

Итоговое задание

В задании требуется написать приложение, моделирующее работу информационной системы оператора сотовой связи. Ниже более подробно описаны предметная область и технические требования.

Предметная область

Существуют следующие виды сущностей:

- Тариф
 - Название
 - Цена
 - Список возможных опций
- Опция
 - Название
 - Цена
 - Стоимость подключения
- Клиент
 - Имя
 - Фамилия
 - Дата рождения
 - Паспортные данные
 - Адрес
 - Номера контрактов(телефонов у данного оператора)
 - Адрес электронной почты
 - Пароль для личного кабинета
- Контракт
 - Номер
 - Тариф
 - Выбранные опции для тарифа

Приложение должно обеспечивать следующую функциональность:

- Для клиентов компании
 - Просмотр контракта в личном кабинете;
 - Просмотр всех возможных тарифов для перехода, смена тарифа;
 - Просмотр возможных опций для тарифа, подключение новых опций, отключение существующих;
 - Блокировка/разблокировка номера(если номер заблокирован, то нельзя сменить тариф и опции; если номер заблокировал не пользователь, то разблокировать сам он его не может);
- Для сотрудников компании
 - Заключение контракта с новым клиентом: выбор нового номера телефона с подключением тарифа и опций. Номер телефона должен быть уникальным.
 - Просмотр всех пользователей и контрактов;

- Блокировка пользователя/разблокировка пользователя;
- Поиск пользователя по номеру;
- Смена тарифа, подключение и отключение опций пользователю;
- Добавление новых тарифов, удаление старых;
- Добавление/удаление опций для тарифа;
- Управление опциями: некоторые опции могут быть несовместимы друг с другом или могут быть добавлены только с определенными опциями, сотрудник добавление и удаление этих правил.

При совершении действий с контрактами на каждой странице до сохранения изменений должна отображаться корзина, в которой отображаются выбранные позиции пользователя.

Технические требования

В итоге требуется получить многопользовательское приложение типа клиент-сервер с соединением по сети.

Все данные хранятся на стороне сервера. Каждый клиент может загружать некоторые данные, после каждой операции изменения данные должны быть синхронизованы с сервером.

Клиент должен иметь графический интерфейс.

Приложение должно обрабатывать аппаратные и программные ошибки.

Используемые технологии:

1 часть

- IDE - Any (Eclipse, IDEA)
- Tomcat 8.0.x
- Servlet/JSP
- DB – MySQL 5.6
- Maven 3.*
- JPA 2.0

2 часть (включая предыдущие)

- AS – Glassfish 4.0/WildFly 8.x
- EJB 3.x
- JSF 2.x
- Spring Framework

Критерии успешного выполнения

1. Функциональность работает (обязательно наличие UI)

2. Maven-based проект, разбитый на модули (билд одной командой, деплой одной командой)
3. Описаны интерфейсы предметной области
4. Подключена БД MySQL
5. Созданы сущности предметной области; маппинг на таблицы в БД
6. Работа с сущностями через DAO
7. Приложение развернуто на AS Glassfish/Wildfly* (актуально для 2й части)
8. Реализована обработка исключений
9. Подключено логгирование
10. Наличие technical solution description
11. Наличие unit-тестов на бизнес логику

Подключение любого фреймворка/библиотеки принимаются в зачет только при условии выполнения пунктов, описанных выше.

Дополнительные требования

!Относится ко второй части обучения!

В задании требуется написать отдельное приложение, которое будет показывать отчет обо всех абонентах сети с выбранным тарифным планом.

1. Приложение должно связываться с основным приложением через веб-сервис.
2. Технологии приложения – EJB (или CDI), JSF.
3. Отчет должен генерироваться в PDF.

Плюсом будет использование Selenium для автоматизации тестирования приложения и/или наличие «killer features».