1. In your opinion, experience, and/or based on your research, please discuss the following areas:
   * + *What is the impact of agile practices on the creation of System Test Plans?*
     + *How do these methodologies affect aspects like test coverage, resource allocation, and adaptability in the testing process?*
     + *Are there any emerging trends or innovative strategies in software test planning?*

Hey all,

The agile methodology focuses on iterating over user stories to build the system one piece at a time. This means that the same idea should be applied to the System Test Plans. It should be iterated over as new pieces of the system are developed.

Different methodologies should not have a huge effect on the amount of test coverage. At the end of the day, enough tests should be written to cover as much code/functionality as possible. Resource allocation may change depending on when the methodology calls for a given task in the test writing process. For example, you may allocate a lot of time before development starts writing testing based on user stories. Then automate them at the end before deploying. Every project, regardless of methodology, will require adaptability in the testing process. Different teams will have different processes, and even if the team is the same, a different system may require different testing processes.

An emerging trend that I’ve seen in software test planning is using AI/ML to better facilitate testing. In fact, on one of my projects, there is a team of 3 dedicated to this. One of the biggest drawbacks is getting people to believe the testing developed using AI is trustworthy. A lot of the team’s time is spent training the operators of the system on how to understand the testing. A lot of people don't understand it, and therefore, don't believe it can be effective.