Name Alex Lundin

Class SE 4381.501 Project Management

Assignment Homework #1

Position on the software crisis

**Statement**

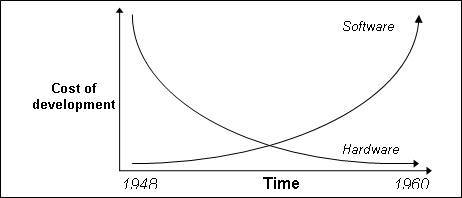
We should attempt to plan Software Projects. Doing what we can, in terms of meeting deadlines, with no formal planning in place is not good enough. We must attempt to improve the “software crisis”.

**Intro**

Software is the business of America today now more than ever. Every field of Engineering has the potential to cross paths with the Software Development cycle. With so many avenues intersecting, it is imperative that we improve the processes to keep projects on track for completion. Mainly we need to improve the methods we use to estimate, plan and track projects so deadlines are met in a timely and cost effective manner.

**Body**

The “software crisis” is a term coined in the 1960’s. This negative description stems from the lack of professional structuring around Software Program Development. Up until this point in the early 1960’s, writing software programs was viewed as a craft, or trade skill, more than a scientific work of production [1]. 4 fundamental indicators of this crisis are: software project not followed to completion, debugging took more time than writing the program in the first place, the functionality provided was not actually what the end user wanted and high maintenance costs[1]. Figure [2] shows the turning point of Hardware and Software costs and marked the beginning of the “software crisis”.



[2] < http://www.chris-kimble.com/Courses/World\_Med\_MBA/Software\_Crisis.html

There a many different Software Project Management ideologies today. Agile uses methods that remedy the previously stated indicators of the Software Crisis. Agile gives plenty of structure to keep the project on track for completion [3]. It also allocates time for team members to debug their work on a smaller scale by breaking the project into tiny two week sprints [3]. Also, Agile accounts for end user involvement in the process, to ensure the features provided are actually needed [3]. And finally the Software produced by Agile is easily maintained due to the small modular based tasks [3].

We can fix the “software crisis” by implementing structure around each project. It is possible to streamline design time and reduce the cost for creating programs. Today, more than ever, Engineering as a whole depends on the choices of Software Engineering as a discipline.

**Conclusion**

Leaving the Software Development Cycle in crisis mode is not a viable option. Too many branches on Engineering depending on timely completion. We must make efforts to improve deliverables.

**References**

[1] “The Software Crisis,” < http://www.chris-kimble.com/Courses/World\_Med\_MBA/Software\_Crisis.html>. 01/22/2018.

[2] “The Software Crisis,” < http://www.chris-kimble.com/Courses/World\_Med\_MBA/Software\_Crisis.html>. 01/22/2018.

[3] “Agile project management with Scrum,” < https://www.pmi.org/learning/library/agile-project-management-scrum-6269>. 01/22/2018.