

# User Guide

## Software Engineering

Team 2, SyntaxSentinels  
Mohammad Mohsin Khan  
Lucas Chen  
Dennis Fong  
Julian Cecchini  
Luigi Quattrociocchi

Table 1: Revision History

Date	Developer(s)	Change
4/4/2025	Mohammad Mohsin Khan and Lucas Chen	User Guide

# SyntaxSentinels User Guide

This guide covers the setup and management of the **SyntaxSentinels** system, including the Compute Server, Frontend, and Express Server. It includes installation instructions, environment configuration, common tasks, and debugging tips.

## Contents

<b>1 Overview</b>	<b>3</b>
<b>2 System Components</b>	<b>3</b>
2.1 Compute Server . . . . .	3
2.2 Frontend . . . . .	3
2.3 Express Server . . . . .	3
<b>3 Prerequisites</b>	<b>3</b>
<b>4 Setup Instructions</b>	<b>4</b>
4.1 Virtual Environment and Dependencies . . . . .	4
4.2 Environment Variables . . . . .	4
<b>5 Running the Servers</b>	<b>5</b>
5.1 Compute Server . . . . .	5
5.2 Frontend . . . . .	5
5.3 Express Server . . . . .	5
<b>6 Usage</b>	<b>5</b>
<b>7 Debugging and Troubleshooting</b>	<b>5</b>
<b>8 Additional Tips</b>	<b>6</b>

# 1 Overview

The SyntaxSentinels system is divided into three major components:

- **Compute Server:** Processes background jobs using Python and interacts with AWS (S3, SQS).
- **Frontend:** A client-facing application set up with Node.js.
- **Express Server:** Handles API requests and integrates with AWS and Firebase.

Each component has its own setup process and environment variables. This guide provides a step-by-step walkthrough for installation, configuration, and debugging common issues.

## 2 System Components

### 2.1 Compute Server

- **Language:** Python 3.11+
- **Key tasks:**
  - Virtual environment creation
  - Dependency installation using `requirements.txt`
  - Running the worker process to handle background jobs from an SQS queue

### 2.2 Frontend

- **Language:** JavaScript (Node.js)
- **Key tasks:**
  - Installing Node.js packages via `npm install`
  - Configuring environment variables for Auth0 authentication and API integration

### 2.3 Express Server

- **Language:** JavaScript (Node.js)
- **Key tasks:**
  - Installing Node.js packages via `npm install`
  - Configuring environment variables for authentication (Auth0), AWS services, and Firebase
  - Running the Express server for API endpoints

## 3 Prerequisites

- **Python 3.11+** (verify with `python --version`)
- **Node.js and npm** (verify with `node --version` and `npm --version`)
- A compatible shell:
  - Windows: Command Prompt or PowerShell
  - Linux/macOS: Standard terminal

## 4 Setup Instructions

### 4.1 Virtual Environment and Dependencies

#### Compute Server

```
cd backend
python -m venv .venv
```

Activate the environment:

**Windows Command Prompt:**

```
.venv\Scripts\activate.bat
```

**Windows PowerShell:**

```
.venv\Scripts\Activate.ps1
```

**Linux/macOS:**

```
source .venv/bin/activate
```

Install dependencies:

```
pip install -r requirements.txt
```

#### Frontend

```
cd frontend
npm install
```

#### Express Server

```
cd server
npm install
```

### 4.2 Environment Variables

Create a `.env` file for each component as described below.

#### Compute Server

Variable	Description	Example
AWS_REGION	AWS region location	us-east-1
AWS_ACCESS_KEY_ID	AWS access key ID	(none)
AWS_SECRET_ACCESS_KEY	AWS secret access key	(none)
S3_BUCKET_NAME	S3 bucket name	syntax-sentinels-uploads
SQS_QUEUE_URL	SQS job queue URL	https://sqs...
EXPRESS_API_URL	Express server URL	http://localhost:3000/api

#### Frontend

Variable	Description	Example
VITE_AUTH0_DOMAIN	Auth0 domain	myauth0domain.us.auth0.com
VITE_AUTH0_CLIENT_ID	Auth0 client ID	123EXAMPLE
VITE_AUTH0_AUDIENCE	Auth0 audience	https://myauth0domain...
VITE_API_URL	Express API URL	http://localhost:3001/api

## Express Server

(You can format the long environment variables in tabularx just like above, or break into multiple tables if needed.)

# 5 Running the Servers

## 5.1 Compute Server

1. Activate the virtual environment.
2. Set all environment variables.
3. Run the worker process:

```
python worker.py
```

## 5.2 Frontend

```
npm run dev
```

## 5.3 Express Server

```
npm start
```

# 6 Usage

# 7 Debugging and Troubleshooting

## 1. Virtual Environment Issues

- Make sure it's activated. If not:

```
source .venv/bin/activate
```

- Reinstall dependencies:

```
pip install --force-reinstall -r requirements.txt
```

## 2. Node.js Dependency Errors

```
npm install
rm -rf node_modules
npm cache clean --force
npm install
```

## 3. Environment Variable Issues

Double-check your `.env` files. Ensure sensitive values are correctly quoted and loaded.

## 4. Port Conflicts

Check which processes are using ports 3000, 3001 and kill or reconfigure as necessary.

## 5. AWS/Firebase Issues

Use:

```
aws sts get-caller-identity
```

## 8 Additional Tips

- Keep documentation updated.
- Use Git for configuration tracking.
- Backup sensitive files securely.
- Test components in isolation.
- Mirror environments between dev and prod.