

Step 14: Sample Style Guide

Style Guide for the *Design & Usability Dashboard* Application

1. Introduction

This style guide covers the *Design & Usability Dashboard* (DUD) web application. DUD is being developed by Carol Mackey and David McKillen during the spring semester of 2008 for the Virginia Commonwealth University INFO 654 class. DUD provides a conclusive, concise "How-to" guide for website design and usability engineering and testing.

1.1.Purpose

The purpose of this style guide is to provide developers with the details they need to design and maintain DUD, a user friendly web application, which adheres to the best practices of web design and usability.

Well-establish techniques are available which can be used to build usability into web applications. DUD will define these techniques and provide users with examples of these techniques.

1.2.Scope

Future applications that want to take advantage of industry best practices of web design and usability may use this style guide as a model.

1.3.Audience

This style guide is a specification to be used by web application designers, web usability testers and web developers as they develop the detailed user interface for the web application DUD.

1.4.Application of Standards

This style guide should be used during development of the detailed user interface, it also be consulted thorough out the entire development cycle, to make sure all standards are followed.

Style Guide for Design & Usability Dashboard

1.5. How to Use This Document

This document should be used throughout the DUD lifecycle. It should be used as a guide to make sure DUD provides the best practices for web site usability and design.

2. Overview of Functionality

DUD contains three main pieces of functionality:

2.1. Overview

A concise overview of the steps of web design and usability evaluation and testing:

- Step 1: General Review of the Style Guide
- Step 2: User Profiling
- Step 3: Contextual Task Analysis
- Step 4: Setting the Usability goals
- Step 5: Platform capabilities and constraints
- Step 6: General Design principles
- Step 7: Interface Reengineering
- Step 8: Conceptual Model Design
- Step 9: Conceptual Model Mock ups
- Step 10: Iterative Conceptual Model Evaluation
- Step 11: Screen design standards
- Step 12: Prototyping screen design standards
- Step 13: Iterative evaluation of screen design standards
- Step 14: Detailed GUI design
- Step 15: Iterative evaluation of the GUI design
- Step 16: Getting user feedback

2.2. How-to Steps

Each step has a "how-to" summary description.

2.3. Examples

Each step has an example file; which demonstrates how to apply the concepts explained in the Overview and How-to sections.

3. User Profiles

3.1. Summary of Methodology

Style Guide for Design & Usability Dashboard

User profiles were obtained by sending questionnaires to web designers and web usability testers who currently work at large companies in the Richmond area. Information was also gleaned from Carol's and David's work experience, as web developers.

3.2. User Profile Summaries with Implications

Web designers and usability testers are trained, salaried employees who design and test web applications developed by large companies. They are highly motivated and skilled in web design and usability testing. Their education level is high and all have a college degree, some have advanced degrees.

Web designers and usability testers are very experienced in their jobs and have a very high level of skill in computer usage and web site usage. This suggests that the users will have a high expectation for the visual appeal and content of the DUD web site.

4. Contextual Task Analysis

4.1. Summary of Methodology

In order to define the tasks that need to be addressed by DUD, Carol and David queried current web designers and web usability testers about their jobs and responsibilities. They also used their experience as professional web developers to define needed tasks.

4.2. Work Environment Analysis with Implications

DUD users have a highly professional work environment. They are surrounded by information technology professionals and web development professionals, who are highly educated and motivated.

Their physical environments range from traditional office cubes, to creative office pods. They work with business experts, business analysts, technical analysts, management, developers, and graphic designers.

4.3. Task Analysis Document with Implications

DUD users experience several different task flows during their involvement in web applications development. The flows are:

- For some projects they get involved at the very beginning of requirements definition and application design. That is the point of a project where usability can make the most impact.
- For other projects, they might be asked to do an analysis or quick testing of an existing system. After their analysis or testing, they make recommendations to the application owner about changes and enhancements that could provide better design and usability.

Style Guide for Design & Usability Dashboard

The implication of these different tasks is that the DUD user may need to provide usability based on mockups and wireframes or they may have a live system to evaluate.

4.4.Task Scenarios with Implications

Below is a description of a task scenario that a DUD user may encounter.

Task: Evaluate existing web site for usability and design.

Task Flow:

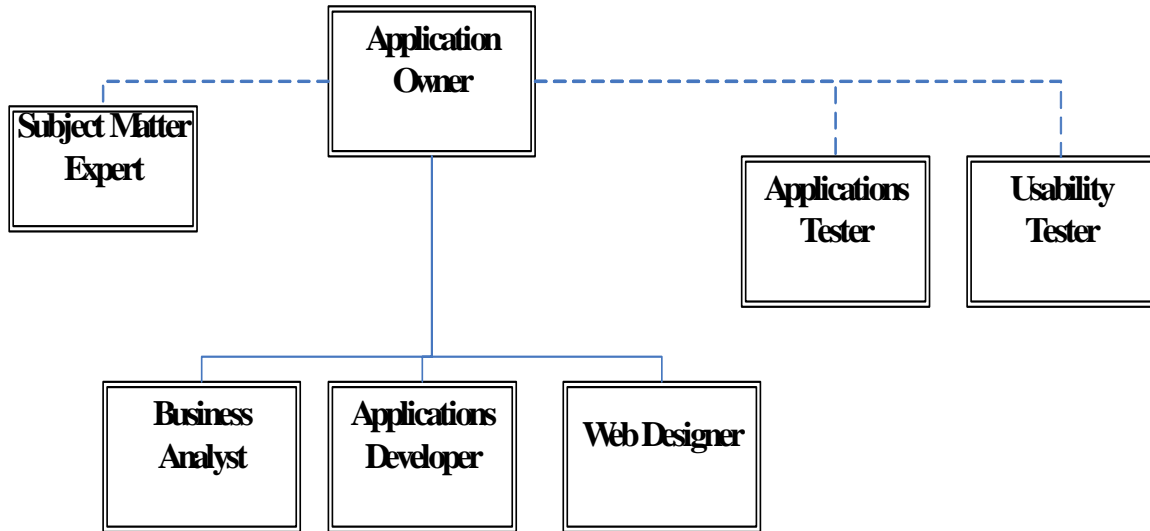
1. An application owner asks the DUD user to evaluate an existing web site.
2. The DUD user evaluates the site and writes an evaluation of the site, which might contain change requests.
3. The application owner may approve change requests as defined in the evaluation.
4. The DUD user enters change requests into the change management system and tracks the progress of the request.

Task Closure: This task can take from one day to multiple months, depending on the size of the web application. The task could be closed after the evaluation is delivered to the application owner if no change requests are identified or approved. The task could be closed after the identified change requests are implemented into the web application.

4.5.Current User Task Organization Model with Implications

The DUD user has a dotted line connection to the application owner. They are an available resource for the applications owner, but not a direct report. With this reporting structure, it will be important for the application owner to be able to track the progress of the DUD user.

Style Guide for Design & Usability Dashboard



5. Platform Capabilities and Constraints

DUD is a web application and as such, will need only a thin client footprint.

5.1. Identification of Hardware Platforms

It is expected that the DUD user will have a computer with a minimal amount of memory and available hard disk space. DUD is not limited to a single platform and can be used from a Windows system or any other desktop or laptop based system.

5.2. Identification of Software Platforms and Tools

The only software that the DUD user needs is a current internet browser, such as Internet Explorer or Firefox.

5.3. Summary of Platform Capabilities and Constraints with Implications

As a web based application, DUD needs only a minimum of hardware and software.

6. Usability Goals

6.1. Qualitative Goals

The qualitative goals that DUD will provide are:

- It will integrate within the software development lifecycle with a minimal disruption to the timeline. DUD users will not need training in how to use DUD, because it will be intuitive and self-explanatory.

Style Guide for Design & Usability Dashboard

- DUD users will gain confidence in their design and usability tasks as they follow the established best practices, as provided by DUD.

6.2. Quantitative Goals

The quantitative goals that DUD will provide are:

- Web designers and usability testers will be able to find all the check lists that they need for their tasks within two minutes of entering DUD.
- Web designers and usability testers will be able to find examples within two minutes of entering DUD.