

Software Testing Case Based Learning Exercise

Ashish Sureka (ABB India), Paramvir Singh (NITJ India), Veena Saini (NITJ India), Saurabh Tiwari (DAIICT India)

Objective: To facilitate the concepts of security testing and functionality testing through real world case.

Topic covered in the class to undertake the study: Testing Principles, Test Objectives, functional testing, use cases and user stories, Non-functional Testing (performance, load, and security)

Case Study:

StalwartX (driver-less vehicle) is an emerging company with breakthrough technology. They are building autopilots and driver-less vehicles. Suppose you are a test manager at StalwartX. It is not easy to achieve self-driving capability under real world conditions and requires a lot of perfection to create a car which can make decisions on its own. For acquisition of data from its surrounding, StalwartX is embedding the cars with eight cameras and one front facing radar. The car is equipped with a super computer, which uses a self-learning process (deep learning) so that it can tackle any situation. Self-driving serviceability varies with jurisdiction and relies heavily on comprehensive software validation and regulatory approval. These driverless vehicles are capable of controlling themselves with no or very less human interference. The security of the car heavily depends upon the apps used to control this device. The car owners login to the Swt app (developed by StalwartX) and control the car. Autopilots are always connected to internet that makes them more vulnerable to cyber threats. The attack sometimes requires the car to be connected to a malicious Wi-Fi hotspot set up by the hacking team and then they can interfere with the controlling of the car. These attacks are capable of controlling the display and judgmental capabilities of the car like braking abruptly, slamming of the door while it is moving. Seeing the rapid growth of automakers, having higher standards of security become very crucial. Customers are keen to know the effectiveness of these autopilot software and how much they can rely upon them.

Questions:

- 1) *What test cases will you apply in order to ensure functionality parameters are up to the mark? Enlist them in the form of user stories.*
- 2) *Check whether the "in-built" system automatically detects anything abnormal. For instance if the system is hacked and steering suddenly turning left and right. Describe various test scenarios.*
- 3) *What if somebody hacks the Swt app and gets the credentials of the car owner? He may track the car, enables keyless functionality, and steals the car itself. What proactive measures will you check in order to ensure cyber security?*
- 4) *How does the car detect whether radar or cameras are malfunctioning or not? What measures the car would take once it gets to know that there is some malfunctioning or weird behavior? Because functionality of the car heavily depends upon the images the cameras and radar captures.*
- 5) *What testing strategy (e.g., unit testing) is more suitable for the problem considered?*