



Metodika testovania

SecIoT

Tím 06

Fellowship of the 6th

Testovanie nášho produktu by nemalo zabrať viac než vývoj produktu. Avšak mali by sme testami pokryť čo najväčšiu časť nášho kódu. Testovanie nášho produktu bude prebiehať v rozličných druhoch testov, keďže jednotlivé časti nášho produktu si žiadajú rôzne druhy testovania.

Testy ktoré použijeme v našom projekte budú: Jednotkové, Zátťažové, Akceptačné a Integračné, z časti možno aj Penetračné.

Jednotkové testy:

Tieto testy budú pokrývať asi najväčšiu časť projektu. Developeri v technickej dokumentácii spíšu, aké vstupy by mali do danej funkčnej časti ísť a čo sa očakáva na výstupe. Následne sa podľa toho vytvoria na danú časť jednotkové testy, v nástrojoch na túto časť vhodnými.

Zátťažové testy:

Budú sa vykonávať hlavne v komunikácií medzi IoT, a celkovo v problematike IoT. Koľko sú schopné zariadenia uniesť.

Akceptačné testy:

Akceptačné testy by mali pokrývať hlavne všetky možnosti vstupov – najmä tie hraničné. A kontrolovať čo sa s produktom deje ak nastanú rozličné situácie.

Integračné testy:

Tieto testy sa budú vykonávať hlavne v daných workflow-och, ktoré sa stanovujú a tento test bude sledovať správanie produktu postupne celým biznis procesom.

Penetračné testy:

Penetračné testy nebudú musieť byť realizované až na takej úrovni ako ostatné testy, najmä hlavne kvôli použitému framework-om, ktoré niektoré validácie a opatrenia v sebe už ukrývajú. Nie je však na škodu sa pozrieť na produkt aj z tejto jeho stránky.

Testovanie bude prebiehať v nasledujúcich krokoch:

1. Preštudovanie dokumentácie k danej časti a prípadné nejasnosti vyjasniť s daným developerom
2. Určiť nástroj na testovanie, stratégiu a o aký typ testovania ide
3. Vytvoriť viaceré testovacie prípady
4. V prípade potreby stiahnuť aktuálnu verziu kódu
5. Spustiť a vykonať testy
6. Overiť, či sa všetko správalo ako bolo na začiatku definované
7. Vytvoriť krátku správu o uskutočnení testu, ktorá bude obsahovať o aký test išiel, aké vstupné parametre sa zadávali a čo sme dostali na výstupe, prípadne z toho vyvodíť záver
8. Ak sa v teste nájde chyba, treba v Azure DevOps vytvoriť v aktuálnom šprinte task => bug, ktorý opisuje danú chybu, s možnosťou pri akých vstupoch chyba nastala či priložiť fotodokumentáciu a následne bug priradiť osobe, ktorá testovanú časť programovala