Пояснительная записка

к контрольному домашнему заданию по дисциплине “Программирование”.

**Тема проекта.**

Разработка информационной системы значимых событий.

**Адрес репозитория.**

<https://github.com/JohnSWD/Events-manager>

**Аннотация.**

Программа создана для управления информацией о различных событиях. Стоит отметить, что прошедшие события автоматически удаляются, поскольку информация о них не является актуальной. Взаимодействие пользователя и программы начинается с окна авторизации, где пользователь может: а) зарегистрироваться; б) авторизоваться; в) войти в приложение в качестве гостя. Для полного доступа к программе (редактирование, добавление и удаление событий) необходима авторизация. Тем не менее незарегистрированные пользователи имеют возможность просматривать события. В главном окне пользователь может взаимодействовать с событиями, осуществлять поиск по событиям. Из него можно попасть в окно с подробной информацией и редактирование, в окно, где создаются новые события, а также вернуть в окно с авторизацией (для не зарегистрировавшихся).

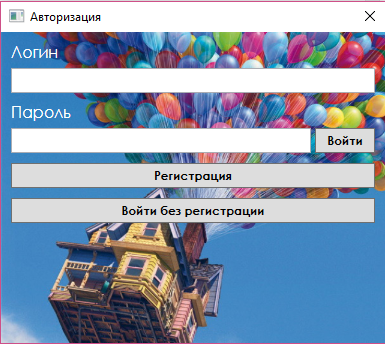
**Используемые технологии и библиотеки.**

Для получения доступа к дополнительным возможностям C# были подключены следующие библиотеки: System.IO, System.Xml.Serialization, System.ComponentModel, System.Security.Cryptography.

**Интерфейс программы.**

Окно авторизации (“Авторизация”).

Поля для логина и пароля проверяются на корректность (отсутствие специальных символов и совпадений (для логина), длина (от 6 до 12)). При корректных данных кнопки “Войти” и “Регистрация”, а также кнопка “Войти без регистрации” (введенная информация в полях не имеет значения) открывают главное окно.



Главное окно (“Список событий”).

В главном окне находятся: специальная область, в которой содержатся все добавленные события; три кнопки для взаимодействия с событиями (добавление, удаление, просмотр подробной информации); кнопка для возвращения к окну авторизации для не зарегистрировавшихся пользователей; строка поиска (невидима для авторизированных); область с информацией о событиях в целом; кнопка для возвращения к окну авторизации для неавторизированных пользователей. Нажав на кнопку "Добавить", пользователь переходит в новое окно, где может создать новое событие. При нажатии на кнопку "Подробнее" открывается окно с полной информацией о событии.

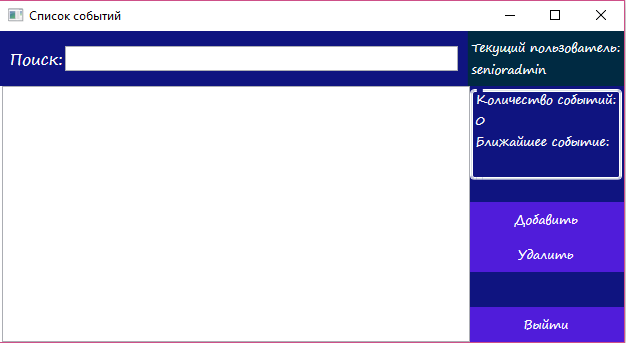
Окно добавления нового события (“Новое событие”).

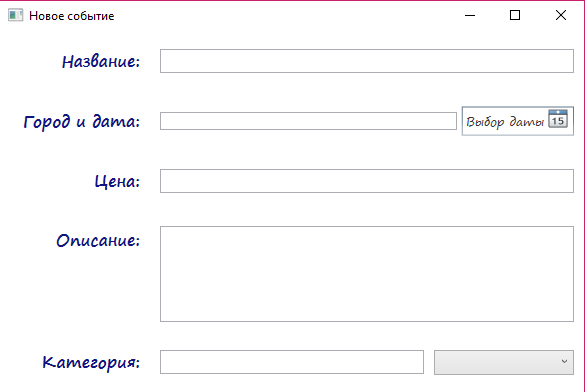
В данном окне пользователь может задать параметры для нового события. Поля проверяются на корректность ввода. Причем при создании события пользователь может либо выбрать категорию из существующего списка, либо создать собственную, если не нашел подходящую категорию в списке.

Окно с подробной информацией и редактированием (“Подробная информация о событии”).

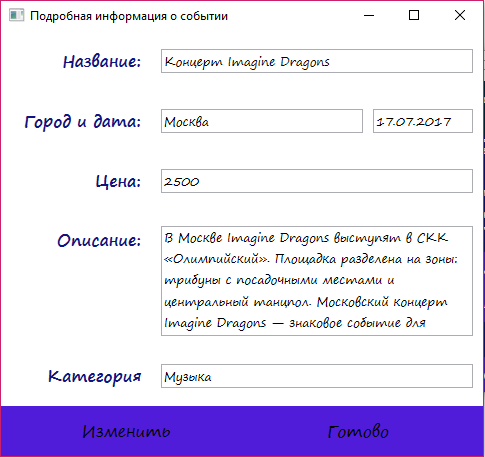
В этом окне пользователь может посмотреть полную информацию о событии, а также изменить ее (кроме данных о категории). Поля также защищены от некорректного ввода.

Окно добавления нового события





Главное окно



Окно с подробной информацией и редактированием

**Состав классов.**

В программе содержатся 6 классов: 2 из них являются сущностями, несущими информацию, оставшиеся являются средствами выполнения различных функций. Первая (главная) сущность - *Event* - содержит в себе важную информацию о событии: название, дата, место проведения, цена, описание. Вторая - *Category* – побочная сущность с названием и ID (каждому событию ставится в соответствие определенная категория (сюръекция)). Класс *User* необходим для авторизации пользователей (объекта класса хранит информацию об определенном пользователе: логин и хешированный пароль). *SrlztnTool* используется для сериализации информации о событиях и категориях (содержит два листа: один с событиями, другой с категориями). *SearchTool* нужен для динамического поиска (содержит конструктор, который при создании получает коллекцию объектов и объект класса TextBox). Класс для логгирования *LogginTool* (в классе находятся два метода: первый для логгирования информации об ошибках, второй - некоторых действий пользователя (приложения)).

**Список используемых источников.**

1. Герберт Шилдт C# 4.0: полное руководство;
2. Мэтью Мак-Дональд WPF 4: Windows Presentation Foundation в .NET 4.0;
3. <https://stackoverflow.com>;
4. <https://msdn.microsoft.com>