CS 410 Software Reverse Engineering

SNHU 3-1 Journal

Paloma Rodriguez 1/22/2024

* **Define**: What is software requirements engineering?

Software requirements engineering is the process of making the software what we need it to do and turning it into a clear description, such as how it should perform, or how we would like it to be set up. This is made clear through steps like “tradeoff studies, analysis and prototyping” (Fahmi, Choi, 2007) or “requirements elicitation, analysis, specification and validation” (Fahmi, Choi, 2007).

* **Purpose**: Why is software requirements engineering an important part of the software development life cycle?

Software requirements engineering is an important part of the SDLC because it helps each individual on the team have a common ground of understanding and even provides clear documentation of what the needs and expectations are of our clients. This provides a sturdy foundation for every step from design, coding, and testing. Each member can communicate with one another and understand what is expected and provide efficiency if an issue may arise and expedite project goals and produce a higher turnover rate.

* **Comparison**: How does the approach of software reverse engineering differ from the approach of software requirements engineering?

Software reverse engineering focuses on analyzing the existing software we have and understanding its mechanics, design, and structure and this is usually done without any prior knowledge or access to the software’s original code and really challenges us to understand how things are working the way they are. On the other hand, software requirements engineering it just trying to define, document and manage the requirements of the software itself and is a process of gathering information so that we can start to actually create the software.

* **Impact**: What are your thoughts on the proposed new integrated approach of round-trip engineering and its impact on the computer science field?

To my understanding, in simple terms, round-trip engineering is when we code but are constantly updating and going alongside our model which serves almost as inspiration or what we are meant to achieve, can be shown, or updated through design diagrams or specifications we dictate. In this case, its impact on the computer science field can be great! Not only are creating another mode of deliverance of communication on a team because just because it works for one team, but it may also not work well for others! But we can also have a consistency maintained throughout the design and then the implementation, which reduces of risk of issues arising and making updates when required.

Resources:

S. A. Fahmi and H. -J. Choi, "Software Reverse Engineering to Requirements," 2007 International Conference on Convergence Information Technology (ICCIT 2007), Gwangju, Korea (South), 2007, pp. 2199-2204, doi: 10.1109/ICCIT.2007.228.

<https://ieeexplore-ieee-org.ezproxy.snhu.edu/document/4420580>