

RESTA UM



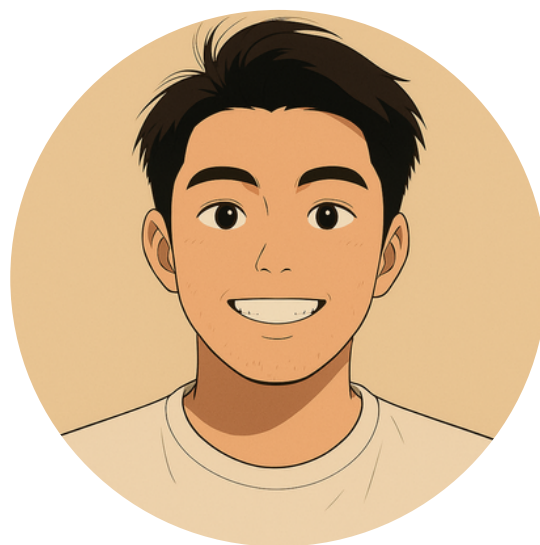
Engenharia de Software II
parte 2

INTEGRANTES



Anderson AKira
MeKaru

Developer



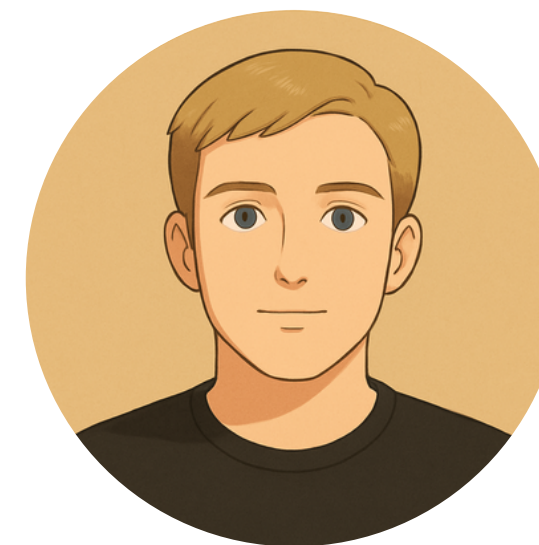
Daniel Tsuyoshi
Izumi

Developer



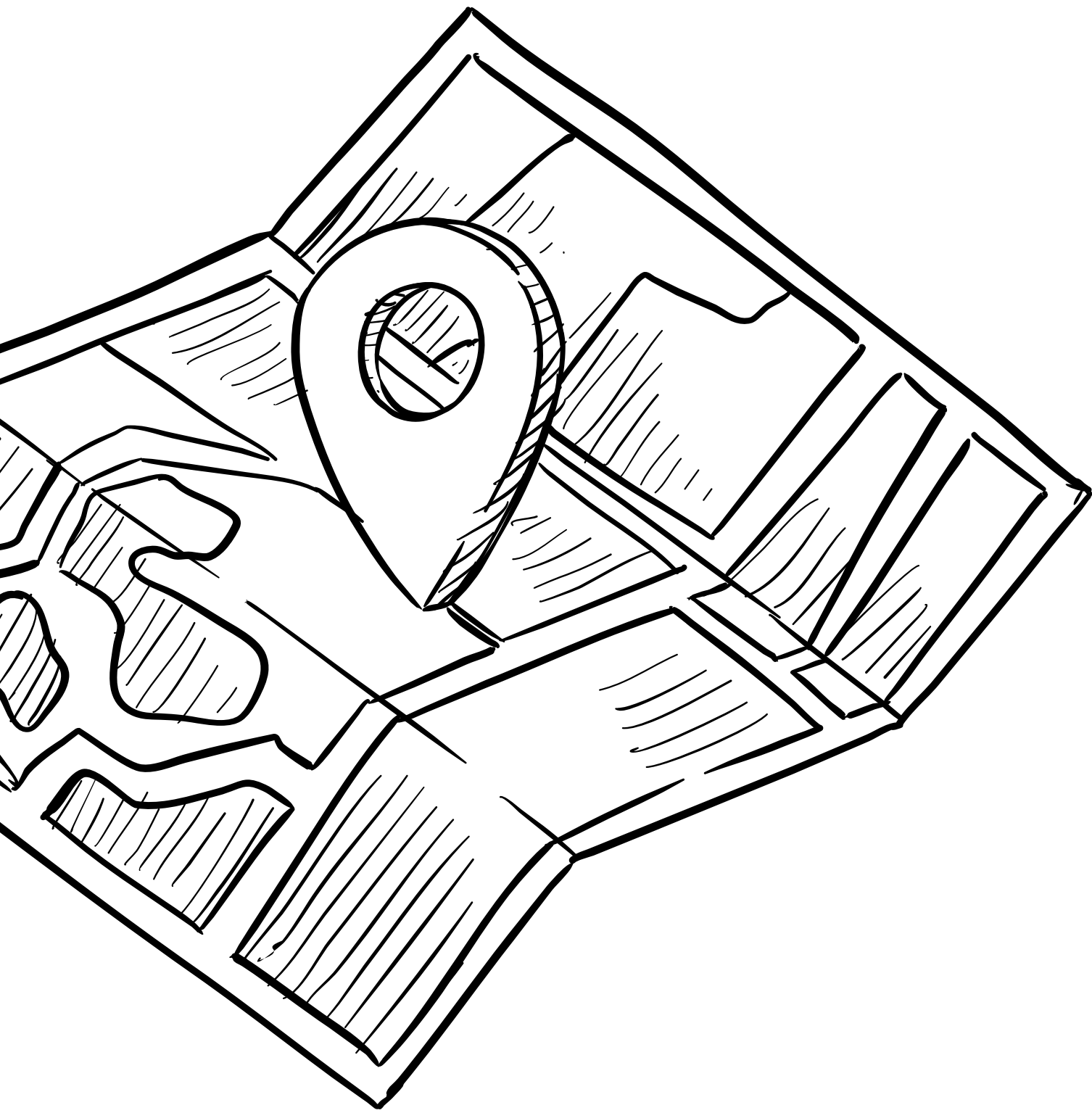
Marcelo Alfonso de
Souza

Product Owner
Developer



Victor Oliveira da
Nóbrega Santana

Scrum Master
Developer



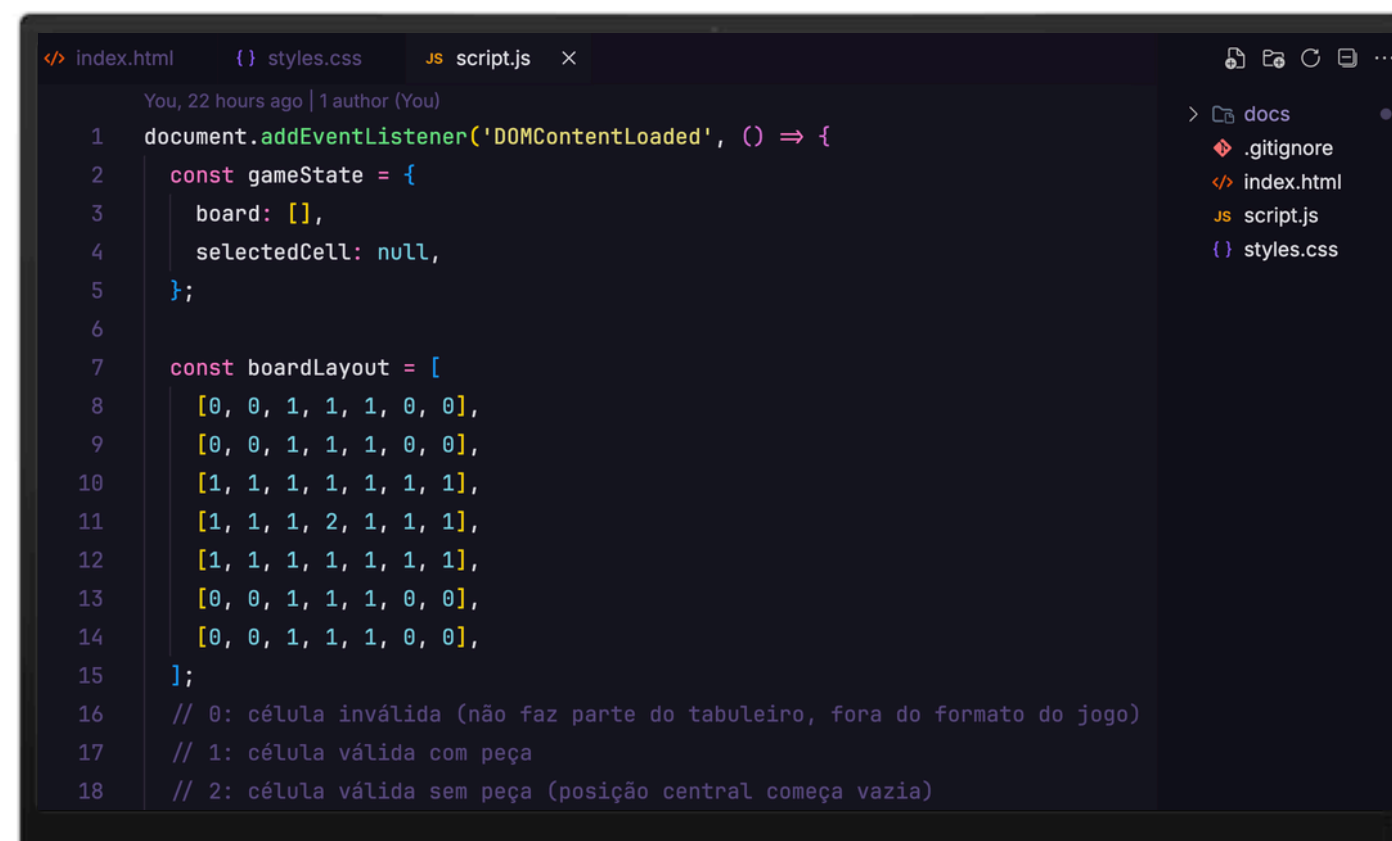
SPRINT 0

(15/09-20/09)

- Objetivos: definir linguagem de programação, serviço de hospedagem e sistema de controle de versão
- Fatores a se considerar: desempenho, compatibilidade, ecossistema, curva de aprendizado, custo, escalabilidade, entre outros
- HTML + CSS + JS
- Vercel
- Git + GitHub

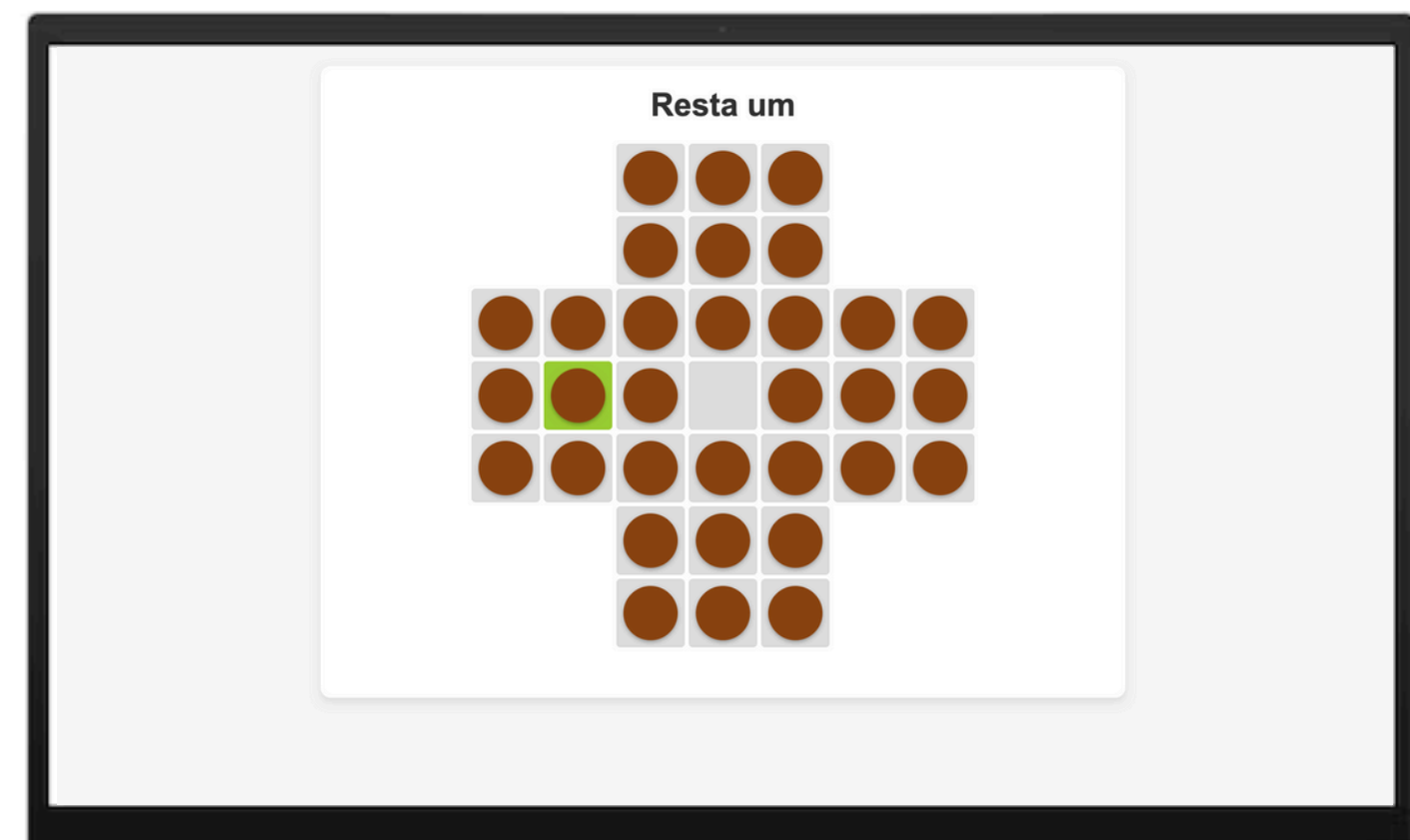
SPRINT 1 (22/09-27/09)

OBJETIVOS



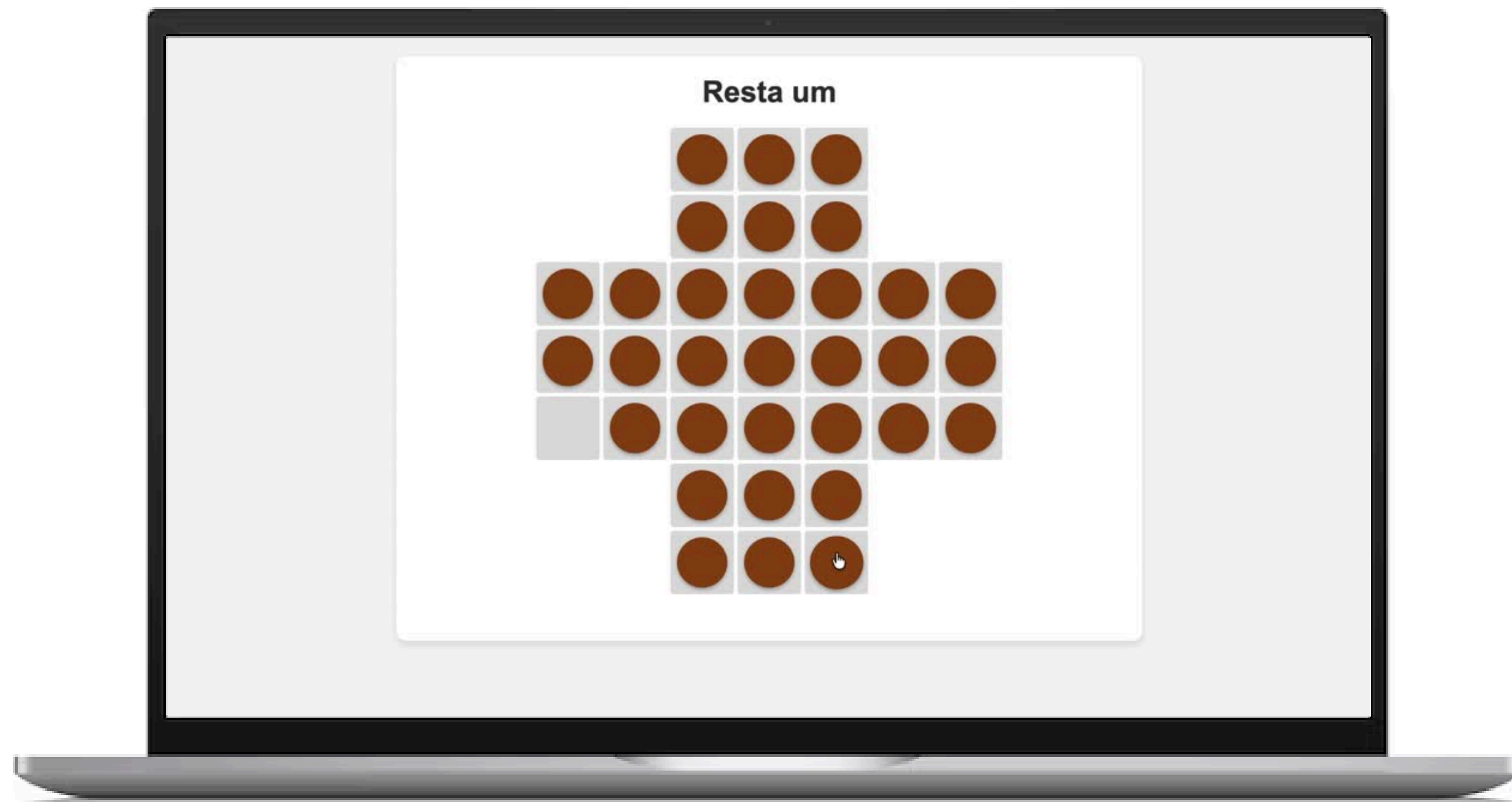
```
1 document.addEventListener('DOMContentLoaded', () => {
2   const gameState = {
3     board: [],
4     selectedCell: null,
5   };
6
7   const boardLayout = [
8     [0, 0, 1, 1, 1, 0, 0],
9     [0, 0, 1, 1, 1, 0, 0],
10    [1, 1, 1, 1, 1, 1, 1],
11    [1, 1, 1, 2, 1, 1, 1],
12    [1, 1, 1, 1, 1, 1, 1],
13    [0, 0, 1, 1, 1, 0, 0],
14    [0, 0, 1, 1, 1, 0, 0],
15  ];
16  // 0: célula inválida (não faz parte do tabuleiro, fora do formato do jogo)
17  // 1: célula válida com peça
18  // 2: célula válida sem peça (posição central começa vazia)
```

Iniciar a codificação do jogo



Construir layout do tabuleiro e das peças; implementar seleção de peça

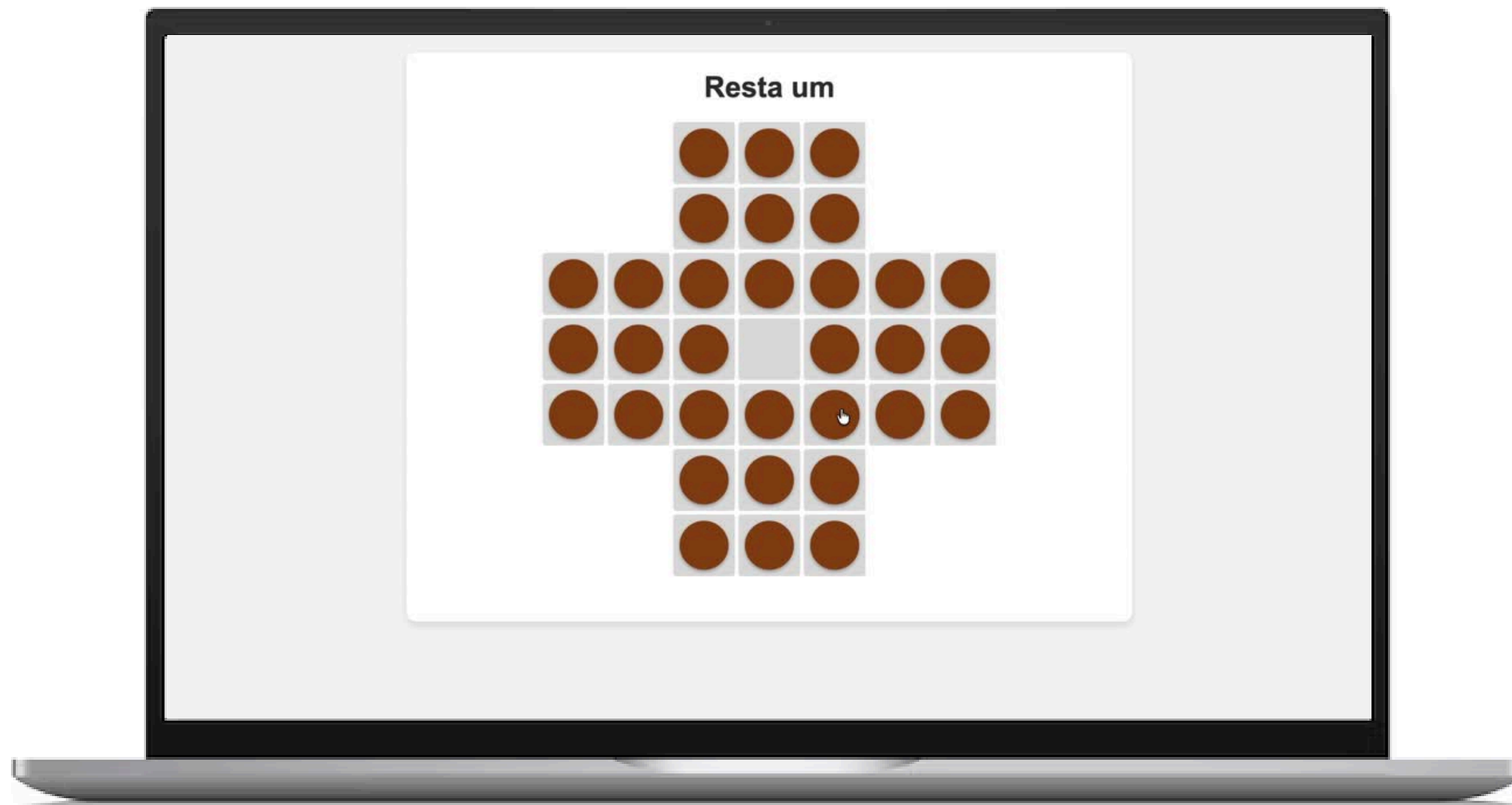
SPRINT 2 (29/09-04/09)



OBJETIVO

Permitir que o usuário mova uma peça para outra casa, removendo-a da posição inicial

SPRINT 3 (06/10-11/10)



OBJETIVO

Validar se a tentativa de movimento efetuada pelo usuário é permitida pelas regras do jogo; se não for, a peça permanece em sua posição atual

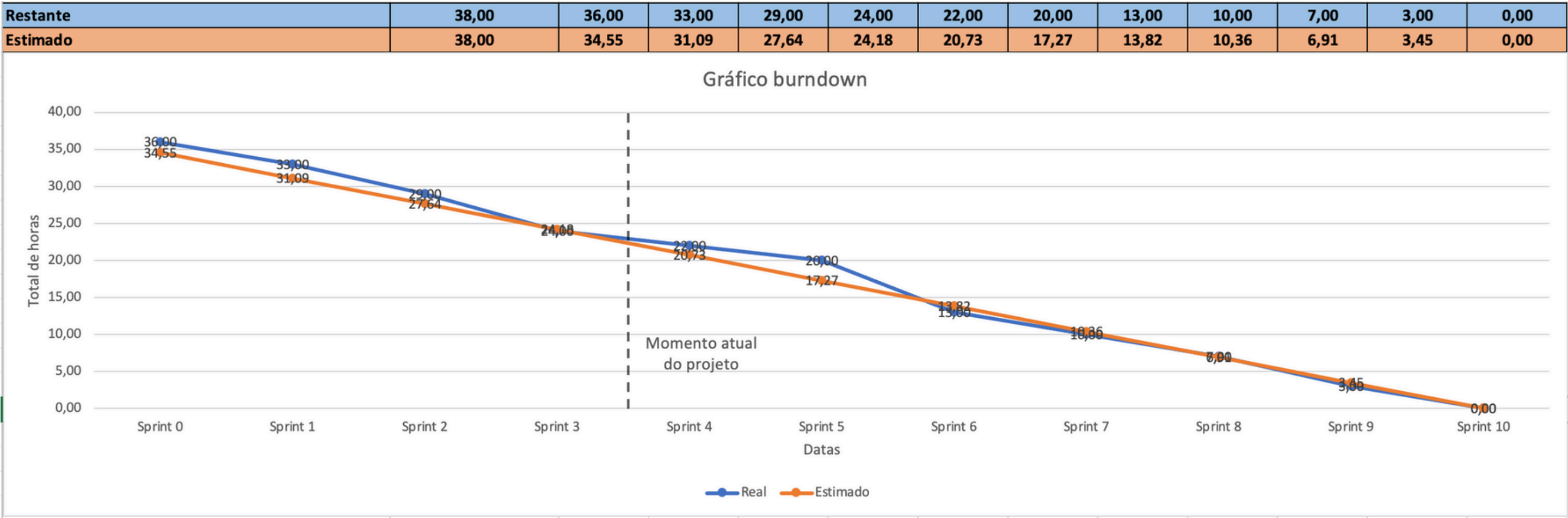
CRONOGRAMA + BURNDOWN



Atividade	Total de horas	Datas previstas										
		Sprint 0	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5	Sprint 6	Sprint 7	Sprint 8	Sprint 9	Sprint 10
Definir linguagem de programação	1	1										
Escolher tecnologias (versionamento, hospedagem)	1	1										
Construir tabuleiro com peças	2		2									
Implementar seleção de peça	1		1									
Implementar movimentação de peça	2			2								
Implementar remoção de peça	2			2								
Fazer validação dos movimentos	5				5							
Construir botão para reiniciar o jogo	2					2						
Desenvolver exibição de feedback para o usuário	2						2					
Executar testes	4							4				
Aplicar correções	3							3				
Fazer deploy da aplicação	2								2			
Monitorar desempenho da aplicação	1								1			
Garantir documentação suficiente	3									3		
Revisar e ajustar pontos do projeto	4										4	
Preparar apresentação final	3											3

SPRINT ATUAL:
Sprint 4

SPRINTS REALIZADAS:
Sprint 0
Sprint 1
Sprint 2
Sprint 3



FLUXO DE TRABALHO

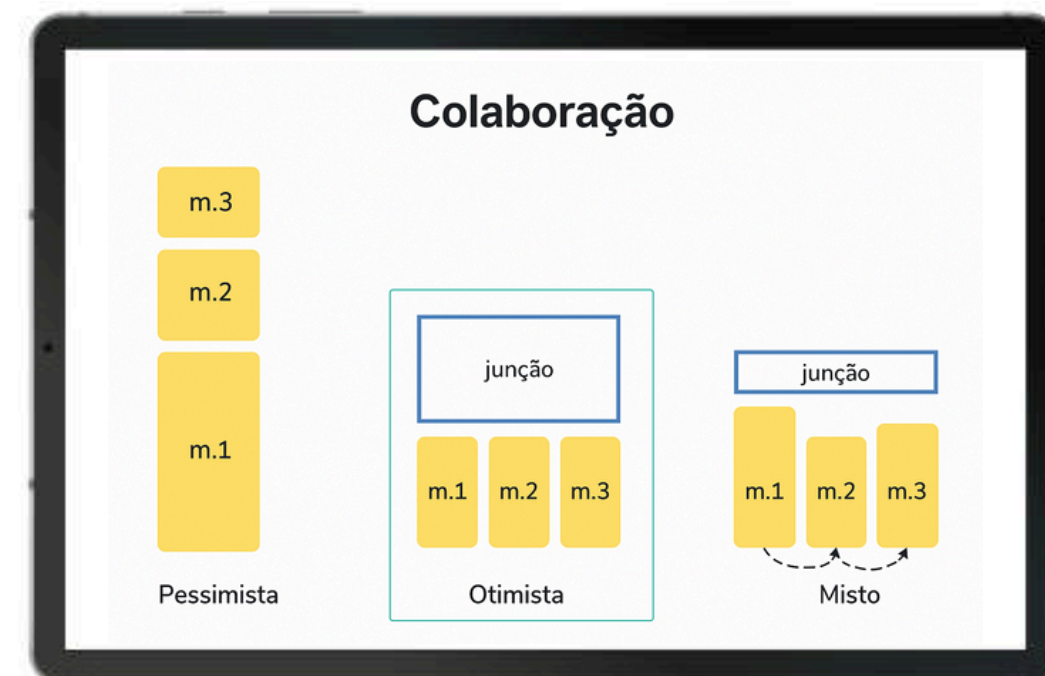
ATIVIDADE	RESPONSÁVEL	EM ANDAMENTO?	CONCLUÍDA?	OBSERVAÇÕES
Definir linguagem de programação	Todos		Sim	
Escolher tecnologias (versionamento, hospedagem)	Todos		Sim	
Construir tabuleiro com peças	Marcelo Alfonso		Sim	
Implementar seleção de peça	Marcelo Alfonso		Sim	
Implementar movimentação de peça	Marcelo Alfonso		Sim	
Implementar remoção de peça	Marcelo Alfonso		Sim	
Fazer validação dos movimentos	Victor Oliveira		Sim	
Construir botão para reiniciar o jogo	Marcelo Alfonso	Sim		
Desenvolver exibição de feedback para o usuário	Victor Oliveira			
Executar testes	Daniel Izumi			
Aplicar correções	Daniel Izumi			
Fazer deploy da aplicação	Todos			
Monitorar desempenho da aplicação	Todos			
Garantir documentação suficiente	Daniel Izumi			
Revisar e ajustar pontos do projeto	Anderson Akira			
Preparar apresentação final	Anderson Akira			

Weekly: toda sexta-feira às 19:00



CONTROLE DE VERSÃO

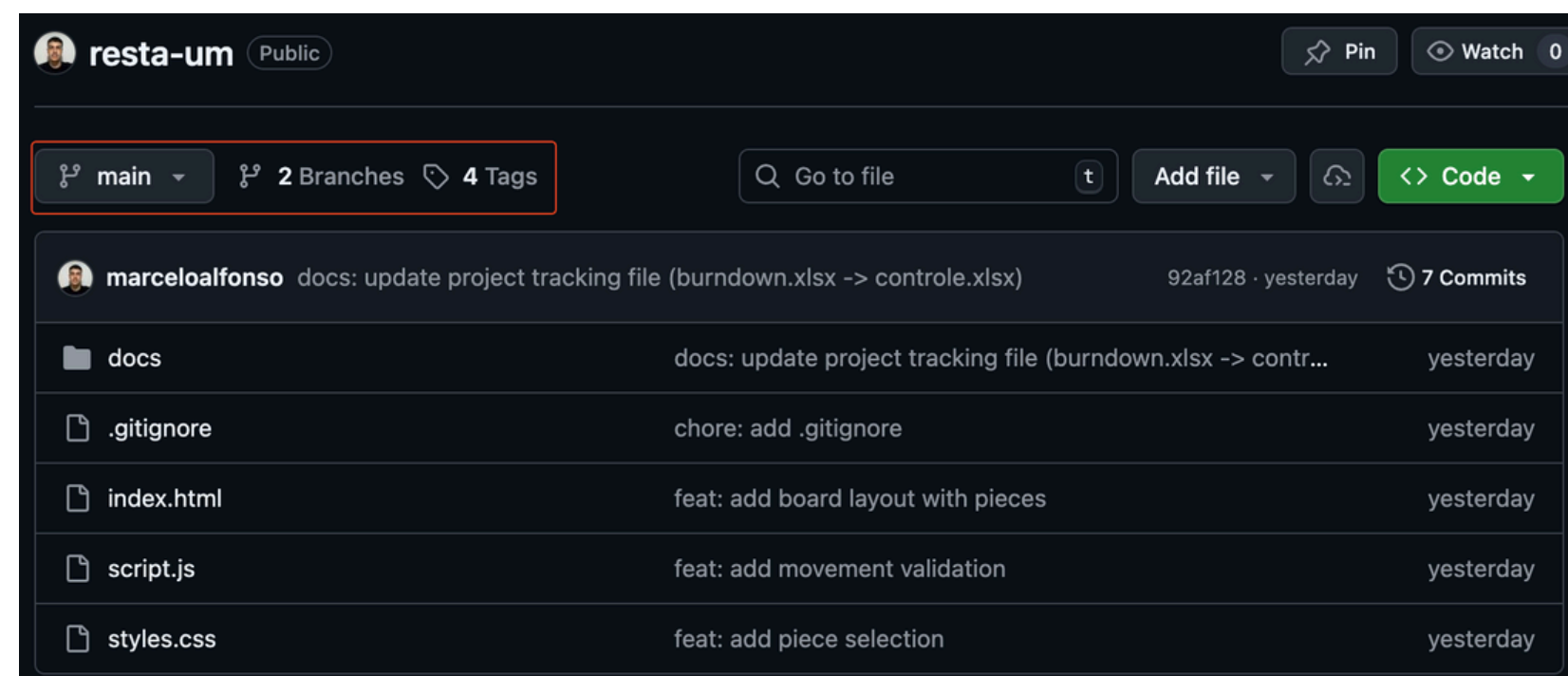
- Repositório GitHub:
<https://github.com/marceloalfonso/resta-um>
- Artefatos: códigos (index.html, styles.css, script.js), arquivos de configuração (.gitignore) e arquivos de documentação (/docs)
- Modelo de armazenamento completo lógico, com otimização interna via delta (forward/reverse)
- Colaboração otimista: todos conseguem editar o mesmo artefato simultaneamente; as modificações são integradas no momento do merge



ESTRATÉGIA DE RAMIFICAÇÃO



- GitFlow: separação em branch principal (main), branch de desenvolvimento (develop) e branches temporárias de funcionalidades (feature/<nome-da-funcionalidade>)
- Uso de tags para rastreabilidade das funcionalidades adicionadas



**OBRIgADO PELA
ATENçÃO!**