

生成AIを使用して どのくらいシステム開発が できるのか？

生成AI部門 Aチーム(横山 ひより、佐藤 永遠、是永 陸)

目次

1.研究目的・内容

2.生成AIとは

3.ChatGPTとは

4.生成AIを使用した各自の成果物発表

5.総評 -生成AIを使用してどのくらいシステム開発ができるのか？

研究目的・内容

目的：生成AIがどの程度開発に使用できるかや
開発効率を高められるのかを調査

調査内容：生成AI(ChatGPT)を利用して
各個人で実際にアプリ開発を行ってみる

ChatGPTを選定した理由としては
無料で使用できるので企業としても導入しやすく
また現在世界中で注目されている生成AIであるため



生成AIとは？

生成AIは、人工知能の一種であり、機械学習モデルを使用して、人間のようクリエイティブなコンテンツを生成することができる



生成AI (Generative Artificial Intelligence) とは

生成AIは、与えられたデータから新しいデータやコンテンツを生成する能力がある。

テキスト、画像、音声、ビデオなどのデータ形式に基づいて新しい情報やメディアを生成することが可能。

生成AIの一例としては、OpenAIのChat GPTが挙げられる。

Chat GPTとは？

自然な対話

人間のような対話を行うAIモデルで、個人のニーズに合わせた自然な対話応答が可能

文脈を理解

前の文脈を理解する能力があり、より深い対話が可能

多様な応答

一意な応答を生成し、面白く詳細な情報を提供する

生成AIの応用例

このPowerPointも
GAMMAという生成AIを
利用しています



アート生成

美術作品や写真、ピクセル画
などのアートを自動的に生成



文章生成

記事やブログ投稿の自動生成
に使用可能



映画スクリプト

映画の台本やシナリオの作成
をサポート



音楽作成

音楽の作曲やアレンジを行う
ことが可能

是永の成果物

Chat GPTを活用しタスク管理アプリ作成

TS TeamS

OISA

216. 発表資料作成

詳細

作成方法

- 生成AIを使用

| | | | | | |
|------|------------|-------|------------|------|------|
| 担当者 | 是永 陸 | ステータス | 未処理 | | |
| 開始日 | 2023-11-29 | 終了日 | 2023-12-07 | | |
| 優先度 | ↑ 大 | バージョン | | カテゴリ | 資料作成 |
| 予定時間 | 8 | 実績時間 | 0 | 進捗度 | 0 |

Chat GPTの活用方法

- ・ 使用言語：HTML、CSS、JavaScript(Vue3)
- ・ プログラミング工程で使用。
- ・ Chat GPTに指示し、コードを生成

※ChatGPTが生成したコードに不備がある場合は、修正



生成AIを開発に使用した メリット

- ChatGPTで生成されたコードをひな形として活用すると効率的に感じた。
- 分かりやすい説明サイトがないマニアックな内容に答えてくれることがある。
- ソースコードやエラーメッセージを使用して指示が出せる。
- 規模を小さく、詳細な指示ほど意図したコードが生成されやすかった。

生成AIを開発に使用したデメリット

- ・ 指示した通りにコードが生成されない時がある。
(生成されたコードに不備がある)
- ・ 簡素なデザインでコードが生成される。
- ・ 使用しているバージョンに対応していないコードが生成される時がある。
- ・ システムの規模が大きくなるほど、既存のメソッドや変数を考慮し上のコード生成ができなくなる。



横山の成果物

制作物 概要

▶ スケジュールアプリ

使用言語：HTML、CSS、JavaScript

生成AIを使用した工程：要件定義、設計、開発

ログアウトしました

ご利用ありがとうございました。

[もう一度ログインする](#)

生成AIで生成された要件定義

Hi

You

HTML5とJavaScriptを使用してスケジュール管理アプリを作成する場合の要件定義を作成してください。
なおそのアプリを使用する人数は5人とします。



ChatGPT

もちろんです。以下に、HTML5とJavaScriptを使用して作成するスケジュール管理アプリの要件定義の一例を示します。これはあくまで基本的な要件の一例であり、実際のアプリに合わせてカスタマイズすることができます。

スケジュール管理アプリの要件定義:

1. ユーザー認証とアカウント管理:

- ユーザーは個別のアカウントを作成できる。
- ユーザーはログインしてアプリにアクセスできる。

2. カレンダービュー:

- ユーザーはカレンダービューで日付を選択してスケジュールを表示できる。
- カレンダーは月表示と週表示の切り替えができる。

3. スケジュール管理:

- ユーザーはスケジュールを追加、編集、削除できる。
- スケジュールはタイトル、開始日時、終了日時、詳細などの情報を持つ。
- スケジュールの色分けやカテゴリ分類ができる。

4. 通知機能:

- ユーザーに予定されたスケジュールの通知を送信できる（ブラウザ通知などを利用）。

5. 共有機能:

- ユーザーは特定のスケジュールを他のユーザーと共有できる。
- 共有されたスケジュールは閲覧のみ可能。

6. データ保存と同期:

- ユーザーのスケジュールデータはブラウザのローカルストレージやクラウドサービスを使用して保存される。
- 複数のデバイス間でスケジュールが同期される。

7. ユーザビリティ:

- インターフェースは使いやすく、直感的である。
- レスポンシブデザインを採用して、異なるデバイスでの利用が可能。

8. セキュリティ:

- ユーザーの個人情報やスケジュールデータは適切なセキュリティ対策で保護される。

9. プライバシー:

- ユーザーは個別のスケジュールを非公開設定にできる。

10. ユーザー数とパフォーマンス:

- アプリは同時に5人のユーザーが利用することを想定して設計される。
- ユーザー数の増加にも対応できるように設計する。

これらの要件を基に、アプリの開発を進める際には、各要件に対する詳細な実装計画やデザインを検討する必要があります。また、開発中にユーザーフィードバックを収集し、アプリを改善するプロセスを取り入れることも大切です。

生成AIで生成された設計

Hi **You**
そのスケジュールアプリの設計を行ってください。



ネットワークなども含まれた大規模な設計が定時されたため、指示を修正

Hi **You**
もっと簡易的なスケジュール管理アプリの場合に必要な画面数とUI構成を考えてください。



ChatGPT

もちろんです。簡易的なスケジュール管理アプリの場合、以下の画面とUI構成が考えられます。

1. **カレンダービュー:**

- カレンダーを表示するための画面。
- 月ごとのカレンダービューと週ごとのカレンダービューの切り替えボタン。
- 日付を選択してその日のスケジュールを表示するカレンダーグリッド。

2. **スケジュール詳細画面:**

- 選択された日付のスケジュール一覧を表示する画面。
- 各スケジュールにはタイトル、開始時間、終了時間が表示される。
- 新規スケジュールの追加ボタン。

3. **スケジュール追加/編集画面:**

- タイトル、日付、開始時間、終了時間の入力フィールドを持つフォーム。
- 保存ボタンでスケジュールを追加または編集。

4. **通知設定画面:**

- 通知の設定やオプションを調整する画面。通知の有効化や時間の設定などが含まれる。

5. **ログアウト画面:**

- ログアウトボタンを含む画面。ユーザーをログアウトさせるためのアクション。

UI構成はシンプルかつ使いやすさを重視します。カレンダービューでは日付を選択してその日のスケジュールを簡単に表示できるようにすることが大切です。スケジュール詳細画面では、各スケジュールの情報を一覧できるようにし、新しいスケジュールを追加するボタンを用意します。スケジュール追加/編集画面は、ユーザーが新しいスケジュールを簡単に入力できるようにするために直感的なフォームを提供します。通知設定画面は、ユーザーが通知をカスタマイズできるようにするための設定項目を提供します。

生成AIで生成された画面

- ・シンプルながらもモダンなUI。
ただしログアウト画面はかなり質素なUIが生成されるなど、生成AIの性質上あくまでよく利用されている画面が出てくる模様。
- ・見た目上はあまり問題がなさそうに見えるが、予定の追加の保存ボタン等が全く機能しない状態。

スケジュールアプリ

2023年8月

| | | | | | | |
|----|----|----|----|----|----|----|
| | | 1 | 2 | 3 | 4 | 5 |
| 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | 29 | 30 | 31 | | |

[前の月](#) [次の月](#)

8月15日から21日の週

予定の追加

予定タイトル: 予定時間:

生成AIを開発に使用したメリット

1 素早い開発

数分でアプリのフレームワークを生成し、初期セットアップにかかる時間を短縮できる。

2 創造的なアイデア

自分の考えを超えたデザインやコードの提案をしてもらえる。

3 簡単な指示でも生成可能

曖昧な指示でも適切なコードを生成することができる。

4 コードのインサイト

生成されたコードの解説や疑問点への回答を簡単に得ることができる。

生成AIを開発に使用したデメリット

1 スキルアップに課題

複雑なコードを生成することもできるが、それを理解しないまま使用していると自身の成長に繋がらない。

2 一貫性のないデザイン

ページ遷移後に生成されるデザインが遷移前の画面と異なることがあり、そうした場合は追加の調整が必要。

3 乱雑なコード

自動生成される変数や関数の名前が適当に生成されることがあり、生成回数が多くなるほど乱雑なコードになる可能性がある。

4 仕様とのズレ

詳細な仕様設計が前もって用意されている場合、ChatGPTの出力が期待と一致しないことがある。

佐藤の成果物

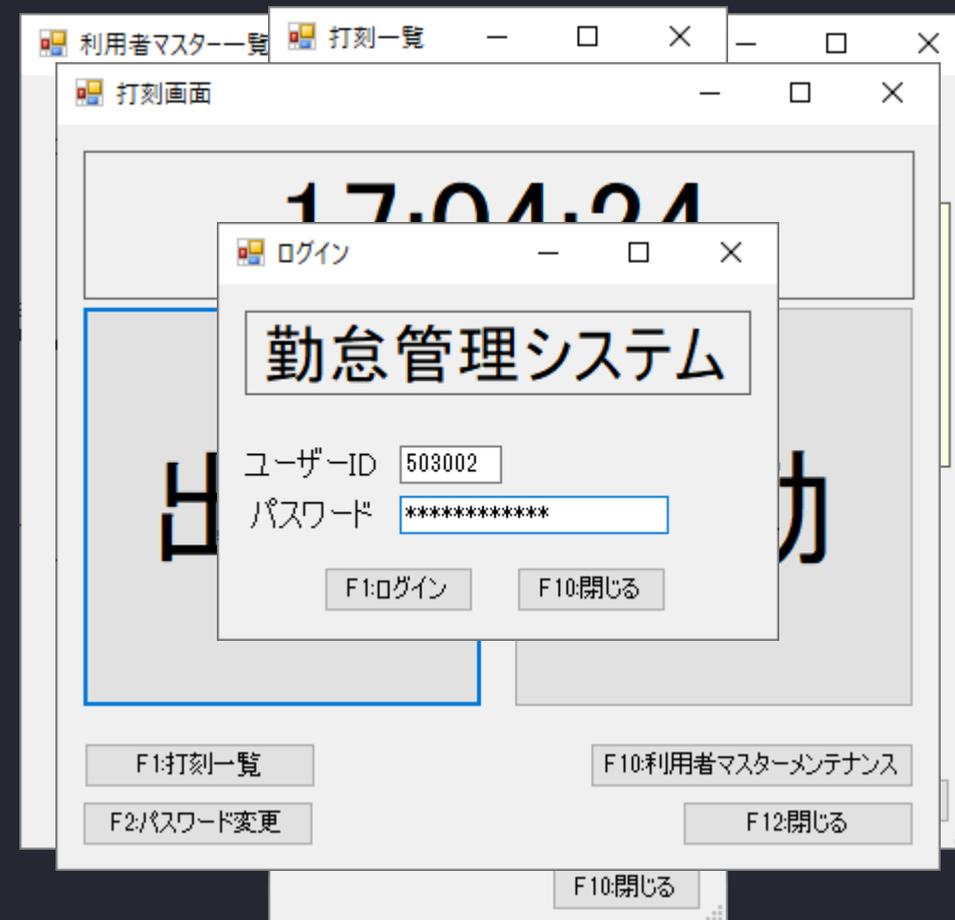
制作物 概要

成果物：勤怠管理システム 使用言語：C#

生成AIを使用した工程：PG, テスト

画面構成

1. ログイン画面
2. 打刻画面
3. 打刻一覧画面
4. パスワード変更画面
5. 利用者マスター一覧画面(管理者権限のあるユーザーのみアクセス可能)
6. 利用者マスターメンテナンス画面(5の画面からのみアクセス可能)



生成AIを開発に使用したメリット

- 比較的狭い範囲での指示だと的確なソースコードが生成される。
- 生成したコードがエラーを含む場合でも、指摘すれば修正してくれる。
- 適切に指示が出せればSQLの構文を忘れても生成してくれる。
- 機能概要を説明するだけで、必要なテストケースの生成が可能。

生成AIを開発に使用したデメリット

- エラー修正の過程で正常だった箇所に影響を及ぼすことがあった。
- 同一内容の指示の中でも回答によってはコントロール名が変わることがある。
- 画面間の値の受け渡しなど、1つの指示で完結しない機能は実装が厳しい。
- カスタムコントロールなど標準以外の機能を使用しづらい。

総評

生成AIを使用して どのくらいシステム開発ができるのか？

使用する範囲さえ見極めれば
開発の効率化に繋がる

- ・簡易的な要件定義や設計書も生成できるが
詳細設計は人間が行った方がよい。
- ・全てを生成AIに任せようとするとう手間がかかるが、
狭い範囲(関数単位)であれば十分に利用可能。