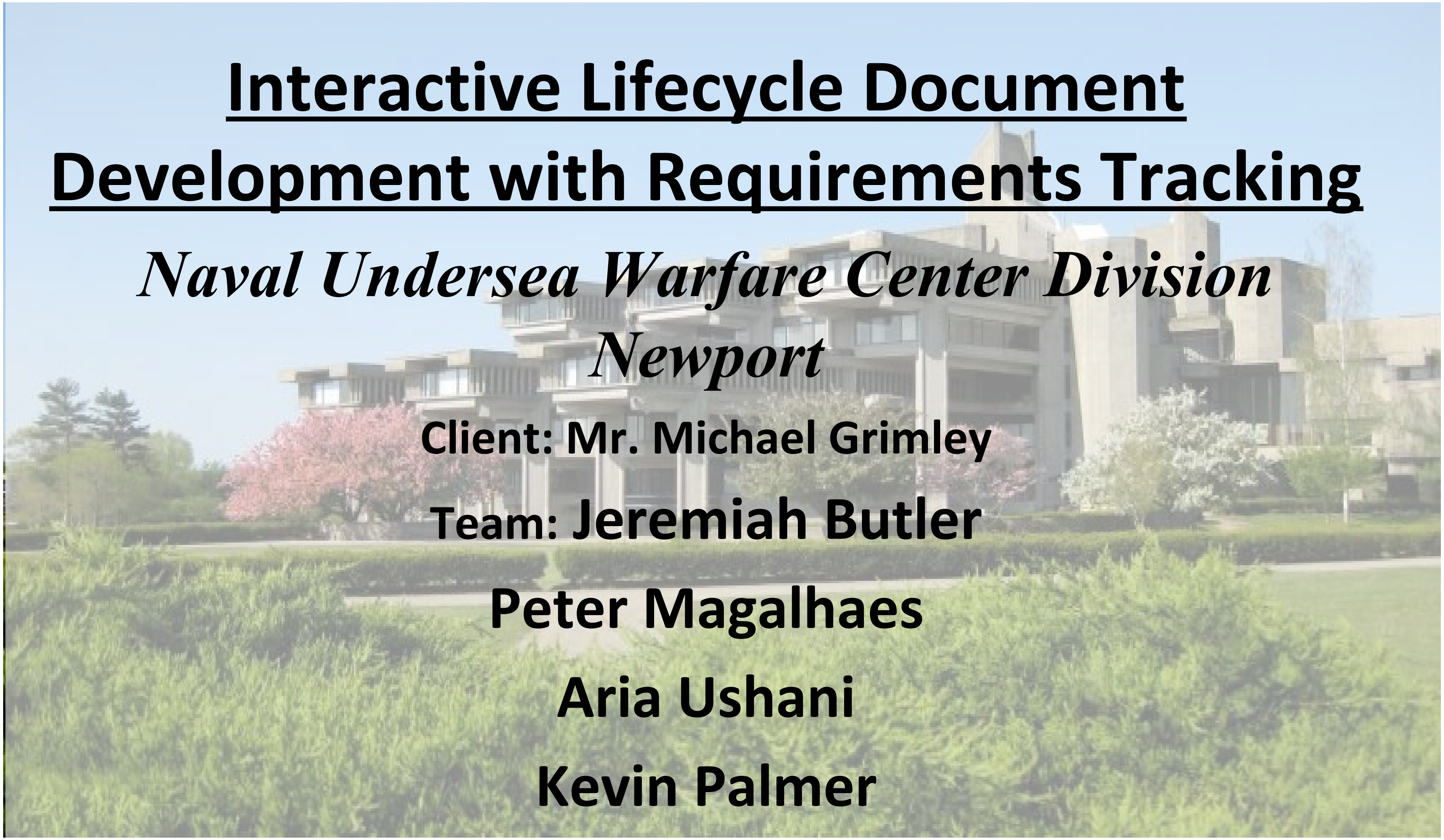
UMass Dartmouth

Computer Science Day 2014



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

Problem

- Currently using Microsoft Word to edit and create documents
- Manually tracking each requirement in a database through each phase of development
- Systems currently available are expensive, complex, and require a database

Challenges

- Learning unfamiliar programming languages
- Mechanism to track requirements through development, design, and testing without the use of a database
- Simple, easy to use graphical interface
- Prompt response from the system for the user
- Auto-generated requirements traceability matrix
- Exporting project information

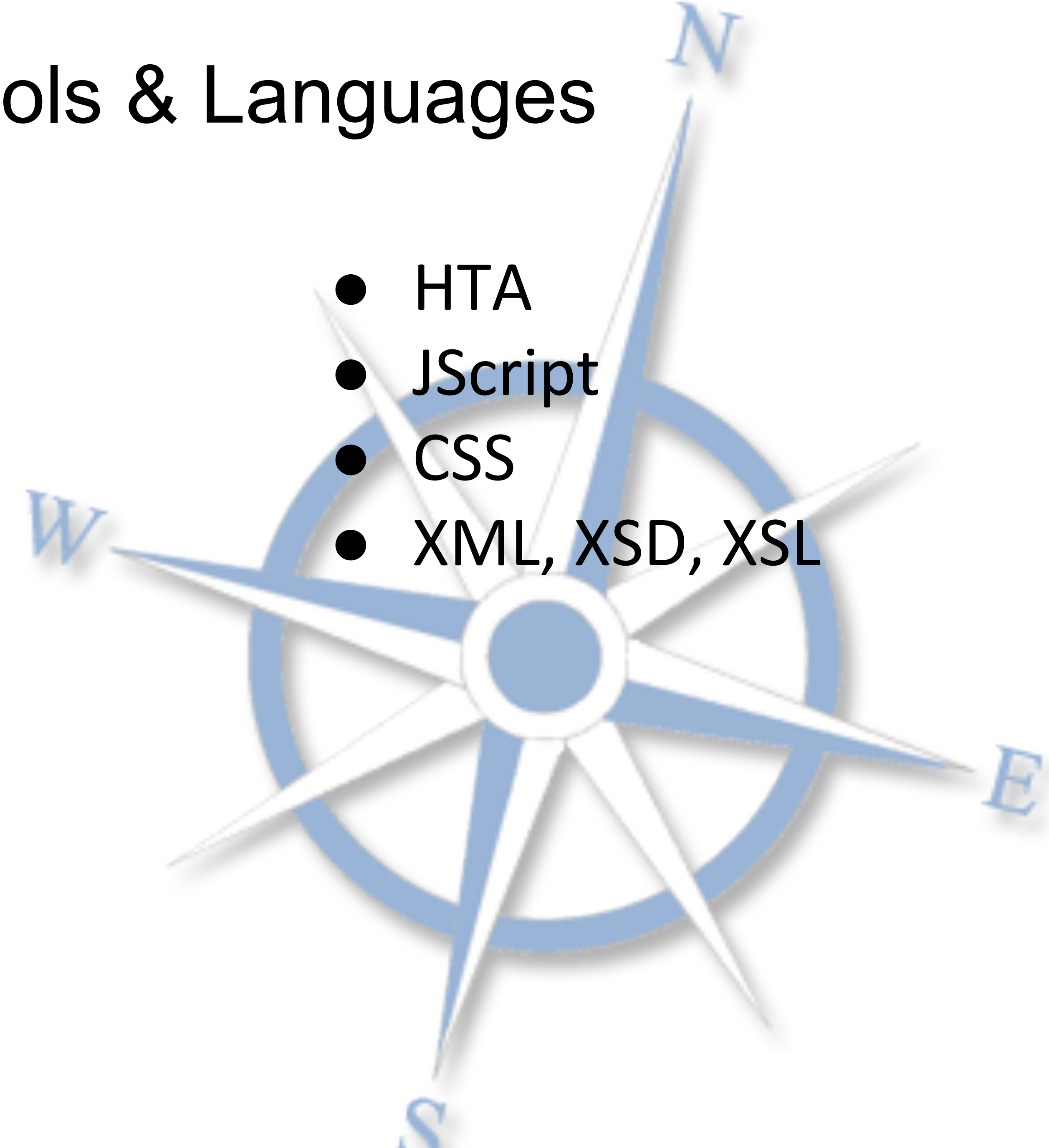
Solution

- Provide a structured XML document to keep track of requirements and information
- A schema will be used to make sure all provided XML's can be read
- XSLT processing is used to present the information in a way that it is readable to anyone
- A custom made javascript editor will present the transformed document to a user, who has no prior XML knowledge, so they can read it and make changes to the documentation

Tools & Languages

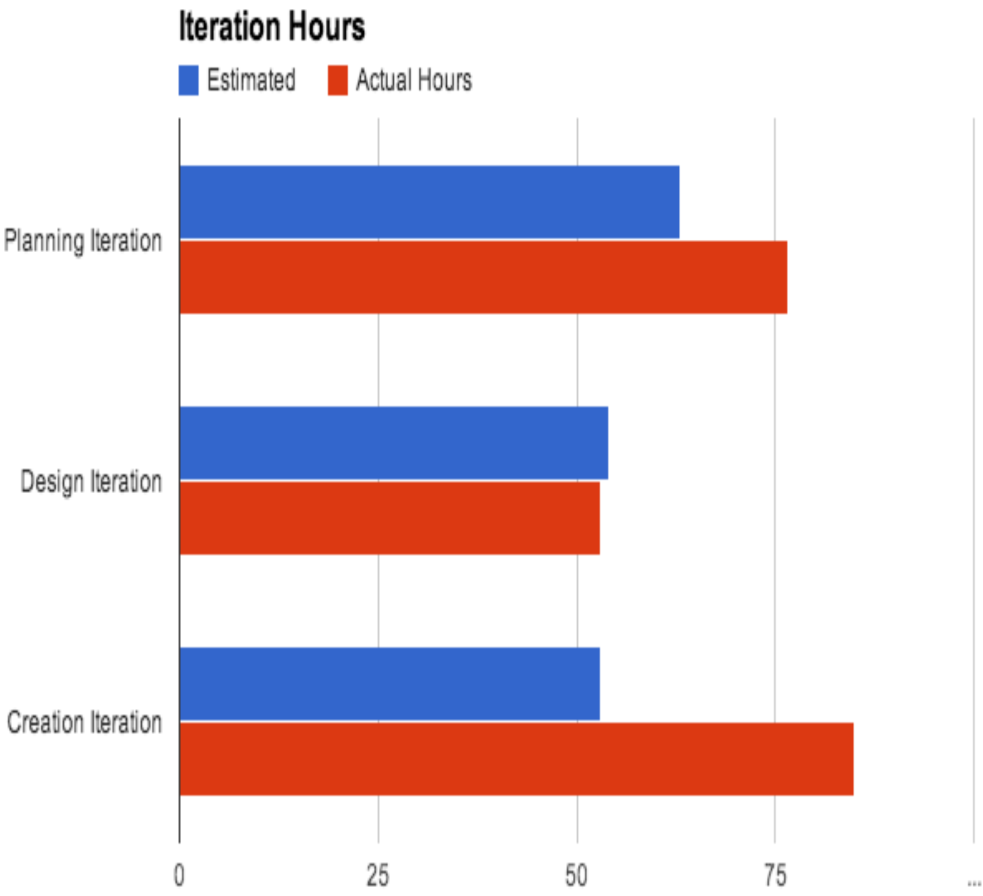
- Github
- Google Drive
- Rally
- oXygen
- Notepad++
- Koding

- HTA
- JScript
- CSS
- XML, XSD, XSL



Software Process

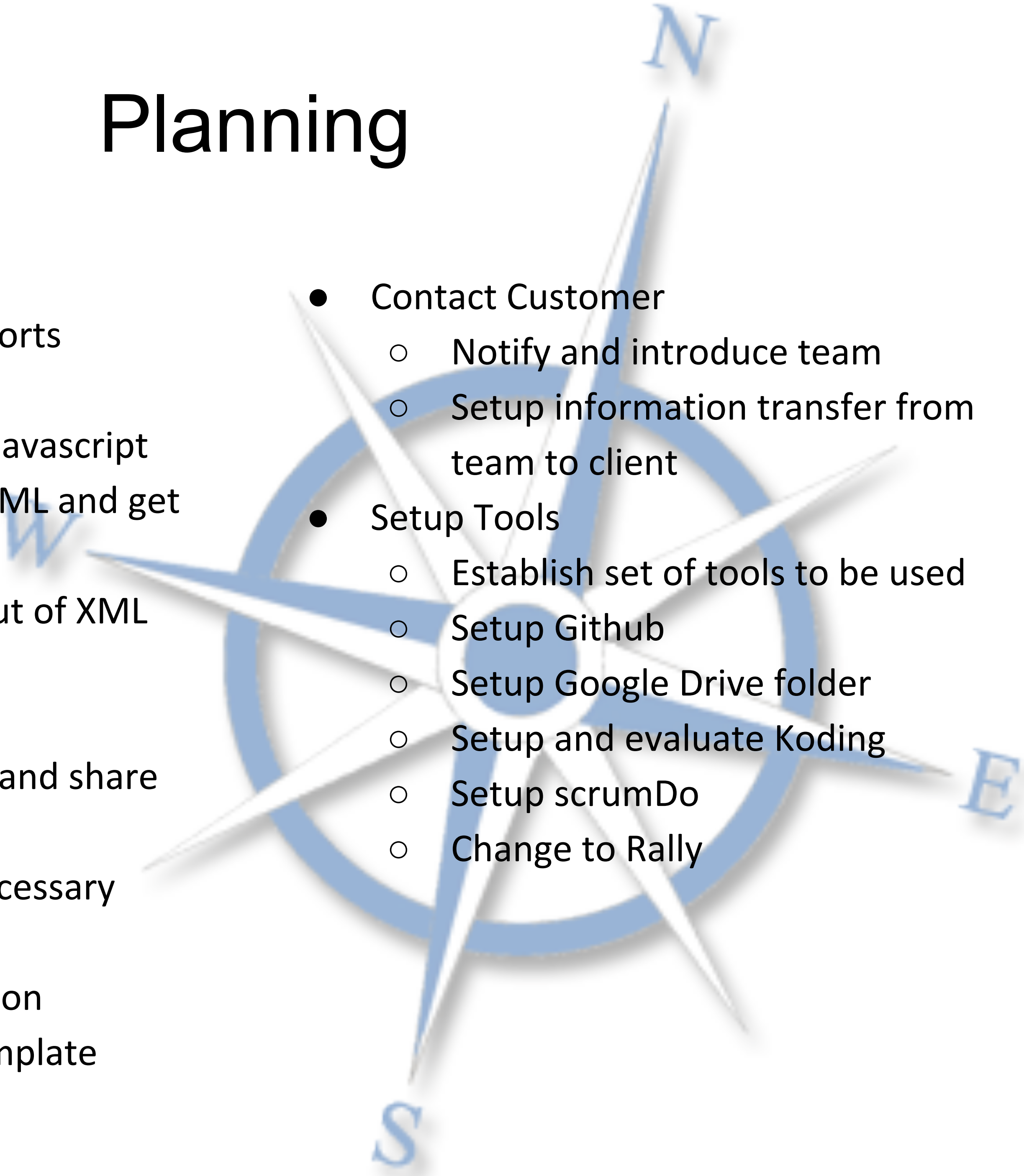
- Planning
  - Sept 13th through Oct 20th
  - Estimated - 63 hours
  - Actual - 76.5 hours
- Design
  - Oct 21st through Nov 27th
  - Estimated - 54 hours
  - Actual - 53 hours
- Creation
  - Nov 28th through March 14th
  - Estimated - 53 hours
  - Actual - 85 hours
- Testing
  - March 15th through May 1st
  - Estimated - 38 hours



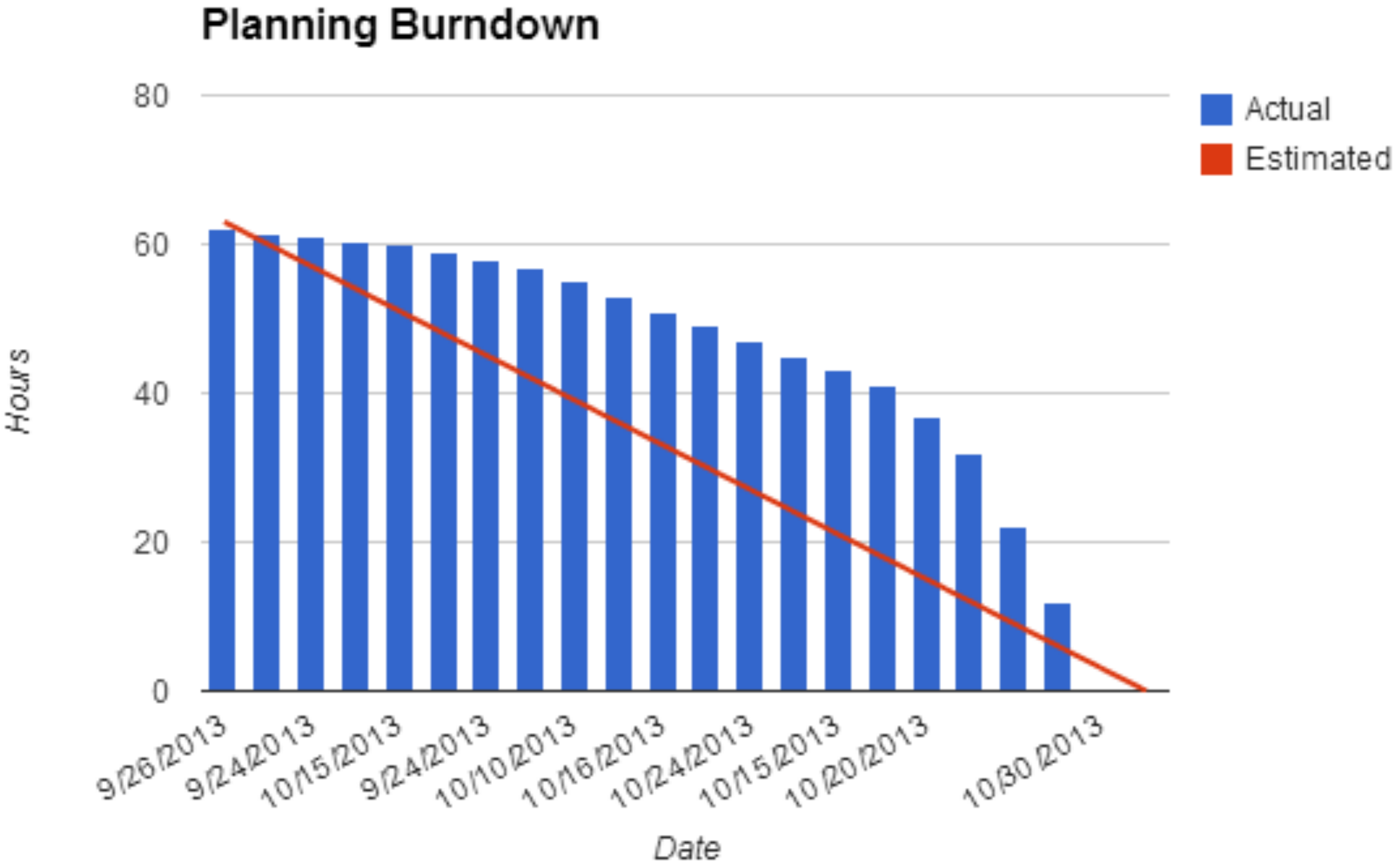
Planning

- Meeting Log
    - Create initial log
    - Update as Necessary
- Weekly Reports
    - Create and share reports
- Research of Languages
    - Research HTML and Javascript
    - Implement JS open XML and get values
    - Style Javascript output of XML for readability
- Documentation
    - Create SRS template and share to members
    - Assign sections to necessary members
    - Fill out assigned section
    - Create Vision doc template
    - Update Vision doc

- Contact Customer
    - Notify and introduce team
    - Setup information transfer from team to client
- Setup Tools
    - Establish set of tools to be used
    - Setup Github
    - Setup Google Drive folder
    - Setup and evaluate Koding
    - Setup scrumDo
    - Change to Rally



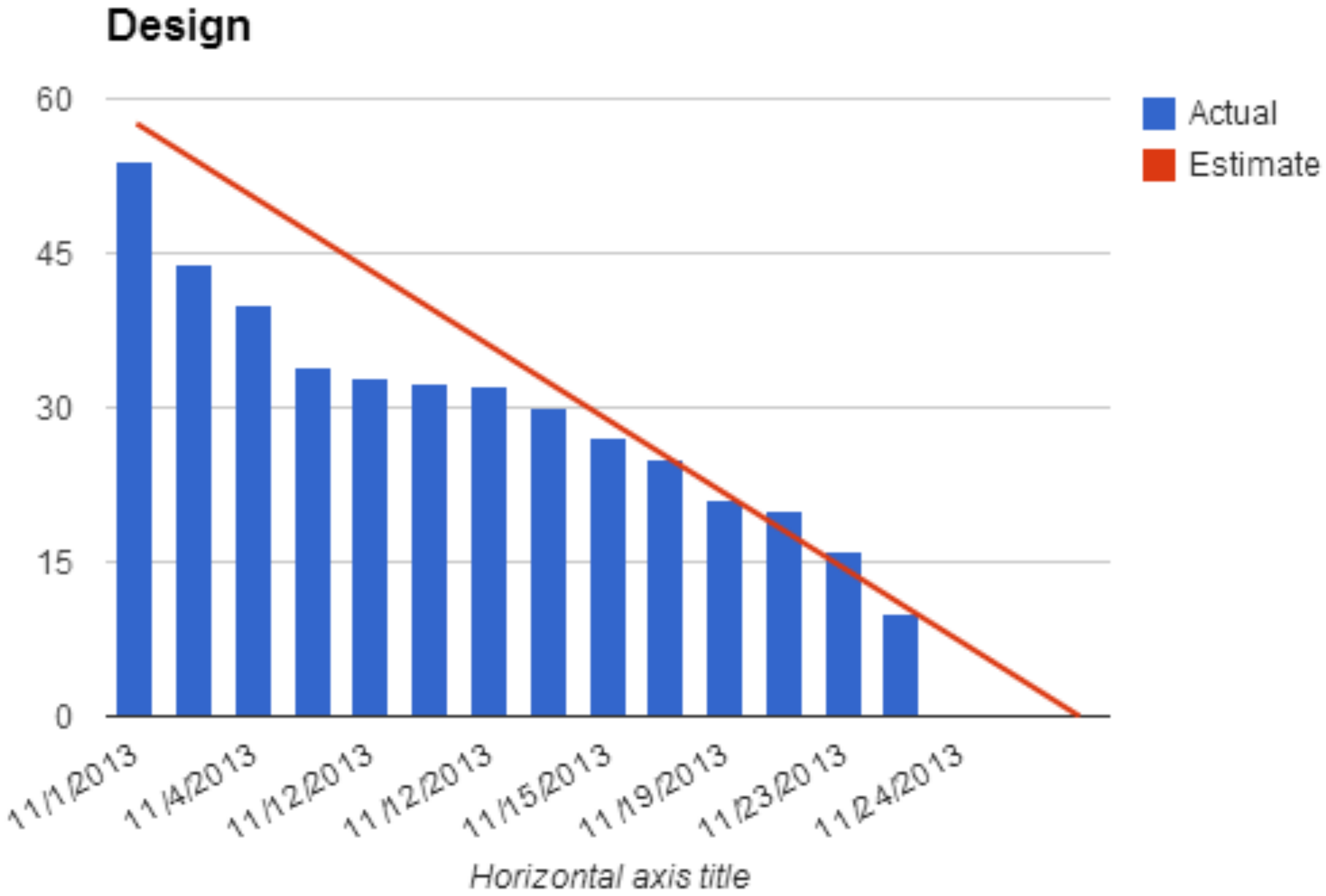
Planning Burndown Chart



Design

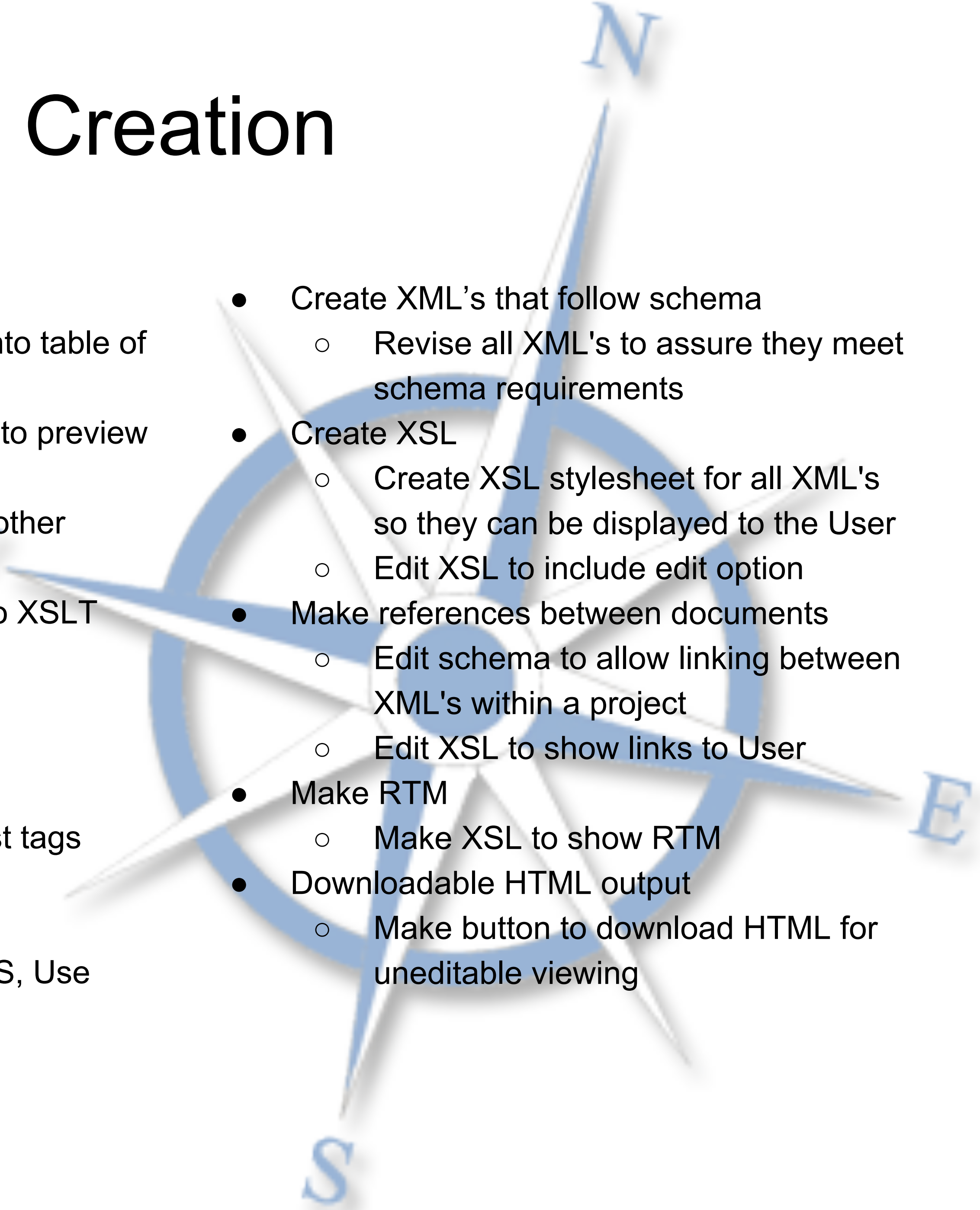
- Influence Meeting
  - Meeting with member of military who has used similar programs
  - Material Organization
  - Create Lists of Questions
  - Document What was learned
- Use Case Document
  - Define UseCases
  - Create Use Case diagrams
- Interface Design
  - Create Interface Templates
- Update Documentation
  - Update SRS to represent changes
  - Software Development Plan
  - Keep vision doc and meeting logs up to date
- Construct Test Docs
  - Create example RTM
  - Research xml and schemas
  - Create test xml's so coding can start

Design Burndown

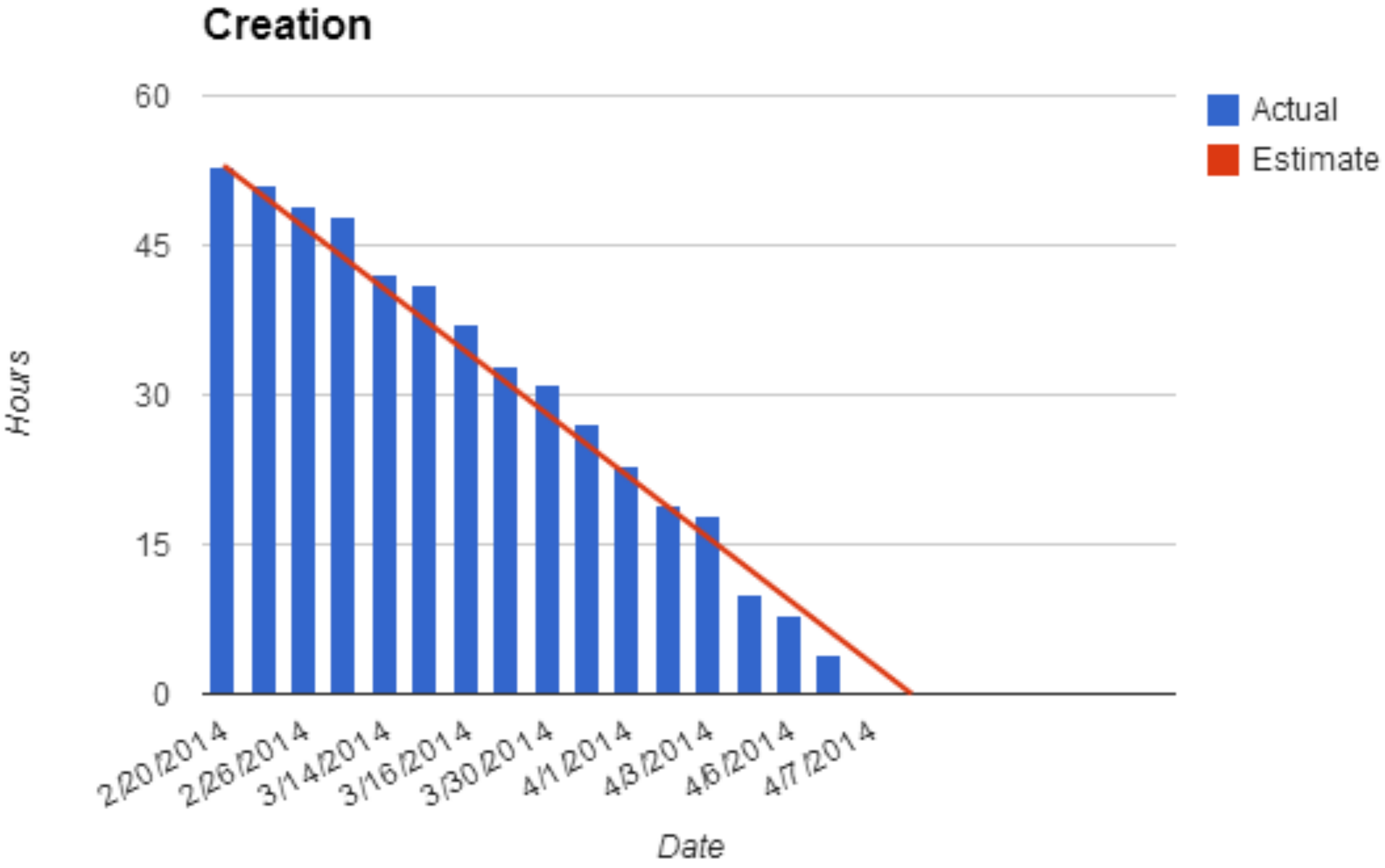


Creation

- Display XML to User
  - Output XML nodeNames into table of contents
  - Output XML nodeValues into preview pane
  - Display tabs for access to other documents in project
  - Change displayed output to XSLT result
- Make XML editable
  - Put preview into textarea
  - Implement save method
  - Implementation of isNewest tags
  - Changes for XSLT
- Create XML Schema
  - Implement Schema for SRS, Use case, and Test case
  - Implement Project schema
  - Review with client
- Create XML's that follow schema
  - Revise all XML's to assure they meet schema requirements
- Create XSL
  - Create XSL stylesheet for all XML's so they can be displayed to the User
  - Edit XSL to include edit option
- Make references between documents
  - Edit schema to allow linking between XML's within a project
  - Edit XSL to show links to User
- Make RTM
  - Make XSL to show RTM
- Downloadable HTML output
  - Make button to download HTML for uneditable viewing

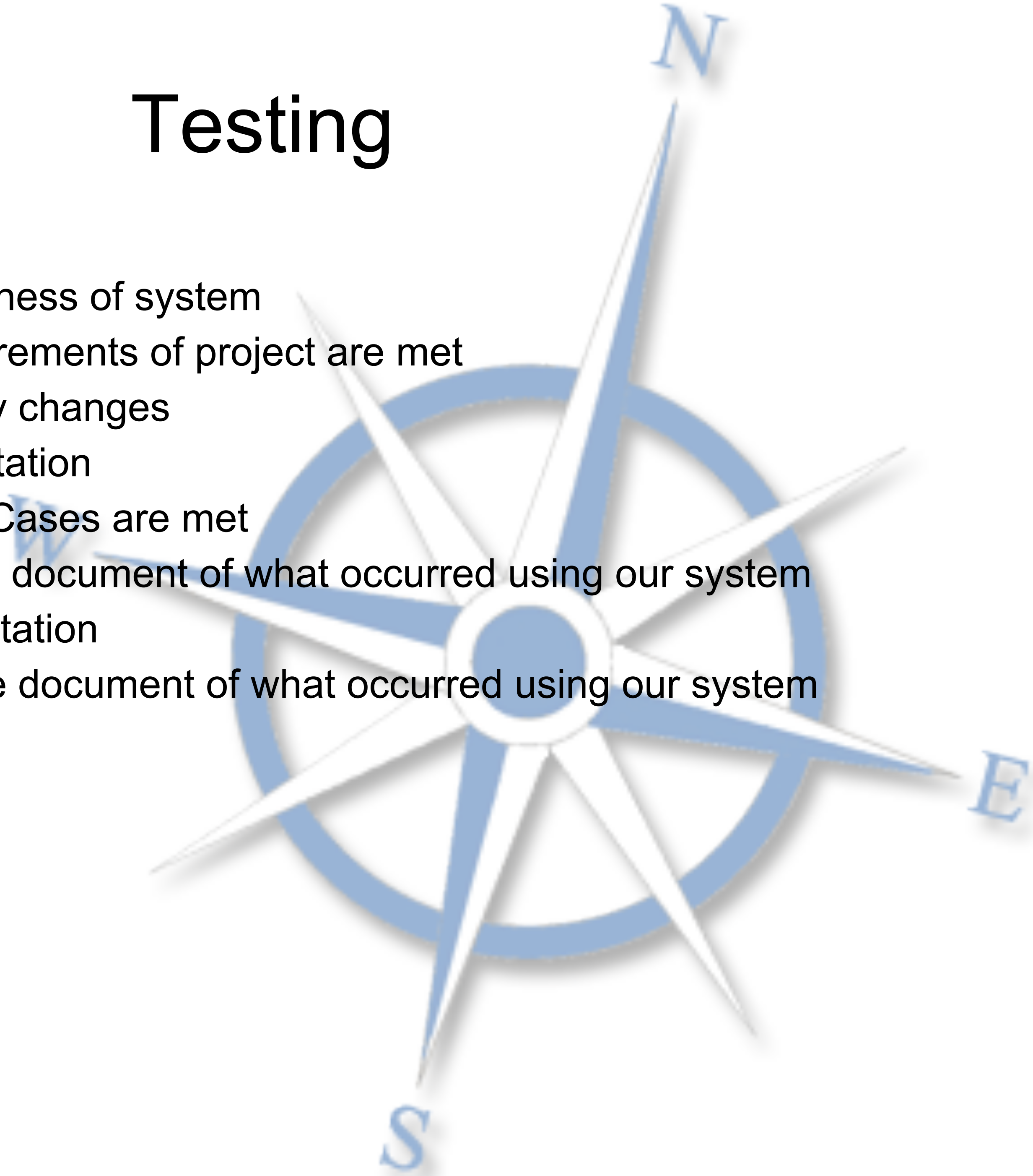


Creation Burndown

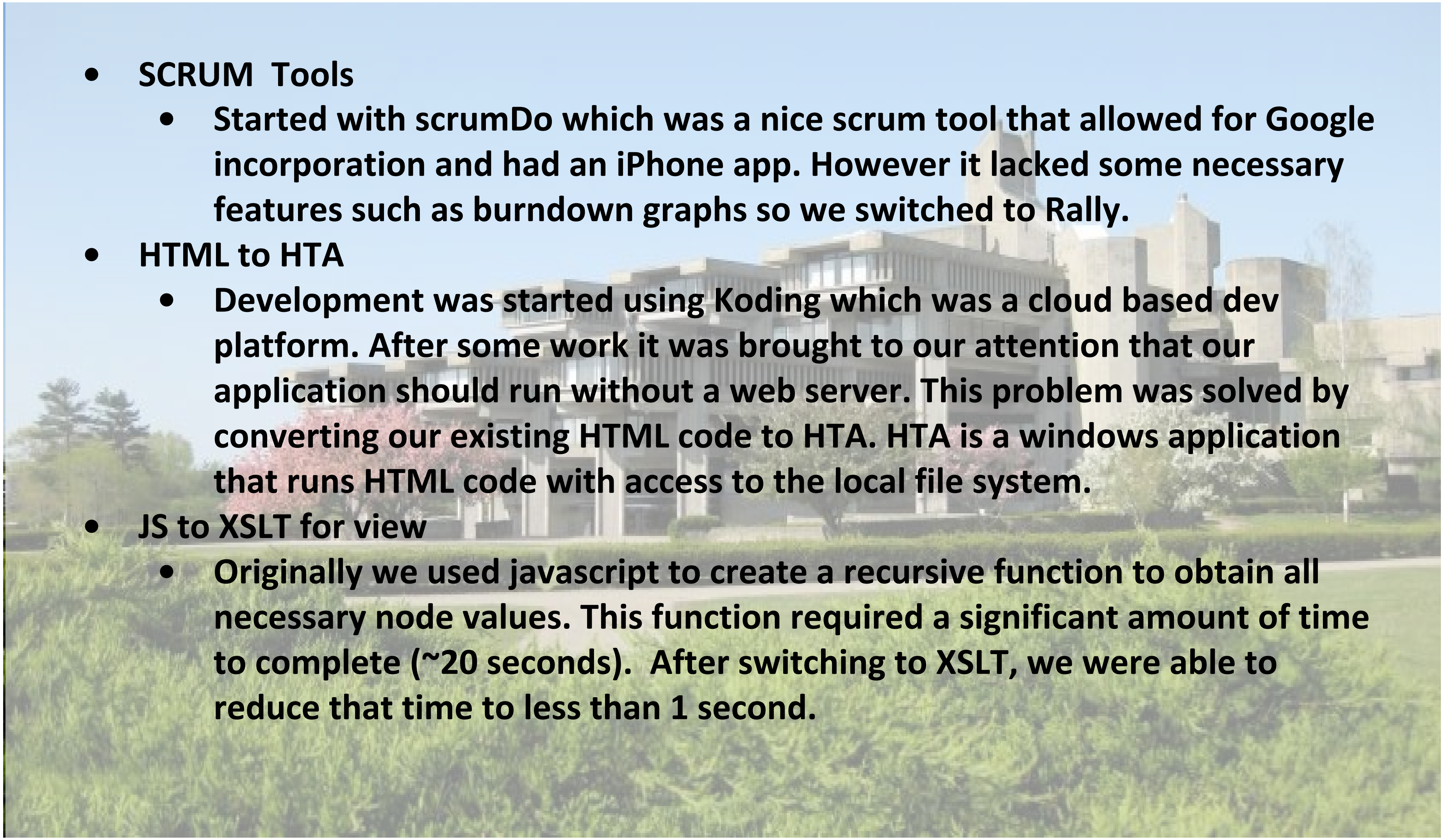


Testing

- Determine completeness of system
  - Ensure all requirements of project are met
  - Make necessary changes
- Use Case Documentation
  - Ensure all Use Cases are met
  - Make Use Case document of what occurred using our system
- Test Case Documentation
  - Make Test Case document of what occurred using our system



Changes



- 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 Rally.
- HTML to HTA
  - Development was started using Koding which was a cloud based dev platform. After some work it was brought to our attention that our application should run without a web server. This problem was solved by converting our existing HTML code to HTA. HTA is a windows application that runs HTML code with access to the local file system.
- JS to XSLT for view
  - 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

Check out our sweet app bro

Questions?