

ACCEPTANCE TESTING

(UAT Initiation and design)

ABSTRACT:

User acceptance tests are tests that are executed by companies that are updating or integrating a new IT system. The system is usually procured by another company which has developed the system; acceptance testing is important to validate functionality and ensure that the system is working within the target IT landscape. A lack of research around management of acceptance testing from the customer's perspective was identified. This thesis aims to explore how management practices affect acceptance testing and related resources, as well as investigate if automated acceptance testing can be applied by finance companies interested in upgrading or replacing critical systems. Data was collected in three steps aimed to show the issue from different perspectives such that biases in answers from interviews with testers could be adjusted for. As a first step a tool- assisted literature review was conducted using GPT-3 (Generative Pre-trained transformer 3) to understand the academic consensus. Then, employees that had worked with user acceptance testing were interviewed to gain insights relating to how people think about the problems that arise during user acceptance testing. Finally, log files from test and production systems were datamined to understand how consensus and thoughts translate to actions in the testing environments.

UAT MANAGEMENT:

As described in the introduction, managing user acceptance testing can be a cumbersome task associated with a great deal of challenges that have to be overcome (Poston et al., 2014a). Traditionally, UAT will follow a detailed plan which includes all requirements and desire including the customer in an iterative development process is not always desirable, UAT often follows an almost waterfall lifecycle even when part of an agile project. Related to this, it should be noted that IT projects using the project.



RESOURCE ALLOCATION FOR UAT:

In general software development smaller teams will have a higher productivity per developer as adding additional manpower has diminishing returns. While this in general holds true for QA personnel as well, it is very difficult to estimate the number of testers needed for a project. A common approach is to try to preserve a ratio between developers and testers - although such ratios diverge quite a lot depending on the nature of the software project. While one may intuitively assume that more testers will result in better quality code, the reverse has actually been observed: developers seem to assume less responsibility for the quality of their written code when the ratio testers to developers is higher. To add some context to this, a common developer.

Operational Acceptance Testing

This slide is 100% editable. Adapt it to your need and capture your audience's attention



CONCLUSION:

We formulated three pillars of knowledge to follow and analyze with regards to user acceptance testing (UAT): consensus, thoughts, actions. Following this, we designed a data collection method for each of the three pillars and compared and contrasted them to uncover new insights into user acceptance testing. The conclusions from our thesis, recommendations to the industry, a discussion regarding the significance of our findings to academia, as well as summary of interesting topics for future research are presented in this chapter.