

(株)C.I. チーム仕様書

内村方哉

古田瑛啓

前田真志

2020 年 8 月 17 日

1 画面仕様

ゲーム選択画面は、主にユーザーがプレーするゲームを選択する画面である。

1.1 スタート・ゲーム選択画面

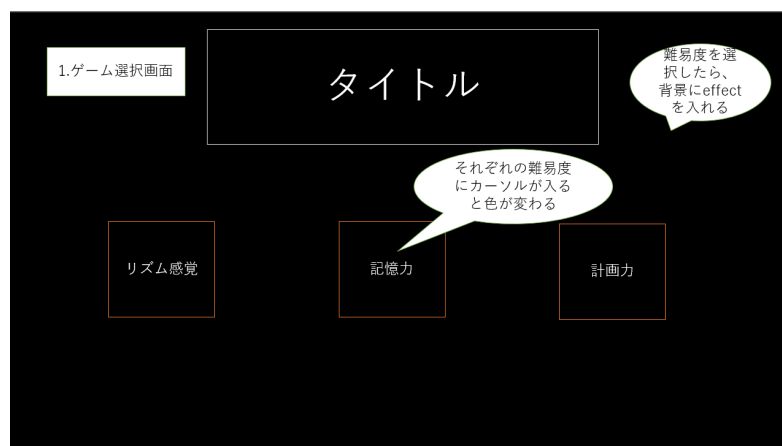


図 1 スタート・ゲーム選択画面

1.2 難易度選択画面

難易度選択画面では、1 つ前のゲーム選択画面で選択したゲームの難易度を選択する画面である。

1.3 リザルト画面

ゲームを終了すると、下のような 3 画面に遷移する。リスタートボタン、メイン画面に戻るボタンがある。

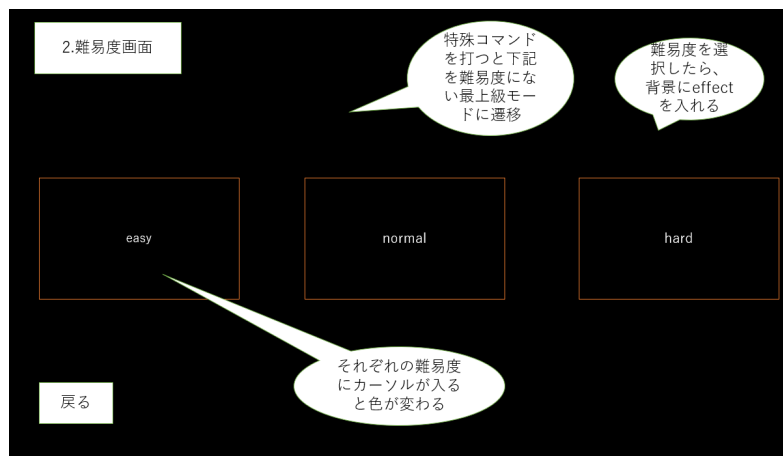


図 2 難易度選択画面

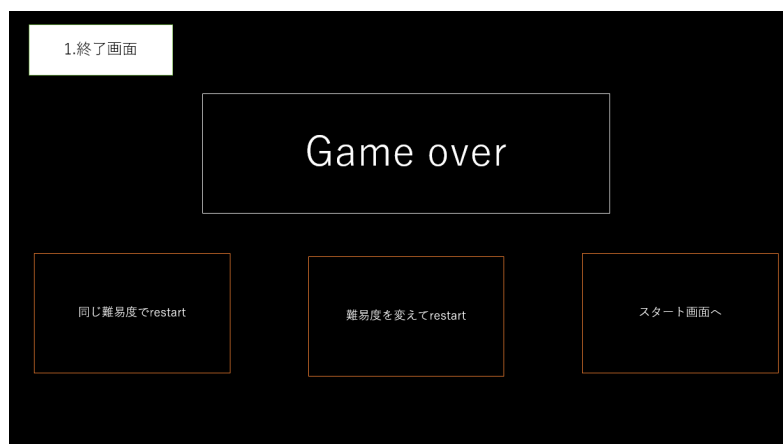


図 3 スタート・ゲーム選択画面

1.4 ゲーム画面：ドンカマチック

複数の円があり，それに対応するキーをタイミングよく押して点数を稼ぐゲームである．それぞれの円の内部に押すべきキーの文字が描写されている．ゲームオーバーないしクリアするとリザルト画面に遷移する．

1.5 ゲーム画面：てくてく

自動で動く自機を操作してゴールに向かうゲーム．十字キーを用いる．クリアないしゲームオーバーになるとリザルト画面に遷移する．

1.6 ゲーム画面：家中何羽

このゲームを選択すると，下のような画面に遷移する．画面中央にあるオブジェクトに何羽鳥がおさまっているのかを数えるゲームとなる．選択肢が画面に現れるので，正しい答えを選択する．

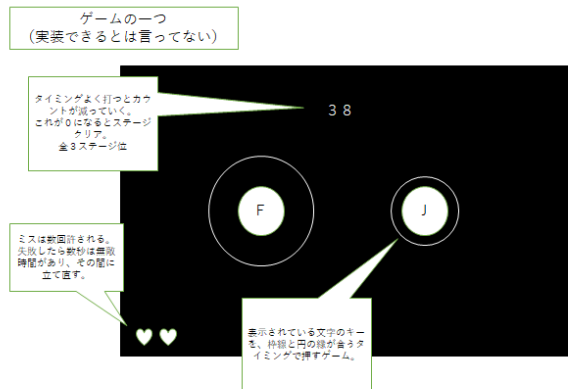


図4 ゲーム画面：ドンカマチック

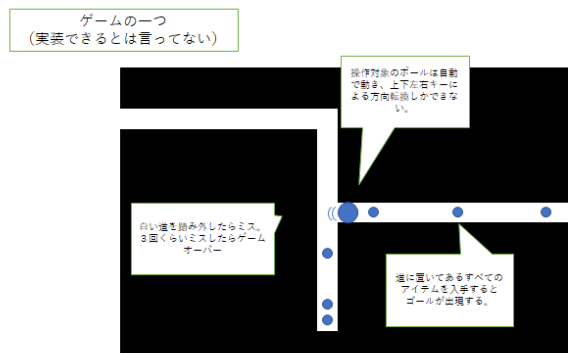


図5 ゲーム画面：てくてく

2 機能仕様

2.1 スタート・ゲーム選択画面

スタート・ゲーム選択画面では、コードを実行したときに最初に現れる画面である。3つのゲームのうち1つを選択するゲーム選択画面が表示され、3つのボタンの中から1つを選択する。選択されたゲームの難易度選択画面に移行する。

2.2 難易度選択画面

難易度選択画面は、スタート・ゲーム選択画面で一つのゲーム選択ボタンをクリックしたとき、あるいは後述のリザルト画面で難易度選択画面に戻った際に表れる画面である。

複数の難易度がボタンとして示されており、これをクリックすることでそれぞれのゲーム画面に移行する。その際、示された難易度に設定される。また、スタート・ゲーム選択画面に戻るボタンも表示されており、間

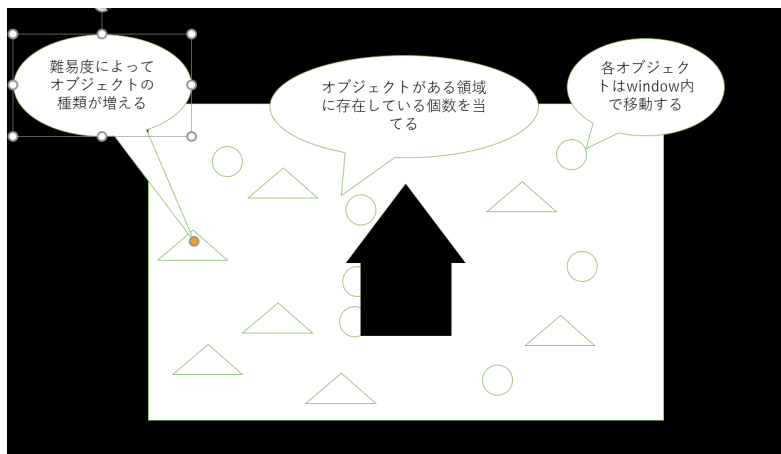


図6 家中何羽

違ったゲームを選択してしまった際はこちらをクリックすることでスタート・ゲーム選択画面に移行できる。

2.3 リザルト画面

リザルト画面は、ゲーム画面でミニゲームを行い、クリアないしはゲームオーバーした際に移行する画面である。

画面中央には今回のゲームで獲得したスコアが表示される。画面下部ではそれぞれ「もう一度プレイ」「難易度を変えてプレイ」「スタートに戻る」の三つのボタンが示されている。

「もう一度プレイ」は、難易度を変えずに同じゲームをプレイするためのボタンである。クリックすると前回と同じゲーム画面に移行する。

「難易度を変えてプレイ」は、同じゲームを難易度を変えてプレイするためのボタンである。クリックすると前回と同じゲームの難易度選択画面に移行する。

「スタートに戻る」は、ゲームをやめてスタート・ゲーム選択画面に移動するためのボタンである。クリックするとスタート・ゲーム選択画面に移行する。

2.4 ゲーム画面：ドンカマチック

黒の画面に、白い丸が複数個浮かんでいる。自分の残機数、スコアなどの文字が表示されて始まる。操作キーはキーボードで、白丸の数によって押すキーは変わる。なので、白丸の中に押すべきキーを明記する。このゲームではマウスは性質上使えない。

ゲームが始まると、それぞれの丸の外から円が縮まってくる。この円が丸の外枠と重なる瞬間に、指定されたキーを押すことでスコアが獲得できる。それぞれの丸で円の縮まる速さは異なり、それぞれで一定のテンポを刻む。一定回数成功することでステージクリアとなる。

円を見逃す、極端にタイミングを外すなど、失敗するとライフが減る。このライフが0になるとゲームオーバーとなり、リザルト画面に遷移する。

2.5 ゲーム画面：てくてく

黒のバックに色のついた自機，幅 1 マスで広がる白の道が見える画面で始まる．操作はマウスか十字キーで行う．十字キーの場合は，上下左右キーの押下を感知し自動で動く自機の向きを変える．マウスを使う場合は，画面を内部的に四分割し，上下左右部をクリックすることで自機の向きをその方向に変更する．

先述の通り，自機は自動で向いている方向に直進する．向きは矢印で示される．向きを変更すると，その矢印の向きも変わり，自機の向きが変わったことがわかる．

道の途中にアイテムが置いてある．これは，小さな図形で存在が確認できる．このアイテムは自機が重なると取得し，そのアイテムは画面上から消える．このアイテムをステージのすべてから回収することで，ゴールのマスが出現する．

ゴールのマスも図形で表示される．先述の通りすべてのアイテムを入手することでゴールが出現する．出現させたのち，ゴールと重なるとステージクリアとなる．

以上の流れをもって，リザルト画面に遷移する．

2.6 ゲーム画面：家中何羽

画面中央に家のようなオブジェクトがあり，画面端から鳥の画像が移動してくる．その画像が家に入っていく．家に 1 羽, 2 羽といったように増えていく．それらを目でみて何羽いるか答えるゲームである．ちなみに家から出ていく鳥も考慮に入れる必要がある．

3 クラス構成

3.1 スタート・ゲーム選択画面

```
class Main//メイン画面を管理
{
    int textX, textY;
    PImage img//画像を格納する変数
    Main{
        // コンストラクタでテキストの座標を決める
    }
    void display()//画面を表示
    int gameSelectButton(int n)//ゲーム選択するボタンを表示
    void loadImage()//画像を読み込む
}
```

3.2 難易度選択画面

```
class Select // 難易度選択画面を管理する
{
    void display()// 各オブジェクト（ボタン，背景文字）の配置場所を設定するメソッド
    int difficultButton(int n) //
    難易度選択ボタン機能の作成
}
```

```
void fileimage()//背景で使う画像を import するメソッド
```

3.3 リザルト画面

```
class result//リザルト画面を管理
    PImage img//画像を格納する変数
    int button(int n)//ボタンオブジェクトを配置
    void display(int scores)//画面を描写
    void fileimage()//画像を読み込む
    void init(int scores, int gameSelect, int difficultSelect, int button) // button
メソッドで処理した返り値によって、ゲーム内容や難易度を初期化する。またゲームスコアは強制的に初期
化される。
```

3.4 ドンカマチック

```
class doncamtic//ドンカマチックの管理
    int life;//ライフ
    int score;//スコア
    int tapN;//音符数
    int[] tapBPM;//それぞれの音符の BPM
    String[] tapKey;//それぞれの投下キー

    PImage img//画像を格納する変数
    void init()//クラス内変数の初期化
    void tap(int key)//キーボードによる正誤判定
    void damage()//
    void nextStage()//
    void calcScore(int scores)//スコアを計算
    void goResult()//ゲームをリスタート
    void fileimage()//画像を読み込み
```

3.5 てくてく

```
class teqTeq
    int[][] stage;//ステージのデータ
    int myX,myY;//自機座標
    int myLife;//自機ライフ
    int myDirect;//自機向き
    int itemCnt;//アイテムの数
```

```
int score;//スコア. たぶん時間になると思う
```

```
PImage img//画像を格納する変数
void init();//クラス内変数の初期化
void loadStage();//ステージ読み込み
void turn();//自機の向き変更
void move();//移動とその後処理
void getItem();//アイテム取得
void fileimage();//画像読み込み
void calcScore(int scores)//スコア計算
void goResult();//リザルト画面に遷移
```

3.6 家中何羽

```
class howManyParrot//おうちに何羽いるでしょうゲームの管理
{
    int answer = 0; //ユーザーが選択したゲームの解答を格納する変数
    PImage img//画像を格納する変数
    void fileimage();//画像を読み込む
    void imgMove();//読み込んだ画像を移動させる
    int inHouseCount();//家に何羽入っているか数える
    int outHouseCount();//家に収まらなかった数をカウント
    void init(int answerCount, )//クラス内変数を初期化
    void displayParrot();//画像を表示
    void (int scores)//スコアをカウント
    void goResult();//リザルト画面に遷移
}
```

4 その他

4.0.1 グローバル変数

```
int scores = 0; // 各ゲームにおけるスコアを計測する変数
int gameSelect; // どのゲームを選択したかを格納する変数
int diffiluctSelect; // どの難易度を選択したかを格納する変数
int leftButtonX, leftButtonY, middleButtonX, middleButtonY,
rightButtonX, rightButtonY; // スタート画面, 難易度選択画面, リザルト画面での3つのボタンの座標
```

5 制作計画

- 6/19:仕様書納品, 修正

- 6/26:制作開始
- 7/4,7/5:試作品 α 版テスト, デバック
- 7/10:試作品 β テスト
- 7/17:完成
- 7/31:リリース