



福岡ゲームコンテスト大賞作品

5びょうごのきみと。

担当箇所紹介

東京工芸大学芸術学部ゲーム学科

制作:すばむ(8名) 紹介者:渡邊大地

使用ゲームエンジン:(Unity2021.3.0f (URP))

もくじ

- ・ゲーム、開発環境の紹介・・・[P2~4](#)
- ・担当箇所、工夫した点の紹介・・・[P5~8](#)
- ・ソースファイル担当部分の紹介・・・[P9~12](#)
- ・最後に（GitHubアクセスURL等）・・・[P13~](#)

5びょうごのきみと。とは

幽霊になってしまった弟をよみがえらせるために
兄が遺跡を目指す、アクションパズルゲーム

ステージには様々なギミックがあり遺跡への道を妨げる



段差



隙間



押せる箱

風

進むには、**弟との協力**が必要不可欠

このゲームの特徴

- ・プレイヤーの動きを5秒後 **そっくりそのまま** 弟も行う。



プレイヤーが坂を上ると
5秒後に**弟**は
同じように坂を上る

(当たり判定がある)

- ・弟は幽霊だが **触れることができる**。



一緒にボタンを押したり、
弟の上に乗ることもできる

開発環境

▶メンバー

プランナー:2人

デザイナー:3人

プログラマー:3人(自分含む)

計8名

▶制作期間

2022/04/19 ~ 2023/01/05

担当箇所 1/4

・アウトゲーム全般の処理

ソースファイルによる処理だけではなく、仕様をもとにUnity内での各画面作りやデザイナーが作って下さったUIや画像などの差し替えも行いました。

早めの段階で基盤となる処理を作ることで、制作序盤にゲームを一通りプレイできる環境を作りました。また、できるだけ各画面を1つのシーンにまとめることで、世界観が決定した際の実装できる演出処理の幅を広げました。

・アウトゲーム中のUI演出処理

演出処理関係は役職・作業内容問わず世界観に合ったより良い演出は何かを追求し、まとまったアイデアをもとに作成しました。

処理の内容としては、汎用性が高くなるよう、親クラスとしてUIのエフェクト処理をまとめることで、アウトゲーム中のUI処理だけでなくインゲーム中のポーズ画面などの処理にも利用できるようにしました。

担当箇所 2/4

・一部インゲーム中のUI処理

クリア画面遷移の基盤づくり(今はほかのプログラマーが演出処理を記述した関係上最初にした時と内容がだいぶ異なります)や、ポーズ画面への遷移・演出処理を行いました。UIアニメーションをできるだけプログラマーが担当することで、デザイナーへの負担を軽減するように励みました。

また、上記の「アウトゲーム演出中のUI演出」でも記述した通り、インゲームでの演出を流用してアウトゲームとの統一性を加味して作成しました。「“もくじ→あそびかた”と“ポーズ→あそびかた”」の演出部分がわかりやすい例です。

・一部インゲーム中のカメラ演出とそれに伴うプレイヤーの挙動処理

スタート時の「鳥を追いながらステージ見渡し→プレイヤー入場」とゴール時の「ゲートに向かう→本全体を映す」処理を実装しました。ステージ見渡し処理に関しましては、カメラの台数が増減しても大丈夫なよう配列化しインスペクターウィンドウ上で変更できるよう処理を記述しました。

また、プレイヤーの移動時間、カメラの切り替えるタイミングなど些細な部分も変更しやすくすることで、演出に関する微調整をできるように心掛けました。

担当箇所 3/4

・ゲームのクリアデータを ローカルストレージ管理する処理

PlayerPrefsを利用するのではなくローカルストレージ保存にすることで、各実行ファイルが入っているフォルダに直接保存内容を格納し個々のフォルダでセーブデータを管理できるようにしました。

これにより1つのPCで複数のセーブデータを持てるようになる他、セーブデータへのアクセスが複数の実行ファイルで混ざってしまわないようにしました。

・ゲームの状態管理処理

ゲームの状態を列挙体で格納しておくことで、ゲームの状態を見て処理を実行するか否かを判別できるようにしました。また、ゲームの状態取得をプロパティで、状態変化を関数で統一することで、安易に状態を変更できないようにする他、万が一予想しない状態変化が発生しても関数から参照してすぐに発見できるようにするなどといったことも考慮して作りました。

担当箇所 4/4

・トランジション演出用シェーダー作成

ゴール時にワープゲートに入るため、その時にゲートに入ったと思えるような演出が欲しいということで制作しました。今回この処理を利用しているのはゴール時のみとなっておりますが、他の場面でも利用できるようテンプレートを用意し自分以外のプログラマーも利用できるようにつくりました。

・BGM,SE付けの基盤作成並びに 担当箇所の音付け

シングルトンパターンを用いて基盤を作成することで、空間内で出したい音（風の音、川の流れる音など）以外の音関係の処理に統一感を持たせました。また、共有することで他のプログラマーも簡単に音関係の処理を導入できるようにしました。

ソースファイル担当部分詳細 1/4

ソースファイル	軽い説明	記述・担当箇所
▼▼ Scriptsフォルダ		
▼ Controllerフォルダ		
DirectingScript.cs	インゲーム中のスタート演出並びにゴール演出処理を行う。	全記述
GameData.cs	ゲームデータの管理並びにセーブデータのセーブ・ロードを行う。	全記述
Game_State.cs	ゲームの状態管理並びに状態の取得・変更処理を行う。	全記述
GetPlayerTriggerScript.cs	インゲーム中の演出に必要なTrigger情報を取得しDirectingScriptに情報を返す処理を行う。	全記述
StageController.cs	インゲーム開始時に必要な初期化処理全般を行う	BGMをロード・再生する処理のみ記述
StartStoryScript.cs	ゲーム開始時のストーリー演出に関する処理の仮組（ファイナル版では未使用）	全記述
TitleScript.cs	タイトルシーン時のボタン処理並びに演出、パネル切り替え処理全般を行う。	全記述

ソースファイル担当部分詳細 2/4

ソースファイル	軽い説明	記述・担当箇所
▼▼ Scriptsフォルダ		
▼ Soundフォルダ		
Sound.cs	シングルトンパターンを用いて2Dサウンド（立体的な音響ではない音）を一括管理している	「IsPlayingSE メソッド」、「_IsPlayingSEメソッド」以外全記述
▼ Storageフォルダ		
StorageManager.cs	ローカルストレージへのデータ保存処理、読込処理などを行っている。	全記述
UserSettings.cs	更新データの管理、保存先の指定、インターフェースによる保存方法データの管理	全記述

ソースファイル担当部分詳細 3/4

ソースファイル	軽い説明	記述・担当箇所
▼▼ Scriptsフォルダ		
▼ UIフォルダ		
PauseUIAnimation.cs	インゲーム中のポーズ画面のアニメーション処理を行う	全記述
StageUIScript.cs	インゲーム中のUI全般の処理を行う	ポーズ画面に関する処理全般、クリア演出以外の演出関係の処理、BGM・SE関係の処理全般を記述
TransitionScript.cs	トランジション処理を行う処理のテンプレート。（ゲーム内では使用されていない。	全記述
UI_Effect.cs	タイトル画面並びにインゲーム中のUI演出に関する処理を行っている。	全記述

ソースファイル担当部分詳細 4/4

ソースファイル	軽い説明	記述・担当箇所
▼▼ Resources フォルダ		
▼ Transition フォルダ		
TransitionColorShader.shader	トランジションイメージ画像にこのシェーダーを付けることで、指定した色のトランジション処理を行う。（ゲーム内では使用されていない。）	全記述
TransitionTextureShader.shader	トランジションイメージ画像にこのシェーダーを付け、指定のテクスチャーを設定することで、そのテクスチャー画像のトランジション処理を行う。	全記述

最後に

・GitHubのURL

<https://github.com/daichi0907/Five-seconds-with-you>

※URLにアクセス頂くことで、これまで紹介した内容の他に
プロジェクトファイル・実行ファイルのダウンロード、
プレイ動画やPR動画のダウンロード視聴ができます。
また、ソースファイルも直接ホームページから閲覧可能
ですのでご利用いただけると幸いです。

福岡ゲームコンテスト2023で 『大賞』を受賞しました。

・福岡ゲームコンテストのURL

<https://fukuoka-gffaward2023.com/>

この作品で得られた成果はとてもうれしく感じましたが、学内展示のお客様やコンテストの審査員の方々に「面白かった」「夢中になって遊んでしまった」という感想を頂けたことが一番のやりがいを感じました。ゲームを作り、お客様に提供して改善点を見つける良い経験でした。