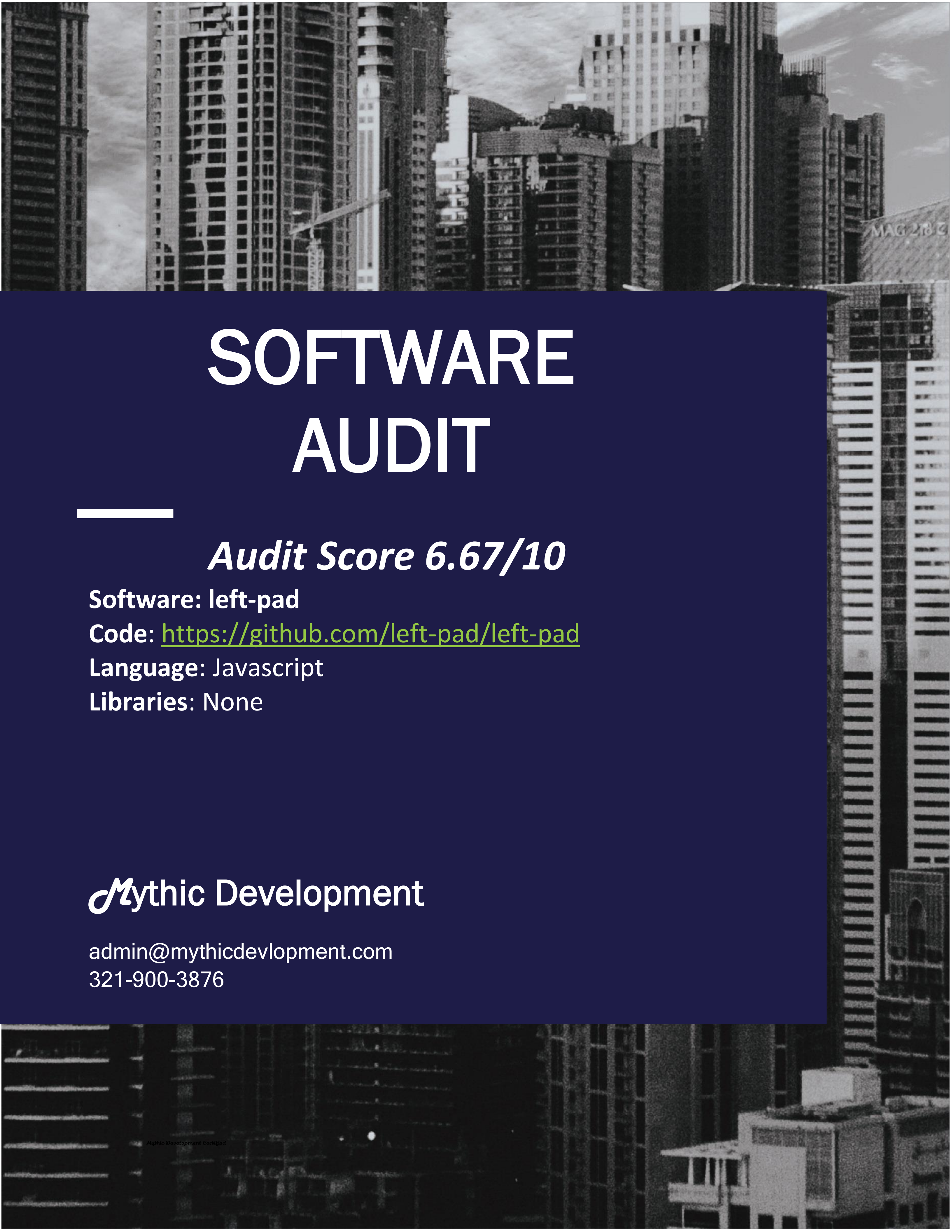

SOFTWARE AUDIT

Audit Score 6.67/10

Software: left-pad
Code: https://github.com/left-pad/left-pad
Language: Javascript
Libraries: None

Mythic Development

admin@mythicdevlopment.com
321-900-3876

Mythic Development Certified

What is a Software Audit?

Mythic Development has conducted an audit of this software. The key factors of such an audit are: licensing, security, and functionality. On the following pages, you'll see a complete assessment of these key fields, including details about how Mythic Development came to the conclusions it came to about the current state of the software. Feel free to contact us at admin@mythicdevelopment.com should you have any further questions.

The licensing audit ensures that all required licenses are acquired and managed effectively for the foreseeable future. In software, there are 2 common forms that a license can take: tooling, and libraries. Tool licenses are rare, but occasionally specific tools must be licensed to provide functionality that the code itself does not include. Such tooling might be a piece of software running on a server for document processing, or a cloud-based API connection. Library licenses are everywhere, and often a giant problem to contend with. Mythic Development will sift through layer after layer of libraries to ensure any open source licenses are appropriate for the software's use-case, and any more restrictive licenses have been obtained properly.

The security audit notifies the organization of any potential vulnerabilities in their code, libraries, and tools. Mythic Development searches through public vulnerability databases to find any at-risk code. As a part of this research, Mythic Development also assesses the risk of the specific vulnerabilities found based on the way the code functions to identify if the vulnerable code is actually a risk to this specific software.

The functionality audit takes 2 forms. First, Mythic Development considers the code quality itself to identify the ease of maintenance and management in the future. Second, Mythic Development identifies the code's use-case and attempts to ensure it meets the needs of the client. In some cases, a client will also contract Mythic Development to develop a series of software tests for the code, which can increase the effectiveness of this portion of the audit. If not, then Mythic Development will simply ensure the code compiles and can be deployed and run any existing automated tests against the code-base and present the results.

Licensing – 9.5/10

Language 9/10:

- License URL: https://github.com/nodejs/node/blob/master/LICENSE
- MIT for nodejs
    - Though the language here is javascript, the actual use case is nodejs, as this library is designed to be imported via npm
    - MIT license technically requires reproduced code to obtain the same license, making it slightly less desirable than most other licenses when included for a dependency

Left-Pad 10/10:

- License URL: https://github.com/left-pad/left-pad/blob/master/LICENSE
- MIT
    - As long as the author has no intent to sell licenses, MIT is a fine choice for the codebase
- Dependencies: None
    - This project only has dev dependencies for testing and benchmarking, not used in the code

Security – 4/10

Language 4/10:

- **Code:** https://github.com/nodejs/node
- NodeJS is not exactly known for security, but is commonly used in production, meaning security issues are identified quickly and repaired readily. The nodejs project has a regular release cycle, ensuring issues are fixed at a regular pace.
- CVEs: https://www.cvedetails.com/vulnerability-list.php?vendor_id=12113&product_id=&version_id=&page=1&hasexp=0&opdos=0&opec=0&opov=0&opcsrf=0&opgpriv=0&opsqli=0&opxss=0&opdirt=0&opmemc=0&ophttprs=0&opbyp=0&opfileinc=0&opginf=0&cvssscoremin=0&cvssscoremax=0&year=0&month=0&cweid=0&order=1&trc=72&sha=94bc04a83d9407dd349a4f1d7130e3451335037d
    - NodeJS has a large number of vulnerabilities, some as recent as August of 2021. Since this project is not version locked to a particular version of nodejs, nearly every vulnerability here can be applied to this codebase. This is a high risk scenario. There be dragons!

Functionality – 6.5/10

General 2/10:

- This library is deprecated, and it is recommended that no one use it anymore. This alone is reason to suspect the code.
- The code is simple on the scale of most projects, but for what it does, the code is incredibly complex. As mentioned in the deprecation message, the accepted way to do this is: `String.prototype.padStart()`
- While the code does work, it has bits written into it to deliberately fool and enhance performance tests, like a cache specific to using spaces.

Tests 9/10:

- The project includes a decent set of test cases, as well as a decent set of performance specific test cases. This is great!
- Unfortunately, even with the tests included, the developer has not achieved 100% code coverage.
- Additionally, the ci-cd pipeline build has been disabled, probably because the library is deprecated

Style 5/10:

- Nearly every line of code is commented. As a general rule, Mythic Development recommends commenting a line of code only if what it is doing is not immediately obvious. In this codebase, comments actually detract from the ability to read the code
  - e.g. one line of comments reads `// loop`
- The choice to use a `while (true)` loop while breaking is a big no-no in the software world. This code is not meant to run infinitely, so the while-true-loop makes the code harder to read, and gives the wrong impression to the developers.

Reliability 10/10:

- Frankly, this code has been used in production by dozens of large libraries for years. Until the package was originally pulled from npm, causing those libraries to crash builds, it worked reliably. There isn't really even a need to test to verify this. The only unreliable thing is that this code is in a library at all.

Disclaimer

This audit analyzed the software, vulnerabilities, dependencies and licenses at the time of the audit. After the audit, all of these things may change, and Mythic Development makes no assurance that they won't. Should an error be found in this audit, Mythic Development warranties the audit for it's entire value and will be obligated to return the full price of the audit to the customer, but no further costs will be obligated to Mythic Development.

This audit is not an endorsement of the software in question, nor representative of the organization behind the software. This audit is not to be used for investment or financial decisions. It is for educational purposes only. This is not a guarantee of security, or a guarantee of functionality. It is an investigative report.

Mythic Development Certified