

# 24 班 最強の勉強アプリを作ろう

～勉強とゲームの融合を目指して～

Making The Best App for Studying

## 要旨

アプリのもととなるプログラムは Python を使って作成。アプリのデザインや UI などは Canva を使って作成し、より大衆ウケするものにする。ゲーム内のキャラクターやアイテムなどのグラフィックは Canva の生成 AI または android の作成アプリを使って作成し幅広い年齢層で好ましいと思えるようなものにする。それらを一つのアプリに集約し、今までにない利便性を持つアプリにする。

## Abstract

The core program of the app is developed using Python. The app's design and user interface are created with Canva to make it more appealing to a wide audience. Graphics for in-game characters and items are also produced using Canva's AI features or Android design apps, aiming to appeal to a broad age range. All of these elements are integrated into a single app, resulting in a uniquely convenient user experience unlike anything currently available.

## 1 研究背景と研究目的・意義

### 1.1 研究背景

勉強アプリによってそれぞれのメリット・デメリットがあり、それらのデメリットをなくし、統合することによって勉強に意欲がない人でも勉強習慣が身に着けられるより良い勉強アプリを作成することができると思った。

### 1.2 リサーチクエスチョン

学生層において勉強習慣を身に着けるためにはどのようなアプリを開発するのが最適か

### 1.3 研究の目的・意義

- ・勉強への積極性を向上させるための方法の模索
- ・人が便利と感じ、継続的に利用できるアプリの作り方、また、そのコンテンツの研究
- ・プログラミング言語の学習とその利用。それぞれの言語の特徴を把握すること
- ・アプリの利用者がより好ましいと思うグラフィックの作成方法の模索
- ・現在市場に存在する勉強アプリのメリットとデメリットを調べる

### 1.4 仮説とその根拠

勉強機能だけでなく娯楽となる機能(ゲームなど)も実装することで勉強への意欲が高まる。

これは私たちが勉強をするうえで使用してきた勉強アプリでは勉強が長続きしなかった原因に、楽しさの不足があると考えたからである。

## 2 研究方法

Android OS で試作品を作る。また、ゲーム要素を抜いた試作品を作り、快適に操作できるか、不便な点はないか、需要を満たしているのか、バグはないかを調査する。

そして、これを第三者にテストしてもらい、このアプリのメリット、デメリットや、やる気の向上につながったかなどの意見を募る。

最終的に、他校や大学にも公開し、より大規模に意見を募る。

( $\alpha$ テストや $\beta$ テストの実施)

### 2.1 作成した勉強アプリの構造について

今回作成した勉強アプリは「勉強画面」と「ゲーム画面」に大別される。

- ・勉強画面

勉強をするための機能が集約された画面で、以下の機能を持つ。

- ・ポモドーロタイマー
- ・勉強時間の分析

- ・ゲーム画面

ポモドーロタイマーで勉強した時間に応じて得られる「ポイント」を使用して遊ぶことができるゲームを遊ぶための画面。

ゲームはRPG形式である。

## 2.1 研究と分析

- ・UIの作成

アプリのシステム部分（ロード画面や設定画面）のほか、アプリの機能やゲームのUIデザインをアプリの内容の決定後、作成した。（参照：図I、図II）

アプリ全体に統一性を持たせたり、機能にあったテーマを選んだり、ゲームの雰囲気をつくりだしたりするためにデザインやフォントの編集を工夫した。

いくつか試作ページを作ったり英語版を作ったりして、必要なものを決定しベストになるようにした。



図I

←色については優しい雰囲気、アプリを使っていてリラックスできるように暖色をベースに作成。全体的に同じような色を使い、アプリに統一感を持たせた。1から自分たちが作ろうとしているアプリに必要な設定項目を考え追加した。

→フォントはよりゲームの雰囲気を出せるよう、文字を詰めすぎないようにし、画像については、RPGゲームの雰囲気をもっとだせるように壮大な、リアルな出来にした。



図II

- ・ゲーム内アイテム・背景の作成

ゲームのテーマ(世界観、色合い)を決定後、Canva(参照:図III)、android内の作成アプリ(参照:図IV、V)を使ってテーマに沿うような形で作成した。

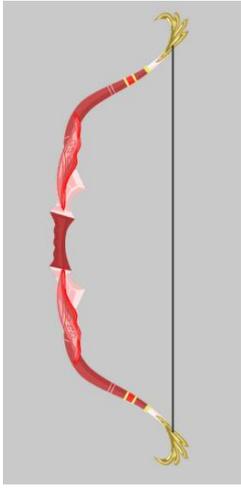


図III

←洞窟をイメージにして作成。全体的な容量削減、大衆ウケする見た目にするためにドットベースにした。

背景にモンスターやキャラクターを配置した際、見えにくくなることを避けるために、一般的な洞窟よりも明るめになるように色合いを工夫した。

キャラクターデザインの担当者と話し合い、よりモンスターやキャラクターに雰囲気があったものにした。



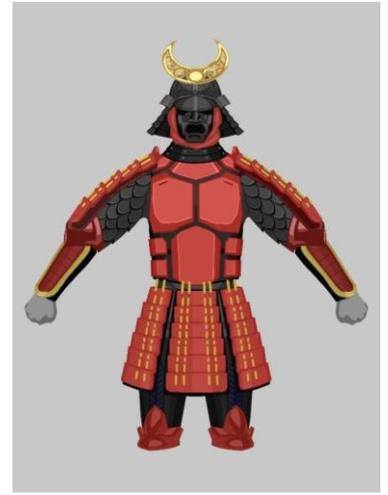
図IV

←複数のゲームを参考にして作成。和風ベースで作成した。

RPG のコンセプトにあった見た目にし、印象深く、かつ高級感ある見た目にするため色合いや装飾品を赤や金色で統一した。

→日本の武将の甲冑をもとに作成。(右の図は真田幸村)

男性ユーザーを増やすためかっこよさを重視し、黒色、赤色を多用した。



図V

#### ・キャラクター及びモンスターのデザイン

##### →モンスターのデザインに関して

ほかのゲームとあまりデザインが似すぎないように、オリジナル性を持たせた。

大衆向けの作品となるよう、人が恐怖や嫌悪感を抱くようなデザインではなく、かわいいと感じるものや、親しみを持ちやすいデザインを心掛けた。

##### →プレイブルキャラクターのデザインに関して

二頭身のキャラクターにすることで、かわいらしさをを持たせた。また、動きを単純化したり、衣装を簡素なものにできるようにしたりすることで、製作面の負担を抑えた。

デフォルトの顔を一つ作り、目・髪の色や形を自由に設定できるようにすることで、キャラクター製作の負担を減らしつつ、様々な姿にできるようにした

#### ・プログラミングについて

開発環境を Unity から Python と AndroidStudio に変更した。これによって使用言語が Python, Java, Kotlin の3言語に増えたが、より効率的にプログラミングができた。

単純なタイマーではなく、勉強のためのポモドーロタイマーを作成した。

全体で約1万3000行のコードとなった。内訳としては、Pythonが約2000行、Javaが約7000行、Kotlin約4000行となった。しかし、評価される時間が少なくプログラマーが一人であったためデバッグまでは行えていない。

コーディングについて、できるだけ可読性を高めバグが発生しないようにするため、依存関係を明確にし、一つにまとめられる処理は関数にするなど工夫を凝らした。

他にも、できるだけたくさんの方がこのアプリを利用できるように Android のバージョンを少し古いものにしたたり、どのスマホにも搭載されている機能のみを使用したりして世界にある 97.5%の 안드로이드端末で使用できるようにした。

## 2.2 結果

・ゲームの機能を省いた試作品をインターネットに公開し、意見を募った。

その結果、1000人ほどから意見をいただき、具体的には

・英語にしか対応していない ・携帯の機種によって、画面に異常が出る

・バグが多い ・様々な機能が、わかりやすいUIとともにまとまっている

などの意見が多かったほか、男女比が、男性：3 女性：7という結果となった。

デザインに関しては、アナログ上で描いたものをデジタル上に持ってくるのが難しい。

## 2.3 考察

シンプルなデザインが女性に好感をもたせているのではないかと。そして、ゲーム機能を

追加することで、男性にも好感が持てるものに改良できるのではないか。

### 3 結論と今後の展望

#### 3.1 結論

結論としてタイマー機能やスケジュール機能などさまざまな勉強アプリのいい機能をとってひとつにしたり、デザインを統一したりして使いやすい最強のアプリを作成し、楽しく手軽に遊べるゲームを追加することでアプリユーザーの学習の意欲を向上させた。

#### 3.2 今後の展望

今回インターネットを介して何名かに試作品を使用してもらったことで得た様々な意見をもとに改善を重ね、完成品を目指す。

UI, プログラム, イラストを約1年進めたことで、実際にアプリを作ることの難しさや、我々の持つ知識や機材による限界を知ることができた。この学びを活用してデザインやプログラムをより手の込んだものにし、ユーザーが使いやすく、面白いと感じられる作品にする。

### 4 引用文献・参考文献

Canva, SwordMaker, bowmaker2, amormaker, 8bit painter, unity