**概要**

1. 概要
   1. ソフトウェアの概要

・九州域内の需要を予測し、電力の調達をサポートすることを目標とする。

* 1. どうしてこの制作を選んだのか？

・5年以上、電力システムに関わっており、システムの要件部分に時間をかけずに、

コーディングに集中して、制作を行えると考えたため。

・既存の需要予測システムに改善の余地があると考えたため。

・比較対象(※1)が明確にあり、改善可否の判断が明確であると考えたから。

1. 機能要件
   1. ユーザ認証機能(「ログイン」画面)

・[ユーザ名]と[パスワード]による認証を行う。

・新規ユーザの追加は、メーカメンテ項目とする。

* 1. ホーム機能(「ホーム」画面)

・ユーザ認証の成功時には、最低限の表示機能を含むWebページを表示する(※2)。

* 1. 九州域内の気象情報の概要表示(「気象情報(概要)」画面)

・最高気温、最低気温、降水確率を表示する。

・雨雲レーダーを表示する。

* 1. 九州域内の気象情報の詳細表示(「気象情報(詳細)」画面)

・1時間単位の気象情報の詳細を表示する。

* 1. 電力の需要予測と実績の表示(「需給予測情報」画面)

・一般送配電事業者が予測した需要予測情報を表示する。

・一般送配電事業者が収集した需要実績情報を表示する。

・本システムにて予測した需要予測情報を表示する。

・一般送配電事業者が予測した需要と本システムにて予測した需要予測情報を、

実績を基に比較表示する。

1. 非機能要件
   1. 性能要件

・画面情報の更新等は、要求から10秒以内に完了すること。

* 1. セキュリティ要件

・ユーザ認証は安全な方法で行われること。

1. データ要件
   1. 保存されるデータ

・最新の気象情報を保存する。過去の気象情報は、今回は保存しない。

・一般送配電事業者が予測した最新の需要予測/需要実績をCSV形式で保存する。

1. インターフェース要件
   1. ユーザインターフェース

・メインウィンドウの共通部には、各画面へ遷移するためのボタンや[更新]ボタンを

表示する。なお、相手サーバ側の負荷を考慮し、自動更新は行わない。

・メインウィンドウの表示部には、共通部で選択されたボタンの画面を表示する。

・メインウィンドウ本体の拡大/縮小、スクロールは考慮しない。

・必要性がない限り、サブウィンドウは使用しない。

・ダイアログやポップアップを表示する際には、メインウィンドウは操作不可とする。

* 1. 外部システムインターフェース

・気象協会Webサーバから気象情報を取得する。

・一般送配電事業者のWEBサーバから需給情報を取得する。

1. 動作環境(想定)
   1. ハードウェア要件

(クライアント)

・プロセッサ(CPU): Intel Core i5プロセッサ

・メモリ (RAM): 8GB 以上

・ストレージ: 500MB の空き容量

・グラフィックカード: 内蔵グラフィック(特定のグラフィック要件はなし)

・ネットワーク: インターネット接続必須(Ethernet または Wi-Fi)

・その他: キーボード、マウス

(サーバ)

・プロセッサ (CPU): Intel Core i7プロセッサ

・メモリ (RAM): 16GB 以上

・ストレージ: 5GB の空き容量

・グラフィックカード: 内蔵グラフィック(特定のグラフィック要件はなし)

・ネットワーク: インターネット接続必須(Ethernet または Wi-Fi)

・その他: キーボード、マウス

* 1. ソフトウェア要件

(クライアント)

・Windows: Windows 11 Pro

(サーバ)

・Windows: Windows Server 2016 Essentials

1. テスト要件
   1. 機能テスト

・サーバ　–　クライアント間の通信を確認する。

・ユーザインターフェースの操作性を確認する。

* 1. パフォーマンステスト

・パフォーマンスモニタなどのツールを使用して、極端なサーバへの負荷がかからないことを

確認する。

1. フォルダの構成

以下にフォルダの構成を記載する。

(フォルダ構成)

C:.

├─00\_backup

├─01\_ドキュメント

├─dmd\_app

│ └─\_\_pycache\_\_

├─dmd\_gui

│ ├─migrations

│ │ └─\_\_pycache\_\_

│ ├─static

│ │ └─dmd\_gui

│ │ ├─capture

│ │ ├─CSS

│ │ │ ├─00\_backup

│ │ │ └─01\_images

│ │ ├─html

│ │ └─new\_demand\_data

│ ├─templates

│ │ ├─dmd\_gui

│ │ │ └─00\_backup

│ │ └─registration

│ └─\_\_pycache\_\_

└─\_\_pycache\_\_

1. その他考察
   1. 制作で苦労したところ

・(技術面)バグを発見した際に、解決に時間を費やしすぎて、時間が足りなくなった。

・(機能面)機械学習が通用しない場合の処理がうまくいかなかった(※3)。

* 1. 今後の展望

・(技術面)設計書などを充実させ、可能な限りバグが発生しないようにコーディングしたい。

・(技術面)効率の良いデバッグの方法などを身に着けて、バグ解決の時間を短縮したい。

・(技術面)レスポンシブ対応を充実させたい。

・(機能面)データ前処理の段階で異常値や外れ値の扱いを工夫し、機械学習モデルの

精度を向上させたい。

・(機能面)機械学習モデルのパラメータチューニングや、他のアルゴリズムの検討を行い、

予測精度を向上させる方法を検討したい。

・(機能面)過去の実績データをCSVファイルではなく、データベースで管理するように

変更したい。また、そのデータベースに最新の需要実績値を保存するように

変更し、学習機能を充実させたい。

(※1)電気予報：<https://www.kyuden.co.jp/td_power_usages/pc.html>

(※2)動作不良時等の切り分けを容易にするため

(※3)過去需要が「コロナ時の需要」を含むため、平時より需要が低く(工場閉鎖などが要因)、

そのデータを使用して機械学習を行い算出した需要の予測結果も低くなってしまった。

また、お盆期間中の平日に関しても、工場閉鎖などによる需要の低下を考慮できておらず、

実際の需要実績値よりも需要予測値が高くなってしまった。