**JTeC: ALarge Collection of Java Test Classes for Test Code Analysis and Processing**

**Authors:**

**\* FedericoCorò∗ SapienzaUniversityofRome Roma,Italy** [**federico.coro@uniroma1.it**](mailto:federico.coro@uniroma1.it)

**\* RobertoVerdecchia∗ VrijeUniversiteitAmsterdam Amsterdam,TheNetherlands** [**roberto.verdecchia@gssi.it**](mailto:roberto.verdecchia@gssi.it)

**\*EmilioCruciani∗Inria,I3SLab,UCA,CNRS SophiaAntipolis,France** [**emilio.cruciani@inria.fr**](mailto:emilio.cruciani@inria.fr)

**\* BrenoMiranda† FederalUniversityofPernambuco Recife,Brazil** [**bafm@cin.ufpe.br**](mailto:bafm@cin.ufpe.br)

**\* AntoniaBertolinoConsiglioNazionaledelleRicerchePisa,Italy[antonia.bertolino@isti.cnr.it](mailto:antonia.bertolino@isti.cnr.it)**

**DATE: October 5–6, 2020, Seoul, Republic of Korea**

**Introduction:**

Test automation has been actively pursued since the 90’s [17] as a solution to reduce the high costs of software testing and improve product quality [12]. A recent study by Zion Market Research [18] estimates that the software test automation market will grow from the 16 Billion Dollar sof 2016 upto 55 Billion Dollars bend of 2022.However, this insurgence of test automation also comes with challenges and risks, which several researchers have been prompt to identify and face.However, this insurgence of test automation also comes with challenges and risks, which several researcher shave been prompt to identify and face. All the above research efforts demand availability of a large data set of test code, to which the proposed methods and tools can be applied.

**ABSTRACT:**

There cent push towards test automation and test-driven development continues to scale up the dimensions of test code that needs to be maintained, analyzed, and processed side-by-side with production code. As a consequence, on the one side egression testing techniques, e.g., for test suite prioritization or test case selection.

**CCS CONCEPTS** •

Software and its engineering → Software testing and debugging.

KEYWORDS: GitHub, Java, LargeScale, Software Testing, Test Suite

ACM Reference Format:

**METHODOLOGY:**

In this section we report the method logy we use together to the dada o four data set. The process, illustrated in Figure1,consistsof six main steps:(i) GitHub repository filtering,(ii) Java repository identification, (iii) test classes identification, (iv) repository selection,(v) local storage of test classes, and (vi) preliminary quality filtering.

**Step1:** GitHub repository filtering. The first step of our process consists of indexing the public GitHub repositories, and is carried out in order to execute efficiently the subsequent phases of our process. In this step we first retrieve the name of the public repositories and the user name of their creators.

**Step2:** Java repository identification. Once obtained a local copy of the indexed repositories mapped to their programming languages, we can effort less ly retrieve the URL soft here postures developed in a specific language.

Beyond our own motivation, we consider JTeC could be used to the benefits of researchers in several areas, including: • Static analysis: static analysis of test code has been applied for many purposes, including, for example, the identification of test smells

**CONCLUSIONS AND FUTURE WORK**

We have presented the JTeC data set that makes ready available to the community of software testing researcher silage collection of Java test classes useful for several potential purposes related to test code analysis and processing. Our aim goes beyond the current version of the dataset: we strive towards the establishment of a continuously updated data set, collecting together rain single source the test code belonging to the vast majority of test cases publicly.