фМИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

**Проектирование информационных систем**

**Тема:** «ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ МОДЕЛИРОВАНИЕ. UML-ДИАГРАММЫ ПОВЕДЕНИЯ. »

**Цель:** Изучение методологии объектно-ориентированного моделирования средствами UML. Ознакомление с основными принципами объектно-ориентированного проектирования программного обеспечения, получение навыков проектирования функциональности информационной системы с применением UML.

Выполнил:

Кальчевский Даниил Андреевич,

3 курс 1 группа

Минск 2023

**1. Постановка задачи**

В ходе данной лабораторной работы на основании требований необходимо разработать диаграмму прецедентов (Use-case diagram).

В результате необходимо иметь визуализированную диаграмму прецедентов, а также отчёт по проделанной лабораторной работе.

**2. Описание программных средств**

Название: Draw.IO

Версия: последняя версия на данный момент - 14.8.7

Разработчик: Draw.IO Team

Адрес загрузки: Вы можете загрузить Draw.IO с официального веб-сайта разработчика по адресу: https://www.draw.io/

Режим использования: Draw.IO представляет собой онлайн-инструмент для создания диаграмм, схем, организационных графиков и других визуальных представлений. Он позволяет пользователям создавать, редактировать и совместно работать над документами в режиме реального времени.

Доступность на платформах: Draw.IO доступен для использования веб-браузерах на различных платформах, включая Windows, macOS, Linux и Chrome OS. Также есть мобильные приложения для iOS и Android.

Совместимость с другими моделями: Draw.IO является независимым инструментом и не требует специальных моделей для работы. Он предоставляет широкий набор функций для создания и редактирования визуальных представлений без привязки к конкретным моделям или форматам данных.

**3. Описание практического задания**

Для проектируемой подсистемы необходимо определить список пользователей и их роли. Вот примерный список актеров и их ролей:

Администратор ­­­– ответственен за управление и настройку подсистемы.

Авторизованный пользователь – использует подсистему для выполнение своих задач.

Иерархия актеров для проектируемой подсистемы:

* Администратор
* Авторизованный пользователь

Для проектируемой подсистемы можно выделить следующие прецеденты:

* управление объектами базы данных – поведение системы, расширяющееся управлением абонементов и сотрудников,
* управление абонементами – поведение системы, расширяющееся созданием, редактированием и удалением абонементов,
* управление сотрудниками – поведение системы, расширяющееся созданием, редактированием и удалением сотрудников,
* добавление абонемента – действие создания нового абонемента,
* редактирование абонемента – действие редактирования абонемента,
* удаление абонемента – действие удаления абонемента,
* добавление сотрудника – действие создания нового сотрудника,
* редактирование сотрудника – действие редактирования сотрудника
* удаление сотрудника – действие удаления сотрудника,
* управление статуса об абонементе – заморозка/отмена абонемента,
* найти абонемент – действие, расширяющееся поиском абонемента, по ключевым словам, и фильтрацией списка абонементов,
* поиск абонемента – действие выбора желаемого абонемента из имеющегося списка,
* просмотр информации об абонементе – действие просмотра информации о выбранном абонементе,
* оформить абонемент – действие оставления заявки на оформление абонемента,
* просмотр личного кабинета – действие просмотра личного кабинета пользователя и получения определённой информации,
* просмотр истории оформленных абонементов – действие просмотра истории оформленных абонементов пользователем,
* изменение пользовательских данных – действие изменения, добавления или удаления личных данных о пользователе,
* отменить абонемент – действие отмены на оформление абонемента.

Взаимодействие актёров с «публичными» прецедентами в данной системе можно описать следующих образом:

* пользователь использует функциональность данной системы для выполнения собственных задач, таких как оформление абонемента.
* администратор использует функциональность данной системы для выполнения собственных задач, таких как управление объектами базы данных, абонементами, сотрудниками.

На рисунке 1 представлена разработанная Use Case диаграмма вариантов использование:

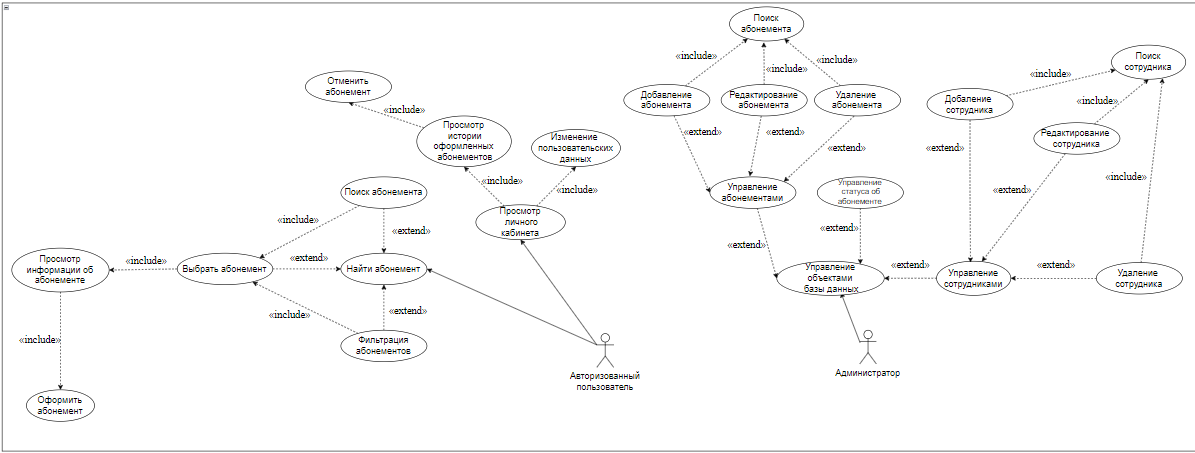


Рисунок 1 – Use Case диаграмма вариантов использованияs

**Вывод:**

По итогу разработанной работы была составлена диаграмма вариантов использования.