Extrême 3D Pro

Code source
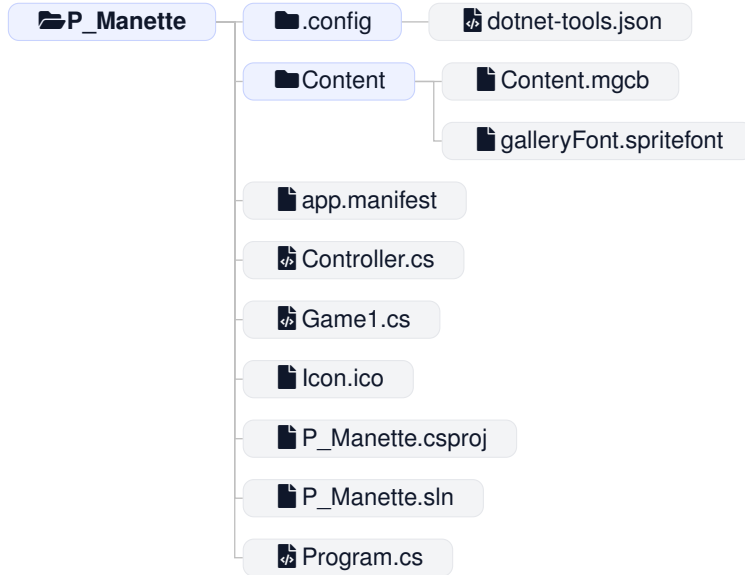
Par Etienne Caulier, Daniel Miramontes

Le 18 novembre 2025

Atelier Programmation - I.DA-P4B
CFPT Informatique

Table des matières

1   Arborescence

- 📂 **P_Manette**
  - 📁 .config
    - `</>` dotnet-tools.json
  - 📁 Content
    - 📄 Content.mgcb
    - 📄 galleryFont.spritefont
  - 📄 app.manifest
  - `</>` Controller.cs
  - `</>` Game1.cs
  - 📄 Icon.ico
  - 📄 P_Manette.csproj
  - 📄 P_Manette.sln
  - `</>` Program.cs

2   Fichiers

```csharp
using System.Diagnostics.Contracts;
using Microsoft.Xna.Framework;
using Microsoft.Xna.Framework.Graphics;
using Microsoft.Xna.Framework.Input;


namespace P_Manette
{
    public class Game1 : Game
    {
        private GraphicsDeviceManager _graphics;
        private SpriteBatch _spriteBatch;
        private Controller contoller;
        private SpriteFont font;

        public Game1()
        {
            _graphics = new GraphicsDeviceManager(this);
            Content.RootDirectory = "Content";
            IsMouseVisible = true;
        }

        protected override void Initialize()
        {
            base.Initialize();

            contoller = new Controller();
        }

        protected override void LoadContent()
        {
            _spriteBatch = new SpriteBatch(GraphicsDevice);
            font = Content.Load<SpriteFont>("galleryFont");
        }

        protected override void Update(GameTime gameTime)
        {
            if (GamePad.GetState(PlayerIndex.One).Buttons.Back == ButtonState.Pressed ||
                Keyboard.GetState().IsKeyDown(Keys.Escape))
            {
                Exit();
            }

            contoller.Update();

            base.Update(gameTime);
        }

        protected override void Draw(GameTime gameTime)
        {
            GraphicsDevice.Clear(Color.CornflowerBlue);

            _spriteBatch.Begin();
            _spriteBatch.DrawString(font, (contoller.connectedController != null) ?
                "Bonjour !" : "Manette non detectee", new Vector2(100, 100),
                Color.White);
            _spriteBatch.End();

            base.Draw(gameTime);
        }
    }
}
```

Listing 1 – Fichier principal Game1 qui lance toute l'application

```
1   using System;
2   using System.Linq;
3   using Microsoft.Xna.Framework;
4   using SharpDX.DirectInput;
5
6   namespace P_Manette
7   {
8       public class Controller
9       {
10          private DirectInput directInput;
11          private Joystick joystick;
12          private JoystickState joyState;
13          public bool buttonPressed;
14          public string connectedController;
15
16          public Controller()
17          {
18              directInput = new DirectInput();
19              DetectedController();
20          }
21
22          public void Update()
23          {
24              var state = joystick.GetCurrentState();
25              int nb = 11;
26              if (PressedButton(nb))
27              {
28                  throw new Exception($"Button {nb} appui ");
29              }
30          }
31
32          public void DetectedController()
33          {
34              var joystickId = Guid.Empty;
35
36              var devices = directInput
37                  .GetDevices(DeviceType.Joystick, DeviceEnumerationFlags.AttachedOnly)
38                  .Concat(directInput.GetDevices(DeviceType.Gamepad,
39                      DeviceEnumerationFlags.AttachedOnly))
39                  .ToList();
40
41              foreach (var device in devices)
42              {
43                  if (device.InstanceName.Contains("3D"))
44                  {
45                      joystickId = device.InstanceGuid;
46                      connectedController = device.InstanceName;
47                      break;
48                  }
49              }
50
51              if (joystickId != Guid.Empty)
52              {
53                  joystick = new Joystick(directInput, joystickId);
54                  joystick.Properties.BufferSize = 1024;
55                  joystick.Acquire();
56              }
57          }
58
59          public bool PressedButton(int numBtn)
60          {
61              var state = joystick.GetCurrentState();
62              this.buttonPressed = state.Buttons.Length > 0 && state.Buttons[numBtn - 1];
63              return buttonPressed;
64          }
65      }
66  }
```

Listing 2 – Fichier qui gère la manette