

Описание учебной задачи "Система хранения данных сетевых клиентов"

1. Общее описание

- 1.1. Система представляет собой многопользовательское веб-приложение, которое позволяет пользователям заносить в базу данных сведения о сетевых клиентах. Также пользователь может запрашивать из базы данных список всех данных и данные по заданным параметрам.

2. Описание сущностей

2.1. Классы-модели:

- Client, поля: идентификатор (clientid - целое число), наименование клиента (client_name - строка до 100 символов), тип клиента (type - строка до 20 символов), дата добавления клиента (added - дата), список адресов (addresses - List<Address>).
- Address, включающая поля: сетевой адрес устройства (ip - строка до 25 символов), физический-адрес устройства (mac - строка до 20 символов), модель устройства (model - строка до 100 символов), адрес места нахождения (address - строка до 200 символов), клиент (client – объект класса Client).

3. Реализация

- 3.1. Реализация приложения выполняется в 3 этапа.

- 3.2. Первый этап включает реализацию на основе технологии сервлетов со следующей функциональностью:

- 3.2.1. Сервлет «Create» обеспечивает создание нового клиента и его адреса, которые включают:

- 3.2.1.1. Ввод данных для всех полей класса Client и проверка допустимости сведений:

- 3.2.1.1.1. Все поля не могут быть пустыми.

- 3.2.1.1.2. Длина полей не превышает ограничения.

3.2.1.1.3. Для поля наименование клиента допустимо использование только русского алфавита, а также символов {- ,.}

3.2.1.1.4. Для поля «type» допустимы значения:

- Юридическое лицо;
- Физическое лицо.

3.2.1.2. Ввод данных для всех полей класса Address и проверка допустимости сведений:

3.2.1.2.1. Все поля не могут быть пустыми.

3.2.1.2.2. Длина полей не превышает ограничения.

3.2.1.2.3. IP адрес состоит из четырех октетов разделенных точкой, каждый октет имеет размер не более трех чисел, числа должны быть в диапазоне от 0 до 255, например: 192.168.000.001

3.2.1.2.4. MAC-адрес состоит из шести октетов разделенных символом «-». Каждый октет имеет размер не более двух символов и состоит из чисел и/или букв латинского алфавита.

3.2.1.3. Приложение должно информировать о попытке ввода недопустимых значений.

3.2.1.4. В случае ввода допустимых значений должны создаваться новые сущности Client и Address, и должен происходить переход на форму просмотра всех клиентов (класс ViewList).

3.2.2. Класс сервлета Update обеспечивает изменение существующих сведений, который включает:

3.2.2.1. Изменение данных всех полей классов-моделей и их проверка в соответствии с п. 3.2.1.1 и 3.2.1.2.

3.2.3. Класс сервлета Delete обеспечивает удаление выбранного в форме ViewList объекта.

3.2.4. Класс сервлета ViewList обеспечивает вывод всех данных сущностей Client и Address в общей таблице.

3.2.4.1. Необходимо обеспечить фильтрацию выводимых в таблицу сведений по содержимому полей: client_name, type, address. Фильтр состоит из двух полей.

- 3.2.4.1.1. Поле выбора типа клиента. Выбор происходит из выпадающего списка.
 - 3.2.4.1.2. Поле ввода текста для поиска по полям client_name и address. Поиск производится по содержанию каждого введенного слова в указанных полях сущностей.
- 3.3. На втором этапе приложение расширяется за счёт подключения сессионных бинов со следующей функциональностью.
 - 3.3.1. Класс UpdateBean типа @Stateless обеспечивает обработку запросов на изменение данных:
 - 3.3.1.1. Занесение новых сведений в базу данных
 - 3.3.1.2. Изменение существующих сведений в базе данных
 - 3.3.1.3. Удаление существующих сведений из базы данных
 - 3.3.1.4. Реализовать в данном классе проверку допустимости сведений, указанных в п. 3.2.1.1 и 3.2.1.2
 - 3.3.2. Класс SelectBean типа @Stateless обеспечивает выполнение запросов от класса ViewList к базе данных на получение выборки
 - 3.3.2.1. Класс должен обрабатывать следующие виды запросов:
 - 3.3.2.1.1. Получение всех записей
 - 3.3.2.1.2. Получение одной записи по имени параметра
 - 3.3.2.1.3. Получение всех записей, имена параметров которых удовлетворяют заданному пользователем шаблону.
- 3.4. На третьем этапе в приложение вводится класс DbManager типа @Singleton
 - 3.4.1. Данный класс обеспечивает взаимодействие с базой данных на основе технологии JPA
 - 3.4.2. Данный класс обеспечивает выполнение следующих функций
 - 3.4.2.1. Установку и поддержание соединения с базой данных
 - 3.4.2.2. Управление объектами вида Entity
 - 3.4.2.3. Непосредственное выполнение запросов от классов UpdateBean и SelectBean.