

相互排除が必要なアルゴリズムの Rust 言語による記述の試み

学生証番号 744104 コンピュータ理工学部 市場 友也（荻原研究室）

1 はじめに

身の回りには排他制御が必要な並行処理にまつわる問題が数多く存在する。並行処理では処理する側にも消費されるデータにも規則を適用しなければならないが、規則を適用すると今度はデッドロックなどの問題が発生することも多い。

本研究では並行プログラミングのための機構を多く備えている Rust 言語を利用して相互排除が必要なアルゴリズムの記述を試みた。

2 検討内容

Rust 言語の機能を使い、典型的な並行処理の問題である「食事する哲学者」と「居眠り床屋」の問題の記述を試みた。

記述では、計算リソースを効率的に使用できているかということと公平性が担保されているかを確認する。これらの問題における公平性とは、消費スレッドと生産スレッドが偏りなく動作することである。また、計算リソースの効率性は消費スレッドが余分に実行されていないということである。

「居眠り床屋」の問題

床屋の椅子は1つだけで、ここに客が座れたら散髪が開始される。理容師は散髪を終えると客が待合室にいないか確認し、いないなら散髪用の椅子で眠る。訪れた客は、誰かが散髪中なら待合室で待つが、理容師が寝ていれば起こして散髪をしてもらう。待合室の椅子がいっぱいならあきらめて帰る。

3 実験

「食事する哲学者」（哲学者は3人とした）と「居眠り床屋」の問題を条件変数と Mutex を用いて記述する。Rust が持っている所有権構造によって条件変数や Mutex が安全に解放され、正しく動いていることを出力で確認する。

本研究では他に、銀行家のアルゴリズムを適用

したプログラムの記述も試みた。また、ライフタイムパラメータを伴うジェネリクス機能を活用したプログラムの記述も行った。

4 議論

「食事する哲学者」の問題において出力は正しいので3人の哲学者がそれぞれ箸をロックして安全に解放できている。これはMutexによって共有資源“chopsticks”が適切に相互排除されているからである。

「居眠り床屋」の問題の出力からは適切に計算リソースが活用できていることが分かる。これは、待ち行列で客の人数を管理するとともに、条件変数を利用して理容師を起こすという動作が実現できているからである。

5 まとめ

Rustの所有権構造と相互排除に必要な条件変数やMutexを使って典型的な問題を記述した。

Rustの特徴的な機能には他に、オブジェクトがロックされた後でスコープを抜けると自動的にアンロックされる仕組みや、ライフタイムパラメータを指定したジェネリクスの機能などがある。ただし、プログラムが複雑になる上、それらについて十分な知識がないとデッドロックなどの問題が発生する原因ともなる。

Rustで記述しても並行処理が容易に記述できるわけではなく、十分な検討が必要である。

参考文献

- [1] E. Dijkstra: "Cooperation sequential processes", (<https://www.cs.utexas.edu/users/EWD/transcriptions/EWD01xx/EWD123.html>). 参照 2024-01-17)
- [2] Mozilla: "The Rust Programming Language 日本語版", (<https://doc.rust-jp.rs/book-ja/title-page.html>), 参照 2018-12-08).

Web アプリケーションに対するリファクタリングの適用実験と品質向上に関する考察

学生証番号 053246 情報理工学部 岡部 竜典（荻原研究室）

1 はじめに

ハッカソンで開発した WebAR アプリケーションは、インバウンド観光客向けの情報提供を目的としていたが、コードの品質に課題があった。限られた時間内での開発は、長い関数や理解しにくいコードを生み出し、後の拡張性に支障をきたす可能性があることを明らかにした。これらの問題に対処するため、マーチン・ファウラーによるリファクタリングのテクニックをプロジェクトに適用し、コードの整理と品質向上を図った。

2 提案手法

本研究では、コードの整理と品質向上を目指し、マーチン・ファウラーが提唱するリファクタリングの手法を WebAR アプリケーションに適用した。具体的には、関数の抽出、関数名・変数名の変更、関数の移動、問い合わせによる一時変数の置き換えを行い、コードの可読性を向上させた。また、パラメータオブジェクトの導入や関数群のクラスへの集約は検討したもの、最終的に外部モジュール化という個人的な工夫を採用し、プロジェクトのコード構造改善に貢献した。

3 実験概要

実験では、サインアップ機能に焦点を当て、選定したリファクタリング手法を段階的に適用した。このプロセスは 2~3 週間にわたり行われ、主要な処理に改善が施された。特に、Firebase を用いたユーザ認証、イベント取得、バリデーション、サインアップ処理、エラーハンドリングに注目し、コードの可読性や効率性を向上させる目的でリファクタリングが

行われた。実験終了後、参加者に対してアンケートを実施し、リファクタリングによるコードの品質向上と効率性の改善を評価した。

4 議論

リファクタリングにより得られた主要な気づきは、データと処理の統合がコード構造を明確にし、保守や拡張を容易にしたことである。設計上の洞察からは、関数間の関連性の低さが明らかになり、外部モジュールへの分離が有効であることが示された。コード量は約 2 倍に増加したが、アンケート結果はリファクタリングによる可読性の向上と理解時間の短縮を示している。これらの結果から、リファクタリングはコードの外観改善に留まらず、開発プロセスの効率化、エラー対応の迅速化に寄与し、保守性を向上させることが確認された。

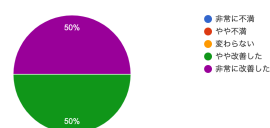


図 1. コードの可読性

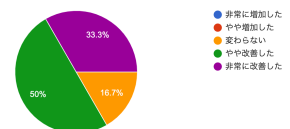


図 2. 理解時間

5 まとめ

リファクタリングを通じてソフトウェア品質を向上させ、コードの明確化と機能統合に成功した。アンケート結果は、リファクタリングが可読性と理解時間の短縮をもたらし、開発プロセスの効率化に寄与したことを示した。

参考文献

- [1] Martin Fowler: “リファクタリング-既存のコードを安全に改善する-”, 株式会社オーム社(2021).

筋力トレーニングの実施状況を 検出可能な支援アプリの開発に向けて

学生証番号 054047 情報理工学部 濱野 洋大（荻原研究室）

1 はじめに

近年、健康志向の高まりなどから筋力トレーニングが注目されている。スマートフォンのアプリなどでトレーニングの進捗を管理することが考えられるが、実施の有無や回数を入力するだけでは、トレーニングが本に行われたのかは分からない。

そこで、トレーニングの様子をカメラで検出し、それを実施記録として残すことができるアプリが開発できれば、達成感を得ることができ、モチベーションの維持にも繋がるであろう。

本研究では、スクワットと腕立て伏せの回数を数えることができる機能の開発を行なった。

2 機能の概要

本研究では、入力された動画に対して姿勢推定を行い、スクワット、腕立て伏せが正しく行われた場合の姿勢を検出できた場合に1回とカウントするようにした。姿勢推定にはオープンソースのライブラリであるMediaPipe[1]を用いた。

図1は動画ファイルを入力とした場合の実験例である。検出されたスクワットの回数が画面左上に表示されている。

3 アプリケーションの実装

本研究では、動画入力から回数をカウントする機能の実現に重点を置いて開発を行なった。

スクワットの条件は以下の通りである。

- ・ 膝の角度が 85 度以下になっている
- また、腕立て伏せの条件は以下の 2 つである。
- ・ 肩から足までが一直線になっている
- ・ 膝の角度が 70 度以下になっている

4 まとめ

本研究では、筋力トレーニングのモチベーションを維持させることが難しいということから、その問

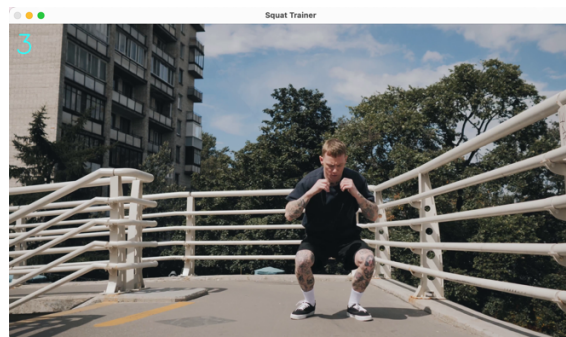


図1: アプリの画面

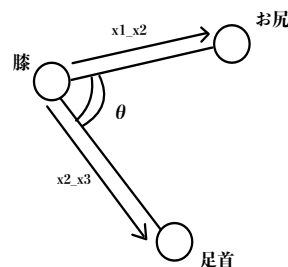


図 2: 角度計算の図

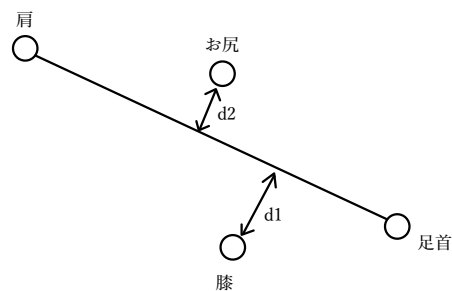


図 3: 点と直線の距離の図

題を解決すべく、モチベーション維持のためのアプリの提案を行なった。

アプリの機能として、トレーニング仲間と結果を共有して共に頑張るようにしたり、進捗を競い合ったりすることで、さらなるモチベーション維持につながるのではないかと考える。

参考文献 [1] Google: 「MediaPipe」

<https://developers.google.com/mediapipe>

学習者が項目間の関連付けを行える 暗記学習支援アプリの提案と試作

学生証番号 054452 情報理工学部 山本 敏暉（荻原研究室）

1 はじめに

語学や歴史をはじめ、学習のためにはある程度の基本的な事項の暗記が必要である。本文では英単語、歴史学習における事件や人名、自然科学における概念や法則のように、暗記しておく必要がある対象を基本用語と呼ぶ。

従来の一般的な単語帳では、基本用語とその説明を一对一の関係で覚える。しかし、それぞれを個別の知識として覚えることになるため、学習効率の面から問題があると考えられる。

本研究では、学習者が覚えたい基本用語の関連付けをすることができる暗記学習支援アプリを提案、試作を行う。

2 提案手法

本研究では、学習者が覚えたい基本用語を任意に追加でき、基本用語を相互に関連付けすることができる暗記学習支援アプリを提案する。

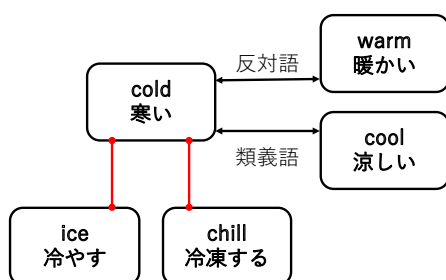


図1 基本用語の関連付け

図1は基本用語（ここでは英単語）間の関連付けの概念図である。学習者は複数の基本用語間に学習の助けになると思われる何らかの関係があった場合、それらを自由に結びつけることができる（図の赤の実線）。また、記憶したい基本用語も追加できる。

従来の暗記方法と比べ、提案手法による学習

は、以下の2点で効果が期待できる。

- 1, 関連性を意識した自発的な学習を促す
- 2, 関連した基本用語を意味のあるグループとして学習でき、記憶の定着に役立つ[1]

3 試作したアプリ

本アプリはiOSアプリとして開発した。また、基本用語の情報、基本用語の関連をJSONを用いて記述した。今回実装した機能は、基本用語の追加、関連用語の追加である。

本アプリは歴史科目の基本用語を用いることを想定して試作した。そのため、追加できる関連は「出来事」と「人物」の2種類である。基本用語の情報を見る画面に、追加した関連用語のリストがあり、関連用語をタップすることで、その用語の情報を画面に表示できる。

4 議論

今回、基本用語の関連付けをどのように行うことができるのか、データとしてどう表現したら良いのかという点について注目してアプリの開発を進めたため、関連付けできる項目の種類が「出来事」と「人物」に限定されている。利用者がより自由に学習を進められるように、現状のデータの表現方法については再考が必要である。また、アプリとしての使い勝手についての検討も必要である。

今後、提案手法の有効性を検証し、従来の暗記方法との違いを明らかにするためには、実用性を高め、ユーザに実際に使ってもらう必要があると考えられる。

5 参考文献

[1] Craik, F.I.M., & Lockhart, R.S. 1972 Levels of processing: A framework for memory research. *Journal of Verbal Learning and Verbal Behavior*, 11, 671- 684

ウェブフレームワークFlaskへのアスペクト指向的な機能の追加に関する試行

学生証番号 054614 情報理工学部 桐山 詩音 (荻原研究室)

1 はじめに

Python言語で作られたウェブフレームワークFlask[1]は、そのシンプルさで注目されているが、横断的関心事(ログ記録、認証など)の管理に課題がある。本研究では、アスペクト指向プログラミングを用いて、Flaskにおける横断的関心事の効率的な管理を実現する方法を提案し、開発の効率化を目指す。

2 提案手法

本研究では、Flaskベースのウェブアプリケーションにおけるカースキーム(ウェブページの色彩配置やテーマ色)の統一的管理を実現するため、アスペクト指向プログラミングを採用する。このアプローチの中核として、ビューコントローラーとカースキームコントローラーを用いる。

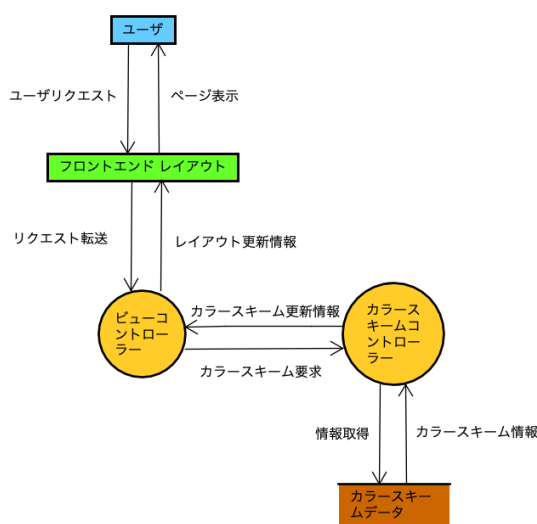


図1 システム構成要素間の関係

図1では、ユーザーの操作からカースキームがウェブページに適用されるまでのプロセスが示している。このプロセスは、ビューコントローラーがユーザーのリクエストを受け取り、カースキームコントローラーにカースキームの選択を依頼することから始ま

る。カースキームコントローラーは、適切なカースキーム情報をビューコントローラーに返し、これがフロントエンドレイアウトに適用され、ユーザーに表示されるページに反映される。

3 実装

本研究では、Flaskで小テストを作成・編集・実施を行えるウェブアプリケーションを作成した。そのアプリケーションにビューコントローラーとカースキームコントローラーの2つの機能を導入した。ビューコントローラーは、アプリケーション全体にわたる一貫したカースキームの管理を簡素化する。各テンプレート(ウェブページの構造やデザインを定義するために使われるファイル)に一貫したカースキームを一律に渡す。カースキームコントローラーは、カースキームに関連する設定やロジックを一箇所に集約する。これらの機能を用いて、アプリケーションにダークモードを実装した。

4 まとめ

本研究では、アスペクト指向プログラミングを用いて、Flaskベースのウェブアプリケーションでカースキームを一貫性のある方法で管理・適応する手法を提案した。このアプローチは、コードの重複を減少させ、保守性と拡張性を向上させた。ただし、Flaskの動的プログラミングのサポートが限られているため、カースキームの動的な切り替えは主にフロントエンドでの実装に依存している。この制約は今後の研究での改善の余地として考えられ、動的なカースキームの切り替えや柔軟な管理手法の開発が求められる。

参考文献

[1] "Flask", (オンライン),
(<https://flask.palletsprojects.com/en/3.0.x/>)
参照 2023-09-26)

Bluetooth 通信を活用した新たな スタンプラリーアプリケーションの提案

学生証番号 054641 情報理工学部 出口楓真（荻原研究室）

1 はじめに

近年、スタンプラリーにもデジタル化が浸透している [1]。しかし、物理的なスタンプや電子スタンプの設置に手間がかかるため、イベントの運営者と参加者が、スタンプラリーを通じてより有益なコミュニケーションのきっかけになることを目的とし、Bluetooth 通信を活用したスタンプラリーアプリケーションの開発を提案する。

2 提案手法

本研究では、Bluetooth通信におけるデータの送受信とデバイス間の距離測定を行うことによりコミュニケーションを促すきっかけになると考えた。イベントの運営者とイベントの参加者が保持しているそれぞれのデバイスを10cm以内に近づけることによりスタンプの送受信が行えるアプリケーションの開発を行う。



図1 通信方法

3 作成したアプリケーション

本研究では、イベントの作成・参加から達成までを実装した。

大学のオープンキャンパスを想定し、運営者がアプリ内でイベントの作成を行い、参加者はイベ

ントのポスター等からアプリをインストールしてスタンプラリーに参加する。

参加者は、運営者に声をかけ、デバイスを10cm以内に近づけてスタンプを集める。全て集めるとスタンプラリーの達成である。

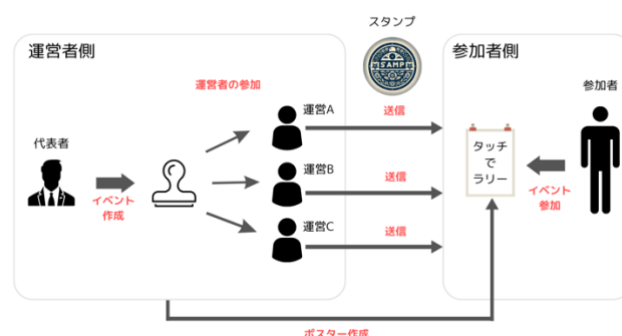


図2 イベントの流れ

4 議論

開発の結果、通信における遅延がなく、スムーズにスタンプデータの送受信が行えた。また、送受信可能な距離範囲の設定により意図しないデータの送受信を防ぐことが可能であった。しかし、不足機能が多く、改良が必要である。

5 まとめ

Bluetooth通信のデータの送受信と距離測定の技術を活用して新しいスタンプラリーアプリケーションの開発は有効であると考えられる。改良は必要だが、追加機能とユーザの意見を反映させることでコミュニケーションが促進する可能性を探る。

参考文献

[1] 日本スタンプラリー協会 「スタンプラリー開催支援サービスのご紹介」

(<https://stamprally.org/services>)