**Jodionísio Muachifi (97147), João Mourão (102578), João Rodrigues (102487) | Grupo 106**

Versão deste relatório: **2022-06-25**, v1.0

RELATÓRIO – *ELABORATION & CONSTRUCTION*

Construção

Conteúdos

[Construção 1](#_Toc105786362)

[1 Introdução 2](#_Toc105786363)

[1.1 Sumário executivo 2](#_Toc105786364)

[1.2 Controlo de versões 2](#_Toc105786365)

[1.3 Referências e recursos suplementares 3](#_Toc105786366)

[2 Arquitetura do sistema 3](#_Toc105786367)

[2.1 Objetivos gerais 3](#_Toc105786368)

[2.2 Requisitos com impacto na arquitetura 3](#_Toc105786369)

[2.3 Decisões e justificação 4](#_Toc105786370)

[2.4 Arquitetura do software 4](#_Toc105786371)

[2.5 Arquitetura física de instalação 5](#_Toc105786372)

[3 Incremento 1 6](#_Toc105786373)

[3.1 Casos de utilização no Incremento 1 6](#_Toc105786374)

[3.2 Histórias de utilização selecionadas 6](#_Toc105786375)

[3.3 Estratégia e estado da implementação 7](#_Toc105786376)

[Apêndice 15](#_Toc105786377)

[4 Especificação dos casos de utilização 15](#_Toc105786378)

[4.1 Pacote: sistema de resevas 15](#_Toc105786379)

[4.1.1 CaU 1 Fazer reserva online 15](#_Toc105786380)

[4.2 Pacote: sistema de pagamento **Erro! Marcador não definido.**](#_Toc105786381)

[4.2.1 CaU 1 Fazer pagamento online **Erro! Marcador não definido.**](#_Toc105786382)

# Introdução

## Sumário executivo

Este relatório apresenta os resultados da construção dos incrementos, adaptado os resultados esperados na etapa de *Elaboration* eConstruction, do método OpenUP.

A caraterização dos cenários a suportado é detalhada nos casos de utilização apresentados em apêndice (secção 5)

O primeiro incremento, desenvolvido na Iteração 3, foca a validação da arquitetura proposta. Foram considerados sobretudo as funcionalidades relacionadas com a plataforma da qual permite o cliente reservar tendas, Camping RV e Caravana, assim como efetuar o respetivo pagamento, verificando primeiramente à disponibilidade dos lugares.

## Controlo de versões

|  |  |  |
| --- | --- | --- |
| Quando? | Responsável | Alterações significativas |
| 30-05-2022 | Jodionísio Muachifi | * Configuração do Jira Software/github para gestão do projeto * Criação de planning do projeto no Jira: Roadmaps, instalação de Apps para integrar com Discord e Github. |
| 31-05-2022 | Jodionísio Muachifi, João Mourão, João Rodrigues | Discussão e definição das tecnologias a serem utilizadas no projeto: HTML, CSS, JavaScript (Bootstrap framework). |
| 02-06-2022 | Jodionísio Muachifi | Versão 01 - Web App: desenvolvimento da home page, menu baseando-se no protótipo, usando Bootstrap. |
| 03-06-2022 | Jodionísio Muachifi | Versão 02 - Web App: desenvolvimento das outras páginas de conexão com menu. |
| 04-06-2022 | Jodionísio Muachifi | Versão 03 - Web App: desenvolvimento de páginas de pesquisa individual para tenda, caravana e Camping RV. |
| 04-06-2022 | João Rodrigues, João Mourão | Desenvolvimento em JavaScript. Verificar disponibilidades. |
| 05-06-2022 | João Rodrigues, João Mourão | Desenvolvimento em JavaScript. Implementação da interface da página de reservas e do sistema de reservas. |
| 06-06-2022 | João Rodrigues, João Mourão | Conclusão do desenvolvimento. |
| 08-06-2022 | Jodionísio Muachifi | Desenho das arquiteturas e descrição dos pontos 2.4 e 2.5 |
| 10-06-2022 | João Mourão | Ponto 3.1, 3.2, 3.3 e organização da secção 4. |
| 10-06-2022 | Jodionísio Muachifi | Sumarização da iteração 3 (funcionalidade especifica do incremento 1), descrição dos requisitos com impacto na arquitetura. |
| 12-06-2022 | Jodionísio Muachifi | Desenvolvimento de código criar conta/Login usando Local Storage JavaScript. |
| 16-06-2022 | Jodionísio Muachifi | Transformação Digital: Integração do Progressive Web Application para ter-se versão mobile. |
| 20-06-2022 | João Rodrigues, João Mourão | Desenvolvimento do sistema de reserva de atividades. |
| 20-06-2022 | Jodionísio Muachifi | Implementação de CI Testing com SonarCloud e Github Actions |
| 23-06-2022 | Jodionísio Muachifi | Descrição das caraterísticas da implementação de CI Testing, SonarCloud, teste com Selenium e Story Map. |
| 24-06-2022 | João Rodrigues | Ponto 5 do relatório . |

## Referências e recursos suplementares

[**Utilização do plugin da library JQuery para desenvolver código Js para adicionar item ao carrinho**](https://www.jqueryscript.net/tags.php?/Shopping%20Cart/)

[Utilização do FrameWork BootStrap para desenvolvimento da interface front-end](https://getbootstrap.com/)

[Utilização do Progressive Web Application](https://web.dev/)

Link para acessar a plataforma: [CampingSpot](https://www.campingspot.cf/code/)

# Arquitetura do sistema

## Objetivos gerais

* Os clientes podem aceder à plataforma em qualquer lado, a partir de um browser, sem necessidade de instalar software específico. Os clientes podem utilizar a plataforma CampingSpot em ambientes *desktop* e *mobile* (a partir de um Browser).
* A web App possui um sistema que integra recursos externos que atende pagamentos eletrónicos para suportar transações desmaterializadas. Dentre os tipos de pagamentos temos: Paypal, MBWay e Cartão Bancário Visa.
* Os clientes registados poderão receber sempre descontos de 10% em cada reserva de tendas, Camping RV ou Caravana. No entanto, devem verificar a disponibilidade dos lugares com antecedência na plataforma.

## Requisitos com impacto na arquitetura

| Requisitos | Descrição |
| --- | --- |
| RD-4 | A plataforma deve suportar o carregamento rápido mesmo usando JavaScript. Isto é possível pela utilização do Cloudflare. |
| RD-5 | Todos os serviços que suportam o sistema são configurados de forma a suportar Local Storage e para assegurar uptime. |
| RS-1 | A informação relativa a clientes (dados pessoais em caso de registrar-se, e pagamentos efetuados) deve seguir à LGPD.  A plataforma faz a utilização de certificados SSL fornecidos pelo Cloudflare, i.e., o domínio está totalmente customizado e seguro. |
| RU-3 | A plataforma deve-se ajustar para ter uma apresentação adequada ao ecrã, designadamente para *smartphones*, *tablets* ou sistemas de secretária. |

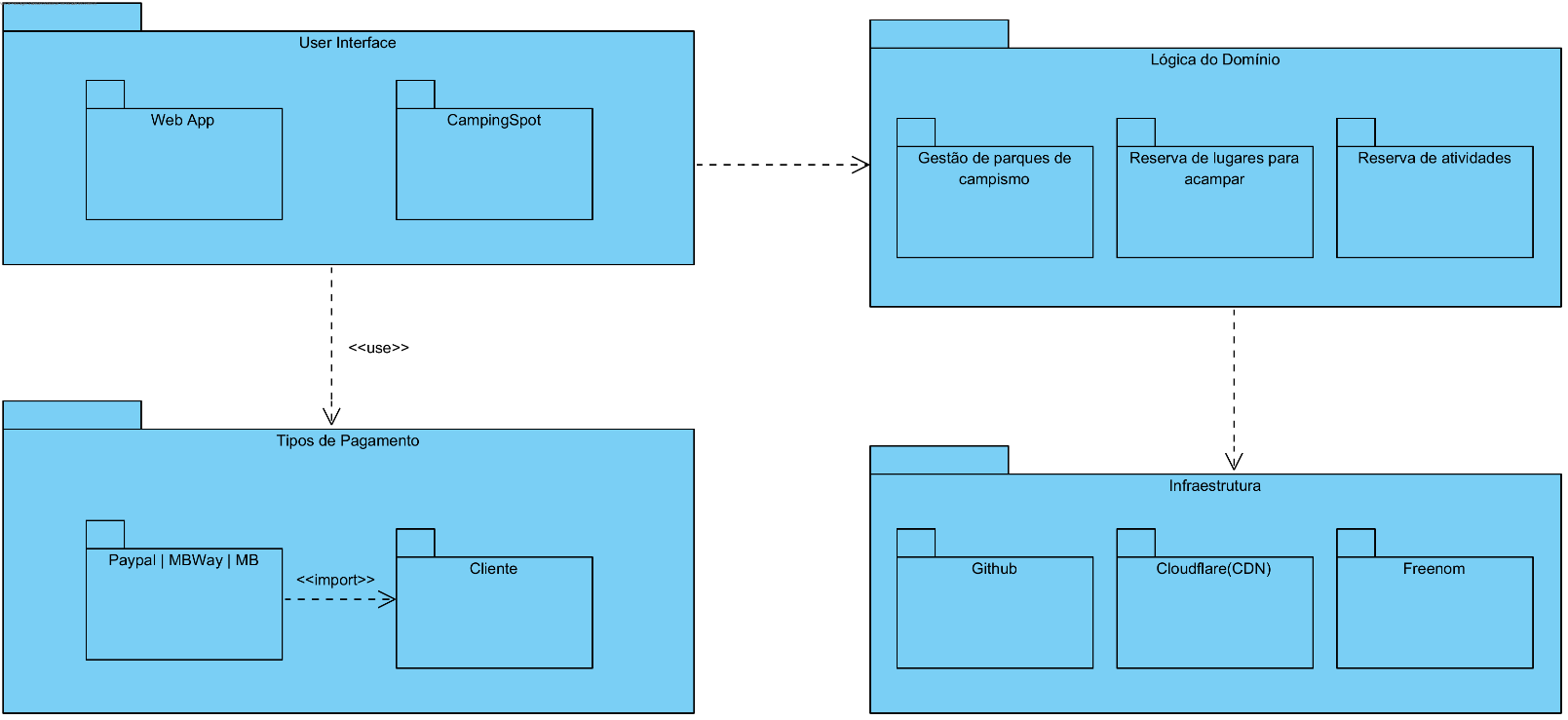
​

## Decisões e justificação

Tendo em conta os objetivos para a arquitetura, e os requisitos levantados na Análise, foram tomadas as seguintes decisões:

* Optamos por utilizar o HTML, CSS e JavaScript (incluindo Bootstrap framework) para a implementação do front-end porque estamos bem familiarizados com estas linguagens de programação.
* Não se utilizou diretamente tecnologias de desenvolvimento back-end, mas sim recursos externos como Cloudflare para distribuição rápida de conteúdo devido o carregamento lento do JavaScript. Além disso, os dados da web App foram geridos a partir do Local Storage a fim de facilitar o desenvolvimento do projeto e, não usamos base de dados nativas como MySQL, MariaDB, etc, porque nem todos elementos do grupo têm domínio de modelação de base dados relacionais (DBMS).

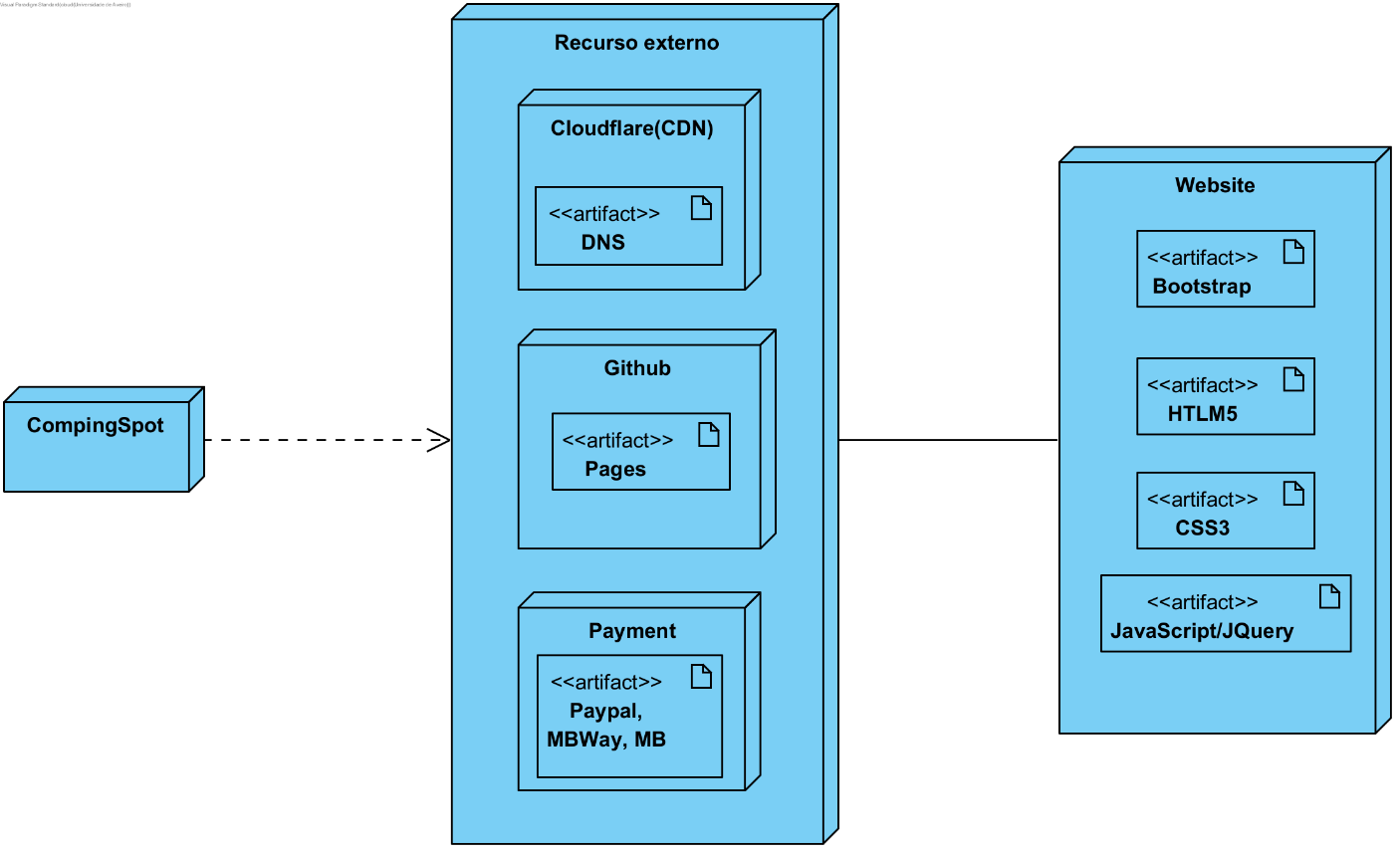
## Arquitetura do software



A articulação entre os módulos decorre da seguinte forma:

* Primeiramente, temos a interface do utilizador constituída pela web App do projeto (CampingSpot), do qual toda lógica do domínio vai depender e interagir com esta interface.
* De seguida. temos a infraestrutura do projeto composto por GitHub, Cloudflare e Freenom (onde adquirimos nosso domínio gratuitamente), do qual vai interagir e depender de lógica do domínio.
* Finalmente, temos os tipos de pagamento na plataforma, do qual vai depender e interagir com a interface do utilizador e por conseguinte o cliente vai depender que tipo de pagamento poderá escolher.

## Arquitetura física de instalação



A plataforma CampingSpot utiliza recurso externos para colocar a Web App em produto e. dentre eles inclui Cloudflare que fornece um serviço de distribuição de conteúdo para servidores mais próximos, GitHub onde hospedamos o código da plataforma e utilizamos o serviço de Pages a fim de customizar o DNS. A web App também utiliza recurso externo a fim efetuar dos clientes poderem realizar os devidos pagamentos logo após a reserva.

Para colocar-se a web App e seus serviços em produção, teve-se de escolher tecnologias web rápidas para o desenvolvimento do projeto tais como: HML5, CSS3, JavaScript/JQuery e o framework Bootstrap 5. Assim tornou-se possível nesta fase testar algumas funcionalidades dos requisitos funcionais da plataforma.

# Incremento 1

## Casos de utilização no Incremento 1

No primeiro incremento implementado, o foco esteve na validação da arquitetura proposta, através da implementação de funcionalidade representativa do *core* do negócio. Para isso, selecionámos e desenvolvemos o sistema de reservas e uma parte do sistema de pagamento, permitindo ao cliente fazer uma reserva online e o respetivo pagamento.

A especificação detalhada dos casos de utilização encontra-se em anexo (secção 5). A partir dessa análise, definiram-se as histórias de utilização a implementar.

## Histórias de utilização selecionadas

As histórias (*user stories*) incluídas nesta interação fazem parte do *backlog* do projeto, acessíveis em [link para backlog](https://1drv.ms/u/s!AjyXAfYgUMc3vMtSlSbmS8Dp_AMCWA?e=nx3A3c)

Histórias incluídas nesta interação:

|  |  |
| --- | --- |
| História/*use case slice* | Critérios de aceitação |
| **Quero fazer uma reserva online**  Sou um visitante do site do CampingSpot, quero pesquisar pela disponibilidade de uma reserva num período à minha escolha, escolhendo ficar numa tenda, caravana ou camping RV e escolhendo o número de pessoas que me acompanham. | **Cenário 1: Reserva disponível**  Estou no site do CampingSpot. Insiro a data de check-in como 14 de agosto e a data de check-out como 21 de agosto. Escolho ficar numa caravana e digo que seremos 4 pessoas nesta reserva.  Ao clicar em “pesquisar disponibilidade” sou redirecionado para outra página onde me é indicado o número de vagas e o preço a pagar do pedido feito anteriormente. Para além disso também é mostrada uma pequena descrição em relação às caravanas. De seguida, clico no botão “pagamento” e sou novamente redirecionado para outra página onde aparecem os detalhes da reserva, uma zona para preencher informações em relação a mim e uma outra zona para escolher o método de pagamento. Por fim, clico no botão “pagar” onde sou redirecionado para a página de pagamento.  **Cenário 2: Reserva indisponível**  Estou no site do CampingSpot. Insiro a data de check-in como 14 de agosto e a data de check-out como 21 de agosto. Escolho ficar numa tenda e digo que seremos 4 pessoas nesta reserva.  Ao clicar em “pesquisar disponibilidade” surge um pop-up que diz não haver vagas suficientes para esta reserva. |

## Estratégia e estado da implementação

Foi implementado o sistema de reservas, de modo que já é possível fazer reservas online e verificar a disponibilidade destas reservas. Foi também implementado o sistema de pagamento. No entanto, este ainda não está totalmente completo.

Para o desenvolvimento da interface foi utilizado HTML5, CSS3 e JavaScript/JQuery. Para facilitar a utilização destas tecnologias utilizámos Bootstrap 5. Não utilizámos nenhuma tecnologia diretamente back-end no projeto e para compensar isto recorremos a JavaScript e à utilização da Local Storage para servir como base de dados.

Utilizou-se o GitHub como sistema de versionamento de código assim como para hospedagem do mesmo. Optou-se na escolha do Visual Studio Code como editor para o desenvolvimento do projeto e, Jira como software para o monitoramento das tarefas e acompanhamento do projeto.

**Como conclusão, estas soluções trazem para a arquitetura rapidez e facilidade no trabalho que se está a desenvolver e, tudo isto, à custo zero**.

# Incremento 2

## Casos de utilização no incremento 2

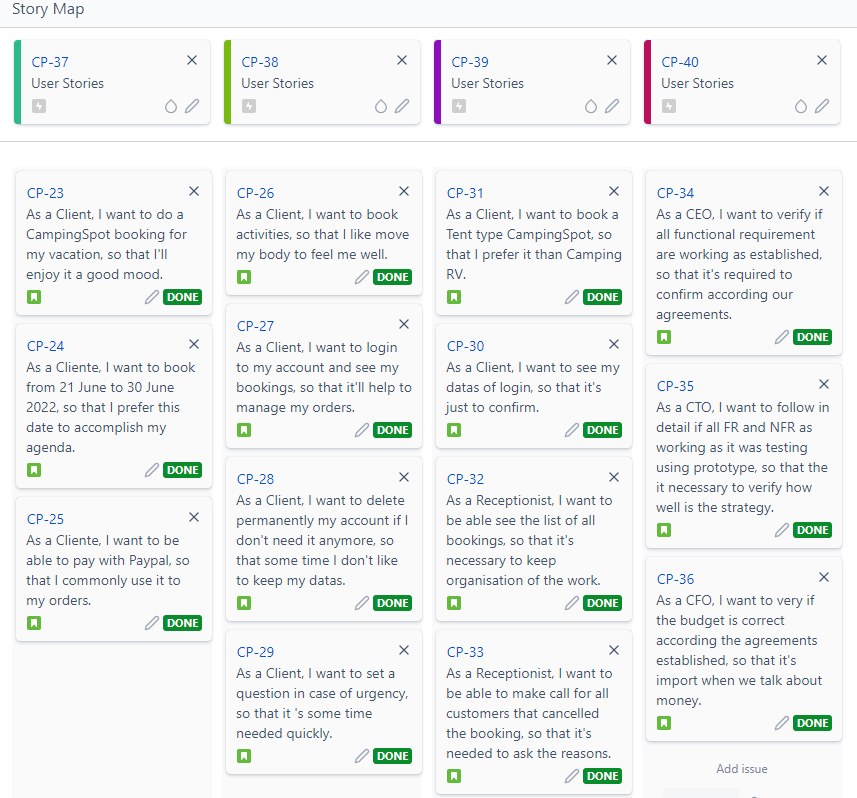
Neste projeto tivemos como prioridade o desenvolvimento do sistema de reservas. Neste incremento foram trabalhados os casos de utilização relacionados com o sistema de atividades e para além destes também houve um foco no desenvolvimento de um sistema de login/registro e numa versão da aplicação web para dispositivos móveis.

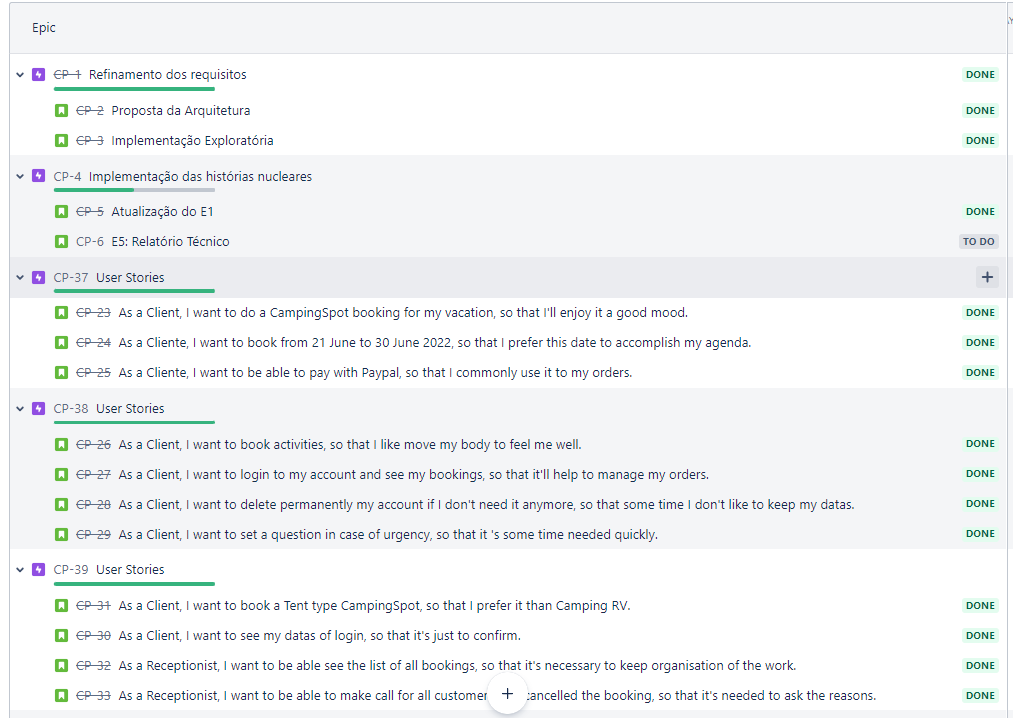
A especificação detalhada dos casos de utilização encontra-se em anexo (secção 5).

## Histórias de utilização selecionadas

Selecionou-se simuladamente user stories do Cliente, CEO, CTO, e CFO para criar histórias da plataforma CampingSpot, Story Map e RoadMap.

Histórias desenvolvidas nesta interação:





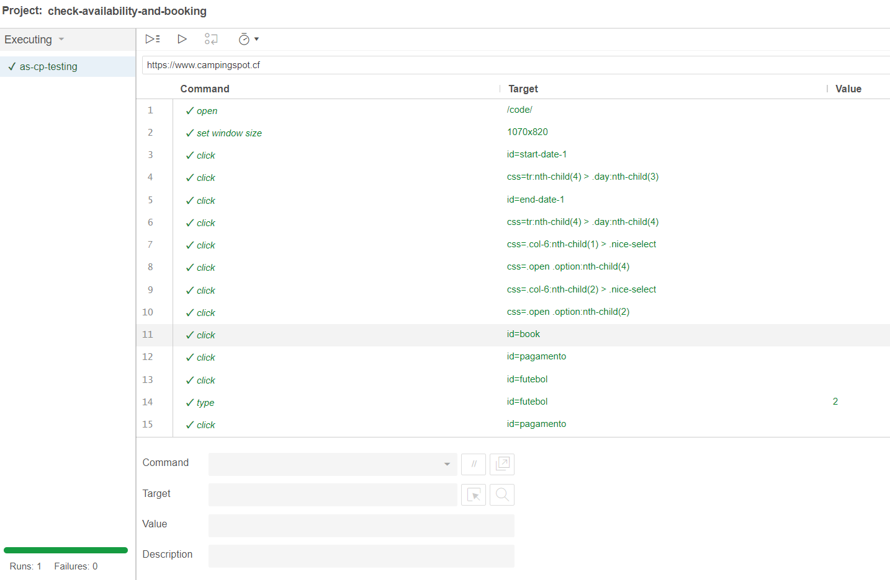
## Aceitação e garantia de qualidade

Para garantia de qualidade na plataforma desenvolvida utilizou-se Continuous Integration (CI) para testar os bugs existentes no código usando SonarCloud, Github Actions e Discord para receber Push Notification. Além disso, utilizou-se teste com Selenium a fim de fazer o record/run do website e não só. Conforme foi recomendado, o ficheiro .side está na pasta test suites.

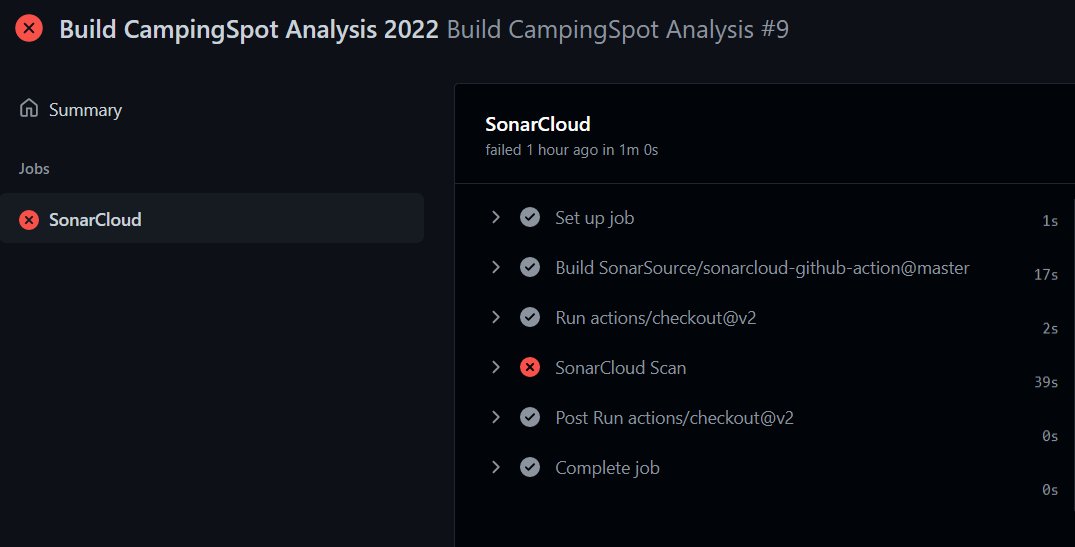
Testes automáticos são importantes para acelerar a produtividade no que concerne o desenvolvimento usando metodologias ágeis. Quanto mais automatiza-se as tarefas, maior é a produtividade e, por conseguinte, haverá confiabilidade nos códigos desenvolvidos.

A integração destes testes não é uma tarefa difícil. Apenas com pouco conhecimento aprendido e recomendações apropriadas dadas pela SonarCloud e Github, foi simples a implementação do CI Testing. Para valorizar, aplicamos cuidadosamente as instruções obtidas workshop com Nokia Company assim o ênfase dado pelo professor no que tange CI/CD e, sua tendência no mercado de trabalho.

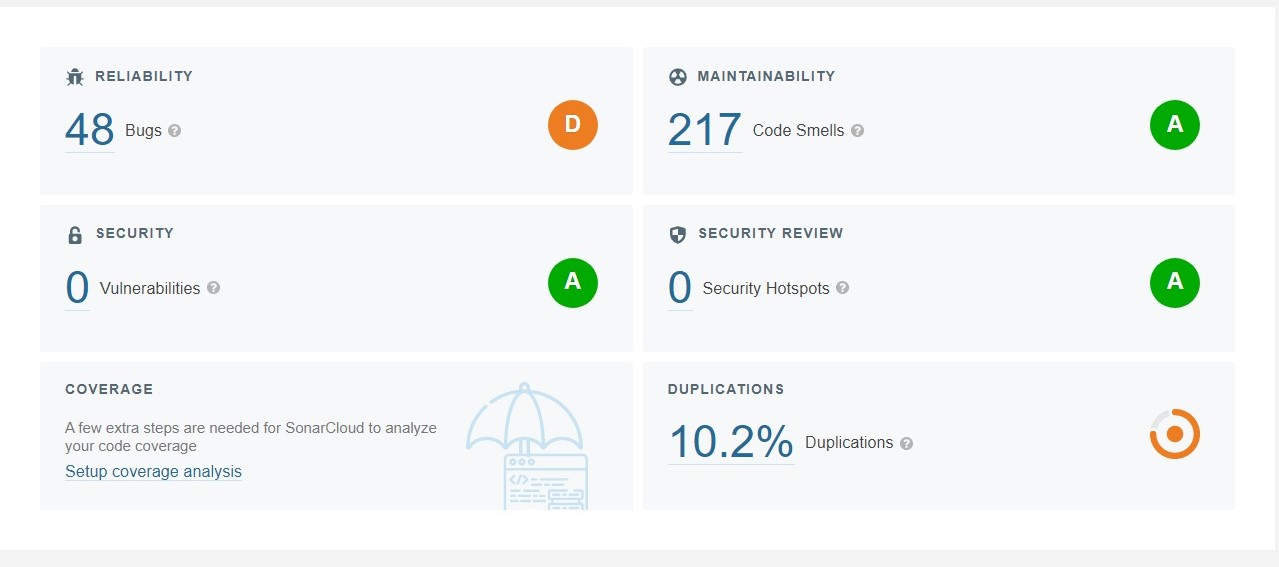
* Teste com Selenium



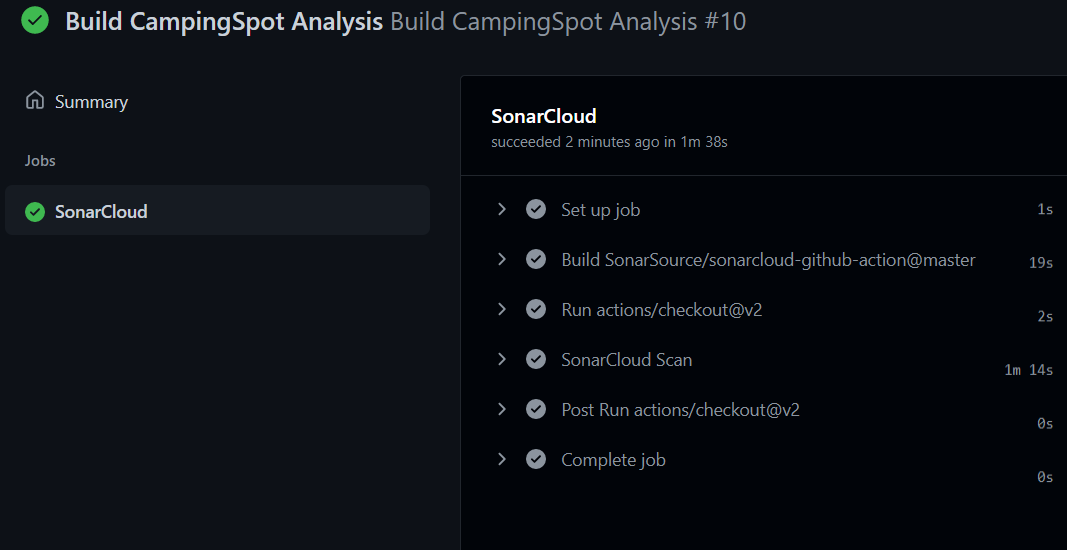
* Continuous Integration com SonarCloud e Github Actions



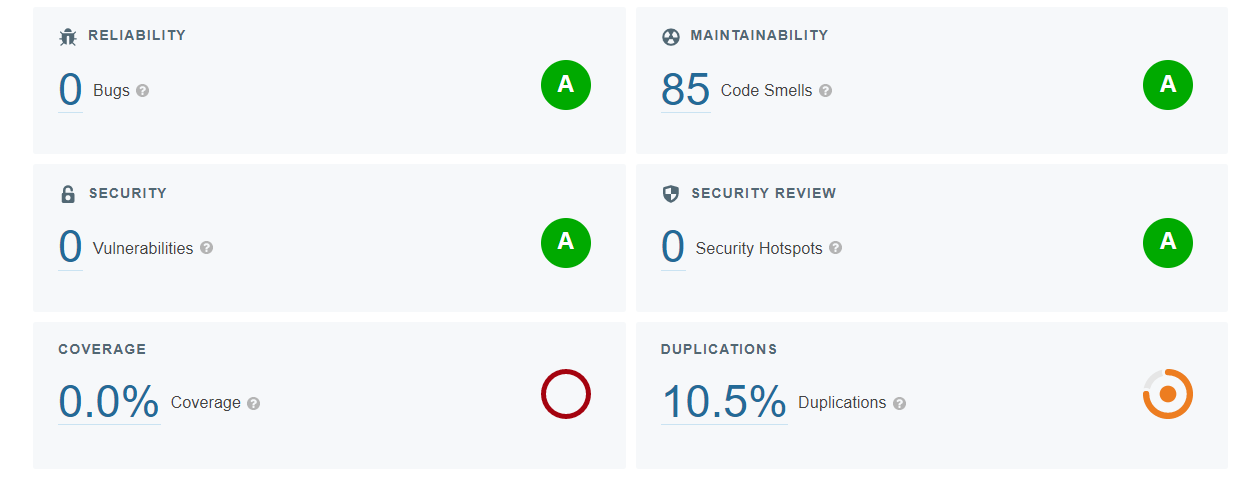
Antes da correção de Bugs



SonarCloud-Bugs encontrado



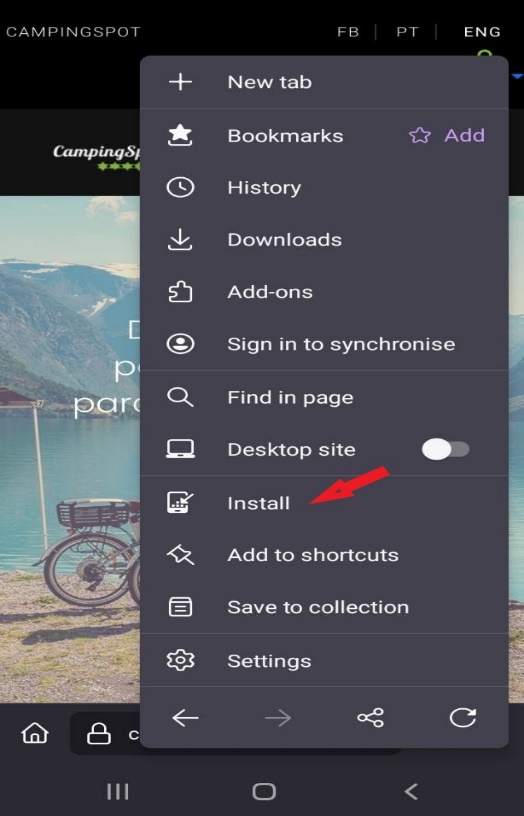
Depois de corrigir Bugs



SonarCloud - Bugs Corrigido

* Transformação digital
* Empresas que expandem o seu negócio para o mundo Digital.
* Possibilidade de efetuar marcação de férias através de uma aplicação mobile.

No que concerne a digitalização da plataforma CampingSpot, implementou PWA (Progressive Web Application), que é uma evolução híbrida entre as páginas web e uma aplicação mobile, i.e, PWA permite empacotar todas páginas web desenvolvidas e ser instalada no telemóvel usando um navegador.



1



2



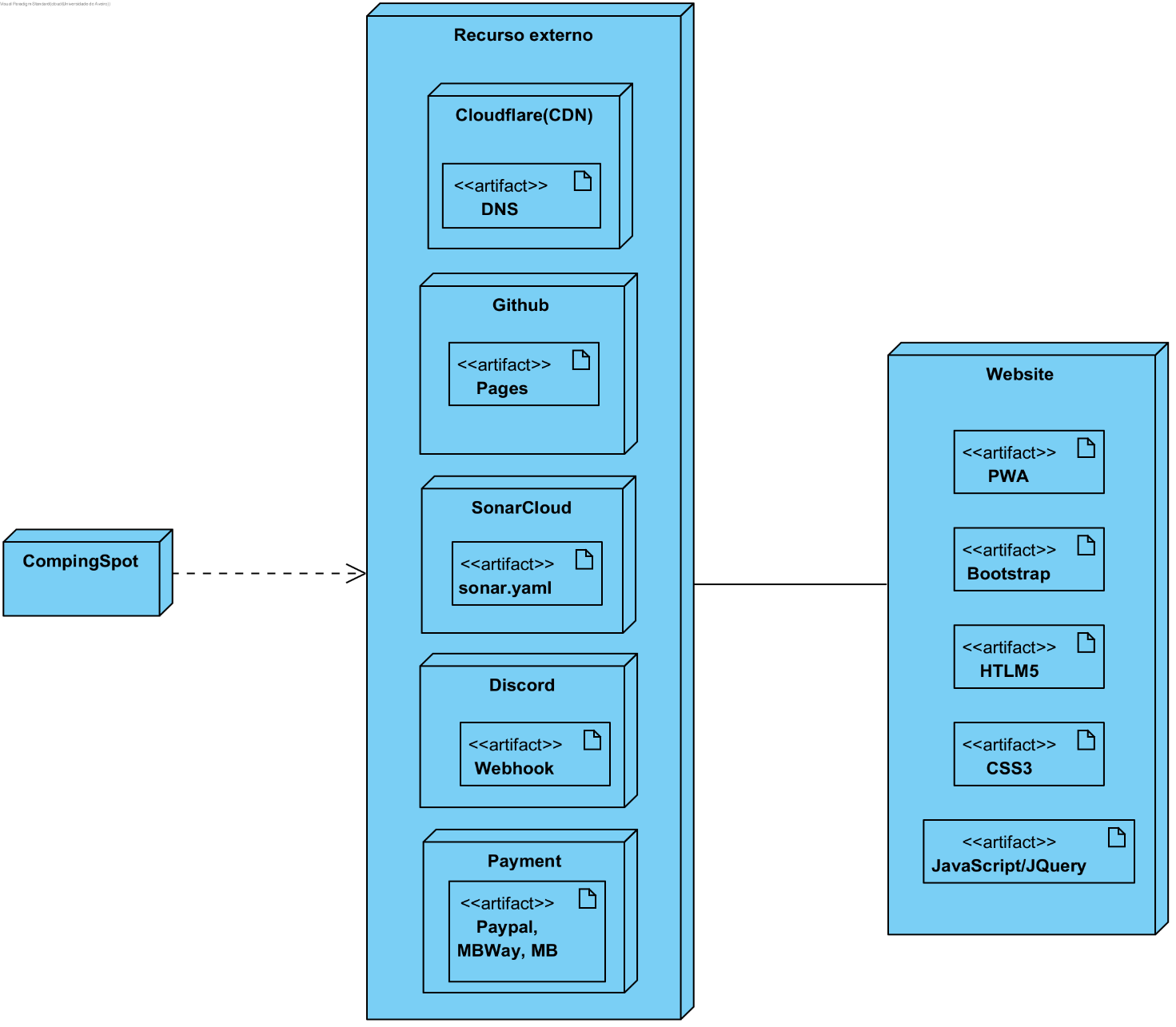
3



4

* Atualização do Deployment Diagram

Neste incremento acrescentou-se no Deployment Diagram no sistema externo como SonarCloud e Discord para Push Notification assim PWA como uma tecnologia para termos uma aplicação mobile hibrida tal como mostra figura abaixo.



## Estado da implementação

Neste incremento foram implementados um sistema de atividades e um sistema de login/registro. O sistema de atividades não está completamente operacional, pois este não está conectado com a base de dados (local storage) do sistema de reservas. O sistema de login/registro está pronto e resume-se à criação de uma conta ou login caso esta já esteja criada. Também foi implementada uma página de perfil de usuário, estando, no entanto, o backend desta página em falta. Para além disso foi feita uma versão da aplicação web para dispositivos móveis. Uma coisa que ficou por implementar foi o backend do sistema de pagamento.

Para aceder ao produto online clique [aqui](https://www.campingspot.cf/code/login.html)

Apêndice

# Especificação dos casos de utilização

### Pacote: sistema de resevas

### CaU 1 Fazer reserva online

|  |  |
| --- | --- |
| Caso de | Fazer reserva online |
| Breve descrição | O cliente verifica a disponibilidade para o intervalo de datas pretendido e, se estiver disponível prossegue para a reserva. |
| Pré-condições: | O cliente tem de ter conta no site e disponibilidade na data pretendida. |
| Pós-condições | Será necessário ser feito um pagamento posteriormente. |
| Fluxo base: | **1.Criar conta/login**  Para poder aceder às funcionalidades do site o utilizador terá de fazer login, ou se não tiver conta criar uma.    **2. Verificar disponibilidade**  Na página inicial existe uma barra que irá  pedir para inserir uma data inicial e uma data final onde o utilizar indicará o início e o fim da estadia, também terá de escolher que tipo de aluguer a fazer (caravana, tenda...) e a quantidade.  **3.Reserva**  O cliente depois de receber a confirmação da disponibilidade da data será redirecionado para uma página que indicará o número de vagas para o equipamento pretendido com o botão para prosseguir por baixo. Ao clicar neste encontrar-se-á numa página com os detalhes da sua compra e um formulário a preencher com as suas informações e onde terá de escolher o método de pagamento. |
| Fluxos alternativos: | Na página inicial encontram-se outros botões não tão visíveis como a barra de pesquisa que darão conexão a outra página que verifica disponibilidades. |

### CaU 2 Fazer reserva online de atividades

|  |  |
| --- | --- |
| Caso de | Fazer reserva online de atividades |
| Breve descrição | O cliente tem a opção de comprar algumas entradas que dão acesso a atividades a serem realizadas durante a sua estadia |
| Pré-condições: | O cliente tem de ter conta no site e disponibilidade na data pretendida. |
| Pós-condições | Será necessário ser feito um pagamento posteriormente. |
| Fluxo base: | **1.Criar conta/login**  Para poder aceder às funcionalidades do site o utilizador terá de fazer login, ou se não tiver conta criar uma.    **2. Verificar disponibilidade**  Na página inicial existe uma barra que irá  pedir para inserir uma data inicial e uma data final onde o utilizar indicará o início e o fim da estadia, também terá de escolher que tipo de aluguer a fazer (caravana, tenda...) e a quantidade.  **3.Reserva**  O cliente depois de verificar a disponibilidade e ter prosseguido na compra poderá comprar alguns ingressos para algumas atividades, apenas terá de escrever na caixa de texto a quantidade que pretende adquirir e terá de clicar em no botão “Pagamento”. Ao clicar neste encontrar-se-á numa página com os detalhes da sua compra e um formulário a preencher com as suas informações e onde terá de escolher o método de pagamento. |