**SWAD LOGGING FRAMEWORK**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Content** | **Author** | **Traceability** |
| **1.1** | **Initial version** | **Akhil Suresh**  **Amudha Bharathi**  **Anjana Rajeevan**  **Arun Sasi**  **Midhun Mohan C**  **Ujwal V John** | **TBD** |

1. **INTRODUCTION**

This Software architecture Design (SD) document describes the system architecture of Application Logging System. The Logging framework allows software-applications to record their internal states and processes depending on the Log level.

* 1. **Purpose**

The purpose of this document is to provide a comprehensive software architecture design for a logging framework. This framework is designed to facilitate efficient and flexible logging for applications, ensuring that log messages can be captured, stored, and forwarded in various configurations. The document aims to outline the architecture, components, interfaces, and data flows, ensuring clarity and traceability

* 1. **Scope**

The scope includes the detailed design of components, their interactions, interfaces, and the data flow between them. It also covers the allocation of requirements to specific components and ensures traceability throughout the architecture.

* 1. **REFERENCES**
* SOFTWARE REQUIREMENT SPECIFICATIONS (SRS)

1. **Architecture Design**
   1. **Component Diagram**

This session covers the High-level view of component diagram for the logging framework, which consists of three main components:

A screenshot of a computer

AI-generated content may be incorrect.

*fig-1: Logging framework component diagram*

* 1. **Components and Their Roles:**

1. **Logger Library**:
   * Role: Receives log requests from the client application and processes them based on the configuration specified in the YAML file.
   * Interfaces:
     1. l**ogToconsole:** Logs directly to the console.
     2. l**ogTofile:** Logs directly to a file.
     3. l**ogToNetwork:** Sends logs to the Log Server component for network logging.
2. **Logging Service**:
   * Role: Receives logs from the Logger Library in Client mode and processes them based on the configuration specified in the YAML file.
   * Interfaces:
     1. l ogToconsole: Logs directly to the console.
     2. l**ogTofile:** Logs directly to a file.
     3. l**ogToNetwork:** Sends logs to the Log Server component for network logging.
3. **Log Server**:
   * Role: Manages network logs and stores them in the network file.
   * Interfaces:
     1. logToNetwork**:** Receives logs from the Logger Library or LoggingService and sends them to the network.
     2. logToNetwork**:** Stores logs received from the network.
   1. Component Interactions:
4. **Logger Library:**

* In Standalone mode, it directly logs to the console, file, or network using the respective interfaces.
* In Client mode, it sends logs to the LoggingService via the sendlogtoservice interface.

1. **LoggingService:**

* Active only in Client mode, it processes logs received from the Logger Library and logs them to the console, file, or network using the respective interfaces.

1. **Log Server:**

* Receives network logs from either the Logger Library (Standalone mode) or LoggingService (Client mode) and stores them in the NetworkFile.
  1. Sequence Diagram

A screenshot of a computer screen

AI-generated content may be incorrect.

*fig-2: Logging framework sequence diagram*

The Logging Library receives a request from the client application. The YAML file configuration is loaded into the Logging Library at the same time.

1. **Standalone logger Mode**:
   1. In YAML files, the configured mode is Standalone.
   2. Verify the configuration to see where we need to log(console/file/network)

* The logger library uses the logToconsole interface to log straight to the console if the configuration is console.
* If the configuration is a file, the logger library uses the logTofile interface to log directly to the file.
* If the configuration is network, the logger library uses the logToNetwork interface to deliver the log to the Log Server component. After that, send the log message to the network file.

1. **Client Mode:**
   1. In YAML files, the configured mode is Client.
   2. Logger Library send the log to LoggingService component via sendlogtoservice interface.
   3. Verify the configuration to see where we need to log(console/file/network)

* The LoggingService uses the logToconsole interface to log straight to the console if the configuration is console.
* If the configuration is a file, the LoggingService uses the logTofile interface to log directly to the file.
* If the configuration is network, the LoggingService uses the logToNetwork interface to deliver the log to the Log Server component. After that, send the log message to the network file.
  1. Deployment Diagram

A diagram of a login system

AI-generated content may be incorrect.

*fig-3: Logging framework deployment diagram*

1. **Application:**
   * **Elements:**
     + **Logging Library:** Receives log messages from the application.
     + **App Executable:** The main application that generates log messages.
   * **Interaction:** The application writes log messages to the Logging Library.
2. **Log Service Host:**
   * **Elements:**
     + **Logging Service:** Processes log messages received from the Logging Library.
     + **Load/Reload Config:** Manages configuration settings for the Logging Service.
     + **Logging Service Config:** Contains configuration details for the Logging Service.
   * **Interaction:** The Logging Library in the application sends log messages to the Logging Service on the Log Service Host. The Log Service Host writes logs to the console and file.
3. **Remote Server Host** 
   * **Elements:**
     + **Remote Log Server:** Handles log messages forwarded from the Logging Service.
     + **Load/Reload Config:** Manages configuration settings for the Remote Log Server.
     + **Remote Log Server Config:** Contains configuration details for the Remote Log Server.
   * **Interaction:** If configured, logs are forwarded from the Logging Service on the Log Service Host to the Remote Log Server on the Remote Server Host. The Remote Server Host writes logs to a file.

**Flow of Log Messages:**

* **Application:** Generates log messages and sends them to the Logging Library.
* **Logging Library:** Processes log messages based on the configuration and sends them to the Logging Service.
* **Logging Service:** Logs messages to the console, file and optionally forwards them to the Remote Log Server.
* **Remote Log Server:** Receives forwarded log messages and stores them in a file.

1. **Traceability**

TBD.