**Beet Seed 1**

Requirement tested: Only the key that comes with the front door lock can unlock it

Test plan:

* check with my key - that came with the lock
* check with my next door neighbor’s key - that has a different type of lock
* check with my downstairs neighbor’s key - that has the same type of lock
* check with my boyfriend’s key - another key that came with the lock

Results:

Only my key and my boyfriend’s key (both came with the lock when it was purchased) were able to unlock our front door. The key from the neighbor with a different lock couldn’t enter the lock and could not unlock the front door.

The key from the neighbor with the same type of lock was able to enter the lock but could not unlock it.

Why this test was chosen:

This test allows to check different possibilities: different keys that should be able to unlock the lock, a key from the same type of lock that was not supposed to unlock the lock and a key that didn’t even belong to that type of lock.

**Beet Seed 2**

Verification checks if the test object requirements are complied with; it only checks if the test object meets the requirements, without considering if the requirements set are aligned with what the end user expects and wants from the object.

Validation assesses if the requirements established are in accordance with what the end user needs and wants; it takes into account the suitability of the requirement.

**Beet Sprout 2**

| **Type of Company** | **Pros** | **Cons** |
| --- | --- | --- |
| Product | * Focus on only one product or one type of product allows to gain more expertise in the product/product type * More experienced with the product requirement allows to better assess their suitability to the end user’s needs | * Possibility of restrained views that can compromise the planning of different types of testing to detect other defects |
| Startup | * Fresh eyes to assess requirements suitability and to design test plans * Eagerness to find the most bugs/defects early on to avoid major monetary losses | * Lack of experience can lead to poor test planning * Need to do a more in depth research about the test object to assess how to test the requirements and to evaluate if the requirements comply with the end user’s needs |
| Outsourcing | * Cost reduction * Ability to “use” someone with experience and knowledge | * Work quality is the subcontracted company’ responsibility - less control over the results |
| Outstaffing | * Cost reduction * Ability to have someone with experience and knowledge that is not interfering with the company’s work | * The outstaffed person is only working for the company temporarily - loss of experience and knowledge after the project is over |
| Academy | * Ability to learn by working with experienced colleagues * “Fresh eyes” that can improve the assessment of requirements suitability and design of test plans | * Lack of experience can lead to poor test planning * Mentoring needed - time consuming |
| Recruitment Agency | * Allows to save time searching for the right person for the job | * Recruiter might not know how to properly assess if the candidate has the appropriate skills for the job |

**Beet Sprout 3**

I’ve had a smart band that had a feature that allowed the monitoring of several aspects such as activity time, calories spent, steps taken, and other for different types of activity. When I chose the “Walking” activity there was a feature that allowed the band to pause the monitoring automatically when it detected that the user stopped walking.

The validation of this requirement, in my opinion, was done correctly, since it is a requirement that helps the user. The verification of this requirement was not properly carried out, because when I simply slowed down a bit the band would automatically pause the monitoring, even if I didn’t fully stop walking.

**Mighty Beet 2**

1. Testing shows the presence of defects, not their absence

This principle is important in testing because it helps the tester and the client realize that just because no defects were found during the tests performed it doesn’t mean that no defect is going to be found in the future, including after product launch. Also, even if a defect is never found (during testing or after) that doesn’t mean that the product has quality. It just indicates that the product complies with all the requirements set without evaluating if the requirements are in line with what the end user wants and expects from the product.

1. Beware of the pesticide paradox

This principle shows that if the tester keeps using the same data and testing methods, there will come a point when it’s not going to find any more bugs or defects, because all the bugs/defects that were possible to find using this data and testing methods were already found. That doesn’t mean that there are no more bugs/defects to be found, but this data and methods cannot find anything else. There needs to be a change in the input used and/or the methodology of the testing to allow for other bugs/defects to be found.