МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ

ФЕДЕРАЦИИ

ФГБОУ ВО «ДонНТУ»

Факультет Интеллектуальных систем и программирования

Кафедра «Программная инженерия» им. Л.П. Фельдмана

Лабораторная работа №4

по курсу: «Профессиональная практика программной инженерии»

по теме: «Создание самодокументирующегося кода»

Выполнил:

ст. гр. ПИ-21в

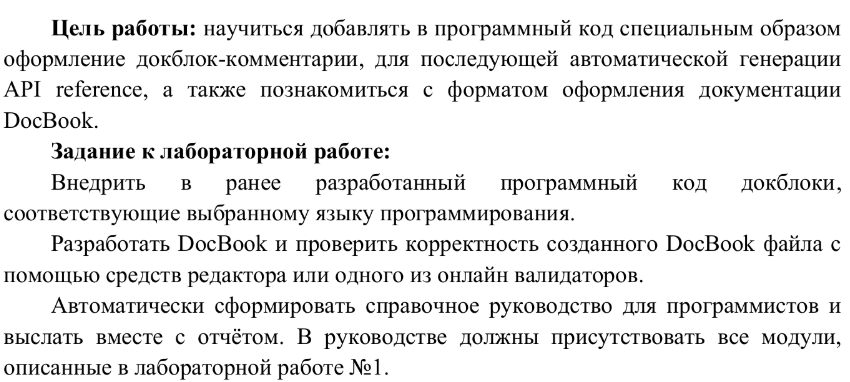
Рустамов В.Р.

Проверил:

асс. каф. ПИ

Филипишин Д.А.

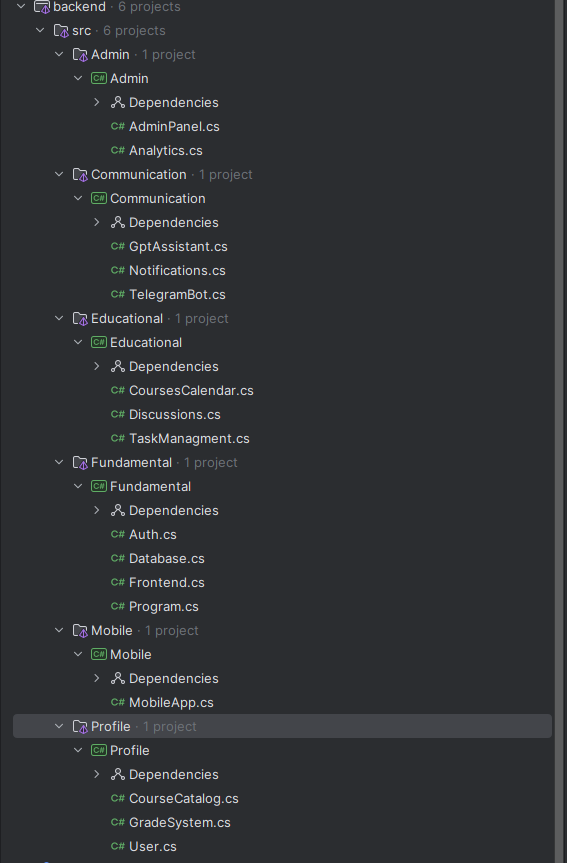
ДОНЕЦК – 2025



**Тема:** платформа для онлайн обучения.

**Репозиторий:** <https://github.com/McConderEz/PPPI>

**Структура проекта:**



**Листинг программы с докблоками:**

namespace Admin;  
  
/// <summary>  
/// Класс для управления административной панелью.  
/// </summary>  
public class AdminPanel  
{  
 /// <summary>  
 /// Управление пользователями: просмотр, добавление, удаление.  
 /// </summary>  
 public void ManageUsers()  
 {  
 Console.WriteLine("Управление пользователями: просмотр, добавление, удаление.");  
 }  
  
 /// <summary>  
 /// Управление курсами: создание, редактирование, удаление.  
 /// </summary>  
 public void ManageCourses()  
 {  
 Console.WriteLine("Управление курсами: создание, редактирование, удаление.");  
 }  
  
 /// <summary>  
 /// Просмотр статистики: количество пользователей, курсов, активность.  
 /// </summary>  
 public void ViewStatistics()  
 {  
 Console.WriteLine("Просмотр статистики: количество пользователей, курсов, активность.");  
 }  
}

namespace Admin;  
  
/// <summary>  
/// Класс для сбора и анализа данных платформы.  
/// </summary>  
public class Analytics  
{  
 /// <summary>  
 /// Сбор данных о пользователях: логи, активность, предпочтения.  
 /// </summary>  
 public void CollectUserData()  
 {  
 Console.WriteLine("Сбор данных о пользователях: логи, активность, предпочтения.");  
 }  
  
 /// <summary>  
 /// Анализ данных о курсах: популярность, завершенность, отзывы.  
 /// </summary>  
 public void AnalyzeCourseData()  
 {  
 Console.WriteLine("Анализ данных о курсах: популярность, завершенность, отзывы.");  
 }  
}

namespace Communication;  
  
/// <summary>  
/// Класс для сбора и анализа данных платформы.  
/// </summary>  
public class Analytics  
{  
 /// <summary>  
 /// Сбор данных о пользователях: логи, активность, предпочтения.  
 /// </summary>  
 public void CollectUserData()  
 {  
 Console.WriteLine("Сбор данных о пользователях: логи, активность, предпочтения.");  
 }  
  
 /// <summary>  
 /// Анализ данных о курсах: популярность, завершенность, отзывы.  
 /// </summary>  
 public void AnalyzeCourseData()  
 {  
 Console.WriteLine("Анализ данных о курсах: популярность, завершенность, отзывы.");  
 }  
}

namespace Communication;  
  
/// <summary>  
/// Класс для отправки уведомлений пользователям.  
/// </summary>  
public class Notifications  
{  
 /// <summary>  
 /// Отправка уведомления пользователю.  
 /// </summary>  
 /// <param name="userId">Идентификатор пользователя.</param>  
 /// <param name="message">Сообщение для отправки.</param>  
 public void SendNotification(int userId, string message)  
 {  
 Console.WriteLine($"Уведомление отправлено пользователю {userId}: {message}");  
 }  
}

namespace Communication;  
  
/// <summary>  
/// Класс для взаимодействия с Telegram-ботом.  
/// </summary>  
public class TelegramBot  
{  
 /// <summary>  
 /// Отправка сообщения через Telegram-бота.  
 /// </summary>  
 /// <param name="chatId">Идентификатор чата.</param>  
 /// <param name="message">Сообщение для отправки.</param>  
 public void SendMessage(long chatId, string message)  
 {  
 Console.WriteLine($"Сообщение отправлено в Telegram чат {chatId}: {message}");  
 }  
}

namespace Educational;  
  
/// <summary>  
/// Класс для управления календарем курсов.  
/// </summary>  
public class CoursesCalendar  
{  
 /// <summary>  
 /// Добавление события в календарь.  
 /// </summary>  
 /// <param name="eventName">Название события.</param>  
 /// <param name="eventDate">Дата события.</param>  
 public void AddEvent(string eventName, DateTime eventDate)  
 {  
 Console.WriteLine($"Событие добавлено в календарь: {eventName} на {eventDate}");  
 }  
  
 /// <summary>  
 /// Просмотр событий в календаре.  
 /// </summary>  
 public void ViewEvents()  
 {  
 Console.WriteLine("Просмотр событий в календаре.");  
 }  
}

namespace Educational;  
  
/// <summary>  
/// Класс для управления обсуждениями на форуме.  
/// </summary>  
public class Discussions  
{  
 /// <summary>  
 /// Создание нового обсуждения.  
 /// </summary>  
 /// <param name="topic">Тема обсуждения.</param>  
 public void CreateDiscussion(string topic)  
 {  
 Console.WriteLine($"Создано новое обсуждение: {topic}");  
 }  
  
 /// <summary>  
 /// Добавление сообщения в обсуждение.  
 /// </summary>  
 /// <param name="discussionId">Идентификатор обсуждения.</param>  
 /// <param name="message">Сообщение.</param>  
 public void AddMessage(int discussionId, string message)  
 {  
 Console.WriteLine($"Сообщение добавлено в обсуждение {discussionId}: {message}");  
 }  
}

namespace Educational;  
  
/// <summary>  
/// Класс для управления заданиями.  
/// </summary>  
public class TaskManagement  
{  
 /// <summary>  
 /// Создание нового задания.  
 /// </summary>  
 /// <param name="taskName">Название задания.</param>  
 public void CreateTask(string taskName)  
 {  
 Console.WriteLine($"Создано новое задание: {taskName}");  
 }  
  
 /// <summary>  
 /// Оценка задания.  
 /// </summary>  
 /// <param name="taskId">Идентификатор задания.</param>  
 /// <param name="grade">Оценка.</param>  
 public void GradeTask(int taskId, int grade)  
 {  
 Console.WriteLine($"Задание {taskId} оценено на {grade} баллов.");  
 }  
}

namespace Fundamental;  
  
using System;  
  
public class Auth  
{  
 /// <summary>  
 /// Метод для регистрации нового пользователя.  
 /// </summary>  
 /// <param name="username">Имя пользователя.</param>  
 /// <param name="password">Пароль пользователя.</param>  
 public void RegisterUser(string username, string password)  
 {  
 Console.WriteLine($"Пользователь {username} успешно зарегистрирован.");  
 }  
  
 /// <summary>  
 /// Метод для входа пользователя в систему.  
 /// </summary>  
 /// <param name="username">Имя пользователя.</param>  
 /// <param name="password">Пароль пользователя.</param>  
 public void LoginUser(string username, string password)  
 {  
 Console.WriteLine($"Пользователь {username} вошел в систему.");  
 }  
  
 /// <summary>  
 /// Метод для выполнения двухфакторной аутентификации.  
 /// </summary>  
 /// <param name="token">Токен для аутентификации.</param>  
 public void TwoFactorAuthentication(string token)  
 {  
 Console.WriteLine($"Двухфакторная аутентификация успешна. Токен: {token}");  
 }  
}

namespace Fundamental;  
  
/// <summary>  
/// Класс для управления базой данных.  
/// </summary>  
public class Database  
{  
 /// <summary>  
 /// Выполнение SQL-запроса.  
 /// </summary>  
 /// <param name="query">SQL-запрос.</param>  
 public void ExecuteQuery(string query)  
 {  
 Console.WriteLine($"Выполнен SQL-запрос: {query}");  
 }  
  
 /// <summary>  
 /// Обновление данных в таблице.  
 /// </summary>  
 /// <param name="table">Название таблицы.</param>  
 /// <param name="data">Данные для обновления.</param>  
 public void UpdateData(string table, object data)  
 {  
 Console.WriteLine($"Данные в таблице {table} обновлены.");  
 }  
}

namespace Fundamental;  
  
/// <summary>  
/// Класс для управления фронтендом платформы.  
/// </summary>  
public class Frontend  
{  
 /// <summary>  
 /// Отображение главной страницы.  
 /// </summary>  
 public void DisplayHomePage()  
 {  
 Console.WriteLine("Отображение главной страницы.");  
 }  
  
 /// <summary>  
 /// Отображение страницы курса.  
 /// </summary>  
 /// <param name="courseId">Идентификатор курса.</param>  
 public void DisplayCoursePage(int courseId)  
 {  
 Console.WriteLine($"Отображение страницы курса {courseId}.");  
 }  
}

namespace Mobile;  
  
/// <summary>  
/// Класс для управления мобильным приложением.  
/// </summary>  
public class MobileApp  
{  
 /// <summary>  
 /// Отображение списка курсов.  
 /// </summary>  
 public void DisplayCourses()  
 {  
 Console.WriteLine("Отображение списка курсов в мобильном приложении.");  
 }  
  
 /// <summary>  
 /// Выполнение задания.  
 /// </summary>  
 /// <param name="taskId">Идентификатор задания.</param>  
 public void CompleteTask(int taskId)  
 {  
 Console.WriteLine($"Задание {taskId} выполнено.");  
 }  
}

namespace Profile;  
  
/// <summary>  
/// Класс для управления каталогом курсов.  
/// </summary>  
public class CourseCatalog  
{  
 /// <summary>  
 /// Добавление нового курса.  
 /// </summary>  
 /// <param name="courseName">Название курса.</param>  
 public void AddCourse(string courseName)  
 {  
 Console.WriteLine($"Курс добавлен: {courseName}");  
 }  
  
 /// <summary>  
 /// Просмотр списка курсов.  
 /// </summary>  
 public void ViewCourses()  
 {  
 Console.WriteLine("Просмотр списка курсов.");  
 }  
}

namespace Profile;  
  
/// <summary>  
/// Класс для управления системой оценок.  
/// </summary>  
public class GradeSystem  
{  
 /// <summary>  
 /// Оценка курса.  
 /// </summary>  
 /// <param name="courseId">Идентификатор курса.</param>  
 /// <param name="grade">Оценка.</param>  
 public void GradeCourse(int courseId, int grade)  
 {  
 Console.WriteLine($"Курс {courseId} оценен на {grade} баллов.");  
 }  
  
 /// <summary>  
 /// Просмотр оценок.  
 /// </summary>  
 public void ViewGrades()  
 {  
 Console.WriteLine("Просмотр оценок.");  
 }  
}

namespace Profile;  
  
using System;  
  
public class User  
{  
 private string \_username;  
 private string \_bio;  
  
 /// <summary>  
 /// Метод для создания профиля пользователя.  
 /// </summary>  
 /// <param name="username">Имя пользователя.</param>  
 /// <param name="bio">Информация о пользователе.</param>  
 public void CreateProfile(string username, string bio)  
 {  
 \_username = username;  
 \_bio = bio;  
 Console.WriteLine($"Профиль пользователя {\_username} создан. Описание: {\_bio}");  
 }  
  
 /// <summary>  
 /// Метод для редактирования профиля пользователя.  
 /// </summary>  
 /// <param name="newBio">Новая информация о пользователе.</param>  
 public void EditProfile(string newBio)  
 {  
 \_bio = newBio;  
 Console.WriteLine($"Профиль пользователя {\_username} обновлен. Новое описание: {\_bio}");  
 }  
}

**Документация DocBook:**

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE book PUBLIC "-//OASIS//DTD DocBook XML V4.5//EN" "http://www.oasis-open.org/docbook/xml/4.5/docbookx.dtd">

<book>

<title>Документация платформы для онлайн-обучения</title>

<!-- Краткое описание проекта -->

<preface>

<title>Описание проекта</title>

<para>

Данная платформа предоставляет пользователям доступ к обучающим курсам, позволяя преподавателям загружать свои материалы и управлять курсами, а студентам — учиться в удобном для них темпе.

</para>

</preface>

<!-- Модули проекта -->

<chapter>

<title>Модули проекта</title>

<section>

<title>1. Регистрация и аутентификация пользователей</title>

<para>

Пользователи могут регистрироваться как студенты или преподаватели. Реализована возможность входа через социальные сети и двухфакторная аутентификация для повышения безопасности.

</para>

</section>

<section>

<title>2. Профили пользователей</title>

<para>

Пользователи могут создавать и редактировать свои профили, добавлять информацию о себе, своих интересах и навыках, что помогает другим участникам сообщества лучше их узнать.

</para>

</section>

<section>

<title>3. Каталог курсов</title>

<para>

Преподаватели могут создавать и управлять курсами, добавляя описание, материалы, видео и задания. Студенты могут просматривать доступные курсы и записываться на них.

</para>

</section>

<section>

<title>4. Система оценок и отзывов</title>

<para>

Студенты могут оставлять отзывы и оценки курсам и преподавателям. Это помогает другим пользователям выбрать подходящий курс и повышает качество обучения.

</para>

</section>

<section>

<title>5. Система управления заданиями</title>

<para>

Преподаватели могут создавать задания для студентов, а студенты могут загружать свои работы и получать оценки. Это способствует более глубокому пониманию материала.

</para>

</section>

<section>

<title>6. Форум и обсуждения</title>

<para>

Встроенный форум позволяет пользователям обсуждать курсы, задавать вопросы и делиться опытом, создавая активное сообщество вокруг обучения.

</para>

</section>

<section>

<title>7. Календарь курсов</title>

<para>

Календарь отображает расписание курсов, вебинаров и дедлайнов по заданиям, позволяя пользователям планировать свое время и не пропускать важные события.

</para>

</section>

<section>

<title>8. Система уведомлений</title>

<para>

Уведомления информируют пользователей о новых курсах, изменениях в расписании, ответах на вопросы и новых заданиях, что позволяет быть в курсе всех событий.

</para>

</section>

<section>

<title>9. Административная панель</title>

<para>

Модуль для администраторов, позволяющий управлять пользователями и курсами, просматривать статистику и проводить модерацию контента на платформе.

</para>

</section>

<section>

<title>10. Мобильное приложение</title>

<para>

Разработка мобильного приложения для доступа к платформе, просмотра курсов и выполнения заданий, что обеспечивает удобство использования на ходу.

</para>

</section>

<section>

<title>Заключение</title>

<para>

Эта платформа для онлайн-обучения будет способствовать развитию навыков пользователей, предоставляя доступ к качественным образовательным ресурсам. Каждый модуль будет улучшать функциональность и пользовательский опыт.

</para>

</section>

</chapter>

<!-- Документация по классам -->

<chapter>

<title>Модуль Admin</title>

<section>

<title>Класс AdminPanel</title>

<para>Класс для управления административной панелью.</para>

<section>

<title>Метод ManageUsers</title>

<para>Управление пользователями: просмотр, добавление, удаление.</para>

</section>

<section>

<title>Метод ManageCourses</title>

<para>Управление курсами: создание, редактирование, удаление.</para>

</section>

<section>

<title>Метод ViewStatistics</title>

<para>Просмотр статистики: количество пользователей, курсов, активность.</para>

</section>

</section>

<section>

<title>Класс Analytics</title>

<para>Класс для сбора и анализа данных платформы.</para>

<section>

<title>Метод CollectUserData</title>

<para>Сбор данных о пользователях: логи, активность, предпочтения.</para>

</section>

<section>

<title>Метод AnalyzeCourseData</title>

<para>Анализ данных о курсах: популярность, завершенность, отзывы.</para>

</section>

</section>

</chapter>

<chapter>

<title>Модуль Communication</title>

<section>

<title>Класс Notifications</title>

<para>Класс для отправки уведомлений пользователям.</para>

<section>

<title>Метод SendNotification</title>

<para>Отправка уведомления пользователю.</para>

<para>Параметры:</para>

<itemizedlist>

<listitem>

<para><emphasis>userId</emphasis>: Идентификатор пользователя.</para>

</listitem>

<listitem>

<para><emphasis>message</emphasis>: Сообщение для отправки.</para>

</listitem>

</itemizedlist>

</section>

</section>

<section>

<title>Класс TelegramBot</title>

<para>Класс для взаимодействия с Telegram-ботом.</para>

<section>

<title>Метод SendMessage</title>

<para>Отправка сообщения через Telegram-бота.</para>

<para>Параметры:</para>

<itemizedlist>

<listitem>

<para><emphasis>chatId</emphasis>: Идентификатор чата.</para>

</listitem>

<listitem>

<para><emphasis>message</emphasis>: Сообщение для отправки.</para>

</listitem>

</itemizedlist>

</section>

</section>

</chapter>

<chapter>

<title>Модуль Educational</title>

<section>

<title>Класс CoursesCalendar</title>

<para>Класс для управления календарем курсов.</para>

<section>

<title>Метод AddEvent</title>

<para>Добавление события в календарь.</para>

<para>Параметры:</para>

<itemizedlist>

<listitem>

<para><emphasis>eventName</emphasis>: Название события.</para>

</listitem>

<listitem>

<para><emphasis>eventDate</emphasis>: Дата события.</para>

</listitem>

</itemizedlist>

</section>

<section>

<title>Метод ViewEvents</title>

<para>Просмотр событий в календаре.</para>

</section>

</section>

<section>

<title>Класс Discussions</title>

<para>Класс для управления обсуждениями на форуме.</para>

<section>

<title>Метод CreateDiscussion</title>

<para>Создание нового обсуждения.</para>

<para>Параметры:</para>

<itemizedlist>

<listitem>

<para><emphasis>topic</emphasis>: Тема обсуждения.</para>

</listitem>

</itemizedlist>

</section>

<section>

<title>Метод AddMessage</title>

<para>Добавление сообщения в обсуждение.</para>

<para>Параметры:</para>

<itemizedlist>

<listitem>

<para><emphasis>discussionId</emphasis>: Идентификатор обсуждения.</para>

</listitem>

<listitem>

<para><emphasis>message</emphasis>: Сообщение.</para>

</listitem>

</itemizedlist>

</section>

</section>

<section>

<title>Класс TaskManagement</title>

<para>Класс для управления заданиями.</para>

<section>

<title>Метод CreateTask</title>

<para>Создание нового задания.</para>

<para>Параметры:</para>

<itemizedlist>

<listitem>

<para><emphasis>taskName</emphasis>: Название задания.</para>

</listitem>

</itemizedlist>

</section>

<section>

<title>Метод GradeTask</title>

<para>Оценка задания.</para>

<para>Параметры:</para>

<itemizedlist>

<listitem>

<para><emphasis>taskId</emphasis>: Идентификатор задания.</para>

</listitem>

<listitem>

<para><emphasis>grade</emphasis>: Оценка.</para>

</listitem>

</itemizedlist>

</section>

</section>

</chapter>

<!-- Продолжение для других модулей (Fundamental, Mobile, Profile) -->

</book>

**DOCX-версия документации, сгенерированная на сайте** [**https://pandoc.org/try/**](https://pandoc.org/try/)

Данная платформа предоставляет пользователям доступ к обучающим курсам, позволяя преподавателям загружать свои материалы и управлять курсами, а студентам — учиться в удобном для них темпе.

1. Модули проекта

1. Регистрация и аутентификация пользователей

Пользователи могут регистрироваться как студенты или преподаватели. Реализована возможность входа через социальные сети и двухфакторная аутентификация для повышения безопасности.

2. Профили пользователей

Пользователи могут создавать и редактировать свои профили, добавлять информацию о себе, своих интересах и навыках, что помогает другим участникам сообщества лучше их узнать.

3. Каталог курсов

Преподаватели могут создавать и управлять курсами, добавляя описание, материалы, видео и задания. Студенты могут просматривать доступные курсы и записываться на них.

4. Система оценок и отзывов

Студенты могут оставлять отзывы и оценки курсам и преподавателям. Это помогает другим пользователям выбрать подходящий курс и повышает качество обучения.

5. Система управления заданиями

Преподаватели могут создавать задания для студентов, а студенты могут загружать свои работы и получать оценки. Это способствует более глубокому пониманию материала.

6. Форум и обсуждения

Встроенный форум позволяет пользователям обсуждать курсы, задавать вопросы и делиться опытом, создавая активное сообщество вокруг обучения.

7. Календарь курсов

Календарь отображает расписание курсов, вебинаров и дедлайнов по заданиям, позволяя пользователям планировать свое время и не пропускать важные события.

8. Система уведомлений

Уведомления информируют пользователей о новых курсах, изменениях в расписании, ответах на вопросы и новых заданиях, что позволяет быть в курсе всех событий.

9. Административная панель

Модуль для администраторов, позволяющий управлять пользователями и курсами, просматривать статистику и проводить модерацию контента на платформе.

10. Мобильное приложение

Разработка мобильного приложения для доступа к платформе, просмотра курсов и выполнения заданий, что обеспечивает удобство использования на ходу.

Заключение

Эта платформа для онлайн-обучения будет способствовать развитию навыков пользователей, предоставляя доступ к качественным образовательным ресурсам. Каждый модуль будет улучшать функциональность и пользовательский опыт.

1. Модуль Admin

Класс AdminPanel

Класс для управления административной панелью.

# Метод ManageUsers

Управление пользователями: просмотр, добавление, удаление.

# Метод ManageCourses

Управление курсами: создание, редактирование, удаление.

# Метод ViewStatistics

Просмотр статистики: количество пользователей, курсов, активность.

Класс Analytics

Класс для сбора и анализа данных платформы.

# Метод CollectUserData

Сбор данных о пользователях: логи, активность, предпочтения.

# Метод AnalyzeCourseData

Анализ данных о курсах: популярность, завершенность, отзывы.

1. Модуль Communication

Класс Notifications

Класс для отправки уведомлений пользователям.

# Метод SendNotification

Отправка уведомления пользователю.

Параметры:

*userId*: Идентификатор пользователя.

*message*: Сообщение для отправки.

Класс TelegramBot

Класс для взаимодействия с Telegram-ботом.

# Метод SendMessage

Отправка сообщения через Telegram-бота.

Параметры:

*chatId*: Идентификатор чата.

*message*: Сообщение для отправки.

1. Модуль Educational

Класс CoursesCalendar

Класс для управления календарем курсов.

# Метод AddEvent

Добавление события в календарь.

Параметры:

*eventName*: Название события.

*eventDate*: Дата события.

# Метод ViewEvents

Просмотр событий в календаре.

Класс Discussions

Класс для управления обсуждениями на форуме.

# Метод CreateDiscussion

Создание нового обсуждения.

Параметры:

*topic*: Тема обсуждения.

# Метод AddMessage

Добавление сообщения в обсуждение.

Параметры:

*discussionId*: Идентификатор обсуждения.

*message*: Сообщение.

Класс TaskManagement

Класс для управления заданиями.

# Метод CreateTask

Создание нового задания.

Параметры:

*taskName*: Название задания.

# Метод GradeTask

Оценка задания.

Параметры:

*taskId*: Идентификатор задания.

*grade*: Оценка.