

企画番号 2024-04

音楽ゲームを作る

(代表者) 三宅優斗 Y220093

(メンバー) 伊達明芳 Y220036

阪颯太 Y220183

吉澤悠人 Y220072

山崎悠希 Y220035

西村信哉 Y220023

アドバイザー教員 佐野彰

Unity 開発担当・・・伊達明芳、阪颯太

作曲担当・・・三宅優斗、西村信哉

デザイン担当・・・吉澤悠人

譜面作成担当・・・山崎悠希、吉澤悠人

【目的】

音楽ゲームを開発し、プレイヤーが音楽に合わせて楽しめる体験を提供することを目的とし、プロジェクトを通じてゲーム開発に必要なプログラミング技術やデザインスキルを学び、実践する場とする。

【計画】

1.6月上旬にゲームテーマ、タイトルや基本コンセプトを決定する。

そのゲームに必要な機能をリストアップし、ゲームの方向性を明確にする。

2.6月下旬から7月上旬までに、Unity 開発担当は YouTube にて載っている音楽ゲームの作り方の動画を参考に、音楽ゲームのシステムの基盤を作成する。(ノーツ生成、判定、スコアの計算など) また、デザイン担当はゲームデザインの作成(主に背景画面やリザルト画面など)や作曲担当は音楽理論や作曲の仕方などの勉強。こういった感じで、各々の担当で進めるべきところを進める。

3.7月中旬から8月までは、Unity 開発担当は YouTube の解説動画を元に作られたシステムから本ゲームの方向性に合わせるように改良する。デザイン担当は最終的なゲーム素材(ロゴ、背景画像、ボタンデザインなど)を完成させる。作曲担当は、実際に作曲を進めて完成させる。ここから、ゲームの譜面作成担当が活動開始。

4. 完成したゲームに楽曲と譜面を組み込み、iPadでのテストプレイを開始。バグの発見や操作性のフィードバックを受け、改善を繰り返す。9月末までに最終版を完成させ、5名ほ

どの第三者にテストプレイを依頼。結果を基に最終調整を行い、完成度を高める。

【活動経過】

<Unity 開発>

Unityでのゲーム開発経験がなかったため、音楽ゲームのシステムの基盤を YouTube に載っている音楽ゲームの作り方の動画を参考に作成した。ここで Unity におけるゲーム制作の方法も同時に学んでいった。Unity 開発の担当間でのプロジェクト共有は Unity Version Control を使用した。基盤が完成した後は自分たちが必要だと考えた機能の追加や変更を行った。以下に追加した機能について説明する。

・タップ操作

基盤の状態では PC のキーボードを使用して操作するようになっていたため、スマートフォンでタップ操作ができるように変更を行なった。YouTube に載っているタップ操作の作り方の動画を参考にし、このプロジェクトに合うようにコードを書いていた。

・ノーツの種類追加

曲の譜面情報は Json 形式で保存されている。以下に Json ファイルの例を掲載する。

```
{
  "notes": [
    [],
    [
      {
        "lpb": 4,
        "num": 44,
        "block": 0,
        "type": 1,
        "notes": []
      }
    ],
    [
      {
        "lpb": 4,
        "num": 48,
        "block": 1,
        "type": 2,
        "notes": [
          {
            "lpb": 4,
            "num": 52,
            "block": 1,
            "type": 2,
            "notes": []
          }
        ]
      }
    ],
    [],
    []
  ]
}
```

次に Json ファイルに書かれている情報について説明する。一番最初の"notes"は全体のノーツデータを管理するトップレベルの配列である。“notes”配列の中にさらに青い角かっこで書かれている 5 つの配列は難易度を示している。中括弧の中の”lpb”は 1 拍あたりの分割数を表している。”num”はノーツが配置されるタイミングを数値で示している。”block”はノーツが配置されるレーン番号を示している。”type”はノーツの種類を示している。このプロジェクトでは 1 はタップノーツ、2 はホールドノーツ、3 はフリックノーツ、4 はドラッグノーツとして処理している。”notes”はホールドノーツの終点部分のデータが格納される。上記の例では 2 つ目の難易度にタップノーツが一つ流れ、3 つ目の難易度にホールドノーツが一つ流れることを示している。

ノーツの種類を追加するのは Json ファイルの”type”から値を取得することで実現している。しかし、ホールドノーツは始点の情報と終点の情報を両方取得しなければならないため実装する難易度が高かった。

・難易度変更

Json ファイルの”notes”配列の中に 5 つの配列があるため最大 5 つの難易度を作成することができるが、今回は 3 つの難易度から選択できるようにした。難易度変更はこの二次元配列の何番目の配列の情報を取得するかを指定することで実現した。二次元配列の Json ファイルを読み込みには Newtonsoft.Json を用いたが、基盤の状態では一次元配列を読み込むという状態であったため二次元配列を読み込めるようにするのに苦労した。

・ポーズ画面

プレイ画面の右上にポーズボタンを配置し、そのボタンを押すことでポーズ画面が出現し、ゲームが一時停止するようにした。ポーズ画面出現時のアニメーションも作成した。ポーズ画面ではポーズ画面を閉じ、続きから再開できるボタンと曲の最初からリトライできるボタン、選曲画面に戻るボタンを追加した。

・設定画面

選曲画面の左上に設定ボタンを配置し、そのボタンを押すことで設定画面が出現し、各種設定が行えるようにした。ポーズ画面出現時と同様に設定画面出現時もアニメーションを作成した。設定項目はプレイ画面の背景変更、ノーツの速さ変更、タイミング調整、音量調整がある。音量調整はスライダーを用い、それ以外はボタンで値が増減するようになっている。

・スクロールして選曲

基盤の状態では PC のキーボードの矢印キーを使用して曲を選択できるようになってい

た。これをスマートフォンでも操作できるように ScrollView を用いてスクロールして選曲できるようにした。

・ハイスコアや設定の保存機能

ハイスコアや設定の保存には PlayerPrefs を用いた。ハイスコアは楽曲と難易度ごとに保存する必要がある。そのため、文字列補間を使用することで、楽曲の ID と難易度の値を動的に組み合わせてキー名を生成できるようにした。

<作曲>

今まで人生の中で作曲をしたことがなかったため YouTube や本を用いてコード理論（コード理論については成果・結果の<作曲>のところを参照）を学習した。その後、Cubase という音楽開発ソフトの環境構築を行った。正直、プログラミングの環境構築を行ったことがある人にはわかるかもしれないがこの環境構築がかなり苦勞した。この環境構築のところは YouTube を見ながら行った。環境構築を終えた後、「NEO」という曲を取り組み始めた。今回の「NEO」は約2分半の曲だが1曲の中にリズムカルな部分であったり、比較的静かな部分、そしてアップテンポがあったりと工夫しながら作った。

<譜面作成・デザイン>

フリー曲や作曲担当の人が作った曲にそれぞれ3つの難易度の譜面を作った。基本的には曲のリズムに沿って等間隔にノーツ（ノーツについては成果・結果の<Unity 開発>のところを参照）を配置して、難易度があがるにつれてノーツの間隔を短くしていく。一番上の難易度ではできるだけ曲のリズムをほとんど忠実に譜面として再現しつつも、テストプレイを踏まえて叩きやすい譜面になるように作った。

デザインにおいては基本的に既存のサービス（モバイル版の音楽ゲーム）の画面遷移や、画面構成を自分で紙に書き起こして把握した。そして、今回作成するゲームに想定される機能や基幹機能をどう当てはめるかを考えた。

【成果・結果】

<Unity 開発>

ゲームのタイトルからゲーム画面まで一連の流れ、スマホのタッチ操作、4種類のノーツの実装を行った。（タップ、ホールド、ドラッグ、フリック）

ここで各4つのノーツについて説明していく。

- ・タップノーツ…単純に画面をタップするだけでクリアできる本格的なノーツ
- ・ホールドノーツ…指を一定の時間画面に押し続ける必要があるノーツ
- ・フリックノーツ…指で画面をタップするだけでなく、指定された方向にスワイプ（フリ

ック) するノーツ

- ・ドラッグノーツ…ノーツの軌道に沿って指を動かす必要があるもの

Json ファイル (データを整理して保存したり、やりとりしたりするためのシンプルな形式のファイル。リストやグループも作れ、簡単で分かるやすく、ウェブやアプリでよく使われる) を読み込んで、音楽に合わせてノーツが流れるようにできた。設定画面にて、難易度設定やノーツの速度、音楽のタイミングを変更する設定を行うことができた。

<作曲>

「Cubase」という音楽開発ソフトを用いて一から開発した。音楽ゲームにおけるコード進行 (音楽で使われる和音 (コード) を並べる順番のことである。曲の雰囲気や感情を作る大切な要素で、例えば「明るい」、「切ない」などの印象を決めることができるもの) とはどのようなものなのかを YouTube や本で学習しながら行った。今回、「NEO」という曲を作り音楽ゲームではありがちなテンポを少し早くして音楽ゲームらしさのある曲を作ることができた。ちなみに音楽ゲームらしさのある曲とは曲の速さが操作の楽しさに直結することであり、速いテンポでは連打やリズム感が求められ、遅いテンポでは正確なタイミングが重要となる。今回の「NEO」という曲は速いテンポと遅いテンポを両方取り入れ、遊ぶ人達に飽きさせないように工夫した。

<譜面制作・デザイン>

実際作ってポスターセッションで遊んでくれた人の中で音楽ゲームをかなり遊んでいる人の意見としてチュウニズムというゲームの譜面にとっても忠実で原作リスペクトをしているとお褒めの言葉もいただいた。

タイトル画面

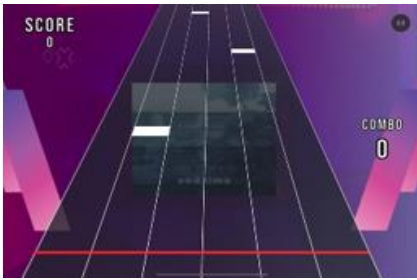


プレイ画面

選曲画面



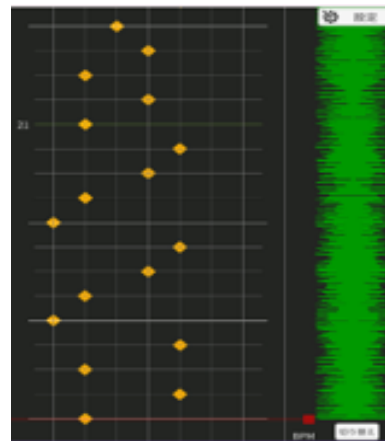
リザルト画面



Cubase の制作風景



譜面の制作風景



参考文献

すけとうだら”【Unity】Unity で音ゲーの作り方を徹底解説！Part1【ゆっくり解説】”YouTube

<https://www.youtube.com/watch?v=WWeyn4TI0II&list=PLB5zYfuqaXlg-mLIoSf7Sey2N6AFSxP8u>

M4RBL. “【Unity】スマホ音ゲーのレーンの作り方【ゆっくり解説】”YouTube

<https://www.youtube.com/watch?v=vTZ2tjKdeeM>

@kazuma_f. 「【Unity】Newtonsoft.Json の基本的な使い方」.Qiita.

< https://qiita.com/kazuma_f/items/55a0b7ff628ab596e6ee >

堀あかり. 「【Unity】初心者でもわかる！AudioMixerを使ったスライダーで音量を調節する

機能の作り方」.note

< https://note.com/calm_otter351/n/need65fbb2810 >

watanabe. 「Unity で UI を一新:スクロールビュー実装チュートリアル[ScrollView]」.あのゲームの作り方 Web 版.

< <https://anogame.net/unity-ui-scrollview-tutorial/> >

@kazuki_kuriyama. 「Unity でセーブデータを作る方法まとめ」.Qiita.

< https://qiita.com/kazuki_kuriyama/items/20847a040d5f52835ba6 >

こじゃら. 「【Unity】アニメーションするダイアログの作り方」.ねこじゃらシティ

< <https://nekojara.city/unity-animated-dialog> >

合同会社バクランテ (2024 年) 『すぐできる！よくわかる！ビジネスに活かせる Canva 入門』 イーストプレス.

秋山公良 (2009 年) 『よくわかる作曲の教科書』 ヤマハミュージックエンタテインメントホールディングス