

УДК 004

## ЭТАПЫ РАЗРАБОТКИ МОБИЛЬНОГО ПРИЛОЖЕНИЯ С ПРИМЕНЕНИЕМ ТЕХНОЛОГИЙ ДОПОЛНЕННОЙ РЕАЛЬНОСТИ ДЛЯ ПРОГРАММНО-АППАРАТНОГО КОМПЛЕКСА ПО ПРОМЫШЛЕННОЙ БЕЗОПАСНОСТИ

**А. И. БУТЬКО, М. В. ИЗОИТКО, В. А. МАЛЫШКИН**  
(Представлено: И. С. РУСЕЦКИЙ)

*Визуализация нештатных ситуаций в программно-аппаратном комплексе по промышленной безопасности реализована с применением технологий дополненной реальности. Мобильное приложение дополненной реальности идентифицирует объекты нефтехимического промышленного предприятия, воспроизводит анимированные треки аварий на различных производствах предприятия и проводит проверку знаний при возникновении нештатной ситуации техногенного характера.*

**Введение.** Мобильное приложение дополненной реальности и проверки знаний создано с помощью платформы Unity [1] на языке С# [2] для интерактивного обучения и тестирования пользователей.

Для обнаружения потенциальной нештатной ситуации техногенного характера нужно вывести камеру мобильного устройства на объекты, составляющие модель нефтехимического промышленного предприятия.

В случае правильной идентификации опасности на экране мобильного устройства воспроизводится анимированный трек, визуализирующий возникновение и развитие аварии. Затем предлагается набор нескольких возможных алгоритмов действий, из которых нужно выбрать единственно правильный вариант. Обучающий сеанс считается пройденным успешно, если обнаружены все опасности, предусмотренные сценарием, и для каждой из них определен правильный алгоритм действий.

### **Этапы разработки приложения.**

#### **1. Подготовка 3D-моделей и их интеграция**

Объекты макета нефтеперерабатывающего предприятия были отсканированы с помощью Polycam, что позволило создать 3D-модели с высокой точностью. Затем, эти модели были загружены в Model Target Generator для обучения и создания базы данных моделей, которая поддерживается Vuforia. После завершения этого процесса обученная модель была экспортирована и интегрирована в Unity.

#### **2. Настройка Vuforia**

На следующем этапе в проект Unity был добавлен фреймворк Vuforia. Для этого было выполнено:

– Создание лицензии Vuforia. Чтобы Vuforia могла работать в приложении, нужно было зарегистрироваться на сайте разработчиков Vuforia и получить лицензионный ключ, который позже был добавлен в настройки Unity.

– Установка Vuforia SDK. Для начала работы с Vuforia необходимо было установить соответствующий пакет в Unity. Vuforia Engine SDK была добавлена через Unity Package Manager.

– Включение Vuforia в разделе "Player Settings".

– Добавление ключа лицензии для активации Vuforia в проекте.

– Импорт обученной модели через Vuforia Model Target.

– Настройка ARCamera, которая используется для распознавания объектов дополненной реальности. Камера была добавлена на сцену, и ее параметры были настроены для корректной работы с панелями AR.

#### **3. Создание интерфейса и AR панелей.**

Проект состоит из пяти AR панелей, каждая из которых отображает свою уникальную инструкцию и работает с определенной 3D-моделью. В MainCanvas добавлена MainMenuPanel, который содержит кнопку "Start". Когда пользователь нажимает кнопку, происходит переключение на первую AR панель.

Структура AR панелей: каждая панель содержит такие элементы, как текстовые инструкции, таймер обратного отсчета, кнопки для взаимодействия и тесты.

#### **Реализация AR панелей.**

Инструкции для пользователя. Каждая панель содержит InstructionText, который выводит на экран текстовые инструкции для пользователя. Например, в первой панели текст отображает: «Найдите первое производство». Это помогает направлять пользователя по сцене и указывает, что он должен сделать.

Отсчет времени. На каждой панели есть CountdownText, который выводит таймер. Таймер начинается обратный отсчет с 20 секунд после того, как пользователь активирует камеру. Это создает ощущение срочности и дает пользователю ограниченное время на взаимодействие с дополненной реальностью.

AR эффекты. Когда камера распознает объект, происходит триггер отображения 3D-эффектов (в данном случае это: огонь, огонь с дымом, дым). Эффекты появляются только после успешного распознавания.

AR панели были созданы как отдельные панели внутри GameObject, который содержится внутри MainCanvas. Каждая панель включает несколько UI-элементов:

--InstructionText — текстовое поле, которое выводит инструкцию, например, "Найдите первое производство".

--CountdownText — таймер обратного отсчета, который показывает оставшееся время до завершения задания.

--QuestionPanel — панель с тестовым вопросом и двумя кнопками для ответов.

--ARCamera — общая камера, которая будет работать для всех AR панелей, находящихся на сцене. Она настроена для распознавания ModelTarget.

4. Связь панелей и логика переключений.

Для управления переключениями между панелями был разработан скрипт MenuController. Скрипт отвечает: за переключение между MainMenuPanel и AR панелями, активирование таймера на каждой панели, возврат к главному меню после завершения всех тестов.

```

```csharp
public class MenuController : MonoBehaviour
{
    public GameObject MainMenuPanel;
    public GameObject[] ARPanels;
    public void ShowARPanel(int panelIndex)
    {
        MainMenuPanel.SetActive(false);
        for (int i = 0; i < ARPanels.Length; i++)
        {
            ARPanels[i].SetActive(i == panelIndex);
        }
    }
    public void ShowMainMenu()
    {
        foreach (var panel in ARPanels)
        {
            panel.SetActive(false);
        }
        MainMenuPanel.SetActive(true);
    }
}
```

```

Этот скрипт позволяет переходить от одной панели к другой по очереди. В Unity Inspector настроен OnClick для кнопки StartButton с объектом MainMenuPanel и методом ShowARPanel(0).

5. Реализация таймера и эффектов.

Для каждой AR панели был настроен таймер обратного отсчета длиной в 20 секунд. Таймер отображается через TextMeshPro. Когда время истекает, пользователю предлагается ответить на вопрос с помощью двух кнопок.

Эффекты, такие как всплывающие подсказки и анимации 3D-моделей, были импортированы из Unity Asset Store. Они активируются, когда камера распознает соответствующую модель на сцене.

6. Настройка тестов.

Каждая AR панель содержит QuestionPanel с тестом. Реализовано два варианта ответов, представленных кнопками. При выборе ответа запускается обработка, которая выделяет правильный или неправильный вариант цветом:

- Зеленый для правильного ответа.
- Красный для неправильного ответа.
- 

```

```csharp
public class QuizController : MonoBehaviour
{
    public Button AnswerButton1;

```

```
public Button AnswerButton2;
public void SelectAnswer(Button button, bool isCorrect)
{
    if (isCorrect)
    {
        button.image.color = Color.green;
    }
    else
    {
        button.image.color = Color.red;
    }
}
}
...
```

В Inspector добавлен вызов этого метода в OnClick каждой кнопки, при этом параметры передаются в зависимости от того, является ли ответ правильным или нет.

#### 7. Подведение итогов и перезапуск.

После того как пользователь проходит все пять тестов, отображается итоговая панель ResultPanel, на которой выводится количество правильных ответов. На этой панели также есть кнопка RestartButton, которая возвращает пользователя к началу приложения и позволяет пройти тест заново.

```
...csharp
public class ResultController : MonoBehaviour
{
    public TextMeshProUGUI ResultText;
    public void ShowResults(int correctAnswers)
    {
        ResultText.text = $"Правильных ответов: {correctAnswers}/5";
    }
    public void Restart()
    {
        // Перезагрузка сцены
        UnityEngine.SceneManagement.SceneManager.LoadScene(0);
    }
}
...
```

Этот скрипт позволяет обрабатывать вывод результатов и перезагружать приложение по нажатию кнопки "Начать тест заново".

**Заключение.** В программно-аппаратном комплексе по промышленной безопасности технологии дополненной реальности позволяют реализовать визуализацию аварийных ситуаций техногенного характера.

Мобильное приложение дополненной реальности состоит из нескольких AR панелей, которые работают по единой логике, взаимодействуют с камерой устройства, идентифицируют объекты нефтехимического промышленного предприятия, воспроизводят анимированные треки аварий на различных производствах предприятия и запускают тесты для пользователей.

#### ЛИТЕРАТУРА

1. Платформа Unity [Электронный ресурс].– Режим доступа: <https://unity.com/ru>.– Дата доступа: 01.10.2024.
2. Руководство по C# [Электронный ресурс].– Режим доступа: <https://learn.microsoft.com/ru-ru/dotnet/csharp/>.– Дата доступа: 01.10.2024.