



トランセン課題翻訳

Chapter II

Essential Points

このプロジェクトは複雑な取り組みであり、指定された制約内で意思決定を行う必要があります。一部のモジュールの実装にはある程度の柔軟性が認められており、テーマの範囲内であなたの裁量に委ねられています。ただし、すべての選択は正当化できるものでなければなりません。

ウェブサイトをセットアップする際にnginxを使用する必要があると考えるのであれば、それ自体は問題ありません。ただし、まず自問してください。本当に必要でしょうか？それなしで実現できないでしょうか？同様に、タスクを支援する可能性があるライブラリに直面した際、それが本当に目的を果たすのかを理解することが重要です。興味を引かない下層部分を再構築することを期待されているわけではなく、提案された機能を動作させることが求められています。

特定の機能を実装する際に疑念が生じるような決定に直面することがあることを理解することが重要です。まず最初に、プロジェクトの要件を完全に理解することが

強く推奨されます。何を達成すべきかを把握したら、プロジェクトの枠組み内に留まる必要があります。指定された技術について言及する場合、それは明示的に、要求されたフレームワークや言語に公式に関連するすべてのものが使用可能であることを意味します。

しかし、モジュールを実装しようとする際には、そのモジュールにすべての制約が適用されることを強調します。例えば、指定された内容に従ってバックエンドモジュールを使用してプロジェクトを実現したい場合、デフォルトの言語を使用することはできなくなり、それに応じてプロジェクトを適応させる必要があります。それでもデフォルトの言語を使用してバックエンドを作成したい場合、それも可能ですが、要求された言語やフレームワークを使用していないため、そのモジュールは有効とは見なされません。

結論に至る前に、いくつかのモジュールは意図的に他のモジュールに強く依存していることを指摘することが重要です。

あなたの選択は重要であり、評価の際に正当化されなければなりません。慎重に行動してください。

コードに取り掛かる前に、選択肢を踏まえてアプリケーションの設計をじっくりと考える時間を取りることが重要です。

Have a fun ! :)

Chapter III

Mandatory part

このプロジェクトは、壮大なポンコンテストのためのウェブサイトを作成することに関するものです！

注)

- グローバル機能やモジュールを即座に完全解決するようなライブラリやツールの使用は禁止されています。
- サードパーティライブラリやツールの使用について「使用可能」「必須」「使用不可」などの指示がある場合、それに従う必要があります。
- グローバル機能やモジュールのサブコンポーネントにあたる単純で限定的なタスクを解決する小規模なライブラリやツールの使用は許可されます。
- 評価時には、明確に承認されていないライブラリやツールの使用についてチームがその正当性を説明します。
- 評価時には、評価者がその責任に基づき、特定のライブラリやツールの使用が正当で（許可される）、特定の機能やモジュールをほぼ解決しているか（禁止される）を判断します。

III.1 Overview

あなたのウェブサイトを通じて、ユーザーは他のプレイヤーとPongをプレイできるようになります。ユーザーに快適なユーザーインターフェイスとリアルタイムのマルチプレイヤーオンラインゲームを提供しなければなりません。

- プロジェクトは、以下のガイドラインを最低限の要件として順守する必要があります。これらは最終評価のごく一部に寄与します。
- この課題の第2部では、以下のルールを補完または置き換える追加モジュールが提供されます。
- この課題では、特定の単語が緑色で強調表示されています。これらは、時間とともに進化する技術的選択肢を表しています。課題のバージョンに細心の注意を払ってください。

III.2 Minimal technical requirement

あなたのプロジェクトは、以下のルールに従わなければなりません：

注)

再度申し上げますが、特定のモジュールの選択によって、これらの制約の一部が無効になる場合があります。

- サイト開発の自由度

- バックエンドの有無に関わらず、自由にサイトを開発できます。
- バックエンドを含める場合、ピュアなRubyで記述する必要があります。ただし、この要件は**Frameworkモジュール**によって上書きされる可能性があります。
- バックエンドやフレームワークがデータベースを使用する場合、**Databaseモジュール**の制約に従う必要があります。

- フロントエンド

- ピュアなバニラJavaScriptを使用して開発する必要があります。ただし、この要件は**FrontEndモジュール**によって変更可能です。

- ウェブサイト要件

- ウェブサイトはシングルページアプリケーションでなければなりません。ユーザーはブラウザの「戻る」および「進む」ボタンを使用できる必要があります。
- 最新の安定した**Google Chrome**のバージョンと互換性がなければなりません。
- ユーザーがサイトを閲覧する際、未処理のエラーや警告が表示されてはなりません。

- 実行環境

- すべての機能は、**Docker**が提供する自律的なコンテナで実行される必要があります。これを実現するために、1つのコマンドラインで起動できる必要があります。

例: docker-compose up --build

注)

もしコンテナソリューションとしてDockerを使用する場合、以下の点に注意してください：

- クラスター内のコンピュータがLinuxで動作している場合、セキュリティ上の理由からDockerをルートレスモードで使用する必要があります。これには以下の制約が伴います：
 1. Dockerのランタイムファイルは`*/goinfre`または`/sgoinfre`に配置する必要があります。
 2. コンテナ内で非ルートUIDが使用されている場合、ホストとコンテナ間でいわゆる「バインドマウントボリューム」を使用することはできません。
- プロジェクトや状況、コンテキストに応じて以下のような代替策があります：
 1. Dockerを仮想マシン（VM）内で使用する。
 2. 変更後にコンテナを再構築する。
 3. ルートを唯一のUIDとして設定した独自のDockerイメージを作成する。

これらの制約を理解し、それに応じて適切な方法を選択してください。

III.3 Game

このウェブサイトの主な目的は、他のプレイヤーとPongをプレイすることです。

- **リアルタイムPongゲーム**
 - ユーザーはウェブサイト上で他のプレイヤーとライブでPongゲームに参加できる必要があります。両プレイヤーは同じキーボードを使用します。
 - **Remote playersモジュール**を使用することで、リモートプレイヤーとの対戦機能を強化できます。
- **トーナメント機能**
 - プレイヤー同士が対戦できるだけでなく、トーナメントを提案することも可能にしなければなりません。
 - トーナメントは複数のプレイヤーが順番に対戦する形式で、誰が誰と対戦するのか、またプレイヤーの順番を明確に表示する必要があります。
 - トーナメントの実装には柔軟性がありますが、これらの要件を満たす必要があります。

- **登録システム**

- トーナメント開始時に各プレイヤーがエイリアス名を入力する登録システムが必要です。
- トーナメントが新しく始まるときエイリアスはリセットされます。ただし、**Standard User Management**モジュールを使用することで、この要件を変更することができます。

- **マッチメイキングシステム**

- トーナメントシステムは参加者のマッチメイキングを行い、次の試合を発表する必要があります。

- **ルールの統一**

- すべてのプレイヤーは同じルールに従う必要があります。たとえば、パドルの速度は全員が同一でなければなりません。
- この要件はAIを使用する場合にも適用され、AIは通常のプレイヤーと同じ速度で動作しなければなりません。

- **ゲーム開発**

- ゲームはデフォルトのフロントエンド制約（上記参照）に準拠して開発する必要があります。
- または、**FrontEnd**モジュールや**Graphics**モジュールを使用してこれを上書きすることも可能です。
- 視覚的なデザインは自由ですが、オリジナルのPong（1972）のエッセンスを反映していかなければなりません。

注)

- 即時にグローバルな機能やモジュールを完全に解決するライブラリやツールの使用は禁止されています。
- サードパーティライブラリやツールの使用についての指示（使用可能、必須、使用不可など）は必ず従う必要があります。
- グローバルな機能やモジュールのサブコンポーネントとして単純で特定のタスクを解決する小規模なライブラリやツールの使用は許可されます。
- 評価時には、明確に承認されていないライブラリやツールの使用についてチームがその正当性を説明します。

- 評価者は、その責任のもと、特定のライブラリやツールの使用が正当で許可されるものか、またはグローバルな機能やモジュールをほぼ解決しているため禁止されるものかを判断します。

III.4 Security concerns

基本的な機能を持つウェブサイトを作成するにあたり、以下のセキュリティ上の懸念事項に対応する必要があります：

- パスワードの保存**
 - データベースに保存されるすべてのパスワード（該当する場合）は、ハッシュ化する必要があります。
- SQLインジェクションおよびXSS対策**
 - ウェブサイトはSQLインジェクションやクロスサイトスクリプティング(XSS)攻撃から保護されなければなりません。
- HTTPS接続**
 - バックエンドやその他の機能がある場合、すべての側面でHTTPS接続を有効にすることが必須です（`ws` の代わりに `wss` を使用）。
- フォームとユーザー入力の検証**
 - バックエンドを使用しない場合は基本ページ内で、バックエンドを使用する場合はサーバーサイドで、フォームやユーザー入力に対する何らかの検証を実装する必要があります。
- 全体的なセキュリティ**
 - *JWTセキュリティモジュールや2FA（二要素認証）**を採用するかに関わらず、ウェブサイトのセキュリティを優先することが重要です。
 - 例えば、APIを作成する場合は、ルートが保護されていることを確認してください。
 - JWTトークンを使用しない場合でも、サイトのセキュリティを確保することが必要です。

注)

強力なパスワードハッシュ化アルゴリズムを必ず使用してください。

注)

明らかなセキュリティ上の理由から、認証情報、APIキー、環境変数などはすべて口服一カルの `.env` ファイルに保存し、Gitで無視する設定にする必要があります。公開された状態で認証情報を保存すると、プロジェクトの失敗につながります。

Chapter IV

Modules

プロジェクトの25%を達成しましたね、おめでとうございます！

基本的な機能を持つウェブサイトが完成した今、次のステップはさらなる改善のためにモジュールを選択することです。

- **プロジェクトを100%完了するには、最低でも7つの主要モジュールが必要です。**
- 各モジュールは、ベースラインとなるウェブサイトに変更を加える必要がある場合があります。そのため、モジュールを慎重に確認することが重要です。
- この課題全体を十分に理解するために、内容を注意深く読むことを強くお勧めします。

注)

- グローバルな機能やモジュールを即座に完全解決するライブラリやツールの使用は禁止されています。
- サードパーティのライブラリやツールの使用に関する指示（「使用可能」「必須」「使用不可」など）には必ず従う必要があります。
- グローバル機能やモジュールのサブコンポーネントとして、単純で特定のタスクを解決する小規模なライブラリやツールの使用は許可されます。

- 評価時には、明確に承認されていないライブラリやツールの使用について、チームがその正当性を説明します。
- 評価者はその責任において、特定のライブラリやツールの使用が正当で許可されるものか、またはグローバル機能やモジュールをほぼ解決するため禁止されるものかを判断します。

2つの小規模モジュールは、1つの主要モジュールに相当します。

つまり、

- メイン 7
- メイン 6 + サブ 2
- メイン 5 + サブ 4

IV.1 Overview

Webモジュール

- 主要モジュール:** バックエンドにフレームワークを使用する。
- 小規模モジュール:** フロントエンドフレームワークやツールキットを使用する。
- 小規模モジュール:** バックエンドにデータベースを使用する。
- 主要モジュール:** トーナメントのスコアをロックチェーンに保存する。

User Management

- 主要モジュール:** 標準的なユーザー管理、認証、トーナメント間でのユーザー管理。
- 主要モジュール:** リモート認証の実装。

Gameplay and User Experience

- **主要モジュール:** リモートプレイヤー機能の追加。
- **主要モジュール:** マルチプレイヤー（同じゲーム内で2人以上のプレイヤー）。
- **主要モジュール:** 新しいゲームの追加（ユーザー履歴とマッチメイキング付き）。
- **小規模モジュール:** ゲームのカスタマイズオプション。
- **主要モジュール:** ライブチャット機能。

AI・アルゴリズム

- **主要モジュール:** AI対戦相手の導入。
- **小規模モジュール:** ユーザーとゲームの統計ダッシュボード。

サイバーセキュリティ

- **主要モジュール:** WAF/ModSecurityの導入（強化された設定）とHashiCorp Vaultを使用した秘密情報管理。
- **小規模モジュール:** GDPR準拠オプション（ユーザー匿名化、ローカルデータ管理、アカウント削除を含む）。
- **主要モジュール:** 二要素認証（2FA）およびJWTの実装。

DevOps

- **主要モジュール:** ログ管理のためのインフラ設定。
- **小規模モジュール:** モニタリングシステムの導入。
- **主要モジュール:** バックエンドをマイクロサービスとして設計する。

Graphics

- **主要モジュール:** 高度な3D技術の使用。

Accessibility（アクセシビリティ）

- **小規模モジュール:** すべてのデバイスでのサポート。
- **小規模モジュール:** ブラウザ互換性の拡張。

- **小規模モジュール:** 複数言語対応。
- **小規模モジュール:** 視覚障害者向けアクセシビリティの追加。
- **小規模モジュール:** サーバーサイドレンダリング (SSR) の統合。

Server-Side Pong

- **主要モジュール:** 基本的なPongをサーバーサイドPongに置き換え、APIを実装。
- **主要モジュール:** CLI経由でWebユーザーと対戦できるPongゲームプレイの有効化 (API統合)。

IV.2 Web

これらのモジュールは、Pongゲームに高度なウェブ機能を統合することを可能にします。

主要モジュール

: バックエンドにフレームワークを使用する。

この主要モジュールでは、バックエンド開発に特定のウェブフレームワークを利用する必要があります。そのフレームワークは

Django

です。

注)

このモジュールの制約を使用せずに、デフォルトの言語/フレームワークを使用してバックエンドを作成することも可能です。ただし、このモジュールが有効とみなされるのは、関連する制約を遵守した場合のみです。

小規模モジュール

: フロントエンドフレームワークまたはツールキットを使用する。

このモジュールでは、フロントエンド開発に

Bootstrapツールキット

を使用する必要があります。

注)

このモジュールの制約を使用せずに、デフォルトの言語/フレームワークを使用してフロントエンドを作成することも可能です。ただし、このモジュールが有効とみなされるのは、関連する制約を遵守した場合のみです。

小規模モジュール

: バックエンドにデータベースを使用する - その他の用途。

このモジュールでは、プロジェクト内のすべてのデータベースインスタンスに

PostgreSQL

を使用する必要があります。この選択は、データの一貫性とプロジェクト全体のコンポーネント間の互換性を保証します。また、バックエンドフレームワークモジュールなど、他のモジュールの前提条件となる場合があります。

主要モジュール: トーナメントのスコアをブロックチェーンに保存する。

この主要モジュールは、Pongウェブサイトにトーナメントスコアを安全にブロックチェーンに保存する機能を実装することに焦点を当てています。開発およびテスト目的では、テスト用ブロックチェーン環境を使用することが明確に定義されています。この実装で使用するブロックチェーンは**Ethereum**であり、スマートコントラクトの開発には**Solidity**を使用します。

1. ブロックチェーン統合

- このモジュールの主な目的は、EthereumブロックチェーンをPongウェブサイトにシームレスに統合することです。
- この統合により、トーナメントスコアを安全かつ改ざん不可能な形で保存し、プレイヤーに透明性と信頼性の高い記録を提供します。

2. Solidityスマートコントラクト

- ブロックチェーンとやり取りするために、Solidityを用いたスマートコントラクトを開発します。
- これらのコントラクトは、トーナメントスコアの記録、管理、取得を担います。

3. テスト用ブロックチェーン

- 開発およびテスト目的でテスト用のブロックチェーン環境を使用します。
- これにより、実際のブロックチェーンに関連するリスクを回避しつつ、ブロックチェーン関連の機能を徹底的に検証できます。

4. 相互運用性

- このモジュールは、特にバックエンドフレームワークモジュールなど他のモジュールに依存する可能性があります。
- ブロックチェーン機能を統合するために、バックエンドを調整し、ブロックチェーンとの相互作用をサポートする必要がある場合があります。

このモジュールを実装することで、Pongウェブサイトにブロックチェーンベースのスコア保存システムを導入し、ユーザーにセキュリティと透明性のある機能を提供します。特に、テスト用ブロックチェーン環境を使用することで、ブロックチェーン開発に関連するリスクを最小限に抑えることを重視しています。

IV.3 User Management

このモジュールは、Pongプラットフォームにおけるユーザー管理の領域を掘り下げ、ユーザーの操作やアクセス制御に関する重要な側面に対応します。これには、ユーザー管理と認証の基本要素に焦点を当てた2つの主要な要素が含まれています。それは、複数のトーナメントへのユーザー参加と、リモート認証の実装です。

主要モジュール: 標準的なユーザー管理、認証、トーナメント間でのユーザー管理

- ユーザーは安全な方法でウェブサイトに登録することができます。
- 登録済みのユーザーは安全な方法でログインすることができます。

- ユーザーはトーナメントで使用するユニークな表示名を選択できます。
- ユーザーは自身の情報を更新できます。
- ユーザーはアバターをアップロードでき、何も提供されない場合はデフォルトのオプションが設定されます。
- ユーザーは他のユーザーをフレンドとして追加し、オンラインステータスを確認できます。
- ユーザープロファイルには、勝敗などの統計情報が表示されます。
- 各ユーザーには、1対1のゲーム、日付、関連する詳細を含むマッチ履歴があり、ログイン中のユーザーがアクセスできます。

(注)

注意してください。重複するユーザー名やメールアドレスの管理については、あなたの裁量に委ねられています。この管理方法については、必ずその決定に対する正当な理由を提示する必要があります。

主要モジュール: リモート認証の実装

この主要モジュールでは、以下の認証システムを実装することを目的とします：

OAuth 2.0認証 with 42。 主な機能と目標は以下の通りです：

- 認証システムを統合し、ユーザーが安全にサインインできるようにする。
- セキュアなログインを可能にするために、認証機関から必要な認証情報と権限を取得する。
- ベストプラクティスとセキュリティ基準に準拠した、ユーザーフレンドリーなログインおよび認可フローを実装する。
- ウェブアプリケーションと認証プロバイダー間で認証トークンやユーザー情報を安全に交換できるようにする。

この主要モジュールの目標は、安全で便利な方法でウェブアプリケーションへのアクセスを提供するリモートユーザー認証を実現することです。

(注)

注意してください。重複するユーザー名やメールアドレスの管理については、あなたの裁量に委ねられています。この管理方法については、必ずその決定に対する正

的な理由を提示する必要があります。

IV.4 Gameplay and user experience

これらのモジュールは、プロジェクトの全体的なゲームプレイを向上させることを目的としています。

主要モジュール: リモートプレイヤー

2人の離れたプレイヤーが対戦できる機能を実現します。各プレイヤーは別々のコンピュータに位置し、同じウェブサイトにアクセスして、同じPongゲームをプレイします。

注)

ネットワークの問題、例えば予期しない切断や遅延について考慮してください。可能な限り最良のユーザーエクスペリエンスを提供する必要があります。

主要モジュール: マルチプレイヤー

2人以上のプレイヤーが参加できるゲームを実現します。各プレイヤーにはリアルタイムの操作権限が必要であるため、前述の「リモートプレイヤー」モジュールの利用が強く推奨されます。このモジュールでは、3人、4人、5人、6人など、複数のプレイヤーでどのようにゲームを進めるかを自由に定義することができます。

通常の2人対戦ゲームに加えて、このモジュールでは2人を超える特定のプレイヤー数を選択できます（例: 4人プレイ）。例えば、4人プレイでは正方形のボードを使用し、各プレイヤーが正方形の一辺を担当する形式が考えられます。

主要モジュール: 新しいゲームの追加（ユーザー履歴とマッチメイキング機能付き）

この主要モジュールの目的は、Pongとは異なる新しいゲームを導入し、ユーザー履歴の追跡やマッチメイキング機能を組み込むことです。主な機能と目標は以下の通りです：

- 新しいゲームの開発

- プラットフォームの提供内容を多様化し、ユーザーを楽しませるための新しい魅力的なゲームを開発します。
- **ユーザー履歴の追跡機能の実装**
 - 各ユーザーのゲームプレイ統計を記録し表示するための履歴追跡機能を実装します。
- **マッチメイキングシステムの作成**
 - ユーザーが対戦相手を見つけ、バランスの取れた公平なマッチに参加できるようにするマッチメイキングシステムを構築します。
- **データの安全な管理**
 - ユーザーのゲーム履歴およびマッチメイキングデータを安全に保存し、常に最新の状態に保ちます。
- **パフォーマンスの最適化**
 - 新しいゲームのパフォーマンスと応答性を最適化し、楽しいユーザーベースの体験を提供します。また、バグ修正、新機能の追加、ゲームプレイの改善などのために定期的に更新とメンテナンスを行います。

この主要モジュールの目標は、新しいゲームを導入することでプラットフォームを拡張し、ゲームプレイ履歴を通じてユーザーのエンゲージメントを高め、公平なマッチメイキングを通じて楽しめるゲーム体験を提供することです。

小規模モジュール: ゲームのカスタマイズオプション

この小規模モジュールの目的は、プラットフォーム上で利用可能なすべてのゲームに対してカスタマイズオプションを提供することです。主な機能と目標は以下の通りです：

- **カスタマイズ機能の提供**
 - パワーアップ、攻撃、異なるマップなど、ゲームプレイ体験を向上させるカスタマイズ機能を提供します。
- **シンプルなデフォルト版の選択肢**
 - よりシンプルな体験を好むユーザー向けに、基本機能だけのデフォルト版を選択できるようにします。
- **すべてのゲームに適用可能なカスタマイズオプション**

- プラットフォーム上で提供されるすべてのゲームに対して、カスタマイズオプションを利用可能かつ適用可能にします。
- **ユーザーフレンドリーな設定メニュー**
 - ゲームパラメーターを調整するためのユーザーフレンドリーな設定メニューインターフェイスを実装します。
- **一貫性の維持**
 - すべてのゲームにおいて、カスタマイズ機能に一貫性を持たせ、統一されたユーザーエクスペリエンスを提供します。

このモジュールの目標は、ユーザーがすべてのゲームに対してカスタマイズオプションを利用してゲーム体験を自由に調整できる柔軟性を提供し、同時にシンプルなゲームプレイを好むユーザー向けにデフォルト版も提供することです。

主要モジュール: ライブチャット

このモジュールでは、ユーザー向けのチャット機能を作成する必要があります。主な機能は以下の通りです：

- **ダイレクトメッセージ送信**
 - ユーザーが他のユーザーにダイレクトメッセージを送信できるようにする。
- **ユーザーのロック機能**
 - ユーザーが他のユーザーをロックできるようにし、ロックしたアカウントからのメッセージを表示しないようにする。
- **チャットを通じたゲーム招待**
 - ユーザーがチャットインターフェースを介して他のユーザーをPongゲームに招待できるようにする。
- **トーナメントシステムとの連携**
 - トーナメントシステムが、次のゲームに参加予定のユーザーに通知を送ることができるようにする。
- **プロファイルへのアクセス**
 - ユーザーがチャットインターフェースを通じて他のプレイヤーのプロフィールにアクセスできるようにする。

このモジュールの目標は、ユーザー間のコミュニケーションを強化し、ゲームやトーナメントの体験を向上させることです。

IV.5 AI-Algo

これらのモジュールは、データ駆動型の要素をプロジェクトに導入することを目的としています。主要モジュールでは、ゲームプレイを向上させるためのAI対戦相手を導入し、補助的な小規模モジュールでは、ユーザーやゲームの統計ダッシュボードに焦点を当てています。これにより、ユーザーは自身のゲーム体験を簡潔でありますながら有益に把握できるようになります。

主要モジュール: AI対戦相手の導入

この主要モジュールの目的は、ゲームにAIプレイヤーを組み込むことです。特筆すべきは、このタスクにおいてA*アルゴリズムの使用が禁止されているという点です。主な機能と目標は以下の通りです：

- **挑戦的で魅力的なゲームプレイの提供**
 - ユーザーに対して挑戦的で面白いゲームプレイ体験を提供するAI対戦相手を開発する。
- **人間らしい動作のシミュレーション**
 - AIは人間の行動を模倣する必要があり、キーボード入力をシミュレートする形式で実装します。
 - 制約として、AIはゲームの状況を1秒ごとに1回だけ更新して確認することが許されており、バウンドやその他のアクションを予測する必要があります。
- **AIロジックと意思決定プロセスの実装**
 - AIプレイヤーが賢く戦略的な動きを行えるようにするためのロジックと意思決定プロセスを実装する。
- **代替アルゴリズムと技術の探求**
 - A*アルゴリズムに依存せず、効果的なAIプレイヤーを構築するための他のアルゴリズムや技術を検討する。
- **適応性の確保**
 - AIがさまざまなゲームプレイのシナリオやユーザーの操作に対応できるようにする。

この主要モジュールの目標は、A*アルゴリズムに頼ることなく、興奮と競争心を高めるAI対戦相手を導入し、ゲーム体験を向上させることです。

注)

AIは、**ゲームカスタマイズオプションモジュール**を実装した場合、提供されるパワーアップを活用しなければなりません。

注)

注意：評価の際に、AIがどのように機能するのかを詳細に説明する必要があります。何もしないAIを作成することは禁止されています。AIは時々勝つ能力を持たなければなりません。

IV.6 Cybersecurity

これらのサイバーセキュリティモジュールは、プロジェクトのセキュリティ態勢を強化することを目的としています。主要モジュールでは、**Webアプリケーションファイアウォール (WAF)** と**ModSecurity**の設定、および**HashiCorp Vault**を使用した秘密情報の安全な管理を通じて、堅牢な保護に焦点を当てています。一方、小規模モジュールはこれを補完し、**GDPR準拠**、ユーザーデータの匿名化、アカウント削除、**二要素認証 (2FA)**、および**JSON Web Tokens (JWT)** の導入を通じて、データ保護、プライバシー、および認証セキュリティに対するプロジェクトの取り組みを支えます。

主要モジュール: WAF/ModSecurityの強化された設定とHashiCorp Vaultを使用した秘密情報管理の実装

この主要モジュールの目的は、以下の主要なコンポーネントを実装することでプロジェクトのセキュリティ基盤を強化することです。主な機能と目標は次の通りです：

- **WAF/ModSecurityの構成と展開**
 - ウェブベースの攻撃から保護するために、厳密かつ安全な設定を備えたWebアプリケーションファイアウォール (WAF) およびModSecurityを構成・

展開します。

- **HashiCorp Vaultの統合**

- APIキー、認証情報、環境変数などの機密情報を安全に管理および保存するために、HashiCorp Vaultを統合します。
- これらの秘密情報を適切に暗号化し、隔離された状態で保護します。

この主要モジュールは、WAF/ModSecurityを用いたウェブアプリケーション保護と、HashiCorp Vaultによる秘密情報管理を通じて、安全でセキュアな環境を確保し、プロジェクトのセキュリティ基盤を強化することを目指します。

小規模モジュール: GDPR準拠オプション (ユーザー匿名化、ローカルデータ管理、アカウント削除)

この小規模モジュールの目的は、GDPR（一般データ保護規則）に準拠したオプションを導入し、ユーザーがデータプライバシーの権利行使できるようにすることです。主な機能と目標は以下の通りです：

- **ユーザーデータの匿名化**

- ユーザーが個人データの匿名化をリクエストできる機能を実装し、ユーザーのアイデンティティや機密情報を保護します。

- **ローカルデータ管理ツールの提供**

- ユーザーがシステム内に保存された個人情報を確認、編集、または削除できるツールを提供します。

- **アカウント削除のプロセスを簡略化**

- ユーザーがアカウントと関連データを完全に削除できるリクエストを行える、シンプルでスムーズなプロセスを提供します。これにより、データ保護規則に準拠します。

- **データプライバシーに関する透明性の確保**

- ユーザーが自分のデータプライバシーの権利行使できるように、明確でわかりやすい情報を提供します。これには、容易にアクセス可能なオプションを含めます。

このモジュールは、ユーザーが個人情報を管理し、自分のデータプライバシーの権利行使できる機能を提供することで、ユーザーのプライバシーとデータ保護を強化することを目指します。

GDPRについての注意事項

GDPR（一般データ保護規則）に馴染みがない場合は、特にユーザーデータの管理やプライバシーに関するその原則と影響を理解することが重要です。GDPRは、EU（欧州連合）およびEEA（欧州経済領域）内の個人の個人データとプライバシーを保護することを目的とした規則です。この規則は、組織が個人データをどのように取り扱い、処理すべきかについて厳格なルールとガイドラインを設定しています。

GDPRおよびその要件をよりよく理解するために、[欧州委員会のデータ保護に関する公式ウェブサイト](#)を訪問することをお勧めします。このウェブサイトでは、GDPRの原則、目的、ユーザー権利についての包括的な情報を提供しており、さらに詳細なリソースも用意されています。プロジェクトを進める前に、これらの規則に精通するための時間を確保してください。

主要モジュール: 二要素認証（2FA）およびJWTの実装

この主要モジュールの目的は、二要素認証（2FA）の導入とJSON Web Tokens（JWT）の活用により、セキュリティとユーザー認証を強化することです。主な機能と目標は以下の通りです：

- **二要素認証（2FA）の実装**
 - ユーザーアカウントのセキュリティを強化するため、2FAを導入します。これにより、ユーザーはパスワードに加えて、ワンタイムコードなどの二次的な認証方法を提供する必要があります。
- **JWTの利用**
 - JSON Web Tokens（JWT）を使用して、認証と認可を安全に管理します。これにより、ユーザーセッションやリソースへのアクセスがセキュアに管理されます。
- **ユーザーフレンドリーな2FAの設定プロセス**
 - 2FAを有効にするための簡単な設定プロセスを提供します。SMSコード、認証アプリ、またはメールベースの認証などのオプションを用意します。
- **JWTトークンの安全な発行と検証**
 - 不正アクセスを防ぐため、JWTトークンを安全に発行および検証します。これにより、ユーザーアカウントや機密データの保護を確実にします。

この主要モジュールは、2FAを提供することでユーザーアカウントのセキュリティを強化し、JWTを活用して認証と認可を向上させることを目指します。

IV.7 Devops

これらのモジュールは、プロジェクトのインフラストラクチャとアーキテクチャを強化することに焦点を当てています。主要モジュールでは、**ELK (Elasticsearch、Logstash、Kibana)** を使用した効率的なログ管理のためのインフラ構築、柔軟性とスケーラビリティを実現するバックエンドをマイクロサービスとして設計、および**Prometheus/Grafana**を使用した包括的なシステムモニタリングの実装に取り組みます。

主要モジュール: ELK (Elasticsearch、Logstash、Kibana) を使用したログ管理のためのインフラ構築

この主要モジュールの目的は、ELKスタック (Elasticsearch、Logstash、Kibana) を使用してログ管理と分析のための堅牢なインフラを構築することです。主な機能と目標は以下の通りです：

- **Elasticsearchのデプロイ**
 - ログデータを効率的に保存およびインデックス化し、簡単に検索可能でアクセス可能にします。
- **Logstashの設定**
 - 各種ソースからログデータを収集、処理、変換し、それをElasticsearchに送信します。
- **Kibanaのセットアップ**
 - ログデータを可視化し、ダッシュボードを作成し、ログイベントから洞察を得るためにKibanaをセットアップします。
- **データ保持およびアーカイブポリシーの定義**
 - ログデータの保存を効果的に管理するためのデータ保持とアーカイブポリシーを定義します。
- **セキュリティ対策の実装**
 - ログデータおよびELKスタックの各コンポーネントへのアクセスを保護するためのセキュリティ対策を実装します。

この主要モジュールは、ELKスタックを使用した強力なログ管理および分析システムを構築し、システムの運用やパフォーマンスに関するトラブルシューティング、モニタリング、洞察を効果的に行えるようにすることを目指します。

小規模モジュール: モニタリングシステム

この小規模モジュールの目的は、**Prometheus**と**Grafana**を使用して包括的なモニタリングシステムを構築することです。主な機能と目標は以下の通りです：

- **Prometheusのデプロイ**
 - 各種システムコンポーネントのメトリクスを収集し、健康状態やパフォーマンスを監視するためのモニタリングおよびアラートツールキットとして Prometheusを導入します。
- **データエクスポートーと統合の設定**
 - 各種サービス、データベース、インフラコンポーネントからメトリクスを取得するためのデータエクスポートーおよび統合を設定します。
- **Grafanaでのカスタムダッシュボード作成**
 - Grafanaを使用して、システムメトリクスとパフォーマンスに関するリアルタイムの洞察を提供するカスタムダッシュボードやビジュアライゼーションを作成します。
- **Prometheusでのアラートルール設定**
 - 重大な問題や異常を積極的に検出して対応できるよう、Prometheusにアラートルールを設定します。
- **データ保持および保存戦略の策定**
 - 過去のメトリクスデータの保存に適したデータ保持とストレージ戦略を策定します。
- **Grafanaの認証とアクセス制御の実装**
 - 敏感なモニタリングデータを保護するため、Grafanaの安全な認証およびアクセス制御メカニズムを実装します。

このモジュールは、PrometheusとGrafanaを使用して強力なモニタリングインフラを構築し、システムメトリクスのリアルタイム可視化と問題検出を可能にすることで、システムのパフォーマンスと信頼性を向上させることを目指します。

主要モジュール: バックエンドをマイクロサービスとして設計する

この主要モジュールの目的は、システムのバックエンドをマイクロサービスアプローチを使用して設計することです。主な機能と目標は以下の通りです：

- **バックエンドの分割**
 - バックエンドを、特定の機能や特徴を担当する小規模で疎結合なマイクロサービスに分割します。
- **明確な境界とインターフェースの定義**
 - マイクロサービス間で独立した開発、デプロイ、およびスケーリングが可能になるように、明確な境界とインターフェースを定義します。
- **マイクロサービス間の通信機構の実装**
 - データの交換や調整を促進するために、RESTful APIやメッセージキューなどの通信機構を実装します。
- **単一責任の原則の徹底**
 - 各マイクロサービスが単一の明確なタスクまたはビジネス機能を担当するようにし、保守性とスケーラビリティを向上させます。

このモジュールは、マイクロサービス設計アプローチを採用することで、バックエンドコンポーネントの柔軟性、スケーラビリティ、保守性を向上させ、システムのアーキテクチャを強化することを目指します。

IV.8 Gaming

これらのモジュールは、プロジェクトのゲーミフィケーション要素を強化することを目的としています。主要モジュールでは、新しいゲームの導入、ユーザー履歴の追跡、マッチメイキングを提供し、小規模モジュールでは、利用可能なすべてのゲームにカスタマイズオプションを提供することに焦点を当てています。

主要モジュール: 新しいゲームの追加（ユーザー履歴とマッチメイキング機能付き）

この主要モジュールの目的は、Pongとは異なる新しいゲームを導入し、ユーザー履歴の追跡やマッチメイキング機能を組み込むことです。主な機能と目標は以下の通りです：

- **新しい魅力的なゲームの開発**
 - プラットフォームの提供内容を多様化し、ユーザーを楽しませるための新しい魅力的なゲームを開発します。
- **ユーザー履歴の追跡機能の実装**
 - 個々のユーザーのゲームプレイ統計を記録し表示するための履歴追跡機能を実装します。
- **マッチメイキングシステムの構築**
 - ユーザーが対戦相手を見つけ、公平でバランスの取れたマッチに参加できるようにするマッチメイキングシステムを構築します。
- **データの安全な管理**
 - ユーザーのゲーム履歴およびマッチメイキングデータを安全に保存し、常に最新の状態に保ちます。
- **パフォーマンスと応答性の最適化**
 - 新しいゲームのパフォーマンスと応答性を最適化し、楽しいユーザーエクスペリエンスを提供します。また、バグ修正、新機能の追加、ゲームプレイの改善などのために定期的に更新とメンテナンスを行います。

この主要モジュールは、新しいゲームを導入することでプラットフォームを拡張し、ゲームプレイ履歴を通じてユーザーのエンゲージメントを高め、公平なマッチメイキングを通じて楽しめるゲーム体験を提供することを目指します。

小規模モジュール: ゲームのカスタマイズオプション

この小規模モジュールの目的は、プラットフォーム上で利用可能なすべてのゲームに対してカスタマイズオプションを提供することです。主な機能と目標は以下の通りです：

- **カスタマイズ機能の提供**
 - ゲームプレイ体験を向上させるため、パワーアップ、攻撃、異なるマップなどのカスタマイズ機能を提供します。
- **デフォルト版の選択肢**
 - よりシンプルな体験を好むユーザー向けに、基本機能のみを持つゲームのデフォルト版を選択できるようにします。
- **すべてのゲームに適用可能なカスタマイズオプション**

- プラットフォーム上で提供されるすべてのゲームに対して、カスタマイズオプションを利用可能かつ適用可能にします。
- **ユーザーフレンドリーな設定メニューの実装**
 - ゲームパラメーターを調整するための、使いやすい設定メニューインターフェースを実装します。
- **一貫性のあるカスタマイズ機能**
 - すべてのゲームにおいてカスタマイズ機能に一貫性を持たせ、統一されたユーザー体験を提供します。

このモジュールの目的は、ユーザーがプラットフォーム上のすべてのゲームに対して多様なカスタマイズオプションを利用してゲーム体験を自由に調整できる柔軟性を提供し、同時にシンプルなゲームプレイを好むユーザー向けにデフォルト版も提供することです。

IV.9 Graphics

主要モジュール: 高度な3D技術の実装

この「グラフィックス」モジュールは、Pongゲームの視覚的な要素を向上させることに焦点を当てています。高度な3D技術を活用することで、より没入感のあるゲーム体験を提供します。具体的には、ThreeJS/WebGLを使用してPongゲームを開発し、期待される視覚効果を実現します。

主な機能と目標

- **高度な3Dグラフィックス**
 - 本モジュールの主な目標は、高度な3Dグラフィックス技術を実装し、Pongゲームの視覚品質を向上させることです。ThreeJS/WebGLを活用して、プレイヤーをゲーム環境に没入させる美しい視覚効果を作り出します。
- **没入型ゲームプレイ**
 - 高度な3D技術の導入により、視覚的に魅力的で引き込まれるPongゲームを提供し、ゲームプレイ体験を全体的に向上させます。
- **技術統合**

- このモジュールで選ばれた技術はThreeJS/WebGLです。これらのツールを使用して3Dグラフィックスを作成し、互換性と最適なパフォーマンスを確保します。

モジュールの目的

この主要モジュールは、高度な3D技術を導入することでPongゲームの視覚要素を革新します。ThreeJS/WebGLを活用することで、プレイヤーに没入感があり、視覚的に魅力的なゲーム体験を提供することを目指します。

IV.10 Accessibility

これらのモジュールは、ウェブアプリケーションのアクセシビリティを向上させることを目的としています。具体的には、すべてのデバイスとの互換性を確保し、ブラウザ対応を拡大し、多言語対応を提供し、視覚障害者向けのアクセシビリティ機能を提供し、サーバーサイドレンダリング (SSR) を統合してパフォーマンスとユーザー体験を改善することに焦点を当てています。

小規模モジュール: すべてのデバイスでのサポート

このモジュールの主な目的は、ウェブサイトがすべての種類のデバイスでシームレスに動作するようにすることです。主な機能と目標は以下の通りです：

- レスポンシブデザインの確保
 - ウェブサイトが異なる画面サイズや向きに適応し、デスクトップ、ラップトップ、タブレット、スマートフォンで一貫したユーザー体験を提供できるようになります。
- 多様な入力方法への対応
 - ユーザーがデバイスに応じてタッチスクリーン、キーボード、マウスなどの異なる入力方法を使ってウェブサイトを簡単にナビゲートし、操作できるようになります。

このモジュールは、すべてのデバイスで一貫性があり使いやすい体験を提供することで、アクセシビリティとユーザー満足度を最大化することを目指します。

小規模モジュール: ブラウザ互換性の拡張

このモジュールの目的は、ウェブアプリケーションの互換性を向上させ、追加のウェブブラウザをサポートすることです。主な機能と目標は以下の通りです：

- **追加のブラウザサポート**
 - ウェブアプリケーションが新たなウェブブラウザでもシームレスにアクセス・利用できるように、対応範囲を拡張します。
- **徹底的なテストと最適化**
 - 新たにサポートされるブラウザでウェブアプリケーションが正しく動作し、正しく表示されることを確認するため、徹底的なテストと最適化を行います。
- **互換性の問題解決**
 - 追加されたブラウザで発生する可能性のある互換性の問題やレンダリングの不整合を解消します。
- **一貫したユーザーエクスペリエンスの提供**
 - サポートされるすべてのブラウザで、一貫したユーザーエクスペリエンスを維持し、使いやすさと機能性を確保します。

このモジュールは、追加のウェブブラウザをサポートすることでウェブアプリケーションのアクセシビリティを広げ、ユーザーにより多くの選択肢を提供することを目指します。

小規模モジュール: 多言語対応

このモジュールの目的は、ウェブサイトが多言語に対応し、多様なユーザー層に対応することです。主な機能と目標は以下の通りです：

- **最低3言語のサポートを実装**
 - 幅広いユーザー層に対応するため、ウェブサイトで最低3つの言語をサポートします。
- **言語スイッチャーの提供**
 - ユーザーが簡単にウェブサイトの言語を選択できるように、言語スイッチャー・セレクターを提供します。
- **ウェブサイトの主要コンテンツの翻訳**

- ナビゲーションメニュー、見出し、重要な情報など、ウェブサイトの主要なコンテンツをサポートされる言語に翻訳します。
- **シームレスな操作体験の確保**
 - 選択した言語にかかわらず、ユーザーがウェブサイトをスムーズにナビゲートし、操作できるようにします。
- **言語パックやローカリゼーションライブラリの利用**
 - 翻訳プロセスを簡略化し、異なる言語間で一貫性を保つために、言語パックやローカリゼーションライブラリを検討します。
- **デフォルト言語の設定**
 - ユーザーが次回以降の訪問時に使用する言語をデフォルトとして設定できるようにします。

このモジュールは、多言語でコンテンツを提供することでウェブサイトのアクセシビリティと包括性を高め、多様な国際的なユーザー層にとって使いやすいものにすることを目指します。

小規模モジュール: 視覚障害者向けアクセシビリティの追加

このモジュールの目的は、視覚障害を持つユーザーがウェブサイトをより利用しやすくすることです。主な機能は以下の通りです：

- **スクリーンリーダーや支援技術への対応**
 - スクリーンリーダーやその他の支援技術をサポートします。
- **画像のための明確で説明的な代替テキスト (altテキスト)**
 - 画像に対して、明確でわかりやすい代替テキストを提供します。
- **高コントラストのカラースキーム**
 - 読みやすさを向上させるために、高コントラストのカラースキームを用意します。
- **キーボードナビゲーションとフォーカスマネジメント**
 - キーボードだけでウェブサイトを操作できるようにし、フォーカスの管理を適切に行います。
- **テキストサイズの調整オプション**
 - ユーザーがテキストサイズを調整できるオプションを提供します。

- アクセシビリティ基準に準拠するための定期的な更新

- アクセシビリティ基準を満たすよう、ウェブサイトを定期的に更新します。

このモジュールは、視覚障害者にとってウェブサイトの操作性を向上させるとともに、アクセシビリティ基準に準拠することを目指します。

小規模モジュール: サーバーサイドレンダリング (SSR) の統合

このモジュールは、ウェブサイトのパフォーマンスとユーザーエクスペリエンスを向上させるために、サーバーサイドレンダリング (SSR) を統合することに焦点を当てています。主な目標は以下の通りです：

- ウェブサイトの読み込み速度と全体的なパフォーマンスの向上

- SSRを実装して、ウェブサイトの読み込み速度を改善し、全体的なパフォーマンスを向上させます。

- コンテンツのサーバー側での事前レンダリング

- コンテンツをサーバー側で事前にレンダリングし、ユーザーのブラウザに届けることで、初回のページ読み込みを高速化します。

- SEOの最適化

- 検索エンジンに事前レンダリングされたHTMLコンテンツを提供することで、SEO（検索エンジン最適化）を強化します。

- 一貫したユーザーエクスペリエンスの維持

- SSRの利点を享受しながら、一貫したユーザーエクスペリエンスを維持します。

このモジュールは、サーバーサイドレンダリングを統合することで、ウェブサイトのパフォーマンスやSEOを向上させ、ページ読み込みを高速化し、より良いユーザーエクスペリエンスを提供することを目指します。

IV.11 Server-Side Pong

主要モジュール: 基本的なPongをサーバーサイドPongに置き換え、APIを実装

この主要モジュールの目的は、基本的なPongゲームをサーバーサイドのPongゲームに置き換え、APIを実装することです。主な機能と目標は以下の通りです：

- サーバーサイドロジックの開発
 - Pongゲームのゲームプレイ、ボールの動き、スコアリング、プレイヤー間のインタラクションを処理するためのサーバーサイドロジックを開発します。
- APIの作成
 - コマンドラインインターフェース（CLI）およびウェブインターフェースを介してゲームに部分的にアクセスできるようにするために、必要なリソースとエンドポイントを公開するAPIを作成します。
- APIエンドポイントの設計と実装
 - ゲームの初期化、プレイヤーの操作、ゲーム状態の更新をサポートするためのAPIエンドポイントを設計および実装します。
- レスポンシブなサーバーサイドPongゲームの確保
 - サーバーサイドPongゲームが応答性を保ち、魅力的で楽しいゲーム体験を提供することを保証します。
- ウェブアプリケーションとの統合
 - サーバーサイドPongゲームをウェブアプリケーションに統合し、ユーザーがウェブサイト上で直接ゲームをプレイできるようにします。

このモジュールは、Pongゲームをサーバーサイドに移行することで、ウェブインターフェースおよびCLIを通じたインタラクションを可能にし、ゲームリソースや機能への簡単なアクセスを提供するAPIを構築することを目指します。

主要モジュール: API統合を通じてCLIでWebユーザーと対戦可能なPongゲームプレイを実現

この主要モジュールの目的は、コマンドラインインターフェース（CLI）を開発し、CLIユーザーがゲームのWebバージョンを使用するプレイヤーとPongを対戦できるようにすることです。CLIはWebアプリケーションとシームレスに接続され、CLIユーザーがWebプレイヤーと対戦し、やり取りできるようにします。主な機能と目標は以下の通りです：

- 堅牢なCLIアプリケーションの作成
 - CLIユーザーがPongマッチを開始し、参加できるように、ウェブサイトで提供されるPongゲームプレイ体験を再現する堅牢なCLIアプリケーションを作成します。

- **APIを利用した通信の確立**
 - CLIとWebアプリケーション間で通信を確立するためにAPIを利用し、CLIユーザーがサイトに接続し、Webプレイヤーとやり取りできるようにします。
- **CLI内のユーザー認証機構の開発**
 - CLIユーザーがWebアプリケーションに安全にログインできる認証機構を開発します。
- **リアルタイム同期の実装**
 - CLIとWebユーザー間のゲームプレイインタラクションがシームレスで一貫性のあるものになるよう、リアルタイムの同期を実装します。
- **CLIユーザーの参加とマッチ作成を可能にする**
 - CLIユーザーがWebプレイヤーとPongマッチに参加したり、マッチを作成したりできるようにし、クロスプラットフォームでのゲームプレイを促進します。
- **包括的なドキュメントとガイドの提供**
 - CLIを効果的に使用してWebユーザーとPongマッチを行う方法に関する包括的なドキュメントとガイドを提供します。

このモジュールは、CLIユーザーとWebプレイヤーをAPI統合を通じてシームレスに接続し、統一されたインタラクティブなゲームプレイ環境を提供することで、Pongゲームの体験を向上させることを目指します。

注)

このモジュールを実行したい場合は、前のモジュールを実行することを強くお勧めします。