

Atualizando store-theme | app vtex

Antes de tudo, independente se é app | store-theme, é necessario:

1. branch master
2. workspace master
3. pull na branch master
4. Se certificar que todos os apps de peerdependecies estão instalados na master, caso não esteja, instalar e atualizar major
5. Se certificar que todos os apps de peerdependecies não foram alterados, caso tenha sido alterado, atualizar major também

Caso seja app:

1. vtex release patch stable
2. vtex publish
3. vtex deploy
4. vtex install

Caso seja store-theme:

1. verificar se todos os peerdependecies estão instalados
2. vtex setup && yarn run dev:sass na master, e dar commit logo em seguida
3. vtex release patch stable
4. vtex deploy --force
5. vtex install

Fazendo update de major

Antes de tudo, se certificar que:

1. vtex.admin-graphql-ide@3.x está instalado na master
2. workspace master
3. branch master

Depois de se certificar, fazer o seguinte:

1. vtex use wpmajor --production - cria um workspace de produção
2. instale o app que foi inserido no peerdependecies - exemplo: "yourviews.yourviews": "0.x"
3. vtex setup && yarn run dev:sass
4. vtex release major stable
5. vtex deploy
6. vtex install {nome-da-loja}.store-theme@{versão-major} - versão nova que vai ser gerada
7. ir até o app do graphql presente no admin e procurar por vtex.pages-graphql@2.x
8. dentro do app, fazer a seguinte mutation exemplo:

```
//Exemplo store theme
mutation{
  updateThemeIds(from:"{nome-da-loja}.store-theme@0.x", to:"{nome-da-loja}.store-theme@1.x")
}

//Exemplo genérico
mutation{
  updateThemeIds(from:"{appvendor}.{appname}@{oldmajor:1}.x", to:"{appvendor}.{appname}@{newmajor:2}.x") // as vezes é preciso pular mais de uma versão se dar erro na tag no release major
}
```

9. Verificar se o novo workspace esta com todo o conteudo da loja
10. vtex workspace promote

Instalando QA

vou utilizar como exemplo a vendor lojasmabraz que esta no ar.

É importante se certificar que:

- qual será a gtm usada na loja qa
- vtex.search esta instalado na master
- se tiver app de labs, avisar gp
- A branch vai ser protegida

Passos

1. crie uma branch qa/develop
2. entre na vendor do qa
3. mude todas as referencias da vendor atual para a vendor do qa
4. dentro da branch git push --set-upstream origin qa/develop
5. publicar todos os apps do peerdependencies
6. instalar todos os apps do peerdependencies dentro da store-theme
7. ativar reindexação de busca
8. vtex setup && yarn run dev:sass
9. vtex release patch stable
10. vtex deploy --force
11. vtex install

Anotações

Passo 1

git checkout -b qa/develop

Passo 2

vtex login lojasmarabrazqa

Passo 3

Em peerdependencies da store-theme:

Se conformar que a versão da loja não é relevante. Exemplo: A loja duplicada terá a versão 3.8.31.

Antes:

```
"vendor": "lojasmarabraz",
"name": "store-theme",
"version": "3.0.31",
...
code
...
"lojasmarabraz.avanti-menu": "0.x",
"lojasmarabraz.cookies-modal": "0.x",
"lojasmarabraz.shipping-minicart": "0.x"
```

Depois:

```
"vendor": "lojasmarabrazqa",
"name": "store-theme",
"version": "3.0.31",
...
code
...
"lojasmarabrazqa.avanti-menu": "0.x",
"lojasmarabrazqa.cookies-modal": "0.x",
"lojasmarabrazqa.shipping-minicart": "0.x"
```

Passo 5

Vai entrar em cada app e rodar os seguintes comandos:

```
vtex setup
vtex release patch stable
vtex publish
vtex deploy --force
vtex install
```

Passo 6

Os apps exemplos são:

```
Installed Apps in lojasmarabrazqa at workspace master
avantivtexio.buytogether          0.3.5
avantivtexio.ourstores            3.6.2
lojasmarabrazqa.avanti-menu       0.0.19
lojasmarabrazqa.cookies-modal     0.0.6
lojasmarabrazqa.shipping-minicart 0.0.20
lojasmarabrazqa.store-theme       3.0.38
vtex.google-search-console        1.0.3
vtex.google-tag-manager           3.3.1
vtex.reviews-and-ratings          3.12.1
vtex.search                       2.14.0
vtex.search-page-context          0.2.0
vtex.search-result                3.118.21
vtex.social-selling               0.7.19
vtex.wish-list                    1.15.4
```

Passo 7

vá em /admin > configurações de busca > reindexação e ative

prevenção

1. Fazer release em outro ws. Baixar os apps e publica-los
2. após fazer o publish e install dos apps, ir para store-theme, deslinkar tudo e seguir com o publish/install dela
3. Quando tudo baixado e publicado, mudar para o ws em produção e só instalar os apps publicados

erros conhecidos

erros no install:

erro 1

```
PS C:\Users\maska\Desktop\projetos\Avanti\vtexio-kopenhagen\avanti-kop-
middlewares> vtex install
10:36:05.205 - error: Installation failed!
10:36:05.207 - error: App has route conflict in:
Route: "/_v/postFC/:p3". Declarers: kopenhagen21.avanti-general-
middleware@5.0.0,kopenhagen21.avanti-general-middleware@0.0.1
Route: "/_v/status/:p3". Declarers: kopenhagen21.avanti-general-
middleware@5.0.0,kopenhagen21.avanti-general-middleware@0.0.1
If you are sure you want to install this app, use the parameter '--force' in the
installation command
PS C:\Users\maska\Desktop\projetos\Avanti\vtexio-kopenhagen\avanti-kop-
middlewares> vtex install kopenhagen21.avanti-general-middleware@5.0.0
```

```
10:36:33.016 - error: Installation failed!
10:36:33.018 - error: App has route conflict in:
Route: "/_v/postFC/:p3". Declarers: kopenhagen21.avanti-general-
middleware@5.0.0,kopenhagen21.avanti-general-middleware@0.0.1
Route: "/_v/status/:p3". Declarers: kopenhagen21.avanti-general-
middleware@5.0.0,kopenhagen21.avanti-general-middleware@0.0.1
```

1. /store-theme - atualize o manifest com a versão que o app vai pegar antes:

```
"dependencies": {
  "kopenhagen21.avanti-general-middleware": "0.x"
}
```

depois:

```
"dependencies": {
  "kopenhagen21.avanti-general-middleware": "5.x"
}
```

2. /store-theme - vtex setup

3. /avanti-general-middleware - tente instalar novamente o app