

**FIAP 3SIS**

Sistemas de Informação  
Desenvolvimento Cross Platform  
Professor Paulo Sabra

# Checkpoint 04

## Gerenciamento de Estado

### VISÃO GERAL

No contexto do Flutter, um framework popular para criar aplicativos móveis, o Bloc (Business Logic Component) é uma biblioteca que ajuda a gerenciar o estado de um aplicativo de maneira eficiente.

O Flutter Bloc é uma ferramenta poderosa para gerenciar o estado de aplicativos Flutter de forma estruturada e eficiente. Ele ajuda a manter seu código organizado, facilita a depuração e torna seu aplicativo mais robusto. À medida que você se aprofunda no desenvolvimento de aplicativos Flutter, o Bloc se tornará uma parte valiosa de sua caixa de ferramentas.

### OBJETIVOS

1. Compreensão Básica do Bloc.
2. Uso de Eventos e Estados.
3. Uso de Flutter Widgets.

### ESPECIFICAÇÕES

Utilizando o projeto GitHub Discover que foi feito o *fork* em seu repositório pessoal no GitHub, é solicitado que proceda com a resolução deste teste. Ao concluir, é requisitado que abra uma nova Pull Request, identificando-se com seu nome completo e número de matrícula (RM#####), além de preencher o modelo de solicitação de Pull Request fornecido.

## ATIVIDADES

### Busca de Repositórios (2,5 Pontos)

1. **Events:** crie o evento Busca
2. **States:** crie os estados de Carregamento, Sucesso, Erro e Vazio
3. **Bloc:** construa a lógica de negócio
4. **Screen:** Construa a lógica da tela
5. **Page:** atualize o código para conectar com a Screen
6. **Routes + HomePage:** troque as Pages pelas respectiva Screen

### Detalhes do Repositório (2,5 Pontos)

1. **Events:** crie o evento de carregamento de dados
2. **States:** crie os estados de Carregamento, Sucesso, Erro
3. **Bloc:** construa a lógica de negócio
4. **Screen:** Construa a lógica da tela
5. **Page:** atualize o código para conectar com a Screen
6. **Routes + HomePage:** troque as Pages pelas respectiva Screen

### Busca de Usuários (2,5 Pontos)

1. **Events:** crie o evento Busca
2. **States:** crie os estados de Carregamento, Sucesso, Erro e Vazio
3. **Bloc:** construa a lógica de negócio
4. **Screen:** Construa a lógica da tela
5. **Page:** atualize o código para conectar com a Screen
6. **Routes + HomePage:** troque as Pages pelas respectiva Screen

### Detalhes do Usuário (2,5 Pontos)

1. **Events:** crie o evento de carregamento de dados
2. **States:** crie os estados de Carregamento, Sucesso, Erro
3. **Bloc:** construa a lógica de negócio
4. **Screen:** Construa a lógica da tela
5. **Page:** atualize o código para conectar com a Screen
6. **Routes + HomePage:** troque as Pages pelas respectiva Screen