Title: Designing with Progressive Enhancement: Building the web that works for everyone
Authors: Todd Parker, Patty Toland, Scott Jehl, and Maggie Costello Wachs
Published Date: 2010

The purpose of this review is to give readers an overview of the book entitled “Designing with Progressive Enhancement.”

First, this book is not designed for novices. The user should be comfortable with the basic fundamental building boxes of HTML, CSS and JavaScript to get the most out of this book. It is a relative new book (2010) with discussions of many leading edge technologies like CSS3 and HTML5.

While there are many new technologies that fall under the banner of Web 2.0 (i.e., AJAX, Flex, advanced CSS), many users do not update their browser to embrace these technologies for a variety of reasons:

- They don’t know how to upgrade or is not allow to upgrade in a corporate environment
- They don’t upgrade for security reasons in a corporate environment
- They are tied down to a legacy browser because of specific application.

Moreover, some users may have their JavaScript and CSS turned off.

The goal of the authors is to allow you to create content that is accessible to everyone, regardless of their language, physical abilities, technology platform, etc. using a concept called progressive enhancement.

With progressive enhancement, the developer can create a page that is accessible from the start and guaranteed to work on any web-enabled device (i.e. browsers, phones, gaming systems, etc.) Afterward, the developer can create additional “layers” to progressive enhance the page to work on a more rich interactive page by then adding CSS and JavaScript to external files. Inline styles and event handles are avoid wherever possible as well. The goal is the separate the HTML “presentation” layer from the CSS “style” layer.

The authors uses a concept called “x-ray perspective” in which they analyze a complex interface design, map out the semantic HTML that will support a “basic functioning” experience and the plan for developing advanced CSS and JavaScript that will create the “enhance” experience. To create this enhance experience the authors suggest that the best approach is to create a series of browser capabilities tests that will check for reasonable cross-section of browser capabilities and then only deliver enhancements when both CSS and JavaScript are enable on the user computer.

The authors also point out the different between progressive enhancement and graceful degradation. Graceful degradation (also known as fault tolerance) works off of the principle that some parts of a complex system may fail and will then seeks to find an alternate path that “degrade” to a lesser but still functional experience. For example, the noscript tag is a way to deliver alternate content only to users without JavaScript support.

Progressive enhancement on the other hand takes a different approach. It assumes that all web-based system can be first reduced to a simple, functional experience that can work anywhere. Once the need of many has been met, the more robust enhancements that modern browsers can handle are layered on progressively to build up to a more complex system for an “enhanced” experience.

The book concludes with how to build widgets (i.e., collapsible content, tabs, tooltips, etc.) using progressive enhancements.

Sincerely,
Cornelius Chopin