



**Copa do Mundo  
InovaTech 2026**

# Prova Prática INOVATECH 2026

Copa do Mundo InovaTech 2026 · O Álbum das Lendas

**02 de maio de 2026** · 09h às 12h (3 horas)

Processo Seletivo Simplificado – Residência em Inovação Tecnológica

Edital 30/2026 – GAB/REI/IFPI

## 1. Contexto

Em celebração ao ano de Copa do Mundo, o **Copa do Mundo InovaTech 2026** é uma plataforma colaborativa para colecionadores de figurinhas de jogadores lendários. O objetivo é criar um ambiente onde usuários possam cadastrar “cards”/figurinhas, gerenciar sua coleção pessoal e interagir com as relíquias de outros colecionadores através de avaliações técnicas, comentários e curtidas.

A aplicação deve ser composta por uma **API RESTful** (Backend) e uma **interface web funcional** (Frontend), integradas entre si, utilizando obrigatoriamente o ambiente técnico disponibilizado – SQLite, versões fixadas nas Notas Técnicas 03, 04 e 05.

## 2. Regras de Negócio e Funcionalidades

### 2.1 Autenticação e Perfil

- O sistema deve permitir o **cadastro de novos usuários** (nome, e-mail e senha) e o **login** para obtenção de token de acesso (JWT é diferencial).
- Deve haver uma rota autenticada `/me` para consulta dos dados do perfil logado.

### 2.2 Gerenciamento do Álbum (Cards)

- Usuários autenticados podem publicar figurinhas contendo: Nome do Jogador, Bio/Descrição, Posição do jogador em campo (atacante, zagueiro, goleiro, meio-campista, volante), Tags de raridade (lendária, rara, incomum, comum) e imagem.
- As imagens estão disponíveis em `/assets/figurinhas/` no frontend (pasta `public/assets/figurinhas/`).
- O dono pode **remover** a figurinha ou **“guardá-la no cofre”** (ocultá-la da listagem pública sem excluir).
- Figurinhas ocultas são visíveis apenas para o dono.

- Visitantes (não autenticados) visualizam apenas os **nomes** dos jogadores, sem detalhes completos.

### 2.3 Avaliação Técnica (Votos)

- Usuários autenticados podem votar **uma única vez** em cada figurinha: **UpVote (Craque)** ou **DownVote (Bagre)**.
- A figurinha deve exibir o total de votos positivos, negativos e o **saldo líquido** (UpVotes – DownVotes).
- **Diferencial visual:** o card deve ter destaque **verde** se o saldo for positivo e **vermelho** se for negativo.

### 2.4 Interação (Comentários e Likes)

- Qualquer usuário autenticado pode **comentar** em figurinhas visíveis.
- É permitido dar **like** em figurinhas ou comentários, com opção de **desfazer** a ação (toggle).

## 3. Escopo Negativo

**O que NÃO é necessário nesta prova** – para otimizar seu tempo de desenvolvimento:

- Recuperação de senha via e-mail ou integração com serviços externos de internet.
- Edição de conteúdo de figurinhas após a publicação (regra de imutabilidade).
- Configuração de Docker ou deploy em nuvem; a entrega é **local**.

## 4. Critérios de Avaliação

### Entregas parciais

Entregas parciais serão avaliadas e pontuadas proporcionalmente dentro de cada competência.

#	Competência	Nível máximo (200 pontos para cada uma das competências)
1	<b>Backend</b>	API REST completa, consistente, com uso correto de ORM e autenticação funcional.
2	<b>Frontend</b>	Interface funcional, integrada à API, com navegação fluida e operações completas.
3	<b>Requisitos</b>	Atendimento integral às regras de negócio (votos, saldo, ocultação, regras de acesso).
4	<b>Qualidade</b>	Código limpo, organizado, modular e com adoção de boas práticas.
5	<b>Documentação</b>	README completo com arquitetura, decisões técnicas e instruções de execução.

## 5. Procedimentos de Entrega

**⚠ Leia com atenção — siga a ordem abaixo.**

**Todos os comandos abaixo pressupõem que você está na pasta `~/inovatech`.**

### Passo 1 — Verificar integridade do ambiente ao chegar

Execute no terminal e compare o hash exibido com o hash público divulgado pela coordenação. Se divergir, informe imediatamente.

```
inovatech-verify
```

### Passo 2 — Preparar sua pasta de entrega

Antes de começar a desenvolver, execute:

```
inovatech-preparar-entrega
```

O comando perguntará seu **código PP** (dois dígitos, divulgado pela coordenação), qual backend e qual frontend você usará. Em seguida, copia a stack escolhida para `~/inovatech/entrega_##/` e recria os ambientes automaticamente (`.venv` Python e `node_modules`).

**⚠ Não copie** `.venv` ou `node_modules` manualmente — eles contêm caminhos absolutos que quebram ao mover de pasta.

### Passo 3 — Desenvolver dentro da pasta de entrega

Trabalhe **exclusivamente** dentro de `~/inovatech/entrega_##/`. Exemplos de inicialização dos servidores:

```
# Backend Django
```

```
cd ~/inovatech/entrega_##/backend-django  
source .venv/bin/activate && python manage.py runserver
```

```
# Backend FastAPI
```

```
cd ~/inovatech/entrega_##/backend-fastapi  
source .venv/bin/activate && uvicorn main:app --reload
```

```
# Backend Express
```

```
cd ~/inovatech/entrega_##/backend-express && npm run dev
```

```
# Frontend (React ou Vanilla)
```

```
cd ~/inovatech/entrega_##/frontend-react && npm run dev
```

**Passo 4 — Gerar comprovante ao encerrar**

Antes de sair do computador, execute:

```
inovatech-submit
```

Informe seu nome completo e código PP. **Anote o hash gerado** – ele é seu com-provante de entrega com integridade verificável.

**Resumo dos comandos**

- |                              |   |
|------------------------------|---|
| ✔ inovatech-verify           | Verificar integridade do lab ao chegar        |
| ✔ inovatech-preparar-entrega | Preparar pasta entrega_## e recriar ambientes |
| ✔ inovatech-submit           | Gerar comprovante ao finalizar                |
| i inovatech-versions         | Conferir versões do ambiente (opcional)       |

---

**Boa prova!**

INOVATECH 2026 · Residência em Inovação Tecnológica · GAB/REI/IFPI