**Приложение Б**

**Руководство пользователя**

**репозитория инсталляционных пакетов**

**корпоративной сети**

Содержание

[1 Вступление 3](#_Toc453000877)

[2 Java 4](#_Toc453000878)

[3 WildFly 4](#_Toc453000879)

[3.1 Ошибка при запуске 5](#_Toc453000880)

[3.2 Добавление пользователя 6](#_Toc453000881)

[4 Среда разработки 8](#_Toc453000882)

[5 База данных 8](#_Toc453000883)

[6 Создание проекта в IntelliJ Idea 12](#_Toc453000884)

[6.1 Создание модулей 12](#_Toc453000885)

[6.2 Создание артефакта 17](#_Toc453000886)

[6.3 Создание конфигурации запуска 19](#_Toc453000887)

[6.4 Добавление библиотеки 21](#_Toc453000888)

[6.5 Запуск 22](#_Toc453000889)

[7 Структура проекта 22](#_Toc453000890)

[7.1 Контроллеры 24](#_Toc453000891)

[7.2 Модели 25](#_Toc453000892)

[8 Сборка 28](#_Toc453000893)

[9 Работа с WildFly 29](#_Toc453000894)

[9.1 Деплой на сервер приложений 29](#_Toc453000895)

[9.2 Перезагрузка приложения 30](#_Toc453000896)

[10 Интерфейс пользователя 31](#_Toc453000897)

[10.1 Настройки 31](#_Toc453000898)

[10.2 Пользовательский интерфейс обычного пользователя 33](#_Toc453000899)

[10.3 Пользовательский интерфейс модератора 35](#_Toc453000900)

[10.4 Пользовательский интерфейс администратора 42](#_Toc453000901)

1. Вступление

Репозиторий инсталляционных пакетов разработан на языке Java с использованием Spring framework 4. Руководство ориентировано на разработку в ОС Windows.

Реализованные функции:

Для пользователей:

* Сквозная аутентификация через Active Directory;
* Регистрация заявок на добавление инсталляционных пакетов;
* Сохранений заявок в базе данных;
* Загрузка файлов на сервер;
* Просмотр каталога ПО;
* Просмотр информации о файле;
* Выгрузка файлов с сервера.

Для модератора:

* Сквозная аутентификация через Active Directory;
* Просмотр каталога ПО;
* Просмотр информации о файле;
* Загрузка файлов на сервер;
* Выгрузка файлов с сервера;
* Заполнение информации о файлах; тип информации будет уточнен в процессе разработки;
* Редактирование информации о файлах;
* Сохранение информации о файлах в базе данных;
* Просмотр списка файлов;
* Создание новых категорий для группировки файлов;
* Сохранение категорий в базе данных;
* Удаление категорий;
* Привязка файлов к категориям;
* Просмотр файлов, информация о которых еще не заполнена;
* Просмотр информации о пользователях; тип информации будет уточнен в процессе разработки.

Для администратора системы:

* Все возможности модератора;
* Изменить настройки;
* Удаление файлов;
* Выполнение скриптов для экспорта файлов в SCCM.

1. Java

Для начала разработки необходимо установить Java JDK, которую мы можем скачать [с официального сайта Oracle](http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html).

1. WildFly

Для запуска приложения нужен сервер приложений WildFly. Скачать можно с [официального сайта](http://wildfly.org/downloads/) (рисунок 3.1).

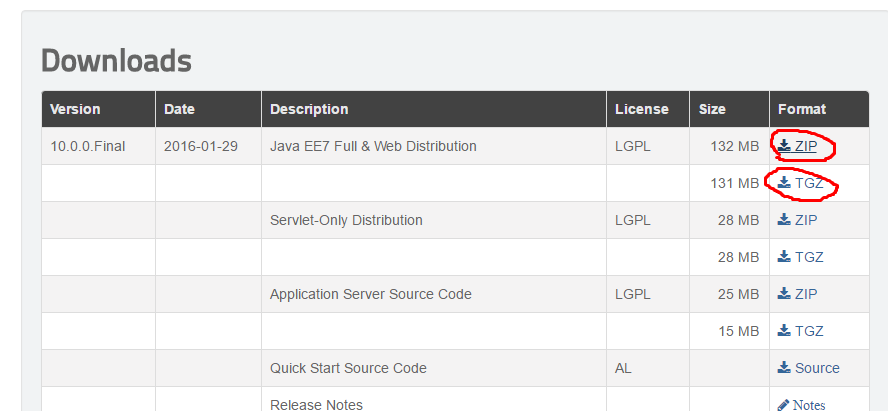


Рисунок 3.1 – Страница загрузки сервера приложений WildFly

Скачиваем и распаковываем архив в любое удобное место. Для запуска сервера необходимо запустить bin/standalone.bat.

* 1. Ошибка при запуске

Во время запуска на Windows может возникнуть ошибка представленная на рисунке 3.2.

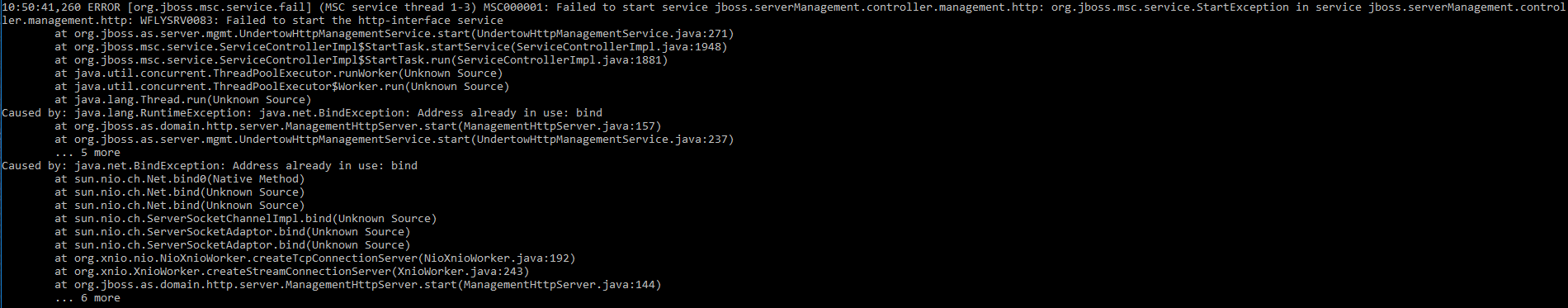


Рисунок 3.2 – Ошибка при запуске WildFly

Это значит, что требуемые для запуска порты заняты. Для того, чтобы изменить порты открываем файл standalone/configuration/standalone.xml, ищем внизу файла строки:

<socket-binding-group name="standard-sockets" default-interface="public" port-offset="${jboss.socket.binding.port-offset:0}">

<socket-binding name="management-http" interface="management" port="${jboss.management.http.port:9990}"/>

<socket-binding name="management-https" interface="management" port="${jboss.management.https.port:9993}"/>

<socket-binding name="ajp" port="${jboss.ajp.port:8009}"/>

<socket-binding name="http" port="${jboss.http.port:8080}"/>

<socket-binding name="https" port="${jboss.https.port:8443}"/>

<socket-binding name="txn-recovery-environment" port="4712"/>

<socket-binding name="txn-status-manager" port="4713"/>

<outbound-socket-binding name="mail-smtp">

<remote-destination host="localhost" port="25"/>

</outbound-socket-binding>

</socket-binding-group>

Необходимо изменить порты в ${jboss.management.http.port:9990} и ${jboss.management.https.port:9993} на другие. В данном случае, порт 9990 занят драйверами NVidia, поэтому меняем 9990 на 9991, сохраняем и запускаем сервер еще раз. Если все прошло без ошибок, то открываем <http://127.0.0.1:9991/>, и мы должны увидеть стартовую страницу, внешний вид которой представлен на рисунке 3.3.

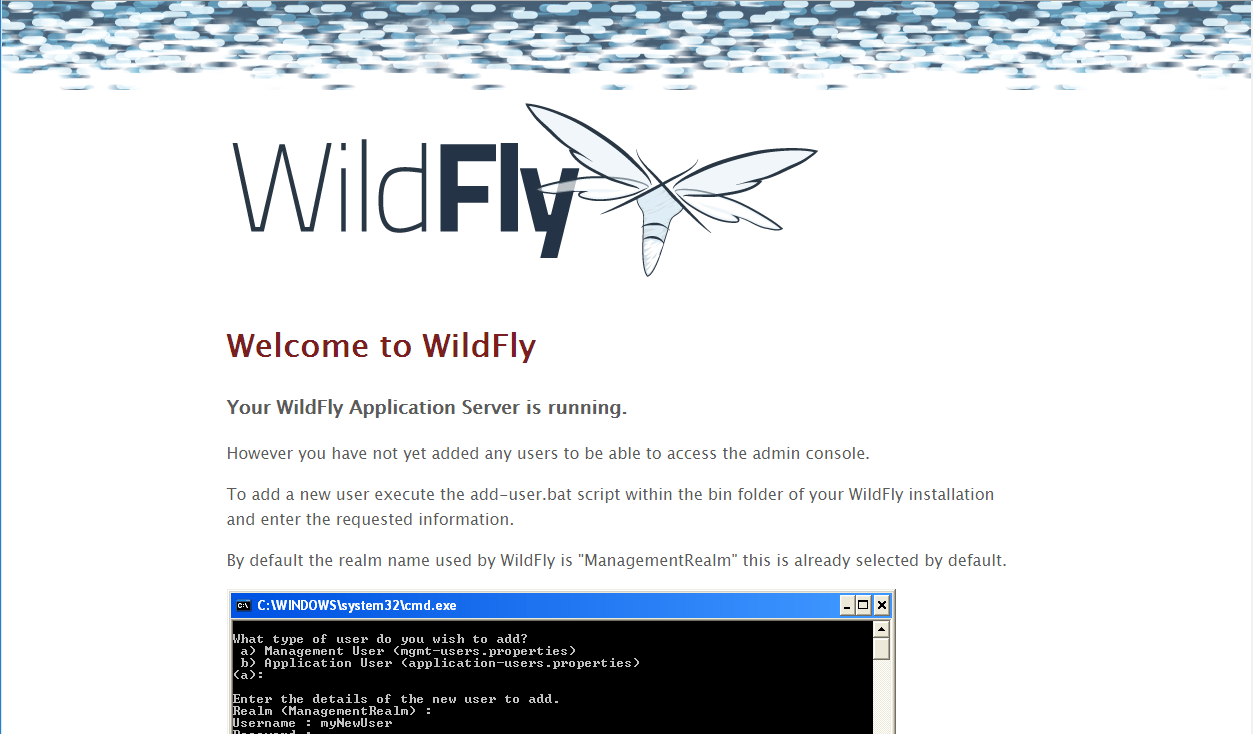


Рисунок 3.3 – Внешний вид стартовой страницы WildFly

* 1. Добавление пользователя

На стартовой странице написано, что нужно добавить пользователя. Для этого, запускаем /bin/add-user.bat. Выбираем a) Management user и заполняем требуемые поля. Результат добавления пользователя представлен на рисунке 3.4.

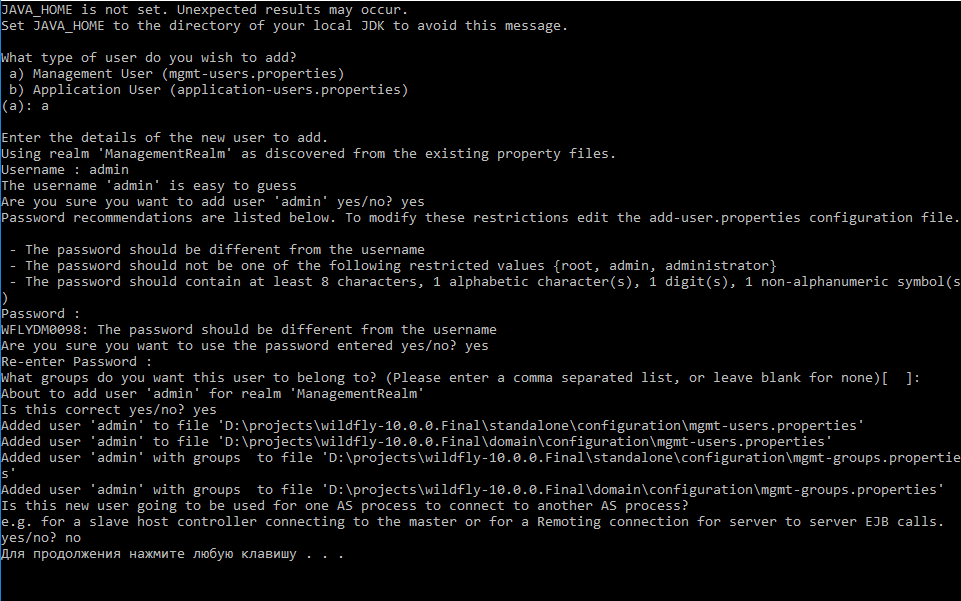


Рисунок 3.4 – Результат добавления пользователя

После добавления пользователя переходим по адресу <http://127.0.0.1:9991/>, где мы должны увидеть панель администратора, внешний вид которой представлен на рисунке 3.5.

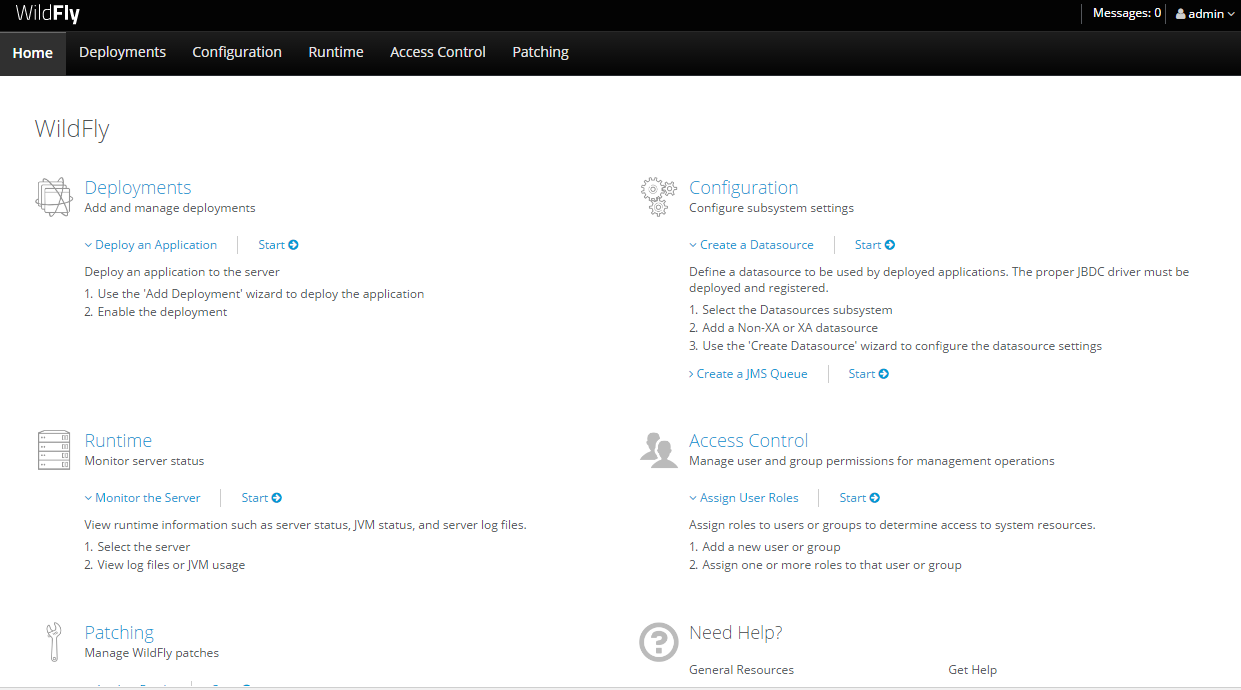


Рисунок 3.5 – Панель администратора WildFly

1. Среда разработки

Будем рассматривать процесс разработки в среде IntelliJ Idea, которую мы можем скачать с [официального сайта](https://www.jetbrains.com/idea/download/).

1. База данных

База данных создается с помощью миграции. Файл миграции лежит в /migrations/.

Для работы с миграциями необходимо скачать Flyway. У этого приложения есть инструмент для работы из командной строки. Скачать его можно на [официальном сайте](https://flywaydb.org/getstarted/download) (рисунок 5.1).



Рисунок 5.1 – Страница загрузки Flyway

Рассмотрим на примере для Windows. Скачиваем и распаковываем архив. Нам нужны будут 2 папки, представленные на рисунке 5.2.

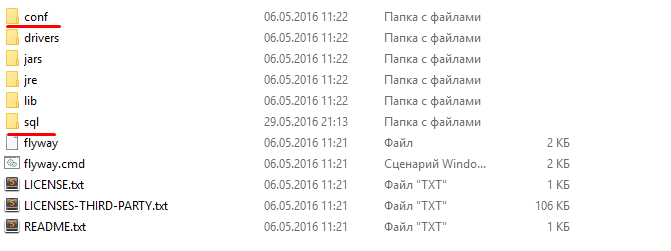


Рисунок 5.2 – Структура программы Flyway

Теперь необходимо настроить адрес до БД. Заходим в распакованную папку и открываем файл conf/flyway.conf.

Находим строчку «flyway.url=». Выше, в комментариях есть примеры того, что надо писать для разных БД. Например, mariadb:

flyway.url=jdbc:mariadb://127.0.0.1:3306/test

Теперь копируем в папку sql/ файл миграции из проекта /migrations/V1\_\_Repository\_migration.sql.

Запускаем cmd, переходим в распакованную папку, например, flyway-4.0.1 и выполняем команду:

flyway **migrate**

Необходимо вести пользователя для БД и его пароль, после чего будет применена миграция. Результат выполнения представлен на рисунке 5.3.

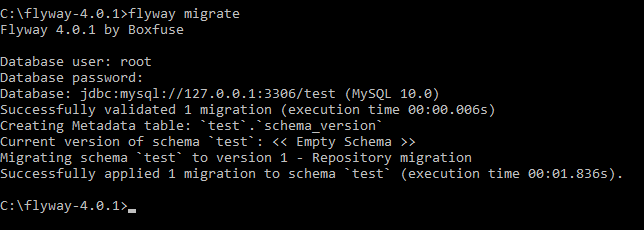


Рисунок 5.3 – Результат применения мигарции

Логическая схема базы данных приведена на рисунке 5.4.

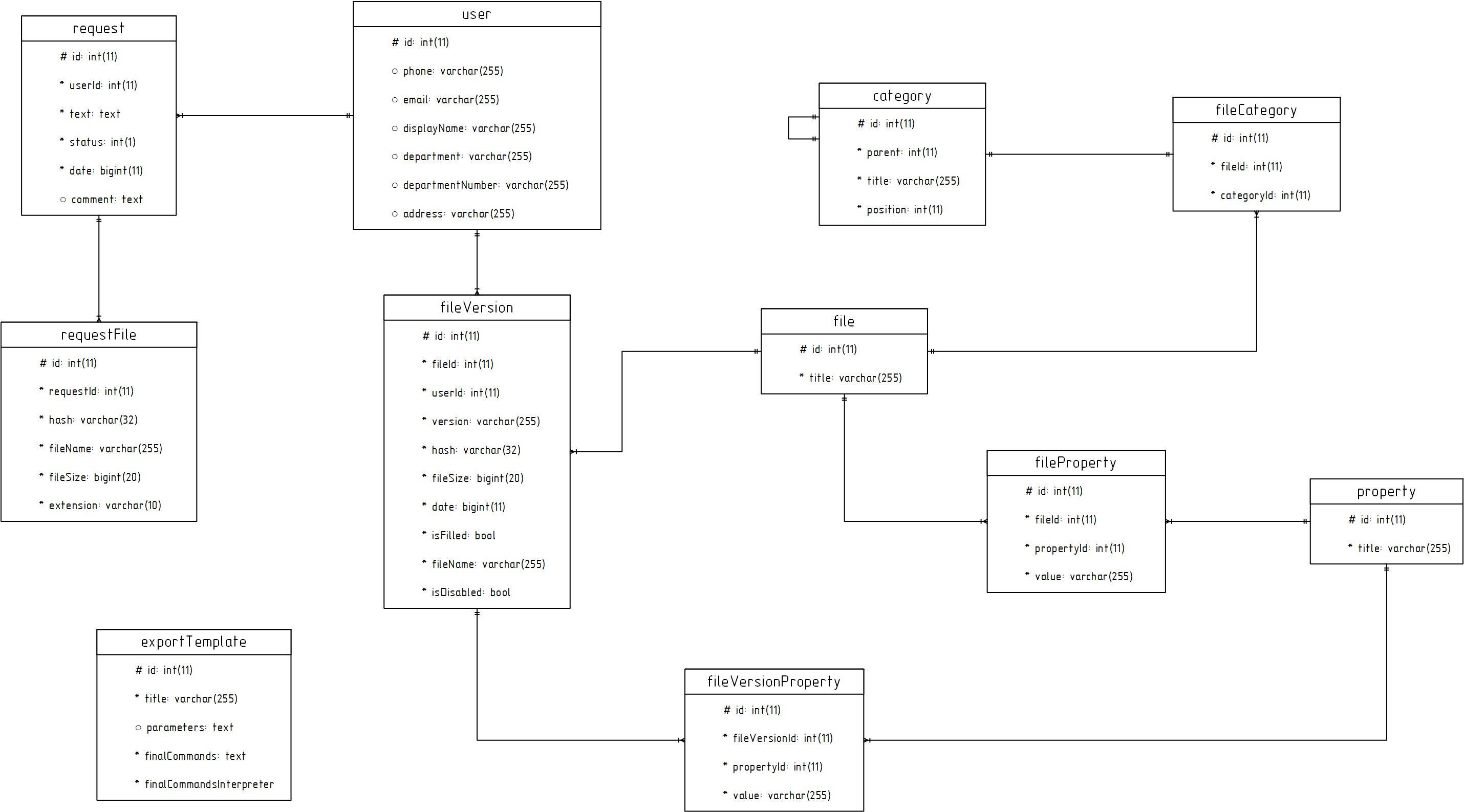


Рисунок 5.4 – Логическая схема базы данных

1. Создание проекта в IntelliJ Idea

Для создания проекта нажимаем «Import project» и выбираем путь до исходных файлов, до файла pom.xml, как показано на рисунке 6.1.

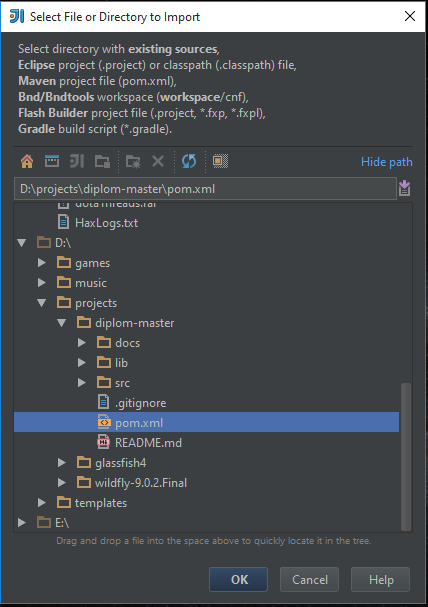


Рисунок 6.1 – Выбор файла pom.xml для импорта проекта

Жмем «ok», далее оставляем все без изменений, нажимаем «next», «next», «finish».

* 1. Создание модулей

Нажимаем «File» -> «Project Structure» -> «Modules», правой кнопкой по «repository» -> «add» -> «Web», как показано на рисунке 6.2.

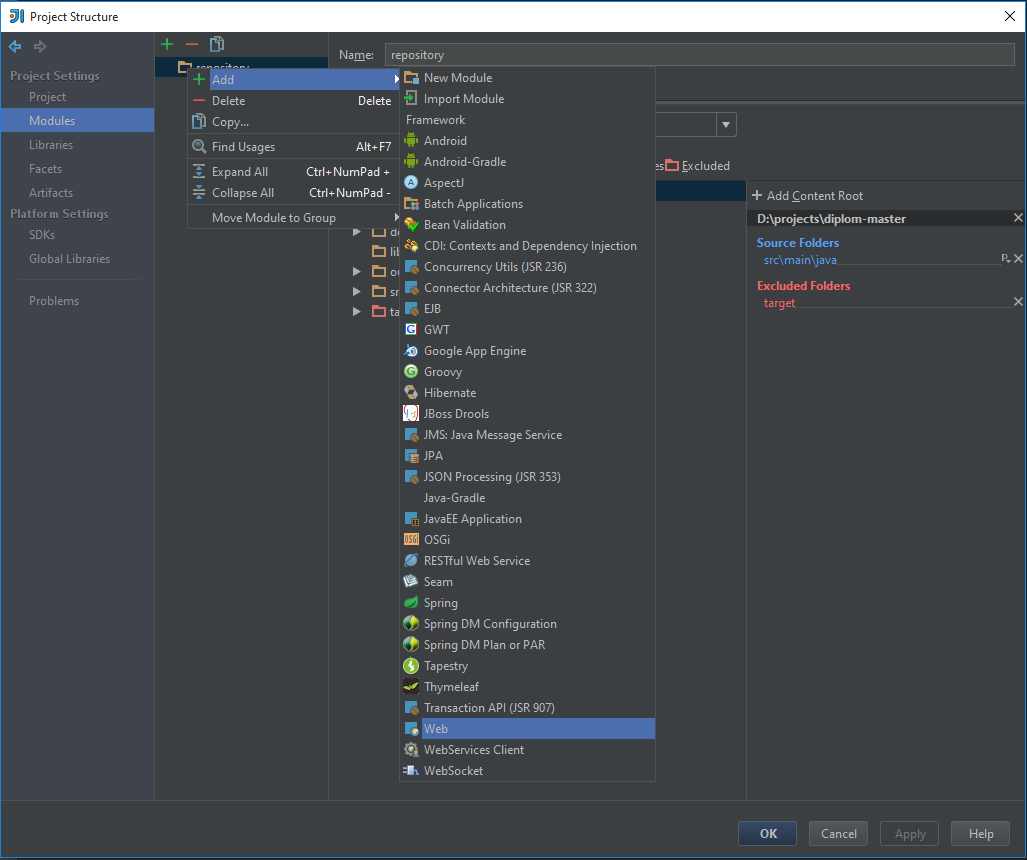


Рисунок 6.2 – Создание модуля «Web»

Далее необходимо настроить пути до папки webapp, как показано на рисунке 6.3.

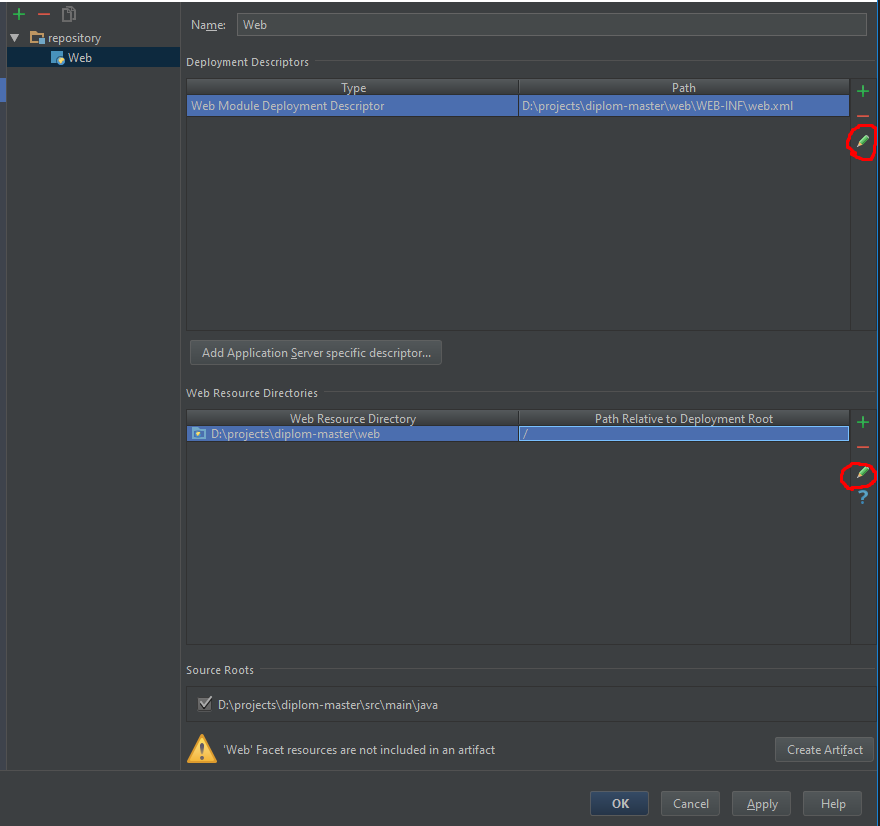


Рисунок 6.3 – Редактирование настроек модуля «Web»

В верхнем пути выбираем путь до web.xml, как показано на рисунке 6.4:

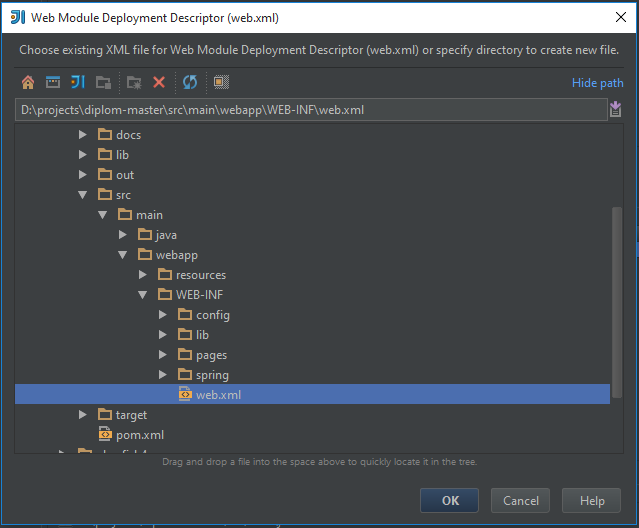


Рисунок 6.4 – Выбор путь до файла web.xml

В нижнем – путь до папки webapp, как показано на рисунке 6.5:

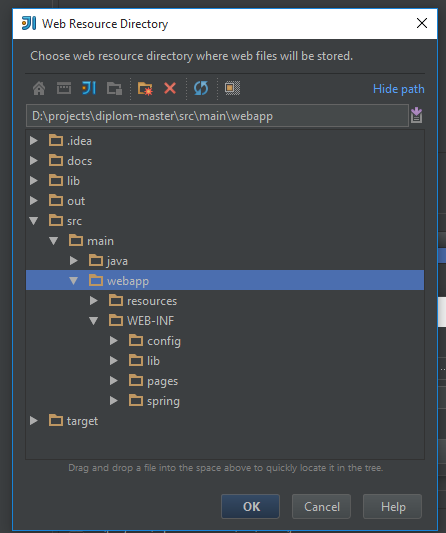


Рисунок 6.5 – Настройка пути до папки webapp

Добавляем еще 1 модуль «Spring», как показано на рисунке 6.6.

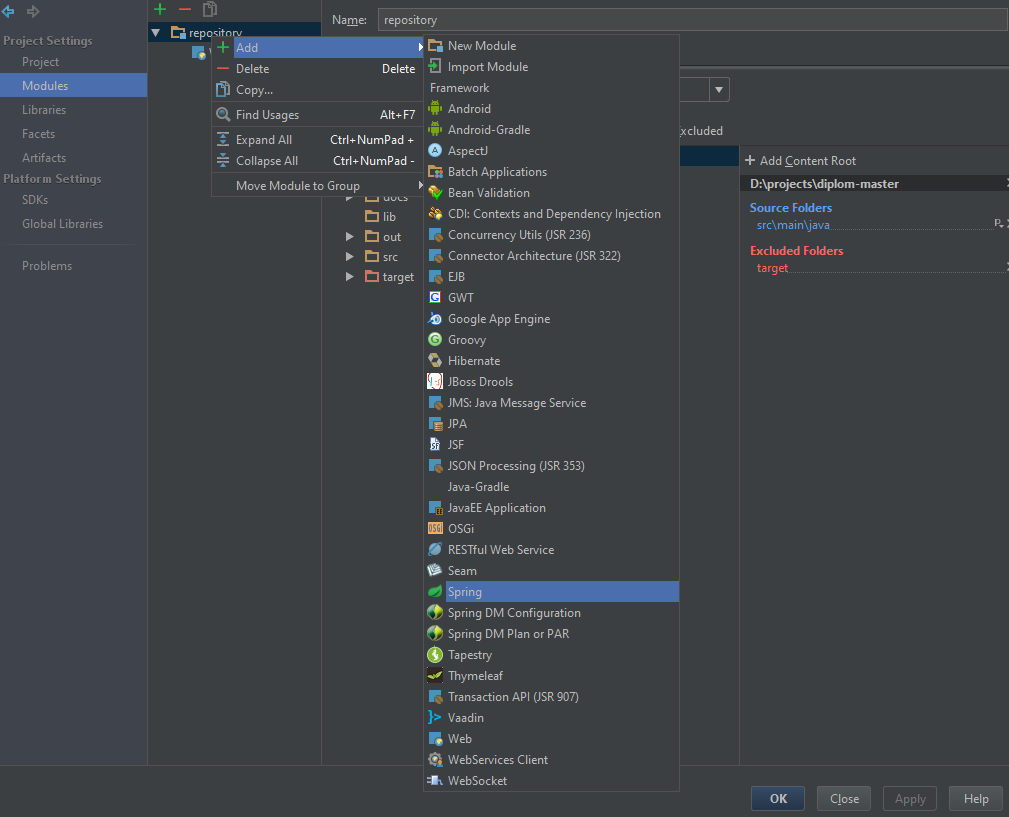


Рисунок 6.6 – Добавление модуля «Spring»

Нажимаем на «+» и в появившемся окне выбираем все файлы и нажимаем «ok», как показано на рисунке 6.7. Далее еще раз «ok».

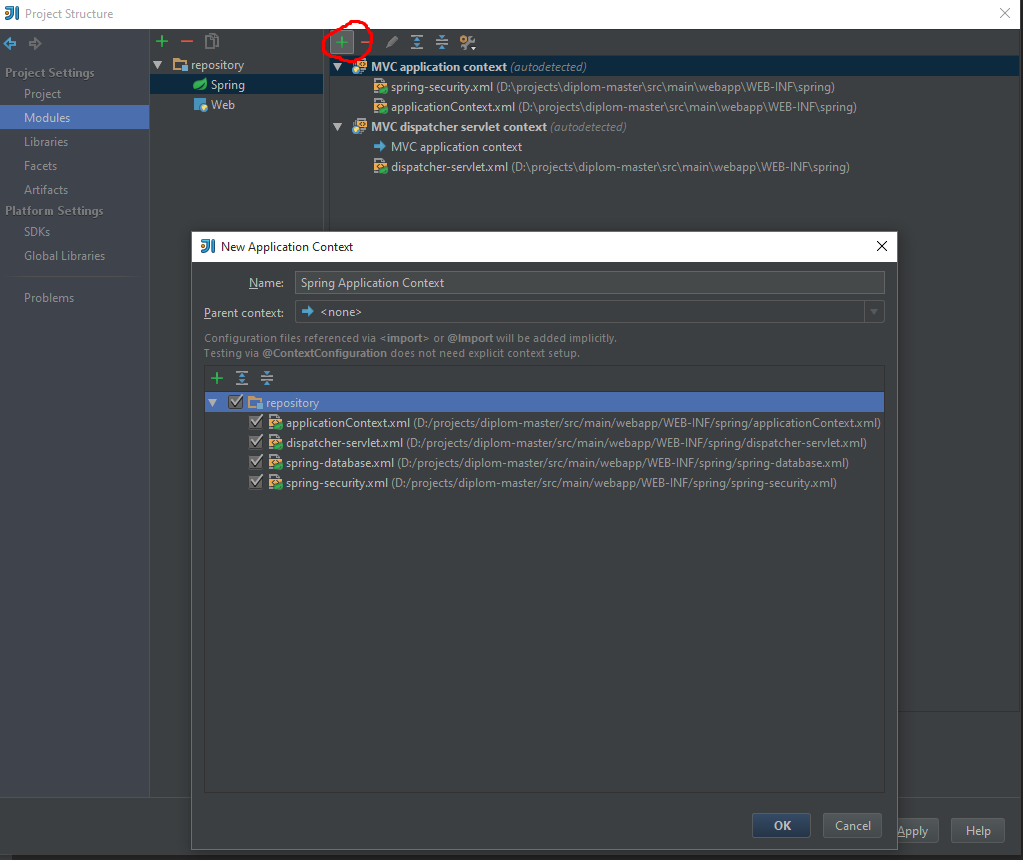


Рисунок 6.7 – Выбор файлов для модуля «Spring»

В результате у модуля Spring должна быть конфигурация, представленная на рисунке 6.8.

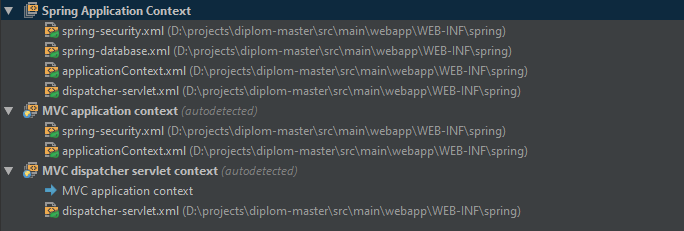


Рисунок 6.8 – Результат конфигурации модуля «Spring»

* 1. Создание артефакта

Необходимо создать артефакт, который будет деплоится на сервер приложений. Для этого, нажимаем «File» -> «Project Structure» -> «Artifacts» -> «+» -> «Web Application: Exploded», как показано на рисунке 6.9.

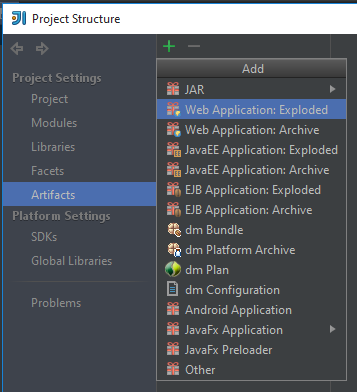


Рисунок 6.9 – Создание артефакта

Вводим любое название в поле Name

Нажимаем на «repository» справа правой кнопкой и, затем, «Put into output root», после чего библиотеки справа должны переместится в артефакт, как показано на рисунке 6.10.

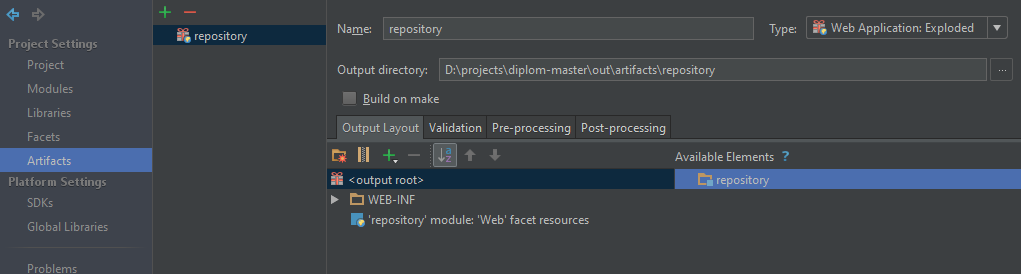


Рисунок 6.10 – Результат добавления библиотек в артефакт

Нажимаем «ok», после чего создается артефакт.

* 1. Создание конфигурации запуска

Далее, для запуска приложения, необходимо создать конфигурацию для запуска и добавить в Idea скачанный ранее сервер приложений WildFly. Для этого, нажимаем «Run» -> «Edit configuration», как показано на рисунке 6.11.

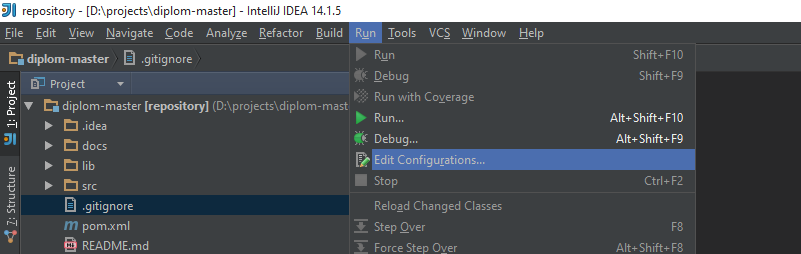


Рисунок 6.11 – Создание новой конфигурации

Далее, «+» -> «JBoss Server» -> «local», нажимаем «Configure», как показано на рисунке 6.12.

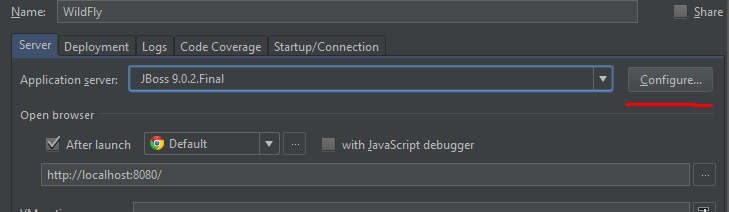


Рисунок 6.12 – Настройка сервера приложений

В появившемся окне нажимаем «+», выбираем путь до папки с сервером, в который распаковывали его в пункте 3 и нажимаем «ok».

В списке сервером должен появится этот сервер приложений. Выбираем его и нажимаем «ok».

Далее продолжаем заполнять настройки конфигурации для запуска. В поле Name можно ввести любое название. Далее переходим во вкладку «Deployment», внешний вид которой представлен на рисунке 6.13.

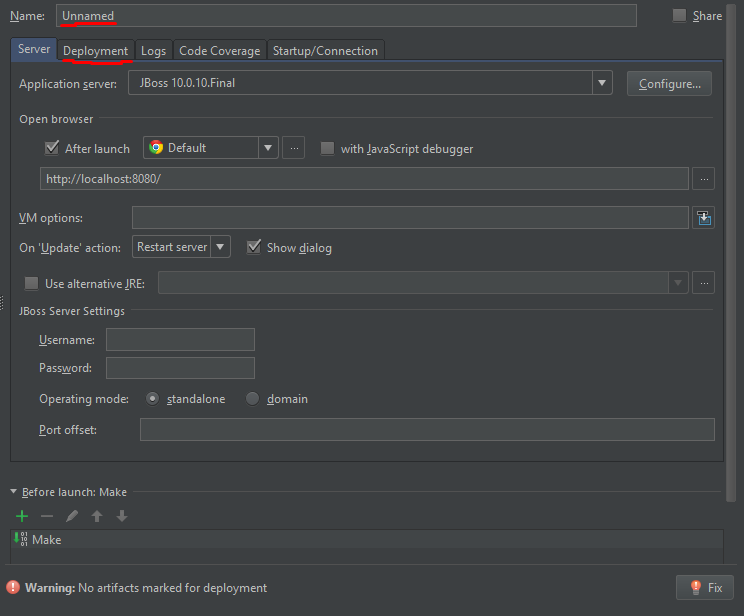


Рисунок 6.13 – Вкладка «Deployment»

Нажимаем на «+» справа -> «Artifact…» как показано на рисунке 6.14.

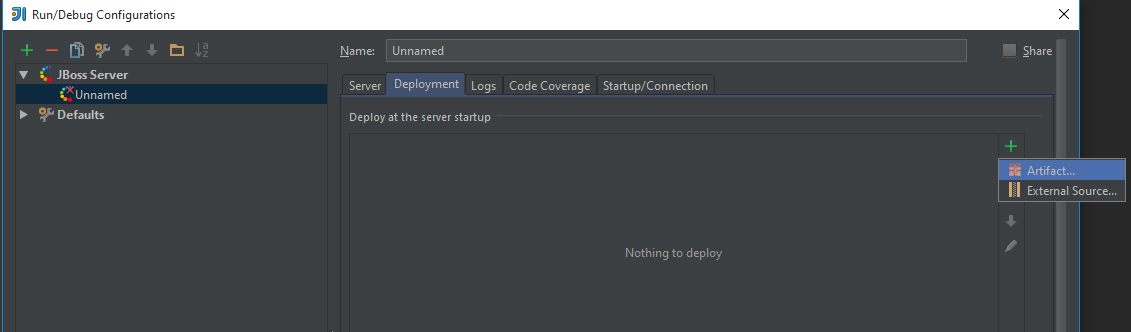


Рисунок 6.14 – Добавление артефакта в конфигурацию

После этого должен появится созданный нами в пункте 4 артефакт. Нажимаем «ok», после чего конфигурация будет создана.

* 1. Добавление библиотеки

Необходимо добавить 1 библиотеку вручную. Нажимаем «File» -> «Project Structure» -> «Libraries» -> «+» -> «Java» как показано на рисунке 6.15.

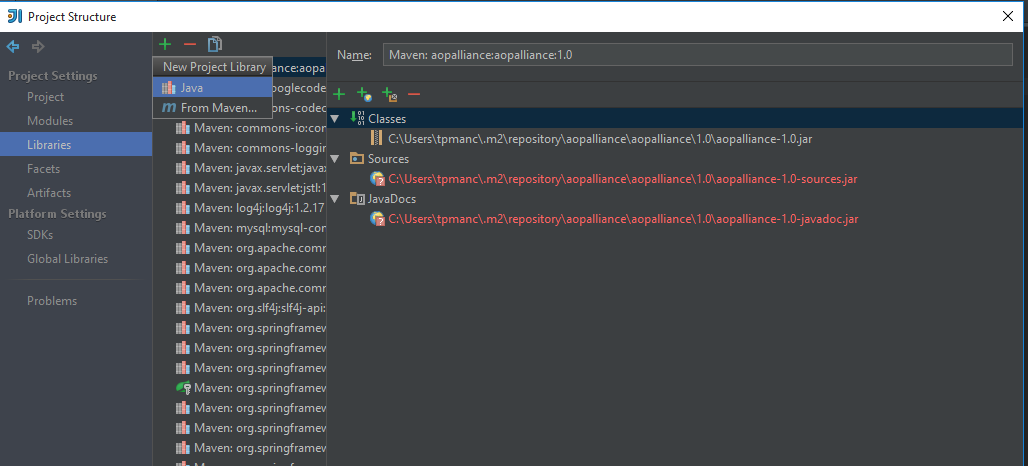


Рисунок 6.15 – Добавление библиотеки в поект

В появившемся окне выбираем путь до папки с проектом, файл lib/pecoff4j-0.0.2.jar и нажимаем «ok».

После добавления библиотеки ее необходимо добавить в артефакт. Также, любую другую библиотеку, которую мы добавили в проект, необходимо добавлять в артефакт.

«File» -> «Project Structure» -> «Artifacts» и двойным кликом по библиотеке перемещаем ее влево. Результат перемещения представлен на рисунке 6.16.

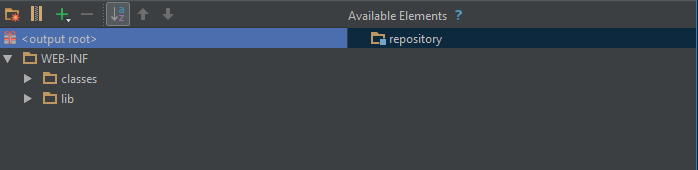


Рисунок 6.16 – Результат перемещения библиотеки

* 1. Запуск

Теперь можно запустить приложение. Справа сверху выбираем название нужной конфигурации и жмем пуск.

Idea сама запустит сервер приложений и задеплоит артефакт на него. При успешном запуске в консоли выведется текст, представленный на рисунке 6.17.

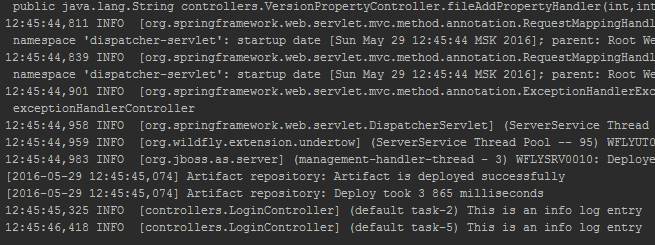


Рисунок 6.17 – Информация об успешном запуске приложения

1. Структура проекта

Структура приложения представлена на рисунке 7.1.

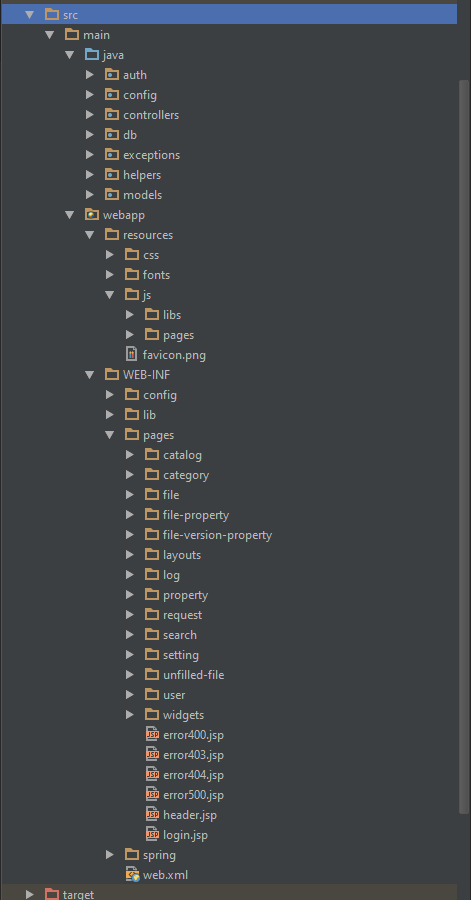


Рисунок 7.1 – Структура проекта

Весь код лежит в папке src/main/java.

* auth – классы для работы с Active Directory
* controllers - контроллеры
* models - модели
* helpers – вспомогательные классы

Виды лежат в src/webapp/WEB-INF/pages/название модели/.

При первом запуске в домашней папке пользователя создается папка repository, в которой создаются файлы конфигурации БД, Active Directory и логирования.

* 1. Контроллеры

Контроллеры должны помечаться аннотацией @Controller.

Для того, чтобы привязать метод контроллера к какому-либо URL, нужно добавить аннотацию, например,

@RequestMapping(value = {"/catalog", "/"}, method =RequestMethod.GET)

Каждый метод контроллера должен возвращать строку, в которой будет путь до нужного файла jsp вида относительно папки src/main/webapp/WEB-INF/pages. Например, метод возвращает строку "catalog/catalog". Тогда будет использоваться файл src/main/webapp/WEB-INF/pages catalog/catalog.jsp.

Для того, чтобы передавать данные в вид, нужно в список входных параметров метода контроллера добавить Model model.

public String users(  
 @RequestParam(value="categoryId", required=false, defaultValue = "0") int categoryId,  
 @RequestParam(value="page", required=false, defaultValue = "1") int page,  
 Principal principal,  
 Model model  
)

Теперь, для передачи данных в шаблон пишем

model.addAttribute("requestCount", requestCount);

Использовать это в шаблоне можно так:

<h2>**${**requestCount**}**</h2>

* 1. Модели

Все модели реализуют интерфейс ModelInterface. Необходимо реализовать методы: add(), validate(), delete(), update().

Для работы в БД пишем:

NamedParameterJdbcTemplate template = new NamedParameterJdbcTemplate(Database2.*getInstance*().getBds());  
MapSqlParameterSource parameters = new MapSqlParameterSource();  
parameters.addValue("id", id);  
List<Map<String, Object>> rows = template.queryForList(*getById*, parameters);  
for (Map row : rows) {  
 Integer fileId = (Integer) row.get("id");  
 String title = (String) row.get("title");  
 return new FileModel(fileId, title);  
}

Здесь создается шаблон и набор параметров, которые мы хотим подставить в SQL запрос.

Диаграмма классов представлена на рисунке 7.2.

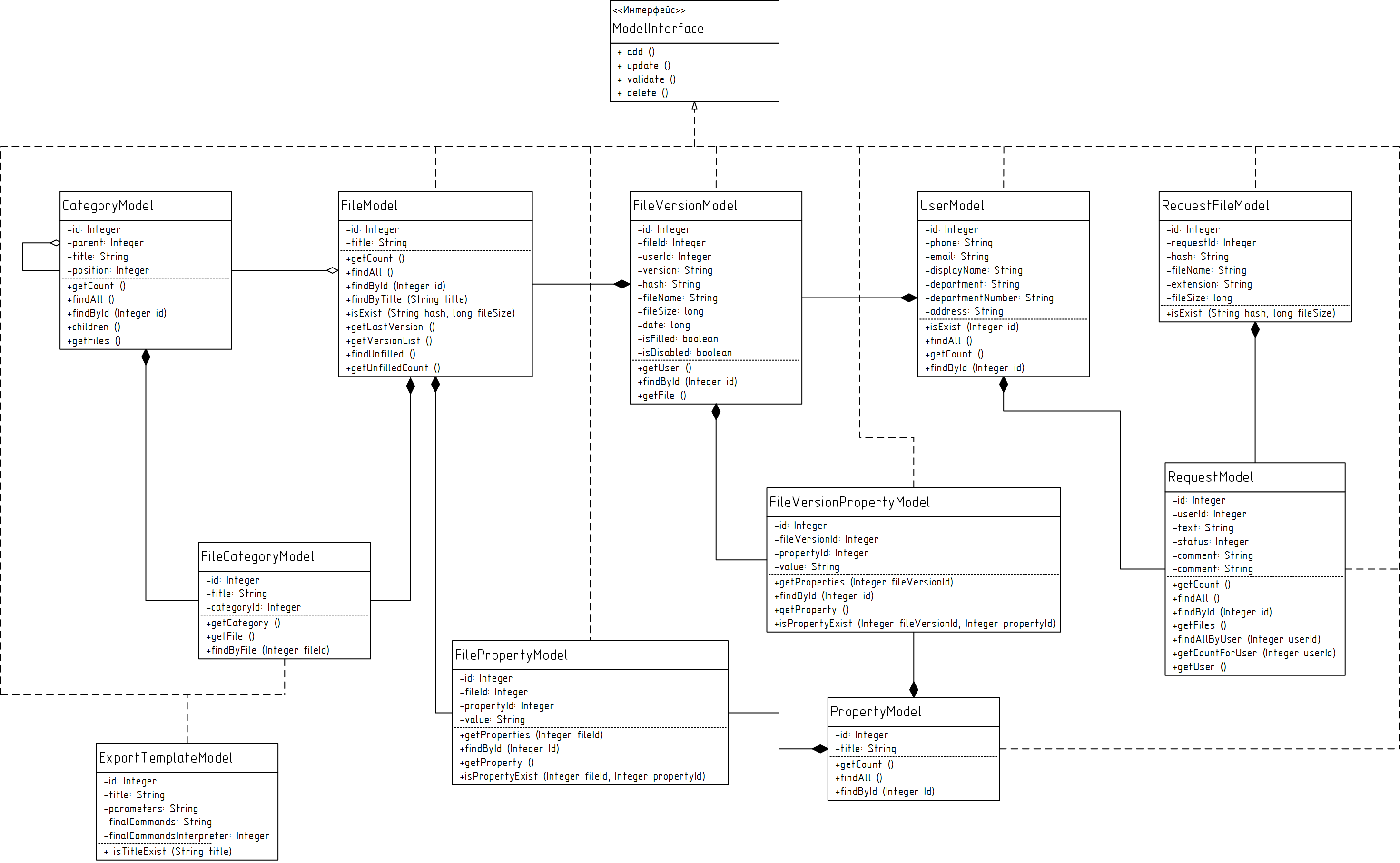


Рисунок 7.2 – Диаграмма классов

1. Сборка

В результате сборки нам нужно получить war архив. Для этого, надо создать еще один артефакт, как показано на рисунке 8.1: «File» -> «Project Structure» -> «Artifacts» -> «+» -> «Web Application: Archive»

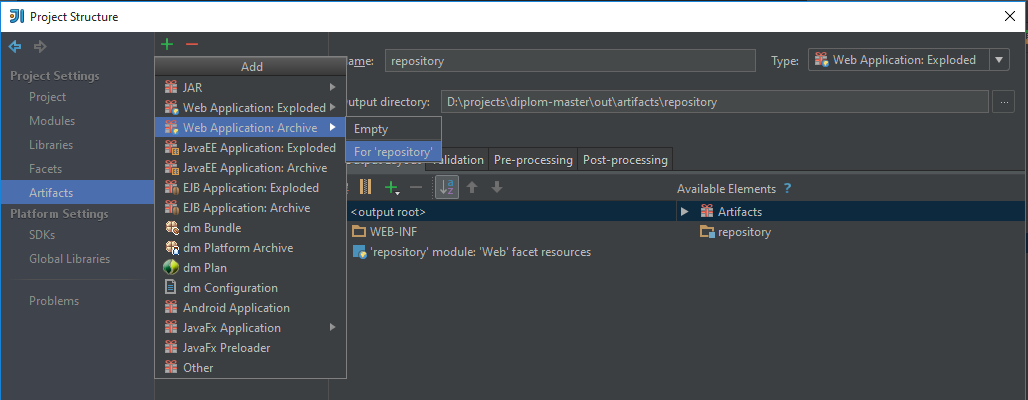


Рисунок 8.1 – Создание артефакта

Обязательно выбираем For «repository». Вместо repository может быть другое название, если предыдущий артефакт был назван по-другому. Нажимаем «For «repository»», создается артефакт и нажимаем «ok», чтобы сохранить его.

Для того, чтобы собрать приложение, необходимо нажать кнопку сверху «Build» -> «Build Artifacts», как показано на рисунке 8.2.

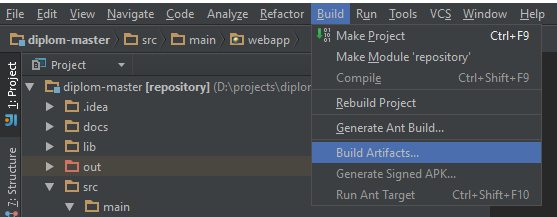


Рисунок 8.2 – Сборка приложения

В появившемся окошке выбираем только что созданный артефакт и Build, как показано на рисунке 8.3.

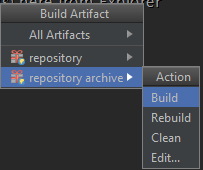


Рисунок 8.3 – Выбор артефакта для сборки

Результат сборки написан в самом низу и представлен на рисунке 8.4:



Рисунок 8.4 – Результат сборки приложения

Идем в папку с проектом, out/artifacts/repository\_archive и видим нужный нам war файл, который можно деплоить на сервер приложений.

1. Работа с WildFly
   1. Деплой на сервер приложений

Должен быть запущен сервер приложений. Пример запуска написан в пункте 3. Заходим в панель администратора, адрес которой пишется в консоль, в самом конце, при запуске сервера (рисунок 9.1).

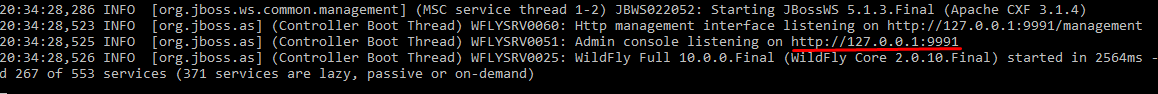


Рисунок 9.1 – Адрес панели администратора

Заходим во вкладку Deployments и видим слева список задеплоенных приложений, которые появились из-за того, что их добавила IntelliJ Idea. Для добавления нового нажимаем «add», выбираем «Upload a new deployment».

Выбираем полученный ранее war файл, как показано на рисунке 9.2.

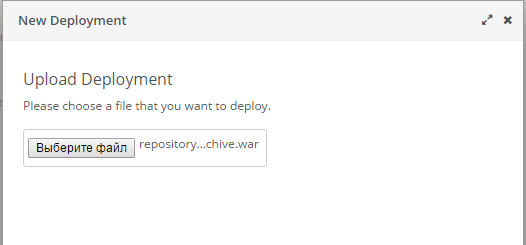


Рисунок 9.2 – Выбор war файла

На следующей форме вводим в поле Name уникальное имя, а в поле Runtime Name адрес, по которому будет открываться приложение, как показано на рисунке 9.3.

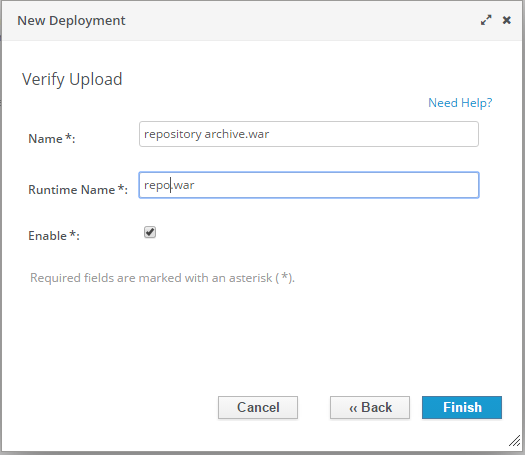


Рисунок 9.3 – Настройка параметров приложения

Нажимаем Finish. Теперь приложение доступно по адресу <http://localhost:8080/repo/>.

* 1. Перезагрузка приложения

Перезагрузка может понадобится, например, когда необходимо поменять настройки в приложении. Для перезагрузки заходим в админку WildFly, переходим на вкладку Deployments. Выбираем нужное приложение и жмем Disable, как показано на рисунке 9.4

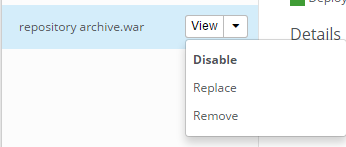


Рисунок 9.4 – Выключение приложения

Теперь включаем, как показано на рисунке 9.5.

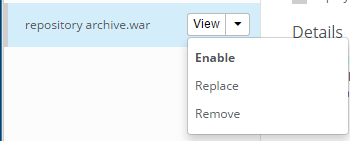


Рисунок 9.5 – Включения приложения

1. Интерфейс пользователя
   1. Настройки

При первом запуске приложения выводится страница для заполнения настроек приложения, внешний вид которой представлен на рисунке 10.1.

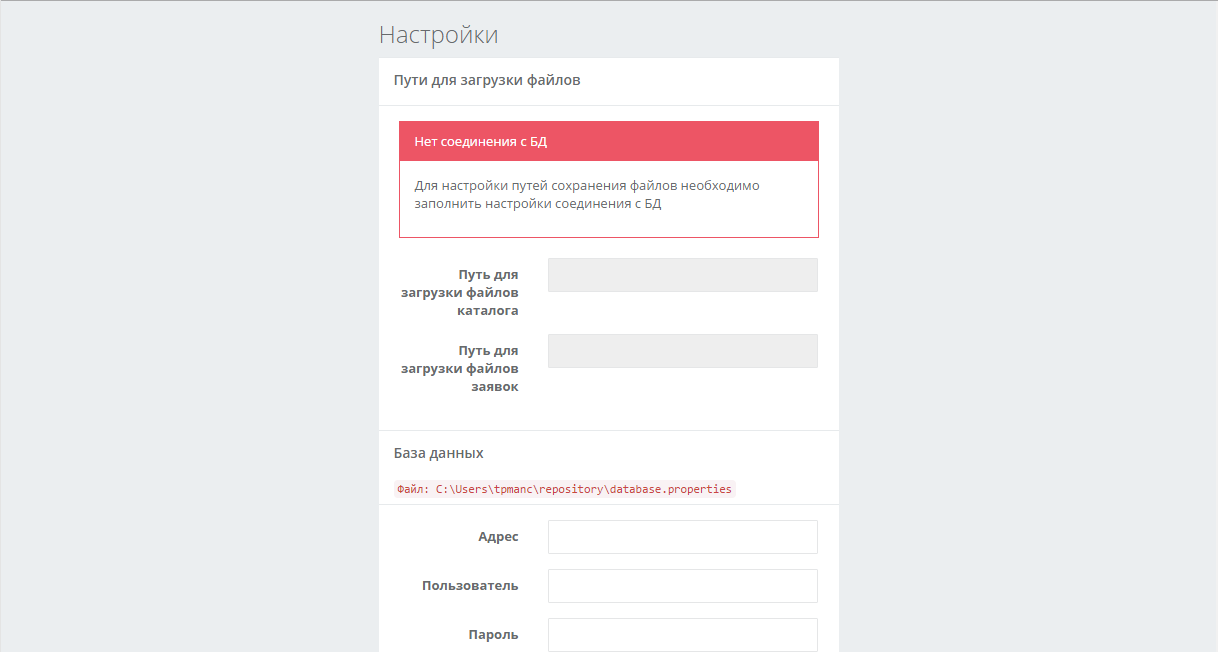


Рисунок 10.1 – Внешний вид страницы настроек

Необходимо заполнить все настройки и перезагрузить приложение.

Теперь, при открытии приложения открывается страница входа, внешний вид которой представлен на рисунке 10.2.

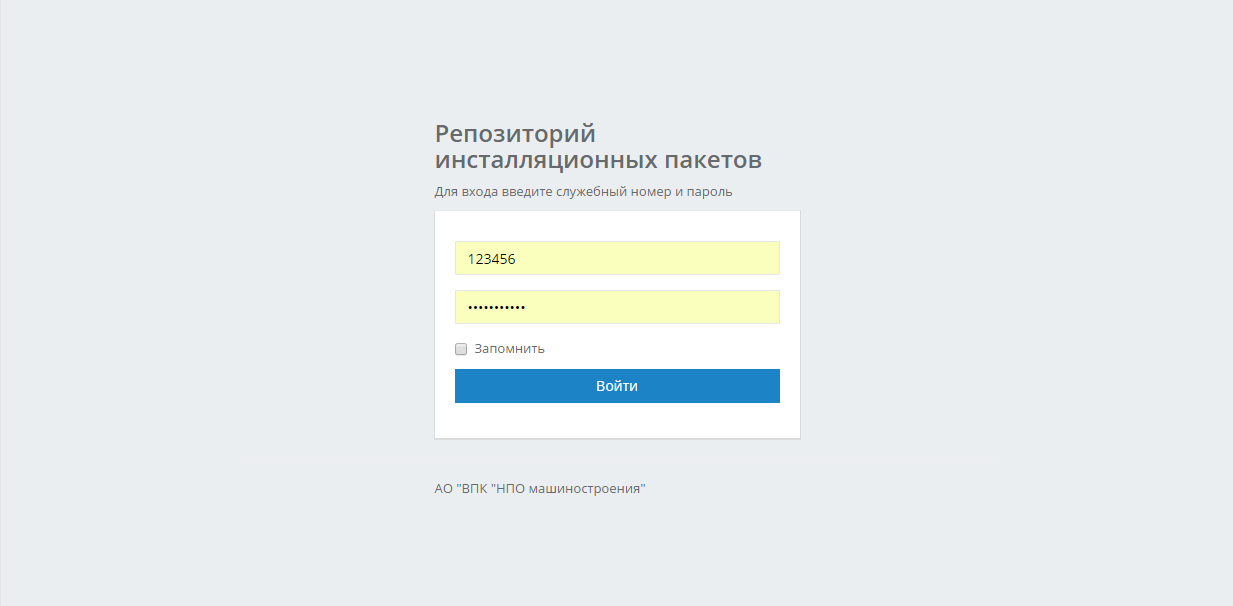


Рисунок 10.2 – Внешний вид страницы входа

Пользователь должен ввести свои данные, после чего нажать кнопку «Войти».

Согласно ТЗ, в системе предусмотрено 3 типа пользователей: модератор, администратор и обычный пользователь.

* 1. Пользовательский интерфейс обычного пользователя

После успешного входа открывается страница каталога инсталляционных пакетов. Ее внешний вид представлен на рисунке 10.3.

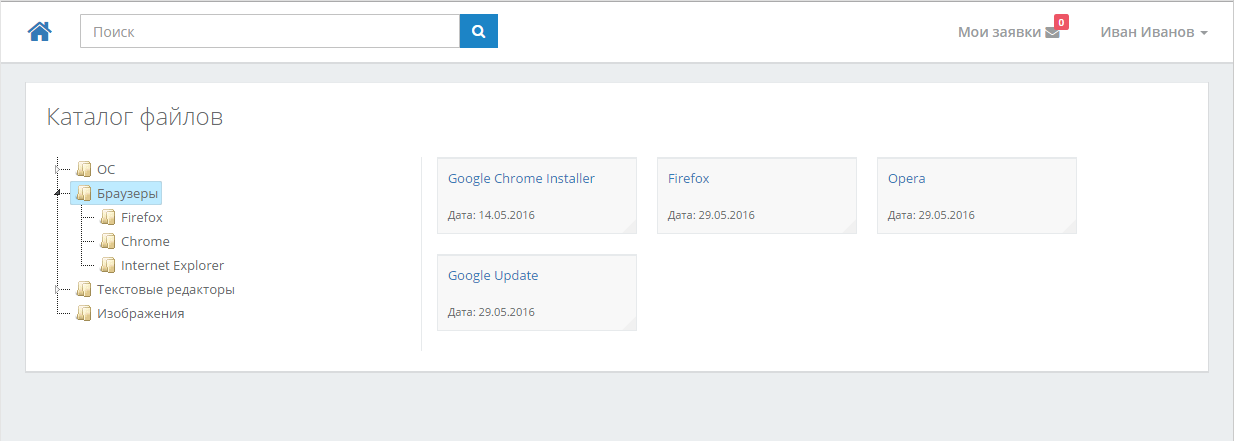


Рисунок 10.3 – Внешний вид страницы каталога инсталляционных пакетов

Пользователь может выбирать необходимые для просмотра категории из дерева категорий слева. При этом, справа будет отображаться список инсталляционных пакетов, которые привязаны к выбранной категории. Для того, чтобы просмотреть информацию об инсталляционном пакете, необходимо нажать на необходимый инсталляционный пакет. После этого откроется страница просмотра информации об инсталляционном пакете, внешний вид которой представлен на рисунке 10.4.

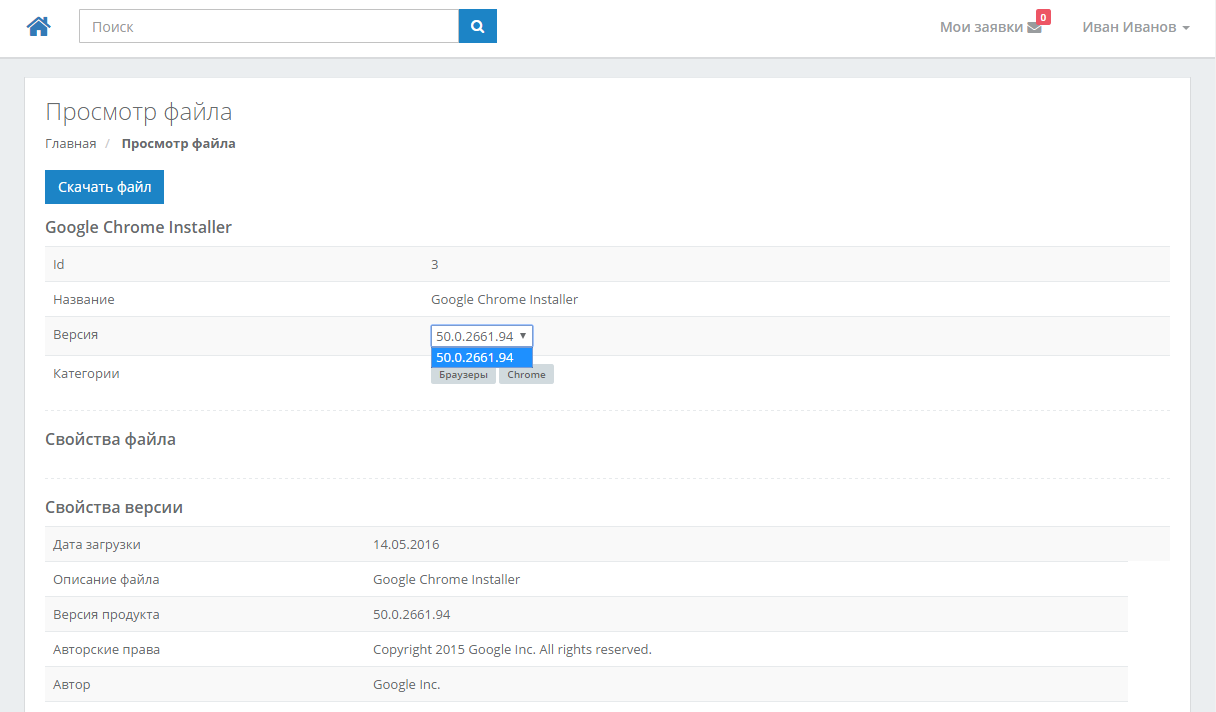


Рисунок 10.4 – Внешний вид страницы просмотра информации об инсталляционном пакете

Для того, чтобы скачать инсталляционный пакет, необходимо нажать кнопку «Скачать файл».

Пользователь может оставлять заявки на добавление ПО. Для того, чтобы просмотреть список своих заявок, необходимо нажать на кнопку «Мои заявки» в верхнем меню. Для того, чтобы добавить заявку, необходимо нажать кнопку «Добавить заявку». Внешний вид страницы добавления заявки представлен на рисунке 10.5.

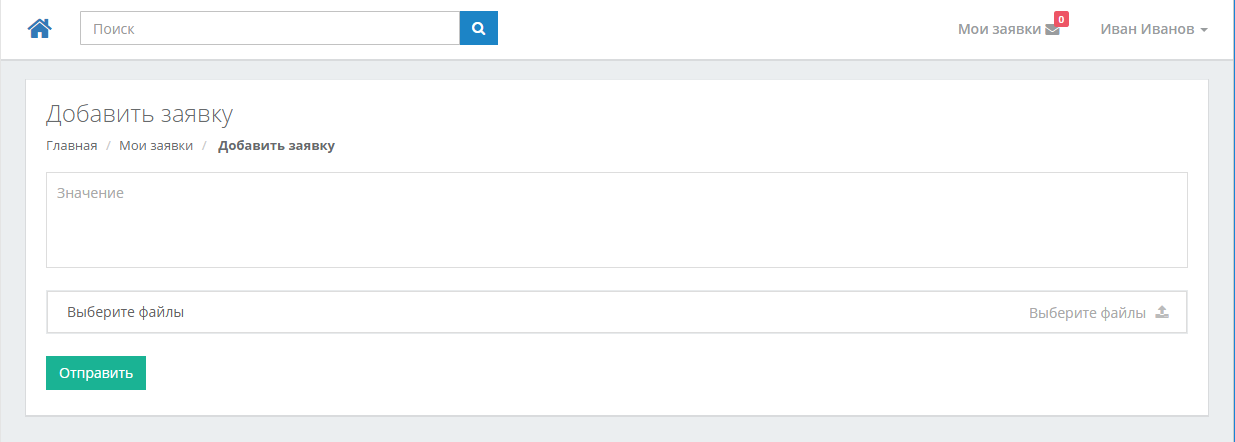


Рисунок 10.5 – Внешний вид страницы добавления заявки

Пользователь должен заполнить комментарий и выбрать файлы для добавления.

Для того, чтобы найти инсталляционный пакет по названию, необходимо ввести текст для поиска в поле «Поиск» верхнего меню и нажать а кнопку поиска.

* 1. Пользовательский интерфейс модератора

После успешного входа модератора открывается страница каталога инсталляционных пакетов. Ее внешний вид представлен на рисунке 10.6.

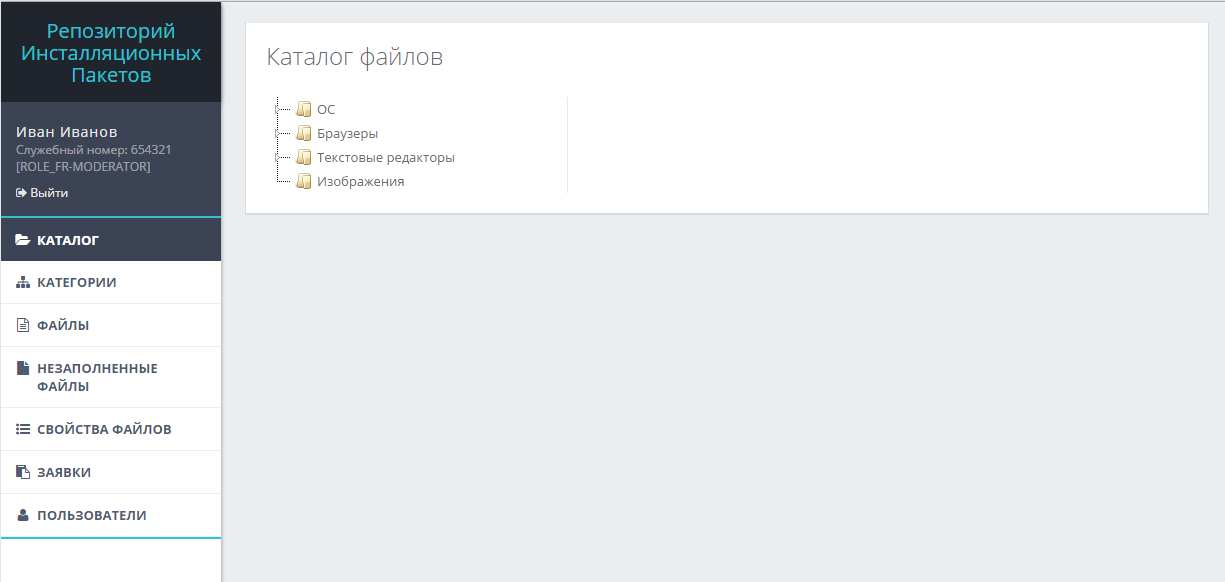


Рисунок 10.6 – Внешний вид каталога инсталляционных пакетов

Для управления категориями необходимо нажать кнопку «Категории» в левом меню. После ее нажатия откроется страница, внешний вид которой представлен на рисунке 10.7.

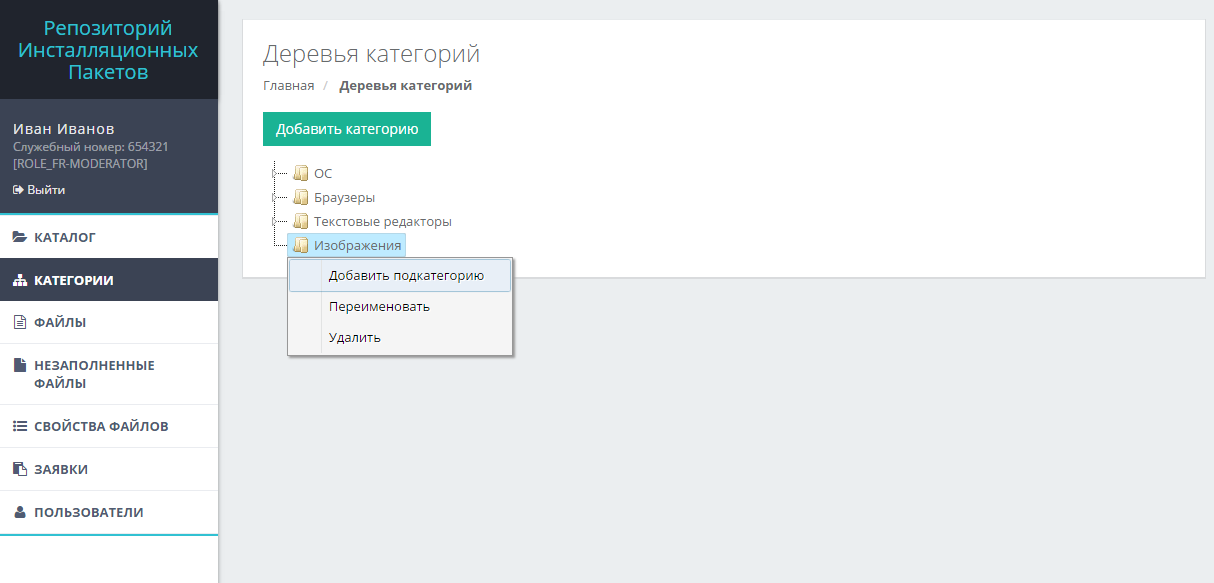


Рисунок 10.7 – Внешний вид страницы категорий

Для добавления категории можно нажать кнопку «Добавить категорию» или нажать правой кнопкой мыши по существующей категории и нажать кнопку «Добавить подкатегорию». Также, при нажатии правой кнопкой по категории, ее можно переименовать или удалить. Категории можно перетаскивать и вкладывать одна в другую, с помощью операции drag and drop.

Для просмотра списка инсталляционных пакетов необходимо нажать кнопку «Файлы» в левом меню, после чего откроется страница со списком существующих инсталляционных пакетов. Для добавления нового пакета необходимо нажать кнопку «Добавить файлы». Для добавления файлов необходимо выбрать файлы и нажать кнопку «Загрузить файлы». Внешний вид страницы добавления файлов представлен на рисунке 10.8.

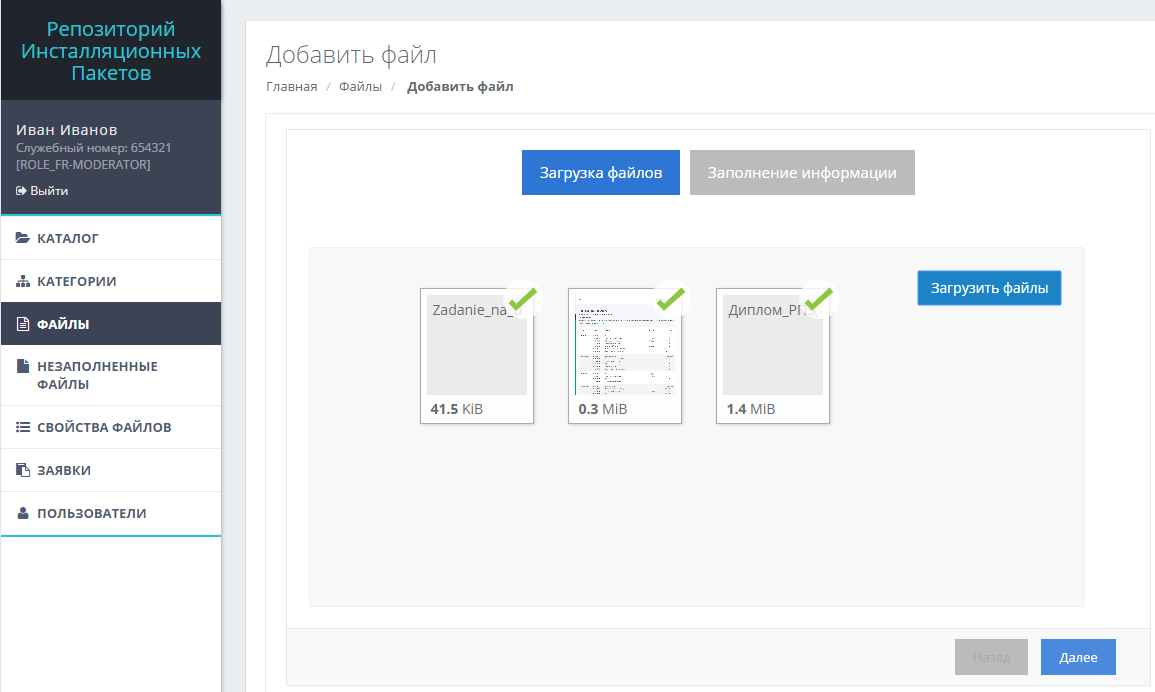


Рисунок 10.8 – Внешний вид страницы добавления файлов

После успешной загрузки, если есть файлы, информацию о которых необходимо заполнить вручную, появляется кнопка «Далее». При нажатии на нее открывается страница с заполнением информации о файлах, где необходимо ввести название и версию, после чего нажать кнопку «Сохранить».

Для просмотра информации об инсталляционном пакете, необходимо на странице просмотра списка файлов нажать на иконку просмотра файла. Внешний вид страницы просмотра информации об инсталляционном пакете для модератора представлен на рисунке 10.9.

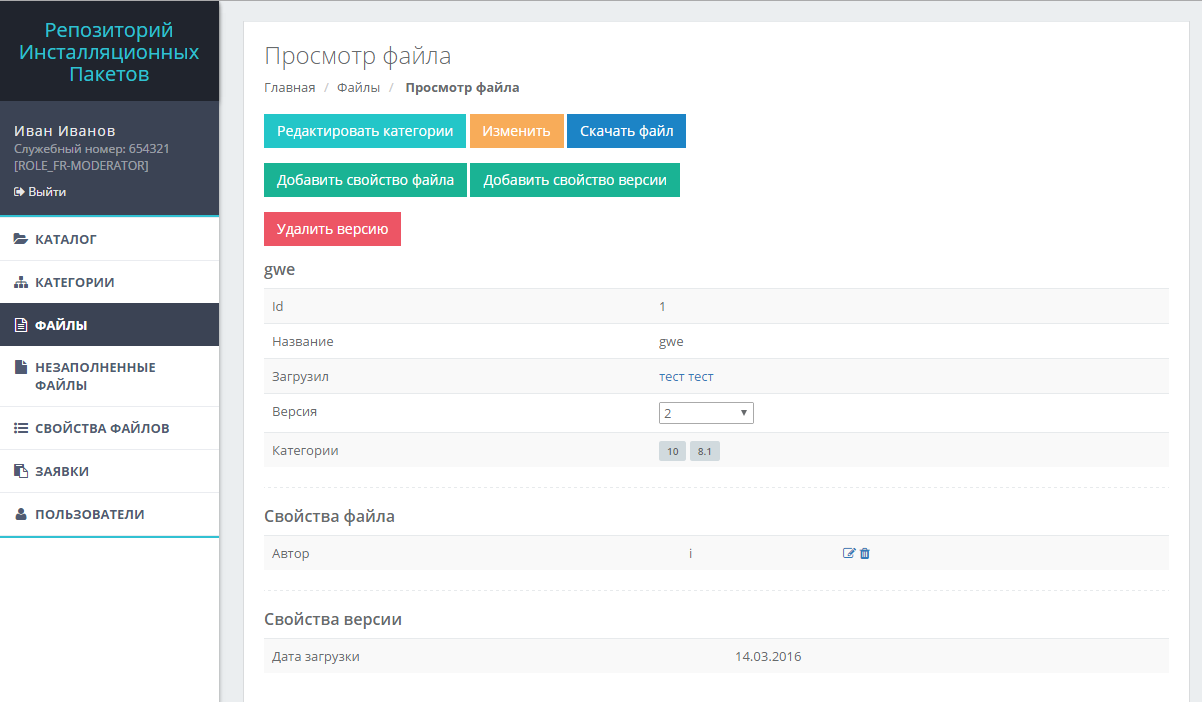


Рисунок 10.8 – Внешний вид страницы добавления файлов

Модератор может изменить список категорий, к которым привязан инсталляционный пакет. Для этого необходимо нажать на кнопку «Редактировать категории», после чего можно выбрать нужные категории из списка всех категорий или удалить уже привязанные категории.

Для изменения названия и версии инсталляционного пакета необходимо нажать кнопку «Изменить», после чего ввести новое название и версию и нажать кнопку «Сохранить».

Для того, чтобы скачать инсталляционный пакет необходимо нажать на кнопку «Скачать файл».

Чтобы добавить новое свойство к версии или ко всем версиям сразу, необходимо нажать кнопки «Добавить свойство версии» или «Добавить свойство файла» соответственно. После этого, нужно выбрать требуемое свойство из списка доступных свойств, ввести его значение и нажать кнопку «Сохранить».

Для того, чтобы удалить версию необходимо нажать кнопку «Удалить версию», после чего версия помечается как удаленная. Для полного удаления версии администратор должен подтвердить окончательное удаление, или восстановить версию.

Для просмотра списка незаполненных файлов необходимо нажать кнопку «Незаполненные файлы» в левом меню. Внешний вид этой страницы представлен на рисунке 10.9.

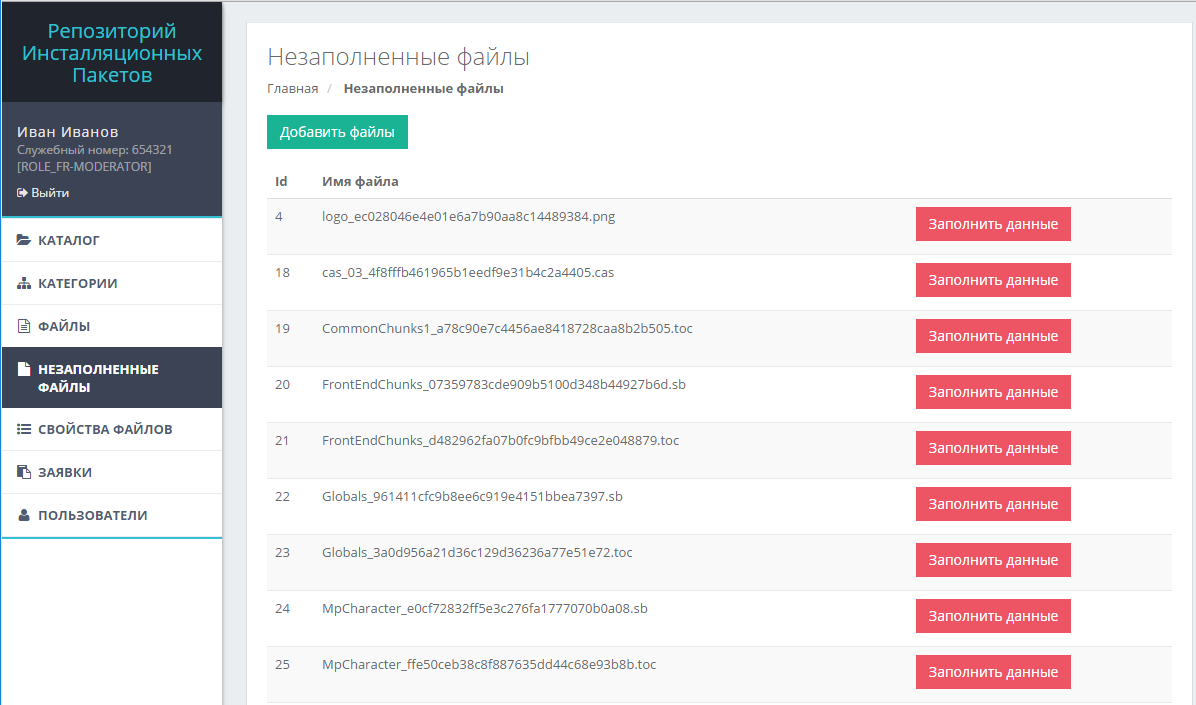


Рисунок 10.9 – Внешний вид страницы незаполненных файлов

На этой странице выводятся файлы, информация о которых еще не заполнена и файлы, которые не привязаны ни к одной категории. Для заполнения информации необходимо нажать кнопку «Заполнить данные» напротив требуемого файла, после чего ввести название, версию и нажать кнопку «Сохранить». Для привязки к категории необходимо нажать кнопку «Привязать к категориям» напротив требуемого файла, после чего выбрать категории из списка доступных категорий и нажать кнопку «Сохранить».

Для просмотра списка свойств, добавленных вручную, необходимо нажать кнопку «Свойства файлов» в левом меню, после чего откроется страница, внешний вид которой представлен на рисунке 10.10.

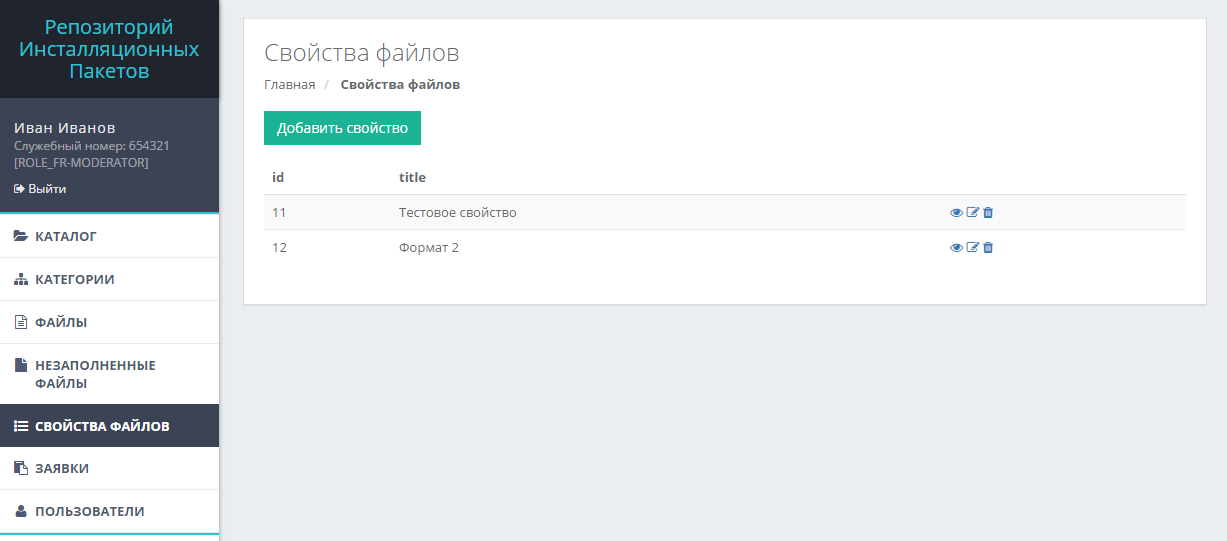
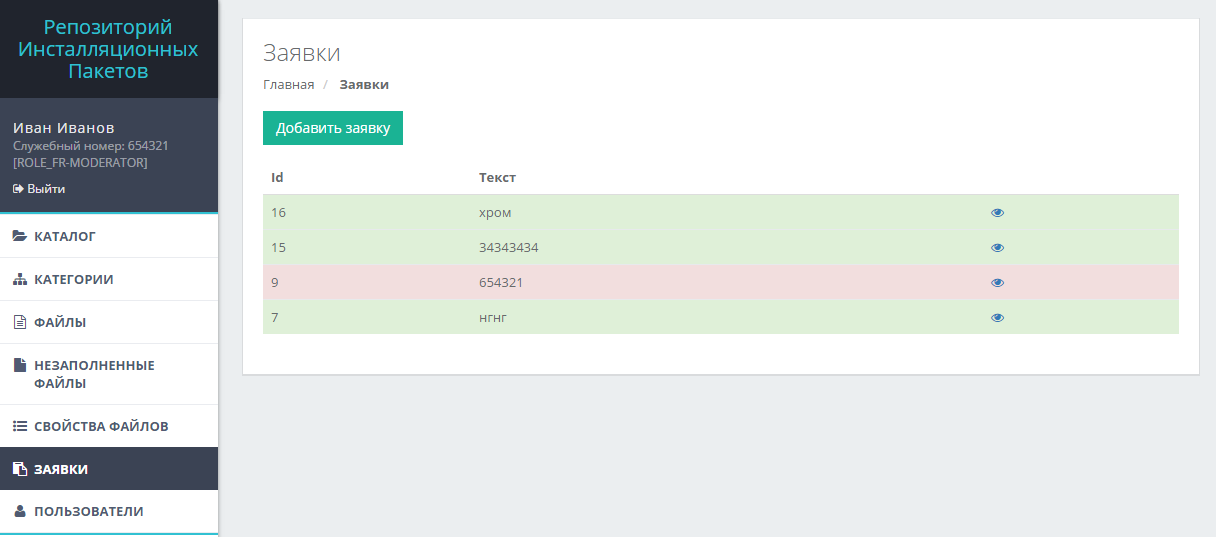


Рисунок 10.10 – Внешний вид страницы свойств файлов

Здесь можно редактировать и удалить существующие свойства, нажав на соответствующую иконку напротив требуемого свойства, или добавить новое свойство, нажав кнопку «Добавить свойство». Для добавления свойства потребуется ввести его название и нажать кнопку «Сохранить».

Для просмотра списка заявок от пользователей необходимо нажать кнопку «Заявки» в левом меню, после чего откроется страница, внешний вид которой представлен на рисунке 10.11.

 Рисунок 10.11 – Внешний вид страницы с заявками от пользователей

Заявки выделяются цветом в соответствии с их статусом: зеленые – принятые, красные – отклоненные и белые – ждущие рассмотрения. Для просмотра заявки нужно нажать на иконку просмотра. На странице просмотра заявки ее можно принять, отклонить или удалить, нажав соответствующую кнопку.

Для просмотра списка пользователей необходимо нажать кнопку «Пользователи» в левом меню, после чего откроется страница, внешний вид которой представлен на рисунке 10.12.

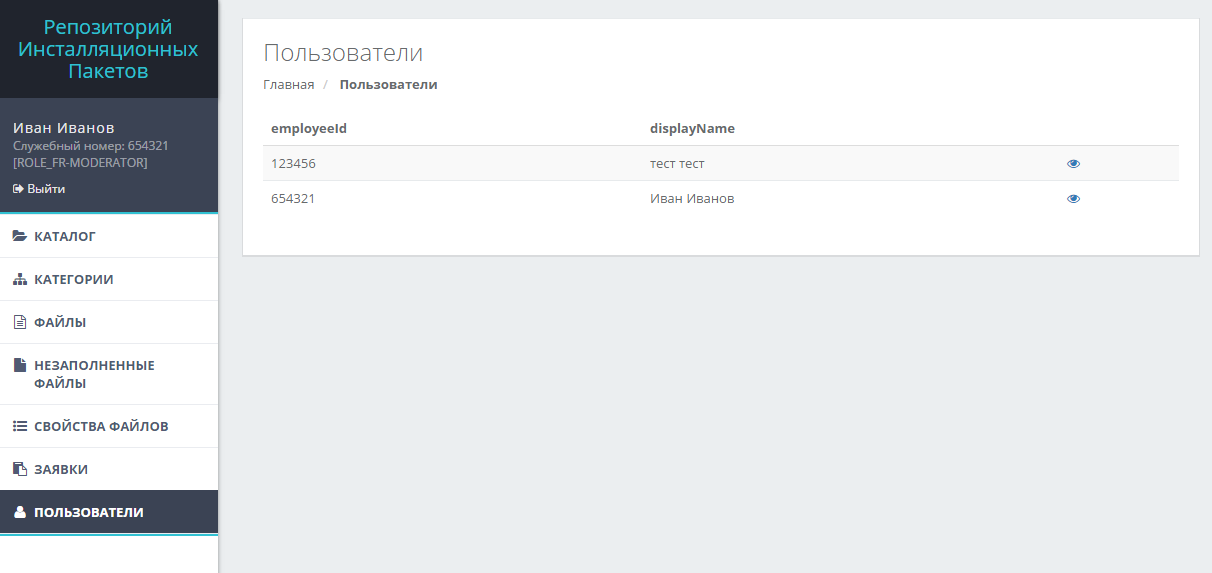


Рисунок 10.12 – Внешний вид страницы пользователей

Для просмотра подробной информации о пользователе нужно нажать на иконку просмотра.

* 1. Пользовательский интерфейс администратора

Страницы каталога, категорий, незаполненных файлов, свойств файлов, заявок и пользователей выглядят аналогично страницам модератора.

Администратор имеет возможность изменять настройки. Для этого необходимо нажать на кнопку «Настройки», после чего откроется страница с настройками, внешний вид которой представлен на рисунке 10.13.

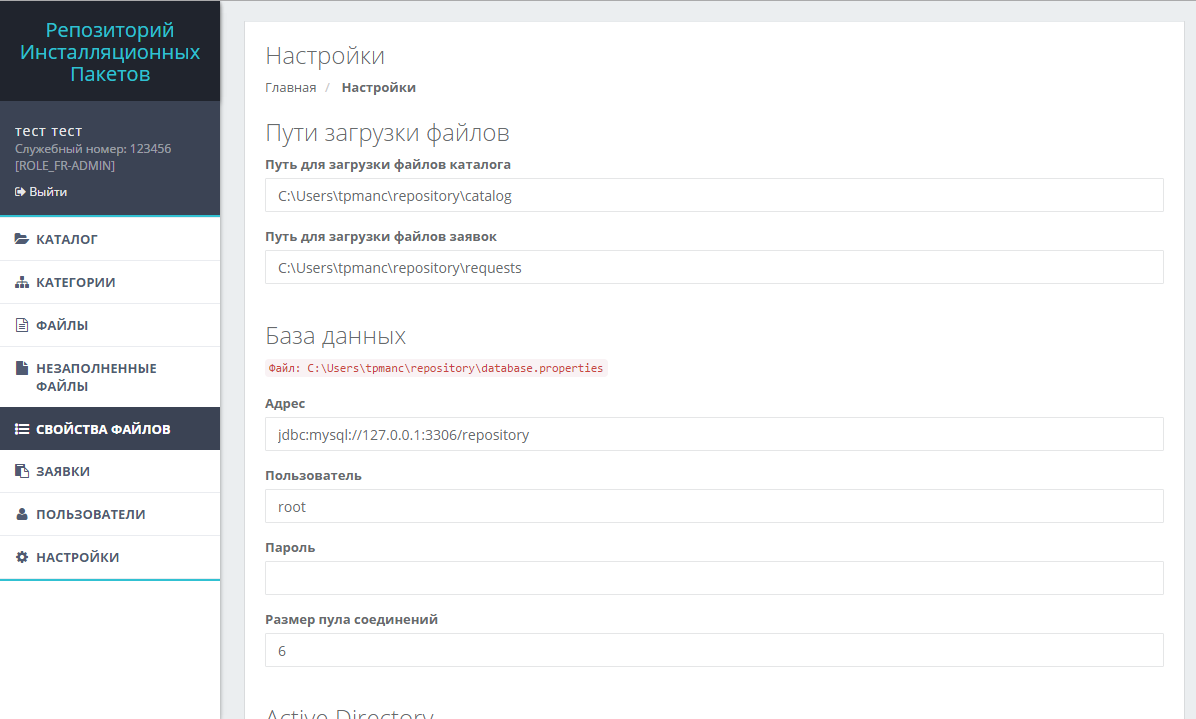


Рисунок 10.13 – Внешний вид страницы настроек

После изменения настроек необходимо нажать кнопку «Сохранить» и перезагрузить приложение.

На странице просмотра информации об инсталляционном пакете администратор имеет дополнительные возможности относительно модератора. Внешний вид страницы просмотра информации об инсталляционном пакете представлена на рисунке 10.14.

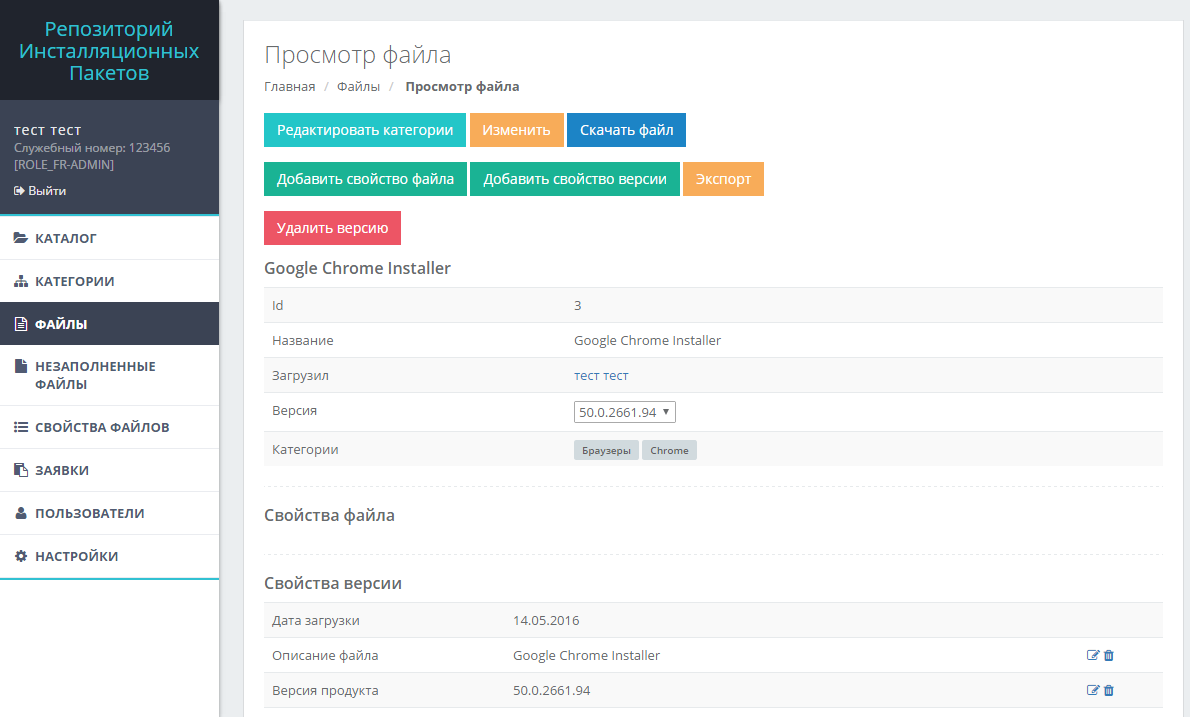


Рисунок 10.14 – Внешний вид страницы просмотра информации об инсталляционном пакете

Администратор может удалить версию, которая помечена как удаленная. У такой версии появляется кнопка «Удалить окончательно» или «Восстановить».

Еще одной возможностью является экспорт файлов. Для экспорта необходимо нажать кнопку «Экспорт». Затем необходимо выбрать один из существующих шаблонов для экспорта или ввести название для нового шаблона. После нажатия кнопки «Далее» откроется страница добавления параметров, внешний вид которой представлен на рисунке 10.15.

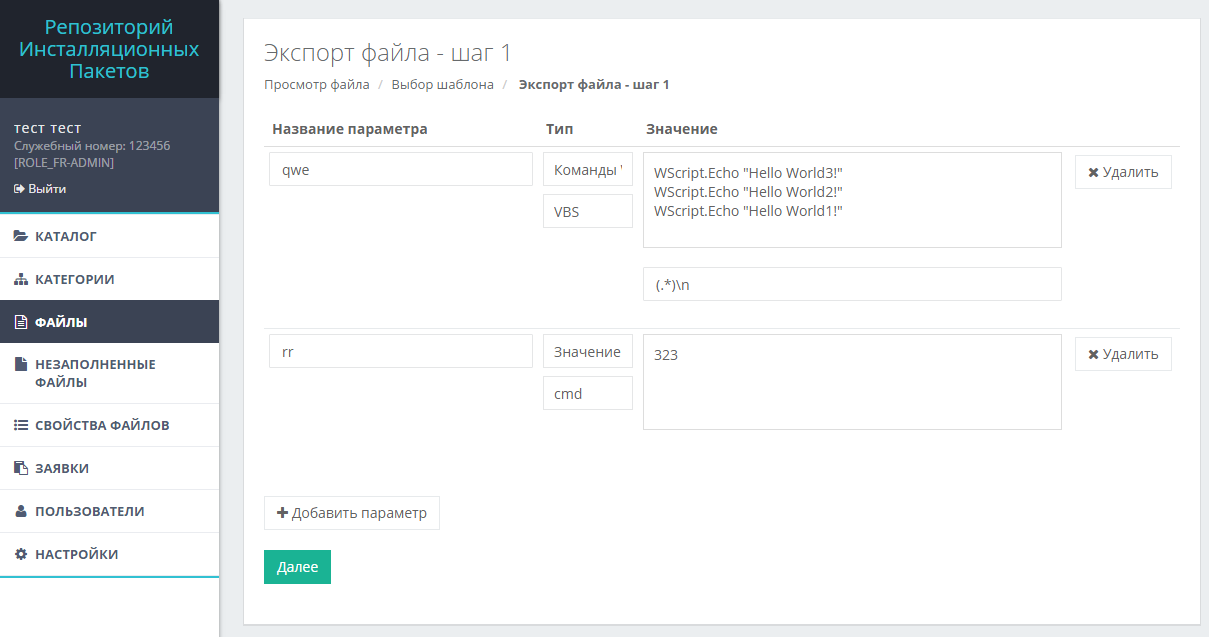


Рисунок 10.14 – Внешний вид страницы добавления параметров для экспорта

После нажатия кнопки «Далее» открывается страница, где для каждого параметра необходимо выбрать одно значение и нажать кнопку «Далее». На последнем этапе необходимо ввести команду для экспорта, выбрать интерпретатор и нажать кнопку «Выполнить».