

# ChatGPTを利用した Webアプリケーションの開発

2023年7月8日

千葉工業大学

小笠原秀人 ([hideto.ogasawara@p.chibakoudai.jp](mailto:hideto.ogasawara@p.chibakoudai.jp))

# 目次

1. Generative AI の現状
2. ソフトウェア・シンポジウム2023ではどうだったか？
3. 授業（演習）での活用
4. 自分で試してみた
5. 感想
6. これからの進展（予想）

# 1. Generative AI の現状

- 新しい言語モデル
  - 昨年末に OpenAI が ChatGPT を発表、文脈を意識した高度な対話を可能にした。無料で提供されていることもあり一気にブレイク
  - API (ChatGPT API) も発表されて、さまざまなアプリケーションへの組み込みが始まった
- テック企業各社はそれぞれの取り組みを行っている
  - Microsoft：検索エンジン Bing への OpenAI の GPTの取り込み
  - Google：独自の言語モデル LaMDA (PaLM2) を組み込んだチャット Bard
  - DeepMind：チャットAI の Sparrow を開発中であると報じられる
  - Facebook：言語モデル LLaMA を研究者向けに提供

# Generative AI を使ったさまざまなツール例

- ChatGPT

- 単純にインタラクティブに使う以外に、ツールに組み込んで使うこともできる。Generative AIブームの立役者。
- これまで2021年9月までの知識しか使えなかったが、新しいプラグイン機能によって最新知識にもアクセスできるようになった

- Bing Chat

- Microsoft が Edge ブラウザに組み込んだ Generative AI チャット。ChatGPT相当の能力にリアルタイムネット検索機能を持っている

- Bard

- Google が作成したチャットボット、動作は軽快。ChatGPTとほぼ同様の能力を持つ予定だが、公開されてまだ日が浅いためかいろいろと不都合がみうけられる

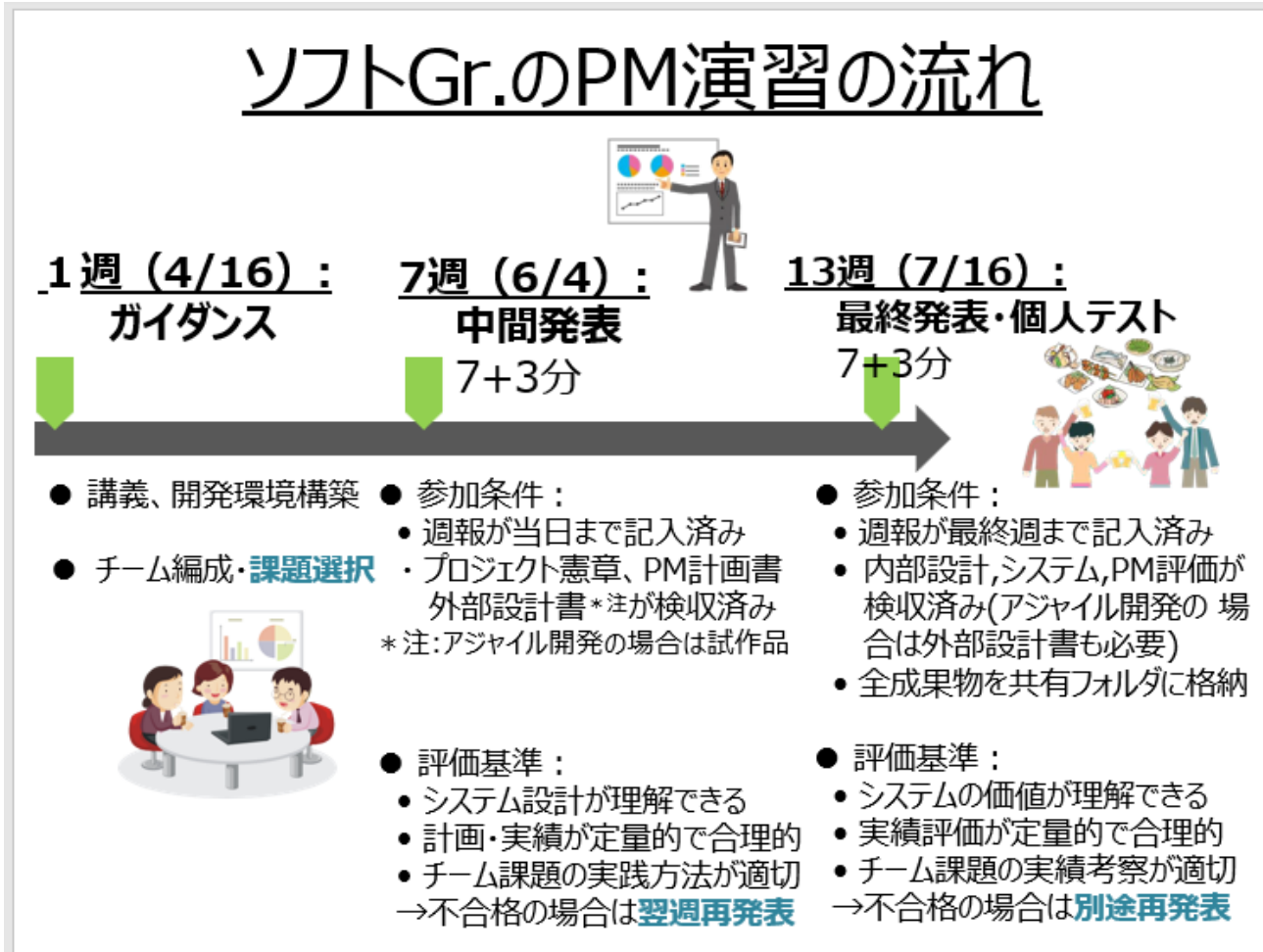
## 2. SS2023ではどうだったか？

- テストプロセスにおける「ChatGPT」の活用の提案
  - ソフトウェア開発におけるテストのプロセスに「ChatGPT」が活用できるかどうかを検討した内容
- Generative AI とソフトウェア開発への応用の試行
  - ソフトウェア開発にどのように役立つのかを事例を用いて紹介。さらに、利用する際のポイントを整理し、これからの進化について予測。
  - 最優秀発表賞を受賞。
- GPT-3とLlamaIndexを用いた生成知識の比較
  - LlamaIndexはChatGPTを使用して独自のデータに対して質問できるようにすることができるフレームワーク。テキストやHTML、PDFなどを入力してインデックスファイルを作り、そのインデックスファイルに対してクエリを投げることで、ChatGPTが学習していない最新の情報に対して質問することが可能。

<https://www.sea.jp/ss2023/programme.php>

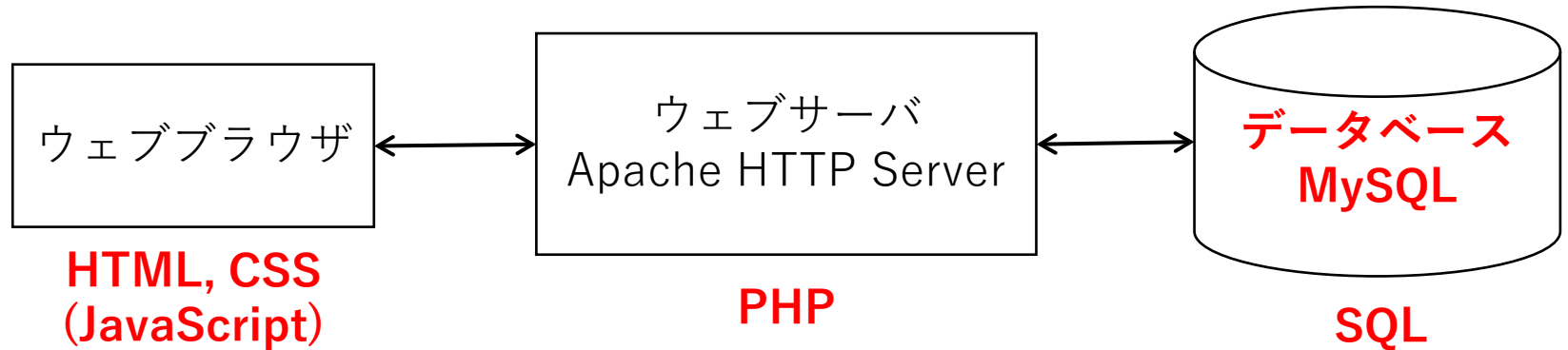
# 3. 授業（演習）での活用

- プロジェクトマネジメント演習（13週間）
  - 研究室に配属された3年生の前期に実施
  - 3～4人でグループを作りWebアプリケーションを開発

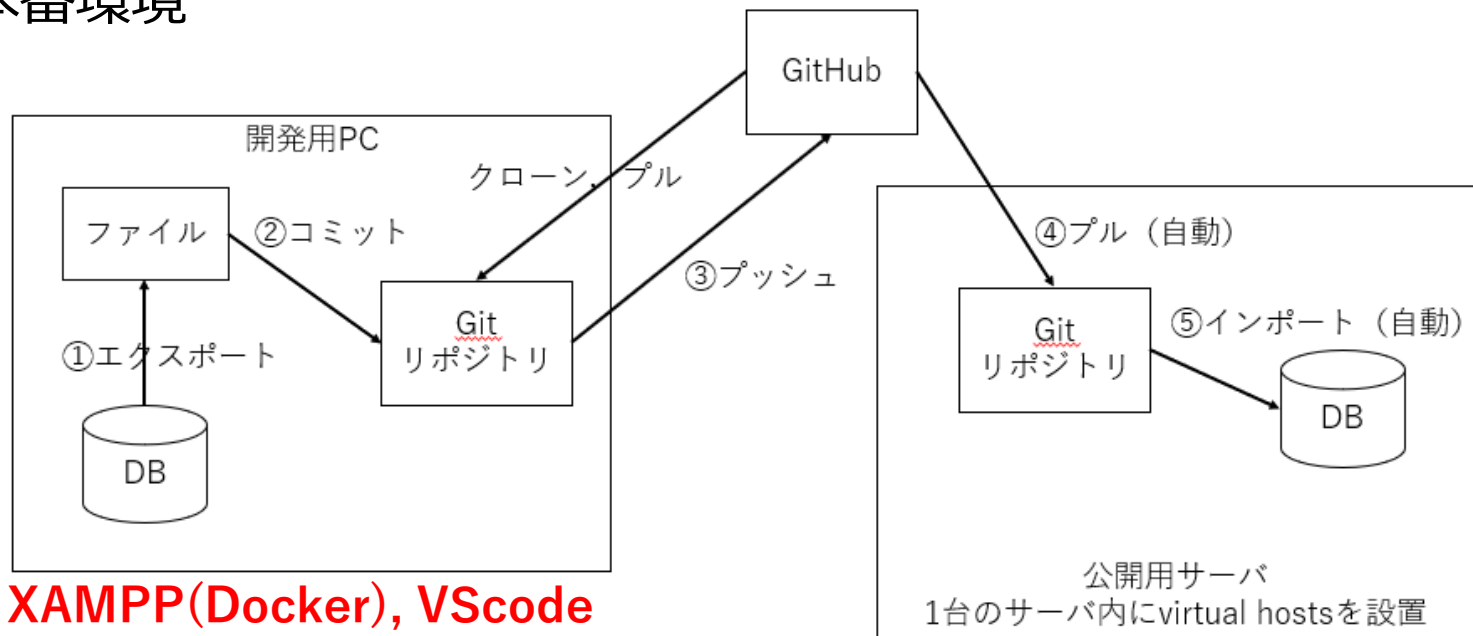


# Webアプリの基本構造と本番環境

## • Webアプリの基本構造



## • 本番環境



# 今年度の方針と反応

- 方針

- 新しい技術なので活用してよい
- 活用した場合、「何ができて、何ができないのか」、「うまく活用するにはどうしたらよいか」ということを考察して欲しい

- 反応（学生とのやり取り、G:学生 O:自分）

- G：Jリーグの全てが分かるすごいサイト作りますよ！
- O：かなりハイスペックだけど？
- G：大丈夫っす。ChatGPTにやらせますから
- O：簡単に動くとは思えないけど。Webアプリの構造知ってる？
- G：もうすでに枠は完成してますので、あとは中身です。
- O：ここからが難しいところだね
- G：来週までにはこの部分は終わると思います

：

どうなったか？ 結局終わらず、作ったものを捨てるように指示。



# 文書作成の場面

- プロジェクト憲章を作成

- ChatGPTへの入力

- これは、Webアプリケーションを開発するプロジェクトです。
    - システム名は、「アルバイトのシフト管理システム」です。
    - 現状の課題は次の2つです。
      - 安易な無断欠勤が多く、職場環境・労働環境が悪くなっている。
      - シフト作成のための連絡方法はLINEで行われていてストレスの元となっている。また、シフト作成に時間がかかっている。
    - このシステムの目的は、労働者と雇用者の双方が管理しやすく、見やすいシフト管理システムを作ることです。
    - 開発期間は2023年4月14日から2023年7月14日です。
    - プロジェクトを立ち上げるにあたり、「プロジェクト憲章」を作成してください。

<https://chat.openai.com/share/99959205-c439-432a-9363-61b6e40a1a50>

# プログラミングではどのような状況か？

## • 2極化

- うまく活用できたグループとそうではないグループ（ぐだぐだ）

## • うまくいったパターン

- 単機能に絞り込んで開発  
利用者：JSの基礎知識なし
- 難しい処理にも活用

利用者：プログラミング能力あり

- 基本的なプログラミングはできる
- DBの設計はできる
- PHPとDBの接続部分は普通に書ける



JavaScriptで開講時期に開催される  
授業が選択肢として出てくる

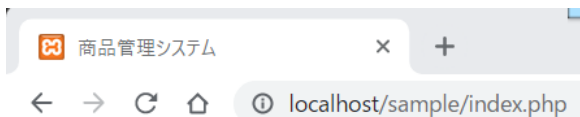
## • うまくいかなかったパターン（ぐだぐだ）

- 利用者：上記赤枠部分の理解がかなり不足
- 典型的なパターン
  - 画面から作り出す
  - かわいい画面ができあがる
  - テーブルを作る
  - つながらない（かなりの時間を費やす）
  - 断念（捨てる、捨てさせる）

## 4. 自分で試してみた

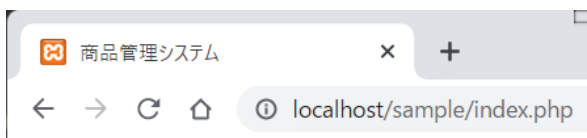
- 自分の中の気持ち
  - SSでの発表を聞き、
  - 学生の活用状況を見て、
  - 簡単なプログラムであれば作れそう、という感触を持つ。
- 開発したWebアプリケーション
  - 3年生前期の授業「ソフトウェア開発管理」で授業をとおして開発する商品管理システム

# 商品管理システムの全体像



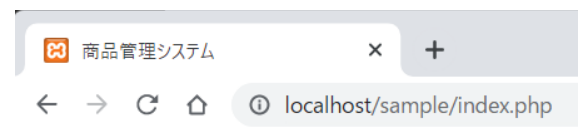
ID	商品名	価格	在庫
1	イヤホン	1500	100
2	モバイルバッテリー	3980	10
3	USB-TypeC接続ケーブル	800	50
6	LOVOT	500000	2
8	Dynabook	200000	2

登録 削除



ID	商品名	価格	在庫
1	イヤホン	1500	100
2	モバイルバッテリー	3980	10
3	USB-TypeC接続ケーブル	800	50
6	LOVOT	500000	2
8	Dynabook	200000	2
9	VLOGCAM	95000	1

登録 削除

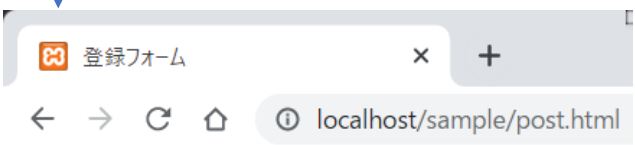


ID	商品名	価格	在庫
1	イヤホン	1500	100
2	モバイルバッテリー	3980	10
3	USB-TypeC接続ケーブル	800	50
8	Dynabook	200000	2

登録 削除

index.php

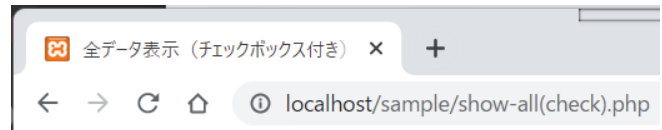
post.html



送信 リセット 戻る

post.php

show-all(check).php



削除	ID	商品名	価格	在庫
<input type="checkbox"/>	1	イヤホン	1500	100
<input type="checkbox"/>	2	モバイルバッテリー	3980	10
<input type="checkbox"/>	3	USB-TypeC接続ケーブル	800	50
<input checked="" type="checkbox"/>	6	LOVOT	500000	2
<input type="checkbox"/>	8	Dynabook	200000	2
<input checked="" type="checkbox"/>	9	VLOGCAM	95000	1

削除 リセット 戻る

delete-item.php

全ファイル共通 : dp.php、style.css

# 開発手順

このあと、ステップ2までを紹介

- ステップ1：DBとテーブル作成
  - テーブルを作成する。初期データとして数個のデータを登録しておく。
  - DB名：sigidudb、テーブル名：productinfo
- ステップ2：データ追加画面作成
  - データ追加画面を作成し、データを追加するプログラムを作成する。
  - ファイル名：post.html、post.php
- ステップ3：データ登録確認
  - ステップ1～2で作成したものを実行し、データが登録されていることを確認する。
- ステップ4：一覧表示プログラム作成
  - 登録されているデータを表形式で表示するプログラムを作成する。
  - ファイル名：index.php
- ステップ5：データ削除プログラム作成
  - 登録されているデータをチェックして、削除するプログラムを作成する。
  - ファイル名：show-all(check).php、delete-item.php
- 開発の前提
  - DBに接続するユーザ（testuser）とパスワード（pass）は事前に設定済み。
  - 全ファイルで共通に使う以下の2つのファイルは自動生成せずに準備する。
    - db.php → エスケープを処理するための関数とDBのユーザ名やパスワードなどの設定し、DBに接続するためのファイル
    - style.css → 色の設定をするためのスタイルファイル

# ステップ1：DBとテーブル作成

## ・入力したメッセージ：

あなたは、商品管理システムを開発する技術者です。  
このシステムでは、次のDBを利用します。

DB名：sigedudb

テーブル名：productinfo

テーブルを使うユーザとパスワード：testuser、pass

テーブルは、次の4つの項目（カラム）を持ちます。

id：プライマリキーです。自動でインクリメントされます。

productname：商品名が入ります。varchar(30)型で、データは必ず存在します

price：価格が入ります。int型で、データは必ず存在します。

stock：在庫数が入ります。int型で、データは必ず存在します。

最初、テーブルには次の3つのデータを登録しておきます。

(1, 'イヤホン', 1500, 100),

(2, 'モバイルバッテリー', 3980, 10),

(3, 'USB-TypeC接続ケーブル', 800, 50);

上記のデータベース名とテーブルを持ち、テーブルを作成し、データを登録するためのSQL文を作成してください。

# 生成されたSQL文

おおよそ想定したものが生成された  
DB名を sigedudb にしてみた！

```
CREATE DATABASE IF NOT EXISTS sigedudb;  
GRANT ALL ON sigedudb.* TO 'testuser'@'localhost' IDENTIFIED BY 'pass';  
FLUSH PRIVILEGES;
```

```
USE sigedudb;
```

```
CREATE TABLE IF NOT EXISTS productinfo (  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    productname VARCHAR(30) NOT NULL,  
    price INT NOT NULL,  
    stock INT NOT NULL  
);
```

```
INSERT INTO productinfo (id, productname, price, stock) VALUES  
(1, 'イヤホン', 1500, 100),  
(2, 'モバイルバッテリー', 3980, 10),  
(3, 'USB-TypeC接続ケーブル', 800, 50);
```

# 実行結果（DBとテーブルの作成完了！）

The screenshot shows the phpMyAdmin interface with the following SQL code entered in the query editor:

```
1 CREATE DATABASE IF NOT EXISTS sigedudb;
2 GRANT ALL ON sigedudb.* TO 'testuser'@'localhost' IDENTIFIED BY 'pass';
3 FLUSH PRIVILEGES;
4
5 USE sigedudb;
6
7 CREATE TABLE IF NOT EXISTS productinfo (
8   id INT PRIMARY KEY AUTO_INCREMENT,
9   productname VARCHAR(30) NOT NULL,
10  price INT NOT NULL,
11  stock INT NOT NULL
12);
13
14 INSERT INTO productinfo (id, productname, price, stock) VALUES
15 (1, 'イヤホン', 1500, 100),
16 (2, 'モバイルバッテリー', 3980, 10),
17 (3, 'USB-TypeC接続ケーブル', 800, 50);
18
```

Buttons for 'クリア', 'フォーマット', and '自動保存されたクエリを取得' are visible below the editor.

The screenshot shows the result of a query in the 'sigedudb' database on the 'productinfo' table. The query executed was:

```
SELECT * FROM `productinfo`
```

The result is displayed as a table with 3 rows:

	id	productname	price	stock
<input type="checkbox"/>	1	イヤホン	1500	100
<input type="checkbox"/>	2	モバイルバッテリー	3980	10
<input type="checkbox"/>	3	USB-TypeC接続ケーブル	800	50

Below the table, there are options for 'すべて表示', '行数: 25', and '行フィルタ: このテーブルを検索'. There are also buttons for '拡張オプション', 'クエリ結果操作', and 'このSQLをブックマーク'.



# ステップ2：データ追加画面作成

## ・入力したメッセージ：

問題なく動きました。データベースとテーブルの作成が終わりました。ありがとうございます。次に、データを追加するための画面を作成します。データを追加するためのプログラムの構成は次のとおりです。

ファイル名：post.html

登録するデータの項目名：商品名、価格、在庫

各項目のフォームの属性：テキストボックス

データ入力後、「追加」ボタンを押すと、post.php が動き、データの追加処理を行います。「キャンセル」ボタンを押すと、入力したデータがクリアされます。また、戻るというリンクがあり、そのリンクを押すと、index.php に戻ります。


まずは、post.html を作成してください。なお、このプログラムは、PC、スマートフォンの両方で使えるようにしたいと思っています。

# 生成されたプログラム

```
1 <!DOCTYPE html>
2 <html lang="ja">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>商品データの追加</title>
7   <style>
8     body {
9       text-align: center;
10    }
11    form {
12      max-width: 500px;
13      margin: auto;
14    }
15    .input-group, h1 {
16      display: flex;
17      justify-content: space-between;
18      align-items: center;
19      margin: 10px 0;
20    }
21    .input-group label, h1 {
22      flex: 1;
23      margin-right: 10px;
24    }
25    h1 {
26      padding-left: 33.33%; /* align w
27    }
28    .input-group input {
29      flex: 2;
30    }
31  </style>
32 </head>
33 <body>
34   <form action="post.php" method="post">
35     <h1>商品データの追加</h1>
36     <div class="input-group">
37       <label for="productname">商品名:</label>
38       <input type="text" id="productname" name="productname" required>
39     </div>
40     <div class="input-group">
41       <label for="price">価格:</label>
42       <input type="number" id="price" name="price" required>
43     </div>
44     <div class="input-group">
45       <label for="stock">在庫:</label>
46       <input type="number" id="stock" name="stock" required>
47     </div>
48     <input type="submit" value="追加">
49     <input type="reset" value="キャンセル">
50   </form>
51   <p><a href="index.php">戻る</a></p>
52 </body>
53 </html>
```

おおよそ想定したものが生成された  
スマホ（レスポンス）対応をしてくれた！

# 開発完了！



ID	商品名	価格	在庫
1	イヤホン	1500	100
2	モバイルバッテリー	3980	10
6	ラグマガ	1200	10
7	LOVOT	750000	2

登録 削除

<https://sel.jpn.org/citweb/chatgpt/index.php>



# 開発実績

- 開発期間

- 2023年6月22日（木）～23日（金）
  - 夕方（17時頃）から開始（1時間）
  - 電車の中でChatGPTとやり取りしながら開発（1時間半）
  - 自宅に着いて、ご飯を食べてお風呂に入ってから作業再開（2～3時間）

- 開発状況

- 最初はスムーズに進んだ
- 途中、レイアウト変更の注文を出したあたりから、ややこしくなってきた。言ったことが伝わらなくてやり取りを繰り返す。
- どうにもうまく対応できないので、途中までうまくいっていたコードを示して（具体例を示す）、これをベースに修正してと依頼。
- 開発完了後、サーバ上にDBを用意して、プログラムをアップ。

- ChatGPTの履歴

- <https://chat.openai.com/share/a5128cca-a084-447e-92c5-9297abf96bfd>

## 5. 感想

- 企業にいた頃、よく外部委託をしていた（要求仕様書と外部設計書を作成し、外部業者に委託）。最後の頃は、中国やベトナムのグループ会社にも委託。
- **その当時と同じような感覚がよみがえってきた。**
  - 今回のやり取りの例
    - 商品データの追加はセンタリングされましたが、商品名、価格、在庫という文字がテキストエリアの上に配置されてしまいました。この3つの文字は、テキストエリアの左側に配置してください。
    - 上記の仕様を満たすpost.phpを作ってください。index.htmlは、スタブとして作成し、商品管理システムというタイトルだけ表示するものを作ってください。
- Webアプリに関して言えば、DBの設計と画面遷移・処理内容（外部設計）が定義されていれば、短時間で開発ができる。
- スマホ（レスポンシブ）対応など、やればできるけどちょっと面倒な作業をお願いできるのは嬉しい。
- 複雑なコードも作れそう。
  - 今回の例
    - チェックボックスを付けて、複数選択可能にする、など。

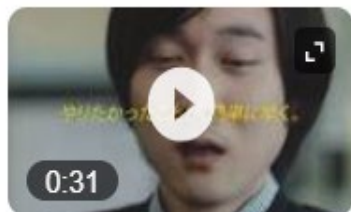
## 6. 今後の進展（予想）

- 文書作成、プログラミングの両方での活用が広がるはず
- 文書作成
  - 定型的な文書（プロジェクト憲章、プロジェクト計画書など）であれば、モデル（例えばPMBOK）に準拠した網羅的な内容を生成できる。
  - 「仏造って魂入れず」がさらに加速する可能性もあるし、逆に、「枠はあるから魂をちゃんと入れよう」となる可能性もある。
- プログラミング
  - 試しに作る、ベースを作る
    - 成果物がある程度明確で、DB設計と外部設計ができているならば、ChatGPTにお願い・相談しながら短時間で作れるはず（Webアプリの場合）。
  - コーディングのサポートを得る
    - 具体的で詳細が指示を出せるならば、複雑なロジックのプログラミングも対応できるはず。専門家にも嬉しい。
  - 共通して言えることは、Webアプリケーションの基本的な作り方の知識は当然必要。

# 酒匂さんの発表の「おまけ」部分

- Generative AI は強力で危険
  - 使い方を間違えるとゴミのような出力を大量に生成する
  - AI の役割、求めている出力（基準）、求めている形式などをうまく整理してガイドしてやると有用な出力を得られる → プロンプトエンジニアリングの重要性、有用なプロンプトのライブラリ化と共有
- ちょっとした作業には極めて便利
  - 簡単な環境設定、ちょっとしたシェルスクリプト、簡単な正規表現、ちょっとしたユーティリティ、Excel のマクロ、基準にあったデータ生成 etc ... 決して難しくはないが、それでも調べながら自分でやると 30分とか1時間とかの時間を使ってしまふものは、あっという間に終わるようになる。
- ノーコード、ローコードにも影響
  - クローズドなノーコード、ローコードを使うくらいなら、オープンなライブラリを組合せてアプリケーションを生成できる Generative AI の方が良いかも？（生成したコードの解析や改善もできる）
- What と 実践を優先
  - 細かいプロンプトテクニックを覚えようとするよりも（細かいテクニックは無効になりがち）、実際に使って、できることできないことを自分で体感して行くことがおすすすめ。あくまでも「問題」に注目したい

# kintone のコマーシャル



キントーンCM 「表計算使ってた時は、天文学的な時間を要し ...

YouTube · kintone

2022/12/12



キントーンCM 「前職では表計算の管理に難儀して...」 30秒

YouTube · kintone

2022/12/12



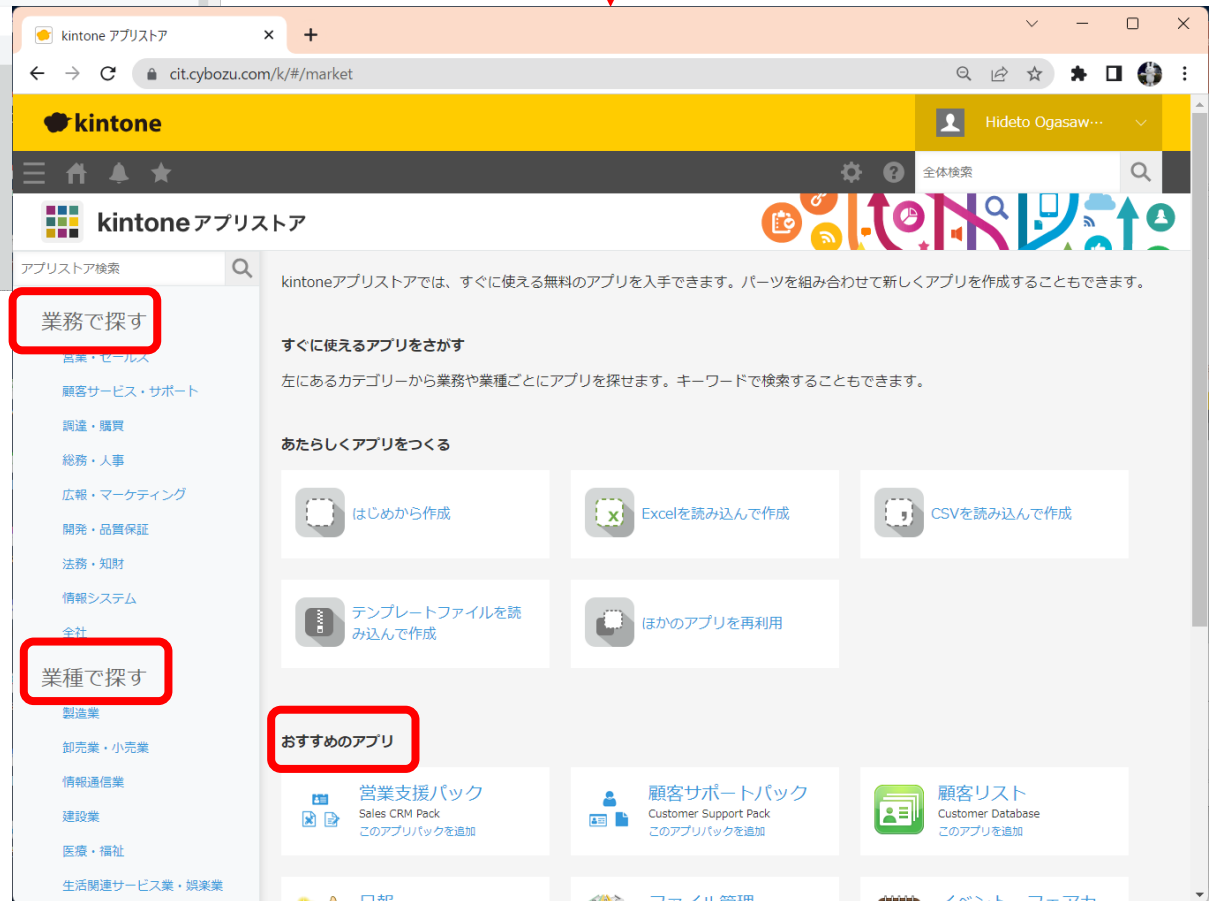
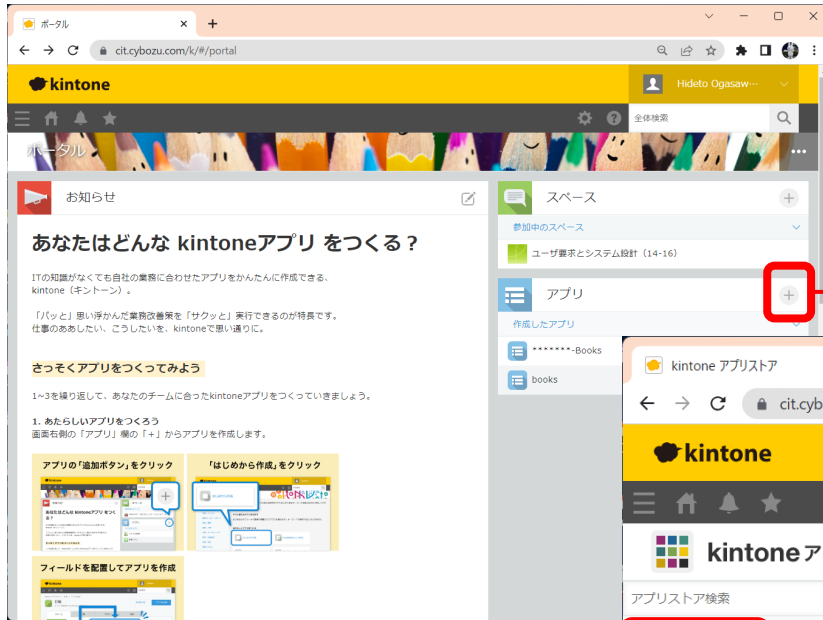
キントーンCM 「Nice」 30秒

YouTube · kintone

2022/12/12



# Kintoneでアプリを作る時の画面



参考にするアプリ  
がたくさんある！

# 100種類を超えるサンプルアプリがある

The screenshot shows the Kintone App Store interface. The browser address bar displays 'cit.cybozu.com/k/#/market/category/chotatsu'. The page features a navigation menu on the left with categories like '業務で探す' (Search by Business) and '業種で探す' (Search by Industry). A red box highlights the '調達・購買' (Procurement/Purchasing) category in the menu, with an arrow pointing to a grid of application cards. A callout box at the bottom right states: '調達・販売だと5種類のアプリがある' (There are 5 types of apps for procurement and sales).

調達・購買 | kintone アプリストア

cit.cybozu.com/k/#/market/category/chotatsu

kintone Hideto Ogasaw...

全体検索

kintone アプリストア

アプリストア検索

業務で探す

- 営業・セールス
- 顧客サービス・サポート
- 調達・購買**
- 総務・人事
- 広報・マーケティング
- 開発・品質保証
- 法務・知財
- 情報システム
- 全社

業種で探す

- 製造業
- 卸売業・小売業
- 情報通信業
- 建設業
- 医療・福祉
- 生活関連サービス業・娯楽業

**調達支援パック**  
サイボウズ株式会社  
このアプリパックを追加

購買業務での材料データ管理、発注申請、サプライヤー情報、打ち合わせの議事録などの情報を全て紐づけて管理が可能です。このアプリパックを活用することで、各種データの一元管理、社内での情報共有、誤発注防止が実現できます。(パック内容: サプライヤーマスタ、材料マスタ、購買)

**物品購入申請**  
サイボウズ株式会社  
このアプリを追加

物品の購入が必要になったとき、事前に申請するアプリです。社外にいても確認・処理ができるので、申請から決裁までのスピードアップが図れます。

**FAQ**  
FAQ  
このアプリを追加

社内手続きに関する、よくある質問と回答をFAQ形式で登録できるアプリです。全社で利用することで、従業員からの問合せ対応時間を短縮することができます。

**材料データ資料管理**  
サイボウズ株式会社  
このアプリを追加

材料ごとの図面、品質基準書、QC工程図、信頼性試験結果などの資料を一元管理できるアプリです。品質・設計担当者も合わせて登録することで、責任者を明確にすることができます。

**仕切値管理**  
Price Adjustment Management  
このアプリを追加

製品メーカーが代理店ごとの仕切値を管理できるアプリです。代理店ごとに異なる製品や仕切値を記録することができます。また他の代理店の仕切値や、他製品の仕切値を同時に閲覧することも可能です。更にロックアップ機能を設定することで、代理店リストや製品リストと連携でき、入力

調達・販売だと5種類のアプリがある

# 期待：オープンなライブラリの広がり

- ノーコード、ローコードにも影響
  - クローズドなノーコード、ローコードを使うくらいなら、オープンなライブラリを組合せてアプリケーションを生成できる Generative AI の方が良いかも？（生成したコードの解析や改善もできる）
- 成果物
  - kintoneで用意されているようなアプリ（100種類以上）
  - アプリをテストするためのテストセット
  - 用途に合わせたカスタマイズ方法やツール
- 効果
  - ユーザ企業内での業務システム開発の加速
  - ベンダー依存からの脱出
  - ソフトウェア技術者の需要増大
  - ソフトウェア技術者がその実力を発揮できる世界の広がり

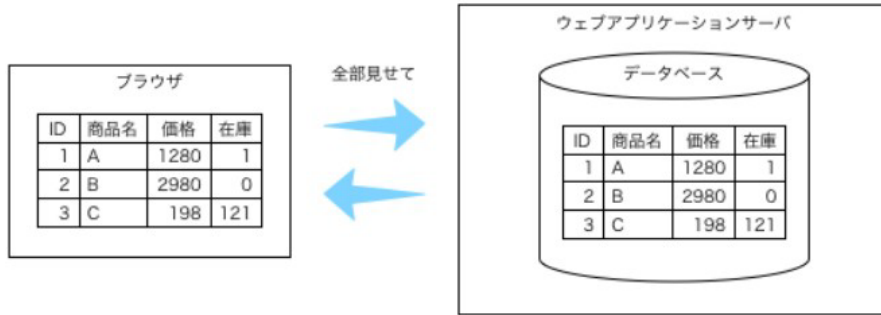
# 付録

- プログラミングのための講義資料の例

# パターン集(前ページの(1)~(4))

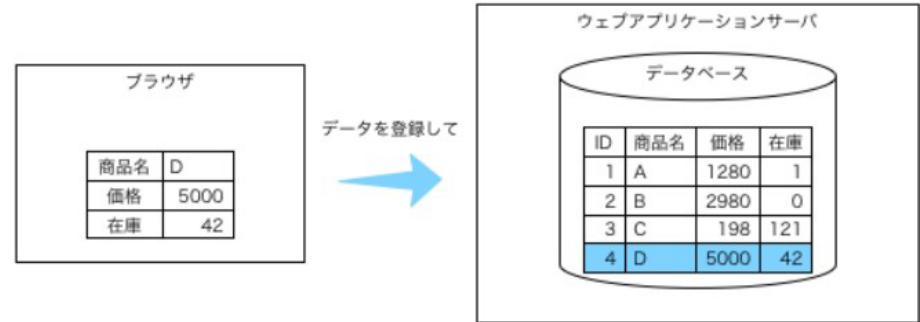
## 全データの表示

テーブル内のデータをすべて表示します。



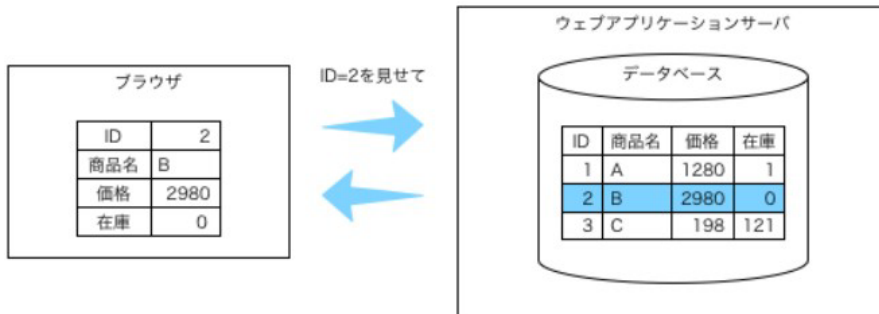
## データの追加

フォームから送られたデータをデータベースに登録します。



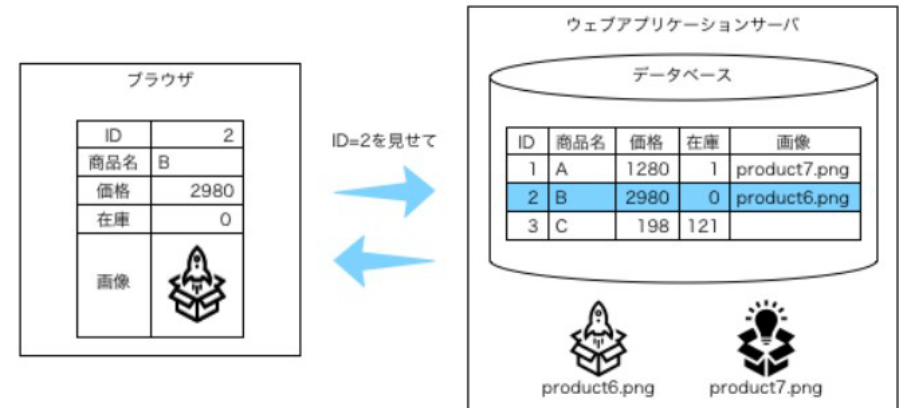
## 特定のデータの表示や削除

テーブル内の特定のデータを表示(削除)します。



## 画像の表示

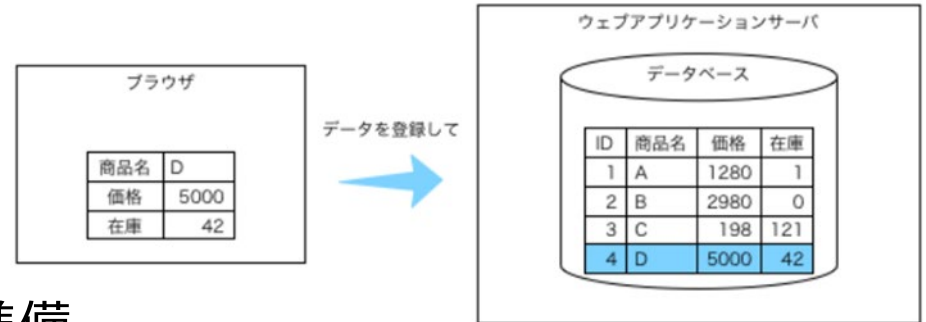
データベースのアイテムに画像を対応させます。



## データの追加

フォームから送られたデータをデータベースに登録します。

# データの追加



### • ステップ1：データベースの準備

# データベースの作成

```
create database mydb charset=utf8mb4;  
grant all on mydb.* to testuser@localhost identified by 'pass';
```

# テーブルを作る.

```
create table productinfo (  
  id int primary key auto_increment, # ここはいつも同じ  
  productname varchar(30) not null,  
  price int not null,  
  stock int not null # 最後にはカンマがないことに注意.  
);
```

# データを登録する

```
insert into productinfo (id, productname, price, stock) values  
(1, 'イヤホン', 1500, 100),  
(2, 'モバイルバッテリー', 3980, 10),  
(3, 'USB-TypeC接続ケーブル', 800, 50);
```

左記をコピーし、phpMyAdminのSQLにペーストすると、半角が全角に変わってしまうことがあります。エラーとなってしまうことがある。エラーが出る場合は、manabaに登録した「DBの設定.txt」を使う。

## • ステップ2：フォームの作成

- HTMLのform要素でデータを入力するためのUIを作る。次のことが大事。
  - form要素のaction属性には送信先のPHPを書く
  - form要素のmethod属性はpost（検索ならget）
  - input要素のname属性にはわかりやすい名前を付ける
  - input要素のvalue属性は不要だが、初期値を入れたい時は設定する（設定方法は自分で探すこと）

登録フォーム

localhost/sdm/6/

商品名

価格

在庫

追加 リセット

```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <meta charset="UTF-8" />
    <link rel='stylesheet' href='style.css' />
    <title>登録フォーム</title>
  </head>
  <body>
    <form action="show6(add).php" method="post">
      <table>
        <tr>
          <th>商品名</th>
          <td><input name="productname" type="text" /></td>
        </tr>
        <tr>
          <th>価格</th>
          <td><input name="price" type="text" /></td>
        </tr>
        <tr>
          <th>在庫</th>
          <td><input name="stock" type="text" /></td>
        </tr>
      </table>
      <input type="submit" value="追加" />
      <input type="reset" value="リセット" />
    </form>
  </body>
</html>
```

post.html

## • ステップ3：データ追加用のプログラムを作成

### • 実行する内容は簡単（単純）

- ① post.html で入力されたデータを取得する
- ② 取得したデータを追加できるSQL文を作成する
- ③ 作成したSQL文を実行する

### • post.htmlでデータを設定し、「追加」ボタンを押下する。

### • phpMyAdmin でデータが追加されたことを確認する。

```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <meta charset="UTF-8" />
    <link rel='stylesheet' href='style.css' />
    <title>データの追加完了</title>
  </head>
  <body>

    <?php
    # 送信されたデータの取得
    $name = $_POST['productname']; # 商品名
    $p     = $_POST['price'];       # 価格
    $s     = $_POST['stock'];       # 在庫

    require 'db.php'; # 接続
    $sql = 'insert into productinfo (productname, price, stock) values (:name, :p, :s)';

    $prepare = $db->prepare($sql); # 準備

    $prepare->bindValue(':name', $name, PDO::PARAM_STR); # 埋め込み1
    $prepare->bindValue(':p', $p, PDO::PARAM_STR);      # 埋め込み2
    $prepare->bindValue(':s', $s, PDO::PARAM_STR);      # 埋め込み3

    $prepare->execute(); # 実行
    ?>

    <p><a href="show-all.php">確認</a></p>
  </body>
</html>
```

左記①の処理

左記②の処理（SQL文の作成）

↑  
左記③のコード