

Unity UI Tutorial System

A **fully customizable, dynamic UI tutorial system** for Unity that helps guide players by highlighting buttons, displaying tooltips, and managing tutorial sequences. This system is designed to be **modular, scalable, and easy to integrate** into any game project.

Preview

<https://github.com/user-attachments/assets/01dd8ff6-89cd-4f7b-9308-72f4a0b2855b>

Features

- **Dynamic Tutorial Sequences** – Easily create and manage multiple tutorial sequences.
 - **Auto-Generated Enums** – Automatically generate `ButtonID` and `SequenceID` enums from `ScriptableObject`s.
 - **Button Highlighting** – Focuses on buttons and makes them stand out during tutorials.
 - **Customizable Tooltips** – Position tooltips near buttons dynamically.
 - **Event-Based System** – Hooks into Unity UI buttons with event-driven interactions.
 - **Editor Integration** – Simple tools to manage tutorials from the Unity Editor.
-

Installation & Setup

1 Clone the Repository

```
git clone https://github.com/BlackSentryDevXo/UI-Onboarding-Tutorial-Mask-System-Unity.git
```

2 Import into Unity

- Open your Unity project.
- Drag and drop the `TutorialSystem` folder into your `Assets` directory.

3 Add the `UITutorialManager` to Your Scene

1. Create an empty `GameObject` and name it `UITutorialManager`.
2. Attach the `UITutorialManager.cs` script.
3. Assign:
 - **Overlay Panel** (to dim the screen during tutorials).
 - **Tooltip Prefab** (for showing instructions).
 -

4 Define Your Buttons

1. Create a `ScriptableObject` for buttons:
 - **Go to Assets > Create > UI > Buttons DB**
 - Name it `ButtonsDatabase`.
 - Add the button names.
 - Click **Generate Enum** (or use **Tools > ButtonID Enum Generator**).
2. **IMPORTANT!!** Add `UIButton.cs` to every button you want highlighted
3. **IMPORTANT!!** For every button you want highlighted, Assign `ButtonID` to each UI button in the Inspector.

5 Create Tutorial Sequences

1. **Go to Assets > Create > UI > SequenceID DB.**
2. Add a new `SequenceDatabase`.
3. List the tutorial sequence names.
4. Click **Generate Enum**.
5. Assign tutorial sequences in `UITutorialManager`.

Your Tutorial manager should look something like this when everything is well setup

6 Install NaughtyAttributes (Optional)

To take advantage of enhanced Inspector UI flexibility, install the **NaughtyAttributes** package: - **Download from GitHub:** [NaughtyAttributes Repository](#) - **Import into Unity** via Package Manager. - This allows for better Inspector usability in the project.

How to Use

Starting a Tutorial

Call the `StartTutorial` function with a `SequenceID`:

```
UITutorialManager tutorialManager = FindObjectOfType<UITutorialManager>();  
tutorialManager.StartTutorial(SequenceID.seq_shop);
```

Highlight a Button with a Tooltip

```
tutorialManager.FocusOnButton(ButtonID.shop, "Click here to open the shop!");
```

Ending a Tutorial

```
tutorialManager.EndTutorial();
```

Customization

Modify Tooltip Appearance

- Edit the **TooltipUI Prefab** to change text styles, colors, or animations.
- Modify **TooltipUI.cs** to change how tooltips position themselves.

Add Custom Logic to Steps

Each step can execute custom logic using the **OnStepCompleted** event:

```
var step = new UITutorialStep {  
    buttonID = ButtonID.shop,  
    message = "Click here to open the shop!",  
    OnStepCompleted = () => Debug.Log("Shop button clicked!")  
};
```

Here are some visuals of how a well setup project should look like in action

Adjust Button Highlight Effects

Modify **UIButton.cs** to change how buttons are visually highlighted during a tutorial.

Contributing

We welcome contributions! Feel free to **fork** the project, submit **pull requests**, or **report issues**.

How to Contribute

1. **Fork the repository.**
2. **Create a new branch** (feature/tutorial-improvement).
3. **Make your changes & commit.**
4. **Push the branch & open a PR.**

Future Improvements

- **UI Animation System** – Add smooth transitions for highlighting buttons.
- **Localization Support** – Multi-language support for tutorials.
- **Analytics Integration** – Track tutorial completion rates.

License

This project is licensed under the **MIT License**. Feel free to use and modify it!

Credits

Developed by **BlackSentryDev**. Inspired by various UI tutorials in the gaming industry.

Support the Project

If you find this useful, **give it a review on the asset store!**