

# Application Error Handling & Reporting

---

## Error Reporting

To report errors on console or into remote logger service (Papertrail), use logger module.

### LOGGER SETTINGS

Define following environment variables to use logger module

Variable Name	Description
PAPERTRAIL_HOST	(Required) Hostname of papertrailapp service
PAPERTRAIL_PORT	(Required) Port number used for papertrailapp service
PAPERTRAIL_LOGLEVEL	(Optional) Level for log capturing, default set to error (debug, info additional values)
PAPERTRAIL_LOGGING	(Optional) Whether to log messages to papertail service or not. Default set to true
CONSOLE_LOGGING	(Optional) Whether to log messages on console or not. Default set to false
APP_NAME	(Required) Name of the application to use when reporting in papertrailapp service

### USING LOGGER MODULE

Include logger module in your code

```
var logger = require('../common/lib/app-logger');
```

Once the logger is included, you can make a call to following logger functions: `logger.log()` - You need to pass first parameter as type of log message (info, error, debug) `logger.info()` - To report message of the info `logger.error()` - To report message of type 'error' `logger.debug()` - To report message of type 'debug'

## EXAMPLES

```
// info: test message my string {}
```

```
logger.log('info', 'test message %s', 'my string');
```

```
// info: test message my 123 {}
```

```
logger.log('info', 'test message %d', 123);
```

```
// info: test message first second {number: 123}
```

```
logger.info('test message %s, %s', 'first', 'second');
```

```
// Error message
```

```
logger.error('Something went wrong')
```

## Proper Error Handling

Please follow below guidelines for error handling and reporting in application code:

- For every possible error, do check if the error was generated/returned. If it was generated, use logger to report the error.
- Log the error message using `logger.error()` call.
- If there is additional data that might be needed to troubleshoot the error further, use `logger.debug()` call to log additional details include complete object.

