

**Комплект оценочных материалов по дисциплине
«Программирование игр и симуляторов»**

Задания закрытого типа

Задания закрытого типа на выбор правильного ответа

1. Выберите один правильный ответ

Какой язык программирования чаще всего используется для разработки мобильных приложений под Android?

А) Swift

Б) Kotlin

В) C#

Г) PHP

Правильный ответ: Б

Компетенции (индикаторы): ПК-2

2. Выберите один правильный ответ

Какой инструмент обычно используется для разработки приложений под iOS?

А) Android Studio

Б) Xcode

В) Visual Studio

Г) Eclipse

Правильный ответ: Б

Компетенции (индикаторы): ПК-2

3. Выберите один правильный ответ

Как называется основное окно, в котором пользователи взаимодействуют с мобильным приложением?

А) Экран

Б) Меню

В) Панель инструментов

Г) Контейнер

Правильный ответ: А

Компетенции (индикаторы): ПК-2

4. Выберите один правильный ответ

Какой компонент пользовательского интерфейса позволяет пользователю выбирать из нескольких вариантов?

- А) Кнопка
- Б) Поле ввода
- В) Переключатель
- Г) Выпадающий список

Правильный ответ: Г

Компетенции (индикаторы): ПК-2

Задания закрытого типа на установление соответствия

1. Установите правильное соответствие для каждого термина.

	Термин	Определение
1)	Wireframe	А) Визуальная часть приложения, с которой взаимодействует пользователь.
2)	UI (User Interface)	Б) Общий опыт и удобство использования приложения.
3)	Mockup	В) Черно-белый набросок интерфейса, отражающий структуру приложения.
4)	Prototyping	Г) Детализированный дизайн экрана с цветами и стилями.
5)	UX (User Experience)	Д) Интерактивная модель интерфейса, имитирующая работу приложения.

Правильный ответ:

1	2	3	4	5
В	А	Г	Д	Б

Компетенции (индикаторы): ПК-2

2. Установите соответствие между понятием и его описанием, выбрав правильные пары.

	Понятие	Описание
1)	Application	А) Компонент, который представляет независимое окно в многооконном приложении

- | | | | |
|----|------------------|----|---|
| 2) | Window | Б) | Механизм конфигурации и создания приложения MAUI |
| 3) | MainPage | В) | Основной контейнер для управления окнами в .NET MAUI |
| 4) | MauiAppBuilder | Г) | События, управляющие жизненным циклом приложения и его окон |
| 5) | Lifecycle Events | Д) | Основная страница, которая отображается при запуске однооконного приложения |

Правильный ответ:

1	2	3	4	5
В	А	Д	Б	Г

Компетенции (индикаторы): ПК-2

3. Установите соответствие между API .NET MAUI и его функционалом, выбрав правильные пары.

- | | API .NET MAUI | | функционал |
|----|---------------|----|---|
| 1) | Permissions | А) | Предоставляет информацию о текущем местоположении устройства |
| 2) | Geolocation | Б) | Позволяет выбрать или сделать фото/видео с устройства |
| 3) | MediaPicker | В) | Фиксирует изменения ускорения устройства по осям X, Y и Z |
| 4) | Accelerometer | Г) | Отвечает за управление вибрацией устройства |
| 5) | Vibration | Д) | Позволяет запрашивать доступ к камере, геолокации, микрофону и другим функциям устройства |

Правильный ответ:

1	2	3	4	5
Д	А	Б	В	Г

Компетенции (индикаторы): ПК-2

Задания закрытого типа на установление правильной последовательности

1. Запрос разрешений перед использованием функционала устройства. Расположите шаги в правильном порядке:

- А) Использование функционала, если разрешение получено
- Б) Проверка, было ли ранее предоставлено разрешение
- В) Обработка случая отказа в разрешении
- Г) Запрос разрешения у пользователя

Правильные ответы: Б, Г, А, В.

Компетенции (индикаторы): ПК-2

2. Получение текущего местоположения с помощью Geolocation API. Расположите шаги в правильном порядке:

- А) Отображение местоположения на карте или передача данных
- Б) Обработка полученных координат
- В) Проверка доступности службы геолокации
- Г) Запрос текущего местоположения

Правильные ответы: В, Г, Б, А.

Компетенции (индикаторы): ПК-2

3. Выбор фото с устройства с помощью MediaPicker. Расположите шаги в правильном порядке:

- А) Отображение изображения в приложении
- Б) Ожидание завершения выбора файла
- В) Вызов MediaPicker.PickPhotoAsync()
- Г) Загрузка и обработка изображения

Правильные ответы: В, Б, Г, А.

Компетенции (индикаторы): ПК-2

Задания открытого типа

Задания открытого типа на дополнение

1. Напишите пропущенное слово (словосочетание).

В .NET MAUI для работы с разрешениями используется класс _____, который позволяет запрашивать доступ к функциям устройства.

Правильный ответ: Permissions

Компетенции (индикаторы): ПК-2

2. Напишите пропущенное слово (словосочетание).

Чтобы получить текущее местоположение пользователя, используется _____, который предоставляет данные о координатах устройства.

Правильный ответ: Geolocation API

Компетенции (индикаторы): ПК-2

3. Напишите пропущенное слово (словосочетание).

Метод _____ позволяет устройству включить вибрацию на заданное время.

Правильный ответ: Vibration.Vibrate()

Компетенции (индикаторы): ПК-2

4. Напишите пропущенное слово (словосочетание).

В MediaPlayer используется метод _____, который позволяет пользователю выбрать изображение из галереи.

Правильный ответ: PickPhotoAsync()

Компетенции (индикаторы): ПК-2

5. Напишите пропущенное слово (словосочетание).

В .NET MAUI основной класс приложения называется _____, и он отвечает за жизненный цикл программы.

Правильный ответ: Application

Компетенции (индикаторы): ПК-2

6. Напишите пропущенное слово (словосочетание).

Метод _____ запускает отслеживание ускорения устройства по осям X, Y и Z.

Правильный ответ: Accelerometer.Start()

Компетенции (индикаторы): ПК-2

7. Напишите пропущенное слово (словосочетание).

В .NET MAUI для отправки и обработки сообщений между компонентами приложения применяется механизм _____, обеспечивающий взаимодействие объектов.

Правильный ответ: MessagingCenter

Компетенции (индикаторы): ПК-2

8. Напишите пропущенное слово (словосочетание).

Пространство имен _____ включает в себя API для работы с сенсорами, такими как гироскоп и акселерометр.

Правильный ответ: Microsoft.Maui.Essentials

Компетенции (индикаторы): ПК-2

Задания открытого типа с кратким свободным ответом

1. Какие контейнеры компоновки используются в .NET MAUI для верстки экранов?

Правильный ответ: Grid / StackLayout / FlexLayout / AbsoluteLayout

Компетенции (индикаторы): ПК-2

2. Какие способы навигации поддерживает .NET MAUI?

Правильный ответ: Shell Navigation / NavigationPage / Modal Navigation

Компетенции (индикаторы): ПК-2

3. Какие базовые классы используются для создания пользовательского интерфейса в .NET MAUI?

Правильный ответ: ContentPage / Shell / TabbedPage / FlyoutPage

Компетенции (индикаторы): ПК-2

4. Какие элементы управления можно использовать для отображения списка данных?

Правильный ответ: CollectionView / ListView / TableView

Компетенции (индикаторы): ПК-2

Задания открытого типа с развернутым ответом

1. Создайте простое мобильное приложение на .NET MAUI, которое отслеживает количество шагов пользователя с помощью акселерометра и отображает их на экране.

Требования:

Создать интерфейс с одной кнопкой Старт / Стоп и текстовым полем, отображающим количество шагов.

Использовать акселерометр для определения движения пользователя.

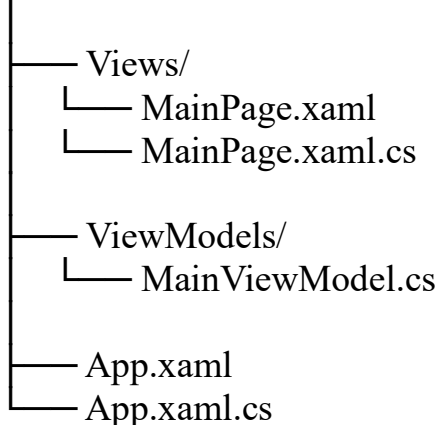
При нажатии на кнопку:

- Если акселерометр выключен — включить его.
- Если акселерометр включен — выключить его.

- Обновлять счетчик шагов при движении устройства.
- Подсказки:
- Используйте Accelerometer.ReadingChanged для обработки данных.
- Учитывайте только значительные изменения в показаниях для корректного увеличения шагов.

Ожидаемый результат: мобильное приложение соответствующее требованиям.

StepCounterApp/



Views/MainPage.xaml

```

<ContentPage xmlns="http://schemas.microsoft.com/dotnet/2021/maui"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    xmlns:vm="clr-namespace:StepCounterApp.ViewModels"
    x:Class="StepCounterApp.Views.MainPage">
    <ContentPage.BindingContext>
        <vm:MainViewModel />
    </ContentPage.BindingContext>
    <VerticalStackLayout Padding="30" Spacing="20">
        <Label Text="Steps:" FontSize="32" HorizontalOptions="Center" />
        <Label Text="{Binding StepCount}" FontSize="48"
HorizontalOptions="Center" />
        <Button Text="Start Tracking" Command="{Binding StartCommand}" />
        <Button Text="Stop Tracking" Command="{Binding StopCommand}" />
    </VerticalStackLayout>
</ContentPage>
  
```

ViewModels/MainViewModel.cs

```

using Microsoft.Maui.Devices.Sensors;
using System.ComponentModel;
using System.Windows.Input;
namespace StepCounterApp.ViewModels;
public class MainViewModel : INotifyPropertyChanged
{
    public event PropertyChangedEventHandler PropertyChanged;
    private int _stepCount;
    private AccelerometerData _lastReading;
  
```

```

private bool _isTracking;
public int StepCount
{
    get => _stepCount;
    set
    {
        _stepCount = value;
        OnPropertyChanged(nameof(StepCount));
    }
}
public ICommand StartCommand => new Command(StartTracking);
public ICommand StopCommand => new Command(StopTracking);
public MainViewModel()
{
    Accelerometer.ReadingChanged += Accelerometer_ReadingChanged;
}
private void StartTracking()
{
    if (!_isTracking)
    {
        Accelerometer.Start(SensorSpeed.UI);
        _isTracking = true;
    }
}
private void StopTracking()
{
    if (_isTracking)
    {
        Accelerometer.Stop();
        _isTracking = false;
    }
}
private void Accelerometer_ReadingChanged(object sender,
AccelerometerChangedEventArgs e)
{
    var acc = e.Reading;
    var magnitude = Math.Sqrt(acc.Acceleration.X * acc.Acceleration.X +
        acc.Acceleration.Y * acc.Acceleration.Y +
        acc.Acceleration.Z * acc.Acceleration.Z);
    // Простейшая эвристика: шаг, если "толчок" выше 1.2g
    if (magnitude > 1.2)
    {
        // Простой антидребезг: шаги не чаще 400 мс
        var now = DateTime.UtcNow;

```



```

        if (_lastStepTime == null || (now -
_lastStepTime.Value).TotalMilliseconds > 400)
        {
            StepCount++;
            _lastStepTime = now;
        }
    }
}
private DateTime? _lastStepTime;
protected void OnPropertyChanged(string propertyName) =>
    PropertyChanged?.Invoke(this, new
PropertyChangedEventArgs(propertyName));
}
App.xaml.cs
public partial class App : Application
{
    public App()
    {
        InitializeComponent();
        MainPage = new NavigationPage(new
StepCounterApp.Views.MainPage());
    }
}

```

Для Android добавь в Platforms/Android/AndroidManifest.xml:

```
<uses-permission android:name="android.permission.BODY_SENSORS" />
```

Критерии оценивания:

- приложение соответствует паттерну MVVM;
- используются только те разрешения, которые необходимы приложению;
- приложение поддерживает Android 34 API;
- приложение поддерживает iOS 18;
- используются графические изображения иконок/фонов.

Компетенции (индикаторы): ПК-2

2. Разработайте мобильное приложение "Заметки" на .NET MAUI, которое позволяет пользователю добавлять и удалять текстовые заметки.

Ожидаемый результат:

NotesApp/

```

├── Models/
│   └── Note.cs
├── Views/
│   └── MainPage.xaml

```

```

├── MainPage.xaml.cs
├── ViewModels/
│   └── MainViewModel.cs
├── App.xaml
└── App.xaml.cs
Models/Note.cs
namespace NotesApp.Models;
public class Note
{
    public string Text { get; set; }
}
ViewModels/MainViewModel.cs
using System.Collections.ObjectModel;
using System.Windows.Input;
using NotesApp.Models;
namespace NotesApp.ViewModels;
public class MainViewModel : BindableObject
{
    private string _newNoteText;
    public string NewNoteText
    {
        get => _newNoteText;
        set
        {
            _newNoteText = value;
            OnPropertyChanged();
        }
    }
    public ObservableCollection<Note> Notes { get; } = new();
    public ICommand AddNoteCommand => new Command(AddNote);
    public ICommand DeleteNoteCommand => new
Command<Note>(DeleteNote);
    private void AddNote()
    {
        if (!string.IsNullOrEmpty(NewNoteText))
        {
            Notes.Add(new Note { Text = NewNoteText });
            NewNoteText = string.Empty;
        }
    }
    private void DeleteNote(Note note)
    {

```

```

        if (Notes.Contains(note))
        {
            Notes.Remove(note);
        }
    }
}
Views/MainPage.xaml
<ContentPage xmlns="http://schemas.microsoft.com/dotnet/2021/maui"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    xmlns:vm="clr-namespace:NotesApp.ViewModels"
    x:Class="NotesApp.Views.MainPage">
    <ContentPage.BindingContext>
        <vm:MainViewModel />
    </ContentPage.BindingContext>
    <VerticalStackLayout Padding="20">
        <Entry Placeholder="Введите заметку"
            Text="{Binding NewNoteText}" />
        <Button Text="Добавить"
            Command="{Binding AddNoteCommand}" />
        <CollectionView ItemsSource="{Binding Notes}">
            <CollectionView.ItemTemplate>
                <DataTemplate>
                    <SwipeView>
                        <SwipeView.RightItems>
                            <SwipeItems>
                                <SwipeItem Text="Удалить"
                                    BackgroundColor="Red"
                                    Command="{Binding Source={RelativeSource
AncestorType={x:Type vm:MainViewModel}}, Path=DeleteNoteCommand}"
                                    CommandParameter="{Binding .}" />
                            </SwipeItems>
                        </SwipeView.RightItems>
                    </SwipeView>
                    <Frame Margin="5"
                        Padding="10"
                        BorderColor="LightGray"
                        CornerRadius="5">
                        <Label Text="{Binding Text}" />
                    </Frame>
                </DataTemplate>
            </CollectionView.ItemTemplate>
        </CollectionView>
    </VerticalStackLayout>
</ContentPage>

```

```
App.xaml.cs
public partial class App : Application
{
    public App()
    {
        InitializeComponent();
        MainPage = new NavigationPage(new NotesApp.Views.MainPage());
    }
}
```

Критерии оценивания:

- Поля для ввода текста заметки;
 - Кнопки Добавить для сохранения заметки;
 - Списка всех сохраненных заметок;
 - Кнопки Удалить рядом с каждой заметкой;
 - При нажатии на Добавить заметка сохраняется и отображается в списке;
 - При нажатии на Удалить заметка удаляется;
 - Использовать Preferences или FileSystem.AppDataDirectory для сохранения списка заметок.
 - Приложение соответствует паттерну MVVM;
 - Используются только те разрешения, которые необходимы приложению;
 - Приложение поддерживает Android 34 API;
 - Приложение поддерживает iOS 18;
 - Используются графические изображения иконок/фонов.
- Компетенции (индикаторы): ПК-2

Экспертное заключение

Представленный комплект оценочных материалов по дисциплине «Программирование игр и симуляторов» соответствует требованиям ФГОС ВО.

Предлагаемые оценочные материалы адекватны целям и задачам реализации основной профессиональной образовательной программы по направлению подготовки 09.04.04 Программная инженерия.

Виды оценочных средств, включенные в представленный фонд, отвечают основным принципам формирования ФОС.

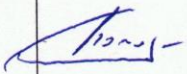
Разработанные и представленные для экспертизы оценочные материалы рекомендуются к использованию в процессе подготовки обучающихся по указанному направлению.

Председатель учебно-методической
комиссии института компьютерных
систем и информационных технологий



Ветрова Н.Н.

Лист изменений и дополнений

№ п/п	Виды дополнений и изменений	Дата и номер протокола заседания кафедры (кафедр), на котором были рассмотрены и одобрены изменения и дополнения	Подпись (с расшифровкой) заведующего кафедрой (заведующих кафедрами)
1.	Дополнен комплект оценочных материалов	протокол заседания кафедры компьютерных систем и сетей № <u>8</u> от <u>10.03.2025</u>	 С.В. Попов