

# パソコン教室向け採点業務 支援システム開発プロジェクト

[PM] 土屋

[メンバー] 柴田・田面・千々岩・富田・西川

A13

# プレゼンの流れ

---

1. 背景
2. 目的 / 目標
3. システム内容
4. 技術検証
5. 模擬試験問題作成に関して
6. 制約条件など

# 1. 背景

## 仮想クライアント詳細

仮想クライアント	株式会社 イー・トラックス
主 要 業 務	<ul style="list-style-type: none"><li>・ パソコン教室の運営 &amp; フランチャイズ事業</li><li>・ スマートフォンの修理事業</li><li>・ ビジネス向け教材開発事業</li><li>・ フィットネス健康関連事業</li><li>・ 映像コンテンツ開発事業</li><li>・ 新商品 &amp; 新業態開発                      etc...</li></ul>
事 業	manalgo (ハロー！パソコン教室)

# 1. 背景

## 企画の背景

---

- 生徒の情報などが紙媒体と既存システムの併用
- 模擬試験の採点を手動で行っている

## 1. 背景

# 採点システムの業務フロー

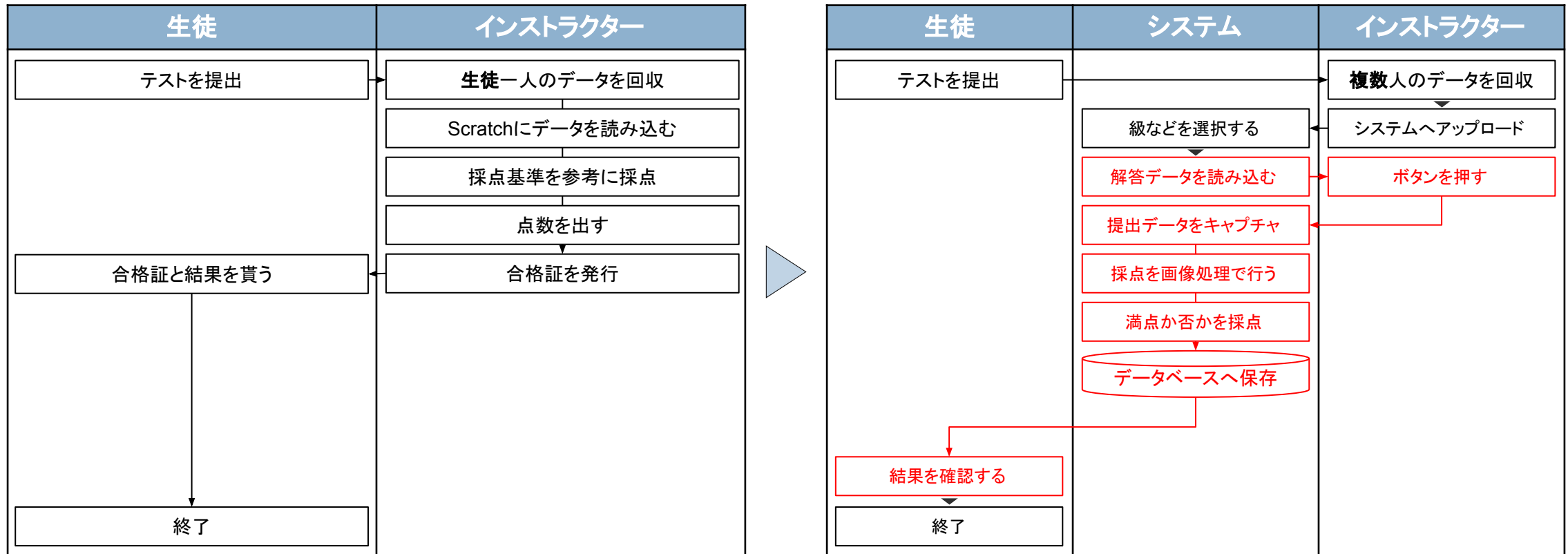
---

- ・決められた試験日があるわけではない
- ・採点を1度に行う数はまばらである

# 1. 背景

## 採点システムの業務フロー図

### Before / After



## 2. 目的 / 目標

### 目的・目標

---

#### 目的

生徒情報と採点結果の紐づけを行い一元管理をする

#### 目標

採点時間を短くしてフィードバック時間を増やす

生徒情報の管理を紙媒体から電子化へ移行する

### 3. システム内容

## 実装予定機能一覧

大機能	機能内容
生徒管理システム	生徒情報や採点の結果のデータを管理する。
採点システム	模擬試験の自動採点を行う。

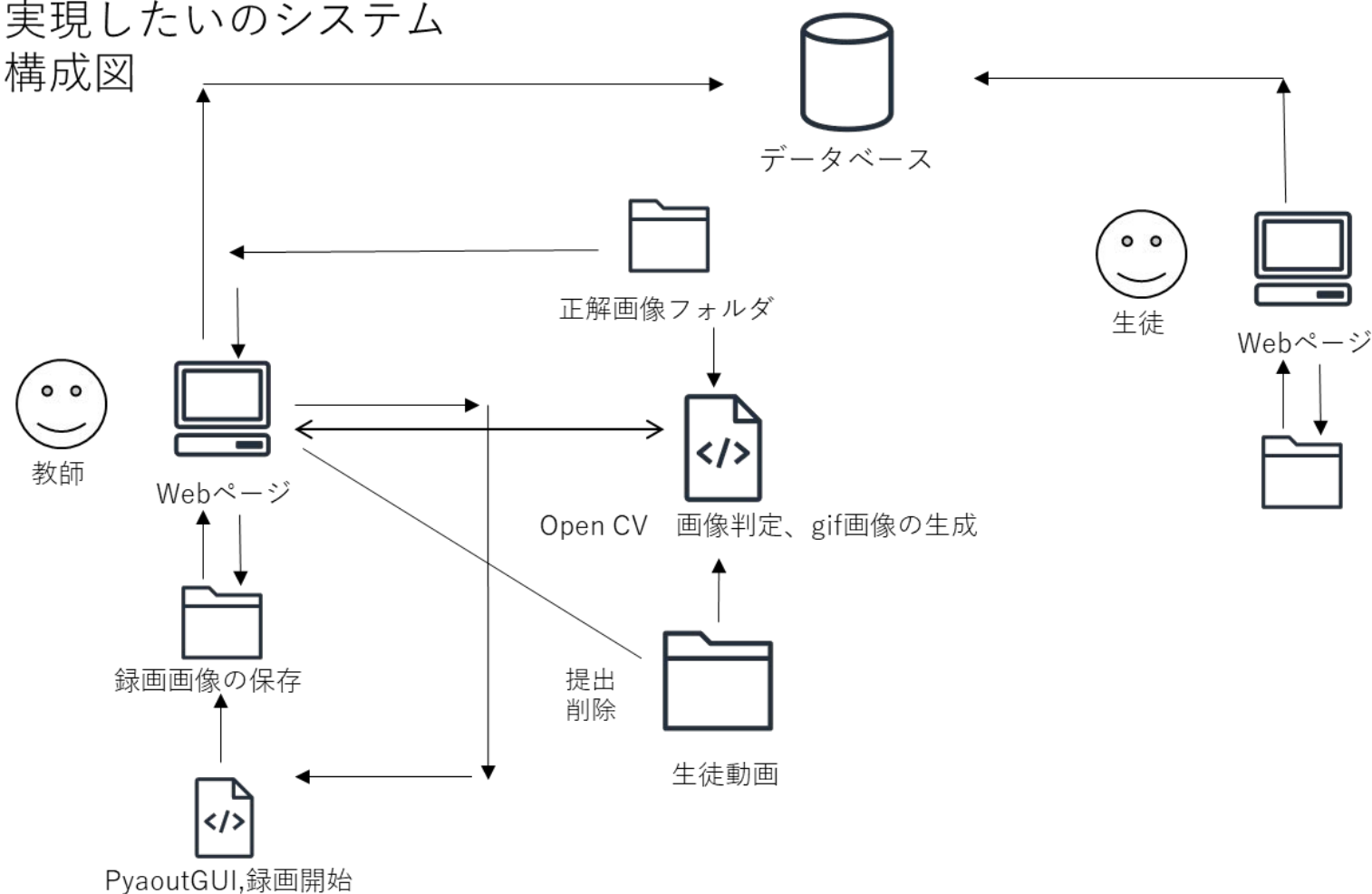
その他	内容
(模擬試験問題作成)	採点システムで使用する模擬試験の問題を「ジュニア・プログラミング検定」に準拠した内容で作成する。(エントリー級からゴールド級の4種類の問題を作成) ※実際の業務では作成しない。あくまで目的を確認するための作業である。 ※現在は作成した問題のみ模擬試験を登録。



### 3. システム内容

## システム構成図

実現したいのシステム  
構成図



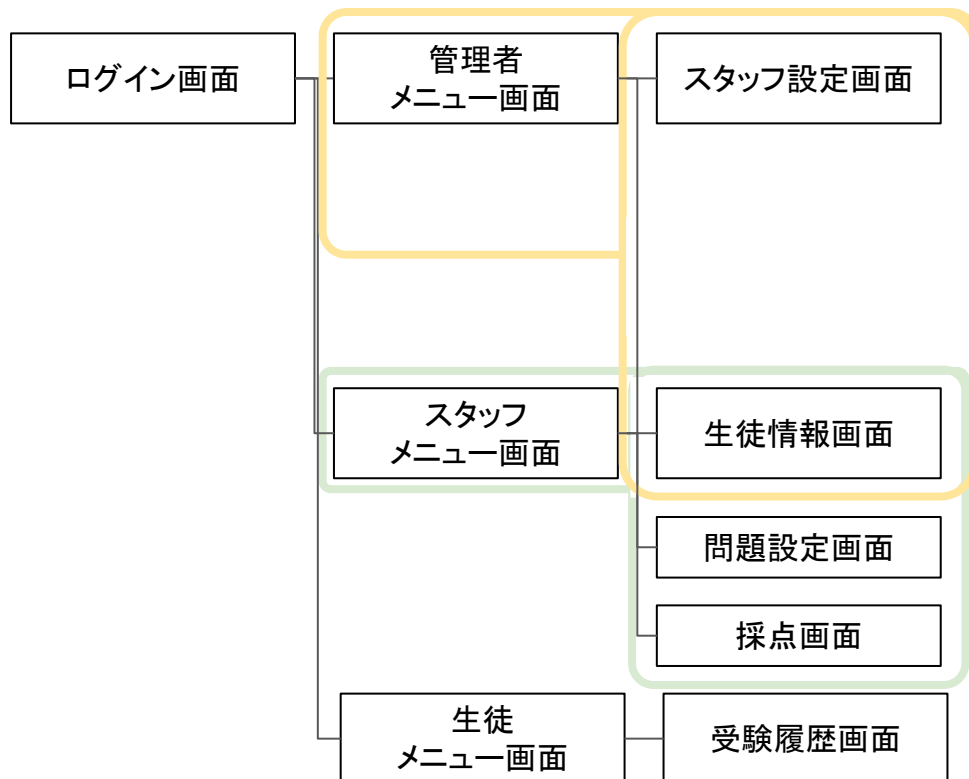
開発環境  
github

使用言語  
Python / HTML  
CSS / SQL

使用ライブラリ  
OpenCV  
PyAutoGUI

### 3. システム内容

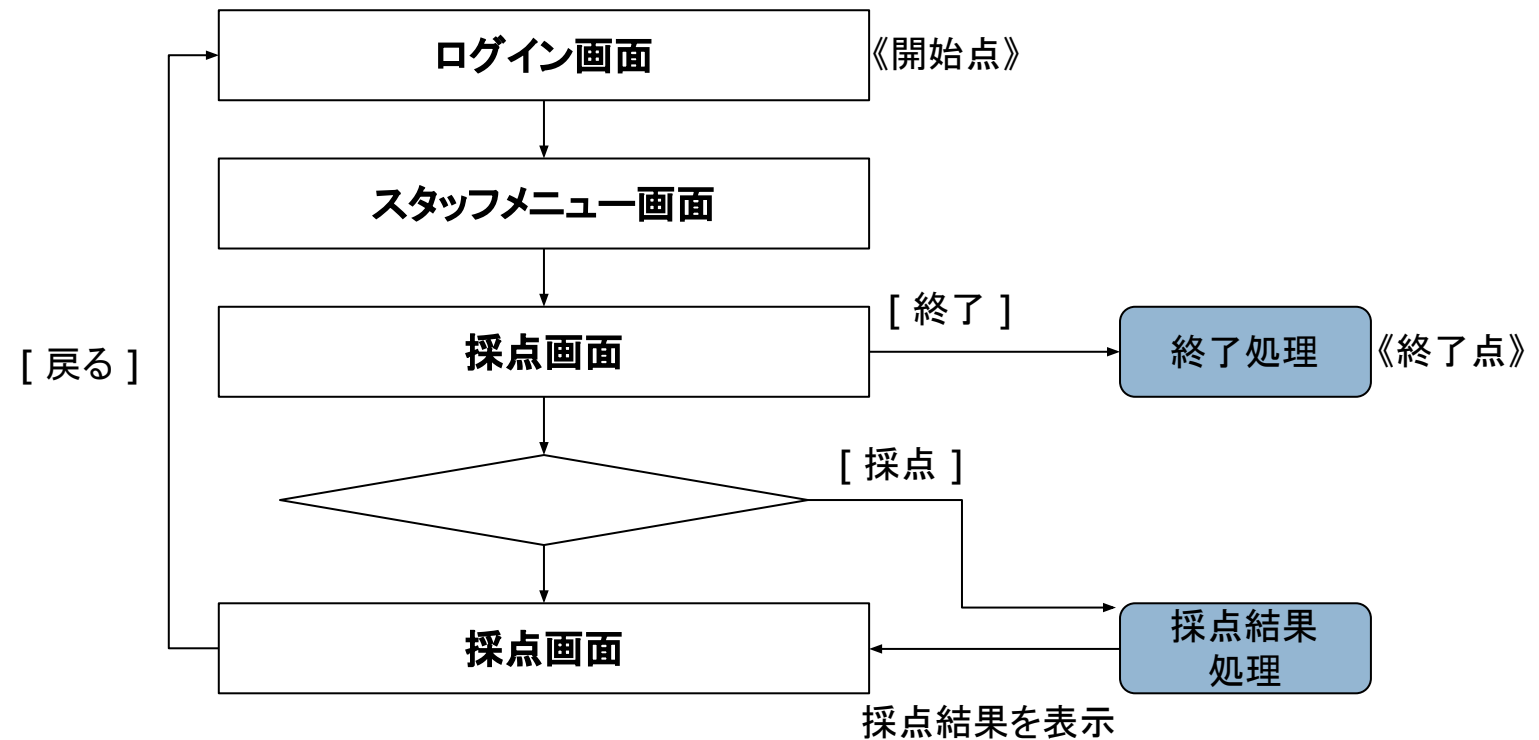
## 画面設計(画面一覧)



画面	機能詳細
スタッフ設定画面	<ul style="list-style-type: none"><li>・ スタッフの個人情報の確認/新規登録/変更/削除</li><li>・ IDとPassの発行/変更/削除</li></ul>
生徒情報画面	<ul style="list-style-type: none"><li>・ 生徒の個人情報の確認/新規登録/変更/削除</li><li>・ 生徒個人ID/Passの発行/削除</li><li>・ 模擬試験の採点結果の表示/管理</li></ul>
問題設定画面	<ul style="list-style-type: none"><li>・ 模範解答データのアップロード</li><li>・ キー入力等の録画に必要なキーの設定</li></ul>
採点画面	<ul style="list-style-type: none"><li>・ 採点問題の選択</li><li>・ 対象生徒の選択</li><li>・ 回答データのアップロード</li><li>・ 採点で使用する回答データを保存するフォルダの選択</li></ul>
受験履歴画面	<ul style="list-style-type: none"><li>・ 生徒別(本人)の成績情報の確認</li><li>・ これまでに受けた各試験の結果の確認</li></ul>

### 3. システム内容

## 画面設計(画面遷移)



# 3. システム内容

## 画面設計(画面レイアウト1/2)

ログイン画面

Scratch Scoring System

ユーザID

パスワード

メニュー画面(管理者)

Scratch Scoring System

ホーム  
スタッフ設定  
生徒情報  
ログアウト

○○校  
●●教室長 さん

スタッフ設定画面

Scratch Scoring System ○○校 ●●教室長

ホーム  
スタッフ設定  
生徒情報

スタッフ設定

新規作成

校舎名:  スタッフ氏名:

ID	氏名	フリガナ	所属校	電話番号	メールアドレス	入社日	ログインID	パスワード
***	**	***	○○	080-****-****	*****@gmail.com	20**/**/**	***	*****

メニュー画面(スタッフ)

Scratch Scoring System

ホーム  
生徒情報  
採点  
問題設定  
ログアウト

○○校  
スタッフ△△ さん

メニュー画面(生徒)

Scratch Scoring System

ホーム  
採点履歴  
ログアウト

○○校  
□□ さん

Scratch Scoring System ○○校 ●●教室長

ホーム  
スタッフ設定  
生徒情報

スタッフ設定

新規作成

校舎名:  スタッフ氏名:

ID	氏名	フリガナ	所属校	電話番号	メールアドレス	入社日	ログインID	パスワード
***	**	***	○○	080-****-****	*****@gmail.com	20**/**/**	***	*****

スタッフ情報変更

ID:  ログインID:  ログインPW:

氏名:  フリガナ:

所属校:

電話番号:  メール:

入社日:

Scratch Scoring System ○○校 ●●教室長

ホーム  
スタッフ設定  
生徒情報

スタッフ設定

新規作成

新規スタッフ登録

ID:  ログインID:  ログインPW:

氏名:  フリガナ:

所属校:

電話番号:  メール:

入社日:

校舎名:  スタッフ氏名:

ID	氏名	フリガナ	所属校	電話番号	メールアドレス	入社日	ログインID	パスワード
***	**	***	○○	080-****-****	*****@gmail.com	20**/**/**	***	*****
***	**	***	○○	070-****-****	*****@gmail.com	20**/**/**	***	*****
***	**	***	○○	080-****-****	*****@gmail.com	20**/**/**	***	*****
***	**	***	○○	090-****-****	*****@gmail.com	20**/**/**	***	*****
***	**	***	○○	080-****-****	*****@gmail.com	20**/**/**	***	*****

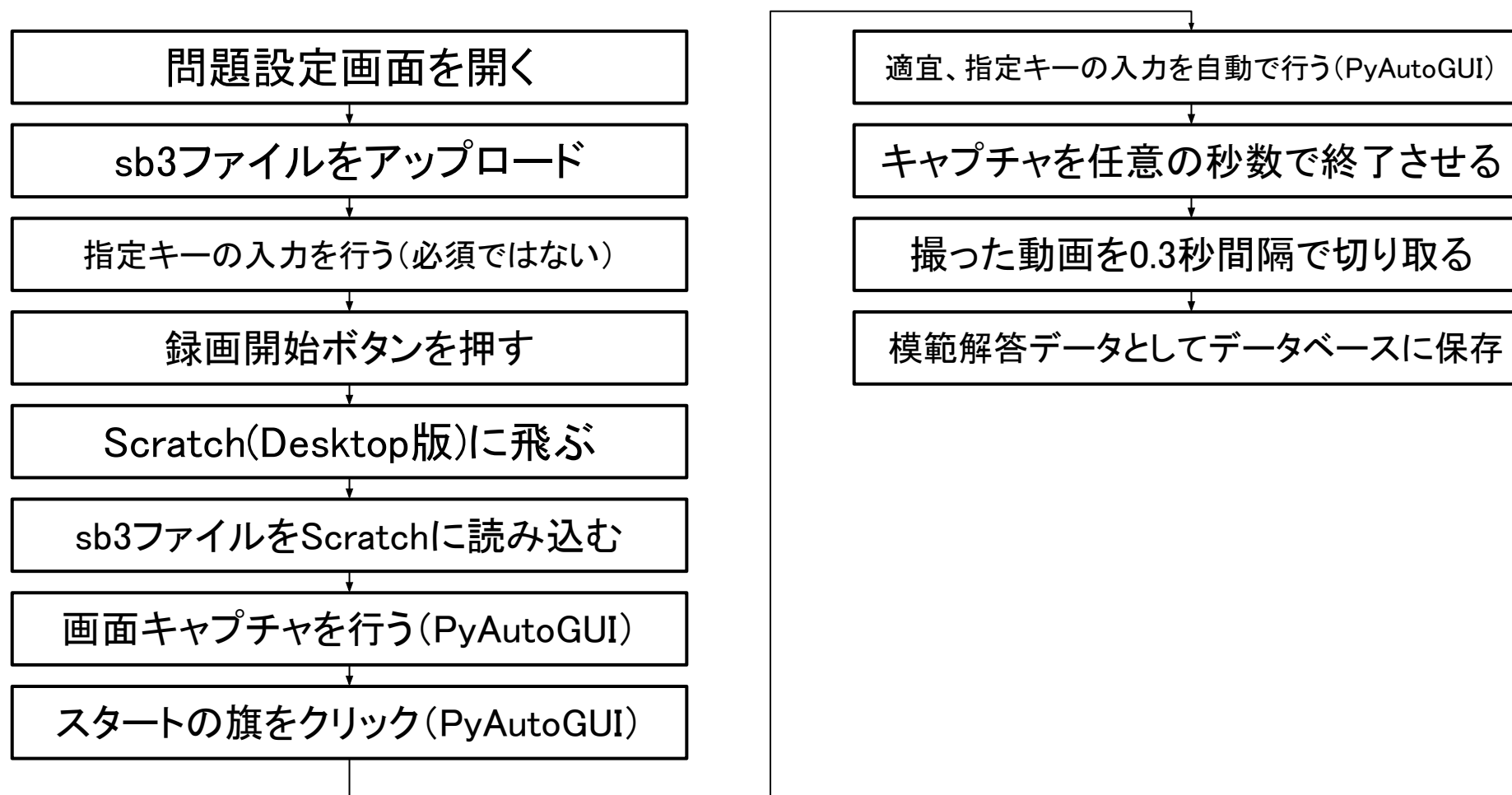
## 画面設計(画面レイアウト2/2)

## 問題設定画面

[illegible]12

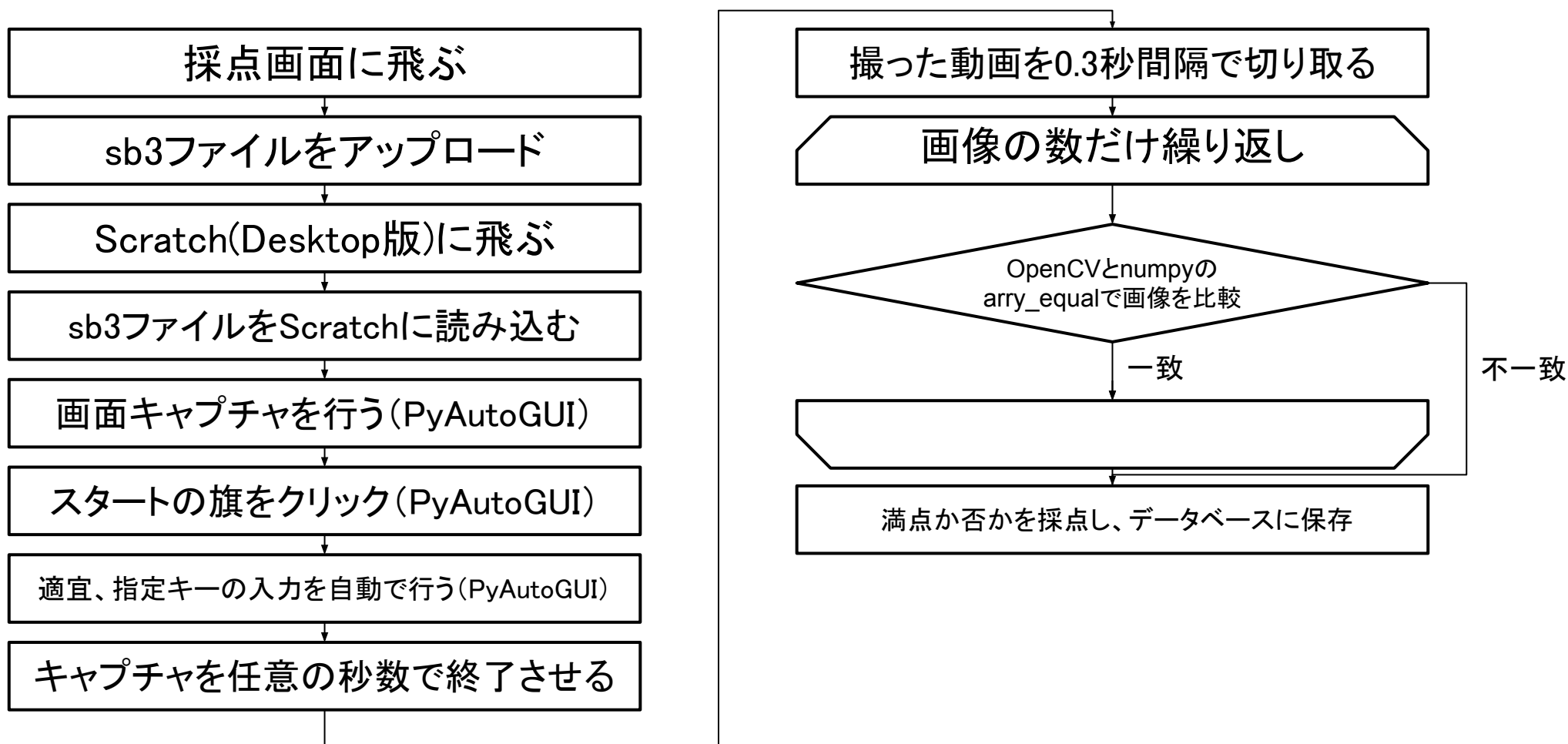
### 3. システム内容

## 模範解答データ 処理フロー(画像)



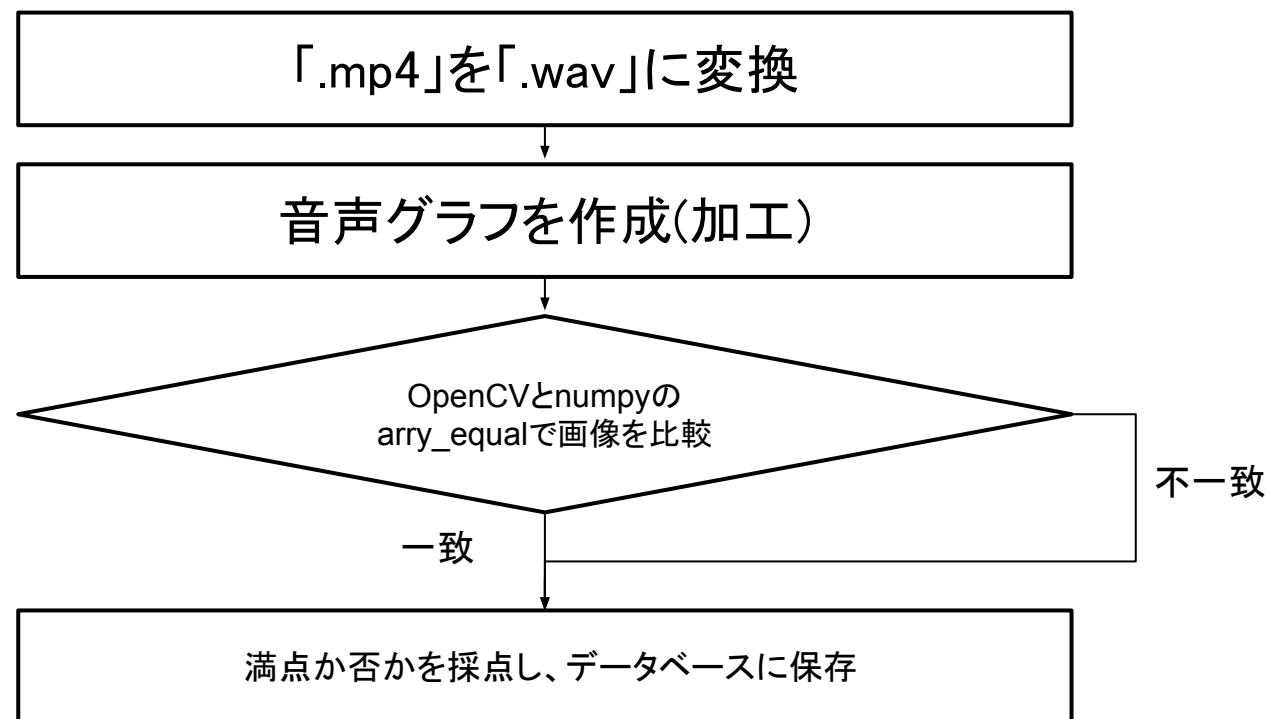
### 3. システム内容

## 採点システム 処理フロー(画像)



### 3. システム内容

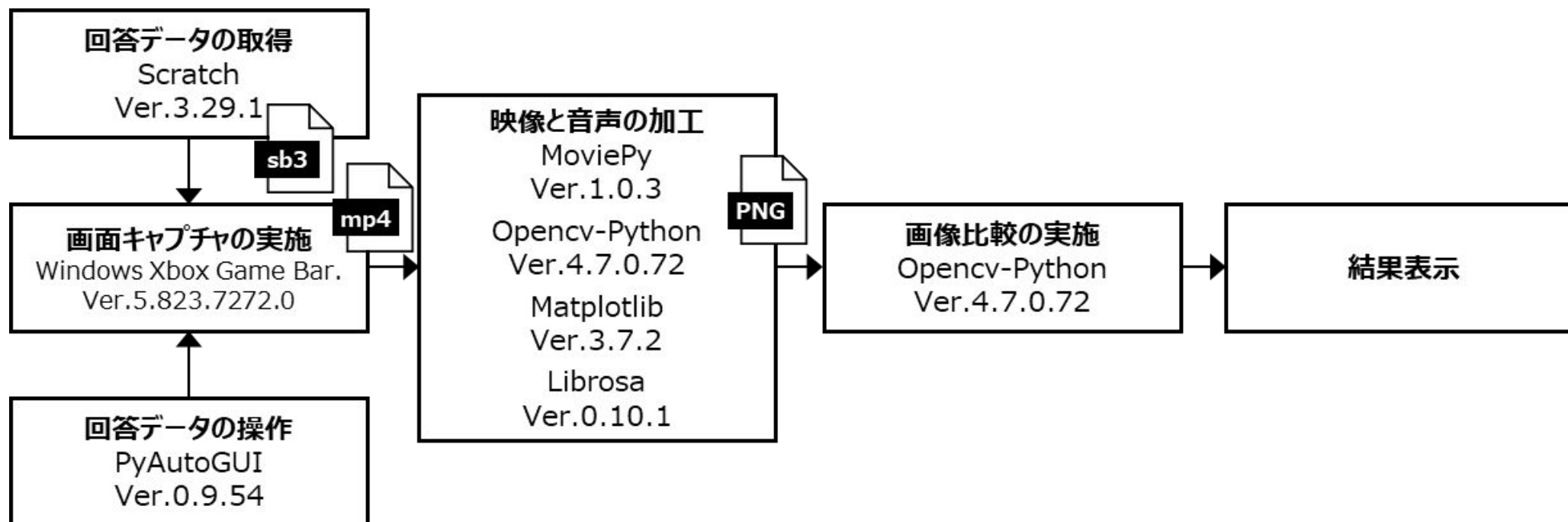
## 採点システム 処理フロー(音)





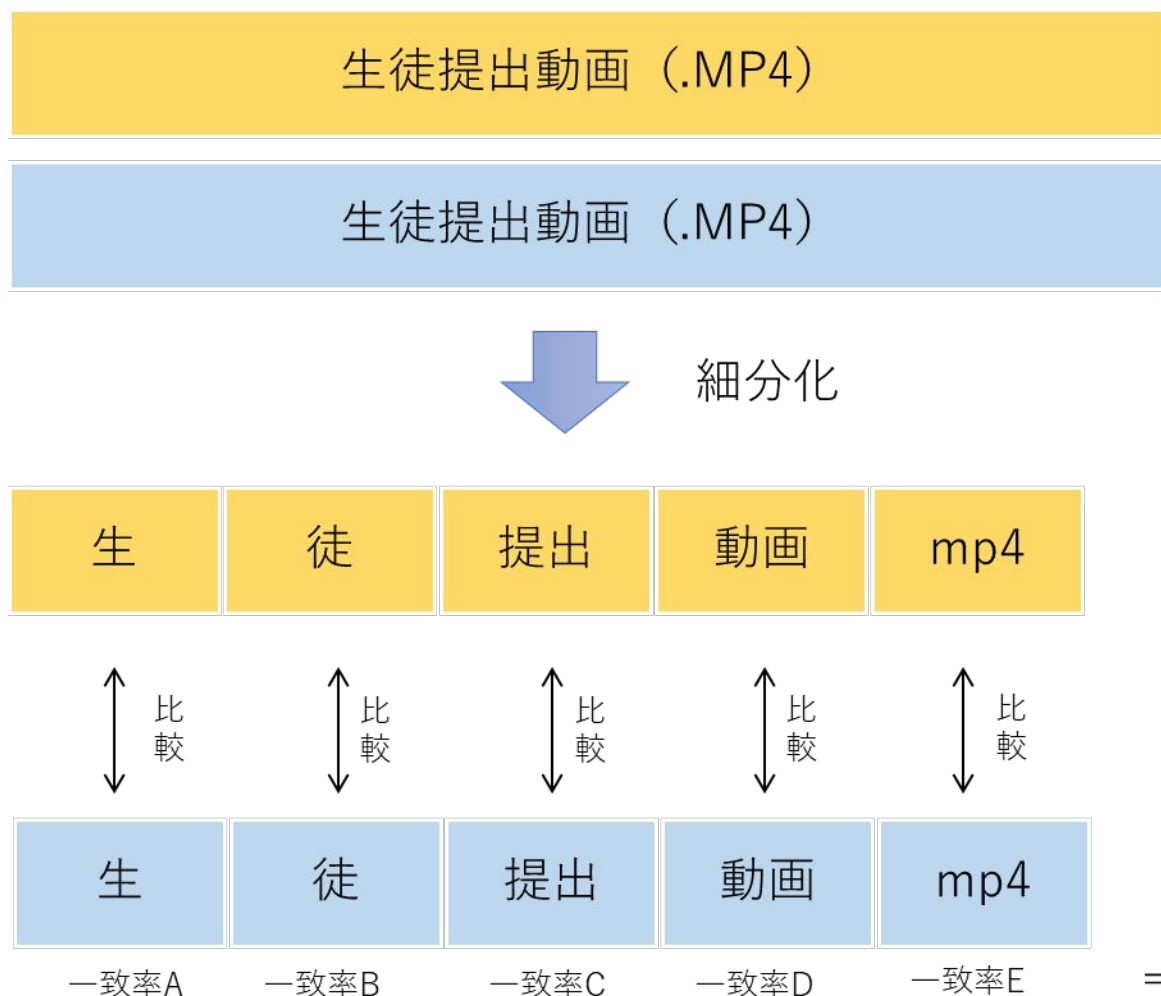
### 3. システム内容

## 採点データ 処理の流れ(比較)



### 3. システム内容

## 条件処理内容の視覚化



#### 画像生成について

1秒間に3枚程度の画像を生成する  
各級の録画時間は20秒前後

#### 判定動作条件

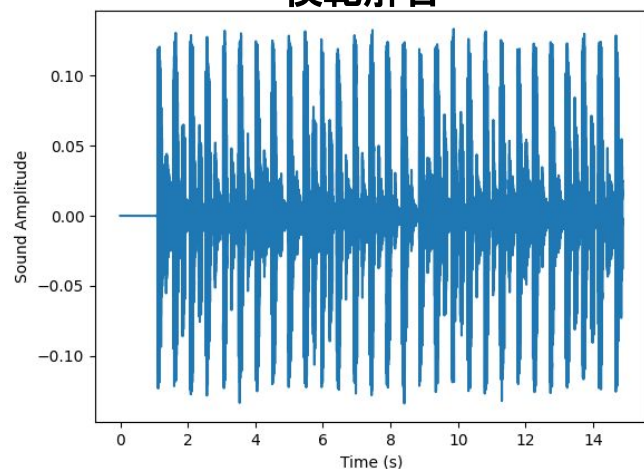
一致率が98%以上であれば正解とする

= 満点か否か採点を行う

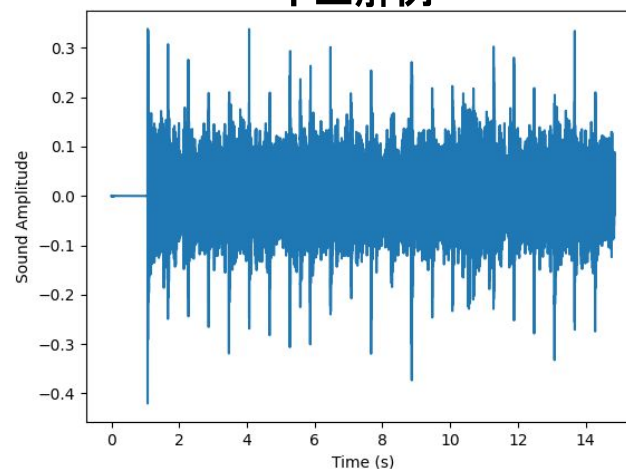
### 3. システム内容

## 音の比較

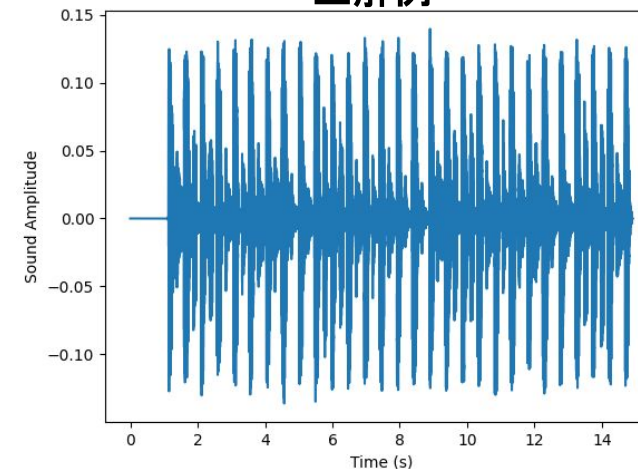
模範解答



不正解例



正解例



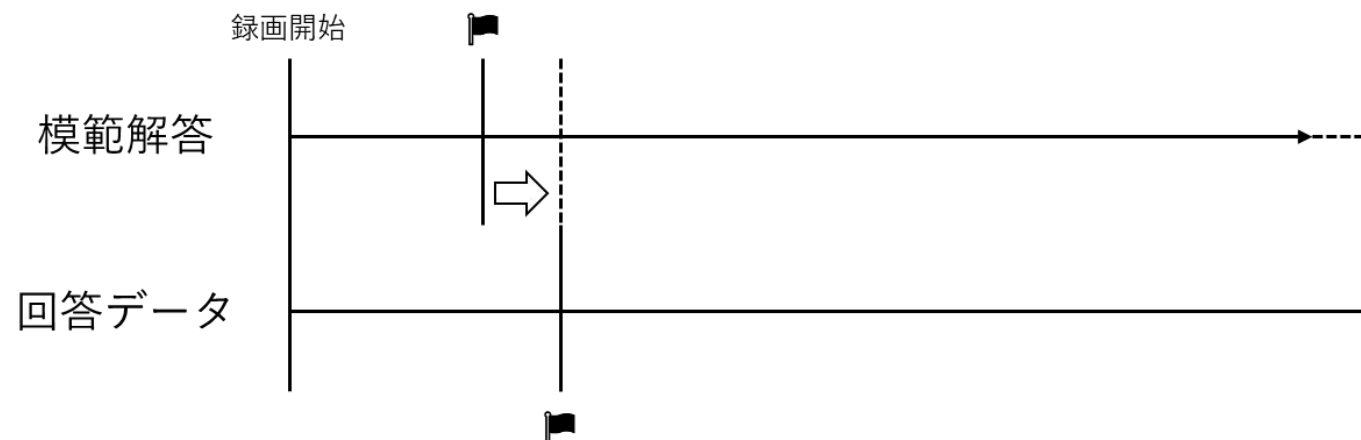
### 音の比較について

波形を抽出し、画像での比較を実施

一致度 93%で正解とする / 精度 100%

### 3. システム内容

## タイムラグの影響



- ・ 開始のタイムラグにより正答が誤答になる  
→ 模範解答と回答データの動き始めを合わせる事で調整

### 3. システム内容

## 動作の比較

対象級	正解データ					
	1	2	3	4	5	6
エントリー	正答	誤答	誤答	誤答	正答	正答
ブロンズ	誤答	誤答	誤答	誤答	誤答	誤答
シルバー	誤答	誤答	誤答	正答	誤答	誤答
ゴールド	判定不可					

不正解データの採点はすべて正常

正解データは半分以上が撮影中のタイムラグにより、誤答の判定  
ゴールド級は、動き始めの画像が一致しないため判定不可

### 3. システム内容

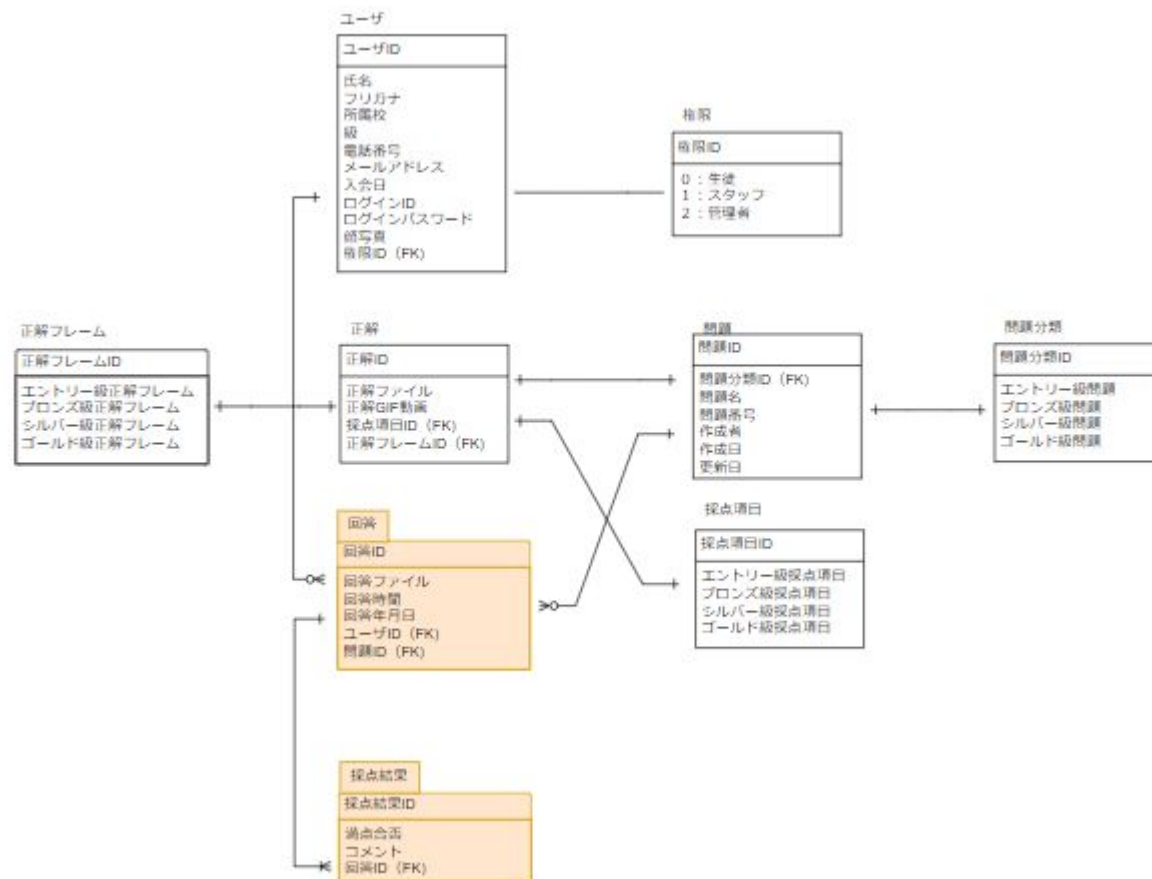
## 動作・音について

---

- キーのイベント登録  
→ 再度調査中
- Scratch実行中、タイムラグによる採点の影響  
→ 対応策を検討中

### 3. システム内容

## DB設計(リレーション)



### 3. システム内容

## DB設計(テーブル一覧 / 定義)

#### 《 テーブル一覧 》

エンティティ一覧

No.	エンティティ	エンティティ種別	定義
1	ユーザ	リソース系'	個人情報一覧
2	権限	リソース系'	生徒・スタッフ・管理者の識別
3	正解	イベント系	正解情報
4	正解フレーム	イベント系	正解 GIF 動画のフレーム
5	問題	イベント系	問題情報
6	問題分類	イベント系	問題の 4 つの級の分類
7	採点項目	イベント系	各級の採点項目
8	回答	イベント系	各生徒の回答情報
9	採点結果	サマリ系	採点結果情報

#### 《 テーブル定義 》 別途資料参照。



## 5. 模擬試験問題作成に関して

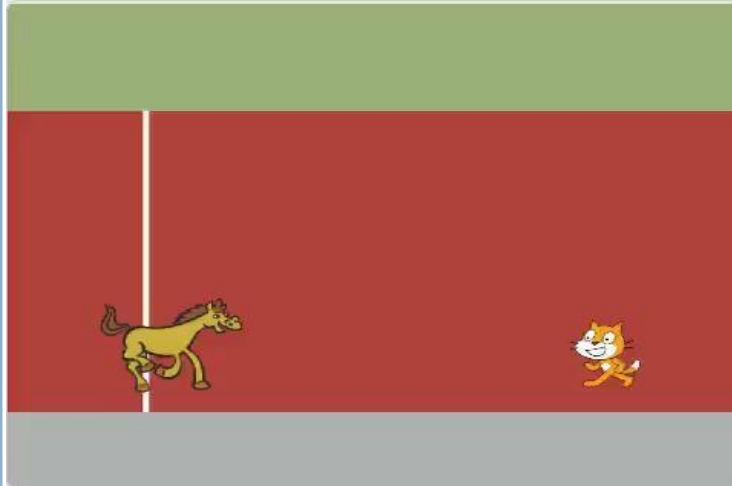
# 各級の作成内容

級	作成内容
Entry	制御やイベントなどのブロックを使って、動きや見た目の変化を中心とした簡単なスクリプトを作成する。
Bronze	変数や演算・音・ペンなどのブロックを使って、キー入力やマウス操作などに反応する簡単なスクリプトを作成する。
Silver	複数の制御ブロックや入れ子構造などを使って、キー入力やマウス操作と演算を組み合わせた複雑なスクリプトを作成する。
Gold	クローンやリストのブロック、新しいブロックなどを使って、様々なスプライトを連動させたスクリプトを作成する。

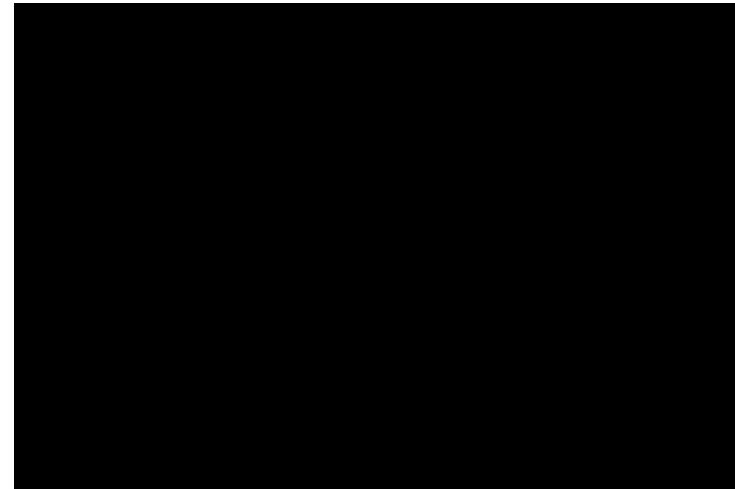
## 5. 模擬試験問題作成に関して

### 作成した各級のもんだい

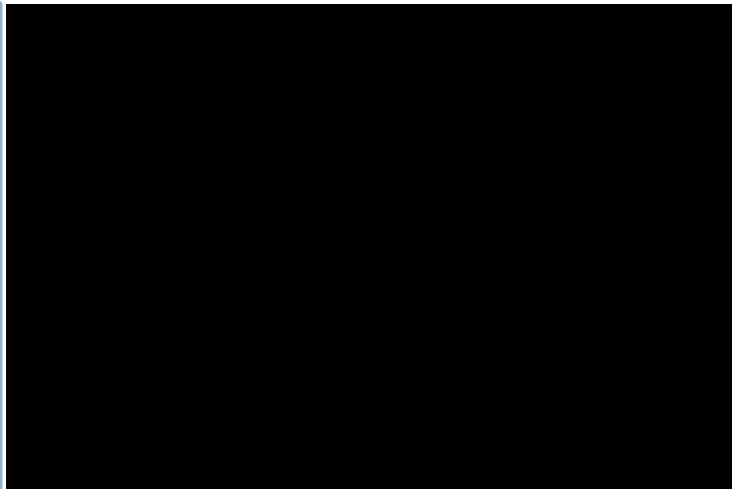
エントリー級



ブロンズ級



シルバー級



ゴールド級



## 6. 制約条件など

### 制約条件

---

- アレンジは考慮しないものとする
- ブロック「乱数」の問題は含まないものとする
- 採点を行う際は、半角入力に設定するものとする

# 参考文献

---

# SI Object Browser ER | 「ER図とは？書き方やテクニックをわかりやすく解説」より

<https://products.sint.co.jp/ober/blog/create-er-diagram>

# Qiita | 「注文・カート機能のER図について」より

<https://qiita.com/codeReona/items/f7f0eef3ababef130018>

# Qiita | 「エンティティとは DB設計」より

<https://qiita.com/ooyy0121/items/bc35c3e1799d5203aa14>

# Qiita | 「ER図を基本からまとめてみた【入門】」より

<https://qiita.com/kanfutrooper/items/7ed3f3229a466d641cdb>

# Knooto | 「システムのアクセス制御(捜査権限管理)設計」より

<https://knooto.info/software-design-access-control/>

# アンドエンジニア | 「データベース設計には必須！ER図に書き方を解説！」より

<https://and-engineer.com/articles/YZ9pUBIAACAe386>

# Qiita | 「【DB設計】No.1『DBのリレーションとER図』」より

<https://qiita.com/kohki4115/items/75e0618b3830f96bfa1b>

# Edrawsoft | 「ER図の例とテンプレート」より

<https://www.edrawsoft.com/jp/er-diagram-examples.html>

# IPA | 「機能要件の合意形成ガイド」より

<https://www.ipa.go.jp/archive/files/000004509.pdf>