

**Федеральное государственное автономное образовательное
учреждение высшего образования
Национальный исследовательский
университет ИТМО**

Факультет программной инженерии и компьютерной техники

Направление подготовки:

Системное и Прикладное Программное Обеспечение

(09.03.04 Программная инженерия)

Дисциплина «Основы программной инженерии»

Отчет

По лабораторной работе №1

Вариант №1559

Студент:

**Карташев Владимир Сергеевич,
группа Р3215**

Практик:

Цопа Евгений Алексеевич

г. Санкт-Петербург, 2024 г.

Оглавление

Задание

Выполнение

~Introduction введение

Purpose цель

Document conventions описание технического сленга

Intended Audience and Reading Suggestions целевая аудитория

Project scope масштаб проекта

References справки / ссылки

~Overall Description общее описание

Product features особенности продукта

Operating environment рабочая среда

~Functional requirements функциональные требования

~Non functional requirements нефункциональные требования

Usability requirements требование к удобству использования

Performance requirements требование к производительности

Security & safety requirements требование безопасности & сохранности

Software quality attributes атрибуты качества программного обеспечения

~External interface requirements требования к внешнему интерфейсу

~UseCase-диаграмма

~Пример интерфейса

Прецедент 1: поиск нужного порновидео

Прецедент 2: загрузка своего порновидео

Вывод

Задание

Вариант №1559: xnxx.com: xnxx delivers free sex movies and fast free porn videos (tube porn). now 10 million+ sex vids available for free! featuring hot pussy, sexy girls in xxx rated porn clips - <https://xnxx.com/>

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.

Выполнение

~Introduction введение

Purpose цель

XNXX.com — это синий порносайт с огромным количеством извращенного контента, как по количеству видео, доступных в их архивах, так и по хорошо освещенным категориям.

Document conventions описание технического сленга

Бекенд - это часть программного обеспечения, работающая на серверной стороне и обрабатывающая логику приложения, взаимодействуя с базой данных и обеспечивая данные для фронтенда.

Библиотека - коллекция функционала, добавляемая к готовому проекту для решения новых задач.

Фреймворк - набор инструментов, предоставляющих готовую архитектуру для построения приложений.

Фронтенд - это часть программного обеспечения, отвечающая за пользовательский интерфейс и взаимодействие с пользователем.

CSS - каскадные таблицы стилей, используемые для оформления и внешнего вида веб-страниц, созданных с помощью HTML.

Data Flow Diagram - графическое представление потоков данных

HTML - язык разметки, используемый для создания структуры веб-страниц.

JS - язык программирования, используемый для добавления интерактивности на веб-сайты.

Kotlin - современный язык программирования, используемый в основном для создания мобильных и веб-приложений.

Ktor - инструмент, используемый для написания серверных приложений.

MinIO - хранилище данных.

PostgreSQL - система управления базами данных.

Software Requirements Specification - это документ, который описывает функциональные и нефункциональные требования к программному обеспечению.

UseCase - сценарий использования.

Intended Audience and Reading Suggestions целевая аудитория

Студенты 1-4 курса бакалавриата технических ВУЗов.

Project scope масштаб проекта

Малый масштаб проекта: "...к малым проектам относятся проекты стоимостью до 10 млн. амер. долларов...".

References справки / ссылки

Use Case (сценарий использования) - <https://habr.com/ru/articles/699522/>

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

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

Классификация проектов - <https://clck.ru/38qJ3n>

DFD диаграммы - <https://habr.com/ru/articles/668684/>

xnxx.com - <https://www.xnxx.com/>

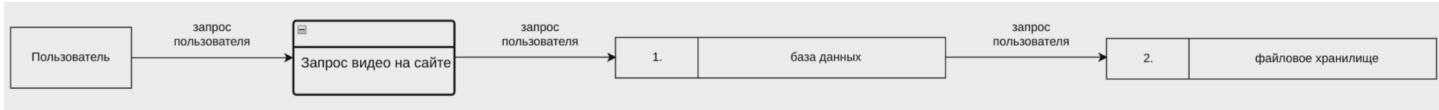
Форум xnxx.com - <https://forum.xnxx.com/>

Прецеденты - <https://clck.ru/38qekT>

~Overall Description общее описание

Product features особенности продукта

Концептуальный уровень DFD диаграммы:



Operating environment рабочая среда

- Для фронтенда используются такие технологии как: HTML, CSS, JS, VUE
- Для бекенда используется Kotlin, Ktor, PostgreSQL, MinIO

~Functional requirements функциональные требования

Для пользователя:

№	Требование
U1	Предоставлять возможность выбрать один из четырех языков: русский, английский, немецкий, Java.
U2	Предоставлять возможность находить контент с помощью поиска по различным критериям: категория, жанр, актеры.
U3	Предоставлять возможность просмотра популярных видео в зависимости от географического положения: Россия, Германия, Средняя Азия, Япония и т.д.
U4	Предоставлять возможность быстрого поиска интересующего контента: Лучшее из, Хиты, Ключевые слова, Картинки, Камеры в прямом эфире, Секс истории, Форум, Порнозвезды, Игры.
U5	Предоставлять возможность просматривать ленту, конфигурируемую по предпочтениям (предпочтения сохраняются в сессии на основе тегов просмотренных видео при помощи логического И).
U6	Предоставлять возможность оставлять комментарии и ставить оценки видеороликам, чтобы обменяться мнениями и помочь другим пользователям в выборе контента.
U7	Предоставлять возможность подписываться на определенные категории контента или на конкретных актеров и получать уведомления о новых материалах.
U8	Предоставлять возможность загрузить свое видео.
U9	Предоставлять возможность поделиться видео с друзьями и коллегами.
U10	Предоставлять возможность создание личного профиля.
U11	Глухонемые пользователи должны иметь возможность включать субтитры.

Для администратора / владельца:

№	Требование
O1	Администраторы должны иметь возможность добавлять, редактировать и удалять контент на сайте, а также управлять категориями и тегами для лучшей организации контента.
O2	Администраторы должны иметь возможность управлять пользователями, включая блокировку или удаление учетных записей, установку ограничений доступа и т.д.
O3	Владельцы могут использовать стандартные методы монетизации: реклама (баннерная, видео, перенаправляющая на другой сайт), платная подписка, продажа контента и управлять ими.
O4	Владельцы должны иметь возможность отслеживать статистику посещаемости, поведения пользователей, популярности контента и другие метрики с помощью аналитических инструментов и генерировать соответствующие отчет.

~Non functional requirements нефункциональные требования

Usability requirements требование к удобству использования

N1	Отображение сайта с полностью работающим функционалом и без нарушения дизайна в следующих популярных браузерах: Chrome 121, Safari 16, Mozilla Firefox 122, Яндекс Браузер 23.
N2	Мобильная версия сайта должна корректно работать на современных разрешениях экранов, таких как: 1280x720, 1920x1080, 3840x2160.

Performance requirements требование к производительности

N3	Время загрузки главной страницы не должно превышать 5 секунд при скорости интернета не менее 20 Мбит/с.
N4	Сайт должен быть способен обрабатывать до 10000 одновременных пользовательских запросов.

Security & safety requirements требование безопасности & сохранности

N5	Контроль доступа к административным функциям сайта при помощи JWT и OAuth.
N6	Пароли пользователей должны храниться в зашифрованном виде с использованием современных хеш-функций, таких как SHA-256.
N7	Система должна иметь механизм защиты от SQL-инъекций и DDoS-атак.
N8	Резервное копирование данных должно выполняться ежедневно с возможностью быстрого восстановления.

Software quality attributes

атрибуты качества программного обеспечения

N9	Система должна вести логи ошибок и событий.
N10	Должна быть написана документация на английском языке ко всему программному обеспечению.
N11	Версии и зависимости используемых библиотек и фреймворков должны регулярно обновляться.

~External interface requirements

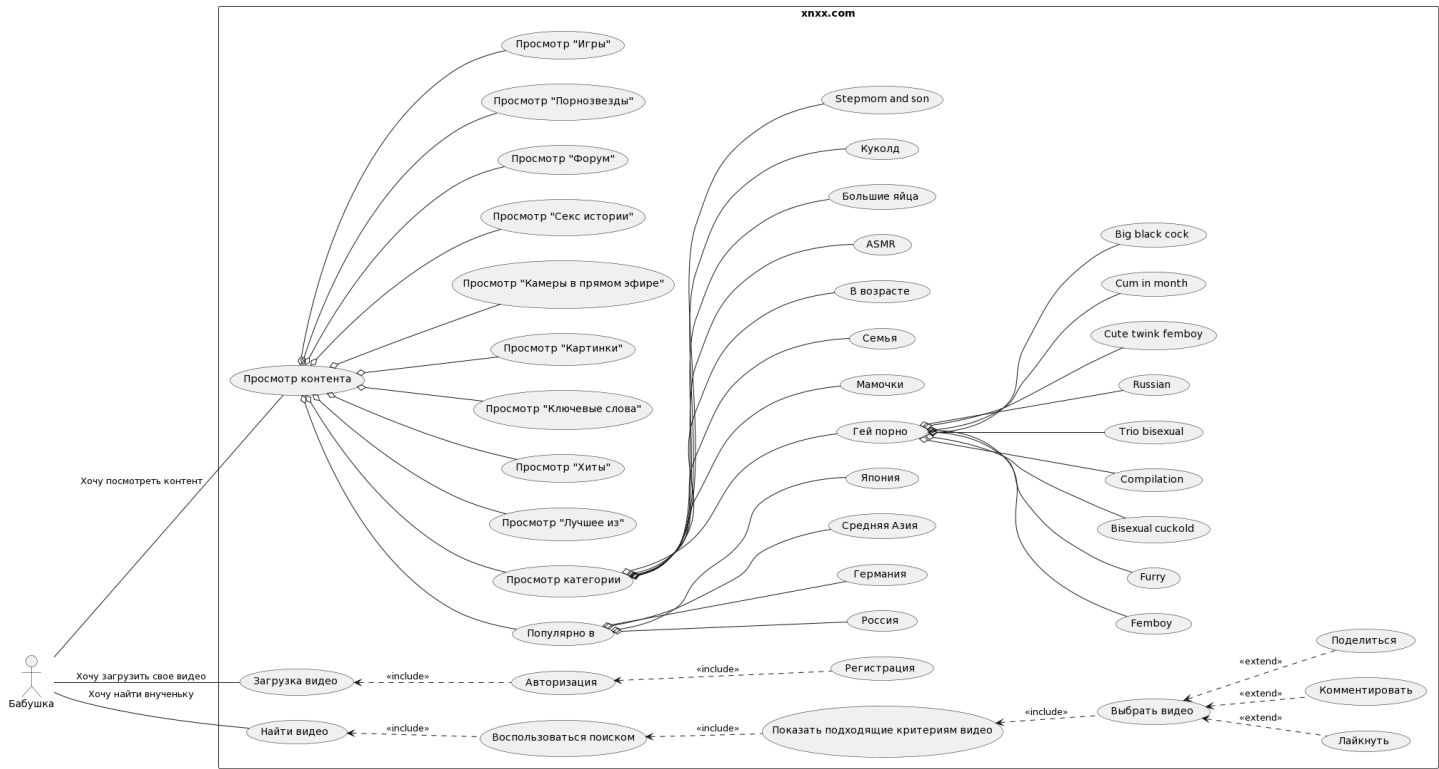
требования к внешнему интерфейсу

N12	Для взаимодействия фронтенда и бэкенда используется REST API.
N13	Ответы сервера предоставляются в формате JSON.

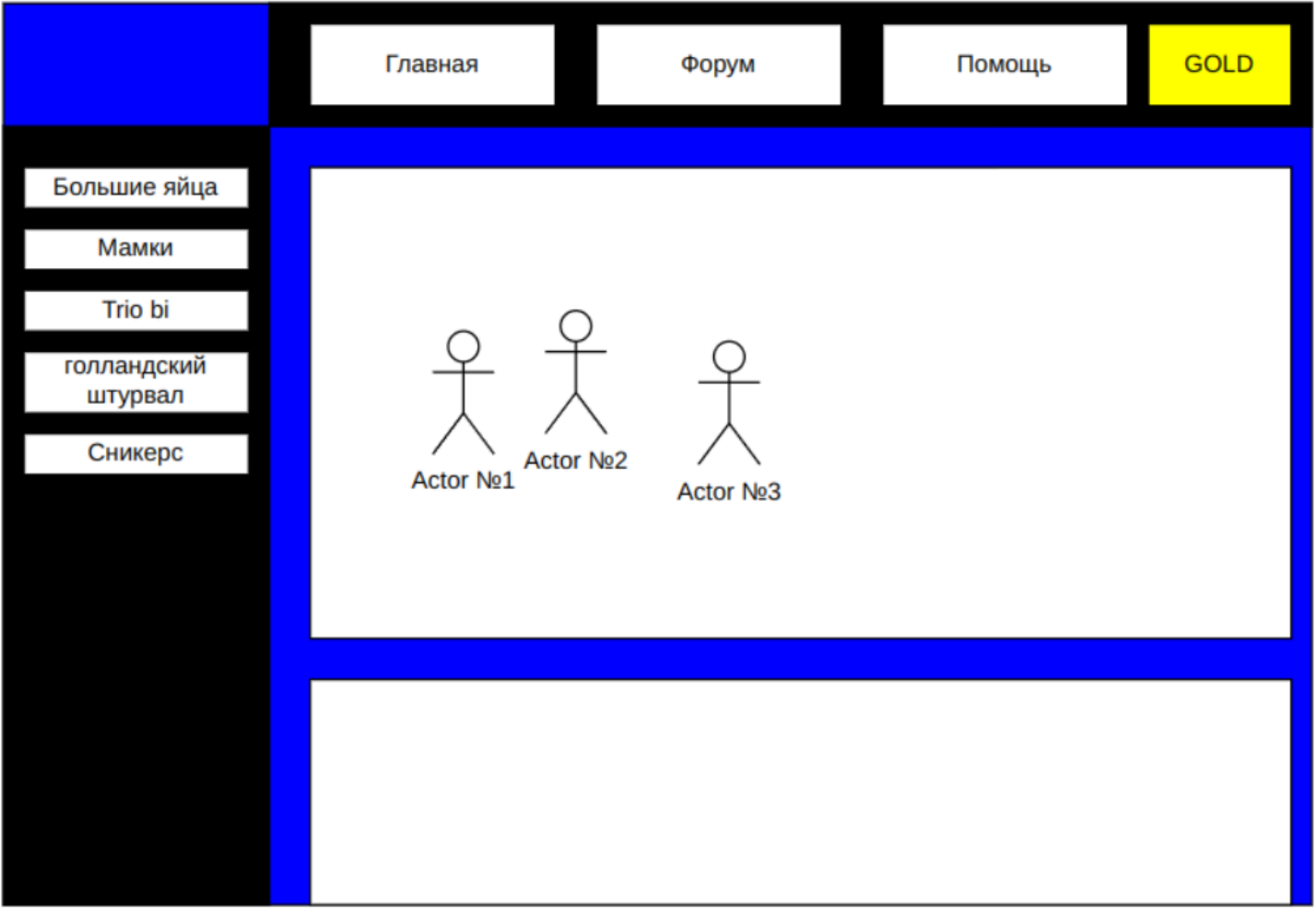
Номер требования	Статус	Кол-во часов	Стабильность
U1	Одобрено	6	Средняя
U2	Предложено	8	Средняя
U3	Одобрено	12	Низкая
U4	Предложено	6	Низкая
U5	Предложено	8	Высокая
U6	Отклонено	4	Высокая
U7	Предложено	2	Высокая
U8	Одобрено	16	Средняя
U9	Одобрено	16	Высокая
U10	Предложено	16	Высокая
U11	Предложено	24	Высокая
O1	Одобрено	12	Высокая
O2	Одобрено	12	Средняя
O3	Одобрено	12	Высокая
O4	Одобрено	21	Высокая
N1	Одобрено	8	Высокая
N2	Одобрено	8	Высокая
N3	Отклонено	16	Средняя
N4	Предложено	16	Низкая
N5	Предложено	6	Высокая

N6	Одобрено	4	Высокая
N7	Предложено	8	Низкая
N8	Отклонено	4	Низкая
N9	Предложено	2	Низкая
N10	Отклонено	12	Средняя
N11	Отклонено	1	Низкая
N12	Одобрено	10	Высокая
N13	Одобрено	10	Высокая

~UseCase-диаграмма



~Пример интерфейса



Прецедент 1: поиск нужного порновидео

Система	Порнохостинг https://www xnxx.com/
Основное действующее лицо	Пользователь
Цель	Поиск нужного видео
Триггер	Пользователь решает найти видео и вводит запрос в строку поиска
Результат	Результат поиска в виде выборки из превью видеороликов вместе с названием и количеством просмотров

Основной поток событий

№	Действующее лицо	Шаг
1	Система	Запрашивает ввод критериев поиска: текстовый запрос, категория, актеры
2	Пользователь	Вводит свой извращенский запрос
3	Система	Создает выборку ID видеороликов по приоритету: Актеры -> категория -> текстовый запрос
4	Система	Отправляет пользователь превью видеороликов вместе с названием, ID видео, количеством просмотров
5	Пользователь	Выбирает желаемое видео
6	Система	Открывает страницу просмотра с видео
7	Система	Загружает видео по его ID и добавляет +1 к числу просмотров
8	Пользователь	Просматривает видео
9	Пользователь	Лайкает / комментирует / делится / игнорит действие

Прецедент 2: загрузка своего порновидео

Система	Порнохостинг https://www xnxx.com/
Основное действующее лицо	Пользователь
Цель	Загрузка видео
Триггер	Пользователь решает загрузить свое видео
Результат	Видео загружено, теперь наш пользователь - порнозвезда

Основной поток событий

№	Действующее лицо	Шаг
1	Система	Запрашивает авторизацию пользователя
2	Пользователь	Авторизуется / Регистрируется
3	Система	Отправляет форму для названия и загрузки видеоролика
4	Пользователь	Придумывает название и загружает видео
6	Система	Присваивает видео ID
7	Система	Сохраняет видео с ID в названии в MinIO и добавляет информацию о нем в базу данных
8	Система	Отправляет HTTP/1.1 200 OK
9	Пользователь	*чувство выполненного долга*

Вывод

В ходе данной лабораторной работы мы ознакомились с методологией RUP и структурой SRS документа, была создана UML UseCase-диаграмма и составлен список требований к сайту, оформленный в виде документа SRS.

Для создания UML был использован сервис plantuml.com;

Исходный код UML диаграммы:

```
@startuml
left to right direction
skinparam packageStyle rectangle

actor Бабушка as User
rectangle "xnxx.com" {
    usecase "Просмотр контента" as ViewContent
    usecase "Популярно в" as PopularIn
    usecase "Просмотр категории" as WatchCategory
    usecase "Гей порно" as GayPorn

    usecase "Загрузка видео" as UploadVideo
    usecase "Авторизация" as Authenticate
    usecase "Регистрация" as Register

    usecase "Найти видео" as FindVideo

    User -- ViewContent : Хочу посмотреть контент

    ViewContent o-- (Просмотр "Лучшее из")
    ViewContent o-- (Просмотр "Хиты")
    ViewContent o-- (Просмотр "Ключевые слова")
    ViewContent o-- (Просмотр "Картинки")
    ViewContent o-- (Просмотр "Камеры в прямом эфире")
    ViewContent o-- (Просмотр "Секс истории")
    ViewContent o-- (Просмотр "Форум")
    ViewContent o-- (Просмотр "Порнозвезды")
    ViewContent o-- (Просмотр "Игры")

    ViewContent o-- PopularIn
    PopularIn o-- (Россия)
    PopularIn o-- (Германия)
    PopularIn o-- (Средняя Азия)
    PopularIn o-- (Япония)

    ViewContent o-- WatchCategory
    WatchCategory o-- (Мамочки)
    WatchCategory o-- (Семья)
    WatchCategory o-- (В возрасте)
    WatchCategory o-- (ASMR)
    WatchCategory o-- (Большие яйца)
    WatchCategory o-- (Куколд)
    WatchCategory o-- (Stepmom and son)
    WatchCategory o-- GayPorn

    GayPorn o-- (Femboy)
    GayPorn o-- (Furry)
    GayPorn o-- (Bisexual cuckold)
```

GayPorn o-- (Compilation)
GayPorn o-- (Trio bisexual)
GayPorn o-- (Russian)
GayPorn o-- (Cute twink femboy)
GayPorn o-- (Cum in month)
GayPorn o-- (Big black cock)

User -- UploadVideo : Хочу загрузить свое видео
UploadVideo <.. Authenticate : <<include>>
Authenticate <.. Register : <<include>>

User -- FindVideo : Хочу найти внученьку
FindVideo <.. (Воспользоваться поиском) : <<include>>
(Воспользоваться поиском) <.. (Показать подходящие критериям видео) : <<include>>
(Показать подходящие критериям видео) <.. (Выбрать видео) : <<include>>
(Выбрать видео) <.. (Лайкнуть) : <<extend>>
(Выбрать видео) <.. (Комментировать) : <<extend>>
(Выбрать видео) <.. (Поделиться) : <<extend>>

}
@enduml