

Projeto Engenharia de Software

FuteApp

Breno Santos
Eduardo Teles
Guilherme Ribeiro
Josef Jaeger
Luca Neves

Back-end: <https://github.com/lucanevess87/fute-api>
Front-end: <https://github.com/lucanevess87/fute-app>

Visão geral do produto

É um aplicativo voltado para Gestores de Peladas, que realizam as atividades hoje de uma forma muito manual e em ferramentas diversas


O nosso produto trará uma centralização e organização de todas as informações necessárias para a pelada em um só local.



































O nosso diferencial para outras soluções será o foco no gestor da pelada, não precisando do engajamento de todos os participantes para funcionar

Processo

- 1. Definição do Tema**
- 2. Reunião com cliente para entender requisitos**
- 3. Criação de Histórias do Usuário**
- 4. Modelo ER (LucidChart)**
- 5. Protótipo (Figma)**
- 6. Detalhamento do backlog e alocação de tarefas (ClickUp)**
- 7. Definição de Sprints (ClickUp)**
- 8. Quadro Kanban para acompanhamento das atividades (ClickUp)**

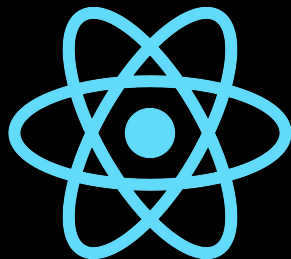
Funcionalidades esperadas

 HISTÓRIAS DO USUÁRIO 10 TAREFAS

-  Criação de pelada  
-  Cadastro de jogadores  1  
-  Criação de evento específico de pelada   
-  Sorteio dos times para um evento de pelada   
-  Histórico de peladas  
-  Anotar Estatísticas do jogo  
-  Ver estatísticas de jogadores  
-  Controle de finanças  2  
-  Encontrar pelada  
-  Visualizar informações da pelada  

+ Nova tarefa

Tecnologias Utilizadas



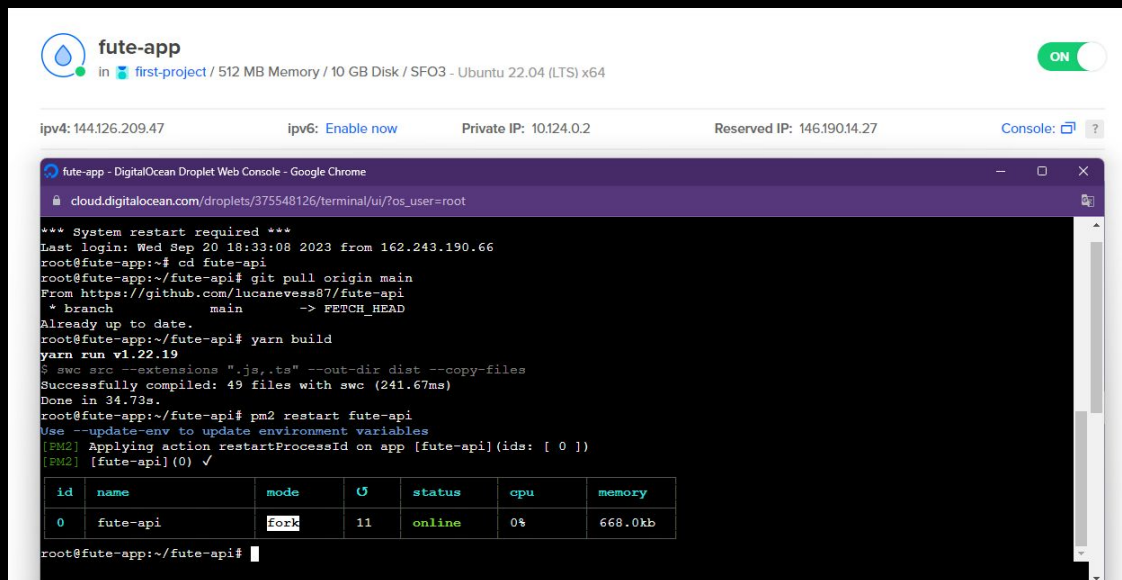
NEXT.JS



chakra

Desafios Finais

- Integração entre Back e Front
- Deploy
- Testes - Escritos mas não implementados



The screenshot shows the DigitalOcean Droplet Web Console for a droplet named 'fute-app'. The console displays the terminal output of a user performing a system restart, pulling the latest code from a GitHub repository, building the application with Yarn, and restarting the service with PM2. Below the terminal output, a table shows the status of the application processes.

```
*** System restart required ***
Last login: Wed Sep 20 18:33:08 2023 from 162.243.190.66
root@fute-app:~# cd fute-api
root@fute-app:~/fute-api# git pull origin main
From https://github.com/lucanevess87/fute-api
* branch      main      -> FETCH_HEAD
Already up to date.
root@fute-app:~/fute-api# yarn build
yarn run v1.22.19
$ swc src --extensions ".js,.ts" --out-dir dist --copy-files
Successfully compiled: 49 files with swc (241.67ms)
Done in 34.73s.
root@fute-app:~/fute-api# pm2 restart fute-api
Use --update-env to update environment variables
[PM2] Applying action restartProcessId on app [fute-api] (ids: [ 0 ])
[PM2] [fute-api] (0) ✓
```

id	name	mode	U	status	cpu	memory
0	fute-api	fork	11	online	0%	668.0kb

root@fute-app:~/fute-api#

Divisão do Time



Back-end



Back-end



Front-end



Front-end



Full-Stack

Lições Aprendidas

- **As histórias do usuário bem definidas e detalhadas no início do projeto foram fundamentais**
- **Manter canais de comunicação claros e regulares entre os membros da equipe foi fundamental para alinhamentos e evitar mal-entendidos.**
- **Deveríamos ter nos organizado melhor para a realização de testes.**