|  |
| --- |
| **Сервис продажи билетов**  **Руководство администратора** |

# Содержание

[Содержание 1](#_Toc489283945)

[Глоссарий 2](#_Toc489283946)

[Общие сведения 3](#_Toc489283947)

[Назначение сервиса 3](#_Toc489283948)

[Необходимые условия 3](#_Toc489283949)

[Покупка билетов 4](#_Toc489283950)

[Описание интерфейса продажи билетов 4](#_Toc489283951)

[Продажа 5](#_Toc489283952)

[Администрирование сервиса 6](#_Toc489283953)

[Настройка сервиса в ОС Windows 6](#_Toc489283954)

[Описание сервиса (файл web.config) 6](#_Toc489283955)

# Глоссарий

|  |  |
| --- | --- |
| WebApp | Веб-сервис, предоставляющий функции регистрации киносеансов |
| WpfHttpClient | Приложение для регистрации покупки билетов |
| URL | (англ. Uniform Resource Locator, URL) – единообразный определитель местонахождения (адрес) ресурса в вычислительной сети |

# Общие сведения

## Назначение сервиса

Сервис предназначен для регистрации киносеансов и продажи билетов.

## Необходимые условия

ОС Microsoft Windows 2008 или выше.

Установлен пакет .NET Framework 4.5.

Установлен пакет Visual C++ 2010 Redistributable Package.

## Покупка билетов

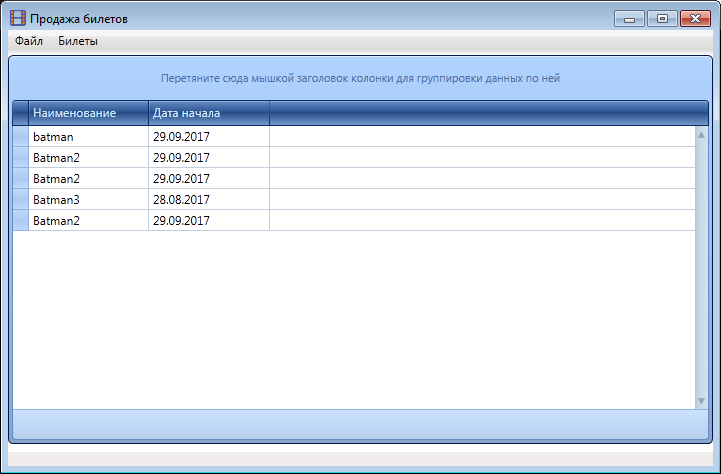
Вкладка «Действия» содержит описание настроек выгрузки и результирующего файла.

**Блок настроек «Настройка действия»** определяет общие настройки процедуры выгрузки данных: идентификатор кадра ретроспективы, шаблон имени и каталога файла, формат файла и т.д.

**Блок настроек «Настройка полей»** определяет состав дополнительных полей, которые необходимо включить в результирующий файл: наименование объекта, к которому принадлежит параметр, наименование параметра, наименование типа параметра, идентификатор параметра, идентификатор таблицы параметра, наименование архивной таблицы и т.д.

**Вкладка «Настройки запроса»** содержит параметры запроса данных из архивов комплекса для формирования выходного файла.

### Описание интерфейса продажи билетов



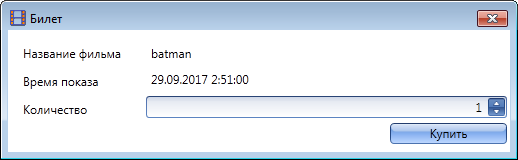
Действия

Выбор киносеанса

### Продажа

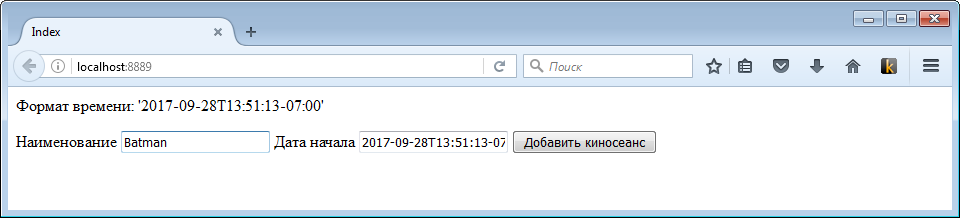
Выберите необходимый киносеанс, нажмите правой кнопкой, выбор пункта купить, либо через пункт меню – билеты.

Форма с билетами представлена ниже:



Выберите необходимое количество билетов.

### Регистрация киносеанса



Регистрация производится по кнопке «Добавить киносеанс»

Формат времени: {ГГ-ММ-ДД}T{ЧЧ:ММ:СС}'

# Администрирование сервиса

## Настройка сервиса в ОС Windows

### Описание сервиса (файл web.config)

Описание сервиса, необходимое для регистрации сервиса в консоли управление Службы Windows находится в файле ServiceDefinition.xml.

Где находится настройка подключения к БД

<add name="MovieEntities" connectionString="metadata=res://\*/Models.MovieModel.csdl|res://\*/Models.MovieModel.ssdl|res://\*/Models.MovieModel.msl;provider=System.Data.SqlClient;provider connection string=&quot;data source=(LocalDB)\v11.0;attachdbfilename=|DataDirectory|\Movie.mdf;integrated security=True;MultipleActiveResultSets=True;App=EntityFramework&quot;" providerName="System.Data.EntityClient" />

### Описание api

**GET запрос получения всех будующих киносеансов**

<http://ip-адрес> сервера:8889/api/movies

Возвращает JSON объект «Movie»

public string Guid { get; set; }

public string Name { get; set; }

public Nullable<System.DateTime> TimeBegin { get; set; }

Идентификатор GUID позволяет построить распределенную систему не заботясь об уникальности идентификатора.

**POST запрос бронирования выбранного киносеанса, с количеством билетов**

<http://ip-адрес> сервера:8889/api/movies

Принимает JSON объект “Ticket”

public string IdMovies { get; set; }

public Nullable<int> Count { get; set; }

**POST запрос контроллера регистрации киносеанса**

<http://ip-адрес> сервера:8889

Принимает JSON объект “Movie”

В будущем необходима обработка объекта билет в отдельной сущности

<http://ip-адрес> сервера:8889/api/tickets

И все манипуляции с киносеансами в отдельной сущности

<http://ip-адрес> сервера:8889/api/movies