**Learning Platform Data Visualization - Technical Documentation**

**Table of Contents**

1. \*\*Introduction\*\*

- Purpose

- System Overview

2. \*\*Installation and Setup\*\*

- Prerequisites

- Installation

- Configuration

- Database Setup

3. \*\*Usage Guide\*\*

- Starting the Application

- Importing Data

- Data Processing

- Data Visualization

- Data Analysis

- Exporting Results

4. \*\*Customization\*\*

- Modifying the Frontend

- Extending Data Analysis

- Integrating with Other Platforms

5. \*\*Troubleshooting\*\*

- Common Issues

- Debugging Tips

6. \*\*Contributing\*\*

- Reporting Issues

- Making Contributions

---

**1. Introduction**

Purpose

This technical documentation provides information on setting up, configuring, and using the Learning Platform Data Visualization project. It is designed to help users, developers, and administrators understand the technical aspects of the platform.

System Overview

The Learning Platform Data Visualization is a web-based platform that allows users to import, process, analyze, and visualize educational data. The platform offers features such as data import from various sources, data processing, customizable visualizations, and data analysis using machine learning techniques.

---

**2. Installation and Setup**

Prerequisites

Before setting up the platform, ensure you have the following prerequisites:

- Python 3.x installed

- Flask, SQLAlchemy, and other required Python libraries

- A MySQL database server

- Web browser for the frontend

Installation

1. Clone the project repository from .https://github.com/Astroeliezer/learncrafty

2. Navigate to the project directory using the command line.

3. Create a virtual environment (recommended) and activate it.

```bash

python -m venv venv

source venv/bin/activate # On Windows: venv\Scripts\activate

```

4. Install the required Python packages.

```bash

pip install -r requirements.txt

```

Configuration

1. Open the `config.py` file in the project directory.

2. Configure the database connection settings (e.g., MySQL).

3. Set the secret key for Flask.

4. Customize other configurations as needed.

Database Setup

1. Ensure your MySQL server is running.

2. Create a new database for the project.

```sql

CREATE DATABASE learning\_platform\_db;

```

3. Run the database migrations to create the necessary tables.

```bash

flask db init

flask db migrate

flask db upgrade

```

---

**3. Usage Guide**

Starting the Application

1. Open the command line and navigate to the project directory.

2. Activate the virtual environment if you haven't already.

3. Run the Flask application.

```bash

flask run

```

4. Access the platform in your web browser at ‘https://astroeliezer.github.io/learncrafty/

Importing Data

1. In the platform's interface, navigate to the "Import Data" section.

2. Click the "Choose File" button to select a data file for import.

3. Click the "Import" button to upload the data to the platform.

Data Processing

1. After importing data, the platform automatically processes it, handling missing values and creating new variables.

2. Processed data can be accessed through the API.

Data Visualization

1. In the "Create Visualization" section, select a visualization type (e.g., Bar Chart, Line Chart).

2. Customizable visualizations are created based on processed data.

Data Analysis

1. The platform offers data analysis features, including machine learning algorithms.

2. Perform data analysis through the "Analyze Data" button.

Exporting Results

1. Export visualization results in various formats (e.g., PNG, PDF) using the platform's export feature.

---

**4. Customization**

### Modifying the Frontend

Customize the frontend by editing HTML and CSS files to match your design preferences. Ensure the proper integration of JavaScript for user interactions.

### Extending Data Analysis

Expand data analysis capabilities by integrating additional machine learning algorithms and statistical methods to suit specific educational needs.

### Integrating with Other Platforms

Integrate the platform with other educational systems using APIs or data connectors to import data seamlessly from different sources.

---

**5. Troubleshooting**

Common Issues

- If the platform does not start, check your Python and package versions.

- Verify that the database connection settings are correct in the `config.py` file.

Debugging Tips

- Use debugging tools provided by Flask to identify and resolve application issues.

---

**6. Contributing**

Reporting Issues

If you encounter any bugs or have feature requests, please report them on the project's https://github.com/Astroeliezer/learncrafty

Making Contributions

Contributions to the project are welcome. Please follow the guidelines in the project's `CONTRIBUTING.md` file for submitting pull requests and code contributions.