

“明日にも提案できる”

連携ノウハウと 高速開発テンプレートのご紹介



agenda

01 | Angularとは？

02 | 現場でのWeb Client
システム開発の流れ

03 | エトピリカテンプレート

04 | MSAL連携

05 | LINE連携

06 | AI・IoT連携



Angularとは？



Angularとは？(開発時)

ブラウザシステムは

HTML
CSS
JavaScript

で構成されています。
それらをいい感じに構築してくれるフレームワークが



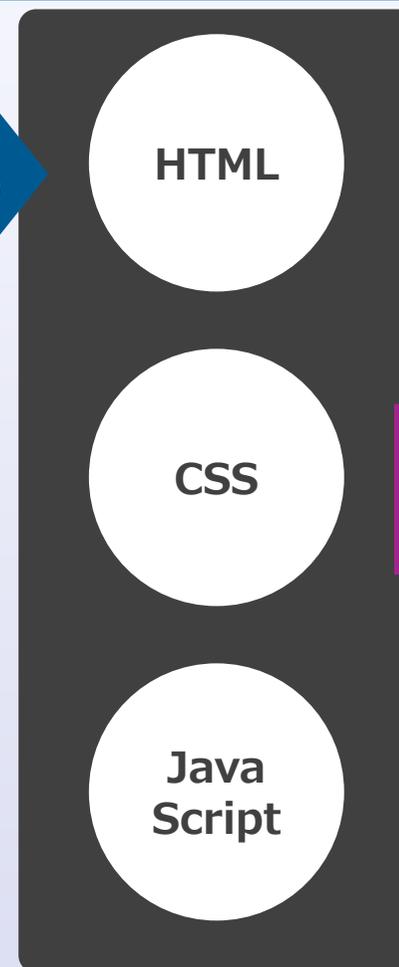
Angularを知らなくてもとりあえず
Magicが動くものを自動生成してくれます。



生成



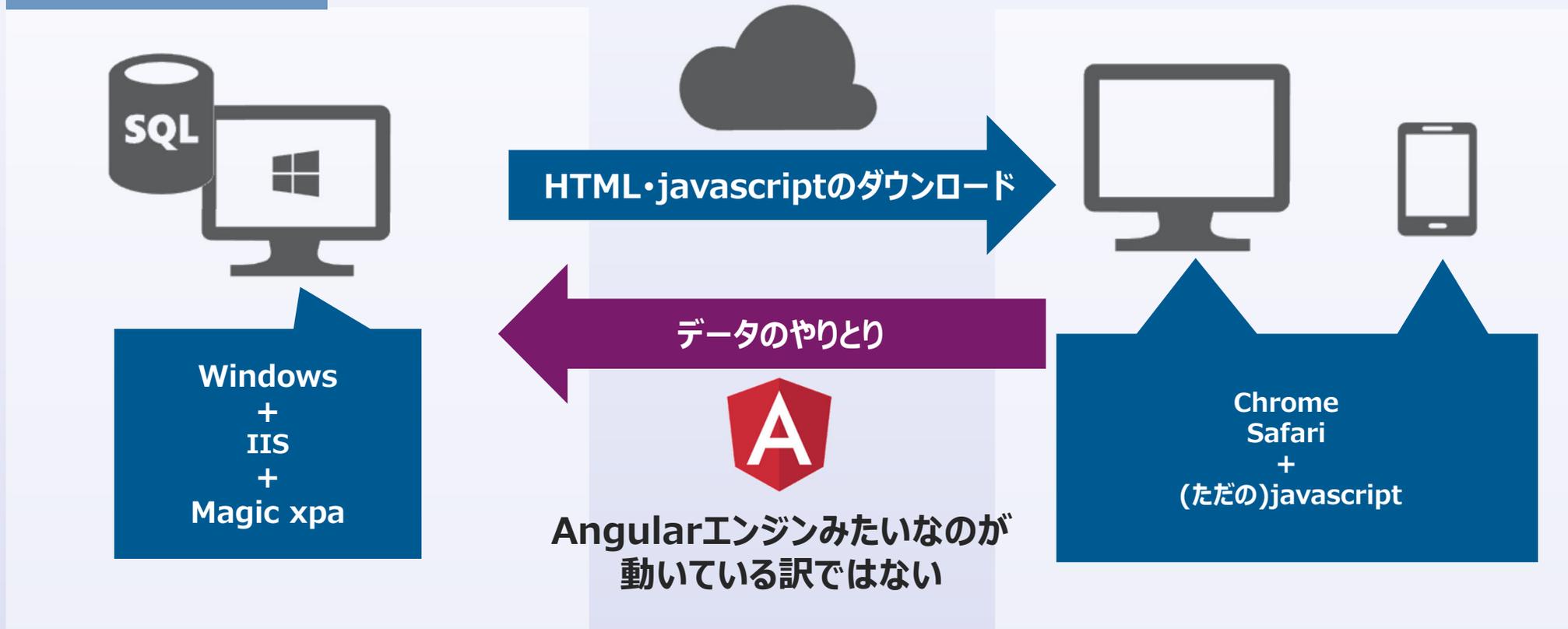
Magic xpa
Web Client



ブラウザ
表示



Web Client



Web Clientの開発の流れ



Web Client開発の流れ

02 | Web Clientの開発の流れ

要件定義(画面設計)

画面はHTMLを書かなければいけない。
オンプレやRIAと違ってドラッグ & ドロップのみで
プロトタイプを作ることは難しい。

おもに

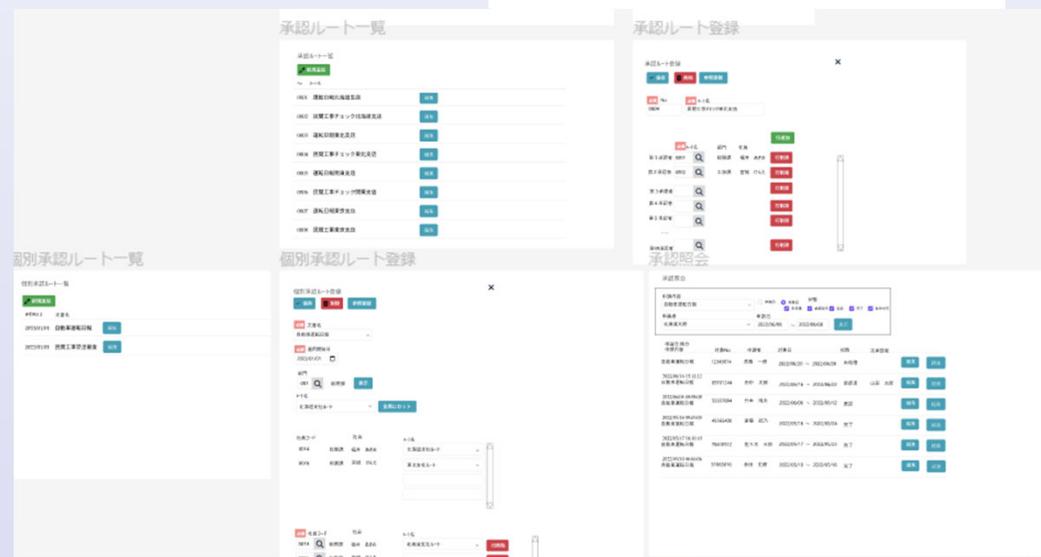
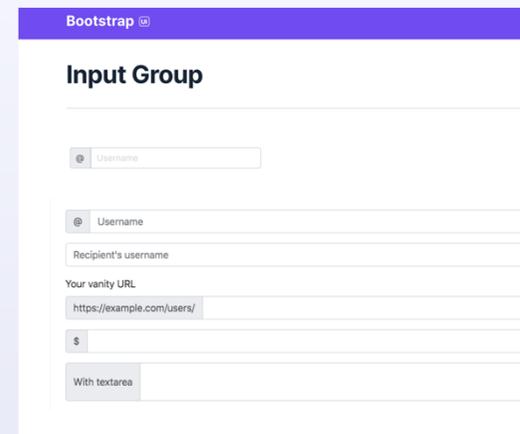
Figma | フィグマ



を使って画面イメージを構築し、エンドユーザーに
確認してもらっています。

Figmaでは後述するBootstrapテンプレートを使用し
画面を構築。

Figmaは複数人による共同開発、確認が可能です。



詳細設計

いつも通りでOKですが
Magicの以下の項目は **英数字のみ** で日本語は使えません

- プログラム名
- 画面コントロール名
- フォルダ名
- 画面名

日本語⇔英数字 プログラム名対比表を作成しています。

日本語	英数字
得意先コード	TokuiCode

Web Client開発の流れ

02 | Web Clientの開発の流れ

Magic Optimizer 最強です

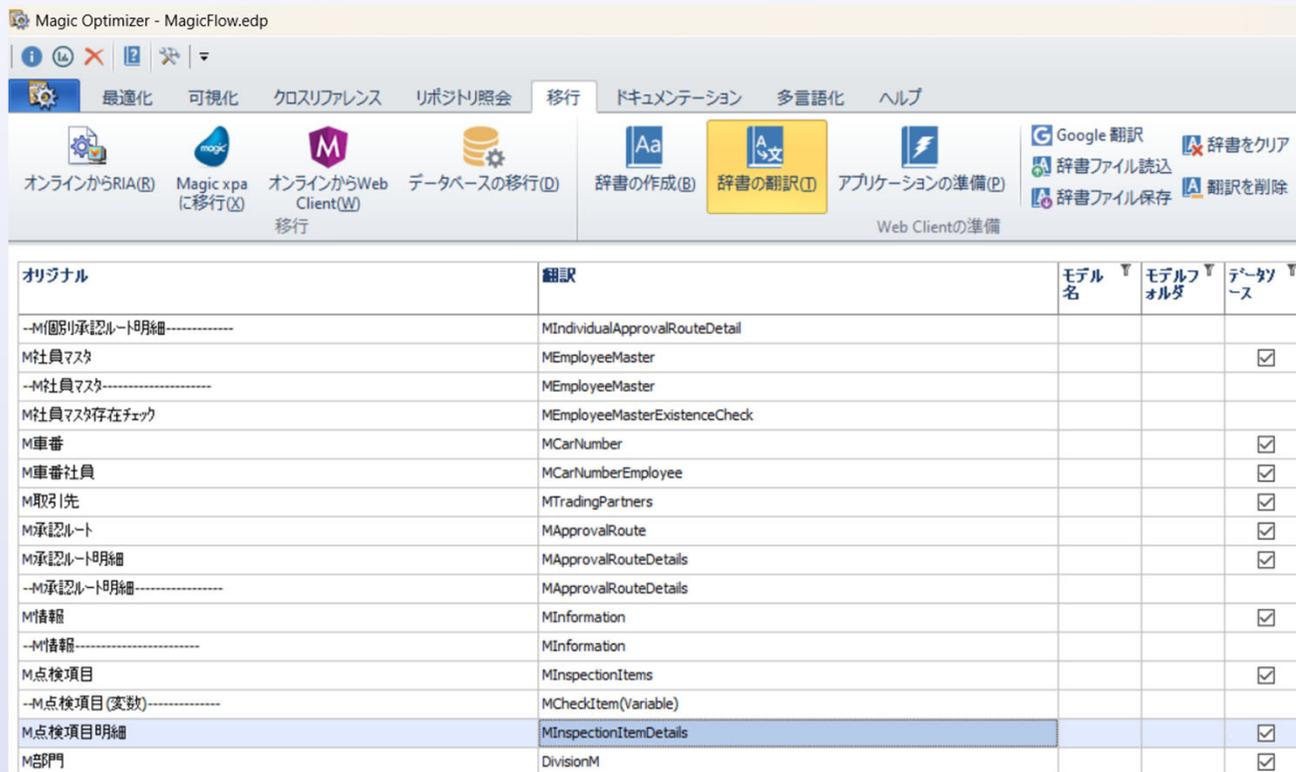
日本語 ➡ Google翻訳 ➡ 英数字

にしてくれて、これだけで数日分の仕事が一瞬で終わります。

**既存システムのリプレイス
一部分をWeb Clientにする場合**

重宝します。

※ソースを変えるのでバックアップはこまめに！ ロジック部分の変数なんかも英語化してえらい目にあいました；



The screenshot shows the Magic Optimizer application window. The title bar reads "Magic Optimizer - MagicFlow.edp". The menu bar includes "最適化", "可視化", "クロスリفرنス", "リポジトリ照会", "移行", "ドキュメンテーション", "多言語化", and "ヘルプ". The ribbon contains various icons for actions like "オンラインからRIA(R)", "Magic xpa に移行(X)", "オンラインからWeb Client(W) 移行", "データベースの移行(D)", "辞書の作成(B)", "辞書の翻訳(I)", "アプリケーションの準備(P)", "Google 翻訳", "辞書ファイル読み込", "辞書ファイル保存", "辞書をクリア", and "翻訳を削除". Below the ribbon is a table with columns: "オリジナル", "翻訳", "モデル名", "モデルフォルダ", and "データベース".

オリジナル	翻訳	モデル名	モデルフォルダ	データベース
--M個別承認ルート明細-----	MIndividualApprovalRouteDetail			
M社員マスタ	MEmployeeMaster			<input checked="" type="checkbox"/>
--M社員マスタ-----	MEmployeeMaster			
M社員マスタ存在チェック	MEmployeeMasterExistenceCheck			
M車番	MCarNumber			<input checked="" type="checkbox"/>
M車番社員	MCarNumberEmployee			<input checked="" type="checkbox"/>
M取引先	MTradingPartners			<input checked="" type="checkbox"/>
M承認ルート	MApprovalRoute			<input checked="" type="checkbox"/>
M承認ルート明細	MApprovalRouteDetails			<input checked="" type="checkbox"/>
--M承認ルート明細-----	MApprovalRouteDetails			
M情報	MInformation			<input checked="" type="checkbox"/>
--M情報-----	MInformation			
M点検項目	MInspectionItems			<input checked="" type="checkbox"/>
--M点検項目(変数)-----	MCheckItem(Variable)			
M点検項目明細	MInspectionItemDetails			<input checked="" type="checkbox"/>
M部門	DivisionM			<input checked="" type="checkbox"/>



デザイン辞書の作成

処理や画面によって使用する色を変えないように
デザイン用の辞書を作成。
SASSに反映させます。

SASSの構築

Syntactically Awesome Stylesheets

の略で日本語に訳すと、
「構造的にすばらしいスタイルシート」

CSSをプログラマ的に記述します。
分岐、繰り返しが可能です。

The screenshot displays the 'etupirka Design Template' interface. It features a dark blue header with the text 'etupirka Design Template'. Below the header, the 'Buttons' section is highlighted, showing a grid of buttons in four colors: Primary (blue), Info (teal), Danger (red), and Success (green). Each color has three states: 'default', 'hover', and 'focus'. Below the buttons, there are sections for 'Warning' (orange), 'TabControl' (dark blue), and 'etupirka Design' (dark blue). The 'Warning' section shows three states: 'default', 'hover', and 'focus'. The 'TabControl' section shows three states: 'default', 'hover', and 'focus'. The 'etupirka Design' section shows three states: 'header', 'sidemenu', and 'hover'. Below these sections, the 'FontAwesome' section is visible, showing various icons and their corresponding class names: magnifying-glass f002, check f00c, pencil f303, rotate-left f2ea, and trash-can f2ed.



デザイン辞書の作成

SCSSディレクトリ構造

処理や画面によって使用する色を変えないように
デザイン用の辞書を作成。
SASSに反映させます。

SASSの構築

SASS構築ルールには **FLOCSS** を採用

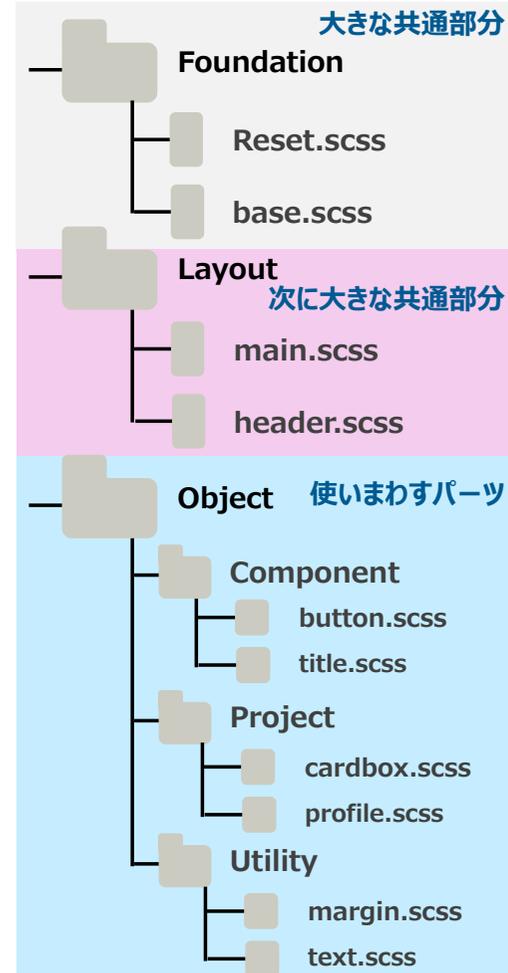
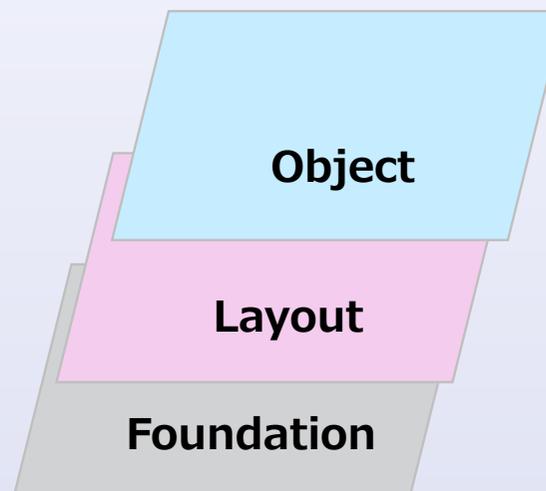
いちばんしっくりきたので・・・。

CSS命名規約には **BEM** を採用

命名規約がないと

```
<input class="takechan5">
```

なんてことになりそうです。



ロジック部分の作成

02 | Web Clientの開発の流れ

ロジック部分は流石Magic。

ほぼ **RIAと同じ開発手法** でいけます！ 難関はコールとルーティングの違い。



デザイン部分の作成

02 | Web Clientの開発の流れ

画面 = HTML+CSS

画面はHTMLとCSSを使ってつくります。
あとは**Magic xpa**がAngularに準じた
TypeScriptコードを出力するので
特にAngularは意識することなく
HTMLやCSSを修正したらアプリの完成！

簡単!



外部の企業などにデザインを委託する場合

ホームページ制作会社などタグやCSSまで構築してもらう場合は
作成されたHTMLに**Magicのタグを挿入する“だけ”**で
綺麗なシステムが出来上がります。

簡単!



そんなわけない。



**フロントデザイン + 基本プログラム
= エトピリカテンプレート**



デザインセンス？

HTML？
CSS？

Angular？

TypeScript？

コマンド入力？

タスク構造？

メニューの部分・・・

ログイン認証
複雑



Q デザインセンスをどうする？

A プロのデザイナーが作成した
落ち着いて出しゃばらないデザインを提供いたします。
違うデザインが良い場合はデザイナーを紹介いたします。



The screenshot shows the 'etupirka' web client interface. At the top, there's a header with the 'magic' logo and the text 'etupirka'. Below the header, there's a sidebar with a globe icon and the text '売上伝票一覧' (Sales Invoice List). The main content area has a search filter for '日付' (Date) with a dropdown menu showing '年/月/日' (Year/Month/Day) and a search button labeled 'Q 表示' (Search) and a '新規登録' (New Registration) button. Below the search filter is a table with columns: '伝票番号' (Invoice Number), '計上日' (Posting Date), '得意先名' (Customer Name), and '金額' (Amount). The table contains two rows of data:

伝票番号	計上日	得意先名	金額	
8414	2023/01/04	有限会社ロビン・スペシャル	1,188	編集
8413	2023/01/04	株式会社アシュラバスター	540	編集



Q HTML、CSSの知識をどうする？

A 基本デザインのHTMLとCSSを準備しました。
Angularを変更しなくてもMagic上からデザインのクラスを
コールできるように改造済み。



The screenshot shows the 'etupirka' web client interface. It features a sidebar with navigation icons and a main content area titled '売上传票一覧' (Sales Invoice List). A search filter for '日付' (Date) is set to '年/月/日' (Year/Month/Day). Below the search bar are buttons for '表示' (Display) and '新規登録' (New Registration). The main area contains a table with columns for '伝票番号' (Invoice Number), '計上日' (Posting Date), '得意先名' (Customer Name), and '金額' (Amount). Two rows are visible, each with an '編集' (Edit) button.

伝票番号	計上日	得意先名	金額	
8414	2023/01/04	有限会社ロビン・スペシャル	1,188	編集
8413	2023/01/04	株式会社アシュラバスター	540	編集



Q Angular? 何それおいしいの?

A おいしいです。



Q TypeScript? 覚えなきゃならない?

A コピペで済むようにサンプル集をガンガン作ります



Q いまさら黒い画面でコマンド入力するの？

A 項目の配置だけでも我慢してください。

```
5 <div [magic]="mgc.M000101">
6   <div class="row">
7     <div class="col-12 ">
8       <div class="card">
9         <div class="card-body">
10          <div class="form-group">
11            <button
12              class="btn "
13              [magic]="mgc.bCreate"
14            >
15              {{mg.getValue(mgc.bCreate)}}
16            </button>
17          </div>
18        </div>
19      </div>
20    </div>
21  </div>
22  <div class="form-group">
23    <magic-subform [magic]="mgc.List">
24    </magic-subform>
25  </div>
26 </div>
```





モデル
リポジトリ



フィールド設計

プログラム
リポジトリ



プログラム設計

データ
リポジトリ



データベース設計

生成

テンプレートなし

テンプレートあり

高速開発は
そのまま！

ID	Name	City	Phone	Email	DateOfBirth	Height	Weight	Comment
1	佐藤 一郎	東京	0301111	satouichir1997/01/200	88	ひこう ん		編集
2	渡辺 かずひろ	アクラ	0209999	kazuhiro1050/03/162	78	コメント 2		編集
3	西下 あきよし	アシガバート	0304444	nishi@em1944/01/174	43	コメント 3		編集
4	須藤 たけい	アスタナ	0123333	suto@em1977/01/156	77	コメント 4		編集
5	Zhang Jane	アスマラ	0359373	zhang@g2001/08/192	66	コメント 5		編集
6	Michael Chan	アテネ	0359373	chan@em2020/04/130	55	コメント 6		編集
7	中山 ひろし	アビア	0345678	nekohiros1989/04/155	44	コメント 7		編集
8	山崎 すずむ	アブジャ	0609876	gohansus2002/11/163	77	コメント 8		編集
9	反町 ける	アブダビ	0401231	gto@gma1070/06/198	99	コメント 9		編集
10	大松 くる	アムステルダム	0501231	dai@ema1975/01/176	68	コメント 10		編集

デザイン履修は後からOK
直感操作デザイン実装済み
学習コストが低い
容易にできるカスタマイズ
素早い納品
容易な保守と運用

etupirka

売上伝票一覧

日付
年/月/日

検索 新規登録

伝票番号	計上日	得意先名	金額	
8414	2023/01/04	有限会社ロビンズスペシャル	1,188	編集
8413	2023/01/04	株式会社アシュラバスター	540	編集
8412	2023/01/04	有限会社ロビンズスペシャル	594	編集
8411	2022/07/16	株式会社アシュラバスター	702	編集
8410	2022/07/16	株式会社パロススペシャル	605	編集
8409	2022/07/16	ハリケーンミキサー企画	999	編集
8408	2022/07/16	シンフォニー新聞社	540	編集
8407	2022/07/16	株式会社パロススペシャル	540	編集
8406	2022/07/16	シンフォニー新聞社	1,286	編集
8405	2022/07/16	有限会社ロビンズスペシャル	1,512	編集
8404	2022/07/16	株式会社アシュラバスター	486	編集
8403	2022/07/16	ハリケーンミキサー企画	1,475	編集
8402	2022/07/16	株式会社パロススペシャル	495	編集
8401	2022/07/16	株式会社アシュラバスター	540	編集
8400	2022/07/16	ハリケーンミキサー企画	1,220	編集

エトピリカ テンプレート

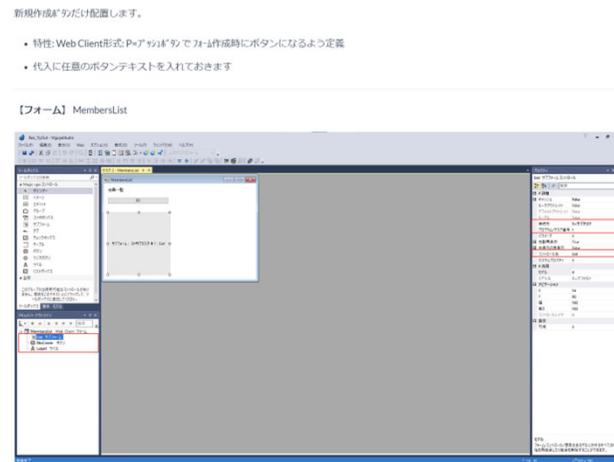


Web Client Utilization Seminar
2023.01

Q Web Client用にタスク構造考えるのがもう面倒!! Magic難しい!!

A サンプルのタスクの形をご提供いたします。
メインテーブルなどを変更いただければそのまま使えます。

またマニュアルも構築中



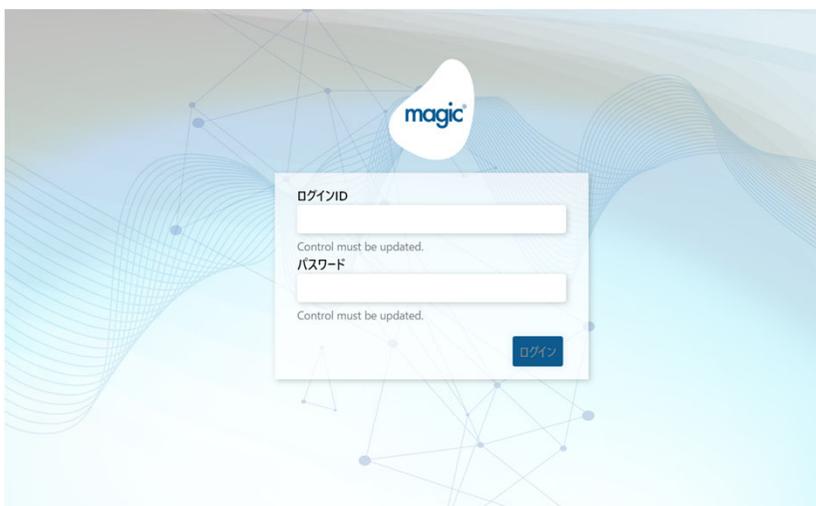
Q メニューの部分はTypeScriptで作ってるの？
もう無理；

A こちらもマニュアルやMagicユーザーズフォーラムにて情報提供します。



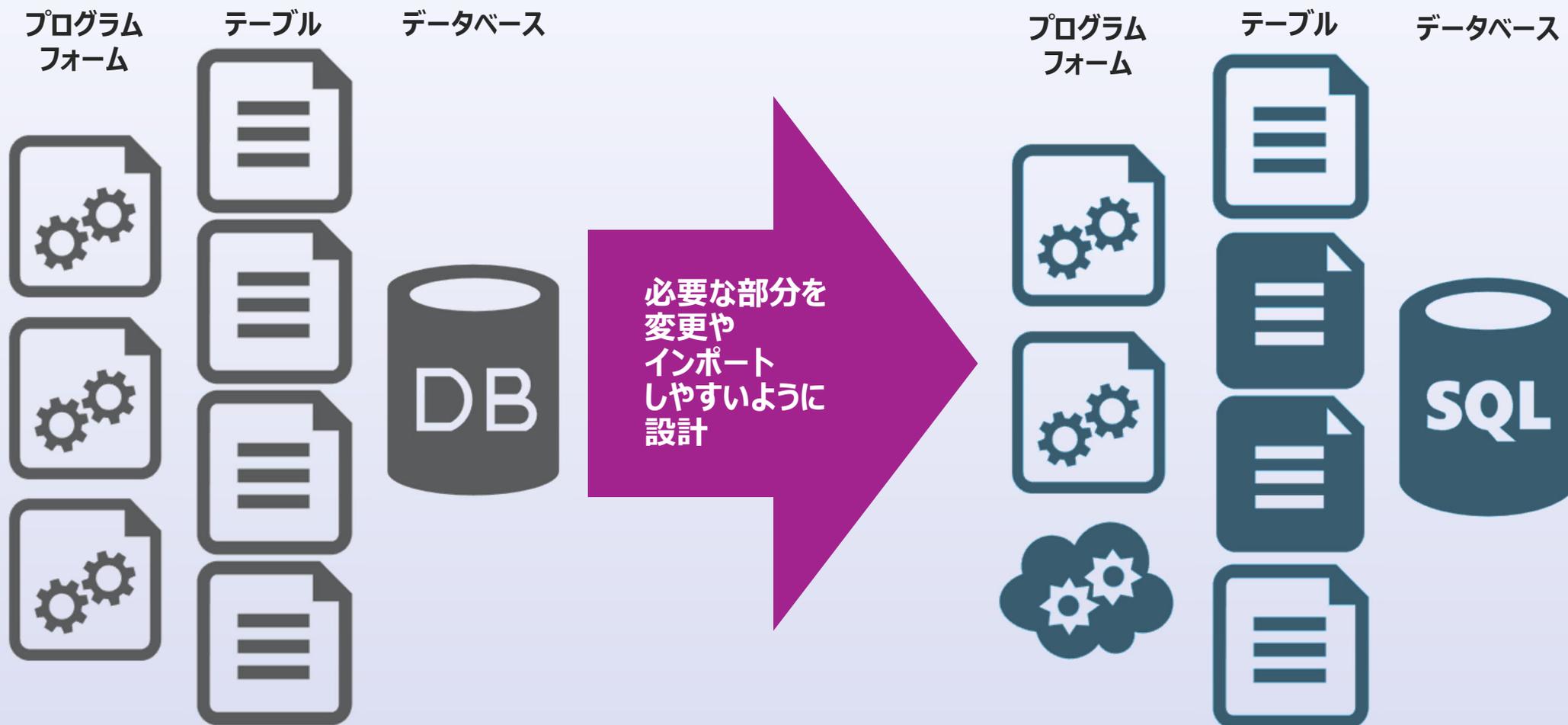
Q ログイン認証複雑。
ログインしていないのにURLいれたら
入力できるようになってる；

A ログインしていない場合は必ずログインさせたいですね。
構築済み！提供いたします。



おさらい) Web Clientの学習コストを引き下げるために

03|エトピリカテンプレート



おさらい) Web Clientの学習コストを引き下げるために

03|エトピリカテンプレート

デザインテンプレート

プロのデザイナーにお願いしたデザインのテンプレートです。

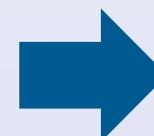
伝票番号	計上日	得意先名	金額
8404	2022/01/04	有限会社ロビンズビル	1,108
8403	2022/01/04	株式会社アツツシビル	540
8402	2022/01/04	有限会社ロビンズビル	594
8401	2022/07/16	株式会社アツツシビル	702
8400	2022/07/16	株式会社ロビンズビル	605
8409	2022/07/16	パワースタキサー会	999
8408	2022/07/16	シフエー不動産	540
8407	2022/07/16	株式会社ロビンズビル	540
8406	2022/07/16	シフエー不動産	1,296
8405	2022/07/16	有限会社ロビンズビル	1,152
8404	2022/07/16	株式会社アツツシビル	486
8403	2022/07/16	パワースタキサー会	1,475
8402	2022/07/16	株式会社ロビンズビル	495
8401	2022/07/16	株式会社アツツシビル	540
8400	2022/07/16	パワースタキサー会	1,220

1 レスポンシブデザインのCSSフレームワークにはBootstrapを採用いたしました。



ここに至るまで1年(笑)

2 CSS・SCSSの命名規約には緩くながらFLOCSS、BEMを採用いたしました。



ここに至るまで3年(笑)

3 Magic側から「CSSクラス」を設定することができます。

ボタンの色

「必須」表示

など

詳細

決定

必須

プログラムテンプレート

使用しているテーブルとフォームの項目をコピーして変えれば動く。
プログラムテンプレートをMagic xpaチームで構築しました。

最初は「デザインのみ」のテンプレートだったんです。
構築している間に思ったんです。
大体のプログラムってパターンがありますよね？

1. マスタなどの単票形式

2. 売上傳票などの伝票形式

3. ヘッダに条件を入れて明細を表示する一覧形式

4. ズームプログラム

5. ファイルダウンロード・ファイルアップロード

6. PDF表示

7. ログイン処理

8. 権限設定も可能なメニュープログラム

9. ルーティング設定

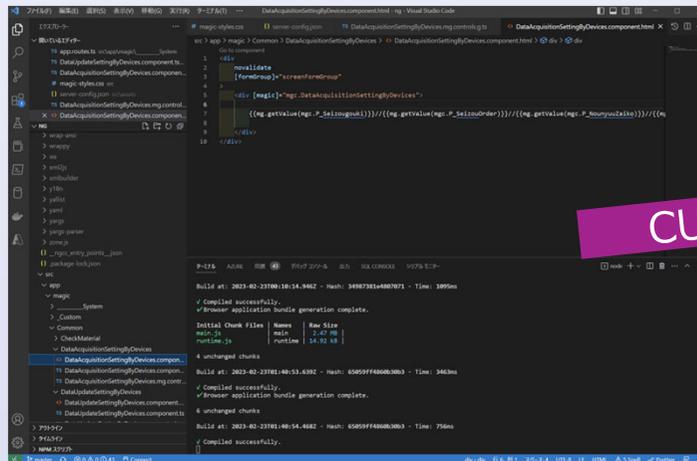
他、要望お待ちしております。

Q3.

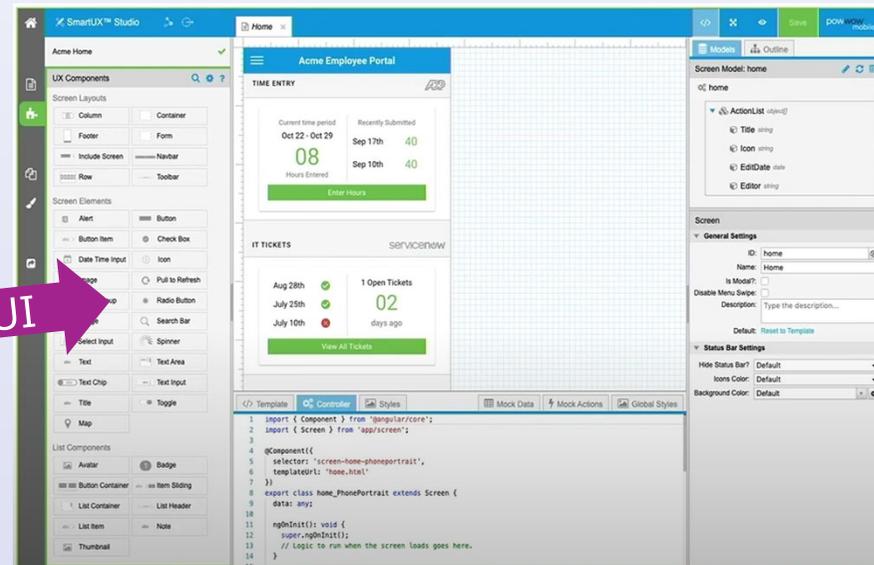


Magic xpa Web Client + SmartUX Designer

「VisualStudio」から「SmartUX Designer」へ
Magic側では入力項目などを「フォームにおくだけ」
SmartUX側でデザイン構築をおこないます。

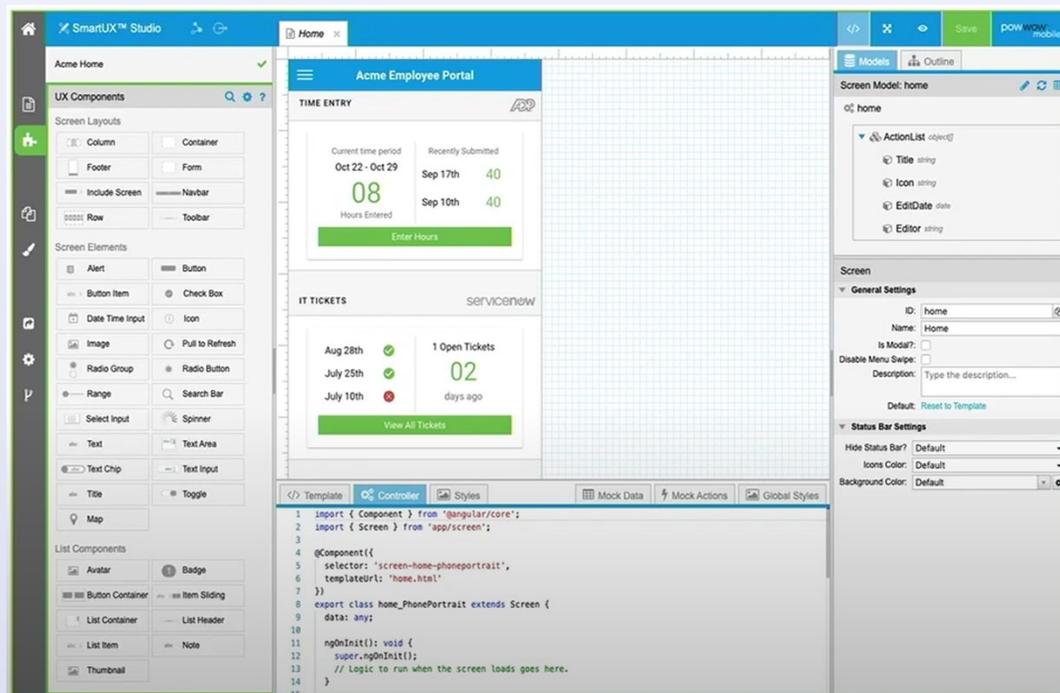


CUI → GUI



日本語化やマニュアル、テンプレート構築などQ3まで頑張ります ; ;

カスタムテンプレートの追加可能 (エトピリカテンプレートの移植 ; ;) SmartUXコンポーネントの新規追加可能 (ダウンロード・アップロードの仕組みも ! ?)



1. ビジュアルデザイン環境

- UXコンポーネント Drag & Dropで画面を構築
- 属性情報の設定もビジュアル的に

2. コードビューも

- HTML5、TypeScript
CSS(**ionic**)



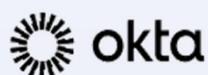
連携事例:MSAL



MSAL(Microsoft Authentication Library)

04 | 連携事例:MSAL

認証を独自のシステムではなくMicrosoftの認証を使いたい。
シングルサインオンに対応してほしいなど要望が多数あります。



他には世界的に有名な
okta社の認証を使用することもあります。

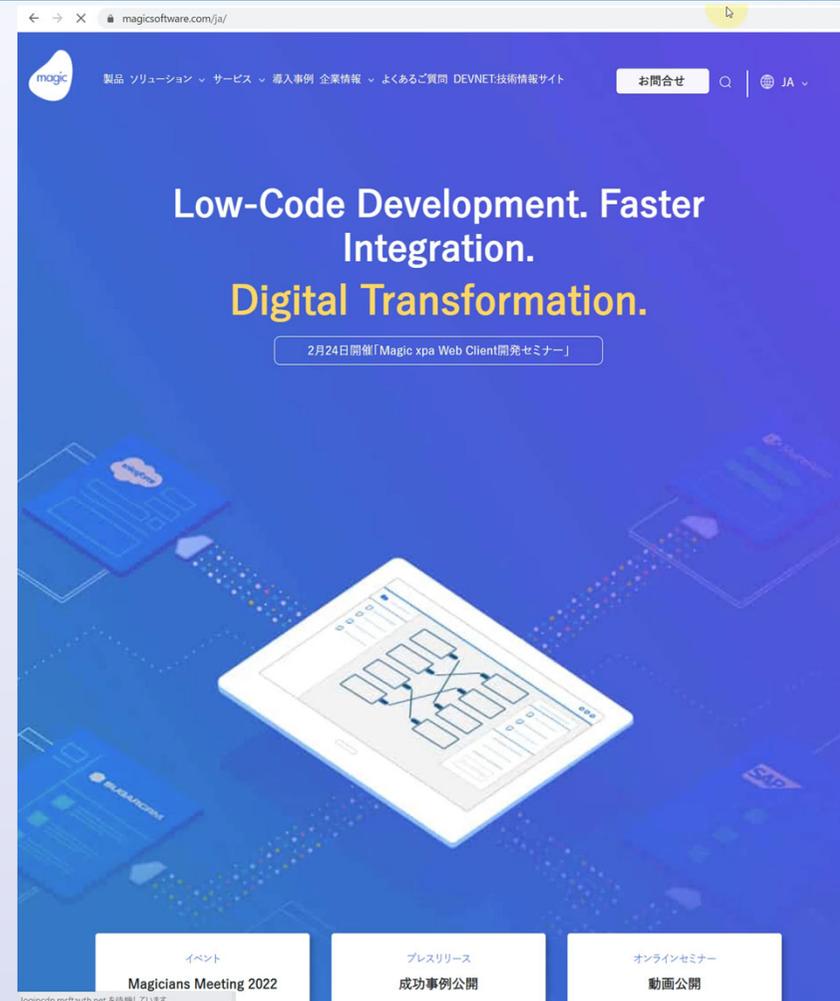
使用モジュール

```
@azure/msal-angular@2.1.2  
@azure/msal-browser@2.22.1
```

ソースは近々
Magicユーザーズフォーラム
にて公開いたしま“せん”。



他のソース解放が先だと思うので ; ; でも興味がありましたら担当営業まで！



Web Client Utilization Seminar
2023.01

連携事例:LINE①



プッシュ通知機能の実装



Firestore Cloud Messaging API



ネイティブアプリの常駐



PWA
(Progressive Web Apps)

※PWAについて今回詳しくはやりません ;

その他



Push7

など

iOS16等でのWEBプッシュ通知解禁が発表されるとはいえ ;



LINE Notify 登録したアカウントからの通知

05 | 連携事例: LINE①

<https://notify-bot.line.me/ja/>



Magic xpa

POST

LINE Notify

通知

- Web Clientのシステムで受注が入った時
- 予約システムでキャンセルが入った時
- オンプレやRIAのシステムで赤伝を入力した時

LINE

iOS

Android



LINE Notify LINE側の設定

連携事例:LINE①

アクセストークンの発行(開発者向け)

パーソナルアクセストークンを利用することで、Webサービスの登録をせずに通知を設定することができます。

トークンを発行する

トークンを発行する

トークン名を記入してください(通知の際に表示されます)

システムからの通知

通知を送信するトークルームを選択してください

Search by group name



1:1でLINE Notifyから通知を受け取る



1/14ワークショップ打ち上げ



11月20日(日)PONTE

発行したトークンはこちらです。

KwICgGlvW4o2qCzQeMgezsc8DPmls

から移動すると、新しく発行されたトークンは二度と
ん。トークンをコピーしてからページを移動して下さい

コピー

閉じる



LINE Notify Magic側の設定

05 | 連携事例:LINE①

データビュー ログ フォーム

1	R=レコード*	S=後	
2	外部コール	.=.NET	public static void func()

```
using System;  
using System.Text;  
using System.Net;  
using System.Web;
```

```
public static class Snippet  
{
```

```
    public static void func()  
    {  
        var token = "LINEで取得したトークン";  
        var url = "https://notify-api.line.me/api/notify";  
        var enc = Encoding.UTF8;  
        var payload = "message=" + HttpUtility.UrlEncode("マジックソフトウェア・ジャパン 竹本 靖夫", enc);
```

```
        using (var wc = new WebClient())  
        {  
            wc.Encoding = enc;  
            wc.Headers.Add("Content-Type", "application/x-www-form-urlencoded");  
            wc.Headers.Add("Authorization", "Bearer " + token);  
            var response = wc.UploadString(url, payload);  
        }  
    }  
}
```

```
}
```

このころはC#で作っていましたが

Magicユーザーズフォーラム ➡ Web Client Tips & Tricks ➡ Magic xpaとグループLINEの連携



Web Client Utilization Seminar
2023.01

LINE Notify 実はHTTPCall一行で済みます

05 | 連携事例: LINE①

HTTPCallで済む

```
HTTPCall('Post','https://notify-api.line.me/api/notify','message=通知内容',  
'Authorization: Bearer トークン','Content-Type: multipart/form-data')
```



連携事例:LINE②





LINE上で動く
アプリ開発
LIFF





LINE上で動く
アプリ開発

LIFF



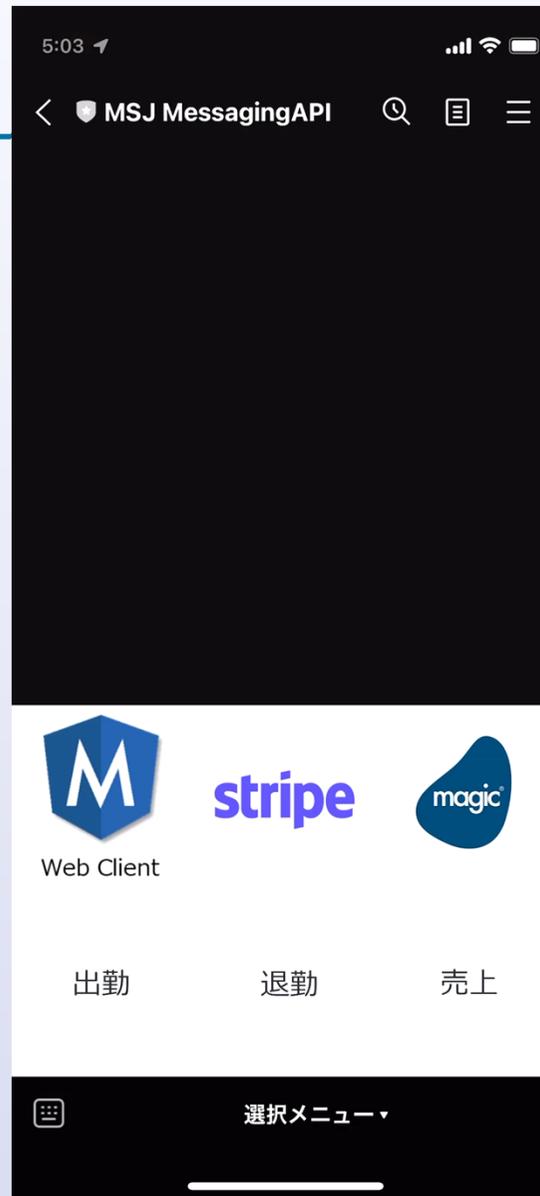
ブラウザアプリ

{ え？わざわざLINEから起動しなくていいのでは？！



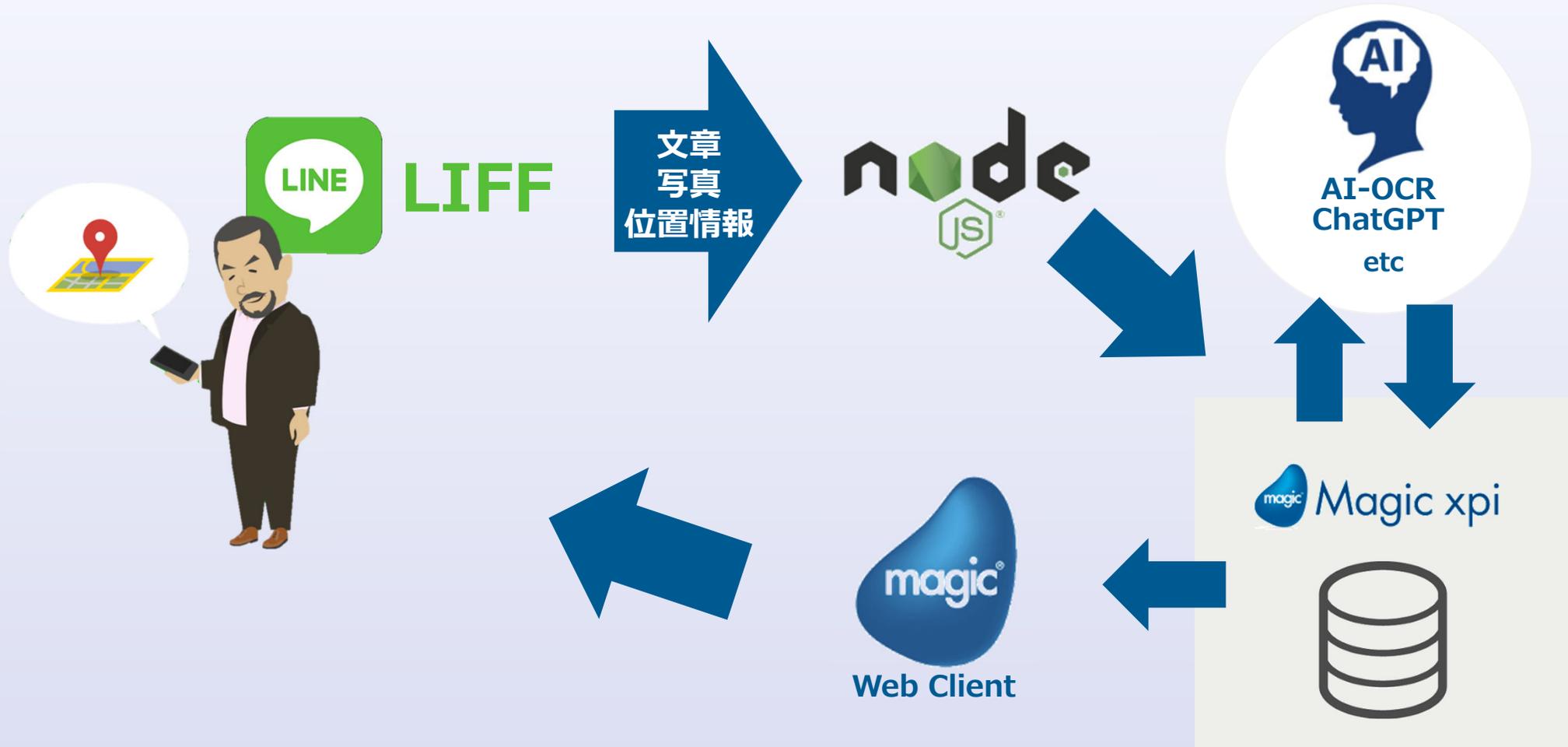
いきなりサンプルデモから

05 | 連携事例: LINE②



Bot(出勤・退勤・今日の売上の部分)

05 | 連携事例: LINE②



LINEが提供している開発者向けサービスのひとつ

国内でも利用者数の多い **LINE上にアプリを展開** することで、多くのユーザーに直接サービスを提供できるようになりました。

今後ますます

需要が高まる予想！

何ができるの？

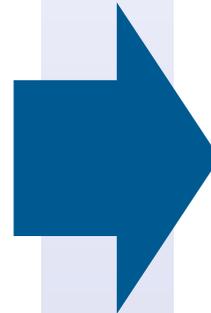
どんな使われ方
してるの？





LIFFで出来ること

- LINEのトークルーム上でWebアプリを起動する
- トークルームにメッセージを送信する
- UXの向上したLINE Botを作成する
- 特定のユーザー相手へメッセージを送信する
- ユーザーIDを利用しログインの手間を省く
- ユーザー独自の履歴表示



Magic xpaで出来ること

システムと関連付け可能

- 個別通知
- 条件による通知
- データベース連携など





サーバーのスペックもありまして；

**実際のデモに興味があるかたは
お近くの営業担当まで**



連携事例:オンライン決済サービス

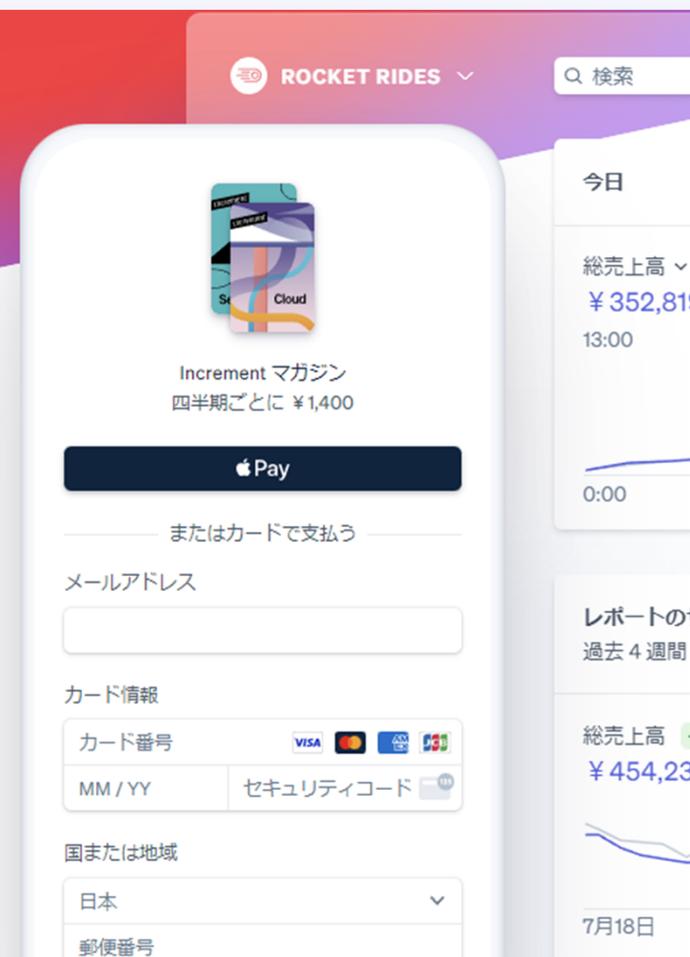


インターネット向け 決済インフラ

スタートアップから大企業まで、あらゆる規模の多数の企業が Stripe ソフトウェアと API を使用して支払いを受け付け、入金し、オンラインでビジネスを管理しています。

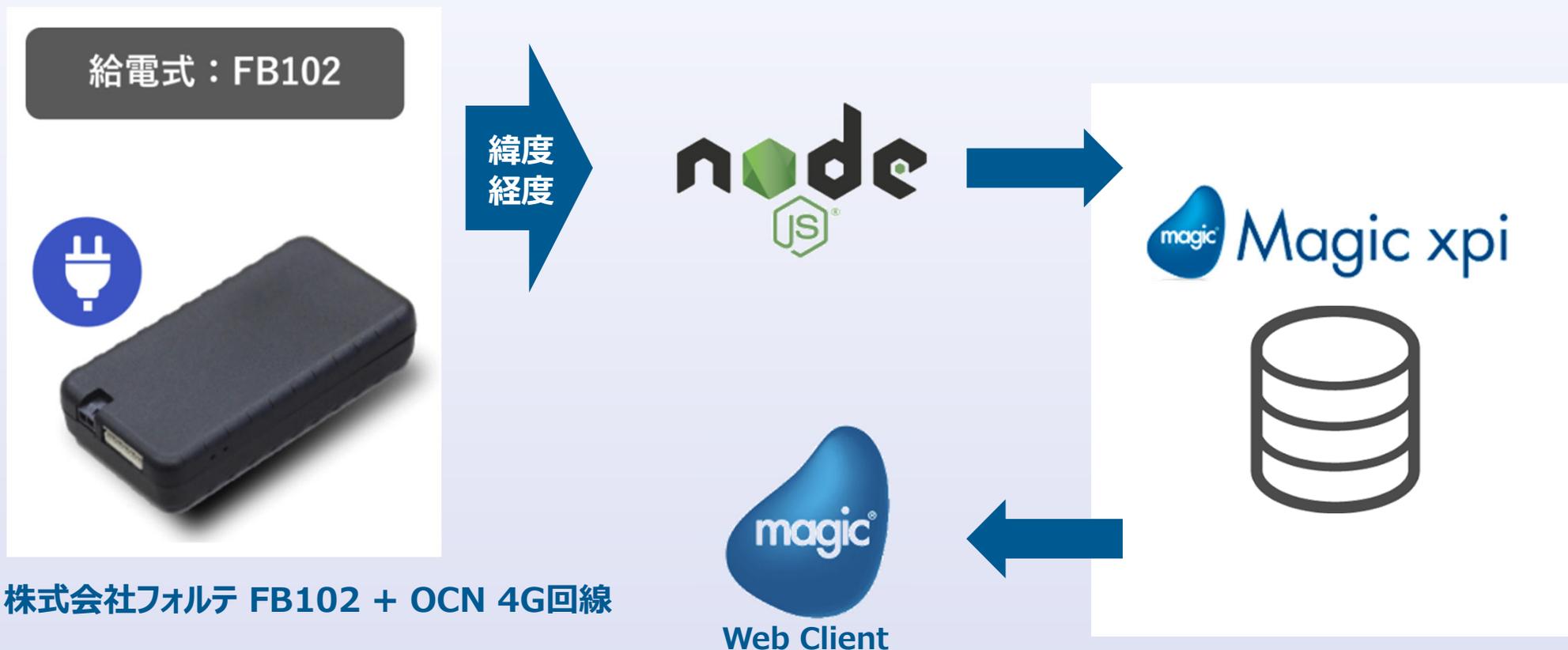
今すぐ始める >

営業チームへお問い合わせ >



連携事例:地理院地図





株式会社フォルテ FB102 + OCN 4G回線

- 人工衛星「みちびき」に対応
- バッテリレスなので夏でも安心



地図の使用には料金がかかたりしますので地理院地図で。

OpenStreetMapはガンガンいくとキックされるというわさもあり。



緯度・経度の情報を
DataViewToJSON でJSON形式にしたのち
CallJS('MapView', BlobANSI変数)
のようにAngular側の関数を呼び出す

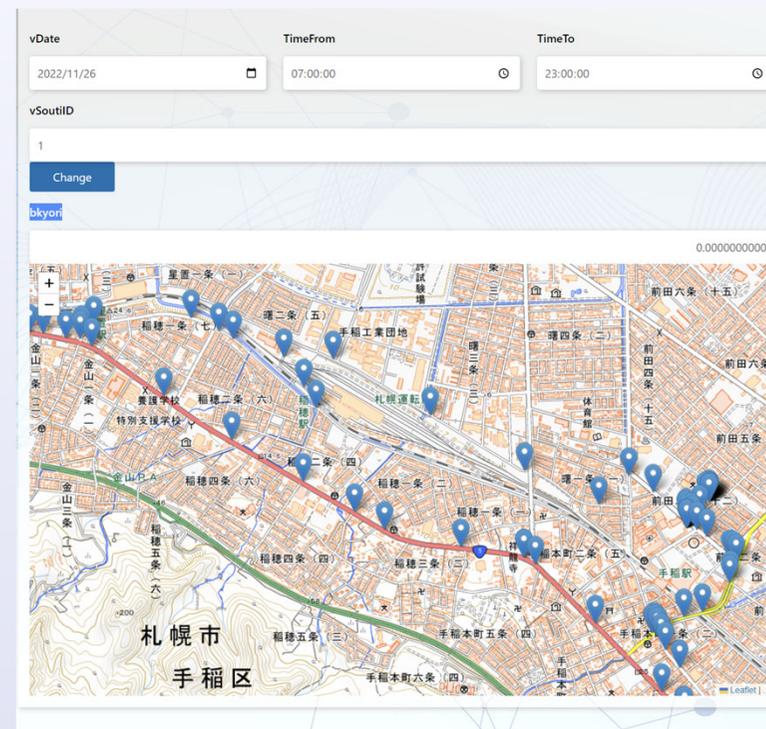
使用モジュール

leaflet@1.8.0

Magic側のソースは近々 Magicユーザーズフォーラム にて公開いたします。



FB102・Node.JS・Magic xpi連携に興味があるかたは
お近くの営業担当まで



配送指示の効率化

急な依頼・問い合わせがあっても、リアルタイムで車両の位置が分かるため、車両の位置をドライバーに確認することなく、配送指示を出すことができます。

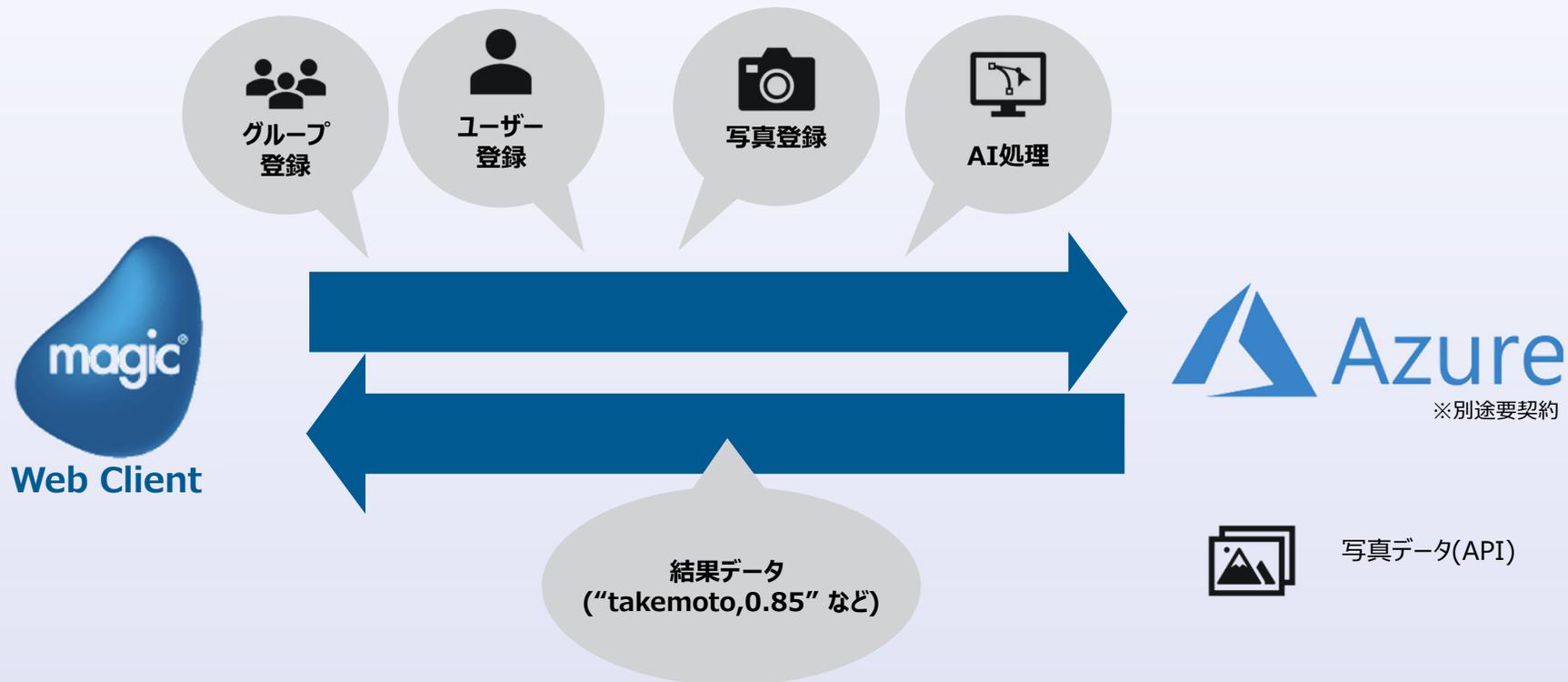
情報の見える化

得意先情報の住所と連携することで地図と販売管理などの情報を視覚化できます。



連携事例: Azure Face API





C#(dll)にて“ほぼ”構築済み。Magic xpaとの連携も可能です。

興味があるかたはお近くの営業担当まで



技術情報:highcharts①



Highcharts(折れ線グラフ)



グラフの情報を
DataViewToJSON でJSON形式にしたのち
CallJS('highcharts', BlobANSI変数)
のようにAngular側の関数を呼び出す

使用モジュール

highcharts@ 10.2.0
highcharts-angular@3.0.0

ソースは近々
Magicユーザーズフォーラム
にて公開いたします。



技術情報:highcharts②



Highcharts(WBS:作業分解構成図)



グラフの情報を
DataViewToJSON でJSON形式にしたのち
CallJS('gantt',BlobANSI変数)
のようにAngular側の関数を呼び出す

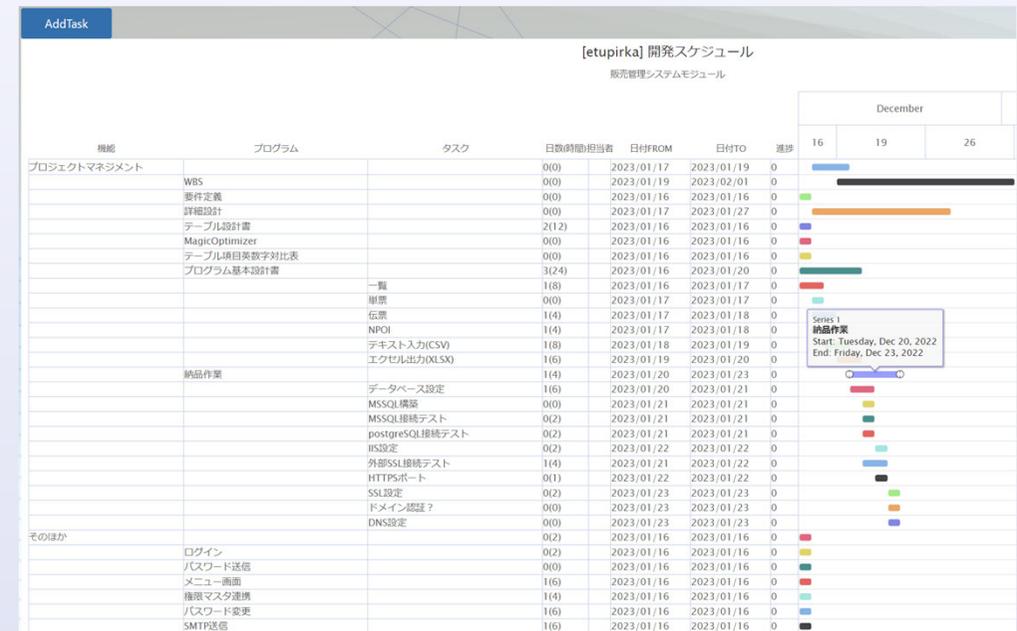
使用モジュール

highcharts@ 10.2.0
highcharts-angular@3.0.0



ソースは近々 Magicユーザーズフォーラム にて公開いたしま"せん"。

なんせソースが中途半端なもので ; でも興味がありましたら担当営業まで !



- 右側の棒の部分のイベントを取ることが可能
- イベント範囲の変更のイベントを取ることにも可能



技術情報:WebUSB API



WebUSB API ①出力

06 | AI・IoT連携

ブラウザからローカルUSBポートについでいる装置を制御するもの。
基本的にドライバーはいらない。

Windows、Mac(Chrome)で動く。
Safariは未対応。

動画は

“EPSON TM-P20B563”

ソースは近々
Magicユーザーズフォーラム
にて公開いたします“せん”。



なんせ機器の相性問題などもあるので；でも興味がありましたら担当営業まで！



WebUSB API ②入力

06 | AI・IoT連携

ブラウザからローカルUSBポートについている装置を制御するもの。
基本的にドライバーはいらない。

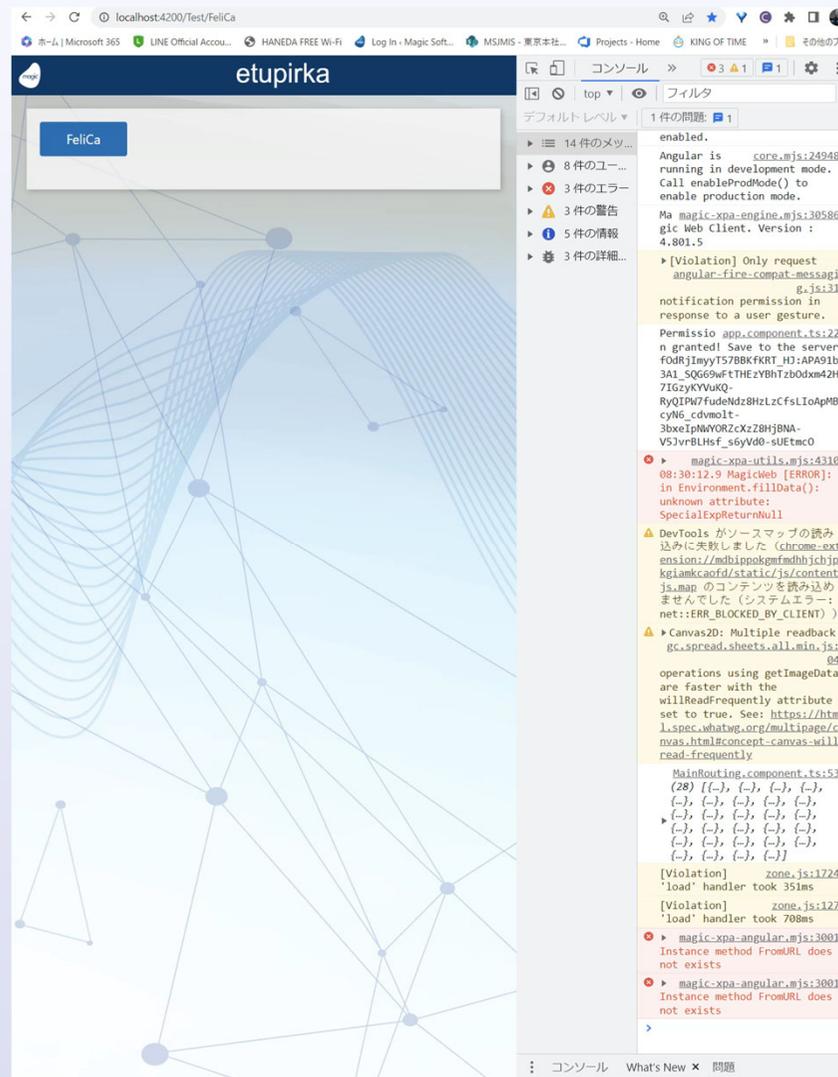
Windows で動く。
Safariは未対応。
Mac+Chromeは動作せず；

動画は
"SONY RC-S380 + Kitaca"
で IDm を読むところ



ソースは近々 Magicユーザーズフォーラム にて公開いたしま"せん"。

なんせ機器の相性問題などもあるので；でも興味がありましたら担当営業まで！



ありがとうございました。

疑問点などはフォーラムや担当営業まで！

