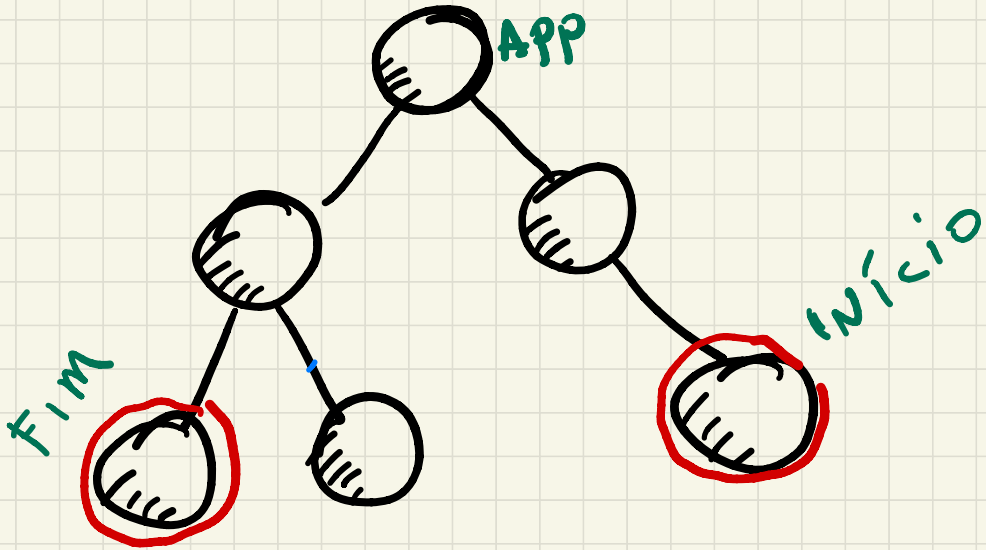
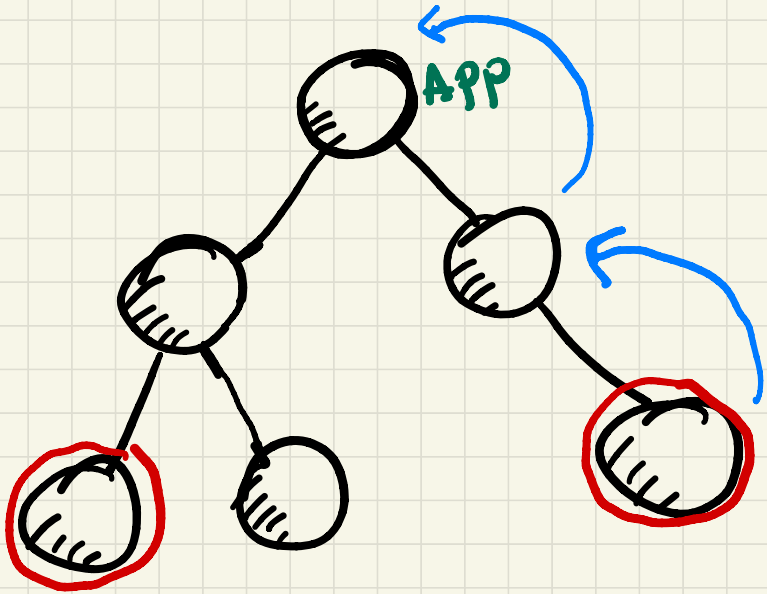
REDUX

- IMPLEMENTAÇÃO DO FLUX
- COMPARTILHMENTO DE ESTADO ENTRE TODOS OS COMPONENTES DE UMA APLICAÇÃO
- OBEDECE UM CICLO DE VIDA FORMADO POR: AÇÕES, ESTADOS E REDUTORES

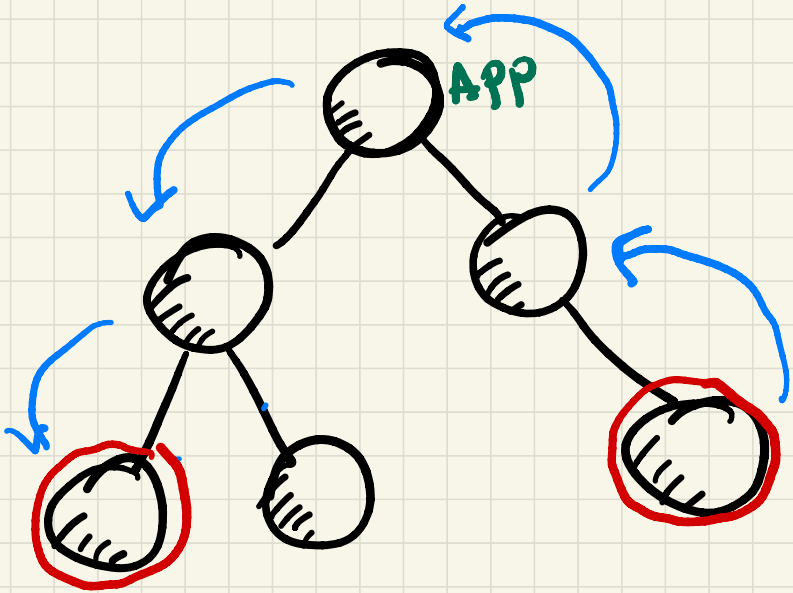
APLIÇÃO SEM REDUX



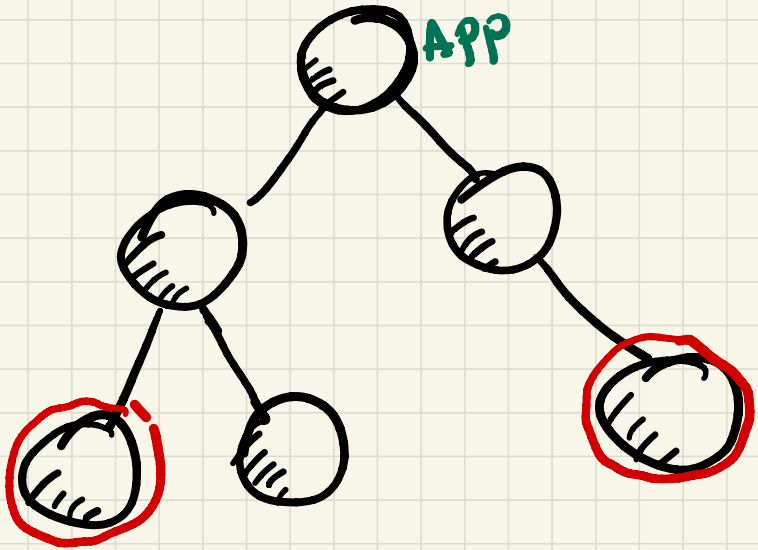
APLIÇÃO SEM REDUX



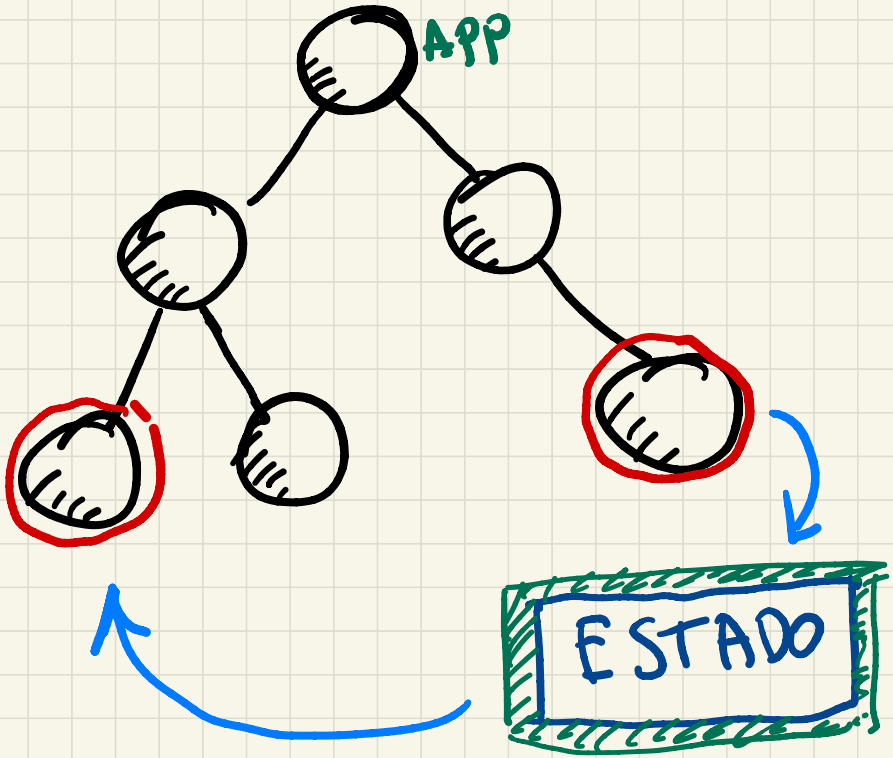
APLIÇÃO SEM REDUX



APLIÇÃO COM REDUX



APLIÇÃO COM REDUX



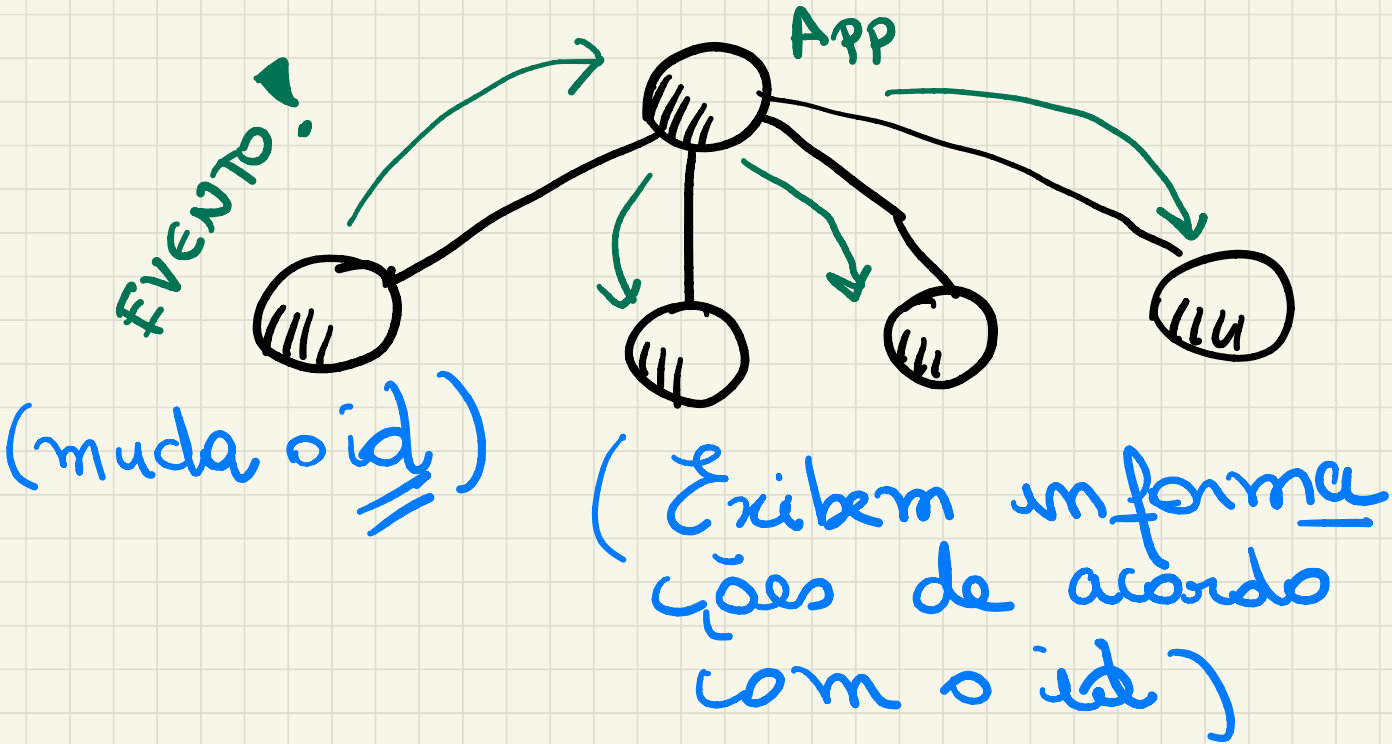
OBJETIVOS DO CURSO

* IMPLEMENTAR UMA APLICAÇÃO **SEM** REDUX, COM COMUNICAÇÃO DIRETA ENTRE OS COMPONENTES.

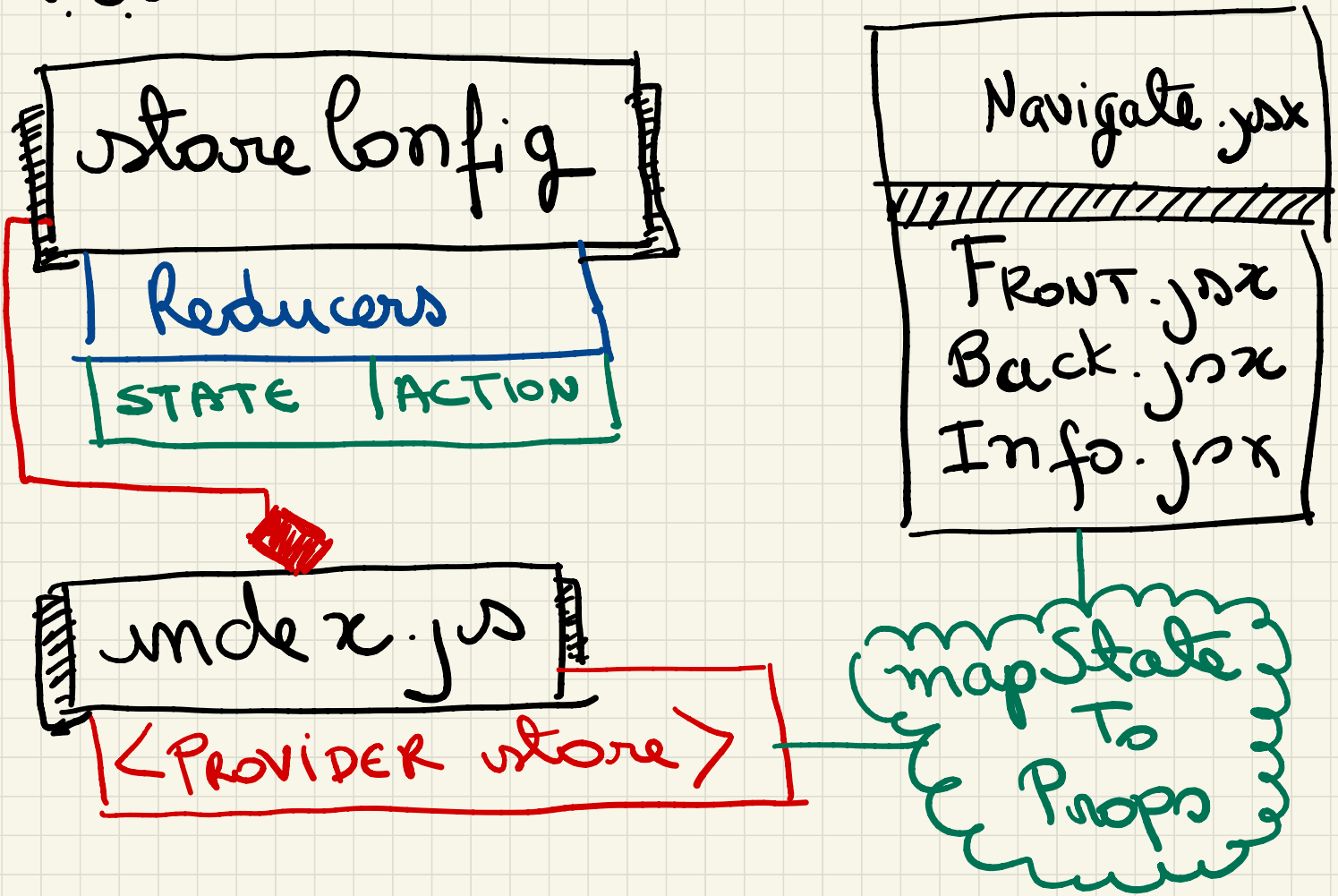
OBJETIVOS DO CURSO

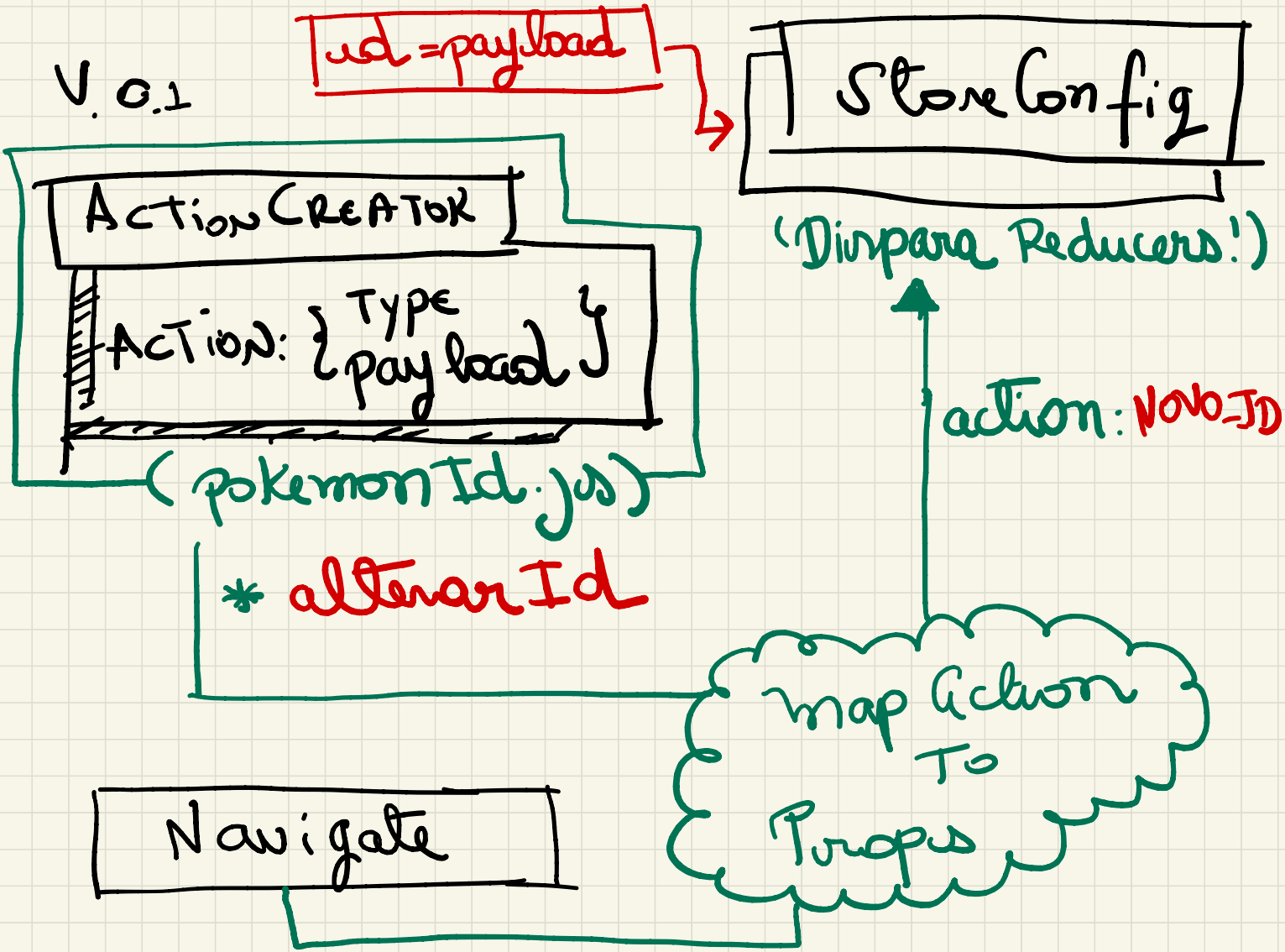
* MODIFICAR A MESMA APLICAÇÃO PARA TRABALHAR COM O REDUX.

APLICAÇÃO DO CURSO



V.00


store Config
Reducers
STATE
ACTION
index.js
<PROVIDER store>
Navigate.jsx
FRONT.jsx
Back.jsx
Info.jsx
mapState To Props

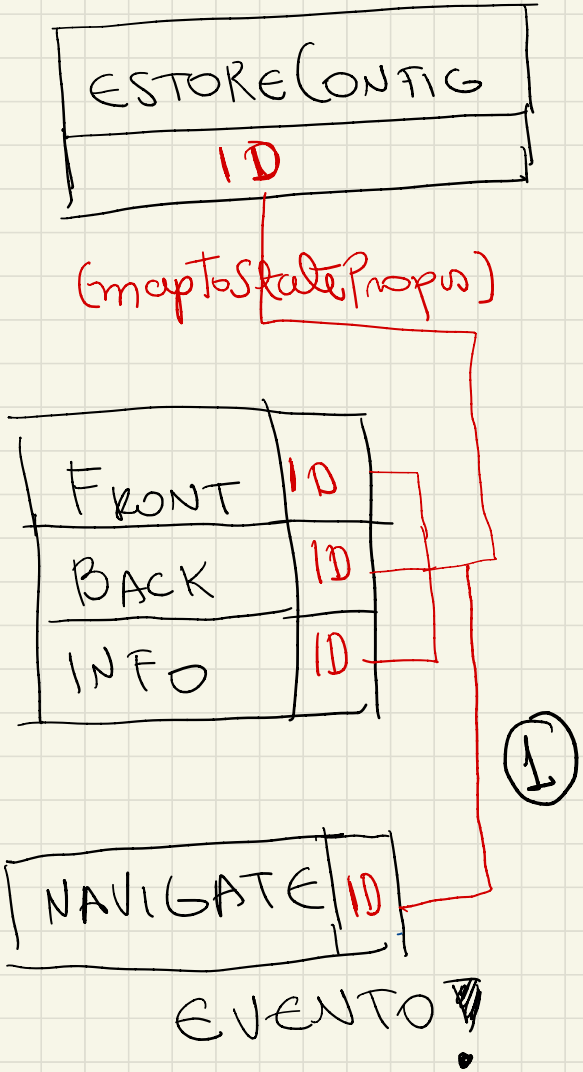

V. 0.1
id = payload
StoreConfig
(Dispara Reducers!)
ActionCreator
Action: { Type payload }
(pokemonId.js)
* alterarId
action: NOVO_ID
mapAction To Props
Navigate

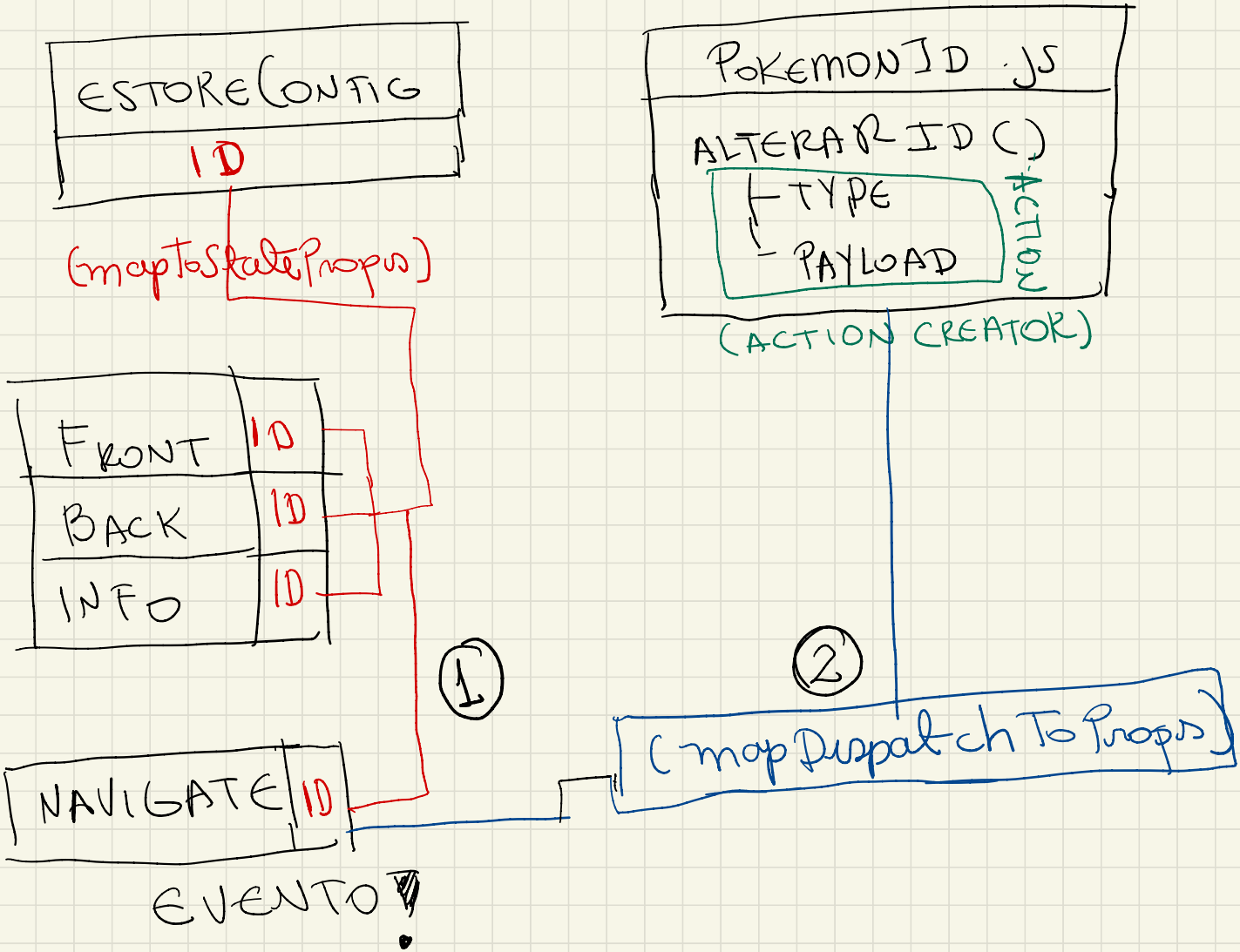
V.0.2

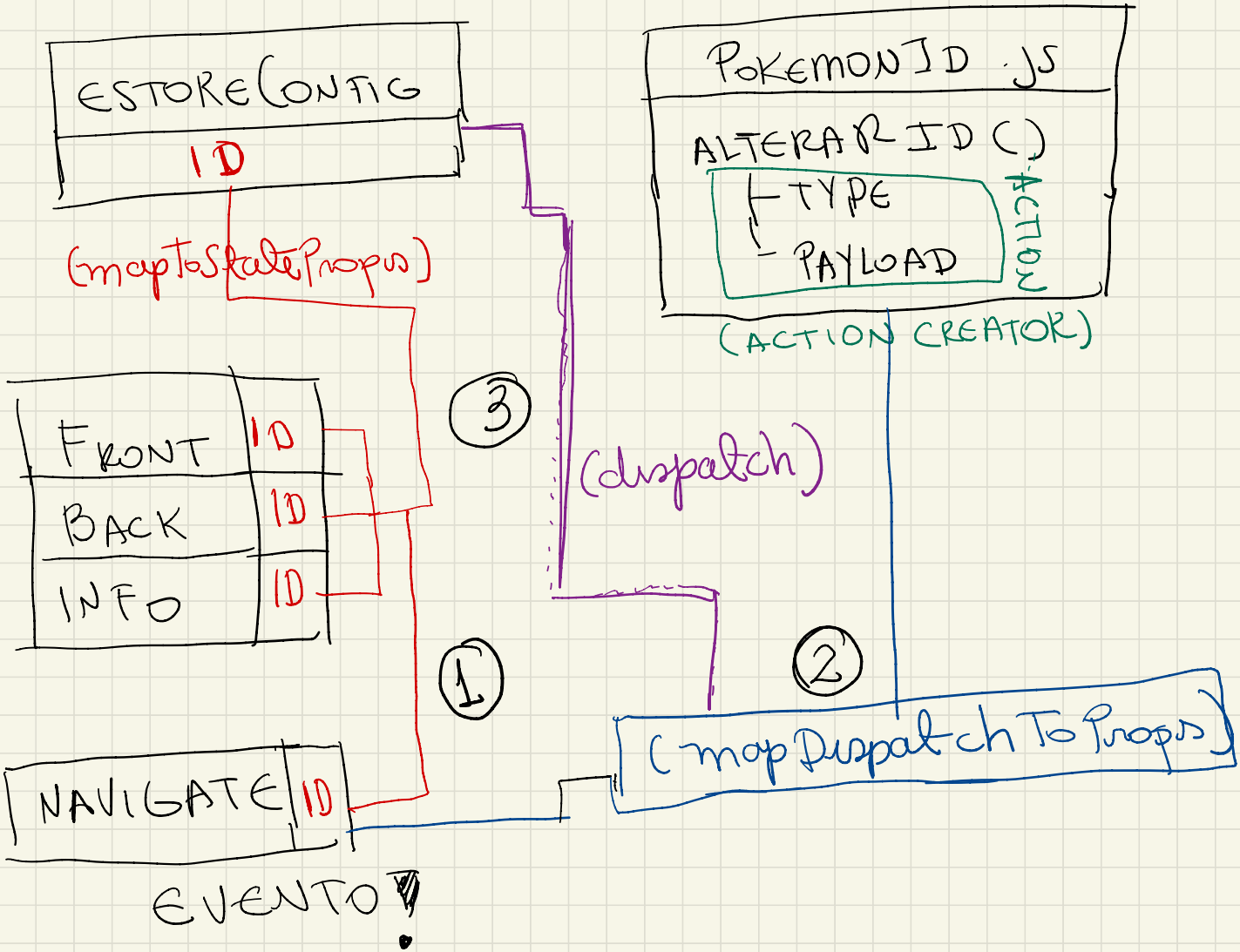
storeConfig.js
(SWITCH NO TYPE da Action)

navigate.js
(Eventos chamando o Dispatcher)

* criação da pasta Reducers
* criação das action types constantes.
* refatoração do Initate Config.


ESTORECONFIG
ID
(mapToStateProps)
FRONT
ID
BACK
ID
INFO
ID
1
NAVIGATE
ID
EVENTO!


ESTORECONFIG
ID
(mapToStateProps)
FRONT
ID
BACK
ID
INFO
ID
NAVIGATE
ID
EVENTO!
1
POKEMONID.JS
ALTERARID()
TYPE
PAYLOAD
ACTION
(ACTION CREATOR)
2
(mapDispatchToProps)


ESTORECONFIG
ID
(mapToStateProps)
FRONT
ID
BACK
ID
INFO
ID
NAVIGATE
ID
EVENTO!
3
1
2
(dispatch)
POKEMONID.JS
ALTERARID()
TYPE
PAYLOAD
ACTION
(ACTION CREATOR)
(mapDispatchToProps)