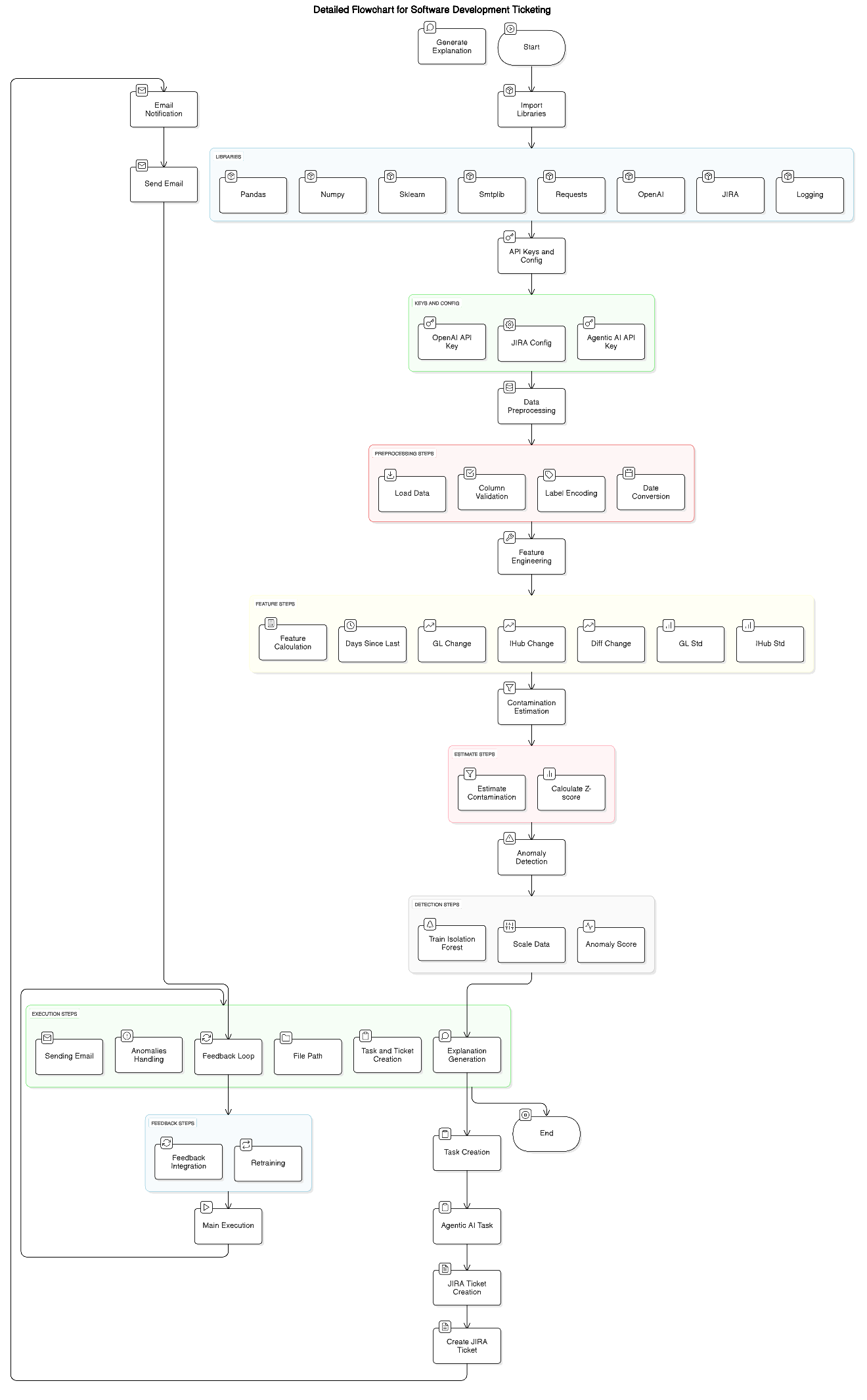
**Architecture diagram & explanation**



**🟣 MAIN COMPONENTS AND FLOW**

**1. Start and Initialization**

* **Start Node** → Kicks off the entire workflow.
* **Import Libraries**:
  + Essential Python libraries: Pandas, NumPy, Sklearn, Smtplib, Requests, OpenAI, JIRA, and Logging.

**2. API Keys and Configuration**

* Inputs required:
  + OpenAI API Key
  + JIRA Config
  + Agentic AI API Key  
    These are credentials and configurations needed for external service integrations.

**3. Data Preprocessing (Red Section)**

* **Load Data**: Reads the dataset, typically from CSV or database.
* **Column Validation**: Ensures required columns (fields) are present and formatted correctly.
* **Label Encoding**: Converts categorical text data into numerical format for ML algorithms.
* **Date Conversion**: Converts date columns to proper datetime objects.

**4. Feature Engineering (Yellow Section)**

Enhances raw data into meaningful features for better anomaly detection:

* **Days Since Last**
* **GL Change**
* **IHub Change**
* **Diff Change**
* **GL Std**
* **IHub Std**

Each feature helps describe transaction behavior in a way the model can understand.

**5. Contamination Estimation (Red Section)**

This block estimates how much of the dataset might be anomalies:

* **Estimate Contamination**: Uses feedback or heuristics to decide anomaly percentage.
* **Calculate Z-score**: Helps detect outliers based on statistical distribution.

**6. Anomaly Detection (Detection Steps)**

* **Train Isolation Forest**: Fits the model to the engineered features.
* **Scale Data**: Normalizes or standardizes features for better model performance.
* **Anomaly Score**: Outputs an anomaly score to classify normal vs. abnormal instances.

**7. Execution Steps (Green Section)**

* **Sending Email**: Sends email notifications with anomaly reports.
* **Anomalies Handling**: Processes detected anomalies for action.
* **Feedback Loop**: Gathers user validation of anomalies (whether they are true/false positives).
* **File Path**: Saves intermediate or final outputs.
* **Task and Ticket Creation**: Triggers creation of tasks (Agentic AI) and tickets (JIRA).
* **Explanation Generation**: Uses OpenAI LLM to describe and explain anomalies in human-readable form.

**8. Task Creation (End Steps)**

* **Task Creation**
  + **Agentic AI Task**: Creates an automated task in the Agentic AI platform.
  + **JIRA Ticket Creation**: Creates a ticket in JIRA.
* **Create JIRA Ticket**: Completes ticket creation for detected anomalies.
* **End Node**: Marks the end of the primary workflow.

**9. Feedback Integration (Blue Section)**

* **Feedback Integration**: Incorporates user feedback (supervised labels) into future model retraining.
* **Retraining**: Updates the model using feedback for continuous learning.
* This feeds back into **Main Execution**, improving accuracy over time.

**10. Email Notification (Left Node)**

* **Send Email**: Separate node that sends out anomaly reports after ticket creation or feedback processing.