**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

Факультет прикладной математики и информатики

Кафедра технологий программирования

Доскоч Роман Дмитриевич

Тестирование проектируемого программного обеспечения

Отчет по лабораторной работе №7

«Проектирование программных систем» студента 2 курса 13 группы

**Преподаватель:**

Зенько Т.А.

**Минск 2021**

**Описание предметной области**

**АИС «Библиографическая система»**

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.

**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**

Вариант использования ”Войти в систему”:

* Ввод неверных входных данных
* Ввод верных входных данных
* Отмена входа

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Test Case Description** | **Test Data / Test Steps** | **Expected Result** |
| **1** | Сценарий «Вход в систему» | Верные данные в текстовых полях логина и пароля | Включается новое меню |
| **2** | Сценарий «Вход в систему» | Неверные данные в текстовых полях логина и пароля | Выдаётся ошибка о том, что введены неверные данные |
| **1.1** | Сценарий «Добавление информации о новом сотруднике» | Верные данные в текстовых полях, описывающих информацию о сотруднике | В базе данных появятся данные о новом авторе |
| **1.2** | Сценарий «Добавление информации о новом сотруднике» | Неверные данные в текстовых полях, описывающих информацию о сотруднике | Выдаётся ошибка о том, что введены неверные данные |

**Список литературы**

1. Статья: <https://www.guru99.com/test-case.html>
2. Статья: <https://www.guru99.com/test-scenario.html>
3. Статья: https://webhamster.ru/mytetrashare/index/mtb0/1410498808fclssvq9e2.html