Reference:	**CSI-Study**
Topic:	**Crowdsourcing Supported Inspection (CSI)**
Lecture-Type:	Bachelor thesis, Master Theses
Start:	As soon as possible
End:	To be defined
Contact:	Dietmar Winkler (dietmar.winkler@tuwien.ac.at)
	Stefan Biffl (stefan.biffl@tuwien.ac.at)

Background

Traditional Software Inspection is a well-established approach to identify defects in software artifacts and models early and efficient [1]. In industry practice, a team of inspectors apply traditional (best-practice) software inspection on a pen & paper basis. Because of resource restriction, i.e., the availability of experts, inspection duration is limited to about 2 hours [1]. Thus, insufficient method and tool support hinder efficient defect detection in large software models.

Human computation and crowdsourcing processes [2][3] can help to overcome these issues by involving a team of distributed crowd workers (addressing resource issues) and by splitting large inspection tasks in a set of small tasks (addressing limitations for inspecting large software models). However, the crowdsourcing supported inspection process need some adaptations to address (a) software inspection needs and (b) crowdsourcing capabilities.

Figure 1 presents an overview on the CSI process with tool support including three main phases: (1) Inspection preparation, (2) Defect detection, and (3) Follow-up. Based on a reference document, e.g., a software specification, crowd workers identify so-called *Expected Model Elements (EMEs)* that represent the building blocks of a software system (Step 2a). In a second step (Step 2b) the crowd workers use these EMEs to identify defects in a software model, e.g., an Extended Entity Relationship (EER) diagram. This second step allows identifying missing EMEs and wrongly modeled EMEs.
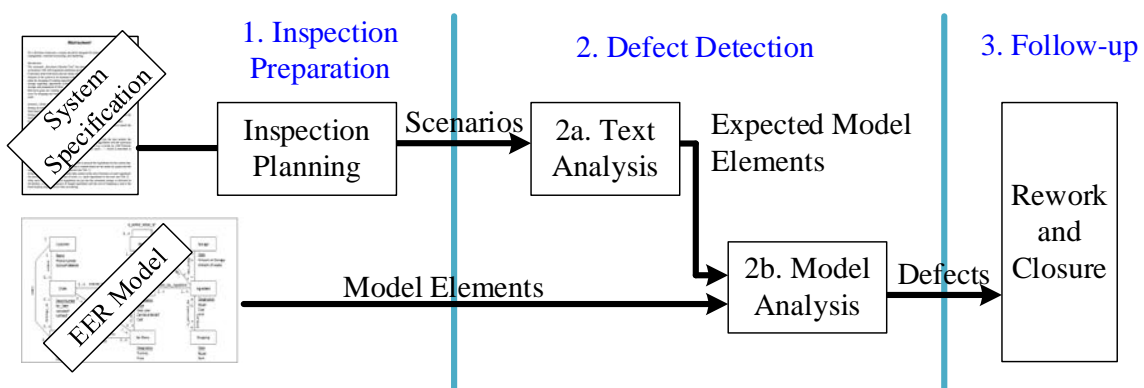

Fig. 1: Crowdsourcing Supported Inspection process [4].

For the evaluation of the CSI process, we designed a controlled experiment (a) to investigate the feasibility of the CSI process and (b) to gain insights into the defect detection performance of a CSI process and a traditional best-practice pen-and-paper (P&P) software inspection process, i.e., the effectiveness of individual participants. Figure 2 presents a brief overview of the study approach.
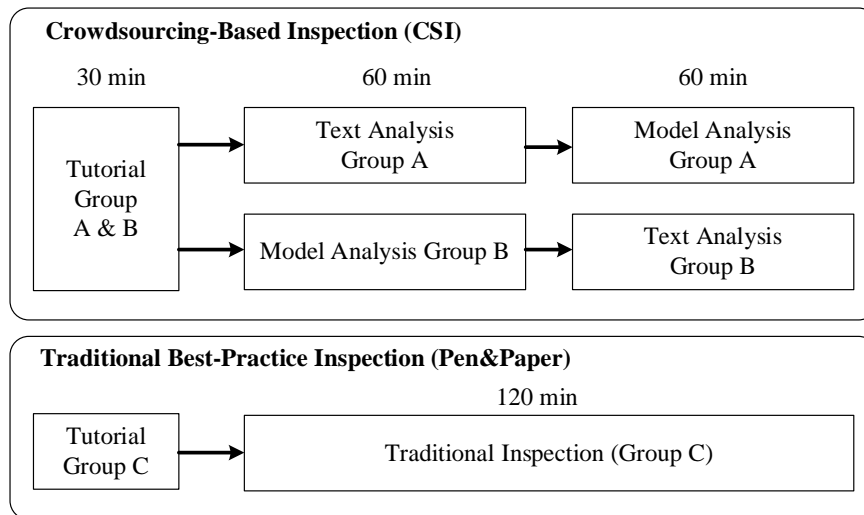
Fig. 2: CSI-Study design.

Embedded within the experiment team, the main goal of this topic is to contribute and support (a) study preparation, (b) study execution, (c) data collection and analysis preparation, and (d) analysis and evaluation. The study setting includes two groups (A and B) who execute defined crowdsourcing tasks and a control group (C) who perform a traditional best-practice inspection.

Tasks
Based on the goal specific tasks include:
- Study preparation. Implementing and review of the study material for the CSI group and the P&P groups. Preparation of the material for experiment execution.
- Experiment execution. On-site support and supervision of study groups prior, during, and after the study (planned effort is 4 half-days).
- Data collection. Data consistency checking and defect mapping, i.e., mapping candidate defects to real defects based on a given set of seeded defects.
- Analysis and evaluation of collected data with respect to research questions. The research questions will be defined in collaboration with the experiment team.

Expertise and Skills Needed
Based on the selected tasks, required expertise and skills may vary. For this project, the following skill set is recommended:
- Basics of planning and conducting empirical studies.
- Communication capabilities for participant supervision.
- Statistical interests and basic skills for data evaluation.
- Nice to have: experience with crowdsourcing application, such as Crowdflower.

You will learn
- How to plan and conduct empirical studies, especially controlled experiments.
- Research question elicitation and basics in paper writing (also relevant for thesis writing).

References

[1] A. Aurum, H. Petersson, C. Wohlin, C., "State of the art: Software Inspections after 25 Years", Software Testing, Verification and Reliability, 12(3):133-154, 2002.

[2] LaToza T.D., van der Hoek A.: "Crowdsourcing in Software Engineering: Models, Motivations, and Challenges", IEEE Softw. 33(1):74-80, 2016.

[3] Mao K., Capra L., Harman M., Jia Y.: "A survey of the use of crowdsourcing in software engineering"; In: J of Systems and Software, 28p., Online; DOI: 10.1016/j.jss.2016.09.015, 2016.

[4] Winkler D., Sabou M., Petrovic S., Biffl S., Kalinowski M., Carneiro G.: "Improving Model Inspection with Crowdsourcing", In: Proceedings of the 4th International Workshop on Crowdsourcing in Software Engineering, ACM/IEEE International Conference on Software Engineering (ICSE), Buenos Aires, Argentinia, 2017. (upcoming).