Тестовое‌ ‌задание‌‌

Постановка задачи

Требуется разработать веб-приложение, представляющее собой упрощенную систему электронного документооборота. Приложение должно реализовывать основные CRUD-операции:

1. Создание документа
2. Редактирование документа
3. Просмотр документа
4. Просмотр списка документов с пагинацией
5. Атрибутивный поиск документов
6. Удаление документа

В‌ ‌системе‌ ‌должно‌ ‌быть‌ ‌реализовано‌ ‌управление‌ ‌следующими‌ ‌типами‌ ‌записей: ‌ ‌

1. Организация, ‌ ‌содержит‌ ‌поля: ‌ ‌
   1. наименование‌ ‌организации; ‌ ‌
   2. физический‌ ‌адрес; ‌ ‌
   3. юридический‌ ‌адрес; ‌ ‌
   4. руководитель. ‌ ‌
2. Подразделение, ‌ ‌содержит‌ ‌поля: ‌ ‌
   1. наименование‌ ‌подразделения; ‌ ‌
   2. контактные‌ ‌данные; ‌ ‌
   3. руководитель. ‌ ‌
3. Сотрудник, ‌ ‌содержит‌ ‌поля: ‌
   1. фамилия;
   2. имя;
   3. отчество;
   4. должность. ‌ ‌
4. Поручение: ‌ ‌
   1. предмет‌ ‌поручения;
   2. автор‌ ‌поручения;
   3. исполнители‌ ‌поручения;
   4. срок‌ ‌исполнения; ‌
   5. признак‌ ‌контрольности;
   6. признак‌ ‌исполнения; ‌ ‌
   7. текст‌ ‌поручения.

Клиентская часть‌ ‌должна‌ ‌быть‌ ‌реализована‌ ‌в‌ ‌виде‌ ‌одностраничного‌ ‌приложения‌ ‌и‌ ‌взаимодействовать‌ ‌с‌ ‌серверной‌ ‌частью‌ ‌через‌ ‌ajax‌ ‌в‌ ‌асинхронном‌ ‌режиме. ‌ ‌Также‌ ‌должна‌ ‌быть‌ ‌возможность‌ ‌локализации‌ клиентской‌ ‌части‌ ‌приложения.

Серверная‌ ‌часть‌ ‌должна‌ ‌быть‌ ‌реализована‌ ‌в‌ ‌виде‌ ‌web-‌приложения‌ ‌с‌ ‌разделением‌ ‌на‌ ‌следующие‌ ‌слои: ‌ ‌

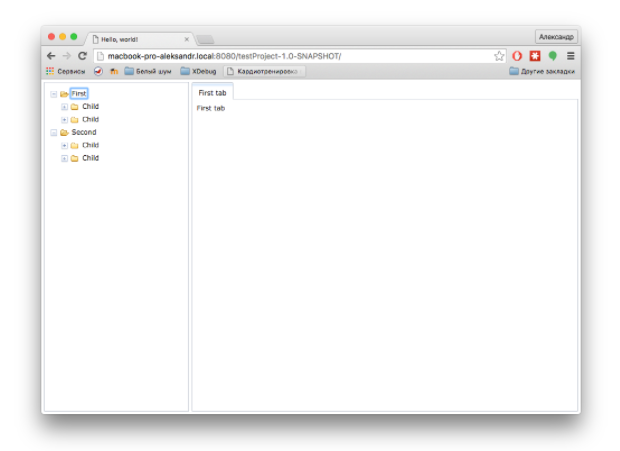
1. Слой‌ ‌взаимодействия‌ ‌с‌ ‌пользовательским‌ ‌интерфейсом‌ ‌-‌ ‌реализовать‌ ‌с‌ ‌использованием‌ Spring-Web в виде ‌REST-сервисов‌;
2. Слой‌ ‌сервисных‌ ‌объектов‌ ‌-‌ ‌реализуют‌ ‌бизнес-логику‌ ‌приложения. ‌ ‌Реализовать‌ ‌с‌ ‌использованием бинов‌;
3. Слой‌ ‌доступа‌ ‌к‌ ‌данным‌ ‌-‌ ‌реализует‌ ‌логику‌ ‌доступа‌ ‌к‌ ‌хранилищу‌ ‌данных. ‌ ‌Реализовать‌ с использованием JPA Spring Data;
4. Конфигурация‌ ‌приложения‌ ‌хранится‌ ‌в‌ ‌отдельных‌ ‌файлах‌‌;
5. Управление‌ ‌состояниями‌ ‌документов‌ ‌реализуется‌ ‌через‌ ‌машину‌ ‌состояний‌‌;
6. Готовое‌ ‌приложение‌ ‌должно‌ ‌представлять‌ Spring Boot приложение, собранное в jar;
7. Зависимости должны‌ ‌быть‌ ‌также‌ ‌запакованы‌ ‌в‌ ‌jar-файл.

Рекомендуемые‌ ‌технологии‌ ‌и‌ ‌средства‌ ‌‌ ‌

1. Apache‌ ‌Maven‌ ‌-‌ ‌управление‌ ‌зависимостями, ‌ ‌упаковка‌ ‌приложения; ‌ ‌
2. Postgresql;
3. slf4j‌ ‌-‌ ‌логгирование;
4. AngularJs или Angular 8(TypeScript).
5. Стек технологий Spring и Spring Boot

Пользовательский‌ ‌интерфейс‌ ‌

После‌ ‌загрузки‌ ‌приложения‌ ‌пользователь‌ ‌попадает‌ ‌на‌ ‌страницу‌ ‌с‌ ‌перечнем‌ ‌записей. ‌ ‌ ‌

 ‌

В‌ ‌левой‌ ‌части‌ ‌страницы‌ ‌отображается‌ ‌категоризированный‌ ‌перечень‌ ‌представлений: ‌ ‌

1. Справочник‌ ‌организации‌ ‌
   1. Оргструктура‌ ‌
   2. Все‌ ‌сотрудники‌ ‌
2. Поручения‌ ‌
   1. Все‌ ‌поручения‌ ‌
   2. Мои‌ ‌поручения‌ ‌
   3. Поручения‌ ‌мне‌ ‌

При‌ ‌нажатии‌ ‌на‌ ‌представление‌ ‌в‌ ‌правой‌ ‌части‌ ‌создается‌ ‌новая‌ ‌закладка, ‌ ‌на‌ ‌которой‌ ‌отображается‌ ‌список‌ ‌записей‌ ‌из‌ ‌этого‌ ‌представления. ‌ ‌Представление‌ ‌должно‌ ‌обеспечивать‌ ‌автоматическую‌ ‌загрузку‌ ‌данных‌ ‌при‌ ‌прокручивании. ‌ ‌Над‌ ‌представлением‌ ‌должна‌ ‌располагаться‌ ‌панель‌ ‌действий‌ ‌из‌ ‌которой‌ ‌пользователь‌ ‌может‌ ‌создать‌ ‌новую‌ ‌запись‌ ‌или‌ ‌изменить‌ ‌выбранную. ‌ ‌При‌ ‌двойном‌ ‌щелчке‌ ‌на‌ ‌записи‌ ‌открывается‌ ‌новая‌ ‌закладка, ‌ ‌на‌ ‌которой‌ ‌отображается‌ ‌форма‌ ‌редактирования. ‌ ‌Вверху‌ ‌формы‌ ‌располагаются‌ ‌кнопки‌ ‌закрытия, ‌ ‌сохранения и‌ ‌дополнительных‌ ‌действия, ‌ ‌в зависимости‌ ‌от‌ ‌типа‌ ‌записи. ‌ ‌

Добавление‌ ‌записи‌ ‌

В‌ ‌представлении‌ ‌доступна‌ ‌кнопка‌ ‌создания‌ ‌новой‌ ‌записи. ‌ ‌При‌ ‌ее‌ ‌нажатии‌ ‌создается‌ ‌новая‌ ‌закладка‌ ‌с‌ ‌формой. ‌ ‌Перед‌ ‌сохранением‌ ‌формы‌ ‌необходимо‌ ‌выполнить‌ ‌валидацию‌ ‌данных, например, ‌ ‌проверить‌ заполнение ‌обязательных‌ ‌полей. ‌ ‌ ‌После‌ ‌выполнения‌ ‌клиентской‌ ‌валидации‌ ‌выполняется‌ ‌серверная. ‌ ‌В‌ ‌случае‌ ‌успешной‌ ‌валидации‌ ‌происходит‌ ‌сохранение‌ ‌данных‌ ‌в‌ ‌БД. ‌ ‌

Редактирование‌ ‌записи‌ ‌

В‌ ‌представлении‌ ‌при‌ ‌выборе‌ ‌записи‌ ‌доступна‌ ‌кнопка‌ ‌редактирования. ‌ ‌При‌ ‌ее‌ ‌нажатии‌ ‌создается‌ ‌новая закладка‌ ‌с‌ ‌формой‌ ‌редактирования‌ ‌выбранной‌ ‌записи.‌ ‌Также‌ ‌запись‌ ‌должна‌ ‌открываться‌ ‌на‌ ‌редактирование при‌ ‌двойном‌ ‌щелчке‌ ‌в‌ ‌представлении.‌ ‌ ‌Перед‌ ‌сохранением‌ ‌формы‌ ‌необходимо‌ ‌выполнить‌ ‌валидацию‌ ‌данных, например,‌ ‌проверить‌ заполнение ‌обязательных‌ ‌полей.‌ ‌ ‌После‌ ‌выполнения‌ ‌клиентской‌ ‌валидации выполняется‌ ‌серверная.‌ ‌В‌ ‌случае‌ ‌успешной‌ ‌валидации‌ ‌происходит‌ ‌сохранение‌ ‌данных‌ ‌в‌ ‌БД.‌ ‌

Удаление‌ ‌записи‌ ‌

В‌ ‌представлении‌ ‌при‌ ‌выборе‌ ‌записи‌ ‌доступна‌ ‌кнопка‌ ‌удаления.‌ ‌При‌ ‌ее‌ ‌нажатии‌ ‌пользователю‌ ‌должно‌ ‌выдаваться‌ ‌диалоговое‌ ‌окно‌ ‌с‌ ‌запросом‌ ‌подтверждения‌ ‌операции.‌

Атрибутивный‌ ‌поиск‌ ‌

Во‌ ‌всех‌ ‌представлениях‌ ‌должна‌ ‌быть‌ ‌возможность‌ ‌сортировки‌ ‌записей‌ ‌при‌ ‌щелчке‌ ‌на‌ ‌столбец‌ ‌и‌ ‌атрибутивный‌ ‌поиск‌ ‌по‌ ‌отображаемым‌ ‌атрибутам.‌ ‌Поиск‌ ‌должен‌ ‌выполняться‌ ‌на‌ ‌сервере‌ ‌и‌ ‌поддерживать‌ ‌динамическую‌ ‌загрузку‌ ‌данных.‌ ‌

Жизненный‌ ‌цикл‌ ‌документа‌ ‌“Поручение” ‌ ‌

Жизненный‌ ‌цикл‌ ‌документа‌ ‌“Поручение” ‌ ‌рекомендуется‌ ‌реализоваться‌ ‌с‌ ‌использованием‌ ‌машины‌ ‌состояний. ‌ ‌ ‌

