

**PURE Progress Plan**  
By Cyrus Eduljee, Supervised by Mohsen Vakilian

As multithreaded application development becomes more popular, an increasing number of programmers are having difficulty creating applications that behave as expected. *Keshmesh* is a plugin for the popular IDE Eclipse that makes bug pattern detection in multithreaded Java applications easier. *Keshmesh* analyzes the Java bytecode to find certain patterns that are commonly associated with concurrency related bugs, and it then reports them to the programmer using the Eclipse *FindBugs* interface.

I have been exposed to many new concepts while working on this project. For one, I learned the fundamentals of multithreaded programming. Multithreaded programming is unlike single-threaded programming in that extra attention must be paid to protecting access to data that is shared amongst threads. If access to this data is not protected, it can lead to unpredictable behavior that is difficult to detect. Contributing to *Keshmesh* ensured that I had a good grasp of the fundamentals concepts, and it exposed me to many of the tools related to ensuring multithreaded applications perform as expected. These fundamentals were further reinforced by reading the chapter on multithreading from *Java How to Program*.

Furthermore, I was also exposed to the Java API that supports multithreaded application development. Contributing to *Keshmesh* allowed me to explore all the tools the Java language provides for the development of robust and error-free multithreaded applications.

In addition, working on this project has exposed me to many of the common pitfalls
programmers encounter when working on multithreaded applications. For instance, a particular bug pattern that is often encountered is called VNA00-J. *Keshmesh* reports these bug patterns when it believes that access to a shared variable is not protected. Identifying these errors with *Keshmesh* has enabled me to become a better programmer because I have learned to identify and correct many of the common concurrency related bugs.

While working on *Keshmesh*, I primarily contributed to the evaluation of the application. One of the issues facing *Keshmesh* is the fact that it can report false positives, meaning that it reports a bug when in fact a bug does not exist. I ran *Keshmesh* on the open source *Apache Tomcat* in an attempt to identify bugs in *Tomcat* and false positive reports by *Keshmesh*. During this process, I constructed graphs for each shared data field that was reported as unprotected by *Keshmesh*. Constructing these graphs makes it easier to identify if the shared field is indeed protected properly, and it provides additional context so that the errors behind false positives can be identified more easily.

My goals for the remainder of the project are to evaluate *Keshmesh* on other open source projects and contribute to the open source community by identifying concurrency related bugs. I also want to compare the performance of *Keshmesh* with similar state of the art applications. Since the PURE program provided me with a good introduction to undergraduate research, I will look to get involved with research during my future semesters at the University of Illinois.