

Name: Arjun Dass

CWID: 10416575

Course: Agile methods of Software Development.

Q1. Describe at least two risks of using prototypes rather than documents to describe requirements and designs. Hint: not all of the actors involved in DSDM are software developers.

Ans. The risks in adopting prototypes over documents are as follows:

- * Development take more time and effort as the prototyping model is expensive since we develop actual throw-away system rather than document.
- * Misinterpretation of objectives by both user and developers may create some confusion. User might have some high expectation about that feature while at the same time developers might undermine his statement calling his statement as an extra feature demand raised by a user.
- * Prototypes might increase the development time excessively. Developers might spend more time on developing prototype rather than on actual development of the user objectives.

Q.1 For each of the risks you just identified, describe some other artifact that could be used besides a prototype (or, in addition to a prototype) that would mitigate the risk.

Ans. The solutions are as follows:

- (a) Instead of just providing the user with a simple standalone prototype model, we can provide the user with a working prototype with some functionalities.
- (b) Prototype development should be the last thing to be done in the development cycle of each Scrum where the other development items should be given priority over prototype. If the problem is complex enough to require more time, we can test it within the environment with the developer test cases.
- (c) We should have a reusable prototype so that for every cycle we can reuse and modify the existing prototype rather than creating a new one for every sprint.
- (d) The roles of the stakeholders should be cleared and explained beforehand hence the objectives might not clash with one another.

(3) For each of the mitigating artifacts you just described, estimate the cost of producing the artifact and compare it to the cost of producing the prototype.

The estimation and comparison of the costs for the solution and the prototypes are as follows:

- (a) The user will be given a standalone for which no coding is required to perform the logic of the prototype hence, if it take 12 days develop a working prototype, one might just develop a standalone, non-working prototype in 2-3 days.

(b) We might just skip developing prototype if more time is spent on the development of the real solution hence the cost of this fix varies with each sprint. Some scrum might develop a fully functional prototype while in some of the sprint we might just develop a prototype having a UI only.

(c) Standalone prototype may be helpful, but only for initial phases. Evolutionary prototype solves the issues of starting from the scratch each time you develop a prototype. In this rather than building prototype for every sprint, we need only one which can be modified again for the next sprint. Hence the cost reduces for such prototype.