我们在做接口自动化的时候，根据接口文档进行接口需求分析，输出接口测试点。对每个接口运用等价类划分法，边界值法进行测试用例设计。然后搭建框架，编写测试脚本，执行脚本生成可视化的测试报告，对报告中的问题进行跟踪记录、提交Bug，对脚本存在的问题进行更新迭代。

我们在设计用例的时候，首先考虑正向的情况，再考虑反向的情况，验证每一个参数非法的情况。例如用户注册接口的测试用例，用户名合法的情况、用户名长度超出限制、未填写，包含特殊字符等。

在框架搭建方面，我们使用Python编程语言，将项目模块化，在test\_case包中放测试用例，data包中放测试用例的数据：yaml/csv文件，utils中放一些工具：md5、日志工具等。logs中存放日志文件。每个接口我们都创建一个py文件，例如用户登录接口，我们可以创建一个test\_api\_login.py，在这个py文件中创建一个同文件名的大驼峰命名的类，每一个测试用例定义一个方法。使用Requests库发送请求，设置请求方式Get/Post等，请求URL和请求参数。我们会使用ddt和yaml进行数据分离。在yaml中填写多条用例的请求参数和请求方以及如何断言。再使用@ddt.file\_data装饰器读取这些数据，执行每条用例。使用Pytest框架进行断言。加上allure库生成可视化的报告，在执行pytest命令加上--alluredir指定生成allure数据的路径，然后执行allure generate 数据路径 -o 报告路径，生成报告。