Содержание:

1. Постановка задачи: словесное описание предметной области. (стр 3)
2. Концептуальное проектирование: Диаграммы вариантов использования. (стр 4)
3. Логическое проектирование: Диаграмма сущность-связь. (стр 5)
4. Физическое проектирование: Схема БД. (стр 6)
5. Тексты основных запросов, функций, процедур и триггеров. (стр 7-8)
6. Интерфейс приложения: Основные окна с описанием. (стр 9-11)
7. Инструментальные средства, которые использованы при разработке. (стр 12)

**Постановка задачи: Словесное описание предметной области.**

**Вариант 4.** «Оплата за междугородние разговоры»

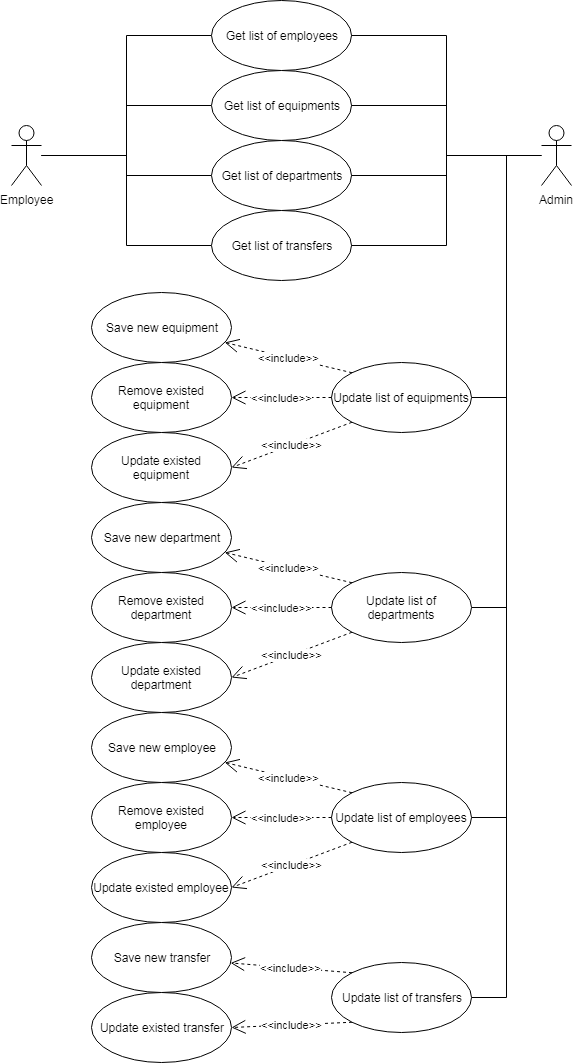
Предприятие связи, предоставляющие услуги междугородней связи.

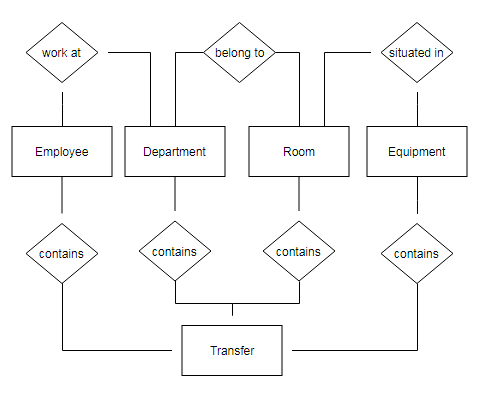
**Словесное описание предметной области:** Технолог междугородней телефонной станции (МТС) вносит в БД информацию о новых клиентах, пользующихся услугами, это: номер телефона клиента, ФИО, адрес, дата регистрации. Также технолог изменяет цены за пользование услугами, эти данные хранятся в следующем виде: дата, название населенного пункта, с которым осуществляется связь, стоимость одной минуты разговора, льготная стоимость (с 20.00 до 6.00). Оператор МТС фиксирует дату разговора, город, с которым произошел разговор, номер абонента, длительность разговора. После этого оператор отсылает абоненту квитанцию об оплате. Он также принимает извещение об оплате и фиксирует, что те или иные разговоры оплачены.

**Необходимо реализовать выполнения следующих функций**:

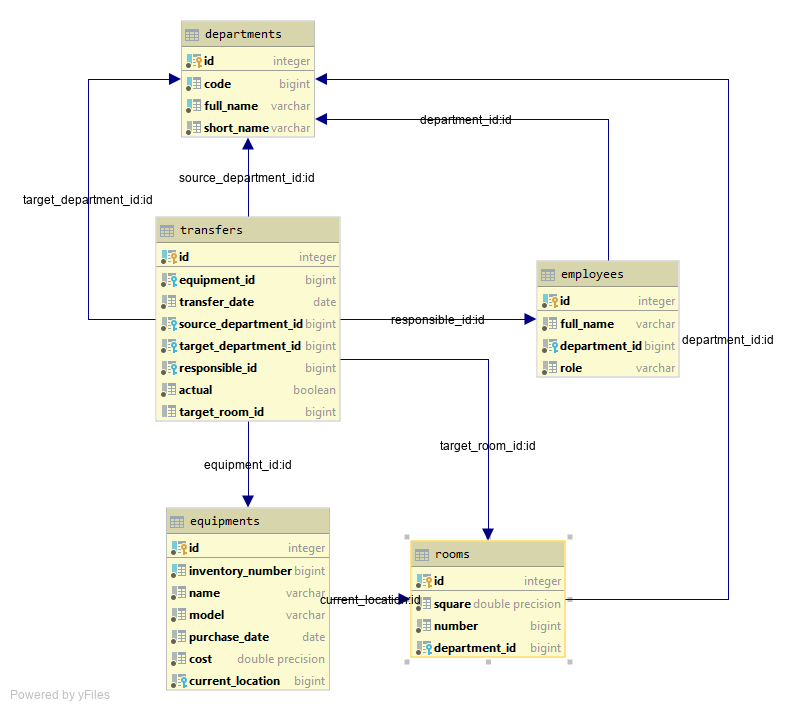
* Добавление/редактирование/удаление информации о клиентах телефонной станции.
* Добавление/редактирование/удаление информации о стоимости услуг связи.
* Добавление/редактирование/удаление информации о телефонном звонке (функция оператора МТС).
* Расчет количества абонентов, соединившихся с выбранным городом за выбранный месяц с разбивкой по дням.
* Просмотр списка абонентов, имеющих задолженности по оплате (срок оплаты – 20 дней) – дата, название организации, предоставляющей услуги связи, телефон, ФИО, адрес абонента.
* Вывести для просмотра стоимость оплаты одной минуты разговора для разных населенных пунктов на заданную дату: название организации, предоставляющей услуги связи - дата, название населенного пункта, стоимость одной минуты, льготная стоимость одной минуты.

**Диаграммы вариантов использования (ВИ, Use Case)**



**Логическое проектирование: Диаграмма сущность-связь (ER-диаграмма)**

**Физическое проектирование: Схема БД.**

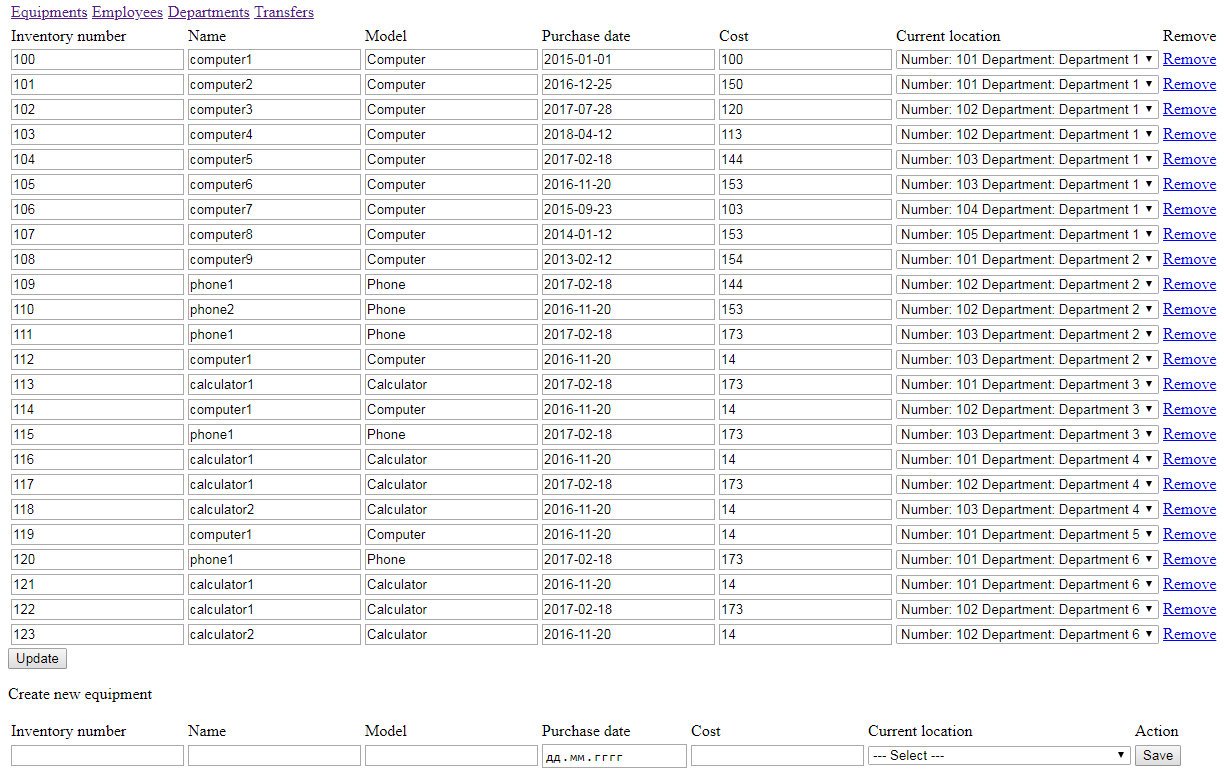


**Тексты основных запросов, функций, процедур и триггеров.**

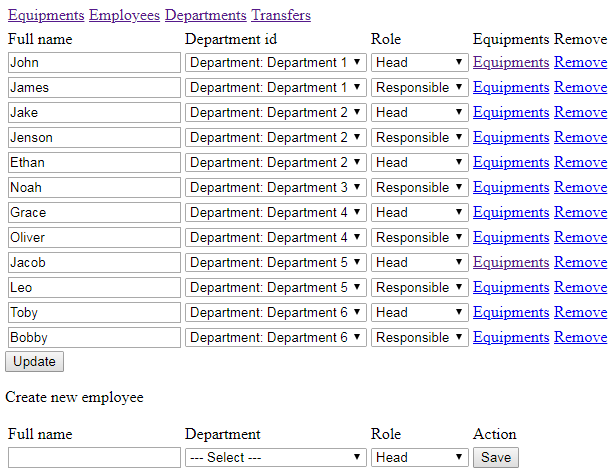
Все запросы в базу данных выполнялись с помощью механизма объектно-реляционного отображения. Поэтому для составления запросов использовался язык JPQL.

|  |  |  |
| --- | --- | --- |
| Текст запроса | Передаваемые аргументы | Результат запроса |
| **SELECT d FROM Department d WHERE d.id = :id** | :id – уникальный идентификатор подразделения | Возвращает объект, который соответствует представлению «подразделение» в БД |
| **SELECT d FROM Department d** | - | Возвращает все объекты, соответствующие представлению «подразделение» в БД |
| **SELECT d FROM Department d WHERE d.code = :code** | :code – номер подразделения | Возвращает объект, который соответствует представлению «подразделение» в БД |
| **SELECT e FROM Employee e WHERE e.id = :id** | :id – уникальный идентификатор сотрудника | Возвращает объект, который соответствует представлению «сотрудник» в БД |
| **SELECT e FROM Employee e** | - | Возвращает все объекты, соответствующие представлению «сотрудник» в БД |
| **SELECT e FROM Equipment e WHERE e.id = :id** | :id – уникальный идентификатор оборудования | Возвращает объект, который соответствует представлению «оборудование» в БД |
| **SELECT e FROM Equipment e** | - | Возвращает все объекты, соответствующие представлению «оборудование» в БД |
| **SELECT e FROM Equipment e WHERE e.currentLocation.department.code = :departmentCode** | :departmentCode – номер подразделения, которому принадлежит комната, где находится оборудование | Возвращает объект, который соответствует представлению «оборудование» в БД |
| **SELECT r FROM Room r** | - | Возвращает все объекты, соответствующие представлению «комната» в БД |
| **SELECT t FROM Transfer t WHERE t.id = :id** | :id – уникальный идентификатор оборудования | Возвращает объект, который соответствует представлению «передача» в БД |
| **SELECT t FROM Transfer t** | - | Возвращает все объекты, соответствующие представлению «передача» в БД |
| **SELECT t FROM Transfer t**  **WHERE t.actual = TRUE AND t.responsible.id = :responsibleId** | :responsibleId – уникальный идентификатор ответственного за оборудование | Возвращает все объекты, соответствующие представлению «передача» в БД, которые являются актуальными на данный момент времени |
| **UPDATE FROM Transfer t SET t.actual = FALSE WHERE t.equipment.id = :equipmentId** | :equipmentId – уникальный идентификатор оборудования, которое было передано в другое подразделение | Делает предыдущие передачи оборудования с таким идентификатором недействительными |

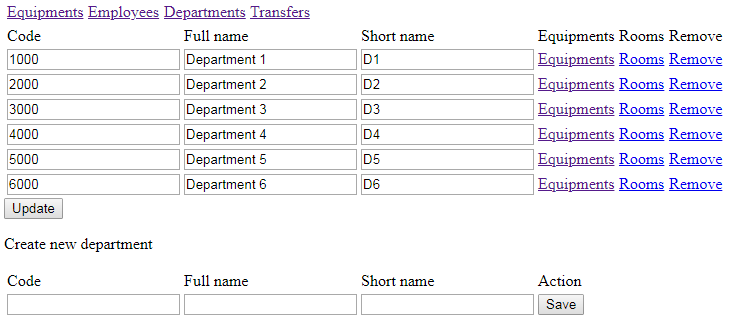
**Интерфейс приложения**



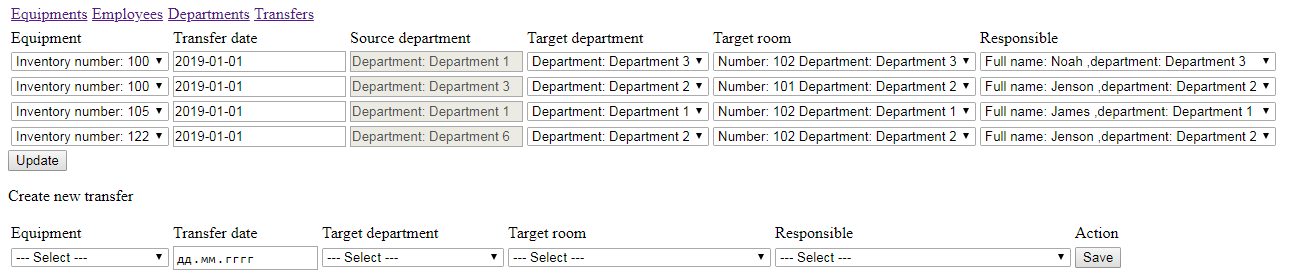
Страница 1. Список всего оборудования с элементами управления



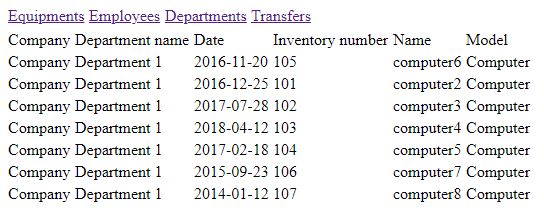
Страница 2. Список всех сотрудников с элементами управления



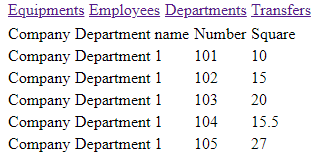
Страница 3. Список всех подразделений с элементами управления



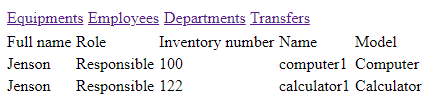
Страница 4. Список всех переводов с элементами управления



Страница 5. Список всего оборудования, закрепленного за данным подразделением.



Страница 6. Список всех комнат, закрепленных за данным подразделением.



Страница 7. Список всего оборудования, закрепленного за данным материально ответственным лицом

**Инструментальные средства, которые использованы при разработке.**

Используемый язык программирования: Java (1.8)

Используемые фреймворки: Spring, Hibernate

Используемая СУБД: PostgreSQL

Используемая IDE: IntelliJ IDEA

Используемая система контроля версий: git

Технологии создания веб-страниц: HTML, JSP