PONDER 02:

QUALITY MODEL

Ryan Dockstader

# Model Application

The model that will be presented in this paper will be directly applicable to modern web applications. To further define a modern web application, we need to look at it for it’s individual pieces. The backend of a modern web app should implement a fully functional REST API. The front end, or User Interface, should in most cases be a Single Page Web application built on a modern framework such as React, Angular, or Blazer.

# The Model

This model consists of 4 key factors. They are:

* Commitment to a Back-End Standard
* Security
* Automated Testability
* Portability between devices

# Commitment to a Back-End Standard

To promote extensibility between desktop, browser, and mobile applications a modern application should implement an API (application programming interface) as a backend. Most modern applications implement either a SOAP API, or a REST API. Depending on the needs of the application, one may be better than the other. However, no matter which is chosen, best practices should be followed. For example, when Implementing a RESTful API, the endpoints should be stateless, cacheable, and uniform.

# Security

One of the largest threats modern web applications face today is a lack of security. To build a high-quality application, security must be in the forefront of the mind of each developer working on the project. Unfortunately, it’s not a matter of if, it’s a matter of when and how bad. Because of this, without keeping security in mind applications can require a complete rewrite, if they survive at all. Without security, any application will most certainly fail.

# Automated Testability

A hot topic across modern web development is code coverage. It is extremely important to cover critical functions and critical paths with automated testing. Since modern applications can become very large very quickly, thanks to modern framework and code generators it is nearly impossible for modern Quality Assurance teams to be able to test every aspect of a new feature, let alone all the existing features. Therefore, code coverage is important. Without automated testing coupled with real hands on tests you cannot have any type of quality assurance

# Portability Between Devices

In the modern era, portability between devices is not only in high demand, but is a requirement of quality web applications. Any user that is going to use an application on a computer is now going to try to use that same application on their phone. No matter if it’s a banking website, a dating website, or even a forum. If you are looking for quality, you’re looking for portability as well.

# Citations

1. “CHAPTER 5.” Fielding Dissertation: CHAPTER 5: Representational State Transfer (REST), <https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm>.
2. “SOAP vs. REST Comparison: Differences in Performance, APIs & More.” Stackify, 21 May 2019, <https://stackify.com/soap-vs-rest/>.
3. Pegschools. “3 Reasons Why Computer Security Is Important.” Branford Hall Career Institute, 28 Aug. 2019, <https://www.branfordhall.edu/3-reasons-why-computer-security-is-important/>.
4. “Clover Support.” About Code Coverage - Atlassian Documentation, <https://confluence.atlassian.com/clover/about-code-coverage-71599496.html>.
5. Why Is Responsive Design So Important?, <https://www.webfx.com/web-design/why-responsive-design-important.html>.