Итоговое задание

В задании требуется написать приложение, моделирующее работу информационной системы некоторой компании, осуществляющей пассажирские железнодорожные перевозки. Ниже более подробно описаны предметная область и технические требования.

# Предметная область

Существуют следующие виды сущностей:

* Станция
  + Название
  + Расписание
    - Номер поезда
    - Время
* Поезд
  + Номер
  + Станции следования
  + Количество мест
* Пассажир
  + Имя
  + Фамилия
  + Дата рождения
* Билет
  + Номер поезда
  + Пассажир

Приложение должно обеспечивать следующую функциональность:

* Для клиентов компании
  + поиск поезда, проходящего от станции A до станции B в заданный промежуток времени;
  + расписание поездов по станции;
  + покупка билета, если
    - есть свободные места
    - пассажир с такими же именем, фамилией и датой рождений еще не зарегистрирован в выбранном поезде
    - до отправления поезда не менее 10 минут
* Для сотрудников компании
  + добавление новых станций, поездов;
  + просмотр всех зарегистрированных на поезд пассажиров;
  + просмотр всех поездов;

# 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.

# II часть

Реализовать отдельное клиент-приложение для электронного табло. Приложение должно отображать список всех поездов, отбывающих или прибывающих на станцию в текущий день. Данные должны подгружаться при старте и храниться на стороне клиента. Перезагрузка данных осуществляется в случае получения уведомления от сервера об изменениях в расписании (или задержке/отмене поезда).

# 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

8. Реализована обработка исключений

9. Подключено логгирование

10. Наличие technical solution description

11. Наличие unit-тестов на бизнес логику

# Используемые технологии

В рамках школы студенты не ограничены в выборе технологий для реализации проекта в рамках стека технологий, используемых в компании в различных проектах (лекции покрывают небольшую часть из них). Список технологий для выбора приведен ниже. Однако проект должен быть написан преимущественно на java (допускается написание незначительных частей, отдельных модулей или дополнительной функциональности на других языках, например, Kotlin или Python). После обсуждения с куратором и по согласованию с ним студент может выбрать любые комбинации технологий и фреймворков, но стоит учитывать, что куратор может быть незнаком с какими-то из них.

Список технологий, используемых в некоторых проектах компании:

Spring Core,Spring Data,AOP,Spring Boot,Spring Security,MVC,Spring Cloud, Microservices, Data processing(Spark, Apache Flume), CDI/EJB, Apache CXF, RedHat Fuse, Oracle eCommerce (ATG, Endeca...), JBoss, Apache Tomcat, Embedded Tomcat H2 (in-memory), Websphere Application Server 9.0, React, Angular/NPM/Webpack, typescript, JSF/ExtJs, JSP/Servlets, JavaScript / TypeScript / HTML / Sass / CSS/LESS/jQuery/GraphQL, SOAP/REST, JDBC, XML/XSLT/XSD,Bash scripting, Oracle Database, Gradle, Git, Maven, Subversion, Jenkins, Grafana, GitlabCI, Sonar, UNIX shell, Bootstrap, Formbased authorisation/JWT, Drools, ARS, DOM, PL SQL, Oracle AQ, IBM MQ, JMS, OracleDB ,Apache Camel, Docker, Microsoft T-Sql, WebDriver (Java), Oracle ATG, python, mapstruct, Lombok, Groovy, openshift, EC2, S3, Mongo, Casandra, hazelcast ,Junit, testing, Selenium, Mockito, Selenid, Prometheus, Kubernetes , Helm, Kibana, AWS, Apache Camel, Go, K8s, ETCD, PostgreSql, Wicket, blockchain, Web Flux, JPA/Hibernate

Задание состоит из 2 частей, каждая часть должна быть реализована с использованием разных неповторяющихся технологий.

Предлагаемый вариант стека технологий для 1 реализации части на основании лекций:

* IDE - Any (Eclipse, IDEA)
* Tomcat
* DB – MySQL
* Maven
* JPA
* Spring Framework (кроме Boot, Data)
* JSP

Для 2 части:

* Maven
* AS – WildFly
* EJB
* JSF
* MQ (для уведомлений от сервера)
* WebServices (для обмена данных между клиентом и сервером)

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

Рекомендуем использовать последнюю стабильную версию технологий, обязательных к использованию.

Плюсом будет использование следующих технологий: Selenium, Sonar, Angular/React (например, админка для табло), Docker, Microservices, использования доступного cloud для развертывания приложения (например, <https://cloud.yandex.ru/> - 1 месяц бесплатного использования) и/или наличие «killer features».