Experiments

Spencer Smith
McMaster University, Canada
smiths@mcmaster.ca

Jacques Carette
McMaster University, Canada
carette@mcmaster.ca

Olu Owojaiye
McMaster University, Canada
owojaiyo@mcmaster.ca

Peter Michalski
McMaster University, Canada
michap@mcmaster.ca

Ao Dong
McMaster University, Canada
donga9@mcmaster.ca

1 User Experiments

1.1 Usability Experiment

The purpose of this experiment is to assess the usability of each software package and collect qualitative data. An initial step is required to define a set of tasks for the experiment. The guideline below would assist domain experts in defining a list usability tasks for each domain. This list will then be used to evaluate the usability of each selected software package. It is expected that the list would allow experimenters perform tasks in different aspects of the software package and provide their evaluation using a standardized usability questionnaire (https://www.usabilitest.com/sus-pdf-generator- 20-29 , https://uiuxtrend.com/pssuq-post-study-system-usability-questionnaire/ - PSSUQ) - (we need to decide on the questionnaire to use eventually)

1.1.1 Task selection criteria for domain experts

The following criteria will be considered by the domain expert when defining the tasks for each software package.

1. Tasks should be executable for subjects with novice to intermediate experience.
2. All tasks should take no more than one hour.
3. Tasks should include the basic/common use cases of the software package.
4. Include tasks that require sequential or hierarchical steps for completion

1.1.2 Procedure

1. Survey participants to collect pre-experiment data (background, experience of subjects)
2. Participants perform tasks based on task defined by domain experts.
3. Observe the study subjects (take notes, record sessions(OBS screen recorder), watch out for body languages and verbal cues)
4. Survey the study subjects to collect feedback (post-experiment interview)
5. Prepare a summary report of experiment

2 Modifiability Experiment

This experiment is designed to gather qualitative data regarding the modifiability of each software package. The initial step outlined below produces a list of likely software changes for members of a scientific computing software domain. This list is then used to analyze whether specific software packages have been designed to accommodate these expected modifications.

Parnas and Clements' definition of likely changes has informed this analysis: "If the system is required to be easy to change, the requirements should contain a definition of the areas that are considered likely to change. You cannot design a system so that everything is equally easy to change. Programmers should not have to decide which changes are most likely" Parnas and Clements [1986].

2.1 Procedure

1. *Domain Expert*: List all likely changes that a domain expert might make in a software package in the domain.
2. Using the short list of software packages listed in the Methodology document:
   a. Identify which of the above changes each package is likely designed to accommodate.
   b. Prepare a short report outlining the likely changes for software in the domain, and which changes each software package is likely designed to accommodate.

References

David Lorge Parnas and Paul C Clements. A rational design process: How and why to fake it. *IEEE transactions on software engineering*, (2):251–257, 1986.