

Санкт-Петербургский государственный политехнический университет

Институт компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

ОТЧЕТ

о курсовом проекте

по дисциплине: «Программное обеспечение распределенных
вычислительных систем»

Тема работы: «Информационная система строительной компании»

Работу выполнил студент

63501/3 *Богданов Н.Е.*

Преподаватель

_____ *Стручков И.В.*

1. Введение

В рамках курса было необходимо разработать приложение для распределенных вычислительных систем. Приложение должно удовлетворять требованиям открытости, масштабируемость и прозрачности, применять технологии EJB и JPA и использовать Web-интерфейс для взаимодействия с пользователем. Заказ услуг по строительным работам. Создание и управления заказами на строительные работы, а так же учёт требуемых ресурсов на складе.

Разработать информационную систему заказа услуг по строительным работам. Создания и управления заказами на строительные работы, а так же учёт требуемых ресурсов на складе.

Для развертывания в качестве сервера приложений выбран GlassFish Server 4.1

2. Роли

- Клиент

- 1) Заказывает работу.
- 2) Принимает результат.
- 3) Оплачивает работу.

- Менеджер

- 1) Составляет смету + смету доработок (на основе списка доработок от Прораба).
- 2) Ведёт учёт ресурсов со склада.(дополнительно заказывает по мере необходимости).
- 3) Ведёт учёт бюджета компании.
- 4) Принимает оплату от клиента.

- Прораб

- 1) Получает список работ.
- 2) Выполняет работу.
- 3) Составляет список доработок.
- 4) Отдаёт работу на приём Клиенту.

3. Описание бизнес процессов

- 1) Процесс оформления заказа. прописываются все требуемые ресурсы и услуги оказываемые прорабом. Если ресурсов на складе не хватает то происходит дополнительный заказ ресурсов.
- 2) Процесс сдачи/приёма работы. Успешная сдача объекта клиент принимает работу прораба и получает смету (составленную менеджером) со списком проведённых работ. В случае если клиент требует доработки, прораб составляет список требуемых работ и(или) ресурсов), а менеджер составляет смету доработок, после чего клиент оплачивает сметы доработок.
- 3) Процесс оплаты счёта. Клиент оплачивает заказ оплачивая все сметы менеджер подтверждает оплату, заказ закрывается.

4. Варианты использования

4.0.1. Диаграмма вариантов использования

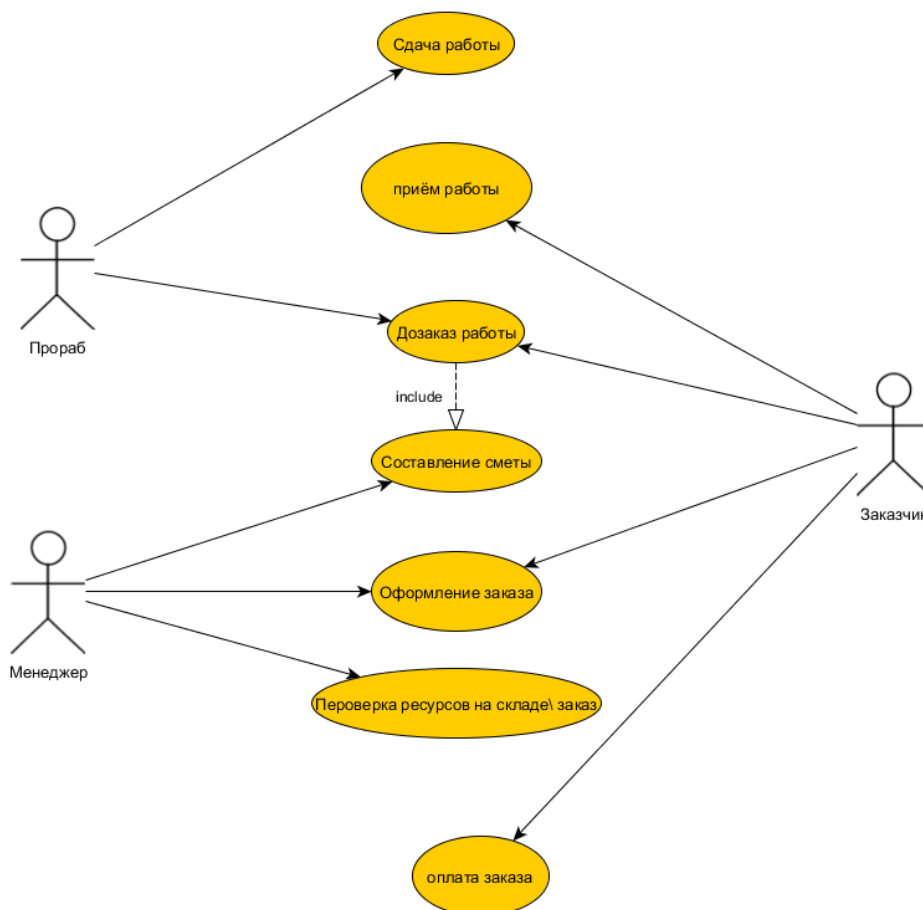


Рис. 1: диаграмма вариантов использования

4.1. Описание вариантов использования

1) Менеджер

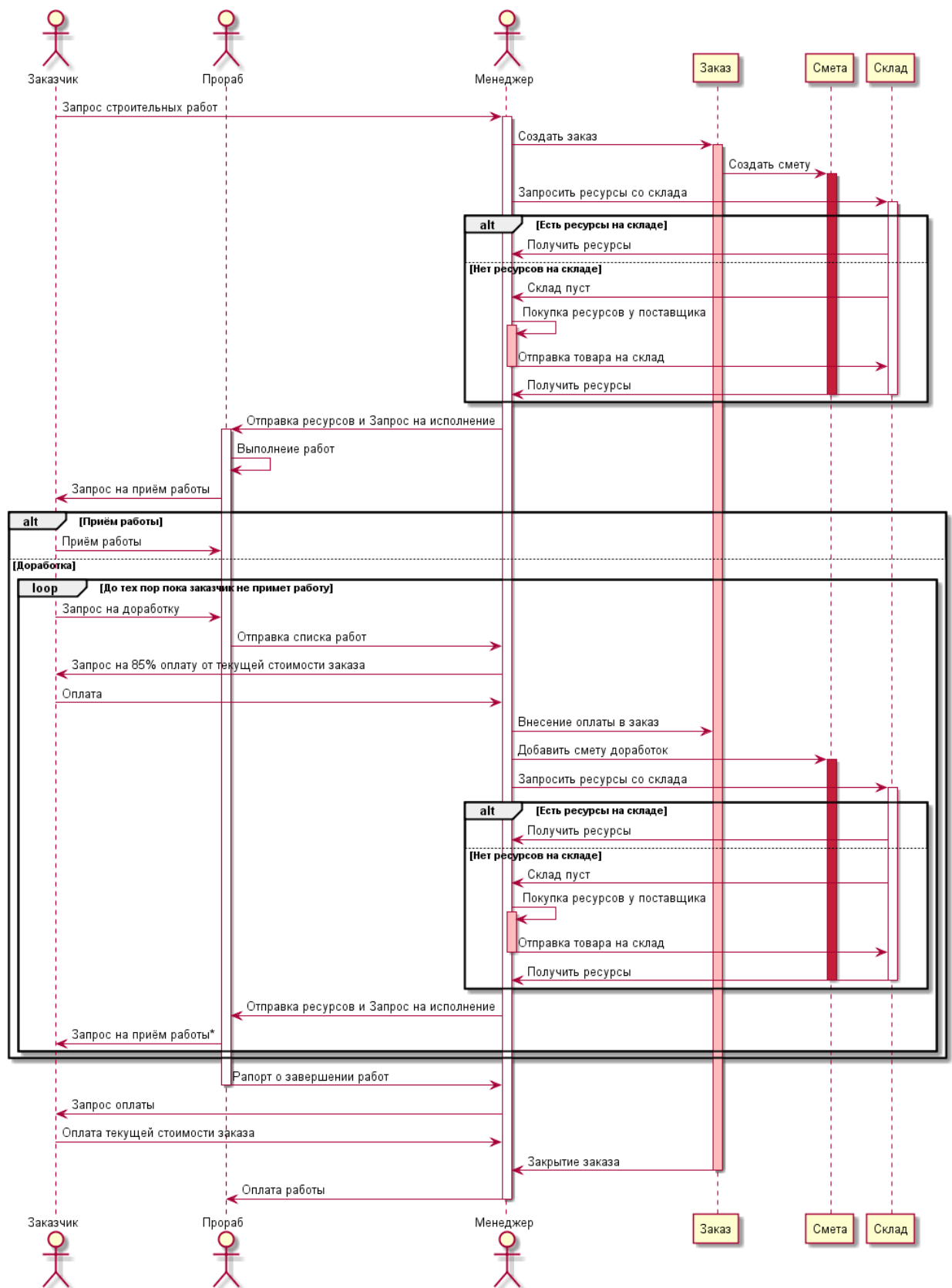
- **Создание заказа.** Менеджер создаёт новый заказ со статусом "открытый" и формирует смету после чего заказ переводится в состояние "в разработке".
- **Создание сметы.** Менеджер прописывает все требуемые ресурсы и услуги оказываемые прорабом, назначает прорабов для выполнения работ и передаётся список работ передаётся каждому из прорабов. Если ресурсов на складе не хватает то менеджер производит дополнительный заказ ресурсов.
- **Заккрытие сметы.** После подтверждения оплаты от клиента смета переводится в статус "закрота".
- **Заккрытие заказа.** После оплаты всех смет заказ переводится в статус закрыт.

2) Клиент

- **Оплата сметы.** Клиент оплачивает смету вводя часть либо полную стоимость сметы, если клиент оплачивает смету полностью то она пишется как оплаченная, если частично то стоимость сметы уменьшается на величину оплаты.
- **Принятие работы выполненной прорабом.** Завершение сметы, клиент принимает работу прораба и если все работы выполнены смета переходит в состояние "завершена" смету .

3) Прораб

- **Завершение работы.** Прораб помечает работу выполненной, данную операцию можно выполнить если статус заказа которому принадлежит работа является "в разработке".



codeuml.com

Рис. 2: диаграмма вариантов использования

4.2. Модель предметной области

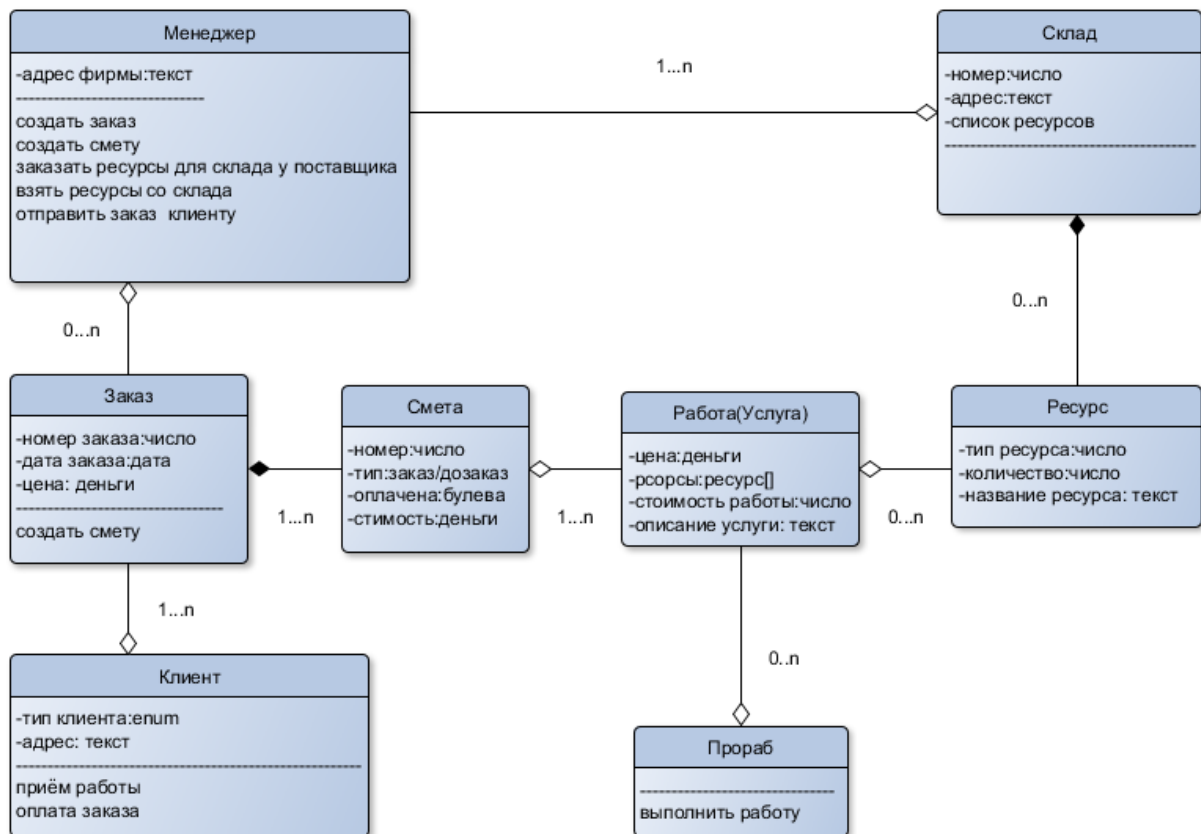


Рис. 3: uml диаграмма классов

4.3. Диаграмма последовательности

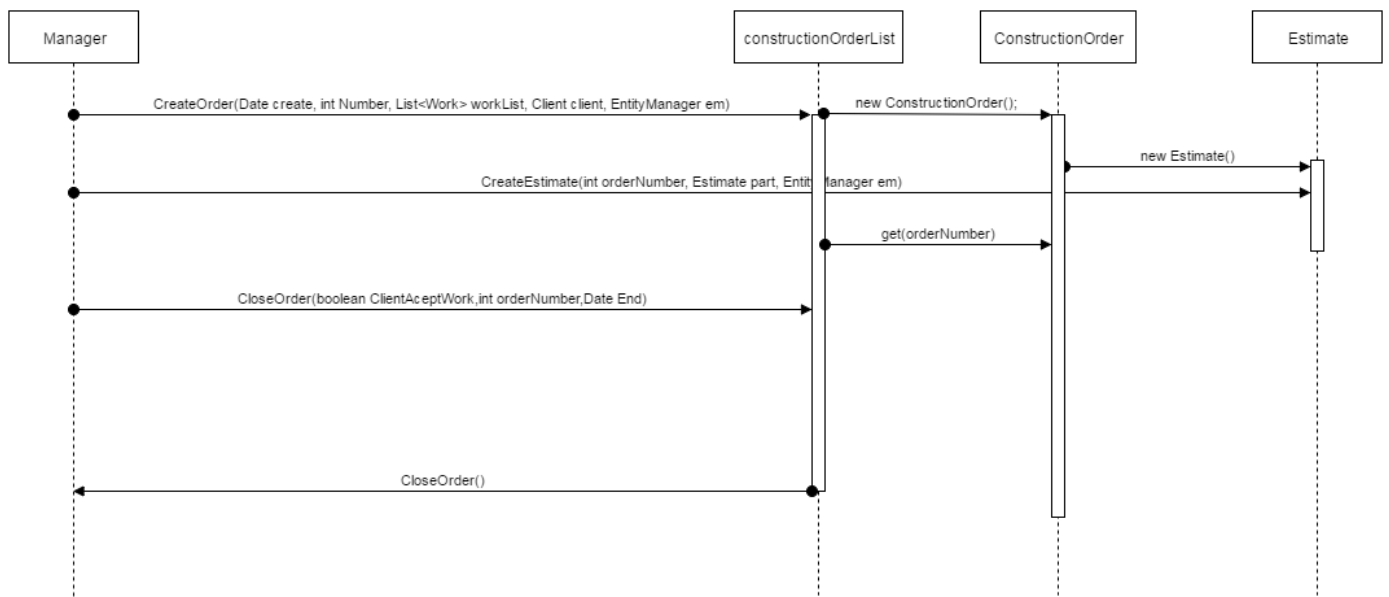


Рис. 4: Диаграмма последовательности менеджера

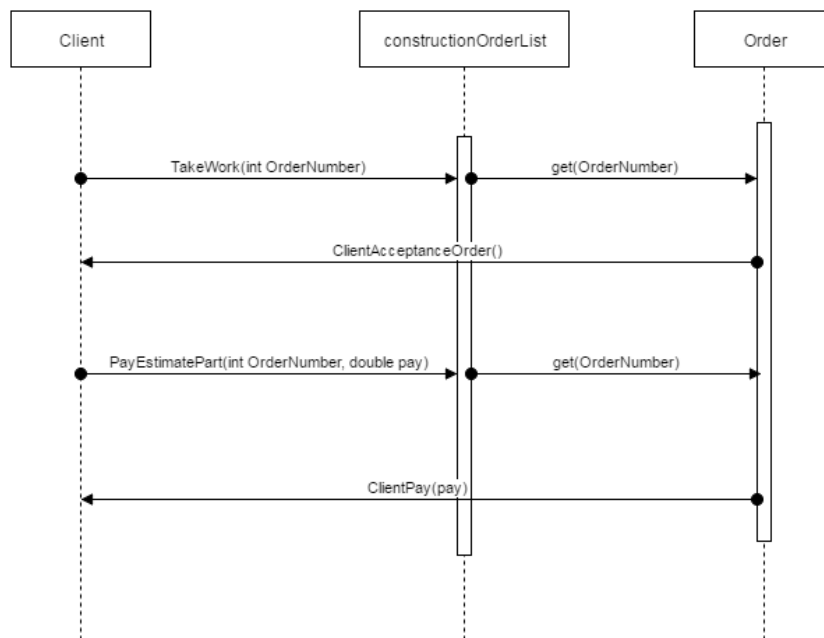


Рис. 5: Диаграмма последовательности заказчика

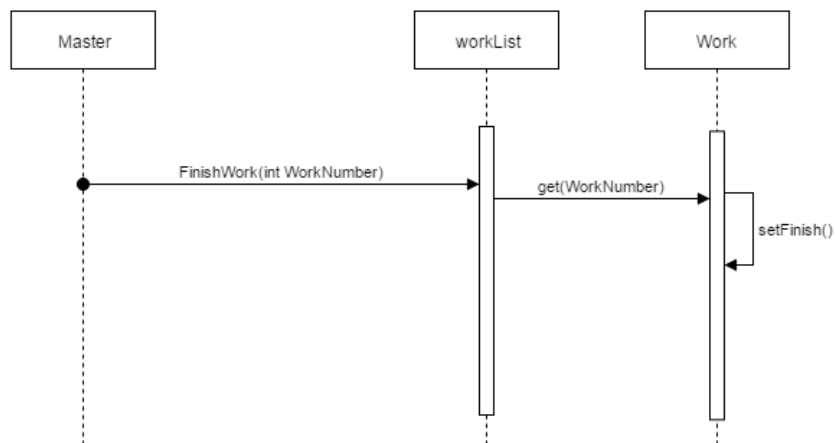


Рис. 6: Диаграмма последовательности прораба

5. Архитектура

5.1. Общая архитектура проекта

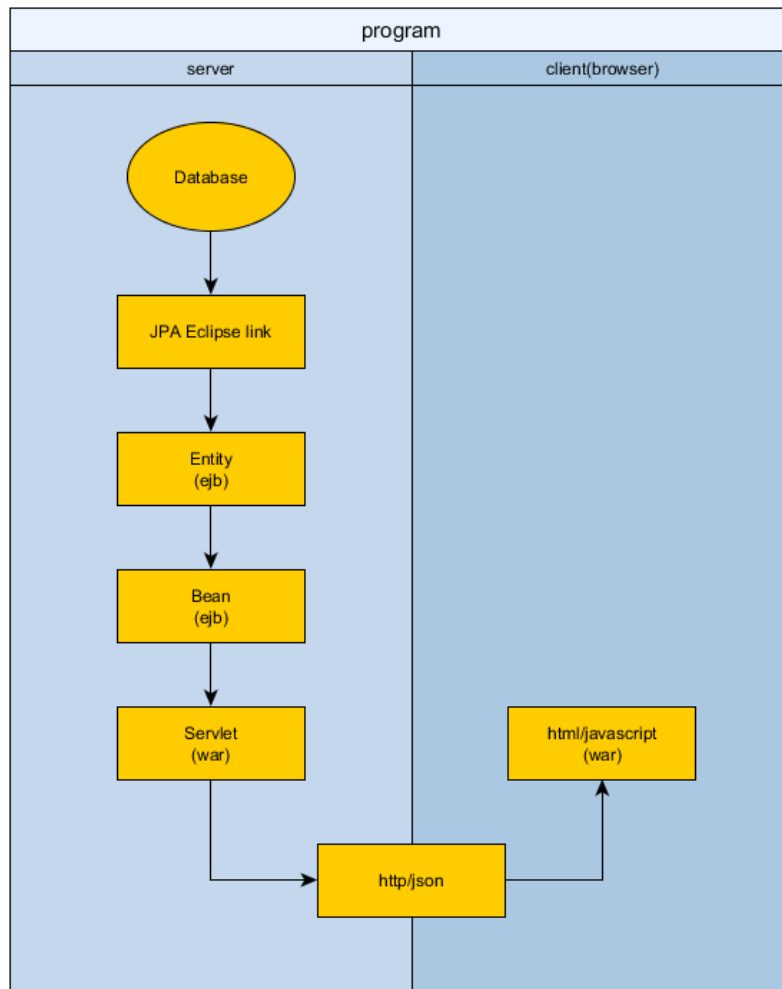


Рис. 7: архитектура клиент-серверного приложения

В качестве типового решения бизнес-логики была выбрана модель предметной области(Domain Model). Классы бизнес логики соответствуют uml диаграмме (рис.2) с дополнительными расширениями. Бизнес логика расположена в разделе Entity (рис.4), взаимодействие с базой данных(слой хранения) осуществлено через ORM - EclipseLink по средством аннотаций. Авторизация пользователей осуществляется через механизм JAAS (Java Authentification and Authorization Service) логика управления в разделе Servlet.

5.2. Сущности, данные

Сущности - объекты с которыми работают роли:

- Order - Заказ
- Estimate - Смета в заказе
- Storage - Склад содержащий список с ресурсами.
- Work - Работы содержащие список ресурсов
- Resource - Ресурсы.
- Manager - Роль менеджера.
- Client - Роль заказчика
- Master - Роль прораба

Представление таблиц в базе данных.

Таблицы:

- StorageList - информация о складе ресурсах на складе.
- Storage - информация о складе.
- Resource - информация о ресурсе его количестве и цене.
- ResourceInformation - информация о ресурсах название и код ресурса.
- Work - информация о работе и ответственном за неё прорабе.
- WorkInformation - информация о стоимости работы(без стоимости ресурсов) и её описание
- WorksAndResource - информация о ресурсах и их количестве нужном для работы.
- Estimate - информация о списке смет для заказов.
- EstimateWorks - информация о списке работ для смет.
- Order - информация о заказе.
- Manager - информация о менеджерах.
- Master - информация о прорабах.
- Client - информация о заказчиках.
- Users - информация о пользователях системы (содержит пароль и логин пользователя).
- Groupusers - сопоставляет пользователя и его роль в системе.

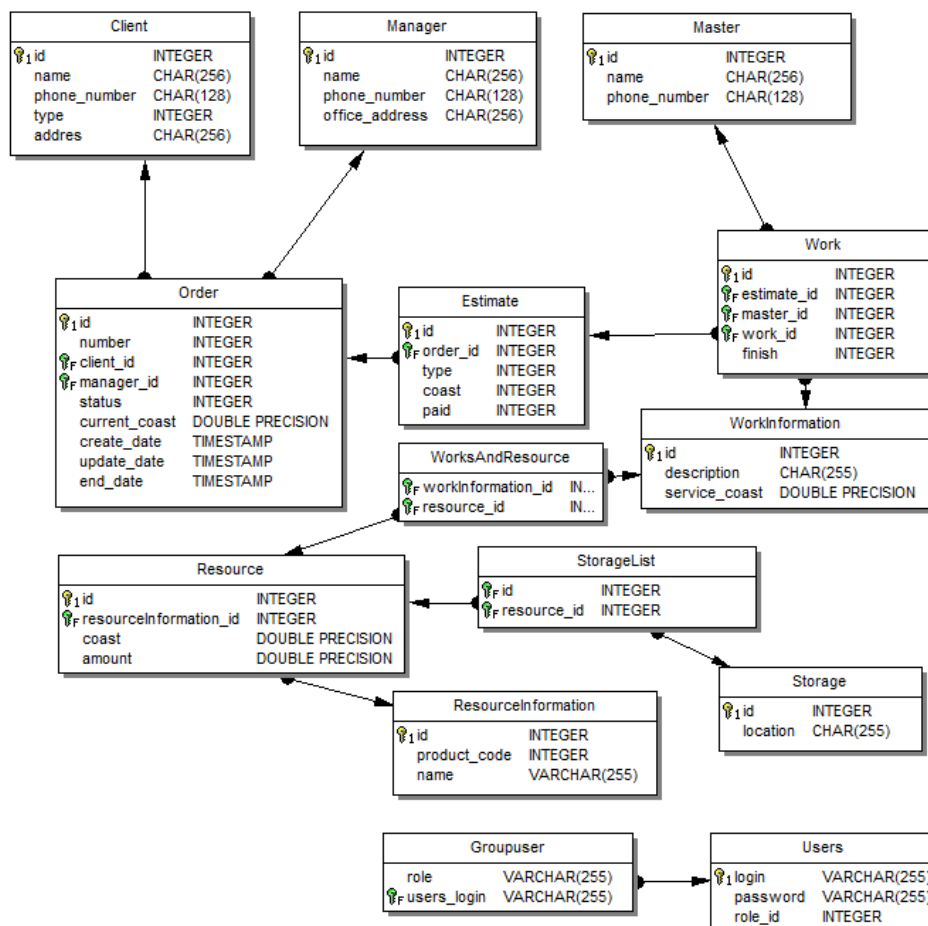


Рис. 8: таблицы в базе данных

5.3. Пакеты и UML диаграммы классов

Проект состоит из пакетов:

- Entity сущности бизнес логики, а так же для работы с ORM через JPA по средством аннотаций.
- Bean составляющих бизнес логику помещённую в EJB-Container.
- API для взаимодействия с web клиентом через REST API(get,put,delete http запросы)
- Sever для указания на какую страницу сайта перейти для конкретной роли.
- Web графическая часть клиента html, css, javascript.

В диаграммах не указана графическая составляющая проекта html страницы с javascript кодом и фреймворком Angular JS, скриншоты из браузера для каждой роли можно увидеть в п.6 Диаграммы были созданы при помощи плагина NetBeanse easyUML.

5.3.1. Entity

Ниже приведена диаграмма из пакета entity отражающего сущности указанные выше в п.3.2 взаимодействие между сущностями совпадает с зависимостями первичных и вторичных ключей между таблицами в базе данных, интерфейс converter нужен для преобразования данных в json формат и обратно методами toJSON - для получения данных в формате json и fromJSON - для преобразования из json в сущность. Класс Users не связан с остальными сущностями т.к. является вспомогательным для определения какой конкретной записи (в Client, Menager, Master) соответствует пользователь авторизовавшийся на сайте.

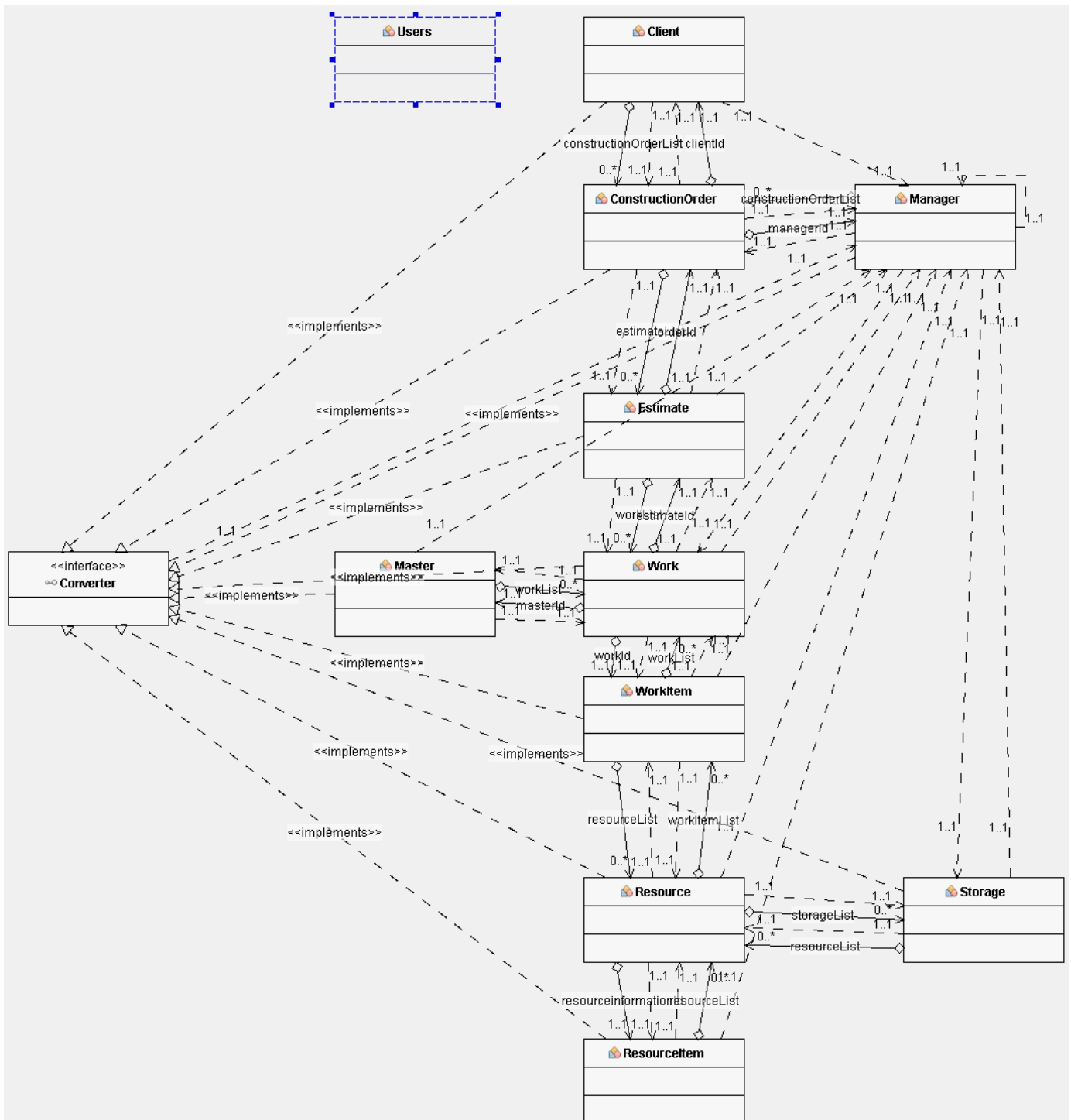


Рис. 9: диаграмма классов из пакета entity

5.3.2. Bean

Было решено использовать 3 @Stateless bean-а соответствующих ролям в проекте, каждый класс отвечает за свою роль, общие методы для всех ролей были выведены в абстрактный класс AbstractRole, а так же обобщенный вывод для всех интерфейсов ролей был указан в BeanRemote методы которые не могут быть реализованы(некоторые методы реализация которых противоречит или нарушает целостность и безопасность логики приложения. Например роль клиента не имеет права вносить правки в стоимость сметы или заказа это может только менеджер или указывать завершение работы за что ответственен прораб) вызывают UnsupportedOperationException с соответствующим пояснением. Некоторые из json объектов не являются сущностями, а командами для выполнения бизнес логики обрабатываются в реализации метода.

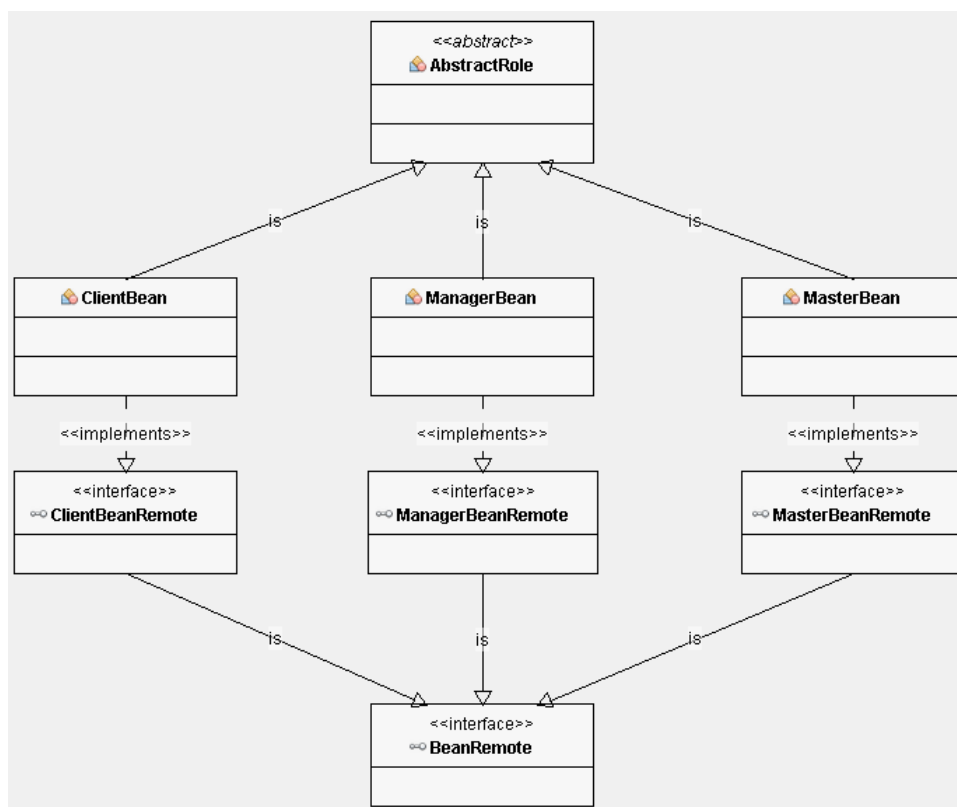


Рис. 10: диаграмма классов из пакета bean

5.3.3. REST API

В пакете api реализация REST API для управления из web клиента по средством get, put и delete http запросов передающих json данные. Для авторизации клиента и подачи ему страницы с соответствующей ролью используется класс AuthenticationService из пакета server

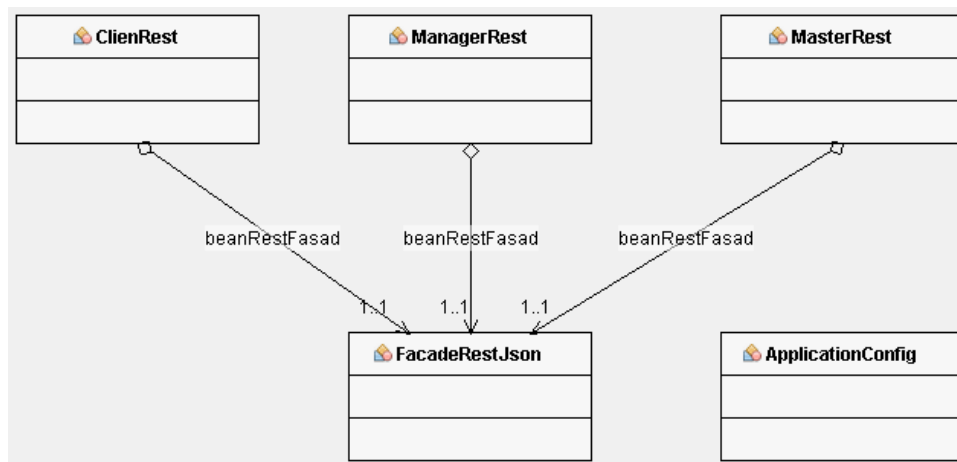


Рис. 11: диаграмма классов из пакета api

REST API

{path} - обозначает url приложения, путь до приложения.

для get запросов:

- **{path}/api/{role}** - для получения общей информации связанной с данным пользователем, role - соответствует роли пользователя, может принимать значения manager,client,master. пример для manager см. рис. 11
- **{path}/api/{role}/profile** - сведения о данном пользователе, ф.и.о. телефон и т.п. см. рис. 10
- **{path}/api/{role}/array/{name}** - для получения массива объектов принадлежащих сущности, где name, это название массива сущности которые могут быть доступны для роли. Для роли master доступен только массив works - список работ для данного прораба, для роли client доступен только массив orders - список заказов данного заказчика. Для менеджера доступны все массивы.
- **{path}/api/{role}/array{name}/{id}** - для получения объекта из массива принадлежащего сущности по id в базе данных.

для put:

- **{path}/api/{role}/array{name}** - для создания новой записи, доступно только для роли manager.

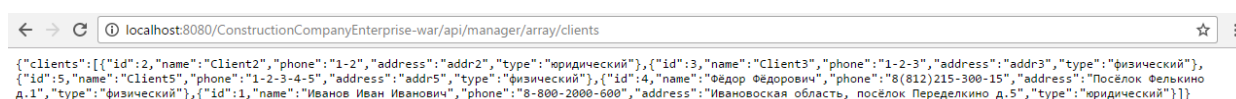
- **{path}/api/{role}/array{name}/{id}/update** - обновление существующей записи массива. Для каждой роли доступны определённые массивы: для роли client только массив orders, для master массив works, для manager - любой из массивов данных.
- **{path}/api/{role}/profile** - обновление для собственных данных роли.

для delete:

Доступны только для роли менеджера: **{path}/api/{role}/array/{name}/{id}** может удалять элементы из любого массива сущностей.

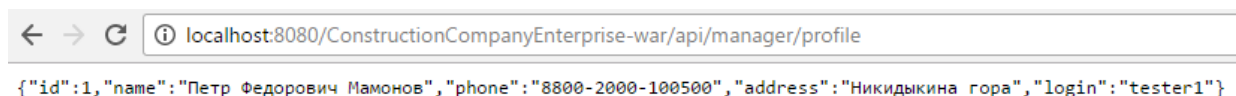
JSON - поля объектов соответствуют записям базы данных (исключение это команды для обновления записей выполненные от роли клиента и прораба) массивы объектов это название сущности в множественном числе например для получения записи о клиентах используя роль менеджера при запросе

{path}/api/manager/array/clients мы получим записи в массиве clients см. рис. 9



```
{
  "clients": [
    {
      "id": 2,
      "name": "Client2",
      "phone": "1-2",
      "address": "addr2",
      "type": "юридический"
    },
    {
      "id": 3,
      "name": "Client3",
      "phone": "1-2-3",
      "address": "addr3",
      "type": "физический"
    },
    {
      "id": 5,
      "name": "Client5",
      "phone": "1-2-3-4-5",
      "address": "addr5",
      "type": "физический"
    },
    {
      "id": 4,
      "name": "Федор Федорович",
      "phone": "8(812)215-300-15",
      "address": "Посёлок Фелькино д.1",
      "type": "физический"
    },
    {
      "id": 1,
      "name": "Иванов Иван Иванович",
      "phone": "8-800-2000-600",
      "address": "Ивановская область, посёлок Переделкино д.5",
      "type": "юридический"
    }
  ]
}
```

Рис. 12: json массив клиентов



```
{
  "id": 1,
  "name": "Петр Федорович Мамонов",
  "phone": "8800-2000-100500",
  "address": "Никидыкина гора",
  "login": "tester1"
}
```

Рис. 13: данные профиля пользователя

```
localhost:8080/ConstructionCompanyEnterprise-war/api/manager/
{"id":1,"name":"Петр Федорович Мамонов","phone":"8800-2000-100500","address":"Никидыкина гора","login":"tester1","orders":
[{"orderId":1,"number":1,"status":"открыт","cost":1000.0,"create":"25.04.2016 21:01:02","update":"25.04.2016 21:01:06","end":"--/--","managername":"Петр Федорович
Мамонов","managerphone":"8800-2000-100500","manageraddress":"Никидыкина гора","clientname":"Иванов Иван Иванович","clientphone":"8-800-2000-600","clientaddress":"Ивановская
область, посёлок Переделкино д.5","estimates":[{"estimateId":1,"type":"основная","status":"не оплачена | не завершена","costCalculation":8730.0,"works":
[{"workItemId":1,"description":"work1","serviceCost":100.0,"costCalculation":2510.0,"resources":[{"resourceItemId":1,"name":"resource
1","productCode":1,"resourceId":1,"cost":100.0,"amount":0.5,"refer":"pa60re"}, {"resourceItemId":2,"name":"resource
2","productCode":2,"resourceId":2,"cost":200.0,"amount":1.5,"refer":"pa60re"}, {"resourceItemId":3,"name":"resource
3","productCode":3,"resourceId":3,"cost":300.0,"amount":0.2,"refer":"pa60re"}, {"resourceItemId":5,"name":"resource
5","productCode":5,"resourceId":4,"cost":400.0,"amount":5.0,"refer":"pa60re"}]}, {"workId":1,"orderNumber":1,"managerName":"Петр Федорович Мамонов","managerPhone":"8800-2000-
100500","clientName":"Иванов Иван Иванович","clientPhone":"8-800-2000-600","clientAddress":"Ивановская область, посёлок Переделкино д.5","status":"не завершена"},
{"workItemId":2,"description":"work2","serviceCost":1000.0,"costCalculation":3110.0,"resources":[{"resourceItemId":1,"name":"resource
1","productCode":1,"resourceId":1,"cost":100.0,"amount":0.5,"refer":"pa60re"}, {"resourceItemId":3,"name":"resource
3","productCode":3,"resourceId":3,"cost":300.0,"amount":0.2,"refer":"pa60re"}, {"resourceItemId":5,"name":"resource
5","productCode":5,"resourceId":4,"cost":400.0,"amount":5.0,"refer":"pa60re"}]}, {"workId":2,"orderNumber":1,"managerName":"Петр Федорович Мамонов","managerPhone":"8800-2000-
100500","clientName":"Иванов Иван Иванович","clientPhone":"8-800-2000-600","clientAddress":"Ивановская область, посёлок Переделкино д.5","status":"не завершена"},
{"workItemId":2,"description":"work2","serviceCost":1000.0,"costCalculation":3110.0,"resources":[{"resourceItemId":1,"name":"resource
1","productCode":1,"resourceId":1,"cost":100.0,"amount":0.5,"refer":"pa60re"}, {"resourceItemId":3,"name":"resource
3","productCode":3,"resourceId":3,"cost":300.0,"amount":0.2,"refer":"pa60re"}, {"resourceItemId":5,"name":"resource
5","productCode":5,"resourceId":4,"cost":400.0,"amount":5.0,"refer":"pa60re"}]}, {"workId":3,"orderNumber":1,"managerName":"Петр Федорович Мамонов","managerPhone":"8800-2000-
100500","clientName":"Иванов Иван Иванович","clientPhone":"8-800-2000-600","clientAddress":"Ивановская область, посёлок Переделкино д.5","status":"завершена"}]},
{"estimateId":2,"type":"дополнительная","status":"не оплачена | завершена","costCalculation":0.0,"works":[]}},
{"orderId":4,"number":3,"status":"открыт","cost":1625.0,"create":"22.04.2016 21:27:29","update":"22.04.2016 21:27:32","end":"--/--","managername":"Петр Федорович
Мамонов","managerphone":"8800-2000-100500","manageraddress":"Никидыкина гора","clientname":"Client3","clientphone":"1-2-3","clientaddress":"addr3","estimates":[]},
{"orderId":2,"number":2,"status":"отправлен на подтверждение, клиенту","cost":5415.0,"create":"25.04.2016 23:11:21","update":"25.04.2016 23:11:21","end":"--/--",
"managername":"Петр Федорович Мамонов","managerphone":"8800-2000-100500","manageraddress":"Никидыкина гора","clientname":"Client2","clientphone":"1-
2","clientaddress":"addr2","estimates":[{"estimateId":3,"type":"основная","status":"не оплачена | не завершена","costCalculation":550.0,"works":
[{"workItemId":4,"description":"work4","serviceCost":550.0,"costCalculation":550.0,"resources":[{"workId":4,"orderNumber":2,"managerName":"Петр Федорович
Мамонов","managerPhone":"8800-2000-100500","clientName":"Client2","clientPhone":"1-2","clientAddress":"addr2","status":"не завершена"}]},
{"estimateId":4,"type":"дополнительная","status":"не оплачена | завершена","costCalculation":0.0,"works":[{"estimateId":5,"type":"дополнительная","status":"не оплачена |
завершена","costCalculation":0.0,"works":[{"storageId":1,"location":"storage addr 1","resources":[{"resourceItemId":1,"name":"resource
1","productCode":1,"resourceId":1,"cost":100.0,"amount":0.5,"refer":"pa60re"}, {"resourceItemId":2,"name":"resource
2","productCode":2,"resourceId":2,"cost":200.0,"amount":1.5,"refer":"pa60re"}, {"resourceItemId":3,"name":"resource
3","productCode":3,"resourceId":3,"cost":300.0,"amount":0.2,"refer":"pa60re"}, {"resourceItemId":5,"name":"resource
5","productCode":5,"resourceId":4,"cost":400.0,"amount":5.0,"refer":"pa60re"}, {"resourceItemId":7,"name":"resource
7","productCode":7,"resourceId":5,"cost":500.0,"amount":1.8,"refer":"pa60re"}]}, {"storageId":2,"location":"storage addr 2","resources":[{"resourceItemId":1,"name":"resource
1","productCode":1,"resourceId":1,"cost":100.0,"amount":0.5,"refer":"pa60re"}, {"resourceItemId":2,"name":"resource
2","productCode":2,"resourceId":2,"cost":200.0,"amount":1.5,"refer":"pa60re"}, {"resourceItemId":3,"name":"resource
3","productCode":3,"resourceId":3,"cost":300.0,"amount":0.2,"refer":"pa60re"}, {"resourceItemId":5,"name":"resource
5","productCode":5,"resourceId":4,"cost":400.0,"amount":5.0,"refer":"pa60re"}, {"resourceItemId":7,"name":"resource
7","productCode":7,"resourceId":5,"cost":500.0,"amount":1.8,"refer":"pa60re"}]}, {"storageId":3,"location":"storage addr 3","resources":[{"resourceItemId":1,"name":"resource
1","productCode":1,"resourceId":1,"cost":100.0,"amount":0.5,"refer":"pa60re"}, {"resourceItemId":2,"name":"resource
2","productCode":2,"resourceId":2,"cost":200.0,"amount":1.5,"refer":"pa60re"}, {"resourceItemId":3,"name":"resource
3","productCode":3,"resourceId":3,"cost":300.0,"amount":0.2,"refer":"pa60re"}, {"resourceItemId":5,"name":"resource
5","productCode":5,"resourceId":4,"cost":400.0,"amount":5.0,"refer":"pa60re"}, {"resourceItemId":7,"name":"resource
7","productCode":7,"resourceId":5,"cost":500.0,"amount":1.8,"refer":"pa60re"}]}, {"storageId":4,"location":"storage addr 4","resources":[{"resourceItemId":1,"name":"resource
1","productCode":1,"resourceId":1,"cost":100.0,"amount":0.5,"refer":"pa60re"}, {"resourceItemId":2,"name":"resource
2","productCode":2,"resourceId":2,"cost":200.0,"amount":1.5,"refer":"pa60re"}, {"resourceItemId":3,"name":"resource
3","productCode":3,"resourceId":3,"cost":300.0,"amount":0.2,"refer":"pa60re"}, {"resourceItemId":5,"name":"resource
5","productCode":5,"resourceId":4,"cost":400.0,"amount":5.0,"refer":"pa60re"}, {"resourceItemId":7,"name":"resource
7","productCode":7,"resourceId":5,"cost":500.0,"amount":1.8,"refer":"pa60re"}]}, {"storageId":5,"location":"storage addr 5","resources":[]}], "works":
[{"workItemId":1,"description":"work1","serviceCost":100.0,"costCalculation":2510.0,"resources":[{"resourceItemId":1,"name":"resource
1","productCode":1,"resourceId":1,"cost":100.0,"amount":0.5,"refer":"pa60re"}, {"resourceItemId":2,"name":"resource
2","productCode":2,"resourceId":2,"cost":200.0,"amount":1.5,"refer":"pa60re"}, {"resourceItemId":3,"name":"resource
3","productCode":3,"resourceId":3,"cost":300.0,"amount":0.2,"refer":"pa60re"}, {"resourceItemId":5,"name":"resource
5","productCode":5,"resourceId":4,"cost":400.0,"amount":5.0,"refer":"pa60re"}]}, {"workItemId":2,"description":"work2","serviceCost":1000.0,"costCalculation":3110.0,"resources":[{"resourceItemId":1,"name":"resource
1","productCode":1,"resourceId":1,"cost":100.0,"amount":0.5,"refer":"pa60re"}, {"resourceItemId":3,"name":"resource
3","productCode":3,"resourceId":3,"cost":300.0,"amount":0.2,"refer":"pa60re"}, {"resourceItemId":5,"name":"resource
5","productCode":5,"resourceId":4,"cost":400.0,"amount":5.0,"refer":"pa60re"}]}, {"workItemId":3,"description":"work3","serviceCost":450.0,"costCalculation":1350.0,"resources":[{"resourceItemId":7,"name":"resource
7","productCode":7,"resourceId":5,"cost":500.0,"amount":1.8,"refer":"pa60re"}]}]}
```

Рис. 14: данные менеджера

6. Авторизация

Авторизация осуществляется с помощью JAAS (Java Authentication and Authorization Service). Во время авторизации пользователь посылает свои креденциалы (credentials, в простом случае это могут быть логин: пароль или пользовательский сертификат). На сервере с процессом контроля связаны две сущности, а именно Realm и Login Module. Login Module осуществляет проверку связи пользователя с каким-то набором групп пользователей (не путать с ролями). Кроме того, авторизация может быть пройдена успешно, однако, пользователь может быть не связан ни с одной из групп.

Веб-приложение может определять соотношение между группами и ролями. Приложение задаёт это соответствие с помощью web.xml.

Login Module и Realm не являются частью веб-приложения, а являются разделяемыми ресурсами сервера приложений, так что они должны быть в classpath сервера и должны быть соответствующим образом зарегистрированы в сервере (в login.conf и domain.xml). Приложение выбирает realm по имени через web.xml.

Для GlassFish Server создать Realm через панель администратора указав источник для данных, это таблицы в базе данных с пользователями и ролями (есть и другие варианты, но в проекте был рассмотрен случай с базой данных как наиболее подходящий):

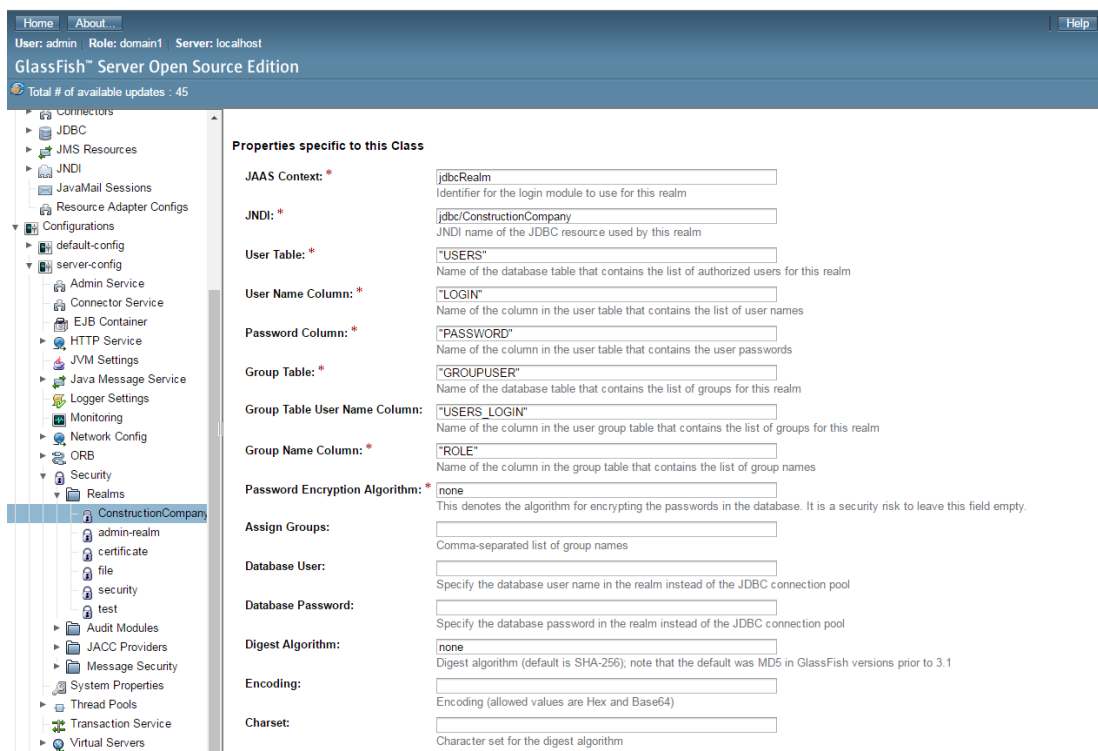


Рис. 15: раздел Realm для данного проекта

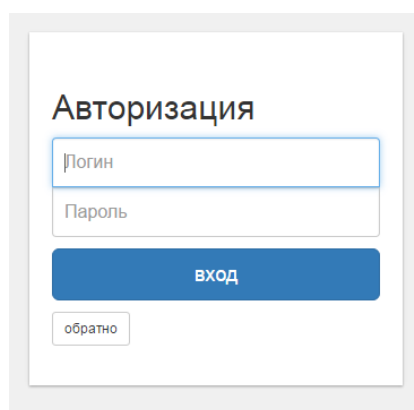


Рис. 16: форма авторизации приложения

7. Тестирование

Так как код бизнес логики был взят из прошлого проекта и все изменения не затронули поведение системы то на тестировании бизнес логики это не отражается.

businesslogic

| Element | Missed Instructions | Cov. | Missed Branches | Cov. | Missed | Cxty | Missed | Lines | Missed | Methods | Missed | Classes |
|----------|---------------------|------|-----------------|------|--------|------|--------|-------|--------|---------|--------|---------|
| Manager | <div></div> | 100% | <div></div> | 100% | 0 | 35 | 0 | 77 | 0 | 9 | 0 | 1 |
| Order | <div></div> | 100% | <div></div> | 100% | 0 | 50 | 0 | 94 | 0 | 33 | 0 | 1 |
| Work | <div></div> | 100% | <div></div> | 100% | 0 | 33 | 0 | 56 | 0 | 19 | 0 | 1 |
| Estimate | <div></div> | 100% | <div></div> | 100% | 0 | 32 | 0 | 54 | 0 | 22 | 0 | 1 |
| Storage | <div></div> | 100% | <div></div> | 100% | 0 | 25 | 0 | 43 | 0 | 13 | 0 | 1 |
| Resource | <div></div> | 100% | <div></div> | 100% | 0 | 18 | 0 | 39 | 0 | 11 | 0 | 1 |
| Client | <div></div> | 100% | <div></div> | n/a | 0 | 8 | 0 | 14 | 0 | 8 | 0 | 1 |
| Role | <div></div> | 100% | <div></div> | n/a | 0 | 7 | 0 | 14 | 0 | 7 | 0 | 1 |
| Master | <div></div> | 100% | <div></div> | n/a | 0 | 2 | 0 | 4 | 0 | 2 | 0 | 1 |
| ErrorMsg | <div></div> | 100% | <div></div> | n/a | 0 | 1 | 0 | 4 | 0 | 1 | 0 | 1 |
| Total | 0 of 1 333 | 100% | 0 of 167 | 100% | 0 | 211 | 0 | 399 | 0 | 125 | 0 | 10 |

Рис. 17: тестовое покрытие бизнес-логики

| Вариант использования | Ожидаемый результат | Фактический результат |
|---|---|---|
| Выбор сметы из списка заказов и нажатие кнопки Оплатить | После введения в диалоговом окне суммы для оплаты смета помечается как оплаченная | смета помечается как оплаченная на странице, а в случае полной оплаты заказа на странице менеджера заказ помечается как оплаченный и его статус переходит |
| Изменения в профиле и их сохранение | Изменяются данные пользовательского профиля | Данные пользователя в базе данных изменяются на новые, так же это отобразит изменения в зависимых отображениях |
| Изменение статуса заказа по кнопке подтвердить | Заказ перейдёт из статуса «отправлен на подтверждение, клиенту» на «на подтверждение оплаты, менеджеру» | Статус заказа изменяется, на странице заказчика исчезает кнопка подтверждения а на странице менеджера появляется |

Таблица 1: Тестирование страницы заказчика

| Вариант использования | Ожидаемый результат | Фактический результат |
|---|---|---|
| Выбор работы из списка и нажатие кнопки завершить | Работа помечается как завершённая | Работа помечается как завершённая, а так же проверяется статус выполнения сметы по количеству завершённых работ |
| Изменения в профиле и их сохранение | Изменяются данные пользовательского профиля | Данные пользователя в базе данных изменяются на новые, так же это отобразит изменения в зависимых отображениях |

Таблица 2: Тестирование страницы прораба

| Вариант использования | Ожидаемый результат | Фактический результат |
|---|---|---|
| Изменения состояния, добавления и удаления различных сущностей (ресурсы, работа, смета, заказ, заказчик, прораб, склад) | После введения в диалоговом окне и проверки на не противоречивость данные сохраняются | Сущности изменяются в соответствии с заданными, в пределах ограничений области допустимых значений для данных |
| Изменения в профиле и их сохранение | Изменяются данные пользовательского профиля | Данные пользователя в базе данных изменяются на новые, так же это отобразит изменения в зависимых отображениях |
| Изменение статуса заказа по кнопке закрыть | Заказ перейдёт из статуса «на подтверждение оплаты, менеджеру» на «закрыт» | Статус заказа изменяется, на странице менеджера исчезает кнопка подтверждения, а на странице заказчика статус заказа изменяется на закрыт |

Таблица 3: Тестирование страницы менеджера

8. Графика

Скиншоты html страницы в браузере для каждой роли:

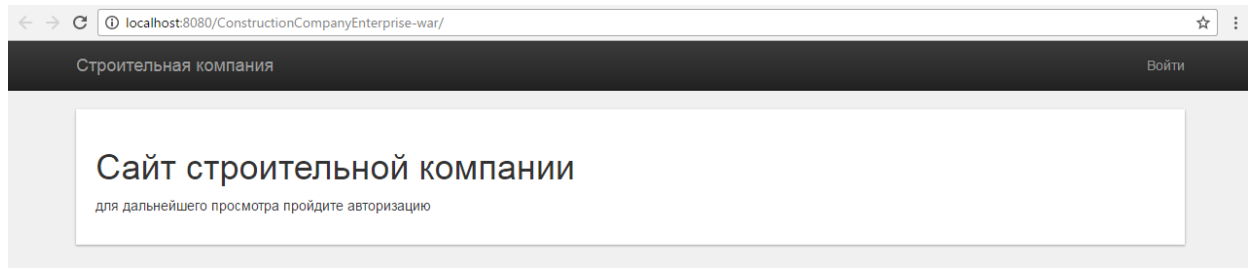


Рис. 18: приветственная страница

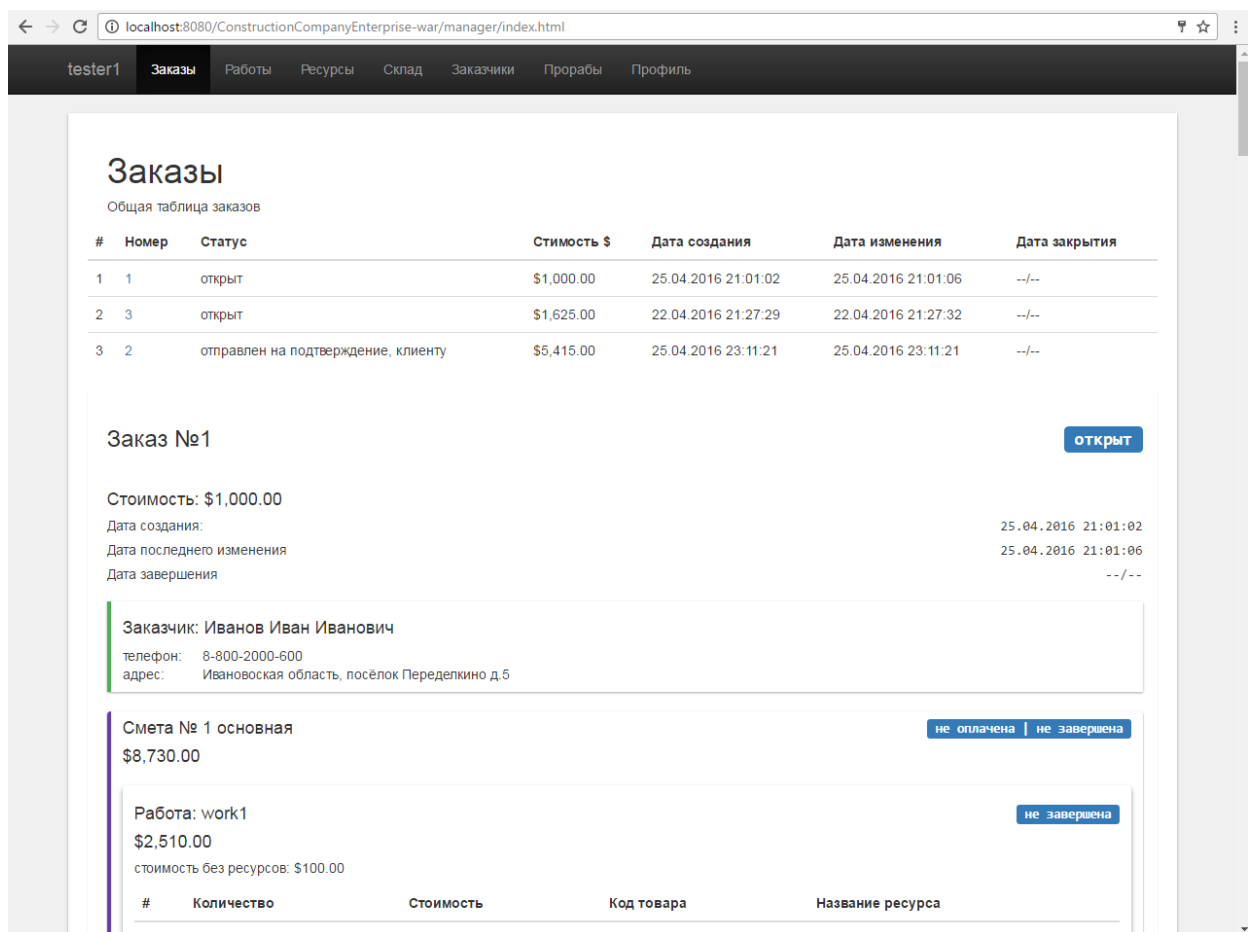


Рис. 19: роль менеджера

[tester2](#)
[Работы](#)
[Профиль](#)

Работы

| № | Стоимость | Полная стоимость | Номер заказа | Название работы | Статус |
|---|------------|------------------|--------------|------------------|--------------|
| # | Количество | Стоимость | Код товара | Название ресурса | |
| 1 | 0.5 | \$100.00 | 1 | resource 1 | |
| 2 | 1.5 | \$200.00 | 2 | resource 2 | |
| 3 | 0.2 | \$300.00 | 3 | resource 3 | |
| 4 | 5 | \$400.00 | 5 | resource 5 | |
| 1 | \$100.00 | \$2,510.00 | 1 | work1 | не завершена |
| <div> <div>Имя менеджера: Петр Федорович Мамонов Телефон: 8800-2000-100500</div> <div>Имя заказчика: Иванов Иван Иванович Телефон: 8-800-2000-600</div> <div>Адрес заказчика: Ивановская область, посёлок Переделкино д.5</div> </div> <div>завершить</div> | | | | | |
| # | Количество | Стоимость | Код товара | Название ресурса | |
| 1 | 0.5 | \$100.00 | 1 | resource 1 | |
| 2 | 0.2 | \$300.00 | 3 | resource 3 | |
| 3 | 5 | \$400.00 | 5 | resource 5 | |
| 2 | \$1,000.00 | \$3,110.00 | 1 | work2 | не завершена |
| <div> <div>Имя менеджера: Петр Федорович Мамонов Телефон: 8800-2000-100500</div> <div>Имя заказчика: Иванов Иван Иванович Телефон: 8-800-2000-600</div> <div>Адрес заказчика: Ивановская область, посёлок Переделкино д.5</div> </div> <div>завершить</div> | | | | | |
| # | Количество | Стоимость | Код товара | Название ресурса | |
| 1 | 0.5 | \$100.00 | 1 | resource 1 | |
| 2 | 0.2 | \$300.00 | 3 | resource 3 | |

Рис. 20: роль прораба

[tester3](#)
[Заказы](#)
[Профиль](#)

Заказы

Заказ №1

открыт

Стоимость: \$1,000.00

Дата создания: 25.04.2016 21:01:02

Дата последнего изменения: 25.04.2016 21:01:06

Дата завершения: --/--

Петр Федорович Мамонов

Ответственный менеджер

телефон: 8800-2000-100500

адрес офиса: Никидыкина гора

Смета № 1 основная

не оплачена | не завершена

\$8,730.00

Работа: work1

не завершена

\$2,510.00

стоимость без ресурсов: \$100.00

| # | Количество | Стоимость | Код товара | Название ресурса |
|---|------------|-----------|------------|------------------|
| 1 | 0.5 | \$100.00 | 1 | resource 1 |
| 2 | 1.5 | \$200.00 | 2 | resource 2 |
| 3 | 0.2 | \$300.00 | 3 | resource 3 |
| 4 | 5 | \$400.00 | 5 | resource 5 |

Рис. 21: роль заказчика

19

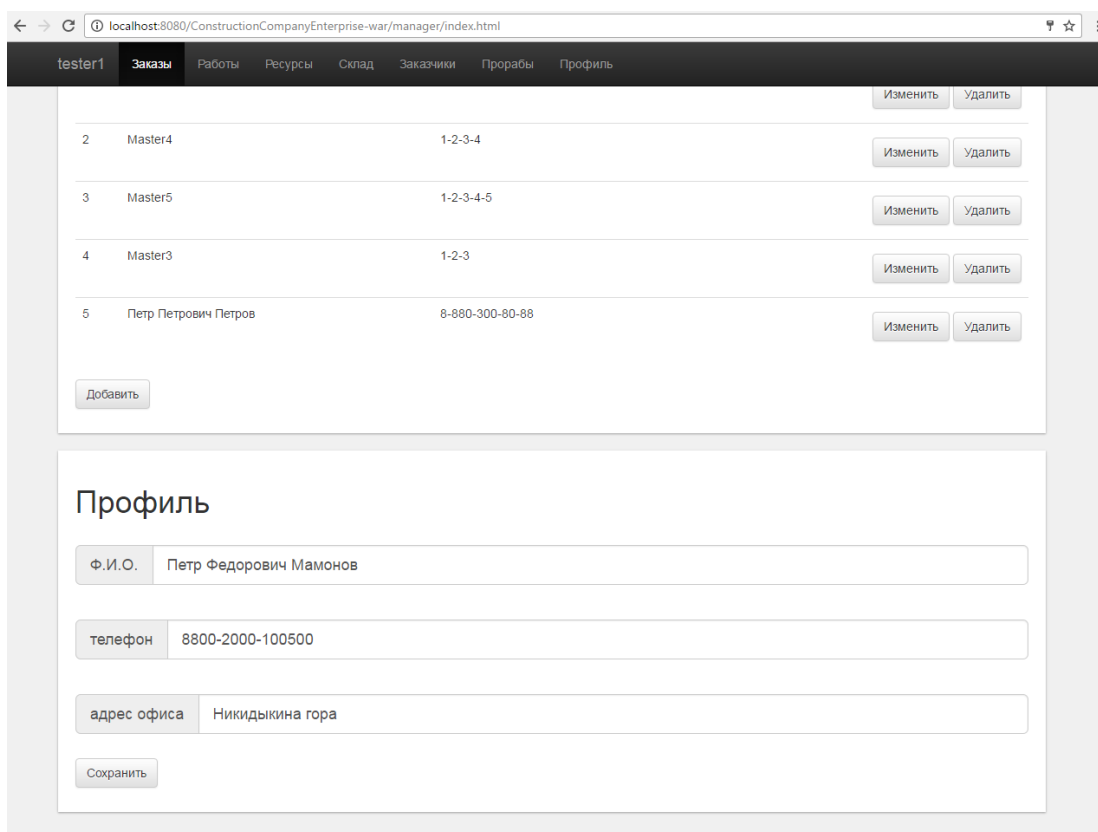


Рис. 22: профиль менеджера

9. Выводы

В качестве сервера приложений использовался GlassFish Server 4.1, в качестве ORM - EclipseLink. Для хранения данных использовалась СУБД PostgreSQL 9.6. Для web клиента использованы сторонние библиотеки :

- Angular JS для работы http запросов и отображения JSON данных полученных от сервера, чтобы сократить трафик обмена информацией между клиентом и сервером.
- Для визуального стилистического оформления html страниц была использована библиотека Bootstrap 3.
- Для работы фреймворка и библиотеки понадобилась подключить jQuery.

Данная работа моделирует пример проектирование архитектуры распределённой информационной системы для бизнес процессов на примере строительной организации. Были изучены технологии EJB (Enterprise Java Beans) и JPA (Java Persistence API).