

情報種別 : 公開
会社名 : (株)NTTデータ
情報所有者 : 技術革新統括本部

NTT Data
Trusted Global Innovator

JaSST'21 Tokyo

Excel方眼紙を廃し、仕様書のレビュー効率・品質を向上した事例

株式会社NTTデータ 技術革新統括本部 システム技術本部
生産技術部 ソフトウェア技術センタ
熊川 一平

熊川 一平 (くまがわ いっぺい)

株式会社NTTデータ 技術革新統括本部 システム技術本部 生産技術部 ソフトウェア技術センタ
テクニカルグレード イノベータ(ソフトウェアプロセス)

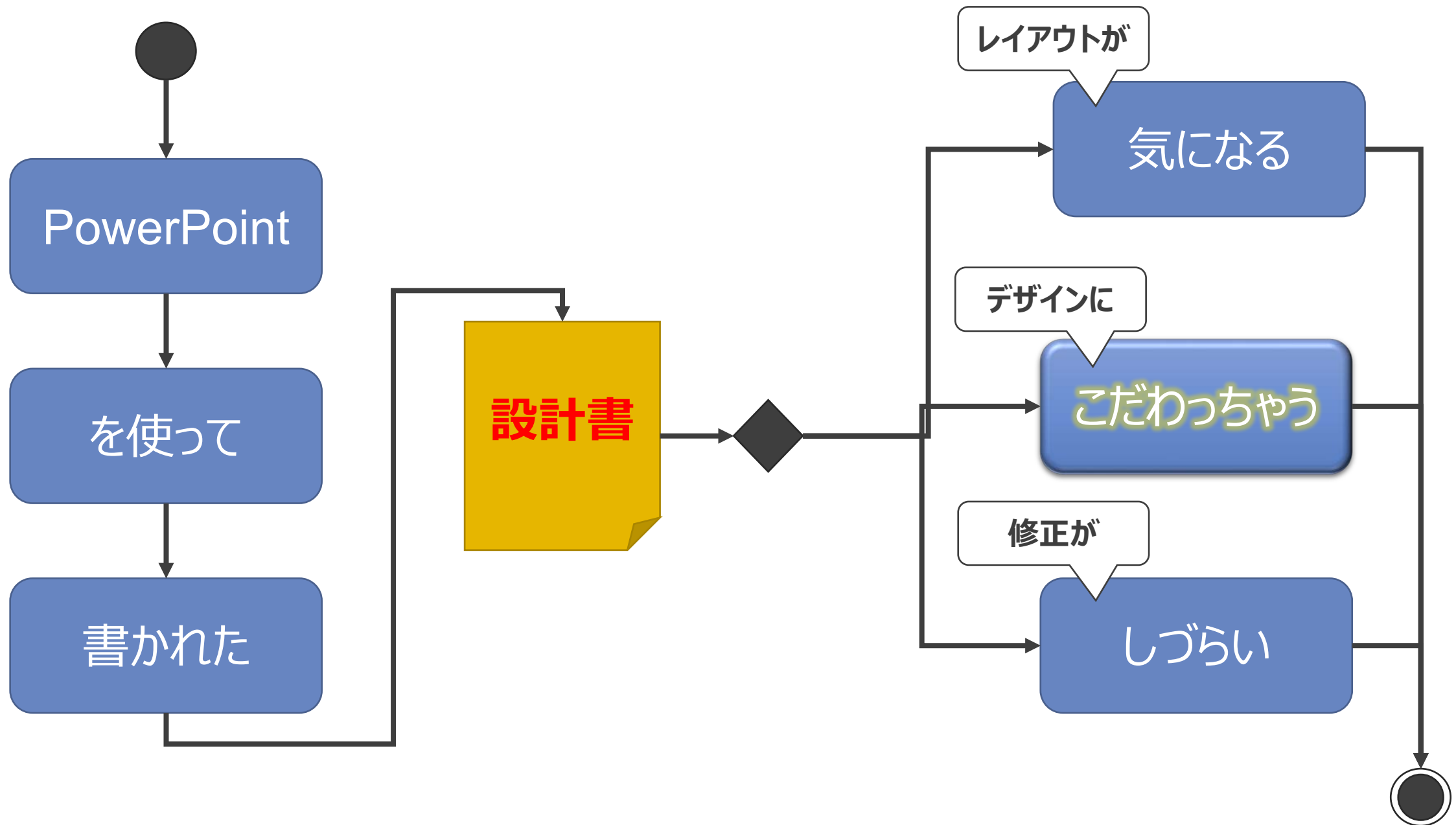
- ソフトウェアテスト/ソフトウェア品質保証を中心としたR&D活動と、現場適用支援に従事

【主な講演歴】

- JaSST'14 Tokyo ベストスピーカー賞
- JaSST'19 Tokyo ベストスピーカー賞
- ソフトウェア品質シンポジウム2017 SQiP Best Report Effective Award
- ソフトウェア品質シンポジウム2019 SQiP Best Paper Effective Award

こんなことに苦しんでいませんか？

[illegible]



メンテされない大量の設計書

設計書_20200122版.xlsx.old

図を描くにはキャンバスが狭い

修正の衝突

印刷レイアウトの罭

もはや職人

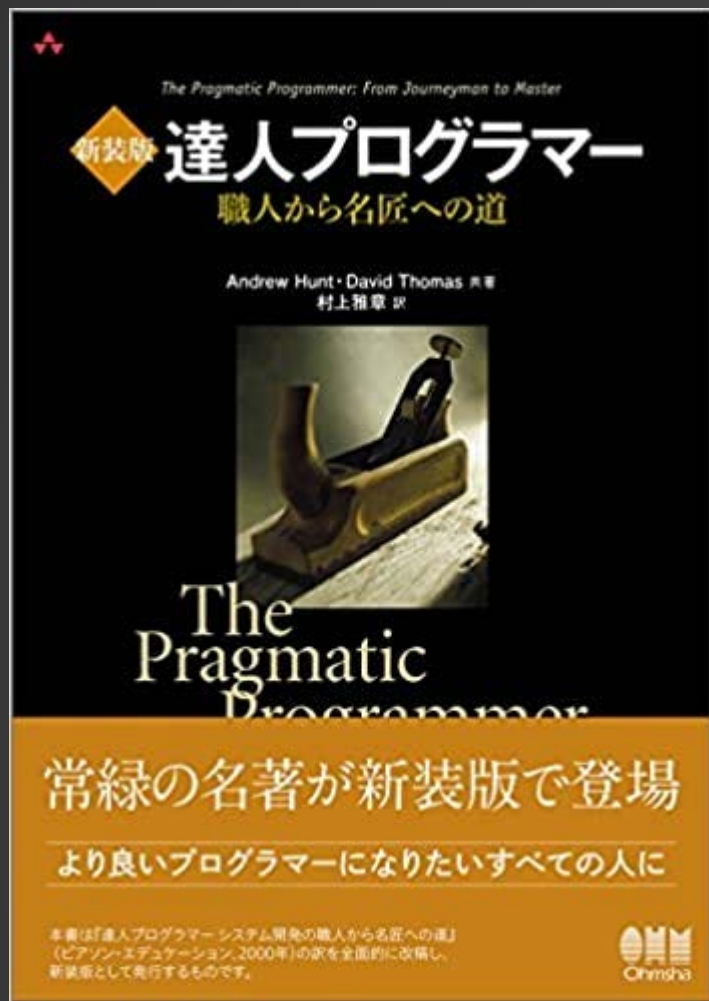
埋めとけば設計した気になる

オートシェイプの自由さが産んだ謎の図

問題は 山積み

A stylized illustration in the background shows a person with dark hair and a blue shirt, looking down at a massive, towering pile of papers and documents. The papers are depicted with various patterns and colors, suggesting a complex and overwhelming amount of information. The person's expression is one of stress or frustration, with a red mark on their forehead. The entire scene is set against a dark gray background.

プレーンテキストのススメ



達人プログラマー 職人から名匠への道

Andrew Hunt (著), David Thomas (著), 村上雅章 (翻訳)

第3章 プレイン・テキストの威力

- 達人プログラマーが取り扱う情報は、知識です。その知識を永続的に格納するためのフォーマットで最も適しているものが、プレインテキストです。
- プレイン・テキストのメリットは、透明性が保証される、様々な活用ができる、テストが用意になる。
- ソースコード管理システムは巨大なUNDOキー。すべてをソースコード管理システムで管理すること。それがソースコードでなくても。
- 実効可能ドキュメント、ドキュメントからコードを生成する。ドキュメントとコードの2つをメンテナンスすることはDRY原則に反する。プレインテキストならば、スクリプト言語によって加工が可能。



Excel

jasst2021.md

C: > Users > 3258936 > Desktop > jasst2021.md > ### プレーンテキストの持つチカラ

```

1  ### プレーンテキストの持つチカラ
2  1. テキスト比較ができる
3  1. 検索が容易
4  1. 2次利用しやすい
5  1. Gitと相性がいい
6

```

Preview jasst2021.md

プレーンテキストの持つチカラ

- 1. テキスト比較ができる
- 2. 検索が容易
- 3. 2次利用しやすい
- 4. Gitと相性がいい

ファイル(F)編集(E)選択(S)表示(V)移動(G)実行(R)ターミナル(T)ヘルプ(H)

jasst2021.md

C: > Users > 3258936 > Desktop > jasst2021.md > # MarkdownやAsciiDocを使えば

1# MarkdownやAsciiDocを使えば

2

3**文書執筆に必要な装飾は大体できる**

4

5* 箇条書きも

6* 簡単だし

7* ネストも掘れる

8

9

表にすることも	可能です
執筆	わずらわしいことはない
レビュー	テキスト差分をとって簡単に
再利用	テキストだから検索も構造化も簡単

10

11

12

13

14

Preview jasst2021.md

MarkdownやAsciiDocを使えば

文書執筆に必要な装飾は大体できる

- 箇条書きも
- 簡単だし
 - ネストも掘れる

表にすることも	可能です
執筆	わずらわしいことはない
レビュー	テキスト差分をとって簡単に
再利用	テキストだから検索も構造化も簡単

0 0 Table Mode: Off

行 9、列 50 スペース: 4 UTF-8 CRLF Markdown Prettier

マージ済み

Opened 2ヶ月前 by



KUMAGAWA IPPEI



Maintainer

Edit

フライトの最大予約数を2から3に修正しました

概要 1 コミット 1 変更 1

All threads resolved



Compare

develop

and

最新バージョン

1 ファイル +1 -1



Search files (Ctrl+P)

content/ja/docs/01.ビジネスルール定義

M+ ビジネスルール定義_予約規定... +1 -1

content/ja/docs/01.ビジネスルール定義/ビジネスルール定義_予約規定.md

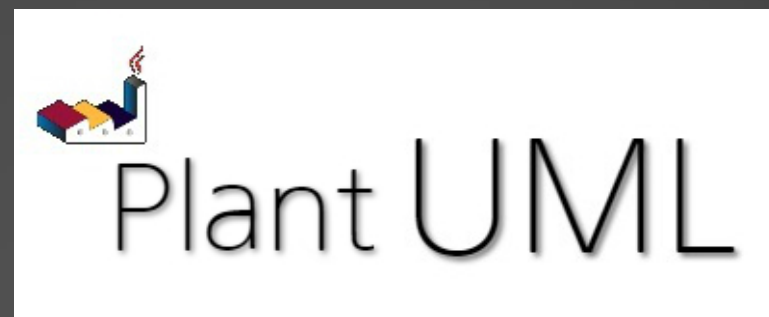
+1 -1



Show all unchanged lines Show 20 lines

17	17	
18	18	## 1チケットで予約できるフライトの数について
19	19	
20		- 1つのチケットで予約できるフライトの数は最大2フライト。また、2フライトを予約する場合は、往復便であること。往復便でない場合は1フライトごとにチケットを予約する必要がある。
	20	+ 1つのチケットで予約できるフライトの数は最大3フライト。また、2フライトを予約する場合は、往復便であること。往復便でない場合は1フライトごとにチケットを予約する必要がある。
21	21	
22	22	## 往復便で選択できる運賃種別について
23	23	

Show 20 lines Show all unchanged lines



ファイル(E)編集(E)選択(S)表示(V)移動(G)実行(R)ターミナル(T)ヘルプ(H)

jasst2021.md

C: > Users > 3258936 > Desktop > jasst2021.md > {} jasst2021

```
1  ``plantuml
2  @startuml
3  start
4  :ソフトウェア開発に必要な図だって;
5  if (use plantUML?) then (yes)
6  :plantUMLを使って書ける;
7  else (no)
8  :Mermaidjsを使って書ける;
9  endif
10 end
11 @enduml
12 ````
```

Preview jasst2021.md

```
graph TD; Start(( )) --> Process[ソフトウェア開発に必要な図だって]; Process --> Decision{use plantUML?}; Decision -- yes --> YesProcess[plantUMLを使って書ける]; Decision -- no --> NoProcess[Mermaidjsを使って書ける]; YesProcess --> Join{ }; NoProcess --> Join; Join --> End(((X)))
```

⊗ 0 △ 0 Table Mode: Off

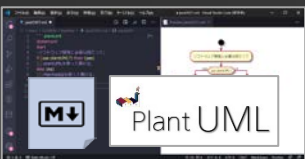
行 10、列 4 スペース: 4 UTF-8 CRLF Markdown Prettier



- ビジネスルール定義
- 業務フロー図
- ユースケース図
- 画面レイアウト
- 帳票レイアウト
- 画面設計書
- 帳票設計書
- 画面遷移図
- BBB業務
- チケット予約業務
- チケットを予約する
- 空席状況を照会する
- DDL
- 外部インターフェース一覧
- 処理方式パターン
- 処理設計
- セッション情報
- ジョブネット定義体
- メッセージ定義体
- 開発ガイドライン
- 共通部品仕様
- 中間成果物

```
graph TD
    Start(( )) --> UC1[航空チケット検索画面]
    UC1 -- "予約を申し込む[エラーの場合]" --> UC1
    UC1 -- "予約を申し込む[ログイン未の場合]" --> UC2[予約方法選択画面]
    UC1 -- "予約を申し込む[ログイン済の場合]" --> UC4[予約内容入力画面]
    UC2 -- "ログインして予約する" --> UC3[ログイン画面]
    UC2 -- "ログインせずに予約する" --> UC4
    UC3 -- "【共通業務】予約入力のためログインする" --> UC4
    UC4 -- "予約入力内容を確認する[エラーの場合]" --> UC4
    UC4 -- "予約入力内容を確認する[正常の場合]" --> UC5[予約内容確認画面]
    UC5 -- "予約登録を確定する[エラーの場合]" --> UC4
    UC5 -- "予約入力へ戻る" --> UC4
    UC5 -- "予約登録を確定する[正常の場合]" --> End(( ))
```

The diagram illustrates the flow of a flight ticket reservation system. It begins with a start node leading to the '航空チケット検索画面' (Flight Ticket Search Screen). From here, users can search for tickets. If there's an error, they stay on the search screen. If not logged in, they go to the '予約方法選択画面' (Reservation Method Selection Screen). If already logged in, they proceed to the '予約内容入力画面' (Reservation Content Input Screen). From the selection screen, users can log in to make a reservation or proceed without logging in. The login screen leads to the input screen for common business cases. The input screen has a self-loop for error confirmation and a path to the '予約内容確認画面' (Reservation Content Confirmation Screen) for normal cases. From the confirmation screen, users can return to the input screen or finalize the reservation, leading to the end node.



執筆



設計担当者

Pull-request



GitLab



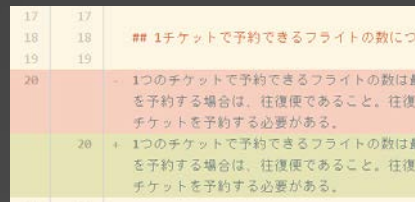
Markdown
plantUML



GitLab Pages

閲覧・利用

レビュー・マージ



レビュー

GitLab CI

静的Webサイト
生成エンジン



DOCSY

全文検索はLunr.js



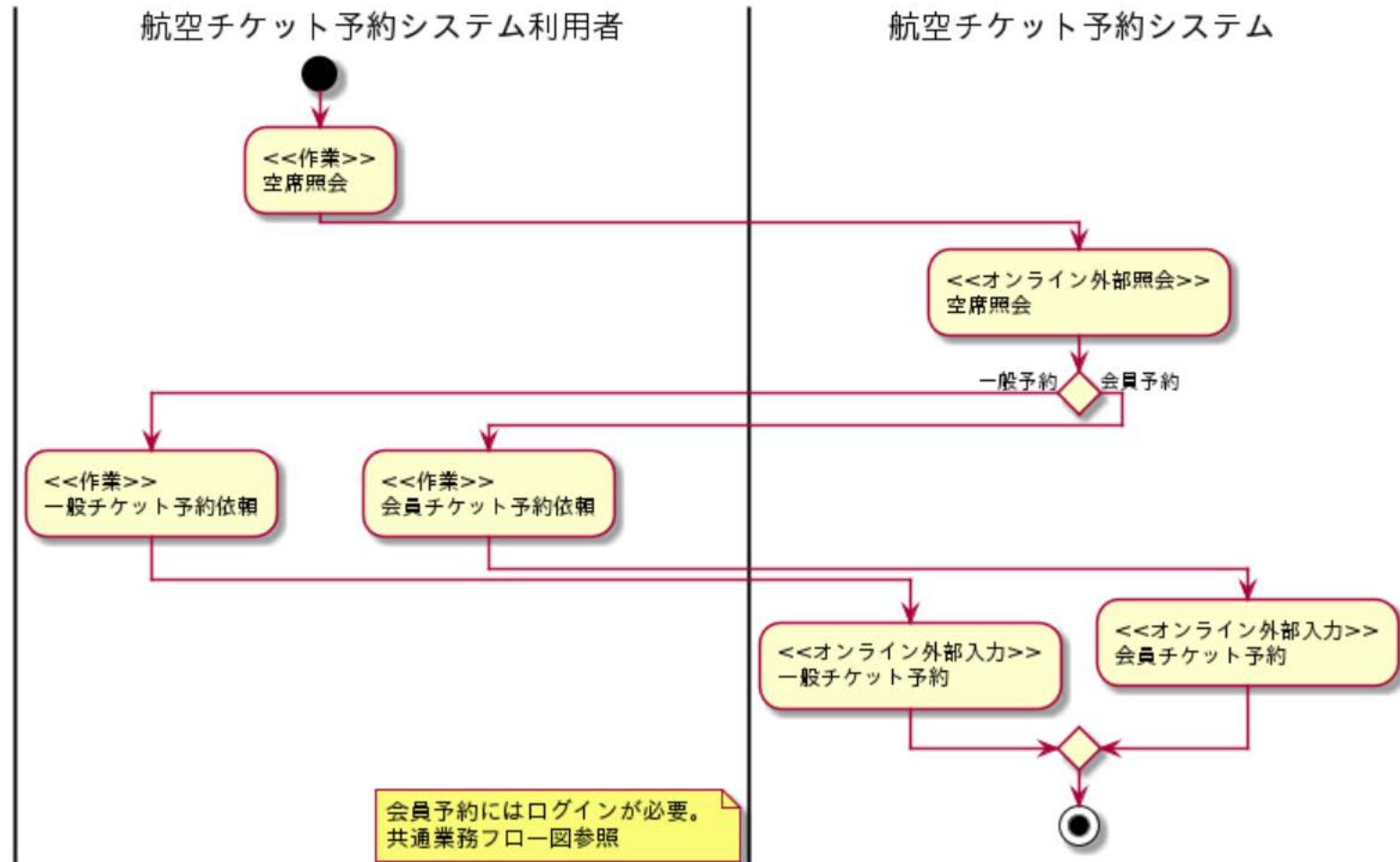
Excel方眼紙をやめ、
Gitに載せることで
効率的なレビューを
ワークフローに組み込む

実際の設計書の例

業務フロー図 チケット予約業務

- 概要

航空チケット予約システム利用者が空席状況照会、チケット予約を行う一連の業務



ビジネスルール定義 予約規定

チケットの支払い期限について

往路の搭乗日とする。

復路の搭乗可能開始時期について

- 復路のフライトは往路のフライトの到着時刻より 2 時間以上経過した出発時刻のフライトから搭乗可能となる。

例) 往路が 10月1日10時00分 到着の場合、復路は 10月1日12時00分 出発のフライトから搭乗可能。

1チケットで予約できるフライトの数について

1つのチケットで予約できるフライトの数は最大2フライト。また、2フライトを予約する場合は、往復便であるこ

チケットの支払い期限について

往路の搭乗日とする。

復路の搭乗可能開始時期について

- 復路のフライトは往路のフライトの到着時刻より 2 時間以上経過した出発時刻のフライトから搭乗可能となる。

例) 往路が 10月1日10時00分 到着の場合、復路は 10月1日12時00分 出発のフライトから搭乗可能。

1チケットで予約できるフライトの数について

1つのチケットで予約できるフライトの数は最大2フライト。また、2フライトを予約する場合は、往復便であること。往復便でない場合は1フライトごとにチケットを予約する必要がある。

往復便で選択できる運賃種別について

往復便を予約する場合は、一般席の往復運賃、特別席の特別往復運賃のどちらかであること。

往復便で空席数が不足している場合について

往路の空席照会からやり直す必要がある。往路のみ、復路のみの変更は許容しない。

詳細予約規定について

搭乗者、予約代表者は以下の条件を満たす必要がある。

- 予約代表者の年齢は18歳以上であること。
- 運賃種別がレディース割の場合、搭乗者の性別はすべて女性であること。
- 運賃種別がグループ割の場合、搭乗者の合計数が利用可能最少人数以上であること。

予約に必要な情報について

- 予約に必要な情報は、搭乗者の氏名、年齢、性別、および予約代表者の氏名、年齢、性別、電話番号、メールアドレスとする。
- 搭乗者以外が予約代表者となることも可能とする。

導入効果・導入障壁

設計書の執筆に関する分析

表. フォーマットごとの執筆に要した時間

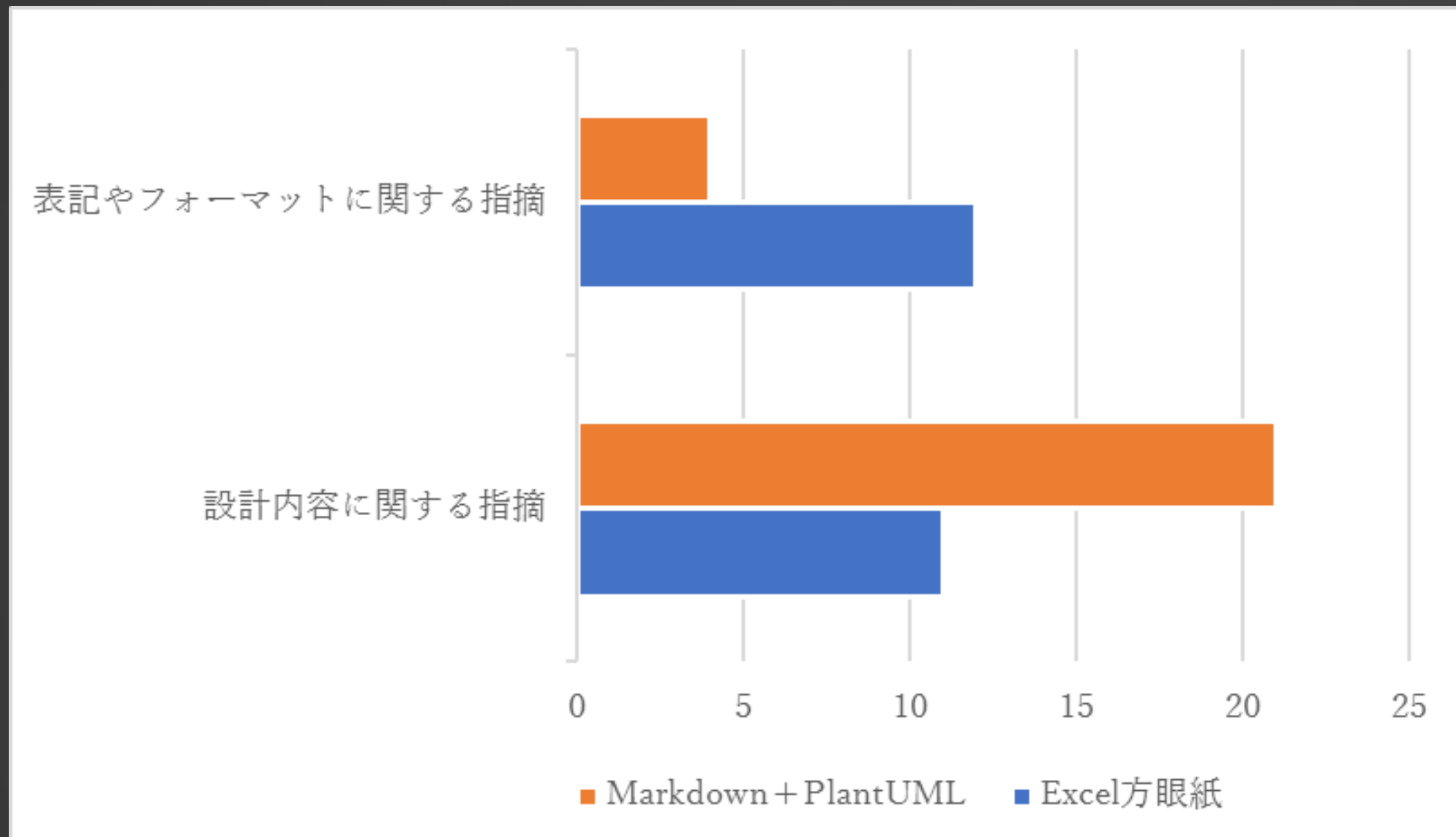
ドキュメント種別	執筆に要した時間（分）	
	Excel 方眼紙	Markdown + PlantUML
ビジネスルール定義	49.8	49.8
業務フロー図	64.8	45
ユースケース図	109.8	49.8
画面遷移図	600	420
処理設計	5175	2400
計	5999.4	2964.6

設計書のレビューに関する分析

表. フォーマットごとのレビューに要した時間

ドキュメント種別	レビューに要した時間（分）	
	Excel 方眼紙	Markdown + PlantUML
ビジネスルール定義	34.8	30
業務フロー図	34.8	30
ユースケース図	145.2	220.2
画面遷移図	180	300
処理設計	1950	525
計	2344.8	1105.2

設計書のレビューに関する分析



導入障壁

- 学習コスト
 - markdownもplantUMLも難しいものではありません。
 - 始めてWordやExcelを使い始めたとき、少し戸惑いがあったかもしれませんが、ちょっとググれば使えましたよね？あの程度の学習で大丈夫です。
- 納品フォーマット
 - MarkdownもplantUMLもオープンなフォーマットなので、そのままでもいいはず。
 - hugoやpandoc,gitbookなど種々の技術により、pdfやhtmlなど様々なものに変換可能です。
- 既存の設計書の流用
 - 難しいです。Word→markdownなどの変換ツールはありますが、そう綺麗に変換されません。
 - また、UMLでないオリジナル設計図などは当然扱いが難しいです（ガラパゴス化）。

• 変化を恐れる心

- これが一番の導入障壁です。
- いいじゃん、いまはExcelで出来てるんだから。
- 新しいことを覚えるの面倒、新しいことをお客様に提案するの大変。
- 私はいいと思うんですが、上司が……お客様が……。

• 変化を恐れる心

- これが一番の導入障壁です。
- いいじゃん、いまはExcelで出来てるんだから。
- 新しいことを覚えるの面倒、新しいことをお客様に提案するの大変。
- 私はいいと思うんですが、上司が……お客様が……。

まずは使ってみましょう。

いきなり全ての設計書を移行させる必要などないのです。

簡単なドキュメントなどをmarkdownやplantUMLで書いてみて、それをGitで管理してみてください。

その輪に少しずついろいろな人を巻き込んでいきましょう。

そしてフォロワーを見つけて、その人を大事にしましょう。

まとめ

メンテされない大量の設計書

設計書_20200122版.xlsx.old

図を描くにはキャンバスが狭い

修正の衝突

印刷レイアウトの罫

もはや職人

埋めとけば設計した気になる

オートシェイプの自由さが産んだ謎の図

いつまでExcelで設計する？
このままで本当にいいの？

勇気をもって変化を。



ExcelならびにPowerPointは、米国 Microsoft Corporation の米国及びその他の国における登録商標、商標または商品名称です。
その他、記載されている会社名、製品名等は、各社の登録商標または商標です。