**Module 4: Critical Thinking**

**Social Contract Theory—Rawls' Theory**

Social contract theory is an agreement among individuals within a social group to abide by certain rules and laws [1]. In other words, the theory considers what is socially acceptable. Additionally, “people live together in society in accordance with an agreement that establishes moral and political rules of behavior” [2]. Therefore, society dictates what is normal and abnormal, what is acceptable and what is not acceptable.

Within the realm of software engineering, there are many practices that are considered rules established by the software community and society as a whole. Some of the arguably smaller social rules include commenting code, adding tabs to code, making code readable with variable names, etc. Commenting code and making code readable are examples of positive behaviors that follow the software society’s rules. These are just the smaller examples, larger examples can also be found in the software engineering realm.

A larger example of the social contract philosophy can be seen with big data and data mining. Data has become a big issue with software consumers. This has become a huge issue because private user data is being used without consumers fully understanding what is happening. This has ended up with people losing trust in tech companies. In this example, it would be a negative example of software engineering not following what is acceptable by society. Even though some of the data mining may lead to a greater outcome, it is not socially acceptable to implement user data without the users fully understanding what is happening and without the user’s consent.

Another area where the theory of social contract is integrated into the software engineering realm is in agile development. Agile development creates a team(which is our society) where the rules and acceptable norms are established [3]. Each team member must accept and follow the societal rules in order for the team to work in a cohesive manner. If any team member does not follow the norms the society will punish the team member justly and try to get the team member, and team as a whole, back on track with the social contract philosophy.

In summary, the social contract theory dictates what is socially acceptable and what is not socially acceptable. Instances of the philosophy can be found in small and larger examples of software engineering. Small examples include making code readable for other developers. Larger examples include big data, data mining, and the agile team development process. Therefore, software engineering is subject to the rules of our entire society as well as the rules of sub groups such as an agile development team. Being subject to the rules of society keeps software engineering ethical within the realm of society.

**References**

[1] chris.drew.98031506. (2023, August 27). Social Contract theory: Definition and critique. Helpful Professor. https://helpfulprofessor.com/social-contract-theory/

[2] Social Contract theory. Ethics Unwrapped. (2022, November 5). https://ethicsunwrapped.utexas.edu/glossary/social-contract-theory#:~:text=Social%20contract%20theory%20says%20that,a%20divine%20being%20requires%20it.

[3] Systems, K. P. C., Power, K., Systems, C., Vergata, U. of R. T., Cagliari, U. of, & Metrics, O. M. A. (2014b, May 1). Social contracts, simple rules and self-organization: Agile Processes in Software Engineering and extreme programming. Guide books. https://dl.acm.org/doi/abs/10.5555/2813544.2813565