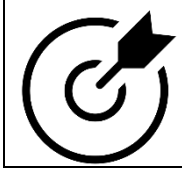
Software Systems Architecture

Lab 1 – Trade-off Analysis

Goal 	Trade-off analysis is important to choose between two solutions in a software architecture. The goal of this exercise is to develop two different solutions that have the same behavior and evaluate the characteristics of each one.

Task 	**Implementation:** The file places.txt contains several names of places. Create an interface with a method to verify if a place has a name in that file. Create 2 implementations of that that interface: (a) one that reads the file when it is initialized and search in memory when the method is called; (b) other that searches directly on the file when the method is called. **Analysis:** Create a test that allows you to perform a trade-off analysis between both solutions, evaluating the amount of memory consumed and the execution time. You can use any tool or approach to perform the measurement. Create a report to present your analysis and to show your conclusions.

Tips 	**Read file in Java:** The class BufferedReader have a method readLine() that return the line as a String, but there are new APIs that can do the same thing with less code… **Profiling:** VisualVM is a tool that was distributed with Java VM in the past. It can connect to a running program and collect data about the CPU and memory consumption. There are other tools that can be used for profiling. **Using the debug:** If you are having a hard time to connect the profiling tool to your application at the right time, you can run it in debug mode and put a breakpoint to stop the execution, connect and configure the profiling tool, and then execute the part of the code that you want.

Delivery 	The delivery will be made by uploading to the lab channel on Teams a .zip or .rar file containing the code developed and a .pdf file with the report. The report should present how the measurements were made and what are the results of the trade-off analysis. Answer this question based on your measurements: in which scenario you would use each one of the solutions?

Rules	
	**You cannot do:** Look at the solution of another student before trying to do yours; Copy the code or text of another student at any circumstance; Do the assignment together with other student. **You can do:** Learning or studying together about how to use a tool to use in the lab, without applying it directly in the lab exercise; Ask for help to another student that already finished his assignment after trying to do it yourself; Talk about solutions and alternatives without looking at each other code; If you are REALY stuck, you can take a look to the other student work, but you should close it before come back to yours.

More	
	**Challenge:** Can you think about an intermediate option that can balance both solutions? – 15 stars **Profiling Tools:** Can you search for another profiling tool other than VisualVM and teach the class how to use it? – 10 stars