**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

Факультет прикладной математики и информатики

Кафедра технологий программирования

Отчет

выполнения лабораторных работ по предмету:

«Проектирование программных систем»

**Выполнил:**

студент 2 курса 14 группы (ПИ)

Петров Андрей Александрович

**Преподаватель:**

Зенько Татьяна Алексеевна

Минск

2021

**ОГЛАВЛЕНИЕ**

[ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ 4](#_Toc71922544)

[ГЛОССАРИЙ 5](#_Toc71922545)

[ДОПОЛНИТЕЛЬНАЯ СПЕЦИФИКАЦИЯ 5](#_Toc71922546)

[ФУНКЦИОНАЛЬНАЯ МОДЕЛЬ СИСТЕМЫ В ВИДЕ ДИАГРАММ ВАРИАНТОВ ИСПОЛЬЗОВАНИЯ 6](#_Toc71922547)

[1. Сценарий событий для прецедента «Добавить данные о публикации» 7](#_Toc71922548)

[2. Сценарий событий для прецедента «Войти в систему» 8](#_Toc71922549)

[3. Сценарий событий для прецедента «Зарегистрироваться в системе» 9](#_Toc71922550)

[4. Сценарий событий для прецедента «Получить информацию о своих публикациях» 10](#_Toc71922551)

[5. Сценарий событий для прецедента «Работа с дублями» 11](#_Toc71922552)

[ДИАГРАММЫ ДЕЯТЕЛЬНОСТИ 13](#_Toc71922553)

[Прецедент вход в систему: 13](#_Toc71922554)

[Прецедент регистрация в системе: 13](#_Toc71922555)

[Прецедент добавление публикации: 14](#_Toc71922556)

[ДИАГРАММЫ СОСТОЯНИЙ 15](#_Toc71922557)

[Диаграмма состояний публикации: 15](#_Toc71922558)

[Диаграмма состояний аккаунта: 15](#_Toc71922559)

[ДИАГРАММЫ КЛАССОВ 16](#_Toc71922560)

[Общая диаграмма: 16](#_Toc71922561)

[Прецедент вход в систему: 16](#_Toc71922562)

[Прецедент регистрация в системе: 16](#_Toc71922563)

[Прецедент добавить публикацию: 17](#_Toc71922564)

[ДИАГРАММЫ ОБЪЕКТОВ 18](#_Toc71922565)

[Диаграмма добавление публикации: 18](#_Toc71922566)

[ДИАГРАММЫ ПОСЛЕДОВАТЕЛЬНОСТЕЙ 19](#_Toc71922567)

[Добавить публикацию: 19](#_Toc71922568)

[Получить список публикаций: 19](#_Toc71922569)

[Вход в систему: 20](#_Toc71922570)

[ДИАГРАММА ПАКЕТОВ 20](#_Toc71922571)

[ДИАГРАММА КОМПОНЕНТОВ 21](#_Toc71922572)

[ДИАГРАММА РАЗВЕРТЫВАНИЯ 21](#_Toc71922573)

[СХЕМА БАЗЫ ДАННЫХ 22](#_Toc71922574)

[ТЕСТЫ 23](#_Toc71922575)

[Unit tests 23](#_Toc71922576)

[Integration tests 23](#_Toc71922577)

[System tests 23](#_Toc71922578)

[Acceptance tests 23](#_Toc71922579)

[Scenario tests 24](#_Toc71922580)

# 

# **ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ**

Библиографическая система предназначена для хранения сведений о публикациях, ссылок между публикациями и расчёта библиометрического показателя – индекса цитирования автора. Операторы системы добавляют в систему данные о публикациях. Научный журнал или издательство присылает им соответствующие сведения в bib-файлах программы BibTeX (см. описание в Википедии). Оператор указывает имя файла, а система считывает данные и водит внутри себя записи о публикации. Если можно однозначно установить автора (авторов), что происходит не всегда, так как могут быть полные тёзки, то система связывает публикацию и автора. Если нет однозначности, то публикация помечается как возможно принадлежащая каждому полному тезке. Если автора в системе нет, то запись о нём автоматически создаётся. Так как в систему могут быть случайно внесены данные об одной и той же публикации дважды, операторам системы доступна функция поиска дублей. В списке найденных дублей система выводит пары (тройки, n-ки) публикаций, которые она считает дублирующимися. Оператор может указать, какие дубли следует удалить из системы. Авторы могут регистрироваться в систему, чтобы получать доступ к списку своих публикаций (на экране и в формате bib-файла), помогать разрешить неоднозначность определения автора, давать сведения о цитировании, получать значения своего индекса цитирования. При разрешении неоднозначности автору высвечивается перечень публикаций, автором которых он, возможно, является. Он может подтвердить своё авторство или отказаться. Если какая-то публикация по ошибке была отнесена к неверному автору, таковой автор может удалить её из списка своих публикаций. Для исправления обратных ошибок система даёт автору возможность поиска публикаций по названию, журналу и т. п., и сообщения о своём авторстве (в случае если он обнаружил, что публикация ошибочно приписана другому). Сведения о цитировании предоставляются автором в виде bib-файла, в котором записан библиографический список из его публикации. Получив этот файл, система находит/добавляет публикации в свою базу и указывает, что публикация автора ссылается на каждую из них. Индекс цитирования автора вычисляется по формуле индекса Хирша. Разработанная модель должна содержать схему базы данных об авторах, их публикациях, цитировании публикаций.

# **ГЛОССАРИЙ**

|  |  |
| --- | --- |
| Библиометрический показатель  (Bibliometric index) | Индекс цитирования автора. |
| Оператор системы  (System Operator) | Работник системы. Добавляет в систему данные о публикациях. |
| Автор  (Author) | Пользователь системы регистраций. Получает доступ к списку своих публикаций. |
| Система дублей  (Duplicate System) | Система предоставляющая операторам системы возможность поиска дублей и их удаления. |
| Регистрация в системе  (Registration) | Процесс после которого автор получает доступ к списку своих публикаций. |

# **ДОПОЛНИТЕЛЬНАЯ СПЕЦИФИКАЦИЯ**

1. **Функциональные возможности**

Система должна поддерживать многопользовательский режим работы. Несколько авторов и/или операторов могут одновременно использовать систему.

1. **Требования по реализации**

Система должна поддерживаться Windows и macOS.

1. **Надежность**

Система должна быть в работоспособном состоянии 24 часа в день 7 дней в неделю, время простоя – не более 10%.

1. **Производительность**

Система должна поддерживать до 1000 одновременно работающих пользователей.

1. **Безопасность**

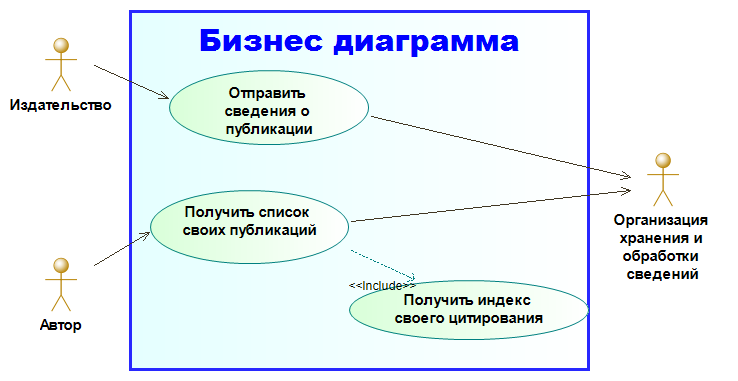
Авторы не должны иметь доступ к публикациям других авторов (за исключением случаев отсутствия у автора его публикации в списке его публикаций). Только операторы могут добавлять и удалять публикации.

1. **Проектные ограничения**

Разработанная модель должна содержать схему базы данных об авторах, их публикациях, цитировании публикаций.

# **ФУНКЦИОНАЛЬНАЯ МОДЕЛЬ СИСТЕМЫ В ВИДЕ ДИАГРАММ ВАРИАНТОВ ИСПОЛЬЗОВАНИЯ**





**Действующие лица:**

*Оператор системы* – добавляет сведения о публикациях в систему, удаляет сведения.

*Издательство* – присылает авторам сведения о публикации в bib-файлах.

*Автор* – пользователь системы, может получать список своих публикаций, индекс своего цитирования, а также разрешать неоднозначности выбора автора.

*База данных* – хранит сведения о публикациях.

**Описание прецедентов**

*Добавить данные о публикации* – доступна операторам системы. Позволяет добавлять данные о публикации в систему.

*Работа с дублями* – запускается оператором системы. Система ищет повторяющиеся публикации и предоставляет возможность оператору системы удалить их.

*Зарегистрироваться в системе* – запускается автором. Позволяет ему создавать аккаунт в системе и получать информацию о своих публикациях.

*Войти в систему* – запускается автором, либо оператором системы. Позволяет пользователю системы войти в систему.

*Получить информацию о своих публикации* – запускается автором. Предоставляет автору список его публикации и его библиометрический показатель. Также предоставляет автору возможность разрешения неоднозначности выбора автора публикации.

*Отправить сведенья о публикации* – издатель отправляет операторам системы сведенья о публикации.

## **1. Сценарий событий для прецедента «Добавить данные о публикации»**

**Краткое описание**

Данный вариант использования позволяет операторам добавлять в систему новые дынные о публикации.

**Основной поток событий**

1. Издательство присылает сведения в bib-файлах программы BibTeX.
2. Оператор указывает имя присланного файла.
3. Система считывает данные и вводит внутри себя записи о публикации.
4. Система связывает публикацию и автора.
5. Данные добавляются в базу данных.
6. Система подтверждает, что сведения добавлены.

**Альтернативные потоки событий**

*2А. Указанный файл не существует*

1. Система обнаруживает, что указанный файл не существует.
2. Система выдает сообщение об ошибке.
3. Вариант использования завершается.

*4А. Неоднозначность выбора автора*

1. Система обнаруживает, что существует неоднозначность выбора автора.
2. Публикация помечается как возможно принадлежащая каждому автору.
3. Выполнение переходит на шаг 6 основного потока.

*4Б. Указанного автора нет в системе*

1. Система обнаруживает, что указанного автора нет в системе.
2. Система создает запись об авторе.
3. Система связывает публикацию с автором.
4. Выполнение переходит на шаг 6 основного потока.

**Предусловия.**

Издательство предварительно должно отправить сведения о публикации.

**Постусловия**

Если вариант использования завершен успешно, то в систему будут введены записи о публикации. Эти записи будут привязаны к автору либо помечены как возможно принадлежащие авторам.



## **2. Сценарий событий для прецедента «Войти в систему»**

**Краткое описание**

Данный вариант использования описывает вход пользователя в библиографическую систему.

**Основной поток событий**

1. Система запрашивает имя пользователя и пароль.
2. Пользователь вводит имя и пароль.
3. Система подтверждает правильность имени и пароля, определяет тип пользователя (оператор, автор) и выводит главное меню, дающее доступ к функциям системы в соответствии с типом пользователя.

**Альтернативные потоки**

*3А. Неправильное имя/пароль*

1. Система обнаруживает, что комбинация имени и пароля не верна.

2. Система сообщает об ошибке и предлагает пользователю либо заново ввести имя и пароль, либо отказаться от входа в систему.

3. Пользователь сообщает системе свой выбор.

4. В соответствии с выбором пользователя либо выполнение переходит на начало основного потока, либо вариант использования завершается.

**Предусловия**

Отсутствуют.

**Постусловия**

Если вариант использования выполнен успешно, система предоставляет доступ к главному меню пользователю, сообщившему верную комбинацию имени и пароля. В противном случае система гарантирует, что пользователю, сообщившему неверную комбинацию имени и пароля, доступ к меню не будет предоставлен.

## **3. Сценарий событий для прецедента «Зарегистрироваться в системе»**

**Краткое описание**

Данный вариант использования позволяет автору зарегистрироваться в системе. Система предоставляет автору доступ к списку его публикаций.

**Основной поток событий**

1. Автор сообщает о желании зарегистрироваться в системе.
2. Система запрашивает данные для регистрации (ФИО, электронная почта, пароль).
3. Система добавляет аккаунт автора.
4. Система выводит сообщение об успешной регистрации.

**Альтернативные потоки**

*2А. Электронная почта уже занята*

1. Система обнаруживает, что электронная почта уже занята.
2. Система сообщает пользователю об ошибке.
3. Система сообщает об ошибке и предлагает пользователю либо заново ввести данные, либо отказаться от регистрации в систему.
4. Пользователь сообщает системе свой выбор.
5. В соответствии с выбором пользователя либо выполнение переходит на шаг 2 основного потока, либо вариант использования завершается.

**Предусловия**

Отсутствуют.

**Постусловия**

Если вариант использования завершен успешно, то в системе создается аккаунт автора и происходит автоматический вход в систему под этим аккаунтом.

## **4. Сценарий событий для прецедента «Получить информацию о своих публикациях»**

**Краткое описание**

Данный вариант использования позволяет автору получать список своих публикаций, помогать разрешить неоднозначность определения автора, давать сведения о цитировании, получать значения своего индекса цитирования.

**Основной поток событий**

1. Автор запрашивает информацию о своих публикациях.
2. Система выводит список публикаций автора и индекс цитирования, также выдает сообщение об авторстве (все публикации связаны только с этим автором, существует неоднозначность).
3. Автор остается в системе.

**Подчиненные потоки событий**

*2А. Существует неоднозначность в определении автора*

1. Система обнаруживает неоднозначность в определении автора публикации
2. Автору предлагается подтвердить либо опровергнуть авторство публикации
3. Система сохраняет ответ автора.
4. Выполнение переходит на шаг 3 основного потока

*2Б. Автор не обнаружил свою публикацию в выведенном ему списке.*

1. Система предоставляет автору возможность поиска публикации по названию.
2. Автор сообщает о своем авторстве либо продолжает пользование системой.
3. Система в зависимости от выбора автора сохраняет сообщение автора либо выполнение переходит на шаг 3 основного потока.

*2В. Автор обнаружил ошибочно приписанную ему публикацию*

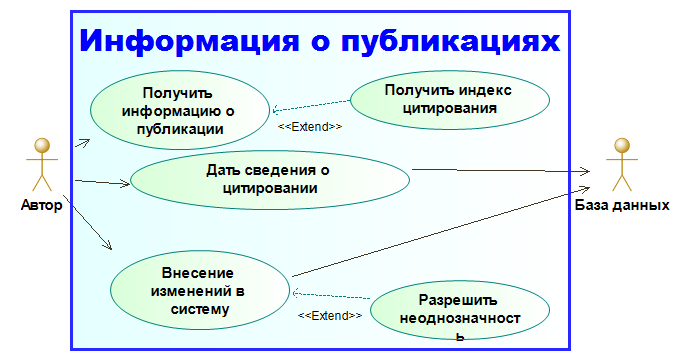
1. Автор сообщает системе об ошибке.
2. Система удаляет публикацию из списка публикаций автора.
3. Выполнение переходит на шаг 3 основного потока.

**Предусловия**

Автор должен находиться в системе.

**Постусловия**

Отсутствуют



## **5. Сценарий событий для прецедента «Работа с дублями»**

**Краткое описание**

Данный вариант использования позволяет операторам системы искать и, при необходимости, удалять дубли.

**Основной поток событий**

1. Оператор вызывает функцию поиска дублей.
2. Система ищет повторяющиеся публикации и отображает их.
3. Оператор выбирает дубли которые нужно удалить.
4. Система удаляет выбранные оператором объекты.
5. Система оповещает оператора об успешном удалении.
6. Работа с дублями завершается.

**Альтернативные потоки**

*2А. Система не нашла дубли*

1. Система оповещает оператора об отсутствии дублей.
2. Выполнение переходит на шаг 6 основного потока событий.

**Подчиненные потоки событий**

*3А.* *Оператор пытается удалить все публикации-дубли*

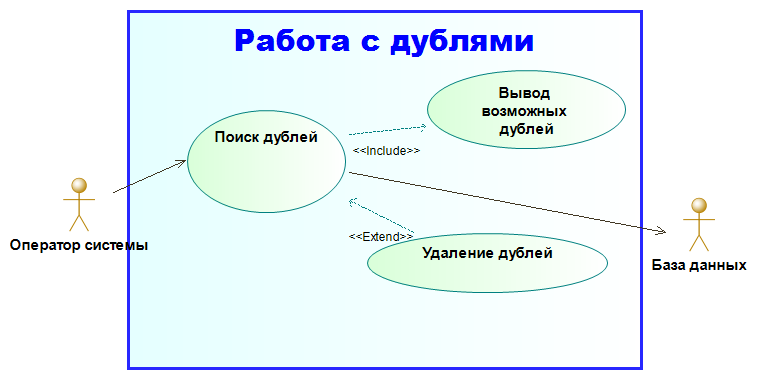
1. Система оповещает оператора о том, что он пытается удалить все публикации-дубли
2. Система запрашивает подтверждение на удаление.
3. В зависимости от выбора оператора система удаляет публикации, либо выполнение переходит на шаг 3 основного потока.
4. Выполнение переходит на шаг 5 основного потока.

**Предусловия**

Оператор должен находиться в системе.

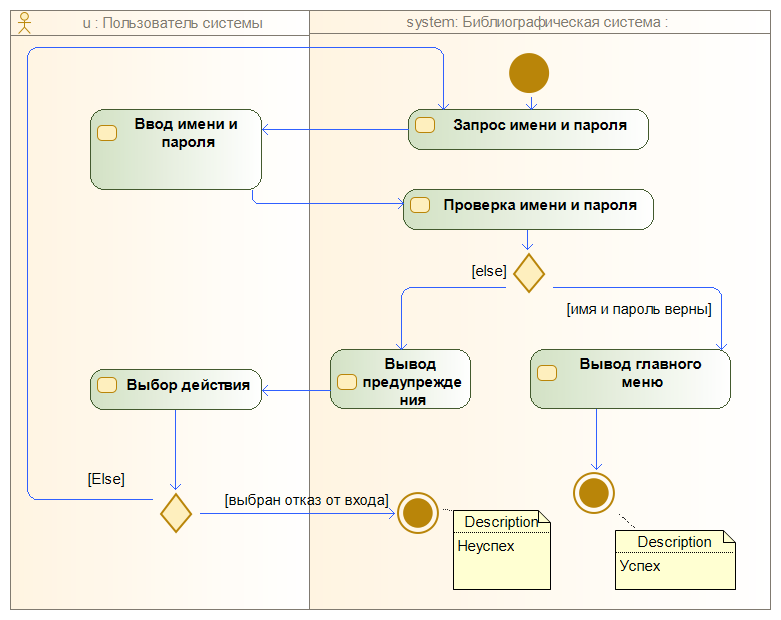
**Постусловия**

Отсутствуют.

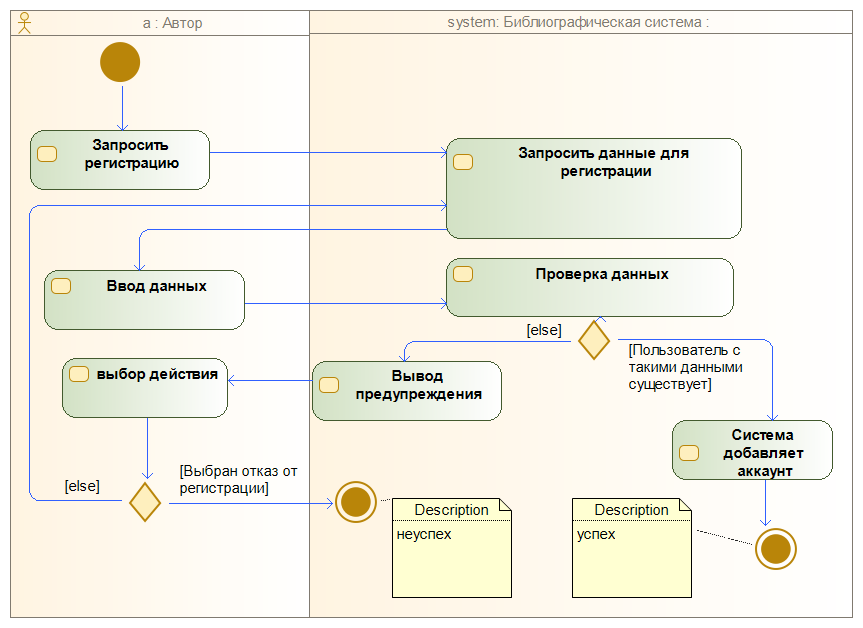


# **ДИАГРАММЫ ДЕЯТЕЛЬНОСТИ**

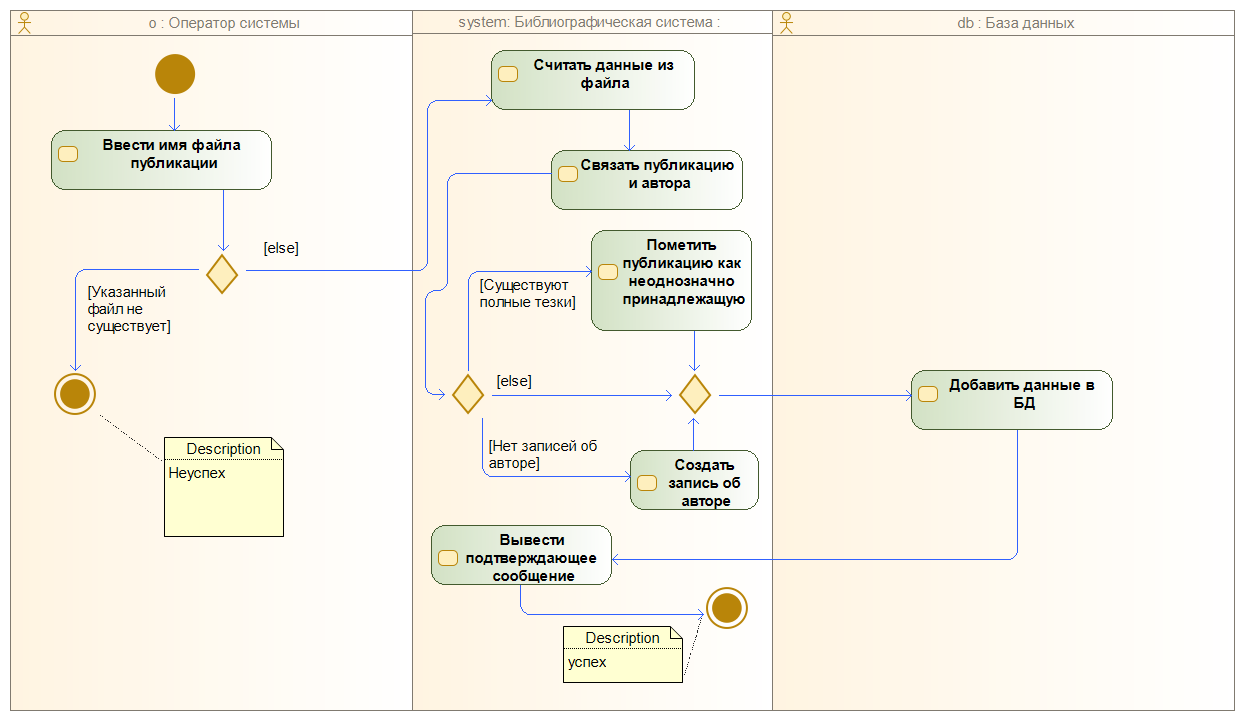
## Прецедент *вход в систему:*



## Прецедент *регистрация в системе:*

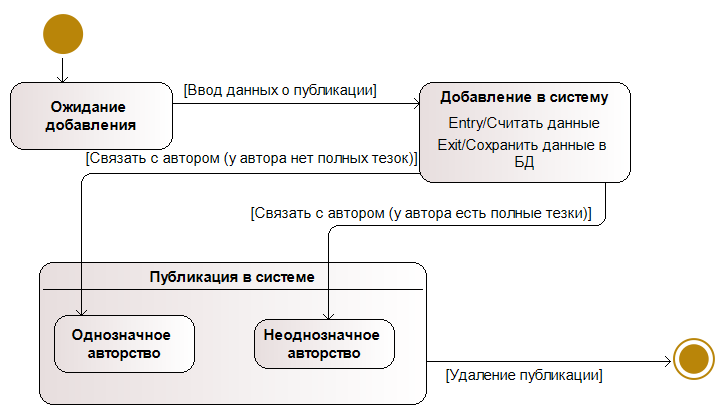


## Прецедент *добавление публикации:*

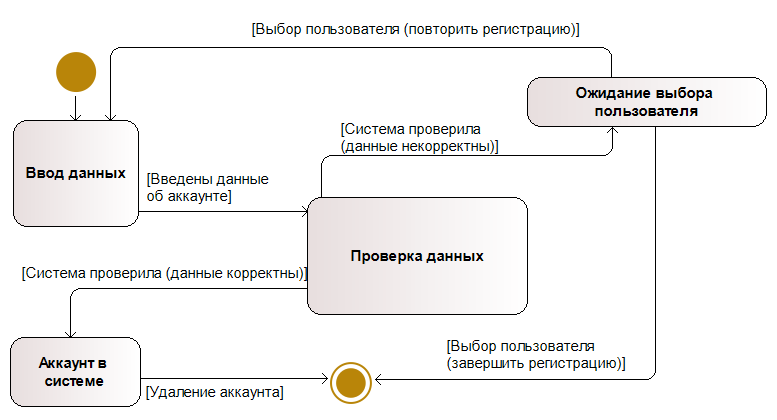


# **ДИАГРАММЫ СОСТОЯНИЙ**

## Диаграмма *состояний публикации:*

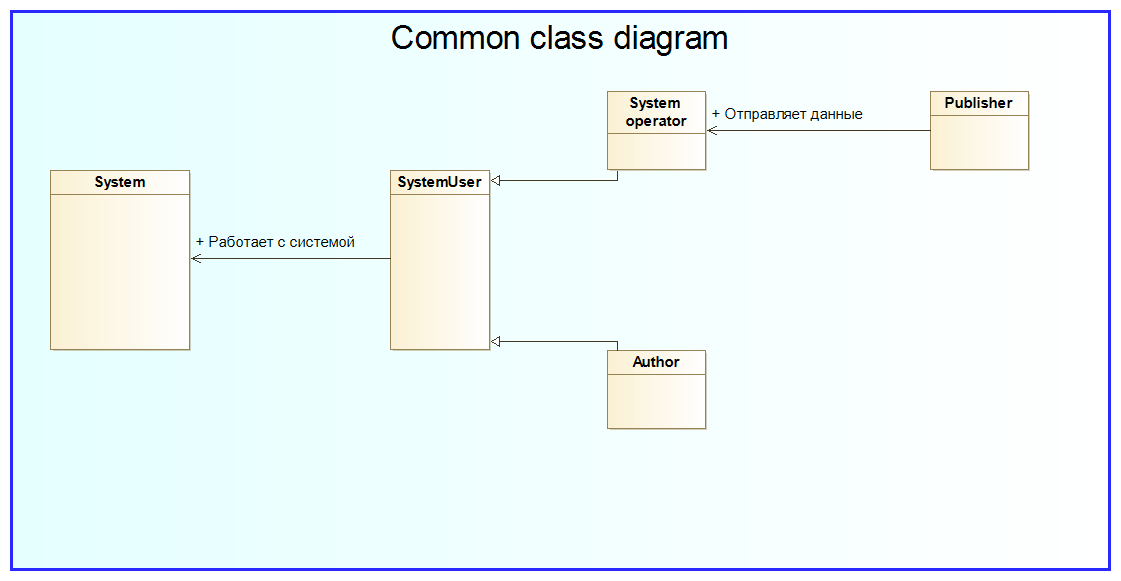


## Диаграмма *состояний аккаунта:*



# **ДИАГРАММЫ КЛАССОВ**

## Общая диаграмма:



## Прецедент *вход в систему:*

Изображение выглядит как стол

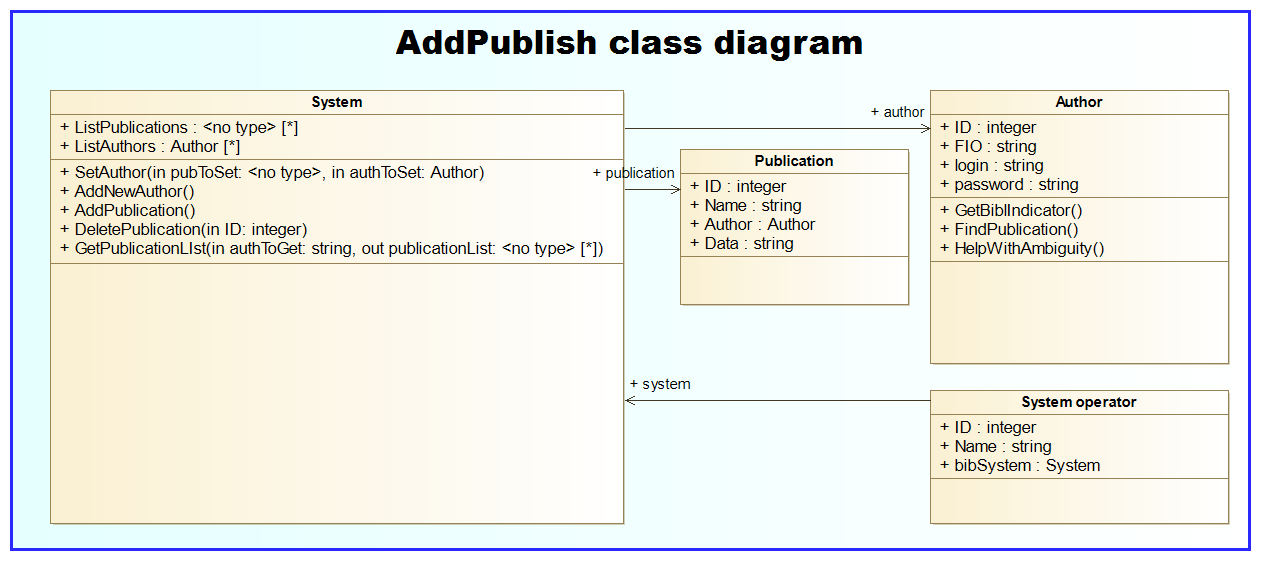
Автоматически созданное описание

## Прецедент *регистрация в системе:*

Изображение выглядит как текст

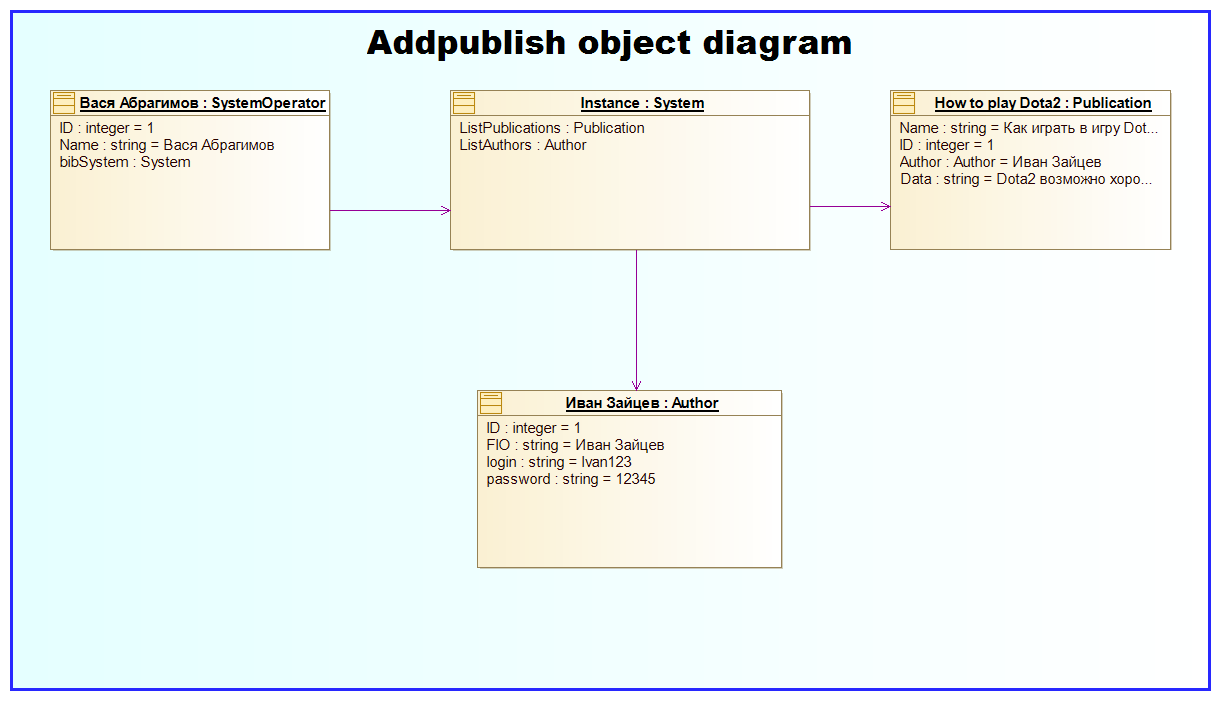
Автоматически созданное описание

## Прецедент *добавить публикацию:*



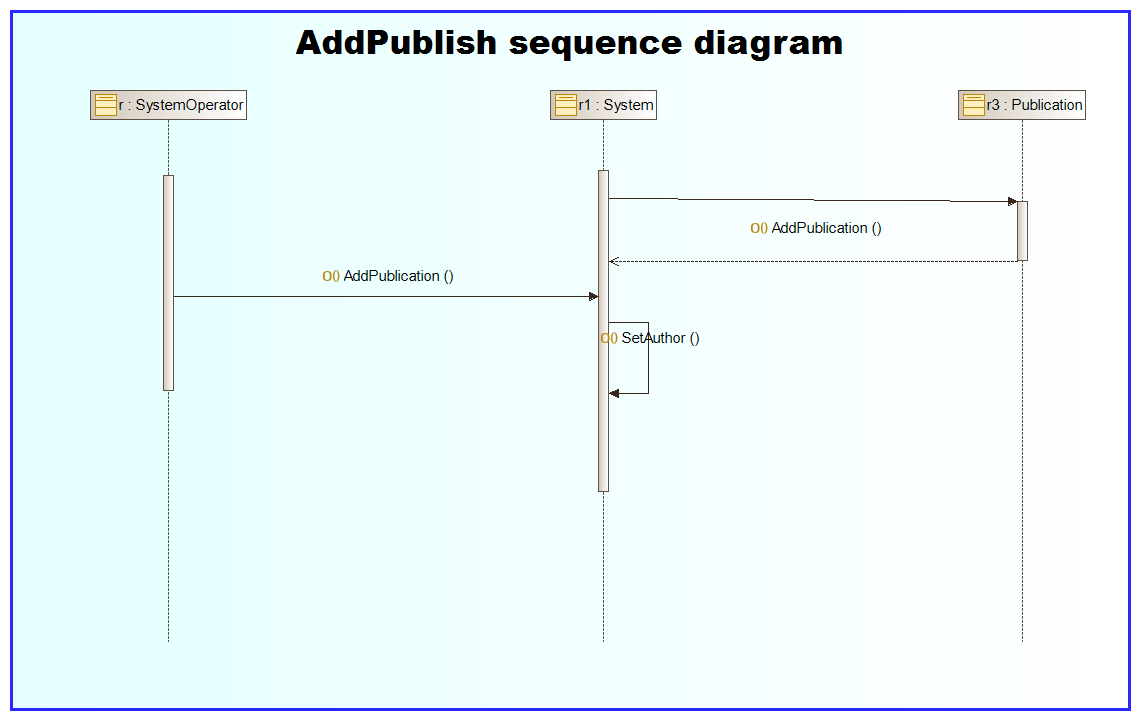
# **ДИАГРАММЫ ОБЪЕКТОВ**

## Диаграмма *добавление публикации:*

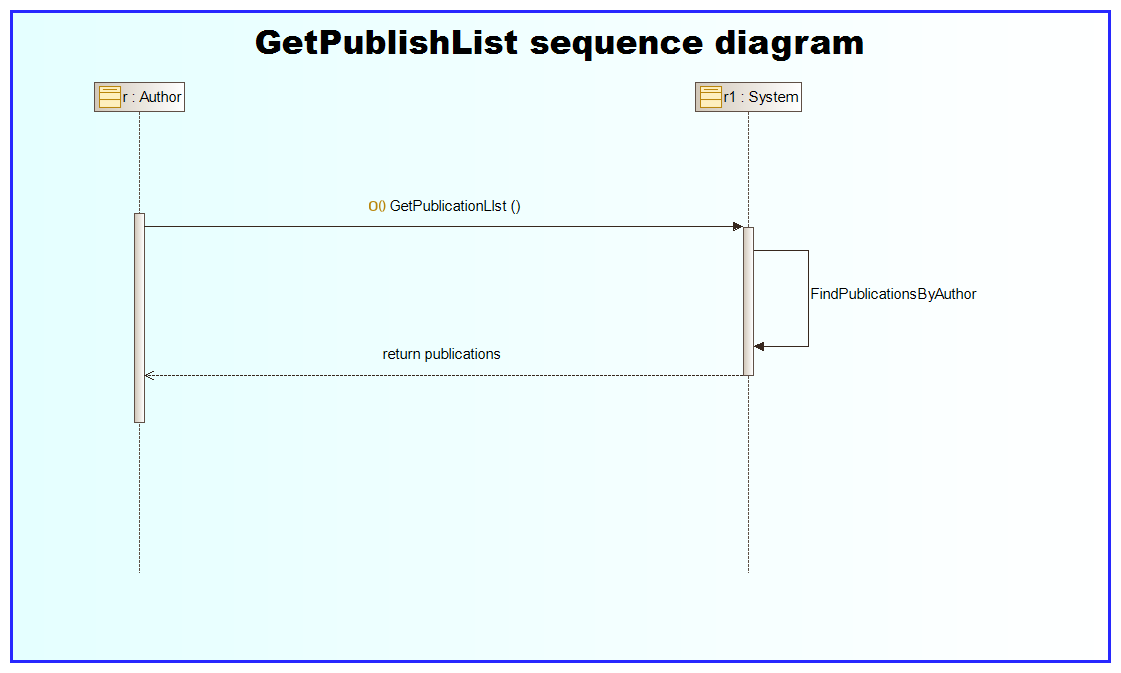


# **ДИАГРАММЫ ПОСЛЕДОВАТЕЛЬНОСТЕЙ**

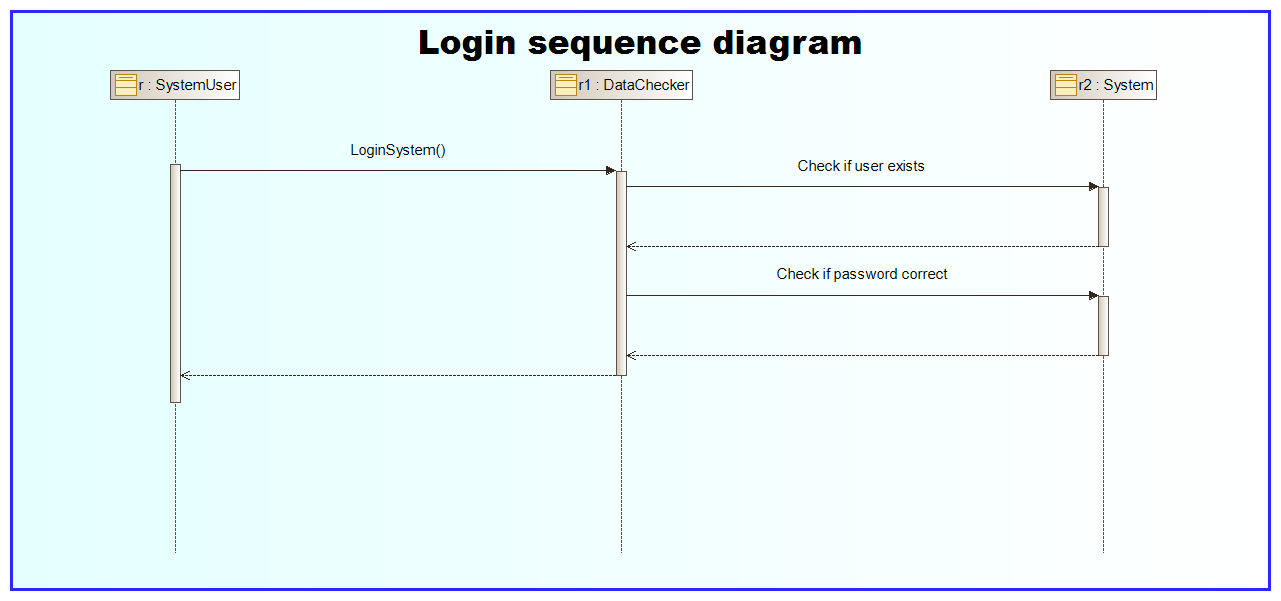
## Добавить публикацию:



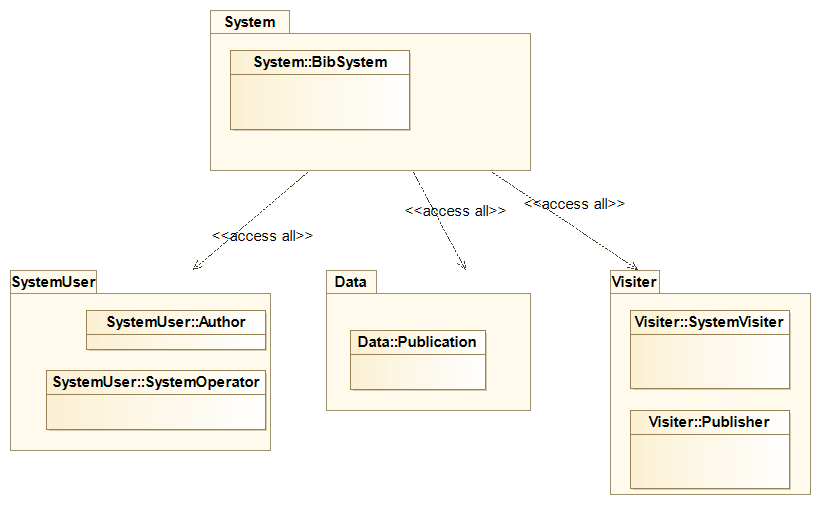
## Получить список публикаций:



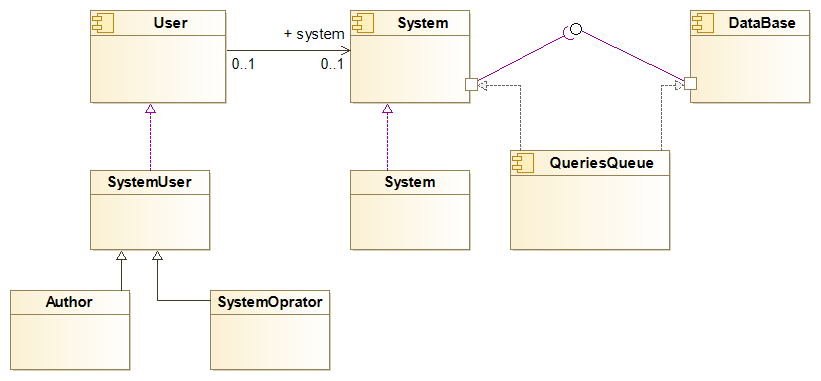
## Вход в систему:



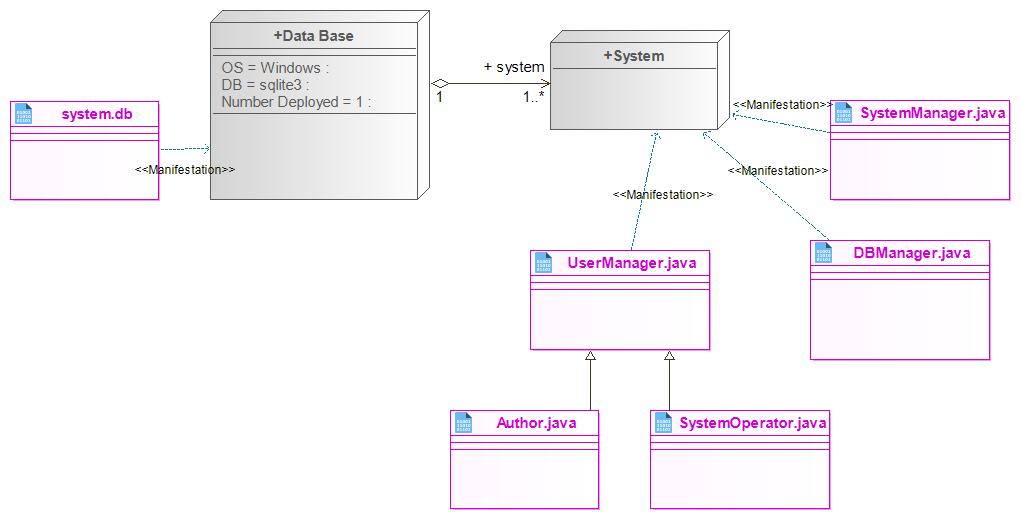
# **ДИАГРАММА ПАКЕТОВ**



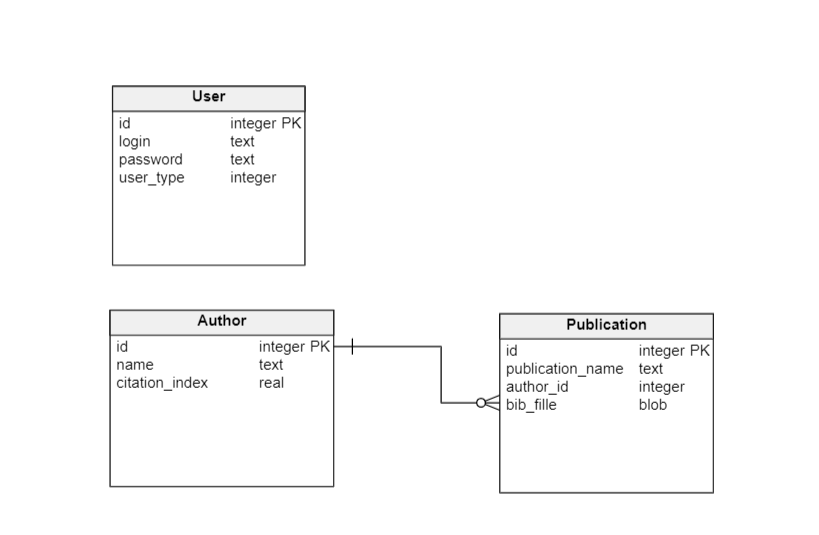
# **ДИАГРАММА КОМПОНЕНТОВ**



# **ДИАГРАММА РАЗВЕРТЫВАНИЯ**



# **СХЕМА БАЗЫ ДАННЫХ**

****

# **ТЕСТЫ**

## **Unit tests**

Тестирование метода **Publication** ***FindPublication(string publicationName);***

Метод возвращает объект класса Publication если публикация нашлась и null в противном случае.

|  |  |  |
| --- | --- | --- |
| №  п/п | Входные данные | Ожидаемые выходные данные |
| 1 | “existPublication” | existPublication |
| 2 | “nullPublication” | null |

Тестирование метода ***List<Publication> GetPublicationList(Author author);***

Метод возвращает список публикаций автора. Возвращает null, если автора не существует

|  |  |  |
| --- | --- | --- |
| №  п/п | Входные данные | Ожидаемые выходные данные |
| 1 | author | List<Publication> |
| 2 | null | null |

## **Integration tests**

Тестирование метода ***SetAuthor(Publication pubToSet, Author author);***

Метод назначает author автором публикации.

Аспекты для проверки:

* Существование публикации.
* Существование автора.
* Принадлежность публикации к другому автору.

## **System tests**

Система регистрации

Пользователь отправляет запрос на регистрацию. Ему отображаются поля, куда он вводит данные для регистрации (Имя, Электронная почта, Логин, Пароль)

Аспекты для проверки:

* Правильность вводимых данных с точки зрения синтаксиса
* Отсутствие соответствующего логина в базе данных
* Отсутствие соответствующей эл. Почты в базе данных

## **Acceptance tests**

Тестирование следующих функций для автора:

* Помощь с разрешением неоднозначностей
* Получение библиографического показателя
* Получение информации о цитировании

Тестирование следующих функций для оператора системы:

* Добавление публикации
* Удаление публикации
* Поиск дублей

## **Scenario tests**

Сценарий событий для прецедента «Войти в систему».

*1.1 Предусловия.*

Для любого взаимодействия с системой необходимо в неё войти, поэтому он должен быть зарегистрирован в системе.

*1.2 Главный сценарий.*

Вариант использования начинает выполняться, когда пользователь подключается к системе и вводит свое имя и пароль. Система проверяет правильность пароля (Е-1). Если пароль не верен, пользователю предлагается сменить пароль (Е-2) или выйти (Е-3).

*1.3 Альтернативные сценарии*

Е-1: введено неправильное имя или пароль. Пользователь должен повторить ввод или завершить прецедент.

Е-2: если пользователь 3 раза ввёл неверный пароль, ему предлагается сменить пароль с помощью электронной почты.

Е-3: пользователь может выйти из системы.

*1.4 Постусловия*

Если вход выполнен успешно, пользователь получает доступ к системе.

*Тестирование:*

I. Успешный тест.

1. Пользователь ввёл логин: “user” и пароль “12345”
2. Отправлен запрос в базу данных. Осуществляется проверка. Данные оказались верны.
3. Пользователю открывается интерфейс системы

II. Ошибочный тест.

1. Пользователь ввёл логин “user” и пароль “11111”
2. Отправлен запрос в базу данных. Осуществляется проверка. Данные оказались верны.
3. Пользователю еще раз открывается форма входа в систему.