpInterval = 2
1202                             getInterval(chosenLocations[2
1203                             loc], chosenProviders[prov], 2
1204                             chosenDefs[def]);
1205                             for (intvl in tmpInterval) {
1206                                 var tmpParts = 2
1207                                 intervalToTemplate(2
1208                                 tmpInterval[intvl].split(2
1209                                 '-'));
1210                                 for (tmpl in tmpParts) {
1211                                     $('#modal-content').2
1212                                     append('<p>' + 2
1213                                     chosenDefs[def] + 2
1214                                     tmpParts[tmpl] + '<2
1215                                     /p>');
1216                                 }
1217                             }
1218                         }
1219                     }
1220                 }
1221             }
1222         }
1223     }
1224 }
1225
1226 function showLoading() {
1227     $('#modal_wrapper').removeClass('hide');
1228     $('#loading_anim').removeClass('hide');
1229 }
1230
1231 function hideLoading() {
1232     $('#modal_wrapper').addClass('hide');
1233     $('#loading_anim').addClass('hide');
1234 }
1235
1236 function hideLoadingWOWrapper() {
1237     $('#loading_anim').addClass('hide');
1238 }
1239
1240 function setError() {
1241     $('#error').removeClass('hide');
1242     $('#error').addClass('show');
1243     $('#error').append('<p>' + lang.needupdate + '<a 2
1244     id="gonnaupdate">' + lang.gonnaupdate + '</a>' + '</p>');
1245     $('#error').append();
1246     $('#gonnaupdate').click(updateBase);
1247     $('#error').append('<span id="unSetError" 2
1248     class="glyphicon glyphicon-remove" aria-hidden="true"><2
1249     /span>');
1250     $('#unSetError').click(unSetError);
1251 }
1252
1253 }

```

```

1254 ;
1255 function unSetError() {
1256     $('#error').removeClass('show');
1257     $('#error').addClass('hide');
1258     setTimeout(function () {
1259         $('#error').empty();
1260     }, 200);
1261 };
1262
1263 function chooseFile() {
1264     var chooser = $('#chooseDirectory');
1265     var val;
1266     chooser.unbind('change');
1267     chooser.change(function(evt) {
1268         saveTo = $(this).val() + '/';
1269         $('#saveTo').text(saveTo);
1270     });
1271     chooser.trigger('click');
1272 }
1273
1274 function openExportModal() {
1275     log('Export modal is opened');
1276     $('#.nano').nanoScroller();
1277     $('#.modal').addClass('nano');
1278     $('#.modal').append('<div class="nano-content" id="modal-2'
1279     content"></div>');
1280     $('#modal-content').append('<div 2
1281     class="change_path_block"></div>');
1282     $('#.change_path_block').append('<input style="display: 2
1283     none" id="chooseDirectory" type="file" nwdirectory 2
1284     nwworkingdir="' + window.location.pathname + '" />');
1285     $('#.change_path_block').append('<span id="currPath">'+2
1286     lang.currPath+ '</span>');
1287     $('#.change_path_block').append('<span id="saveTo">'+2
1288     saveTo+ '</span>');
1289     $('#.change_path_block').append('<button 2
1290     id="change_path_button">'+lang.change_path+'</button>');
1291     $('#change_path_button').click(chooseFile);
1292
1293     $('#modal-content').append('<button class="export_btn" 2
1294     id="export_to_txt">'+ lang.export_to_txt + '</button>');
1295     $('#export_to_txt').click(function () {
1296         exportToTXT();
1297         alert(lang.successful_export);
1298     });
1299
1300     $('#modal-content').append('<button class="export_btn" 2
1301     id="export_to_excel">'+ lang.export_to_excel + '<2
1302     /button>');
1303     $('#export_to_excel').click(function () {
1304         exportToExcel();
1305     });
1306     $('#modal-content').append('<button class="export_btn" 2
1307     id="btn_close_export_modal">OK</button>');
1308     $('#btn_close_export_modal').click(function () {
1309         closeExportModal();
1310     });
1311     $('#modal_wrapper').removeClass('hide');
1312     $('#.modal').removeClass('preinit');
1313
1314     buildChoose();
1315     exportToScreen();
1316
1317     $('#modal-content').append('<button class="export_btn" 2
1318     id="export_to_txt_2">'+ lang.export_to_txt + '</button>2
1319     ');
1320     $('#export_to_txt_2').click(function () {
1321         exportToTXT();
1322         alert(lang.successful_export);
1323     });

```

```

1324     });
1325     $('#modal-content').append('<button class="export_btn" >
1326     id="export_to_excel_2">' + lang.export_to_excel + '<
1327     /button>');
1328     $('#export_to_excel_2').click(function () {
1329         exportToExcel();
1330     });
1331     $('#modal-content').append('<button class="export_btn" >
1332     id="btn_close_export_modal_2">OK</button>');
1333     $('#btn_close_export_modal_2').click(function () {
1334         closeExportModal();
1335     });
1336 }
1337 function closeExportModal() {
1338     location.reload();
1339 }
1340 function openAboutModal() {
1341     $('#modal_wrapper').removeClass('hide');
1342     $('#.modal').removeClass('preinit');
1343     $('#.modal').append('<div class="github-fork-ribbon-
1344     wrapper right">'+
1345         '<div class="github-fork-
1346         ribbon">' +
1347         '<a href="https:
1348         //github.
1349         com/Kostydenis/practi
1350         ce2015">Fork me on
1351         GitHub</a>' +
1352         '</div>' +
1353         '</div>');
1354     $('#.modal').append('<h3>'+lang.btn_about+'</h3>');
1355     $('#.modal').append('<p>'+lang.about_page+'<a >
1356     href="mailto:kostydenis@gmail.com">kostydenis@gmail.com<
1357     /a>'+</p>');
1358     $('#.modal').append('<p>'+lang.about_manual+'<a >
1359     href="https://github.com/Kostydenis/practice2015">'+lang.>
1360     open+'</a></p>');
1361     $('#modal_wrapper').removeClass('hide');
1362     $('#.modal').removeClass('preinit');
1363     $('#.modal').append('<button id="btn_modal_ok">OK</button>
1364     ');
1365     $('#btn_modal_ok').click(closeModal);
1366 }
1367
1368 function openModal(msg) {
1369     log('Opened common modal with message: '+msg);
1370     $('#modal_wrapper').removeClass('hide');
1371     $('#.modal').removeClass('preinit');
1372     $('#.modal').append('<p>'+ msg + '</p>');
1373     $('#.modal').append('<button id="btn_modal_ok">OK</button>
1374     ');
1375     $('#btn_modal_ok').click(closeModal);
1376 }
1377
1378 function closeModal() {
1379     $('#modal_wrapper').addClass('hide');
1380     $('#.modal').addClass('preinit');
1381     $('#.modal').empty();
1382 }
1383
1384 function findProvider(input_def, input_number) {
1385     log('Starting find provider');
1386     var output = {};
1387     var tmpLocations;
1388     var tmpProviders;
1389     var tmpDefs;

```

```

1393
1394     for (loc in locations) {
1395         tmpProviders = getProviders(locations[loc]);
1396         for (prov in tmpProviders) {
1397             tmpDefs = getDefs(locations[loc], tmpProviders[prov]);
1398             for (def in tmpDefs) {
1399                 if (tmpDefs[def] == input_def) {
1400                     var tmpInterval = getInterval(locations[loc], tmpProviders[prov], tmpDefs[def]);
1401                     for (intvl in tmpInterval) {
1402                         var currInterval = tmpInterval[intvl].toString().split('-');
1403                         if (isInInterval(currInterval[0], input_number, currInterval[1])) {
1404                             output.loc = locations[loc];
1405                             output.prov = tmpProviders[prov];
1406                             base_init();
1407                             return output;
1408                         }
1409                     }
1410                 }
1411             }
1412         }
1413     }
1414 }
1415 }
1416 }
1417 }
1418 }
1419
1420 output.error = lang.error_number_not_found;
1421 base_init();
1422 return output;
1423 }
1424
1425 function initBaseTemplate() {
1426     var tip = "";
1427     if (os.platform() === 'darwin') {
1428         tip = '<p class="input_tip">' + lang.input_tip_mac + '</p>';
1429     } else {
1430         tip = '<p class="input_tip">' + lang.input_tip_win + '</p>';
1431     }
1432     var langbuttons;
1433     if (chosenLang === 'ru') {
1434         langbuttons =
1435             '<div id="btn_en" class="switch_lang_btn not_pressed">en</div>' +
1436             '<div id="btn_ru" class="switch_lang_btn">ru</div>';
1437     } else {
1438         langbuttons =
1439             '<div id="btn_en" class="switch_lang_btn">en</div>' +
1440             '<div id="btn_ru" class="switch_lang_btn not_pressed">ru</div>';
1441     }
1442     $('body').empty();
1443     $('body').append(
1444         '<div id="modal_wrapper" class="modal_wrapper hide"></div>' +
1445         '<div id="modal" class="modal preinit"></div>' +
1446         '' +
1447         '<header>' +
1448         '' +
1449         '<h1></h1>' +
1450         '</header>' +
1451     );
1452 }

```



```

1462         '<div class="content">' +
1463         '<div id="error" class="hide">' +
1464         '</div>' +
1465         '<div class="block_number_check preinit">' +
1466         '<input type="text" id="input_number_check" >' +
1467         '>' +
1468         '<button id="button_number_check"><span >' +
1469         'class="glyphicon glyphicon-search" aria->' +
1470         'hidden="true"></span></button>' +
1471         '<p class="NB" id="note_mnp"></p>' +
1472         '</div>' +
1473         '<div class="form_request preinit">' +
1474         '<h2 id="header_form_request"></h2>' +
1475         'tip +
1476         '<select id="input_locations" class="chosen->' +
1477         'select" multiple >' +
1478         '</select>' +
1479         '<select id="input_providers" class="chosen->' +
1480         'select" multiple >' +
1481         '</select>' +
1482         '<select id="input_defs" class="chosen->' +
1483         'select" multiple >' +
1484         '</select>' +
1485         '<button id="btn_form_request"></button>' +
1486         '<button id="btn_clear_form_request"><' +
1487         '/button>' +
1488         '</div>' +
1489         '<div class="update">' +
1490         '<button id="btn_update" ></button>' +
1491         '<span class="last_update">2015-08-3</span>' +
1492         '</div>' +
1493         '<div class="switch_lang">' +
1494         'langbuttons +
1495         '</div>' +
1496         '</div>' +
1497         '<div class="switch_lang">' +
1498         'langbuttons +
1499         '</div>' +
1500         '</div>' +
1501         '<div class="switch_lang">' +
1502         'langbuttons +
1503         '</div>' +
1504         '<div class="switch_lang">' +
1505         'langbuttons +
1506         '</div>' +
1507         '<div class="switch_lang">' +
1508         'langbuttons +
1509         '</div>' +
1510     }
1511     init('ru');
1512 
```

Приложение В: Листинг main.scss

Листинг В.1— main.scss

```

1 @import "fonts.scss";
2
3 $primary-color-light: #7B1FA2;
4 $primary-color: #9C27B0;
5 $primary-color-light: #E1BEE7;
6 $primary-color-text: #FFFFFF;
7 $accent-color: #607D8B;
8 $primary-text-color: #212121;
9 $secondary-text-color: #727272;
10 $divider-color: #B6B6B6;
11 $error-color: rgba(240,0,0,0.8);
12
13 @mixin shadow_lv11 {
14   box-shadow: 1px 2px 4px 0px rgba(0, 0, 0, 0.5);
15 }
16
17 @mixin shadow_lv12 {
18   box-shadow: 0px 4px 8px 0px rgba(0, 0, 0, 0.5);
19 }
20
21 @mixin shadow_lv15 {
22   box-shadow: 0px 0px 15px 7px rgba(0, 0, 0, 0.5);
23 }
24
25 .clearfix {
26   &:after {
27     content: "";
28     display: block;
29     clear: both;
30   }
31 }
32
33 @font-face {
34   font-family: Roboto;
35   src: local("Roboto");
36   src: url(../_fonts/Roboto.ttf);
37 }
38
39 body {
40   font-family: Roboto;
41 }
42
43 header {
44   h1 {
45     position: absolute;
46     top: 0;
47     right: 10px;
48     font-size: x-large;
49     color: $primary-color-text;
50     text-shadow: 0px 0px 20px rgba(0,0,0,1);
51   }
52   img {
53     position: relative;
54     width: 100%;
55     -webkit-filter: opacity(.6);
56   }
57 }
58
59 .content {
60   margin-bottom: 100px;
61   position: relative;
62   z-index: 5;
63 }
64
65 .block_number_check {

```

```

66
67 position: relative;
68 margin-top: -15px;
69 margin-bottom: 10px;
70 width: 100%;
71 transition: margin-top ease 0.2s;
72
73 #input_number_check {
74   position: relative;
75   top: 0px;
76   padding: 10px;
77   left: 5%;
78   height: 60px;
79   width: 90%;
80   margin: 0 auto;
81   font-size: x-large;
82   border: none;
83   @include shadow_lv11;
84 }
85
86 #button_number_check {
87   position: relative;
88   top: 0px;
89   right: 20px;
90   height: 30px;
91   width: 30px;
92 }
93 .error {
94   box-shadow: 0px 0px 25px 5px $error-color !important;
95 }
96 #label_error_number_check {
97   position: relative;
98   top: 0px;
99   color: $error-color;
100   width: 100%;
101   text-align: center;
102 }
103
104 .NB {
105   &:before {
106     content: ' * ' ;
107     color: red;
108     font-size: 14px;
109   }
110   position: relative;
111   top: 0px;
112   left: 5%;
113   width: 90%;
114   color: $secondary-text-color;
115   font-size: 10px;
116   margin: 0 2%;
117
118   #learnmore_mnp {
119     text-decoration: underline;
120     cursor: pointer;
121     &:hover {color: $divider-color;}
122   }
123 }
124 }
125
126
127 .form_request {
128
129   position: relative;
130   transition: margin-top ease 0.2s 0.2s;
131   border: solid 1px $divider-color;
132   width: 90%;
133   margin-left: 5%;
134   z-index: 2;
135   background-color: #FFF;
136   @include shadow_lv11;
137
138   #header_form_request {

```

```

140     margin-left: 5%;
141     margin-bottom: 0;
142     color: $divider-color;
143 }
144 .input_tip {
145     margin: 0 7% 10px;
146     color: #515151;
147     font-size: smaller;
148     &:before {
149         content: '*';
150         color: red;
151         font-size: 16px;
152     }
153 }
154
155 #btn_form_request {
156     margin: 20px auto;
157     margin-left: 5%;
158     height: 40px;
159     width: 150px;
160 }
161 #btn_clear_form_request {
162     height: 40px;
163     margin-left: 10px;
164 }
165
166 }
167
168 #error {
169     position: fixed;
170     top: 10px;
171     left: 10%;
172     width: 80%;
173     border: solid 1px #BA5565;
174     border-radius: 4px;
175     background-color: #F2DEDE;
176     padding: 0px;
177     z-index: 100;
178     @include shadow_lvl2;
179
180     p {
181         margin: 10px;
182         color: #A94442;
183     }
184     span {
185         position: absolute;
186         right: 10px;
187         top: 10px;
188         &:hover {
189             transform: scale(1.1);
190             cursor: pointer;
191         }
192     }
193     a {
194         text-decoration: underline;
195         &:hover {
196             cursor: pointer;
197         }
198     }
199 }
200 .hide {
201     display: none;
202 }
203 .show {
204     display: block;
205 }
206
207 input,
208 select {
209     position: relative;

```

```

210 top: 0px;
211 padding: 10px;
212 left: 5%;
213 height: 60px;
214 width: 90%;
215 margin: 5px auto;
216 font-size: x-large;
217 border: none;
218 @include shadow_lv11;
219
220
221 }
222 input::-webkit-input-placeholder {
223   opacity: 0.5;
224 }
225
226 button {
227   border: none;
228   border-radius: 2px;
229   background-color: $accent-color;
230   color: $primary-color-text;
231   @include shadow_lv11;
232   transform: scale(1);
233   transition: all ease 0.2s;
234
235   &:hover {
236     @include shadow_lv12;
237     transform: scale(1.03);
238     transition: all ease 0.2s;
239   }
240 }
241
242 .update {
243   @extend .clearfix;
244   margin: 10px 0 0 5%;
245   float: left;
246 }
247
248 #btn_update {
249   margin: 0;
250   height: 40px;
251   width: 150px;
252 }
253
254 .last_update {
255   margin-top: 2px;
256   display: block;
257   font-size: smaller;
258   color: $divider-color;
259 }
260
261 .switch_lang {
262   @extend .clearfix;
263   margin-top: 15px;
264   margin-right: 5%;
265   float: right;
266
267   .switch_lang_btn {
268     float: right;
269     width: 40px;
270     height: 26px;
271     color: $primary-color-text;
272     text-align: center;
273     background-color: $accent-color;
274     padding: 3px;
275
276     &:hover {
277       cursor: pointer;
278       color: $primary-color-text;
279     }
280   }
281
282   #btn_ru {
283     border-top-left-radius: 2px;
284     border-bottom-left-radius: 2px;

```

```

283 }
284 #btn_en {
285   border-top-right-radius: 2px;
286   border-bottom-right-radius: 2px;
287 }
288 .not_pressed {
289   color: $divider-color;
290   @include shadow_lv11;
291 }
292 }
293
294 .change_path_block {
295   border: solid 1px $divider-color;
296   @include shadow_lv11;
297   padding: 10px;
298   #currPath {
299     display: block;
300     word-wrap: break-word;
301     color: $divider-color;
302   }
303   #saveTo {
304     display: block;
305     word-wrap: break-word;
306   }
307   #change_path_button {
308     margin-top: 10px;
309     width: 160px;
310     height: 40px;
311   }
312 }
313
314 footer {
315   position: fixed;
316   width: 100%;
317   bottom: 0px;
318   text-align: center;
319   color: $secondary-text-color;
320   z-index: 1;
321
322   p {
323     margin: 0;
324   }
325   a {
326     text-decoration: underline;
327     color: $secondary-text-color;
328     &:hover {
329       color: $divider-color;
330       cursor: pointer;
331     }
332   }
333 }
334
335
336 .ui-autocomplete {
337   list-style-type: none;
338   background-color: #ffffff;
339   border: solid 1px $divider-color;
340   width: 76%;
341   z-index: 10 !important;
342 }
343 .ui-state-focus {
344   color: $accent-color;
345   cursor: pointer;
346 }
347
348 .ui-helper-hidden-accessible {
349   display: none;
350 }
351
352 .modal_wrapper{

```

```

353 position: fixed;
354 top: 0;
355 left: 0;
356 width: 100%;
357 height: 100%;
358 background-color: #FFF;
359 opacity: 0.8;
360 z-index: 1000;
361 }
362 .modal {
363
364 position: fixed !important;
365 width: 70% !important;
366 max-height: 80%;
367 // height: 80%;
368 top: 5%;
369 transition: top ease 0.2s;
370 left: 15%;
371 background-color: $primary-color-text;
372 @include shadow_lvl5;
373 padding: 20px;
374
375 z-index: 1001;
376
377 .nano-content {
378 padding: 20px !important;
379 h3 {
380 margin: 10px 0 5px 0;
381 }
382 h4 {
383 margin: 5px 0 5px 0;
384 }
385 p {
386 margin: 0;
387 }
388 }
389 a {
390 text-decoration: underline;
391 color: $secondary-text-color;
392 &:hover {
393 color: $divider-color;
394 cursor: pointer;
395 }
396 }
397
398 #btn_modal_ok {
399 width: 80px;
400 height: 30px;
401 }
402 .export_btn {
403 width: 160px;
404 height: 40px;
405 margin-right: 15px;
406 margin-top: 15px;
407 }
408
409 }
410 #loading_anim {
411 position: fixed;
412 left: 50%;
413 top: 50%;
414 margin: -32 -32 0 0;
415 z-index: 1001;
416 }
417 .preinit {
418 margin-top: 100%;
419 top: 110%;
420 transition: all ease 0.4s;
421 }
422
423
424 // ----- CHOSEN PLUGIN STYLE -----

```

```
425 <...>
426 // ----- END CHOSEN STYLE
427 // ----- GITHUB RIBBON STYLE
428 <...>
429 //----- END GITHUB RIBBON STYLE
```