



Next.js App Router を例に考える、 技術選定・技術との距離感

2024-01-24 フロントエンドの技術選定 ～2023を振り返る～ Lunch LT

@izumin5210

今日話すこと

新プロダクト(バクラク請求書発行)での技術選定を例に、
技術選定、とくにフレームワークの選定で何を考えたかを振り返る

@izumin5210

- ▶ Wantedly, Inc. (2018-04 - 2022-08)
- ▶ LayerX (2022-09-)
 - バクラク事業部 Enabling Team
 - Backend と Web Frontend 中心にやっています

- ▶ 最近のお気に入りパッケージは
`@eddeeee888/gcg-typescript-resolver-files`



新規プロダクト: バクラク請求書発行

- ▶ バクラクシリーズ 5番目のプロダクト
 - 企業が扱う書類を作成できる
- ▶ エンジニア 3名
 - PdM 兼 TechLead
 - ソフトウェアエンジニア 2名 ※片方が@izumin5210
- ▶ 開発開始からリリースまで
 - 短い!

The screenshot displays the LayerX software interface for creating a request form. The interface includes a sidebar with document types, a main editing area with a form, and a summary table at the bottom.

請求書

2023年09月01日
書類番号: IN202309-0001

〒000-0000
△△県△△市
△△町△△ サンプルビル2F

バクラク
株式会社LayerX
〒103-0012
東京都中央区日本橋蛸壷町
1丁目9-8
人形町PREX 2F
電話: 0300000000
FAX: 03-0000-0000
担当者:
E-mail: dx-members@layerx.co.jp
登録番号: T0123456789012

株式会社△△様
営業部 担当 花子様

下記のとおり、御請求申し上げます。

ご請求金額 **¥3,280** (税込)

振込先 みずほ銀行 東京営業部 普通 01234567
三井住友銀行 神保町拠点 普通 01234567

お支払い期限 2023年9月30日

日付	品名	数量	単価	金額
2023年9月1日	ねじ ※	3個	100	300

※は軽減税率対象であることを示します。

合計(税抜)	¥3,000
消費税	¥280
10%対象: ¥2,000	消費税: ¥200
8%対象: ¥1,000	消費税: ¥80
源泉徴収額	¥-0

技術的に決まっていたこと

- ▶ Web Frontend は React に寄せていく
- ▶ 今後 React でプロダクトを作っていくための“型”を、このタイミングで整備する
 - 標準の技術選定, 共通実装(将来的にライブラリになるもの), etc.

以上！あとは全部決めなきゃいけない！



Webフレームワークなど、一番外側の仕組みをどうするか決めたい

少なくともビルドとルーティングをどうするか決めないと動きづらいはず

Next.js, Remix, 何かを組みわせて自分で頑張る, etc.

みなさんならどう考えて、どう決めますか？

開発スタート時の意思決定(Webフレームワークまわり)

やる

- ▶ Next.js を採用！
- ▶ App Router を使う！

やらない

- ▶ App Router への深入りはしない
App Router は一歩引いて使う。
Server Componentsはごく一部だけ。
データ取得も基本クライアント。
キャッシュや高度なルーティングなど、
複雑な機能や Next.js 独自の機能は
なるべく使わない。

Why Next.js?

- ▶ 前提: 何のために Web フレームワークを導入するのか
 - 今回は「ルーティング機構とそれに最適化されたビルドシステム」がほしい
- ▶ Next.js にはルーティング機構, ビルドまわりなど、よく作り込まれたものが用意されている
 - 今回の Web フレームワーク導入目的にも合致する
 - 同じレベルで自作するとそれなりに手間はかかる
- ▶ Q. Next.js は学習コスト高くない?
 - A. RSC, SSR などに深入りせず、単純な File-system based router としてみれば見かけ上はシンプル
- ▶ 世の中で広く使われているので、困っても情報が見つかりやすいはず

Why App Router?

- ▶ App Router (React Server Components) がもたらす Web 開発の変化への期待と投資
 - Bet Technology
 - 前回の発表: [React Server Components](#) で複雑さに立ち向かう *
 - フルベットはしないけど、利用箇所を限定しながら試す
 - 実アプリで試さないと見えてこないこともある
- ▶ いずれ App Router がメインストリームになるのであれば、早めに移行できるに越したことはない
 - プロダクトが大きくなってからの移行は大変になりがち

意思決定時に重要視したこと (@バクラク請求書発行)

追加学習コストを
下げる

Bet Technology,
but 撤退可能



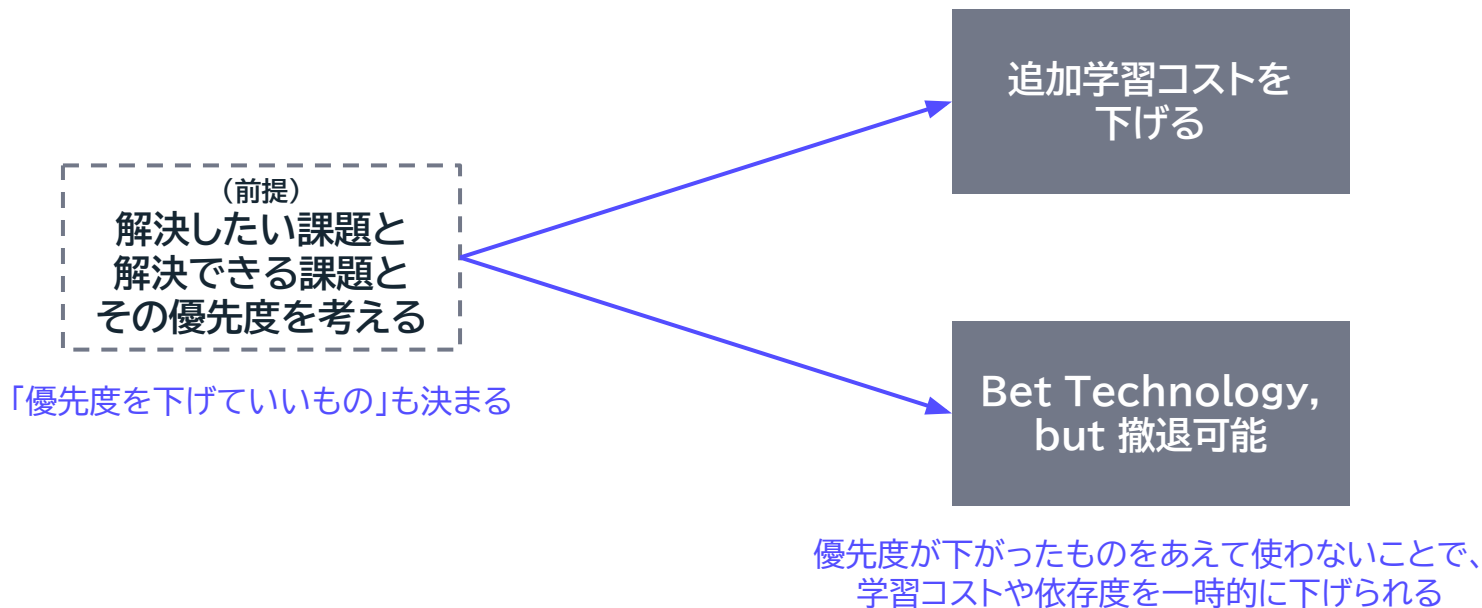
?

意思決定時に重要視したこと (@バクラク請求書発行)

Bet Technology,
but 撤退可能

- ▶ この分野(Next.js, Remix など React ベースの Web フレームワーク)は
発展の真っ最中
 - 👍 まだまだ良くなっていきそう
 - 🙄 App Router が出たばかりすぎてエコシステムも追従しきれてない
 - どれも決定的な差はなく、「いまは絶対こっち！」とは言えない
 - もちろんコンテキストによるところはある
 - 情勢の変化に追従しやすく(= 撤退しやすく)しておきたい

意思決定時に重要視したこと (@バクラク請求書発行)



意思決定時に重要視したこと（@バクラク請求書発行）

これらは「初期は」「一時的に」でもある
アクセルを思いっきり踏んでいい と判断できれば、
ガッツリ使っていけばいい

追加学習コストを
下げる

必要な学習コストを時間的に分散させるために
初期コストを下げている とも言える

Bet Technology,
but 撤退可能

技術選定には理由が伴う

- ▶ 何のためにその技術を使うのか、ちゃんと考える
 - ○○がすごい！ もいいけど、それがプロダクト・チームの今や未来になぜ必要か も重要
 - 今回は「ルーティングとそれに最適化されたビルドシステム」が要求で、逆に言うとそれ以外は極論なくてもいい
 - 「何のため」が決まれば、逆に優先度を下げても良い要素も見つけられる
 - これは「ずっと使わない」ということではない
 - 「将来的に使えると(ユーザ|開発者)体験が良くなりそうだけど、初期は控えめにしておく」みたいな話

距離を取るのも選択肢

- ▶ 決定的な差がない場合、「**適度に距離を取りつつ技術と付き合う**」というのも選択肢になる
 - 決定的な差 is …
 - 「〇〇じゃないと実現できない機能があり、それがプロダクトにとって重要」
 - 「〇〇が他の技術と比べ △△という問題をうまく解いており、それがプロダクトやチームにとって重要」
 - etc.
 - 入れる or 入れない の2択ではなく、**濃淡**もありうる
 - 今使わないからずっと使わない ではない
 - コストやリスクを時間的に分散させているだけ
 - Web フレームワークなど、「**ベッタリ依存するとコードのあらゆるところに染みるタイプの技術**」ではとくに重要

とはいえ Bet したい

- ▶ 撤退可能なパスを残しつつ、多少は Bet Technology のための余白を残しておくといい
 - やっぱり未来を感じる技術に触れるのは楽しい！ し、やってみないとわからないことはある
 - 見えてなかった大きなメリットがあるかも
 - 自分たちのプロダクトへのフィット感がわかるかも
 - あとで剥がせるようにする・人に迷惑をかけないのは前提で、Bet する余白を作っておくといいかも
 - バクラク請求書発行では自分(@izumin5210)が完全にコントロールできるところだけ Server Components を使ってサーバサイドでフェッチして… やってました

まとめ

- ▶ 何のためにその技術を使うのか、ちゃんと考える
 - ○○がすごい！ もいいけど、それがプロダクト・チームの今や未来になぜ必要か も重要
- ▶ 決定的な差がない場合、「**適度に距離を取りつつ技術と付き合う**」というのも選択肢になる
 - 入れる or 入れない の2択ではなく、濃淡もありうる
- ▶ 撤退可能なパスを残しつつ、多少は Bet Technology のための**余白**を残しておくといい
 - やっぱり未来を感じる技術に触れるのは楽しい！ し、やってみないとわからないことはある
 - あとで剥がせるようにする・人に迷惑をかけないのは前提で、Bet する余白を作っておくといいかも

で、プロダクトはどうなった？

- ▶ ちゃんと問題なくリリースできた！
- ▶ が、リリース直前に App Router から撤退し、Pages Router へ移行している
 - App Router だと必要な機能が実装できなかったため
 - navigation のキャンセルが難しかった
 - (Pages Router は怪しい感じだけど一応できる)
 - 深入りしてなかったので移行自体はすぐに完了
 - 撤退しやすいようにしててよかった…
 - App Router に戻れないか、チャンスを伺い中…

まとめ

- ▶ 何のためにその技術を使うのか、ちゃんと考える
 - ○○がすごい！ もいいけど、それがプロダクト・チームの今や未来になぜ必要か も重要
- ▶ 決定的な差がない場合、「**適度に距離を取りつつ技術と付き合う**」というのも選択肢になる
 - 入れる or 入れない の2択ではなく、濃淡もありうる
- ▶ 撤退可能なパスを残しつつ、多少は Bet Technology のための**余白**を残しておくといい
 - やっぱり未来を感じる技術に触れるのは楽しい！ し、やってみないとわからないことはある
 - あとで剥がせるようにする・人に迷惑をかけないのは前提で、Bet する余白を作っておくといいかも



※ 前回





- ▶ LayerX OpenDoor [検索]



- ▶ バクラクフロントエンドVision [検索]