Today's software development projects are challenged by factors such as increasing size, complexity, and the unavailability of developers with the proper expertise. With the advent of advanced networking and communication technology, the practicality as well as the necessity of distributed software development is now being realized. [1]

With the application of software tools for distributed development software projects can incorporate the talent and efforts of remote developers. Expert developers can assist with technical issues without relocating to be on the primary development team. Individuals or separate development organizations can develop components of a system remotely. Software projects utilizing remote development efforts can benefit from reduced travel and staff recruiting costs.

However, distributed software development tools are still in their infancy. Many support issues remain which challenge the effectiveness of distributed software development. [2]

In this research, we propose to select and implement a prototype software tool which supports distributed software development. A literature review will be conducted to assist in the selection of a novel and interesting tool. Several factors that may influence the selection of the tool are:

- The tool must be interesting from the software engineering research perspective. Specifically, this research will look at issues with software engineering tools as they apply in a distributed setting.
- The tool should feature interesting collaborative aspects from the standpoint of Computer Supported Cooperative Work (CSCW) [3].
- The tool should expand upon current research in the field.
- It is desirable that the tool could be developed within the framework of existing CSCW development projects at Virginia Tech. The CSCW development group provides an available user population for evaluation of the tool.

A literature review will be conducted to:

- Identify problem areas in distributed software development
- Discover current knowledge and issues relating to software tools that support distributed software development

Based on the background research, a finite set of issues will be identified and studied through the development and deployment of the prototype tool. The research issues will drive the specification of the functionality of the prototype tool in order to facilitate their study.

Ideas for possible tools:

- A collaborative system for the distributed specification and prototyping of Graphical User Interfaces [4]

Such a system would allow distributed stakeholders to collaborate on the look and feel of a system’s user interface. A collaborative editing space would allow for the shared editing of windows at the component level.

Rapid prototyping is a common software methodology. A customer frequently tests a prototype and the developer seeks feedback. A distributed tool could support a distributed version of this methodology.

A prototyping environment could include annotation capability.

One possible implementation of this tool could be as an extension to the existing "room editing" services within the MOOsburg project. This approach would place a greater emphasis on the role of HCI for the project.

- A distributed software issue (bug) tracking system
This issue tracking system would place special emphasis on supporting distributed software development. This system could be web based, which would ease the task of content distribution among sites.

Common problems with issue/bug tracking systems:

- Incomplete or vague issue descriptions.
- Interfaces are too flexible. They allow users to deviate from creating consistent content. When the content in the database is inconsistent it becomes less valuable.
- Inaccessibility of the author to elaborate on the details of the software issue.

As a distributed tool the negative consequences of incomplete data in a bug tracking system are even greater than when the entire development team is co-located.

- A distributed software issue (bug) tracking system which integrates test cases, and scenarios

Few software development tools support both issue (bug) tracking and test case integration. Combining these two features could make for an interesting tool.

A variation of a distributed issue (bug) tracking tool would be a distributed requirement-tracking tool. A system could track the completion state all known system requirements. Validation test cases could be grouped with the requirement records. Validation tests could consist of a scenario script that describes how to validate the system requirement.

- A distributed source code and document revision system

This type of tool already exists. What would be some novel extensions?

Two different approaches to the development of a software support tool exist.

1. Develop a novel tool.

A novel tool could be more interesting. The research could focus on the tool selection, specification, and design, as opposed to usage studies. Identification of a novel tool is challenging.

2. Develop an instrumented version of an existing tool.

The implementation could include special features to study how the tool is used. Usage studies could be conducted to learn more about the problems and issues related to the software tool and tasks. Critical mass issues would exist with usage studies. Who would the user population be?

REFERENCES


