

# Sprint 1

03/06/2024 à 14/06/2024

## QA

### Equipe

Entender os Tipos de Testes para Websites, levantando potenciais ferramentas para uso.

#### **Tipos de Testes:**

- Teste de Funcionalidade
- Teste de Usabilidade
- Teste de Compatibilidade
- Teste de Compatibilidade Entre Navegadores (Cross-browser)
- Teste de Responsividade
- Teste de Acessibilidade
- Teste de Performance
- Teste de Aceitação de Usuário (UAT - User Acceptance Testing)
- Teste de Regressão
- Teste de Localização

### **Critérios de Aceitação**

1. Definir quais testes e ferramentas (quando necessário) serão utilizadas.
2. Comunicação ao Squad de FrontEnd quais Testes serão aplicados e critérios de aceitação, quando houver.

## Janine

Entender o que significa o Core Web Vitals, quais métricas são analisadas com **foco** no **LCP** (Largest Contentful Paint), **INP** (Interaction to Next Paint) e **CLS** (Cumulative Layout Shift).

Conhecer as 3 ferramentas do Google para medir as Core Web Vitals:

- PageSpeed Insights
- Lighthouse
- Search Console

## Critérios de Aceitação

1. Apresentar um resumo por que as principais métricas são LCP, INP e CLS. Definir a pontuação ideal para cada uma das 3.
2. Comunicar ao Squad de FrontEnd quais métricas serão analisadas e a pontuação de cada uma.

(Máximo 2 páginas para os itens 1 e 2)

## Gustavo

Analisar modelos de Fluxo de QA e elaborar um Fluxo de QA a ser utilizado pelo Squad de QA. Alguns pontos a serem considerados:

- Responsável pelo teste
- Tipo de teste
- Objeto de teste
- Requisitos do teste
- Critérios de aceitação do teste, quando aplicáveis

## Critérios de Aceitação

1. Criação de um Fluxo de QA que atenda as tarefas do Squad de QA.

## Túlio

**Entender** os tipos de testes: End-to-End, integração, e unitários. Levantar **potenciais** ferramentas e seus usos.

### Critérios de Aceitação

1. Exemplificar cada teste, quando aplicável, nos Squad de FrontEnd e BackEnd.
2. Definição das ferramentas potenciais a serem utilizadas futuramente.

(Máximo de 3 páginas para os itens 1 e 2)

## DevOps

### Equipe

**Entender** o uso de Docker, Kubernetes e Terraform. Entender a diferença entre Docker e Docker Compose.

### Critérios de Aceitação

1. Subir **localmente** uma aplicação Nest.js usando Docker Compose.

## UX Design

### Equipe

Realização do "Brainstorm" de comunidades de TI, identificando identidade visual e serviços.

### Critérios de Aceitação

1. Seleção dos recortes mais significativos no Figma.

## Luana

1. UX Research: Realizar uma Pesquisa Atitudinal, Quantitativa com no mínimo 50 pessoas (20% de margem de erro) e **idealmente 100 pessoas** (10% margem de erro), formada de 3 hipóteses das maiores dores que "Desenvolvedores Autodidatas enfrentam para entrar no mercado de TI", com 3 perguntas não inviezadas para cada desafio. A 10ª pergunta do questionário deve convidar o(a) participante para futuras pesquisas.
2. Mapeamento de Experiências: Criação da Persona "Desenvolvedora Autodidata" utilizando os dados da pesquisa do item 1.

### Critérios de Aceitação

1. Resultados da pesquisa.
2. Criação do Artefato Persona no Figma.

## Hugo

1. UX Research: Realizar uma Pesquisa Atitudinal, Quantitativa com no mínimo 50 pessoas (20% de margem de erro) e **idealmente 100 pessoas** (10% margem de erro), formada de 3 hipóteses das maiores dores que "Desenvolvedores com Escolaridade de Nível Superior enfrentam para entrar no mercado de TI", com 3 perguntas não inviezadas para cada desafio. A 10ª pergunta do questionário deve convidar o(a) participante para futuras pesquisas.
2. Mapeamento de Experiências: Criação da Persona "Desenvolvedor com ou em formação em ADS, Ciência da Computação ou afins" utilizando os dados da pesquisa do item 1.

### Critérios de Aceitação

1. Resultados da pesquisa.
2. Criação do Artefato Persona no Figma.

**Sugestão de Convite:** "Você deseja participar de pesquisas de forma síncrona ou assíncrona pelo Whatsapp para contribuir com a **comunidade gratuita de TI**

## Código Certo?"

**Sugestão de Persona:** Empresa de Viagens, Clientes que querem viajar, contexto da pandemia de Covid.



**Aika Sato**  
Vendedora de Seguros

**Sobre**

45 anos  
Casada com Rodrigo  
2 filhos, Fernando de 5 e Felipe de 8

**Necessidades**

"Antes da pandemia costumava viajar com a minha família nas férias, desejamos voltar a viajar, mas não sabemos como planejar os gastos. Eu Rodrigo precisamos de férias, mas também prover recreação para as crianças. Ano passado, tive uma viagem cancelada, devido a pandemia e não sei quais são as regras para recupera-la, ou para viajar novamente, pois não tive apoio da minha agência de viagens."

**Dores**

- Voltar a viajar nas férias das crianças
- Recuperar viagem cancelada.
- Entender as regras de viagem atuais devido a pandemia.
- Encontrar pacotes com bom custo benefício.

**Analésicos**

- Oferecer novos pacotes de viagem viáveis devido a pandemia
- Reavaliar viagem cancelada.
- Comunicar de forma clara as regras de viagens atuais.
- Criar pacotes programados para famílias, com valores acessíveis e atividades recreativas.

## BackEnd

### Equipe

Inicializar **localmente** uma aplicação com Typescript, Nest.js, Node, TypeORM e PostgreSQL.

### Critério de Aceitação

1. A aplicação deve retornar "Hello Word" na rota principal "/".

## Vitor

Entender o que é framework opinado e não opinado e qual o impacto nas equipes de desenvolvimento. **Descrever** qual arquitetura o Nest.js implementa, explicando o papel de:

- controller
- entity
- module
- repository
- service

## Critério de Aceitação

1. Apresentar um resumo (máximo de 2 páginas) das características da arquitetura que o Nest.js implementa.

## Caio César

Entender o que é um ORM. Comparar os ORMs: Prisma, TypeORM e Drizzle, quanto a facilidade de uso, suporte ao Typescript, dentre outros fatores que julgar relevante.

## Critério de Aceitação

1. A apresentar um resumo (máximo de 2 páginas) da comparação dos ORMs: Prisma, TypeORM e Drizzle.

## Carlos

Entender para que serve o Swagger, **principalmente o Swagger UI**, na construção de REST APIs. Entender a diferença entre Swagger UI, Swagger Editor e Swagger API.

### Critério de Aceitação

1. A apresentar um resumo (máximo de 2 páginas) de Swagger no desenvolvimento de REST API com foco em Swagger UI.

## FrontEnd

### Equipe

Entender quais recursos de autenticação a plataforma Clerk oferece que a autenticação nativa do Next.js não possui.

### Critério de Aceitação

1. Subir localmente uma aplicação Next.js consumindo o serviço Clerk. No Clerk deve possuir o cadastro dos 3 membros do squad de FrontEnd e permitir novos cadastros.

### Levi

Entender 8 padrões modernos de renderização, principalmente quanto ao **uso**, tecnologia disponível (Next ou React), **impacto no SEO**. Foco nos 4 primeiros:

- **SSG: Static Site Generation**
- **ISR: Incremental Static Regeneration**
- **SSR: Server-side Rendering**
- **CSR: Client-side Rendering**
- Static Rendering

- Streaming SSR
- Edge Rendering
- React Server Components

### **Critério de Aceitação**

1. A apresentar um resumo (máximo de 2 páginas) dos parões de renderização.

## **Felipe**

Entender como o Next.js trabalha o cache das aplicações.

### **Critério de Aceitação**

1. A apresentar um resumo (máximo de 2 páginas) das estratégias do Next.js para gerenciamento do cache.

## **José Carlos**

Criar uma aplicação React que realize 5 animações simples usando a biblioteca Framer Motion.

### **Critério de Aceitação**

1. Criar localmente uma aplicação React utilizando a biblioteca Framer Motion que implemente 5 animações, sendo 2 obrigatoriamente:
  - Deslizamento de texto da direita para a esquerda e vice versa
  - Deslizamento de texto de baixo para cima