**Empirical Study**

* **Goal** of the study: We have developed a software visualization tool that illustrates the primary constituent artifacts of a software product (classes in the source code, product features, class-feature trace links) and their evolution across several versions.
* **Scope** of the study: We believe our tool assists developers with four different tasks. Thus we conduct this experiment to cover the entire scope of our tool:

1. *Product Comprehension*: questionnaire
2. *Feature Tracing*: quantitative **(Trace Links)**
3. *Product Evolution*: quantitative **(Scenarios)**
4. *Developer’s Rational and Intention*: questionnaire

* **Study**: In this study, we measure the tool’s usefulness (usability and utility) for each of the aforementioned areas. We will use this information to (a) improve the tool and (b) analyze the capabilities of the tool. We also aim to publish the results of this study in form of a conference paper.
* **Experiment:** This experiment include **quantitative** and **qualitative** evaluation of our tool:
* **Quantitative:**

We spend approx. 10-15 minutes to provide a brief overview of the tool to each participant. Each participant later receives one document which contains a list of recommended change scenarios relevant to a certain class across two subsequent versions of Apache Cassandra. These recommendations are either true or false. The recommended changes are divided into two sections:

1) Changes associated with a particular class **(Scenarios)**

2) Changes in associated trace links with the class **(Trace Links)**

**You are simply asked to:**

1. Read the recommended change scenario
2. Inspect the recommended scenario.
   * Regarding (**Scenarios):** Inspect the change in the code:
     + If you are **using the tool** you can easily select the class in the evolutionary graph and select the “Inspect Selected Element” in the eclipse plug-in to see the code associated with the class. You can select to view the associated github commits and Jira relevant issues.
     + If you are **manually performing the task** you need to search for the relevant classes in the associated package and continue with your inspection.
   * Regarding **(Trace Links):** Inspect the feature description and content of the associated class:
     + If you are **using the too**l you can easily click the feature and read the description in the document and again search the content of the code by selecting the “Inspect Selected Element” option.
     + If you are **manually performing the task** you need to open the features document and read the descriptions for the relevant classes in the associated package and continue with your inspection.
3. Reply with correct, incorrect or don’t know to each of our recommended changes.

NOTE: To respect your time, we limit each (**Scenarios)** task to 10 minutes and each **(Trace Links)** task to 5 minutes.

* **Qualitative:**

1. Complete a short Usability Questionnaire with only a couple of simple questions on the tool’s usability.

The entire experiment will take between 60 to 90 minutes.

**Notes:** While you are working with the tool:

* + - Please “think out loud”, i.e., say what you are currently doing or are aiming to do. Comment when you don’t know how to proceed and when you think you are stuck. Comment what you like/don’t like.
    - One of us will take notes on this “thinking aloud”/your comments.
    - At any time, feel free to ask any questions or make any comments you want.