1-1-2021

Traceability Managaement in Software Development

Harry S. Delugach

Traceability Management in Software Development

Harry S. Delugach, Ph.D.

Associate Professor, Computer Science Department

OKT N-351

(256) 824-6614

delugach@uah.edu

RCEU21-CS-HSD-02

Traceability Management in Software Development

Harry S. Delugach, Ph.D.

PROJECT DESCRIPTION

One critical factor in the success of any software package is the degree to which it meets both customers' and users' goals, expectations and requirements. There are several steps in software development aimed at accomplishing this, starting with gathering and articulating those goals, expectations and requirements. For most projects, however, as development proceeds, the actual capabilities of the software tend to drift from the original requirements. One way to manage this problem is to provide what is called **traceability** during development, whereby anything the developers produce is explicitly traced back to one or more requirements. That way, each feature can be shown to have been designed, implemented and tested according to the original requirements.

Most students (and many professional developers!) struggle with traceability, both in understanding its purposes and in the use of tools for accomplishing it efficiently and effectively. Tools for supporting traceability in a project each have their own particular niche(s), advantages and disadvantages. Students need a way of learning how to include traceability knowledge in the

projects they undertake, in such a way as to minimize the burden of maintaining that knowledge, while maximizing the potential benefits of having access to that knowledge.

STUDENT DUTIES, CONTRIBUTIONS AND OUTCOMES

Student duties. The student will first familiarize themselves with the basic concepts involved in traceability – why it is important, what things need to be traced, and to what level of detail. Next the student will survey popular tools that are currently used (at least in part) to support traceability. Three or four tools will be selected based on their availability to students, their appropriateness to medium-scale software development, and their ease of use. These tools will then be explored in-depth, with the aim of providing a practical student user's guide for each one. Each user's guide will consist of a description of the tool, its pros and cons, how to acquire and install it, how to set it up for a short-term (one or two semester) software project, and examples showing how it can be used on a day-to-day basis. These guides can then be used by software project classes.

Contributions of the student. The student will create three or four ready-to-use guides for traceability tools. Since the guides will be written by a student for other students, it should be understandable and usable. Care will be taken to provide a justification for traceability, in

practical terms that students will find accessible. These guides will then serve to help them improve the quality of traceability information recorded in their own projects, thereby increasing the overall quality of the team projects for hundreds of future students. The research student will also produce a white paper describing what the student considers to be the most important purposes of traceability, and how using each tool's guide will further those purposes.

Outcomes for the student. Most computer science students undertake jobs that involve software development, often in a team environment. Usually their particular job will focus on one phase of software development – design, testing, etc. – so that their perspective on the whole project may be limited. Understanding traceability in software development will introduce them to concepts that will be useful in all phases, so that they know where their particular work fits in, as well as the importance of maintaining their share of traceability knowledge for others.

Participation in this project will give the student practical and in-depth exposure to both the theory and practice of traceability. The long-term benefit for the student goes beyond understanding the particular tools in this effort; it will provide them with the background to understand production software development from beginning to end.

STUDENT SELECTION CRITERIA

The student must possess basic programming and debugging skills, and be competent in a programming language at a level and scale comparable to CS 321 or CS 307. Experience in actual software development is desired but not required. The student must possess good technical writing skills, and show examples of their technical writing, although a technical writing course is not required. The student should also have some experience in installing software on one of the major platforms (Windows, macOS, or Linux). **This project is open to students at all academic ranks and from any academic discipline.**

FACULTY MENTORSHIP

The faculty mentor has taught software development for many years, and has also been involved in the development of several significant software packages. The mentor will therefore serve as the focal point for the student's efforts. The student will be assigned office space near the mentor's office in the Computer Science Department. The mentor and student will meet three times a week to evaluate progress and plan the next activities. There will also be one one-hour review conference/demonstration with the student every other week. Additional meetings with the mentor will be available as needed.

SAFETY AND CONTINGENCY PLAN

If health and safety guidelines require it, this project can be accomplished entirely remotely, as long as reliable video-conferencing with screen sharing can be supported. Most of the work of the student consists of exploring traceability concepts and tools on the Internet. The user's guides and white paper to be produced can be exchanged electronically for review and revision without requiring the student's physical presence. The only change to the stated mentorship plan is that the student would not be assigned office space, but instead would be expected to work from their own residence.