

The background features abstract, overlapping green shapes in various shades, creating a modern and dynamic feel. The shapes are primarily on the left and right sides, framing the central text.

Portfolio

町田勇樹

プロフィール

名前 町田 勇樹
(マチダ ユウキ)

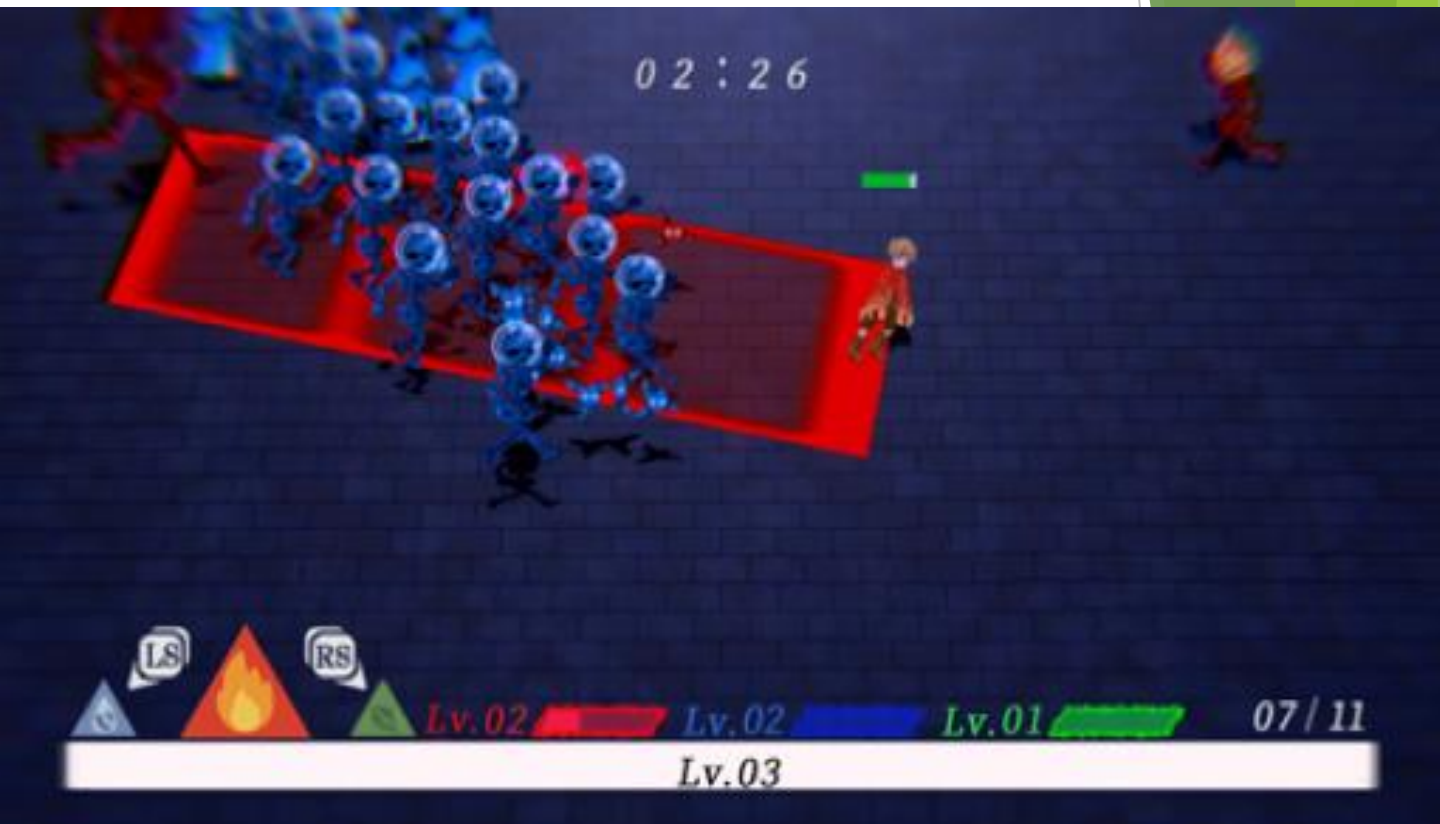
年齢 23歳

所持スキル C/C++、C#、Unity、JavaScript、
Visual Studio、ActionScript、GitHub、PHP、HTML



希望する職種 プログラマー、サーバーサイドエンジニア

作品紹介



ゲームジャンル
2.5Dシューティングアクションゲーム

制作環境 Unity

炎、木、水の3属性を切り替えながら、迫りくる大量の敵を倒し、最後まで生き残るゲームです

役割 UIプログラミング

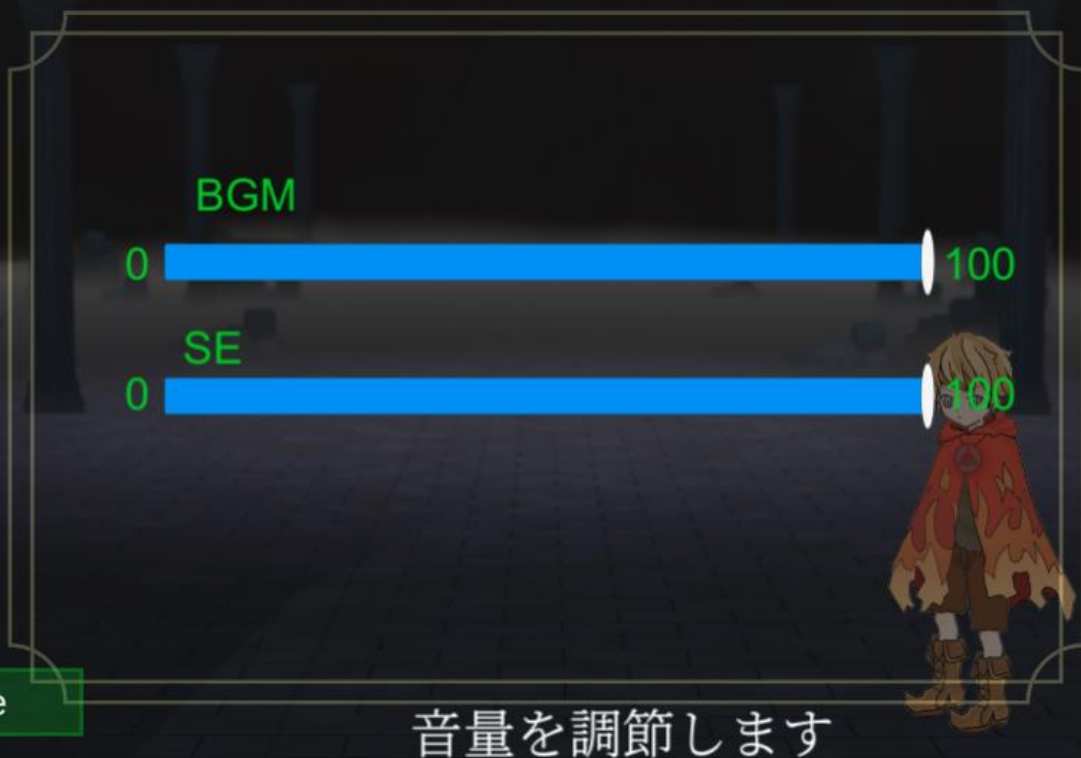
1. 画面変換のボタン及びそのプログラムの作成
2. BGM、SE等のゲームミュージックの導入・反映及びその音量調整のプログラムの作成
3. ゲーム画面に移行時に、音声切り替わるプログラムを作成
4. 武器選択の場面で、その武器の詳細情報を表示するプログラムの作成
5. ボタンの上にカーソルを合わせた際に、そのボタンを押したらどのようなようになるかの説明を、ゲーム画面上に表示するプログラムを作成

画面変換のボタン及びそのプログラム の作成



```
public void GameStart()  
{  
    SceneManager.LoadScene("Weapon Select Scene");  
}  
  
public void OnGameStartMouseOver()  
{  
    int count = (int)(Time.time);  
    _label.text = "ゲームをスタートします";  
}  
..
```

Option画面のプログラム



オプションボタンを押した際に、タイトルシーンにオプション画面に切り替えるためのプログラムです

```
//オプションボタンを押したらオプションの画面に移動する。
public void OptionScene()
{
    //SceneManager.LoadScene("Option");
    title.SetActive(false);
    btn1.SetActive(false);
    // btn2.SetActive(false);
    btn3.SetActive(false);
    btn4.SetActive(false);
    btn5.SetActive(false);
    btn6.SetActive(false);
    obj_back.SetActive(true);
    opt.SetActive(true);
    obj_Title.SetActive(true);
    // btn_test.SetActive(true);
    eventSystem.SetSelectedGameObject(obj_Title.gameObject);
    _label.text = "音量を調節します";
}
```

BGM、SE等の導入・反映及びその音量調整のプログラムの作成



```
// Update is called once per frame
public void SoundVolumeSE(float seSliderValue)
{
    slider_se.value = seSliderValue;

    PlayerPrefs.SetFloat("SEVolume", seSliderValue);
    PlayerPrefs.Save();
}

public void SoundVolumeBGM(float bgmSliderValue)
{
    slider_bgm.value = bgmSliderValue;

    PlayerPrefs.SetFloat("BGMVolume", bgmSliderValue);
    PlayerPrefs.Save();
}

public void SoundSliderOnValueChanged(float newSliderValue)
{
    // 音楽の音量をスライダーの値に変更
    audioSource.volume = newSliderValue;
}
```

ゲーム画面に移行時に、音声 切り替わるプログラムを作成

```
private void Awake()  
{  
    if (instance == null)  
    {  
        instance = this;  
        DontDestroyOnLoad(gameObject);  
    }  
    else  
    {  
        Destroy(gameObject);  
    }  
}
```

```
void OnActiveSceneChanged(Scene prevScene, Scene nextScene)  
{  
    if (beforeScene == "StageSelect" && nextScene.name == "Stage1")  
    {  
        audioSource[0].Stop();  
        audioSource[2].Play();  
    }  
    //ステージ1からタイトル画面に戻る際の音変換  
    if (beforeScene == "Stage1" && nextScene.name == "Title")  
    {  
        audioSource[0].Play();  
        audioSource[2].Stop();  
    }  
}
```

singletoninstanceにして、タイトル画面に戻った際に、音が重ならないようにしました。

武器選択の場面で、武器の情報を表示するプログラムの作成

ゲーム画面



武器の選択時に、その武器の情報を表示するようにプログラムしました

武器選択の場面で、武器の情報を表示するプログラムの作成②

実際のコードです。

//ノーマルワンドのボタンの上にカーソルが置かれると、その情報が出てくる

```
public void OnNormalMouseOver()  
{  
    int count = (int)(Time.time);  
    _label_name.text = "ノーマルワンド";  
    _label_data.text = "狙った場所に攻撃が飛んでいく";  
    //obj1.SetActive(true);  
}
```

//バーストワンドのボタンの上にカーソルが置かれると、その情報が出てくる

```
public void OnBurstMouseOver()  
{  
    int count = (int)(Time.time);  
    _label_name.text = "バーストワンド";  
    _label_data.text = "四方向に攻撃が飛んでいく";  
  
    //obj2.SetActive(true);  
}
```

```
public void OnNormalMouseOver()  
{  
    int count = (int)(Time.time);  
    _labela.text = "20";  
    _labelb.text = "2";  
    _labelc.text = "10";  
    _labeld.text = "なし";  
    _labele.text = "0.3";  
}
```

```
public void OnBurstMouseOver()  
{  
    int count = (int)(Time.time);  
    _labela.text = "10";  
    _labelb.text = "3";  
    _labelc.text = "2";  
    _labeld.text = "なし";  
    _labele.text = "0.2";  
}
```

ボタンの上にカーソルを合わせた際に、そのボタンを押したらどのようなようになるのかの説明を、ゲーム画面上に表示するプログラムを作成



初めてこのゲームをプレイするユーザーにも、わかりやすくなるように心がけました。

具体的には、初見のゲームプレイヤーでも、その画面上のボタンを押したらどのようなことが起こるかの説明を載せました。

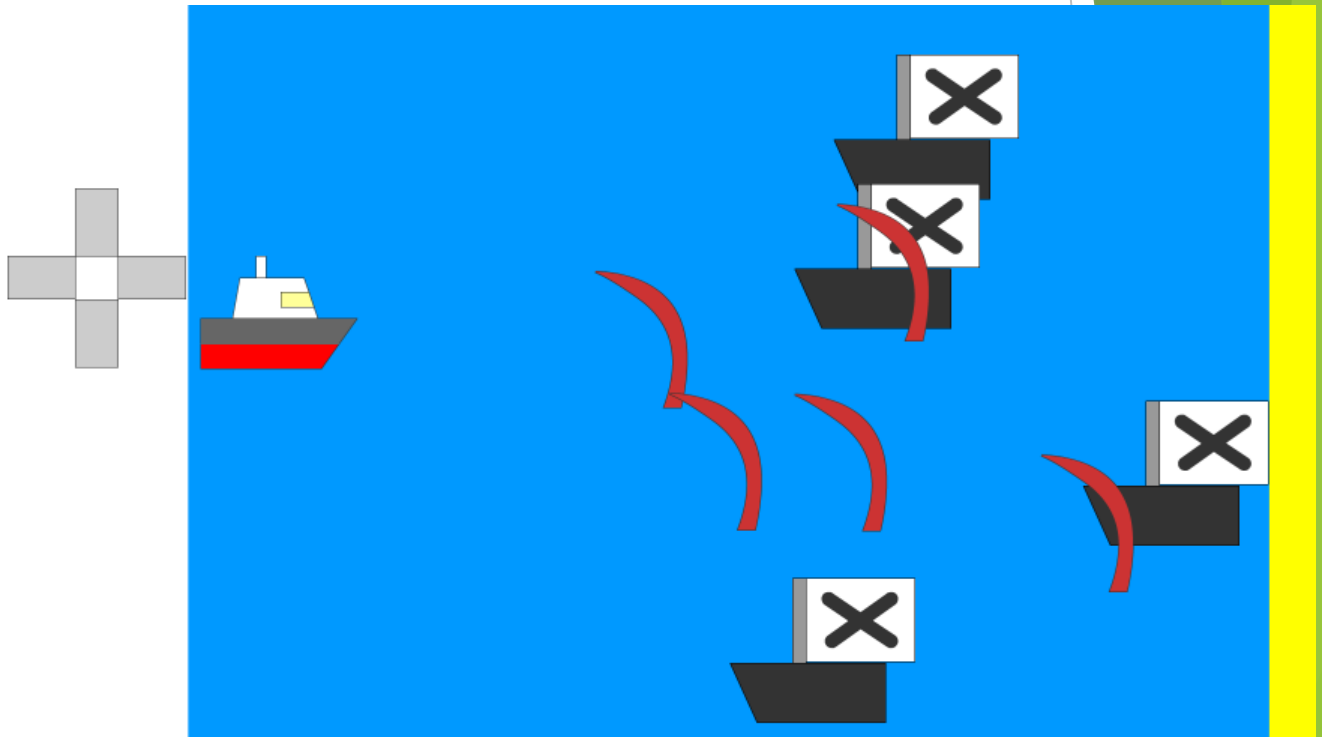
ボタンの上にカーソルを合わせた際に、そのボタンを押したらどのようなようになるのかの説明を、ゲーム画面上に表示するプログラムを作成

実際のコードです。

```
public void OnGameStartMouseOver()  
{  
    int count = (int)(Time.time);  
    _label.text = "ゲームをスタートします";  
}  
public void OnTutorialtMouseOver()  
{  
    int count = (int)(Time.time);  
    _label.text = "ゲームのチュートリアルをします";  
}  
  
//マニュアルボタンを押したらマニュアルの画面に移動する。  
public void ManualScene()  
{  
    SceneManager.LoadScene("Manual");  
}  
  
public void OnManualButtonMouseOver()  
{  
    int count = (int)(Time.time);  
    _label.text = "操作確認の画面に移動します";  
}  
  
public void OnCreditButtonMouseOver()  
{  
    int count = (int)(Time.time);  
    _label.text = "クレジット画面に移動します";  
}
```

Animate 作品

作品名 HUNE



船を操作し、妨害をよけながら、
ゴールを目指す作品

動作環境 Windows