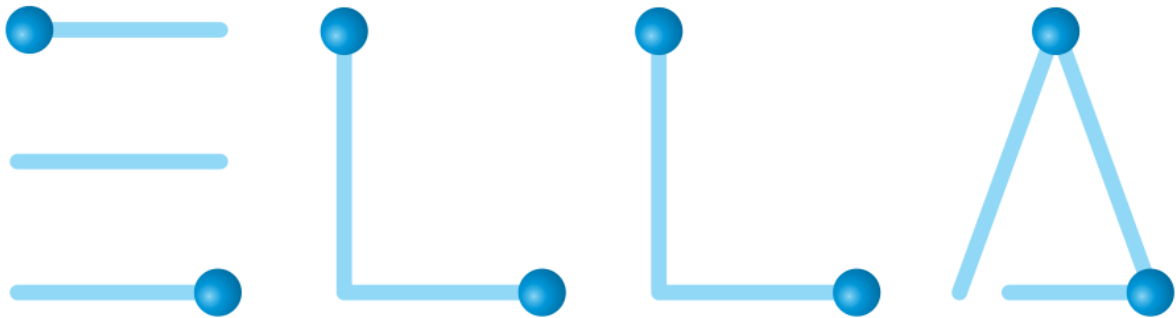




Universidade Federal de Uberlândia  
Instituto de Letras e Linguística  
Equipe ELLA



**English Language  
Learning Laboratory**

# Manual de Implantação

## Histórico de Revisões

Versão	Data	Autor	Requerimento, Seção, Página, Motivo
1.0	31/07/2019	Fabiano Silvério	Criação do Documento

## Documentos Associados

Versão	Documento	Descrição
1.0	ELLA- Manual do Usuário	Manual do Usuário
1.0	ELLA- Termo de Uso	Termo de Uso da Aplicação ELLA

## Glossário

Termo	Definição
IA	Inteligência Artificial
ELLA	English Learning Laboratory
API	Application Programming Interface
Docker	Tecnologia de software que fornece uma camada adicional de abstração e automação de virtualização de nível de sistema operacional, possibilitando o empacotamento de uma aplicação ou ambiente inteiro dentro de um container, o que torna-se portátil para qualquer outro host que contenha outro docker instalado.
Tacotron	Sintetizador de voz da Google
SMTP	Simple Mail Transfer Protocol

---

## 1. OBJETIVOS

O objetivo deste documento é apresentar os passos necessários para a implantação do laboratório virtual ELLA. O laboratório é composto por um conjunto de sistemas que se comunicam, e deverão ser instalados conforme as instruções deste manual.

### 1.1. Laboratório virtual

O laboratório virtual é o site que os alunos irão acessar para assistir às vídeo-aulas e fazer os exercícios.

### 1.2. Site de Coleta

O site de coleta é um site responsável pela coleta de vídeos para alimentar a IA para reconhecimento de fala na língua inglesa. Esse site foi desenvolvido em PHP. Este site deverá ser executado num servidor Apache e depende também de um banco de dados MySQL.

### 1.3. Aplicação de reconhecimento facial para Login

A aplicação de reconhecimento facial é um serviço que realiza o treinamento e geração dos arquivos de “encodings” das faces coletadas no cadastro do usuário e o reconhecimento facial no procedimento de login do sistema, implementada em Python e hospedada em container Docker.

### 1.4. API de verificação de emoções da Voz e da Face

A API de verificação de emoções é um serviço em Java que realiza a comunicação do lado cliente com uma aplicação de análise de emoções do lado do servidor. Esta aplicação funciona como um barramento, abstraindo a chamada aos serviços de análise de emoções, hospedada em containers docker.

### 1.5. Aplicação de análise de emoção da Voz

A aplicação de análise da Voz é um serviço que realiza o treinamento, geração e análise acústica da emoção dos arquivos de áudio coletadas nas atividades de pronúncia do laboratório virtual, implementada em Python e hospedada em container Docker.

### 1.6. Aplicação de análise de emoção da Face

A aplicação de análise da emoção da Face é um serviço que realiza o treinamento, geração e análise de emoção da face coletadas nas atividades de pronúncia do laboratório virtual, implementada em Python e hospedada em container Docker.

---

## 1.7. Aplicação de ChatBot

A aplicação de chat bot oferece um serviço de assistente virtual do laboratório ELLA, com histórias relacionadas a aprendizagem da língua inglesa no laboratório, baseada no projeto RASA boilerplate.

## 1.8. Aplicação de análise de pronúncia de Fonemas

A aplicação de análise de pronúncia da Face é um serviço que realiza o treinamento, geração e análise de pontos de marcação da face coletadas nas atividades de pronúncia de fonemas do laboratório virtual, implementada em Python e hospedada em container Docker.

## 2. PRÉ REQUISITO PARA INSTALAÇÃO

Para implantação do projeto, o ambiente da aplicação ELLA deverá estar configurado com os seguintes aplicativos:

- PHP 7.2 ou superior;
- Apache 2.4 ou superior;
- MySql 8.0 ou superior
- Java versão 8 ou superior;
- Docker versão 18.09 ou superior;
- Python versão 3
- Tacotron

## 3. PROCEDIMENTOS DE INSTALAÇÃO

Para realizar a instalação do laboratório virtual, deve-se seguir os seguintes passos:

### 3.1. Instalação do Site de Coleta

#### 3.1.1. Pré-requisitos

- É necessário ter uma instalação do Docker sendo executada no servidor onde será instalado o site de coleta.
- É necessário ter um servidor SMTP configurado.

#### 3.1.2. Instalação

1. No servidor, criar um diretório **site-coleta** no qual será instalado o site.
2. Acessar o repositório do projeto <https://github.com/ileelufu/labileel/releases/tag/v1.0>

- 
3. Fazer download do projeto [Source Code \(zip\)](#) ou [Source Code \(tar.gz\)](#) e salvar o arquivo na pasta que foi criada no passo 1.
  4. Descompactar o arquivo que foi baixado no passo 3, de forma que o conteúdo do arquivo fique no diretório criado no passo 1.
  5. Certifique-se que o docker esteja sendo executado
  6. Abrir uma janela do terminal, console ou powershell
  7. Acessar, a partir do terminal, o diretório `labileel-1.0` que foi criado após a extração no passo 4.
  8. Executar o comando `docker-compose up` e aguardar o docker instalar e subir os containers necessários.

### 3.1.3 Pós-requisitos

- Criação do container Docker do site de coleta

## 3.2. Instalação da aplicação ELLA

### 3.2.1. Pré-requisitos

- Aplicações Apache, PHP e MySQL instaladas no servidor.

### 3.2.2. Instalação

1. Clonar o repositório GIT da aplicação <https://github.com/ileelufu/lab.git>.
  - a. Seguir os passos definidos no arquivo README.md

### 3.2.3 Pós-requisitos

- Aplicação ELLA implantada no servidor.

## 3.3. Instalação da aplicação de Login de Reconhecimento Facial

### 3.3.1. Pré-requisitos

- Aplicação Docker instalada no servidor.

### 3.3.2. Instalação

1. Clonar o repositório GIT da aplicação <https://github.com/ileelufu/facialRecognitionLogin.git>.
  - a. Seguir os passos definidos no arquivo README.md

### 3.3.3 Pós-requisitos

- Aplicação de Login de Reconhecimento Facial implantada em um container Docker no servidor.

---

## 3.4. Instalação da API de barramento de acesso a verificação de emoções da Voz e Face

### 3.4.1. Pré-requisitos

- Aplicação Java instalada no servidor.

### 3.4.2. Instalação

1. Clonar o repositório GIT da aplicação <https://github.com/ileelufu/EmotionAnalyzer.git>.
  - a. Seguir os passos definidos no arquivo README.md, sendo:
    - i. Criar diretório EmotionAnalyzer no servidor.
    - ii. Copiar o arquivo EmotionAnalyzer.jar para o diretório criado.

### 3.4.3 Pós-requisitos

1. API de conexão entre a aplicação LAB e os serviços de análise da emoção implantada no servidor.

## 3.5. Instalação da Aplicação de Análise da Emoção da Voz

### 3.5.1. Pré-requisitos

- Aplicação Docker instalada no servidor.

### 3.5.2. Instalação

1. Clonar o repositório GIT da aplicação <https://github.com/ileelufu/VoiceAnalyzer.git>.
  - a. Seguir os passos definidos no arquivo README.md

### 3.5.3 Pós-requisitos

- Aplicação de análise da Emoção da Voz implantada em um container Docker no servidor.

## 3.6. Instalação da Aplicação de Análise da Emoção da Face

### 3.6.1. Pré-requisitos

- Aplicação Docker instalado no servidor.

### 3.6.2. Instalação

1. Clonar o repositório GIT da aplicação <https://github.com/ileelufu/FaceAnalyzer.git>.
  - a. Seguir os passos definidos no arquivo README.md

---

### 3.6.3 Pós-requisitos

- Aplicação de análise da Emoção da Face implantada em um Container Docker no servidor.

## 3.7. Instalação do assistente virtual - chatbot ELLA

### 3.7.1. Pré-requisitos

- Aplicação Docker instalado no servidor.

### 3.7.2. Instalação

1. Clonar o repositório GIT da aplicação <https://github.com/ileelufu/ella-bot.git>.
  - a. Seguir os passos definidos no arquivo README.md

### 3.7.3 Pós-requisitos

- Aplicação chatbot implantada em um container Docker no servidor.

## 3.8. Instalação da Aplicação de Análise Facial da Pronúncia de Fonemas

### 3.8.1. Pré-requisitos

- Aplicação Docker instalado no servidor.

### 3.8.2. Instalação

1. Clonar o repositório GIT da aplicação <https://github.com/ileelufu/facialRecognitonPhoneme.git>.
  - a. Seguir os passos definidos no arquivo README.md

### 3.8.3 Pós-requisitos

- Aplicação de análise da Pronúncia de Fonemas da Face implantada em um container Docker no servidor.

## 3.9. Instalação do Tacotron

### 3.9.1. Pré-requisitos

- É necessário ter a instalação do Tensorflow para utilização do Tacotron. Para tal digite no terminal o comando: `pip install --upgrade tensorflow`
- O Tacotron também utiliza o Python3 para sua execução.

### 3.9.2. Instalação

- 
1. No terminal digite, git clone <https://github.com/keithito/tacotron.git> . Será criada uma pasta chamada tacotron, com todos os arquivos necessários
  2. Logo após: pip3 install -r /tacotron/requirements.txt
  3. Para realizar o download de um modelo pronto, digite : Digite: curl   
http://data.keithito.com/data/speech/tacotron-20180906.tar.gz | tar xzC /tmp
  4. Para inicialização do server, digite: python3 demo\_server.py --checkpoint /tmp/tacotron-20180906/model.ckpt
  5. Tudo pronto, basta acessar pelo browser: localhost:9000

**Equipe ELLA:** <https://labvirtual.ileel.ufu.br/lab>