

RoostGPT Permissions

This document outlines the permissions and licenses related to the use of third-party components within RoostGPT.

Introduction

RoostGPT integrates with various third-party components to enhance its functionality and provide a comprehensive user experience. We value transparency and wish to provide clarity on the permissions required to ensure seamless integration with these external components.

Third-Party Components and Permissions

1. Component Name: Git

- **Description:** Git is a distributed version control system.
- **Supported Types:** Github, Gitlab, Bitbucket, Azure DevOps (Both Cloud and Self Hosted)
- **Purpose in Our Software:** RoostGPT uses Git for reading the source code and create automated test for it and create the PR in the same repo for the tests.
- **Required Permissions:**
 - User Read.
 - Repo Read.
 - Repo Write.
 - Create Commit.
 - Create PR.

2. Component Name: Jira/Azure Board

- **Description:** It's used by development teams to track bugs, enhancements, tasks, and other kinds of issues throughout the software development lifecycle.

- **Purpose in Our Software:** RoostGPT fetch tickets based on commits message and use that as a acceptance criteria for the automated test generation.
- **Required Permissions:**
 - API Access
 - Tickets Read.
 - Comments Write

3. Component Name: Log Server

- **Description:** A log server is a centralised system designed to collect, store, and manage logs from various sources, including applications, systems, devices, and network infrastructure. By consolidating log data in one place, log servers facilitate easier monitoring, analysis, and troubleshooting.
- **Supported Types:** LogStash, Log File
- **Purpose in Our Software:** RoostGPT fetch logs from the log server based on the input request to enhance the test generated by using the real life data.
- **Required Permissions:**
 - API Access
 - Logs Read.

4. Component Name: AI Model

- **Description:** An AI Model is an advanced computational model trained on vast amounts of text data to understand and generate human-like language. Leveraging deep learning techniques, LLMs can comprehend context, answer questions, and assist in various language-related tasks, showcasing the pinnacle of natural language processing capabilities.
- **Supported Types:** OpenAI, Vertex AI, Azure OpenAI, LLAMA2, Starchat and other open source models.
- **Purpose in Our Software:** RoostGPT uses AI model to generate automated test case for the source code.
- **Required Permissions:**
 - API Access.

Revision #3

Created 21 August 2023 07:32:51 by Divyesh

Updated 21 August 2023 08:49:52 by Divyesh