***ILP Testing Approaches***

# **Approach 1**

One testing approach that could be used when it comes to the requirements selection from section 1.1 is that used in agile processes: ***integrating the testing process in the development stage***. This type of approach is known as ***proactive*** since we start the testing process at an early stage. As part of this approach, we could use techniques such as ***black-box testing***, where test cases are generated straight from the requirements themselves, rather than relying on actual information about the code itself, as well as ***grey-box testing***, where only some information about the implementation details is available.

# **Approach 2**

Another approach that could be used is that of ***performing testing analysis once the system is fully implemented***. This type of approach is known as ***reactive*** since we start the testing process at a very late stage. This approach uses techniques such as ***white-box testing***, where test cases are generated after being fully aware of the implementation details.

# **Approach 3**

One more testing approach that could be used is a ***combination of the two approaches described***. This approach essentially has the benefits of starting the testing process early, whilst also allowing time at the end to refine/ design test cases based on implementation details. In this approach, we mainly rely on ***grey-box testing***, where only some information about the implementation details is available.

# **Approach 4**

Yet another testing approach that could be used is that of an ***iterative testing approach***. If we were to split the development up into “iterations” or say “cycles”, then at the end of each of these “iterations”, testing is performed. This would perfectly integrate the testing process with the development process, allowing bugs to be identified before further progressing to the next “iteration”, and would also rely on ***grey-box testing***, where only some information about the implementation details is available.

# **Approach 5**

The last testing approach that could be used is that of the ***DevOps continuous testing approach***. In this approach, there is an already implemented automation tool and pipeline, which executes all the tests at once, and instant feedback is generated in case bugs/ errors are spotted. In the context of the ILP project, this is the least feasible option, since the project doesn’t scale up enough; however, it is an approach worth mentioning, since it essentially has very few downsides.