Task D: Teach a technical topic

COMM 3700

|  |
| --- |
| **Your name**:  Mohammad Al Jokhadar |

Develop a brief 2-minute lesson with the following parameters:

1. You will be assigned a concept from this site <https://www.pluralsight.com/blog/it-ops/20-it-terms>
2. The audience is non-technical people who are computer literate
3. Develop an analogy for the concept.
4. Work in the significance of the concept to your diploma (ex. Programming, web, cyber, systems admin, data analytics, DBA)
5. Provide a real-world example of the concept
6. Develop a 2 minute training session that incorporates the technical training process, that incorporates
   1. Teach
   2. Practice
   3. Confirm
      1. Teach, Practice, Confirm can all be accomplished using any of these methods: Lecture, Video, Handout, Questions, Question and Answers Leading to more Questions, Games, Inclass Activity, Demonstration, Group Discussion, Student’s experminent, Roleplay, Student’s read something then comment.
7. Deliver the lesson in class.

Imagine a massive factory, where everyone are being haunted by nightmares of demands, deadlines, due dates, expectations and other horrific entities. Production on the factory line is going slower and slower, and yet, due to the slowness of the productivity on the factory’s shores, more deadlines are rising to the surface of the collective ego that unites all these workers in their mutual struggle : The struggle of getting things done!

The factory in question is our working life. It is the life of our times, where disruptive technologies usher in the aeon of a new technological age. If we picture, in our recent history, how cellular phones have changed our world, or the arrival of Artificial Intelligence in our world. One of the most exciting innovations of our time could provide a solid footing for our dear workers in their struggle with the ghosts that haunt them. The answer is Agile Development.

Why would this be a choice, you may wonder? We are not all developers. We do not have any inclination to further complicate our lives with a new language, with all new syntax and vocabulary. The response would be to best describe the term itself. Agile Development is a new way of seeing the workload we have, and our relationship with it. The larger the expectation, the more stress and worrying that we would all feel towards them. We fail to see the journey as one-step-at-a-time, but always trying to see the long-awaited goal in the mist of our worries about our journey to it.

Agile Development works like this :

The factory owner realizes that the workers are struggling with the workload. The factory owner discovers the idea accidently, and recommends to the workers to not worry about the workload itself, and have their complete trust in the aspect of prioritizing the workload. The factory owner produces a list of requirements that need to be met by the factory’s clients ( known as *User Stories* in the Agile Development world). The list is read to the workers. Afterwards, the workload is divided in chunks, and then everyone would collectively work their hardest for short *Sprints*, which is a set period of time. Realistically, with careful planning, along with encouragement and a great deal of support, this idea could be the support that the workers needed to realize their goals. Furthermore, the factory owner recommends to the staff to complete each portion of the work, asking the supervisor for their input on it, and then submitting it after careful review. The owner would add new goals and prioritize set goals in accordance to the daily productivity of the team, and in response to changing realities in the workplace (Learn.microsoft.com, 2024).

This concept is very tech-oriented by nature, but I see that with the advance of Artificial Intelligence, the role of the factory owner could be replaced entirely by Artificial Intelligence. If there is a current flow of productivity data on a daily basis, as in each worker’s ability to meet the realistic goals (ideally!) set in place, and to calculate the average of performance along a long-period of time, such as each *sprint*. Then, feedback could be generated and delivered to each worker, with constructive criticism being its core. Furthermore, the worker would have a daily meeting with the supervisor to review each day’s accomplishments, and to ensure that the task at hand has been completed correctly. This meeting insures a dialogue between the worker and the supervisor, and a continuity of understanding of each individual’s expectations. I see it as something along the following :

“Thank you for your hard work! Today, you have done well on all the steps towards your productivity goal, but needed some extra help on two points. Keep it up! Here is my recommendations to help : …”

Adding a competitive spirit to the worker’s lives could be a helpful morale boost, and a way to reward their efforts. The ultimate reward is the feeling of accomplishing a goal we have set for ourselves!

How would be practice this notion? By taking note of our daily accomplishments to meet our goals, to realistically re-arrange the goals to be met in a realistic way, and to work in short spurts and keep a regular record of our sprints so we can review what we have done, and how we could improve ourselves!

Agile development, when borrowed from the programming world, could be a productivity-based Disruptive Technology. Work would be divided into realistic goals, arranged and re-arranged in accordance to the realities of the day, and we would work on short sprints, and keep a record of our accomplishments. We would review what we have done, and check if it meets the goals that we have set for ourselves. We would be in this case the factory owner and the workers combined, as we are the best entity to judge our own efforts and accoplishments!

# Sources

Learn.microsoft.com. (2024, September 26). *What is Agile Development? - Azure DevOps | Microsoft Learn.* What is Agile development? <https://learn.microsoft.com/en-us/devops/plan/what-is-agile-development>