



## **Relatório Técnico – Aplicação Web Atlantis Burguer**

**Disciplina: Desenvolvimento de Sistemas em Nuvem**

**Professor: Aparecido Valdemir de Freitas**

**Gabriel Campari (RA: 8145921)**

**Gabriel Imolene (RA: 8154847)**

**Glenda Carvalho (RA: 8149720)**

**Gustavo Almeida (RA: 8145857)**

**Kaue Barbi (RA: 8124571)**

**Lívia Mezashi (RA: 8144678)**

**29 de Maio de 2025**

## ÍNDICE

1. Introdução	3
2. Arquitetura e Desenvolvimento da Aplicação	4
2.1. Visão Geral e Objetivos	4
2.2. Tecnologias Empregadas	5
2.3. Processo de Desenvolvimento	5
3. Containerização da Aplicação com Docker	5
3.1. Configuração do Dockerfile	6
Um Dockerfile foi criado na raiz do projeto para definir a imagem da aplicação:	6
3.2. Passo a Passo da Containerização	6
4. Hospedagem da Aplicação Containerizada em Nuvem	7
5. Conclusão	8
6. Anexos	9

## RESUMO

Este relatório apresenta de forma detalhada o processo completo de desenvolvimento, containerização e implantação em nuvem da aplicação web "Atlantis Burger". A aplicação consiste em um site estático, desenvolvido com HTML, CSS e JavaScript, projetado para promover uma hamburgueria. Após o desenvolvimento, o site foi empacotado em um container Docker, utilizando o servidor Nginx para servir os arquivos estáticos de forma eficiente.

A imagem resultante foi publicada no Docker Hub, tornando-a acessível para qualquer ambiente que suporte Docker. Em seguida, a aplicação foi implantada na nuvem utilizando a plataforma Render, o que permitiu sua disponibilização pública por meio do link: <https://atlantisburger-app-latest.onrender.com>.

Durante o processo, foram tomadas decisões arquiteturais visando a escalabilidade, portabilidade e simplicidade de manutenção da aplicação. Este documento detalha cada etapa técnica do projeto, desde a estruturação inicial do código-fonte até a publicação do container em um ambiente de produção na nuvem.

## 1. INTRODUÇÃO

Este relatório apresenta o processo completo de desenvolvimento, containerização e implantação em nuvem da aplicação web Atlantis Burguer, uma solução projetada para modernizar a presença digital de uma hamburgueria artesanal por meio de um site estático, responsivo e de fácil manutenção. O projeto foi desenvolvido no contexto da disciplina Desenvolvimento de Sistemas em Nuvem, com o objetivo de aplicar, na prática, conceitos fundamentais de arquitetura web moderna, containerização com Docker e implantação em plataformas de nuvem pública.

O desenvolvimento teve início com a criação visual da aplicação utilizando a ferramenta Figma, garantindo uma interface atrativa e alinhada à identidade da marca. A aplicação foi construída com HTML5, CSS3 e JavaScript no front-end, com páginas organizadas de forma modular e servidas estaticamente. Essa abordagem visa facilitar a manutenção e garantir uma boa experiência de uso em diferentes dispositivos.

Para a entrega da aplicação ao navegador, foi configurado um servidor web Nginx, executado dentro de um container Docker. Foi criado um Dockerfile personalizado que possibilitou a construção local da imagem da aplicação, execução de testes funcionais via Docker e posterior publicação no Docker Hub. Essa etapa garantiu o versionamento e o acesso remoto à aplicação de forma padronizada e portátil.

Na etapa final, a imagem Docker foi implantada na plataforma Render.com, que ofereceu um ambiente simplificado para deploy com provisionamento automático de infraestrutura. Utilizando a funcionalidade Deploy from a Docker registry, foi possível transformar a imagem publicada em um serviço web totalmente funcional e acessível ao público. Ao longo deste relatório, são detalhadas as decisões técnicas e operacionais tomadas em cada fase do projeto, evidenciando como a integração entre tecnologias front-end, containerização e hospedagem em nuvem pode resultar em uma solução escalável, eficiente e profissional.

## 2. Arquitetura e Desenvolvimento da Aplicação

### 2.1. Visão Geral e Objetivos

A aplicação foi concebida como um website estático para uma hamburgueria artesanal, visando modernizar sua presença digital. Os principais objetivos eram fornecer uma interface responsiva, intuitiva e visualmente atraente para apresentação do cardápio, informações do estabelecimento e contato. A arquitetura da aplicação é baseada em arquivos front-end servidos estaticamente.

### 2.2. Tecnologias Empregadas

As seguintes tecnologias foram utilizadas no desenvolvimento do front-end:

- HTML5: Estruturação do conteúdo.
- CSS3: Estilização e layout responsivo.
- JavaScript: Interatividade no lado do cliente.
- Figma: Design e prototipação da interface.

## 3. Containerização da Aplicação com Docker

A containerização foi realizada para garantir portabilidade e consistência, utilizando Docker e Nginx.

### 3.1. Configuração do Dockerfile

Um **Dockerfile** foi criado na raiz do projeto para definir a imagem da aplicação:

```
Dockerfile X
backend > Dockerfile
1  #Usar uma imagem base do Nginx (versão alpine, leve)
2  FROM nginx:alpine
3
4  #Remover conteúdo padrão do Nginx
5  RUN rm -rf /usr/share/nginx/html/*
6
7  #Copiar arquivos do projeto
8  COPY . /usr/share/nginx/html/
9
10 #Expor a porta 80
11 EXPOSE 80
12
13 #Iniciar o Nginx assim que o container startar
14 CMD ["nginx", "-g", "daemon off;"]
15
```

### 3.2. Passo a Passo da Containerização

Construção da Imagem Docker: A imagem foi construída localmente com o comando:

```
sudo docker build -t atlantisburger-app .
```

Teste Local da Aplicação Containerizada: Um container foi iniciado para teste com o comando:

```
sudo docker run -d -p 8080:80 atlantisburger-app
```

A aplicação foi acessada para verificação em:

**<http://localhost:8080>**

Disponibilização da Imagem no Docker Hub: A imagem foi enviada para o Docker Hub para facilitar o acesso e deployment:

Login: `sudo docker login`

Tagging: `sudo docker tag atlantisburger-app  
gustavofacul/atlantisburger-app:latest`

Push: `sudo docker push gustavofacul/atlantisburger-app:latest`

A imagem está disponível em:

**[gustavofacul/atlantisburger-app:latest](#)**

## 4. Hospedagem da Aplicação Containerizada em Nuvem

A aplicação web foi empacotada como um container Docker e hospedada em nuvem utilizando a plataforma Render.com, escolhida pela sua facilidade de uso, integração direta com imagens Docker públicas e provisionamento automático de infraestrutura, o que permitiu realizar o deploy de forma simples e eficiente, sem necessidade de configurações complexas.

Para realizar a hospedagem, criamos um serviço do tipo Web Service na plataforma, utilizando a opção Deploy from a Docker registry e fornecendo a URL da imagem: `gustavofacul/atlantisburguer-app:latest`

Após isso, foi necessário aguardar o provisionamento automático da infraestrutura e a conclusão do deploy da aplicação.

Com o processo finalizado, a aplicação passou a estar disponível em uma URL pública: **<https://atlantisburguer-app-latest.onrender.com>** fornecida pela plataforma, permitindo o acesso remoto via navegador e a validação funcional do sistema.

## 5. Conclusão

O projeto de desenvolvimento e containerização da aplicação Atlantis Burger permitiu a aplicação prática de conceitos de desenvolvimento web, arquitetura de containers com Docker e Nginx, além da preparação para implantação em nuvem. Durante o processo, foi possível compreender a importância da portabilidade e consistência que os containers oferecem, simplificando o ciclo de vida da aplicação desde o desenvolvimento local até a distribuição. Dessa maneira, o projeto teve um papel fundamental na ampliação da experiência e no aprimoramento dos conhecimentos adquiridos ao longo do processo.



## 6. Anexos

- LINK - imagem no Docker HUB:  
<https://hub.docker.com/r/gustavofacul/atlantisburguer-app>
- LINK - repositório do projeto GitHub:  
<https://github.com/gusta-ve/AtlantisBurguer-site>