Interactive Lifecycle Document Development with Requirements Tracking

Naval Undersea Warfare Center Division Newport

Client: Mr. Michael Grimley

Team: **Jeremiah Butler**

Peter Magalhaes

Aria Ushani

Kevin Palmer

Abstract

- We were given a task of creating a system to maintain documents throughout the software development process.

- This includes creating new documents as well tracking individual requirements throughout the lifecycle of the project

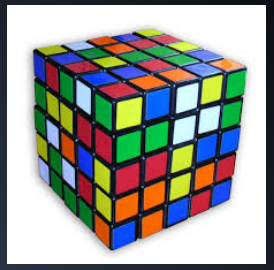
Problem

- Using Microsoft Word to create, maintain, and manage lifecycle documents
- Users manually track individual requirements for each document through various phases
- Systems currently available are overly comprehensive and are typically expensive, complex, and require an underlying database
- Users who may have little computer experience, or even no prior XML experience.

Requirements

- System will allow users to develop and edit lifecycle documents storing information in XML files.
- Users have the ability to view existing requirements, design elements, and test cases at any point in time.
- System is able to track individual requirements throughout each document.
- System shall be able to auto-generate a Requirements Traceability Matrix at any point during development.
- XML instances will conform to an XML Schema designed and developed by the team.
- The XML instances can then be transformed into an HTML-based interactive document, using XSLT, which is also designed and developed by the team,

Challenges



- Learning unfamiliar programming languages
- Functionality to track each requirement through various phases without the use of a database
- Creating a user friendly graphical interface
- Non-technical users must be unaware of the use of XML
- Automatically generating a requirements traceability matrix at any time
- Exporting all documents as interactive HTML files.
- Must work on Windows 7 using Internet Explorer.

Solution

- Create a diverse schema to validate that each XML document is well-formed
- Provide multiple structured XML documents to demonstrate tracking of requirements and information
- Use the XSLT processor to format the information into various formats, which present the information differently
- A custom javascript editor presents the transformed document to the user, so they can read it and make changes to the documentation

Tools

- GitHub

- Google Drive

- Rally Software

- oXygen XML

- Notepad++

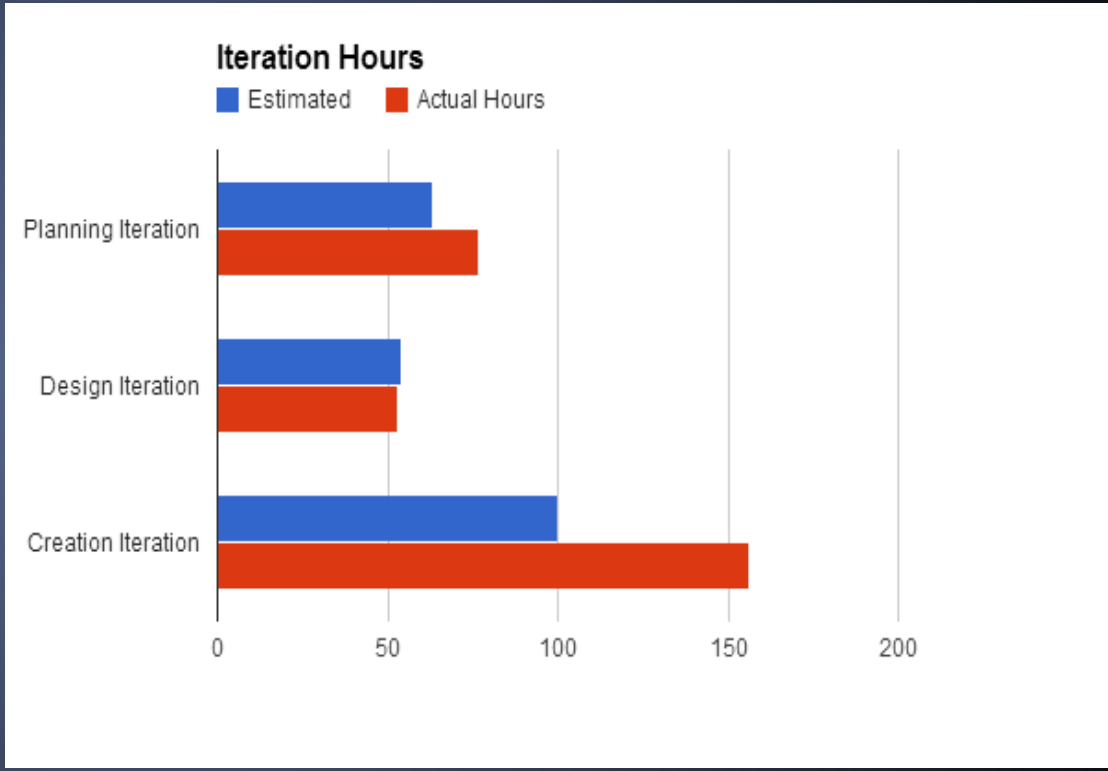
Languages

- HTML
- CSS
- jScript

- XML
- XSD
- XSL

Software Process

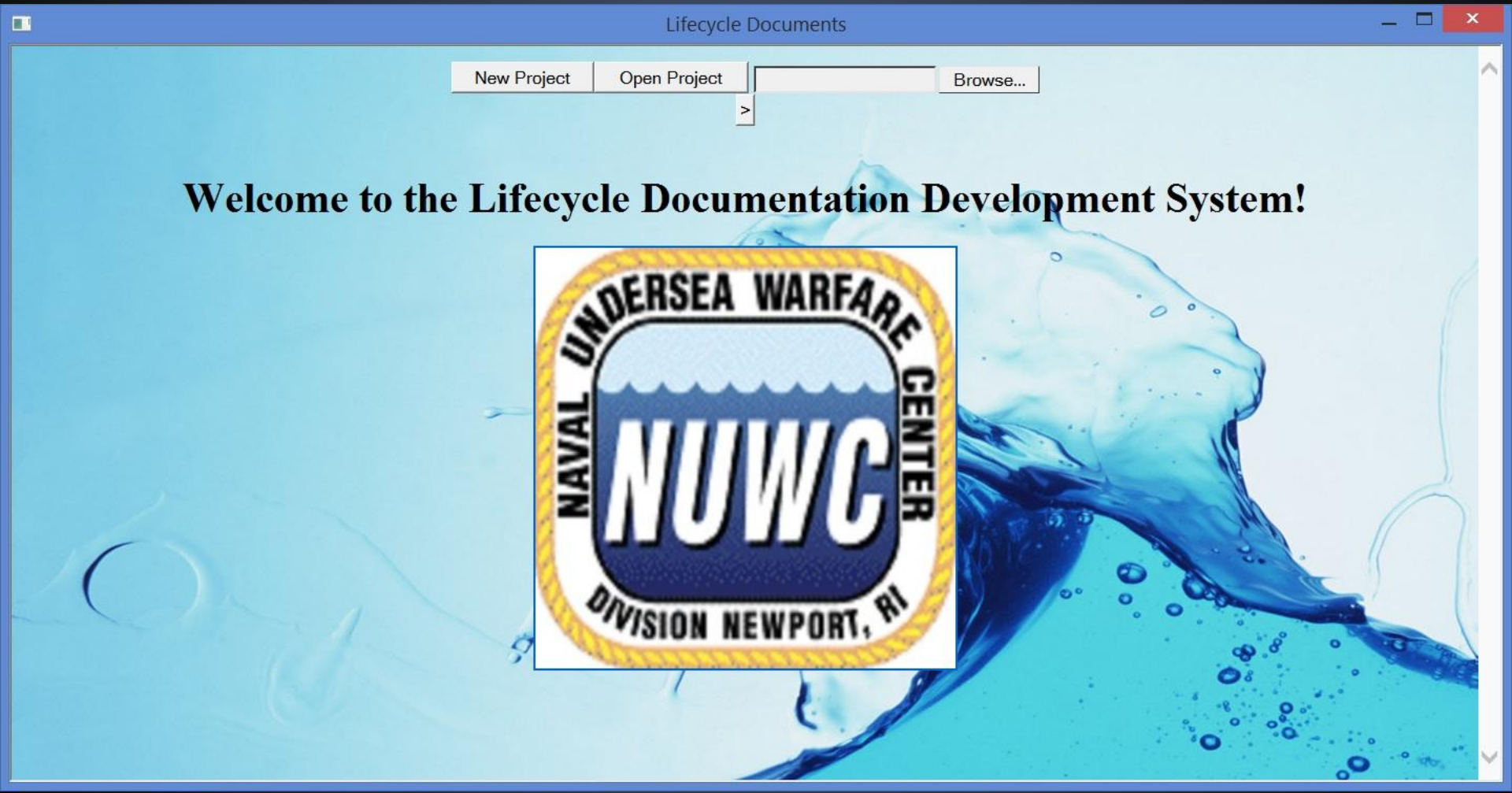
- Planning
  - Sept 13th through Oct 20th
  - Estimated - 63 hours
  - Actual - 76.5 hours
- Design
  - Oct 21st through Nov 27th
  - Estimated - 54 hours
  - Actual - 53 hours
- Creation
  - Nov 28th through April 14th
  - Estimated - 100 hours
  - Actual - 156 hours
- Testing
  - April 14th to present
  - Estimated - 76 hours



Major Changes During Development

- **SCRUM Tools**
  - Started with scrumDo which was a nice scrum tool that allowed for Google incorporation and had an iPhone app. However it lacked some necessary features such as burndown graphs so we switched to Trello, then again to RallyDev.
- **Koding.com**
  - This was our original choice for cloud coding, which is a great idea for users to code simultaneously, unfortunately, Koding was in beta testing phases and caused many problems ranging from compatibility, to unavailability
- **HTML to HTA**
  - Development was started using Koding which was a cloud based dev platform. After quite a bit of work, we realized that our application would operate faster without a web server. This problem was solved by converting our existing HTML code to HTA, since HTA is a windows application that runs HTML code with access to the local file system.
- **JS to XSLT**
  - Originally we used javascript to create a recursive function to obtain all necessary node values. This function required a significant amount of time to complete (~20 seconds). After switching to XSLT, we were able to reduce that time to less than 1 second.

Demonstration



Questions?