**Отчет по Главе 12: Жесты (Gestures)**

**1. Введение**

Данная глава посвящена распознавателям жестов в .NET MAUI. Жесты являются ключевым элементом современного пользовательского интерфейса, особенно на сенсорных устройствах, позволяя пользователям взаимодействовать с приложением интуитивно понятными способами, такими как касания, смахивания, масштабирование и перетаскивание. .NET MAUI предоставляет набор распознавателей жестов (GestureRecognizer), которые можно прикреплять к любому элементу управления, унаследованному от View, для добавления интерактивности.

**2. Основная часть**

**2.1. Основные концепции и принципы**

Распознаватели жестов добавляются в коллекцию GestureRecognizers любого элемента View.

* **TapGestureRecognizer**: Распознает одно или несколько касаний (тапов). Основное событие: Tapped. Ключевое свойство: NumberOfTapsRequired.
* **PinchGestureRecognizer**: Распознает жест "щипок" для масштабирования. Основное событие: PinchUpdated, которое срабатывает при начале, выполнении и завершении жеста.
* **PanGestureRecognizer**: Распознает жест перетаскивания (панорамирования). Основное событие: PanUpdated. Свойство e.TotalX и e.TotalY содержат смещение.
* **SwipeGestureRecognizer**: Распознает быстрый жест смахивания в определенном направлении (Direction). Основное событие: Swiped.
* **PointerGestureRecognizer**: Распознает действия, связанные с указателем (например, курсором мыши). Основные события: PointerEntered (указатель вошел в границы), PointerExited (указатель покинул границы).

**2.2. Примеры кода с пояснениями**

Демонстрация работы с жестами была реализована на странице GesturesDemoPage.xaml. Различные распознаватели были применены к элементам Image и Frame.

<!-- Фрагменты из GesturesDemoPage.xaml -->

<!-- Tap Gestures (с разными обработчиками) -->

<Image x:Name="SingleTapImage" Source="dotnet\_bot.png">

<Image.GestureRecognizers>

<TapGestureRecognizer Tapped="OnSingleTap" />

</Image.GestureRecognizers>

</Image>

<Frame x:Name="DoubleTapFrame">

<Frame.GestureRecognizers>

<TapGestureRecognizer Tapped="OnDoubleTap" NumberOfTapsRequired="2"/>

</Frame.GestureRecognizers>

<!-- ... -->

</Frame>

<!-- Pan Gesture -->

<Image x:Name="PanImage" Source="dotnet\_bot.png">

<Image.GestureRecognizers>

<PanGestureRecognizer PanUpdated="OnPanUpdated" />

</Image.GestureRecognizers>

</Image>

*Пояснение:* Распознаватели жестов добавляются в коллекцию GestureRecognizers элемента. Каждому событию (например, Tapped, PanUpdated) сопоставляется метод-обработчик в code-behind.

**Code-behind (GesturesDemoPage.xaml.cs)** содержит логику для обработки этих событий.

// Фрагменты из GesturesDemoPage.xaml.cs

private void OnSingleTap(object sender, TappedEventArgs e)

{

EventOutputLabel.Text = "Single Tap on Image detected!";

}

private void OnDoubleTap(object sender, TappedEventArgs e)

{

EventOutputLabel.Text = "Double Tap on Frame detected!";

}

private void OnPanUpdated(object sender, PanUpdatedEventArgs e)

{

if (sender is not View view) return;

switch (e.StatusType)

{

case GestureStatus.Running:

view.TranslationX = \_xOffset + e.TotalX;

view.TranslationY = \_yOffset + e.TotalY;

break;

// ...

}

}

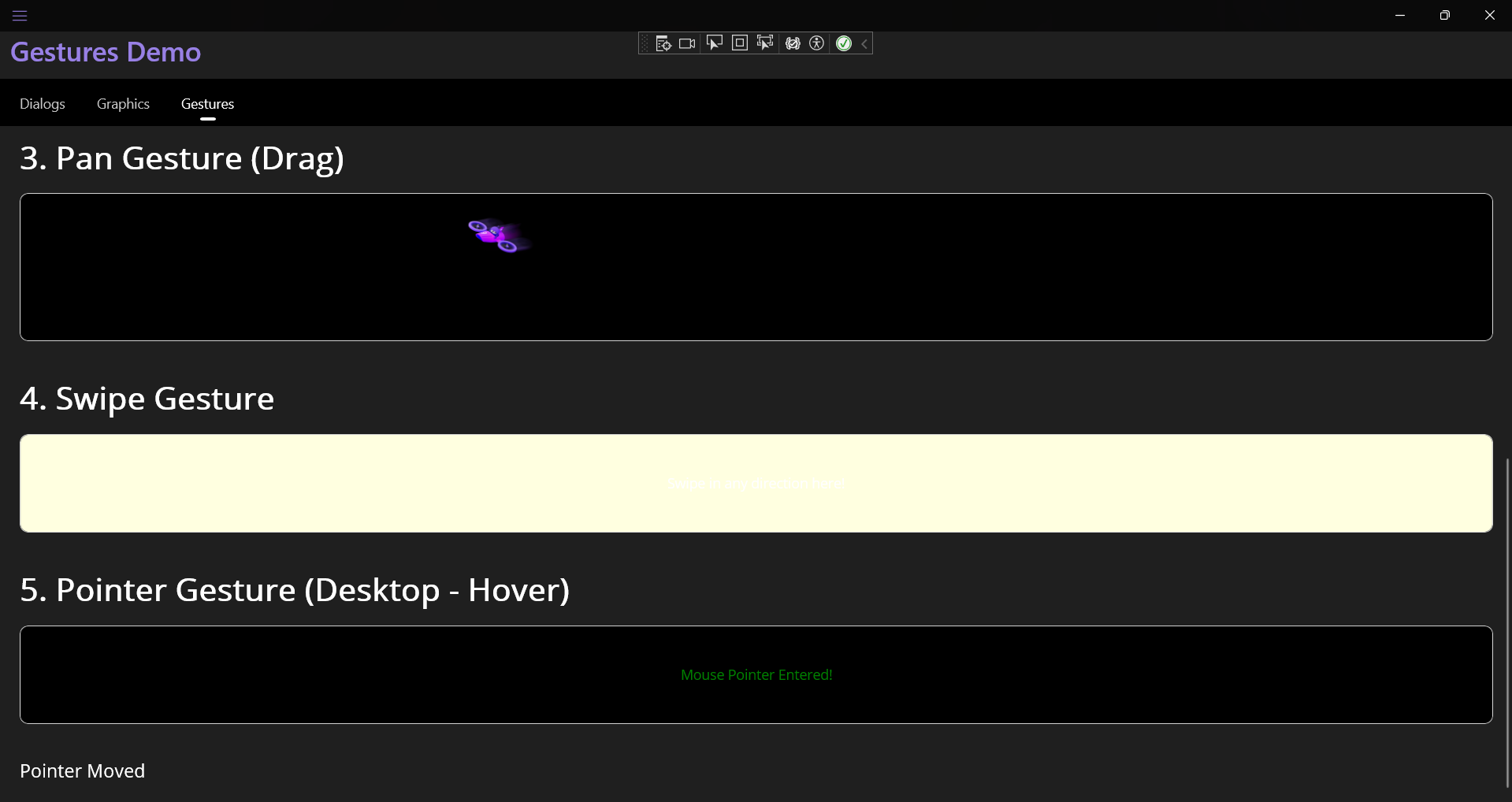
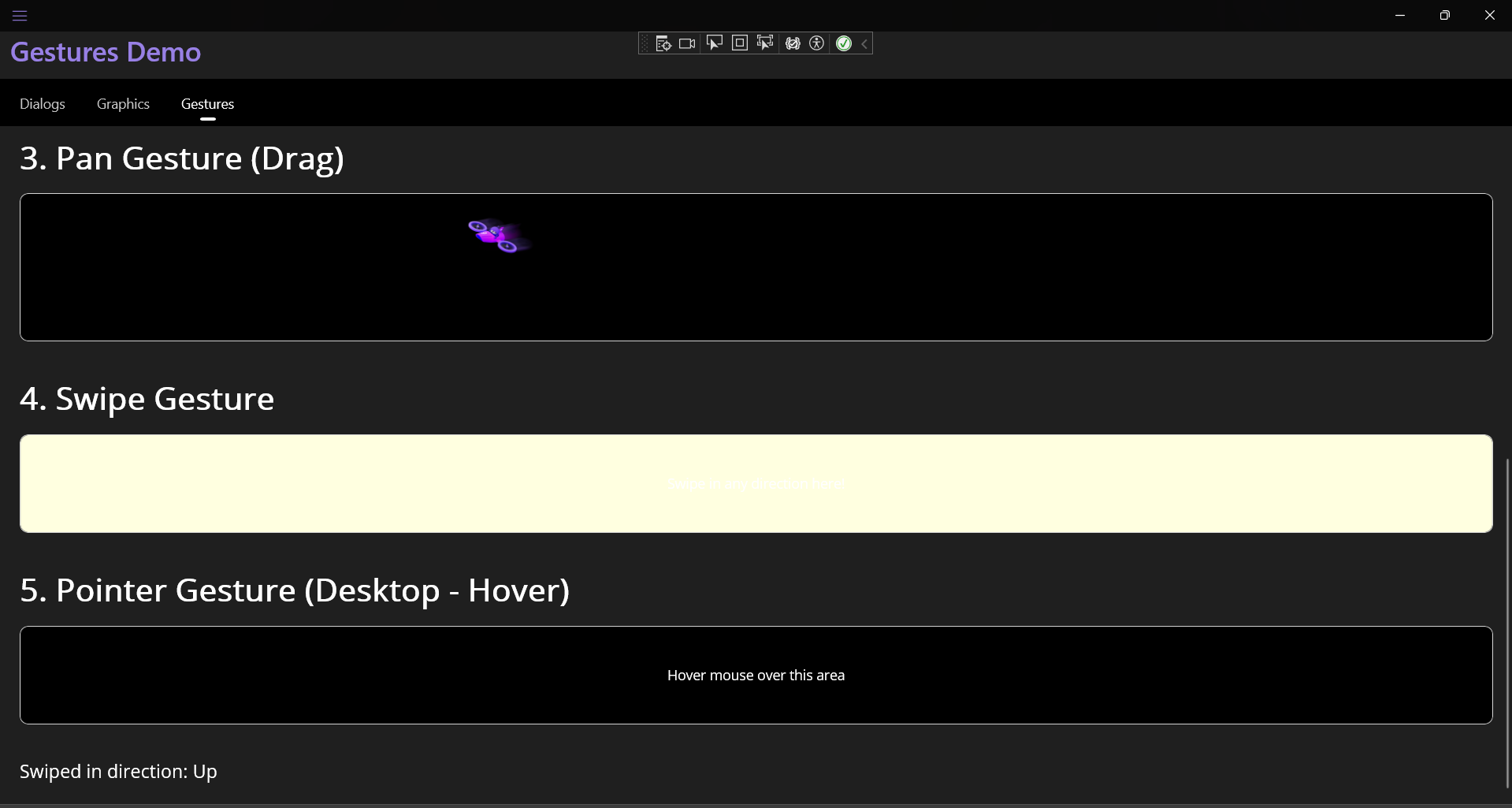
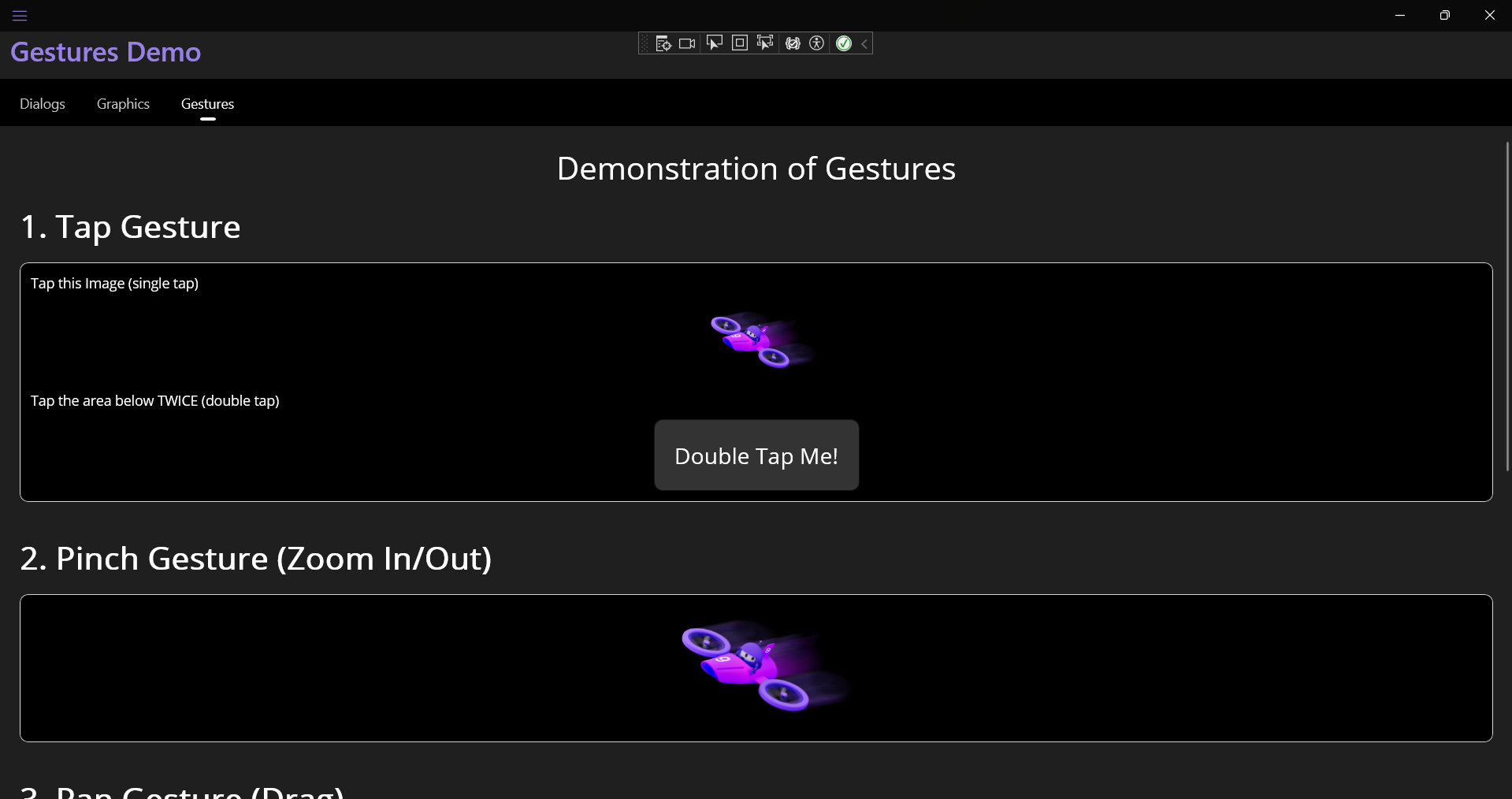
*Пояснение:* Обработчики получают аргументы, содержащие информацию о жесте. В обработчиках, таких как OnPanUpdated, важно отслеживать статус жеста (Started, Running, Completed) для корректной обработки.

**3. Заключение**

Распознаватели жестов в .NET MAUI предоставляют простой, но мощный способ добавления интерактивности в приложение. Они позволяют легко реагировать на стандартные пользовательские действия, делая интерфейс более живым и интуитивно понятным. Правильное использование жестов особенно важно для сенсорных устройств, но некоторые из них, как Tap или Pointer, также актуальны и для настольных приложений с управлением мышью, повышая удобство их использования.

**4. Скриншоты и примеры**

Демонстрация работы жестов была выполнена на странице GesturesDemoPage, запущенной на платформе Windows.

*Рис. 12.1 - 12.3. Демонстрация работы жестов Pan, Swipe и Pointer на странице GesturesDemoPage.*

**Процесс выполнения примеров:**

1. Создана страница GesturesDemoPage.
2. К элементам Image и Frame были добавлены распознаватели жестов.
3. В code-behind реализованы обработчики событий для каждого жеста.
4. В процессе разработки была решена проблема с конфликтом распознавателей одиночного и двойного нажатия путем создания для них отдельных обработчиков.
5. Проведено тестирование жестов, доступных для мыши на Windows (Tap, Pan, Swipe, Pointer), которые работали корректно. Было отмечено, что жест Pinch требует для тестирования сенсорного экрана или эмуляции в эмуляторе.
6. Подготовлены скриншоты работающего приложения.

**5. Документация**

* **Ссылка на GitHub с исходным кодом:** <https://github.com/Nikita-Levuskin/C-sharp-practic/tree/main/1>
* **Описание выполненных шагов:**
  1. Изучены материалы по теме "Жесты" (глава 12).
  2. Создана страница GesturesDemoPage для демонстрации.
  3. Реализованы примеры для TapGestureRecognizer, PinchGestureRecognizer, PanGestureRecognizer, SwipeGestureRecognizer и PointerGestureRecognizer.
  4. Проведено тестирование на платформе Windows и подготовлены скриншоты.