

－ PROFILE －



氏 名

向山 陸登

生年月日

2002年6月25日

趣 味

散歩・アニメ鑑賞

作品データ

<https://bit.ly/4bdly55>

使用言語/年数

C/C++ 4年・C# 4年・HLSL 1年

使用可能ツール

使用歴 1年未満

GitHub・Maya・UnrealEngine

使用歴 2年以上


Photoshop・PremierePro・Unity・VisualStudio

－ CONTENTS －

2P - 回せ！ペン忍道畳

6P - 奇妙なメガネハウスからの脱出

回せ！ぺん忍道畳

- ・制作期間 3ヶ月(02/01～05/31)
- ・制作ツール Unity 2021.3.15f1 VisualStudio 2017 Plastic SCM
- ・制作メンバー プログラマー X 2 レベルデザイナー X 2
グラフィッカー X 4 サウンドエフェクト X 2
- ・受賞履歴  ゲーム大賞 アマチュア部門 一次審査通過
ゲームクリエイター甲子園 デジタルハーツ賞 受賞
- ・関連フォルダ 回せ！ぺん忍道畳.zip



- 概要 -

“畳を回転して道を作り、ゴールを目指す”パズルアクション。
パズルが苦手な人も楽しめるよう、丁寧なレベルデザインや
可愛いキャラクターにこだわりました。



回せ！ペン忍道畳

－ 担当箇所 －

チームリーダー

メンバーの意見の取りまとめや進捗確認

プログラマー

タイトルシーン、チュートリアル、タイマー処理

	タスク名	担当者名	開始日	終了日	進捗確認	01	02	03	04	05	06	07	08	09	10
	草案		02/01	02/10	☑										
	プロトタイプ		02/13	03/17	☑										
	α版		03/20	03/31	☑										
	β版		04/03	04/28	☑										
	マスター版(目安)		05/01	05/19	☑										
	紹介映像(目安)		05/15	05/21	☑										
	作業名		04/01	04/23	☐										

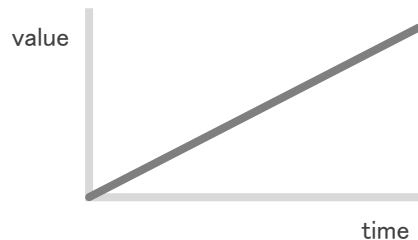
スプレッドシートで進捗管理

・操作感の改善

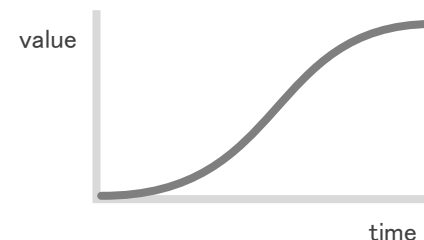
プロトタイプでは`Transform.Rotate`で回転の処理を行っていましたが、`DOTween`の`Ease.InOutQuart`に変更することで気持ちよい操作感を実現できました。



Transform.Rotate



Ease.InOutQuart



回せ！ペン忍道畳

－ ソースコード －

TitleManager(一部抜粋)

他の人が見やすいよう、インデントをそろえています

```
public class TitleManager : MonoBehaviour
{
    private ClearManager clear;
    private Fade fader;
    private GameAward input;
    private savemanager save;
    private Sound sounds;

    [Tooltip("切り替え後のカメラを設定しておく")]
    [SerializeField] private CinemachineVirtualCamera virtualCamera;
    [SerializeField] private SceneObject scene;
    [SerializeField] private GameObject player;
    [SerializeField] private GameObject confeEffect;

    [Header("UI関係")]
    public RectTransform menuBackGround;
    public RectTransform image;
    public GameObject[] butoon;

    private bool isTopicEnabled = false;
    private bool isGoal = false;
    private bool spawnOnTrue = false;
    private int defaultPriority;
    private string filePath;

    void Start()
    {
        // クリアチェックスクリプトを取得
        clear = GetComponent<ClearManager>();

        // フェードスクリプトを取得し、フェードイン
        fader = GetComponent<Fade>();
        fader.FadeIn(Color.black, 1.0f);

        // インプットシステムクラスをインスタンス化
        input = new GameAward();
        input.Enable();

        .
        .
        .
    }
}
```

Timer

命名規則を守り、簡潔なコードを書けました

```
/// <summary>
/// ステージがスタートしてからクリアするまでのタイムを計測するクラス
/// </summary>
public class Timer : MonoBehaviour
{
    private float counter;
    private bool isCounting;

    // 初期化処理
    public void Start()
    {
        counter = 0.0f;
        isCounting = false;
    }

    // 更新処理
    public void Update()
    {
        // 計測
        if (isCounting == true)
        {
            counter += Time.deltaTime;
        }
    }

    // カウンターが有効かどうかを設定する関数
    public void SetCounterEnable(bool flg)
    {
        isCounting = flg;
    }

    // カウンターの値を取得する関数
    public float GetCounter()
    {
        return counter;
    }
}
```

回せ！ペン忍道畳

・グラフィックの向上

単調な印象があったためポストプロセスで調整を行いました。

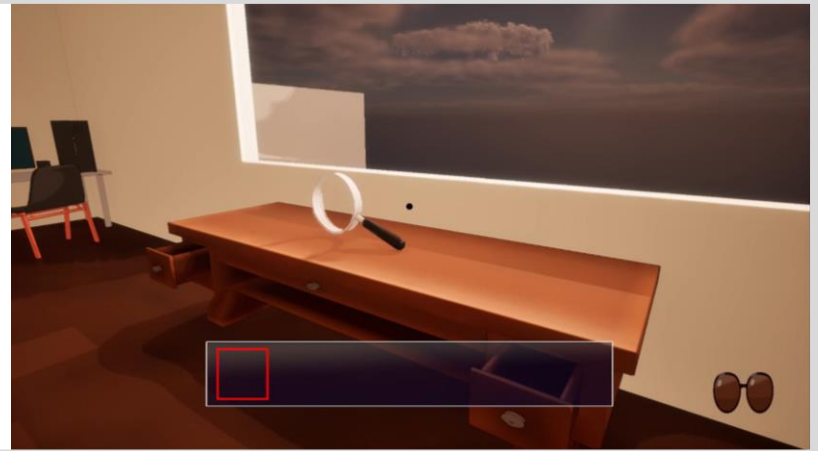
Depth Of Field 建物内に注目してもらえよう建物外の背景を少しぼかしています。

Color Grading 色味に変更を加え、畳の色を目立たせています。



奇妙なメガネハウスからの脱出

- ・制作期間 1ヶ月(10/01~10/31)
- ・制作ツール Unreal Engine 5
- ・制作メンバー プログラマー X 2 プランナー X 1 グラフィック X 2
- ・担当箇所 タイトル、ギミック、インベントリ

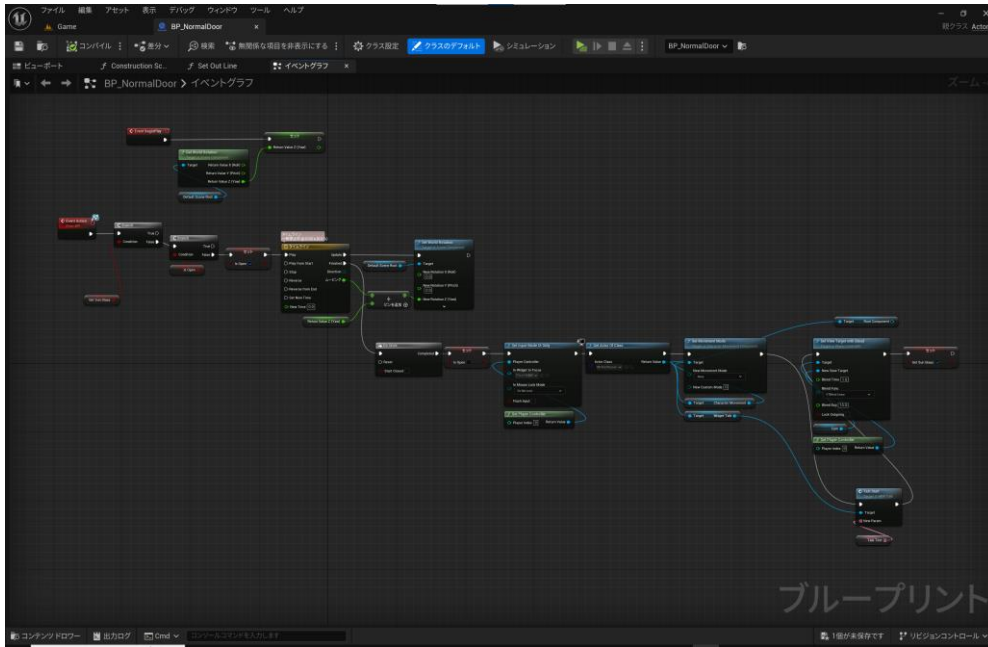


- 概要 -

インターン先で制作。メガネを着脱しながら謎を解き進める、一人称脱出ゲーム。サングラスや3Dメガネ、虫メガネなどメガネによって効果が異なります。



奇妙なメガネハウスからの脱出



- 工夫した点 -

短期間での制作だった為、
ギミックを使い回すことが出来るよう工夫
休日は自宅でも学習を行い、
ウィジェットを中心に実装を行いました。



プランナーからの指示は「ドアを開けた時のテキスト表示」のみでしたが、
カメラを移動する処理を挟み、机の上のアイテムに誘導を行いました。