

Университет ИТМО

Факультет программной инженерии и компьютерной техники

**Лабораторная работа № 1**  
по дисциплине «Методы и средства программной инженерии»

Выполнил:  
Студент группы Р32302  
Русинов Дмитрий Станиславович  
Преподаватель:  
Барсуков илья Александрович

Санкт-Петербург  
2023

# Содержание

<b>Задание</b>	<b>3</b>
<b>1 Introduction</b>	<b>3</b>
1.1 Purpose . . . . .	3
1.2 Scope . . . . .	4
1.3 Definitions, Acronyms and Abbreviations . . . . .	4
1.4 References . . . . .	4
1.5 Overview . . . . .	4
<b>2 Overall Description</b>	<b>5</b>
2.1 Product functions . . . . .	5
2.2 User characteristics . . . . .	5
2.3 Assumptions and dependencies . . . . .	5
2.4 Constrains . . . . .	5
<b>3 Specific Requirements</b>	<b>6</b>
3.1 Functionality . . . . .	6
3.1.1 Регистрация пользователя . . . . .	6
3.1.2 Поиск изображения для пазла . . . . .	6
3.1.3 Решение пазла . . . . .	6
3.1.4 Заработок виртуальных кредитов . . . . .	6
3.1.5 Система комментариев . . . . .	6
3.1.6 Новостная лента . . . . .	6
3.1.7 Покупка VIP-статуса . . . . .	6
3.2 Usability . . . . .	7
3.2.1 Обучаемость . . . . .	7
3.2.2 Интуитивность интерфейса . . . . .	7
3.2.3 Оформление сайта . . . . .	7
3.2.4 Понятность текстов . . . . .	7
3.2.5 Адаптивность сайта . . . . .	7
3.3 Reliability . . . . .	7
3.3.1 Uptime . . . . .	7
3.3.2 Backup . . . . .	7
3.4 Performance . . . . .	7
3.4.1 Количество одновременных пользователей . . . . .	7
3.4.2 Отклик . . . . .	8
3.4.3 Обработка большого количества одновременных запросов . . . . .	8
3.5 Design constraints . . . . .	8
3.5.1 Используемые технологии . . . . .	8
3.5.2 Использование сторонних сервисов . . . . .	8
3.6 Interfaces . . . . .	8
3.6.1 User interfaces . . . . .	8
3.6.2 Hardware Interfaces . . . . .	8
3.6.3 Software Interfaces . . . . .	8
3.6.4 Communication Interfaces . . . . .	9
3.7 Licensing Requirements . . . . .	9

<b>4</b>	<b>Атрибуты требований</b>	<b>9</b>
4.1	Функциональные требования . . . . .	9
4.2	Нефункциональные требования . . . . .	9
<b>5</b>	<b>Преценденты</b>	<b>9</b>
<b>6</b>	<b>Use-case</b>	<b>12</b>
	<b>Вывод</b>	<b>13</b>

## Задание

Вариант №149: «Big Puzzle» - пазлы онлайн. Коллекция пазлов онлайн. Подбор пазла по тематике и сложности. Раздел популярных пазлов. Рейтинг игроков - <http://bigpuzzle.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.

## 1 Introduction

### 1.1 Purpose

Данный документ формирует спецификацию требований к продукту BigPuzzle.ru, формирует функциональные и нефункциональные требования к нему.

## 1.2 Scope

Цель данного документа состоит в том, чтобы описать требования к веб-сайту BigPuzzle.ru. Он будет использоваться сотрудниками для создания продукта, который полностью соответствует описанным в документе требованиям

## 1.3 Definitions, Acronyms and Abbreviations

- ПК, пк - персональный компьютер
- Соц. сети - социальные сети
- email - электронная почта - сервис для отправки и приёма электронных писем
- тег - именованная метка
- пагинация - паттерн отображения большого количества списочных или табличных данных посредством загрузки только их части одновременно
- пиастры - название внутриигровой валюты (рейтинга)
- vip-статус - статус пользовательского аккаунта, позволяющий пользоваться сайтом без рекламы
- Uptime - длительность или доля времени работы системы без сбоев
- Backup - копия данных системы, хранящаяся независимо от основных данных, позволяющая восстановить основные данные в экстренных случаях
- js - язык программирования JavaScript
- HTML - язык гипертекстовой разметки HyperText Markup Language
- CSS - язык описания стилей страницы Cascading Style Sheets
- HTTP - протокол передачи данных HyperText Transfer Protocol

## 1.4 References

BigPuzzle.ru - основной сайт системы

## 1.5 Overview

- раздел 2 - общее описание факторов системы
- раздел 3 - описание всех требований к разрабатываемой системе
- раздел 4 содержит информативные таблицы с описанием атрибутов перечисленных требований
- разделы 5, 6 - содержат прецеденты и use-case системы

## 2 Overall Description

### 2.1 Product functions

Сайт должен содержать главную страницу с ссылками на другие разделы системы. Среди этих разделов должны быть:

1. Клуб
2. Как играть
3. Для детей
4. Новости
5. Контакты
6. Войти, Регистрация, забыли пароль?
7. Топ 20
8. Все пазлы

После выбора пазла пользователь должен перенаправляться на интерактивный сайт, на котором изображение разбивается на выбранное им количество даталей. Пользователь должен иметь возможность выбирать деталь по одной и помещать ее на свободную доску, собирая пазл.

### 2.2 User characteristics

Целевой аудиторией продукта являются любители пазлов в возрасте 8-80 лет, которые имеют доступ к ПК с установленным браузером, доступ в интернет и имеют базовые навыки владения ПК. Пользователи должны иметь возможность создавать аккаунты на сайте для сохранения своих достижений, виртуальной валюты и прочего. Также система должна поддерживать специальный тип пользовательского аккаунта: администратор, который имеет возможность добавлять новые картинки в систему и модерировать комментарии.

### 2.3 Assumptions and dependencies

BigPuzzle.ru предполагает, что пользователи имеют доступ к современному веб-браузеру с интернет-соединением и могут создавать и управлять учетными записями с использованием электронной почты. Сайт также зависит от доступности и стабильности сторонних сервисов, таких как обработка платежей и доставка электронной почты.

### 2.4 Constrains

BigPuzzle.ru должен соответствовать всем соответствующим законам и правилам, связанным с онлайн-торговлей, защитой данных пользователей и конфиденциальностью. Сайт должен учитывать лицензии сторонних сервисов, используемых им, и подчиняться этим лицензиям. Сайт должен быть разработан с учетом возможности дальнейшего масштабирования.

## **3 Specific Requirements**

### **3.1 Functionality**

#### **3.1.1 Регистрация пользователя**

Система должна позволять пользователям регистрироваться на сайте, используя свои электронные почтовые адреса. При регистрации пользователю необходимо указывать свой email-адрес, после успешной регистрации он должен получить подтверждение на этот адрес.

#### **3.1.2 Поиск изображения для пазла**

Система должна позволять пользователям искать пазлы по различным критериям: их сложности, тегам. После поискового запроса подходящие изображения для пазлов должны отобразиться пользователю в формате, поддерживающем пагинацию

#### **3.1.3 Решение пазла**

После выбора изображения система должна позволить пользователю выбрать разбиение этого изображения на кусочки пазлов (8x8, 10x10, 12x12, и так далее). После выбора разбиения пользователь должен получать доступ к интерактивной игре, где можно выбрать кусочек и поместить его на свободное поле, формируя собранный пазл.

#### **3.1.4 Заработок виртуальных кредитов**

Система должна позволять пользователям зарабатывать виртуальные кредиты за прохождение пазлов и занесение своих результатов в таблицу лидеров.

#### **3.1.5 Система комментариев**

Система должна позволять пользователям оставлять комментарии к главной странице игры.

#### **3.1.6 Новостная лента**

Система должна иметь вкладку с новостной лентой, в которой отображаются последние новости, связанные с игрой.

#### **3.1.7 Покупка VIP-статуса**

Система должна позволять пользователям покупать VIP-статус для того, чтобы убрать рекламу во время пользования сайтом. VIP-статус может приобретаться по системе подписки: на месяц, 6 месяцев и год.

## **3.2 Usability**

### **3.2.1 Обучаемость**

Система должна быть легко обучаемой для новых пользователей. Среднее время обучения пользователя интерфейсу не должно превышать 5 минут. Среднее время решения одного пазла зависит от количества кусочков в разбиении, но оно не должно превышать 10 минут при разбиении 18x18.

### **3.2.2 Интуитивность интерфейса**

Интерфейс системы должен быть интуитивно понятным для пользователей. Кнопки и элементы управления должны быть ясными и доступными.

### **3.2.3 Оформление сайта**

Оформление сайта должно быть приятным и привлекательным для пользователей. Дизайн должен быть современным и соответствовать тематике сайта.

### **3.2.4 Понятность текстов**

Тексты на сайте должны быть понятными и доступными для всех пользователей. Язык должен быть легко воспринимаемым и не содержать сложных терминов и аббревиатур.

### **3.2.5 Адаптивность сайта**

Сайт должен быть адаптивным и корректно отображаться на различных устройствах и разрешениях экранов.

## **3.3 Reliability**

### **3.3.1 Uptime**

Система должна быть доступна пользователям в течение 99.0% времени в месяц, за исключением запланированных работ по обновлению и технического обслуживания.

### **3.3.2 Backup**

Система должна поддерживать backup данных пользователей для восстановления после сбоев или атак.

## **3.4 Performance**

### **3.4.1 Количество одновременных пользователей**

Система должна поддерживать не менее 1000 одновременных пользователей без значительных уменьшений производительности



### **3.4.2 Отклик**

Максимальное время ответа на запрос пользователя не должно превышать 2 секунд, а среднее время ответа должно быть около 500 миллисекунд.

### **3.4.3 Обработка большого количества одновременных запросов**

Система должна иметь возможность обрабатывать до 1000 одновременных запросов в секунду без значительных уменьшений производительности.

## **3.5 Design constraints**

### **3.5.1 Используемые технологии**

Для создания клиентской стороны необходимо использовать JS, HTML, CSS.

### **3.5.2 Использование сторонних сервисов**

Для оплаты vip-статуса необходимо использовать сторонние сервисы. Система интерактивного сбора пазла должна использовать Unity.

## **3.6 Interfaces**

### **3.6.1 User interfaces**

Пользовательский интерфейс главной страницы должен состоять из:

- Шапки серого цвета, на которой расположены логотип, поля для регистрации и входа
- Кнопок, направляющих на секции системы, расположенных под шапкой
- Ссылок на социальные сети BigPuzzle.ru
- Изображений самых популярных пазлов

Пользовательский интерфейс сборки пазла должен состоять из:

- Свободного белого поля для расположения пазлов
- Поля, в котором расположены ещё не поставленные пазлы
- Таймера, который засекает время сборки

### **3.6.2 Hardware Interfaces**

Серверная часть должна состоять из высокопроизводительной вычислительной системы. Клиенская сторона состоит из мобильного устройства или ПК с доступом в интернет и современным браузером.

### **3.6.3 Software Interfaces**

Система должна иметь возможность использовать почтовые сервисы для регистрации и авторизации

### 3.6.4 Communication Interfaces

Для передачи информации в сети интернет система должна использовать протокол HTTP.

## 3.7 Licensing Requirements

Все права на публикацию и распространение информации с BigPuzzle.ru принадлежат авторам

## 4 Атрибуты требований

### 4.1 Функциональные требования

№	Требование	Приоритет	Трудоемкость (ч)	Стабильность	Риск
1	Поиск по пазлам	10	30	Высокая	Высокий
2	Решение пазла	10	50	Высокая	Высокий
3	Регистрация пользователя	8	4	Высокая	Высокий
4	Заработок виртуальных кредитов	7	6	Средняя	Низкий
5	Покупка VIP-статуса	7	15	Средняя	Высокий
6	Система комментариев	5	8	Низкая	Низкий
7	Новостная лента	4	5	Низкая	Низкий
8	Добавление нового изображения в систему	10	10	Высокая	Высокий

### 4.2 Нефункциональные требования

№	Требование	Приоритет	Трудоемкость (ч)	Стабильность	Риск
1	Обучаемость	7	6	Высокая	Низкий
2	Интуитивность интерфейса	10	30	Высокая	Низкий
3	Оформление сайта	6	30	Высокая	Низкий
4	Понятность текстов	9	6	Высокая	Низкий
5	Поддержание uptime	10	40	Средняя	Высокий
6	Адаптивность сайта	9	10	Средняя	Низкий
7	Количество одновременных пользователей	10	40	Высокая	Высокий
8	Количество одновременных запросов	10	40	Высокая	Высокий
9	Создание backup	9	10	Высокая	Высокий

## 5 Прецеденты

<b>Прецедент:</b> Просмотр главной страницы
<b>ID:</b> 1
<b>Краткое описание:</b> пользователь открывает главную страницу для поиска нужного ему контента, кнопки или пазла.
<b>Главный актер:</b> пользователь
<b>Второстепенные актёры:</b> нет
<b>Предусловия:</b> нет
<b>Основной поток:</b> Пользователь открывает главную страницу, скроллит её и нажимает на кнопки.

<b>Прецедент:</b> игра в пазл
<b>ID:</b> 2
<b>Краткое описание:</b> после выбора картинки пользователь начинает игру в пазл
<b>Главный актер:</b> Пользователь
<b>Второстепенные актёры:</b> нет
<b>Предусловия:</b> нет
<b>Основной поток:</b> 1. Пользователь находит нужное ему изображение на главной странице или в секции поиска 2. Пользователь нажимает на него и переходит на страницу с интерактивной игрой 3. Пользователь выбирает нужное ему разбиение 4. Пользователь использует панель сбоку, чтобы расположить пазлы на свободном поле 5. После правильного расположения пазлов пользователю засчитывается победа

<b>Прецедент:</b> регистрация
<b>ID:</b> 3
<b>Краткое описание:</b> пользователь впервые заходит на сайт и регистрируется
<b>Главный актер:</b> пользователь
<b>Второстепенные актёры:</b> нет
<b>Предусловия:</b> пользователь ещё не зарегистрирован
<b>Основной поток:</b> 1. Пользователь нажимает на кнопку «Регистрация» и переходит в меню регистрации 2. Пользователь вводит свою электронную почту и придуманный пароль 3. Пользователь вводит код, пришедший ему на электронную почту, его данные сохраняются в системе, и он успешно завершает регистрацию

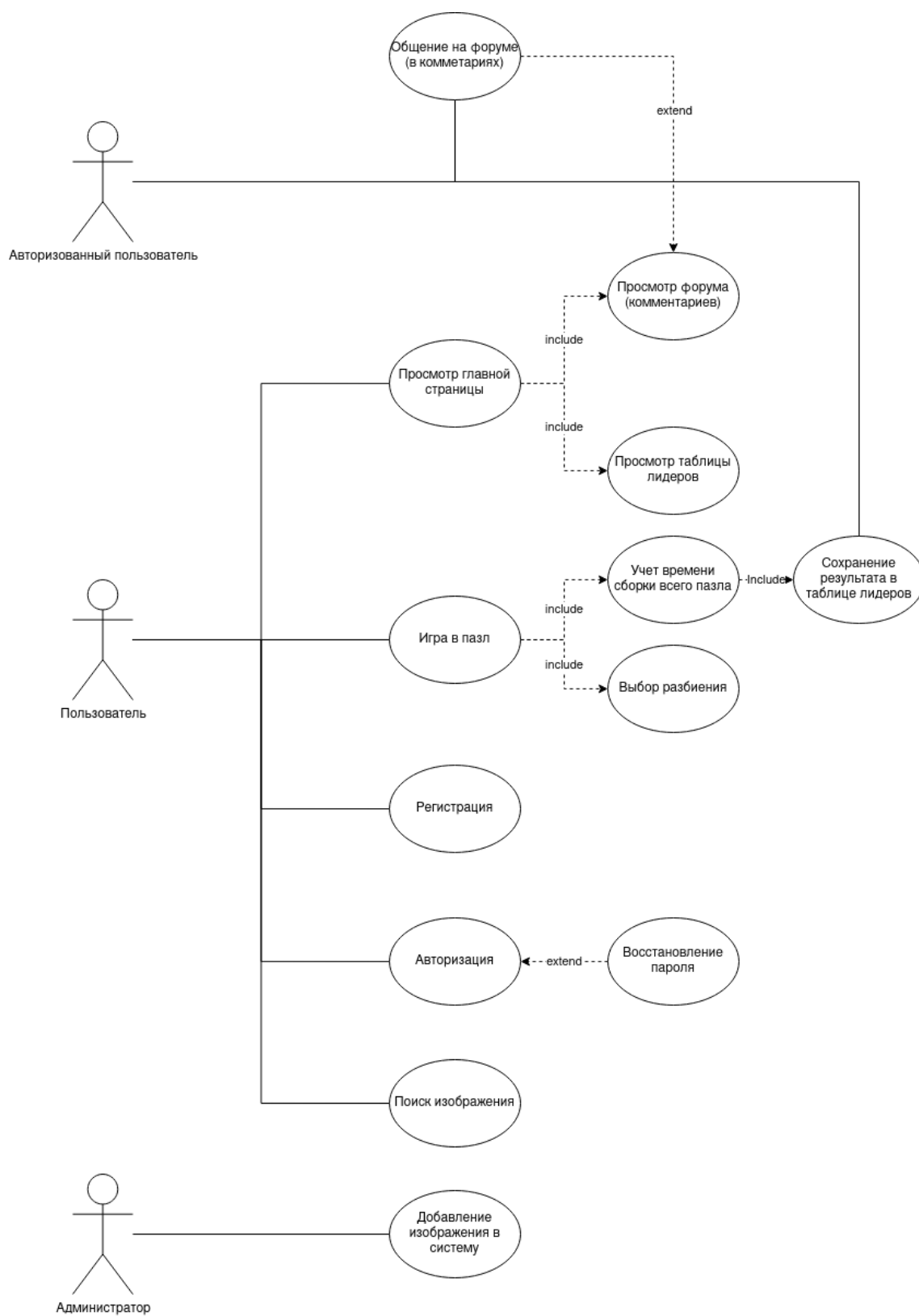
<b>Прецедент:</b> авторизация
<b>ID:</b> 4
<b>Краткое описание:</b> пользователь заходит на сайт и регистрируется
<b>Главный актер:</b> пользователь
<b>Второстепенные актёры:</b> нет
<b>Предусловия:</b> пользователь зарегистрирован на сайте
<b>Основной поток:</b> 1. Пользователь вводит адрес своей электронной почты и пароль в шапке главной страницы и нажимает на кнопку «войти» 2. Система проверяет правильность введенных данных и позволяет пользователю авторизоваться, если данные верны 3. Пользователь успешно авторизуется на сайте

<b>Прецедент:</b> поиск картинки
<b>ID:</b> 5
<b>Краткое описание:</b> пользователь ищет картинку по тегам
<b>Главный актер:</b> пользователь
<b>Второстепенные актёры:</b> нет
<b>Предусловия:</b> нет
<b>Основной поток:</b> 1. Пользователь заходит на страницу поиска картинки по тегам и параметрам 2. Пользователь отмечает несколько тегов 3. Пользователю открывается лента картинок, отфильтрованных по выбранным тегам

<b>Прецедент:</b> восстановление аккаунта
<b>ID:</b> 6
<b>Краткое описание:</b> пользователь восстанавливает пароль от аккаунта
<b>Главный актер:</b> пользователь
<b>Второстепенные актёры:</b> нет
<b>Предусловия:</b> пользователь не помнит пароль от своего аккаунта
<b>Основной поток:</b> 1. Пользователь нажимает на кнопку «забыли пароль?» на главной странице 2. Пользователь вводит адрес своей электронной почты на открывшейся странице 3. Пользователь получает на электронную почту специальный код и вводит его на странице 4. В открывшемся окне пользователь вводит новый пароль 5. Новый пароль сохраняется в системе

<b>Прецедент:</b> добавление нового изображения в систему
<b>ID:</b> 7
<b>Краткое описание:</b> администратор добавляет новое изображение в систему
<b>Главный актер:</b> администратор
<b>Второстепенные актёры:</b> нет
<b>Предусловия:</b> нет
<b>Основной поток:</b> 1. Администратор входит в систему под специальным аккаунтом, расширяющим возможности авторизованных пользователей 2. Администратор выбирает изображение и загружает его в систему 3. Система сохраняет изображение и позволяет остальным пользователям находить это изображение в системе поиска

## 6 Use-case



## Вывод

Благодаря Software Requirements Specification можно очень точно предъявить требования к разрабатываемому ПО, а методология RUP и Use-Case диаграммы будут очень полезны при тестировании и выявлении каких-либо проблем на самых ранних стадиях разработки.