**Logging Guidelines**

The following are several guidelines related to proper logging practice for software engineering. These guidelines are not specific to a certain framework and should be applicable across all projects.

**Sensitive Information**

Logs should never contain any sensitive information. It is strongly recommended to exclude such information from logs unless the presence of such information is needed for debugging purposes. Even so, sensitive information must be masked when it is being logged. The following is a non-exhaustive list of information that is quantified as sensitive:

1. Personal Identifiable Information (PII)
   1. Refer to [Wiki](https://en.wikipedia.org/wiki/Personally_identifiable_information) for the list of PII
2. Financial information
   1. Account data
   2. Payment data
3. Authentication
   1. Password

**Log Details**

Logs should only contain the necessary information, which can vary according to the environment in which the application is running. In other words, there should be different logging levels catered for each environment. Environments such as **dev** and **staging**can have a logging level that shows more details as compared to a **prod** environment. A general rule of thumb is that the logs on **prod** environment should enable troubleshooter to easily identifier area(s) within the application where an issue might reside while logs on **dev** and **staging** environments should contain enough details to enable a developer to debug the application.

**Log Flooding**

Even in the event where detailed log entries are needed for various purposes, the amount of information should be controlled as to avoid log flooding. Developers must avoid logging large object or long query that will flood the log files, which in turn dampens the application monitoring and/or debugging effort. Only the minimal information that is needed for monitoring and/or debugging should be included in the log file.

**Log File Rotation**

All applications should implement the functionality of modern logging framework to automatically rotate application log files based on predetermined time and/or size. Each log file should be treated as a database, as such, should be backed up and retained for a certain period of time. The following are quick guidelines on log file rotation:

* Time-based rotation
* There should be at least **one**log file for each day
* Size-based rotation
* Recommended log file size should be **below 10MB** to ease the debugging effort
* Retention
* Historical log files should be retained for **at least thirty days**
  + Incident occurrence to incident discovery typically takes more than the recommended thirty days
* Files that are older than the retention period should be automatically removed: either by deleting it from the server or relocated to an archive somewhere else
* Compression
* Since numerous historical log files will be retained over a period of time for audit purposes, it is imperative to compress the historical log files in order to keep the host from running out of storage space