

Университет ИТМО

Факультет программной инженерии и компьютерной техники
Направление подготовки 09.03.04 Программная инженерия
Дисциплина «Методы и средства программной инженерии»

Отчет

По лабораторной работе №1
Вариант 1316

Выполнили:

Колесникова С.А.

Степанов М.А.

Преподаватель:

Каюков И. А.

Санкт-Петербург, 2021 г.

Задание:

Вариант №1316:

Rap.ru - сайт о рэп-музыке и хип-хоп культуре. Ресурс о рэп-музыке, хип-хоп культуре, R&B, брейкдансе, диджейском искусстве. Новости, события, видео клипы, интервью, биографии музыкантов, форум - <http://www.rap.ru>

Составить список требований, предъявляемых к разрабатываемому веб-сайту (в соответствии с вариантом). Требования должны делиться на следующие категории:

- *Функциональные.*
 - *Требования пользователей сайта.*
 - *Требования владельцев сайта.*
- *Нефункциональные.*

Требования необходимо оформить в соответствии с шаблонами RUP (документ SRS - Software Requirements Specification). Для каждого из требований нужно указать его атрибуты (в соответствии с методологией RUP), а также оценить и аргументировать приблизительное количество часов, требующихся на реализацию этого требования.

Для функциональных требований нужно составить UML UseCase-диаграммы, описывающие реализующие их прецеденты использования.

Отчёт по лабораторной работе должен содержать:

1. Документ Software Requirements Specification, содержащий список требований к сайту.
2. UseCase-диаграммы прецедентов использования, реализующих функциональные требования.
3. Выводы по работе.

Вопросы к защите лабораторной работы:

1. Методологии разработки ПО. Унифицированный процесс.
2. Требования и их категоризация. Атрибуты требований.
3. Язык UML.
4. Прецеденты использования. UseCase-диаграммы - состав, виды связей.

Software Requirements Specification

1. Introduction

1.1 Purpose

Rap.ru - сайт о рэп-музыке и хип-хоп культуре. Ресурс о рэп-музыке, хип-хоп культуре, R&B, брейкдансе, диджейском искусстве. Новости, события, видео клипы, интервью, биографии музыкантов, форум - <http://www.rap.ru>

1.2 Document conventions

Java – язык программирования

PostgreSQL – объектно-реляционная СУБД

Intelij IDEA – среда разработки, предназначенная для языка Java.

Wildfly – сервер приложений.

Фронтенд – Визуальная составляющая веб-приложения.

Бэкенд – функциональная составляющая веб-приложения.

Html - язык разметки сайта

CSS - набор стилей примененный на сайте

JS - язык программирования (Vue.js – его фреймворк)

JSON – текстовый формат представления данных для передачи.

1.3 References

Use-Case: <https://app.creately.com/diagram/start/dashboard>

Информация о SRS: <https://habr.com/ru/post/52681/>

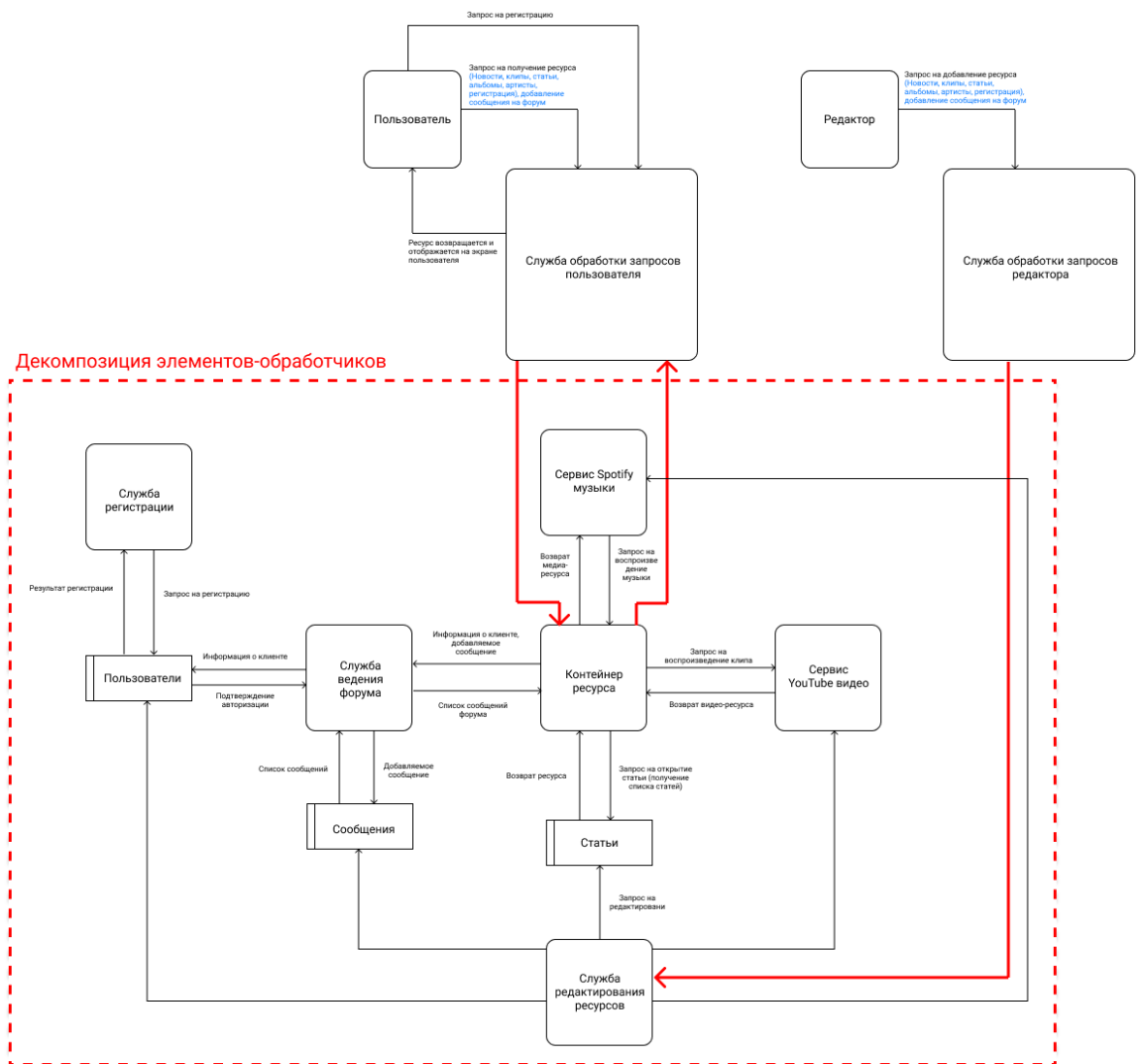
Rap.ru: <https://www.rap.ru/>

Информация о RUP: <https://qaevolution.ru/metodologiya-menedzhment/rup/>

Кому писать в экстренной ситуации: editor@rap.info.ru

2. Overall Description

2.1 Project Features



2.2 Operating environment

Java (JDK 11) EE (компилятор Amazon), PostgreSQL, сервер Wildfly, IntelliJ IDEA

2.3 Design and implementation constraints

- 2.3.1 Для фронтенда используются такие технологии как: html, css, js, vue.js,
- 2.3.2 Для бэкенда используется EJB, Hibernate, Thymeleaf

2.4 User documentation

Азбука, книга по интернет-сёрфингу (версус хованского и славы КПСС, рифмы и панчи)

3. Specific requirements

3.1 System feature 1

3.1.1 Resource name

server.resource.article.*

3.1.2 Description and priority

Предоставляет доступ пользователя к данным из системы Баз данных для чтения статей, созданных редактором. Предоставляется неавторизованным пользователям.

3.1.3 Stimulus/Response sequence

При запуске по ссылке из верхнего меню открывает все доступные статьи в виде упорядоченного по дате добавления статей списке. При нажатии на блок (элемент) с описание статьи осуществляется переход на html-страницу с содержанием статьи.

3.1.4 Functional requirements

Предоставляет возможность пользователю просматривать ленту новостей, сортировать новости по тегам, ленту со статьями, список новостей.

При поступлении запроса на контроллер передаёт запрос контейнеру запроса, который получается данные из бд, формирует JSON-файл со всеми необходимыми полями для описания статьи, запускает шаблонизатор Thymeleaf, формирую ответ в виде html-страницы, отправляет её пользователю.

3.2 System feature 2

3.2.1 Resource name

server.resource.youtube.*

3.2.2 Description and priority

Предоставляет доступ пользователя к видеоресурсам, находящимся на удалённом хостинге youtube. Предоставляется неавторизованным пользователям.

3.2.3 Stimulus/Response sequence

При запуске по ссылке из верхнего меню открывает все доступные видео-ресурсы в виде упорядоченного по дате добавления списке. При нажатии на блок (элемент) с описание видео осуществляется переход на html-страницу с описание видео и плеером для воспроизведения видео с хостинга youtube.

3.2.4 Functional requirements

Предоставляет возможность пользователю просматривать видеоматериалы.

При поступлении запроса на контроллер передаёт запрос контейнеру запроса, который получается данные с видеохостинга youtube, формирует JSON-файл со всеми необходимыми полями для описания видео и ссылкой на видео ресурс, запускает шаблонизатор Thymeleaf,

формирую ответ в виде html-страницы, создаёт запрос к видеоплееру, передаёт ссылку на запрашиваемое видео, отправляет страницу пользователю, на стороне пользователя формируется окончательная страница с содержанием видео и плеером.

3.3 System feature 3

3.3.1 Resource name

server.resource.spotify.*

3.3.2 Description and priority

Предоставляет доступ пользователя к музыкальным ресурсам, находящимся на удалённом сервисе Spotify. Предоставляется неавторизованным пользователям.

3.3.3 Stimulus/Response sequence

При запуске по ссылке из верхнего меню открывает все доступные музыкальные ресурсы (их описания) в виде упорядоченного по дате добавления списке. При нажатии на блок (элемент) с описанием аудиофайла (альбома) осуществляется переход на html-страницу с описанием альбома и плеером для воспроизведения музыки.

3.3.4 Functional requirements

Предоставляет возможность пользователю прослушивать аудиоматериалы.

При поступлении запроса на контроллер передаёт запрос контейнеру запроса, который получает данные с сервиса Spotify, формирует JSON-файл со всеми необходимыми полями для описания альбома и файлами аудиозаписей, запускает шаблонизатор Thymeleaf, формирую ответ в виде html-страницы, отправляет аудиофайл (в виде отдельных пакетов), отправляет страницу пользователю, на стороне пользователя формируется окончательная страница с аудиоплеером.

3.4 System feature 4

3.4.1 System feature 4

server.resource.forum.*

3.4.2 Description and priority

Предоставляет доступ пользователя к форуму, позволяющему общаться с другими пользователями. Предоставляется только авторизованным пользователям.

3.4.3 Stimulus/Response sequence

При запуске по ссылке из верхнего меню открывает страницу с окном авторизации. После авторизации при вводе внизу страниц ресурсов в текстовое поле ввода и нажатии на кнопку отправить, сообщение добавляется в обсуждение ресурса.

3.4.4 Functional requirements

Предоставляет возможность пользователю обмениваться сообщениями с пользователями на форуме.

При поступлении запроса на контроллер передаёт запрос контейнеру запроса, который перенаправляет запрос на службу форума.

Проверяется авторизация пользователя, при подтверждении – открывается доступ к комментированию ресурса. При получении сообщения служба форума проверяет авторизацию и добавляет сообщение к объекту ресурса.

3.5 External interface requirements

3.5.1 Приложение взаимодействует с youtube, при помощи youtube data API.

3.5.2 Приложение взаимодействует с spotify, при помощи spotify web API.

3.6 Non functional requirements

3.6.1 Корректное отображение сайта на мобильных устройствах и планшетах.

3.6.2 Передача информации об авторизации пользователя должна храниться в HTTP-заголовках.

3.6.3 Корректное отображение сайта на современных версиях браузеров: Chrome 80, Safari 11, Mozilla 70, ЯндексБраузер 21.

3.6.4 Медиа-ресурсы не должны кэшироваться на стороне пользователя.

3.6.5 Добавить баннер с запросом на показ рекламы.

3.7 Performance requirements

3.7.1 База данных должна выдерживать 1000 запросов в минуту.

3.7.2 Вёрстка сайта должна отображаться только после загрузки css-файла.

3.8 Software quality attributes

3.8.1 Код должен быть отформатирован в соответствии с нормами форматирования IntelliJ IDEA.

3.8.2 Наименование переменных пишется в соответствии с camelCase, должны отобразить смысл переменной.

3.8.3 Используются автотесты для тестирования соединения, авторизации, получения ресурсов.

3.9 Security requirements

3.9.1 Экранирование запросов к базе данных.

3.9.2 Редактирование сайта разрешено пользователям с ролью «редактор» или выше.

3.10 Supportability requirements:

3.10.1 Сайт должен иметь возможность расширяться горизонтально за счёт подключения дополнительных вычислительных устройств к серверу (на случай, если все внезапно захотят читать новости про реп).