**Test Plan - Property Visit Form**

**Scope – Analyze the Product**

The scope of this project is to validate the correctness of an implemented product, a Property Visit form, of a company that creates websites for real estate agents to showcase their listings. In that context, every end user of the website who is interested in the presented property listing can contact the real estate agent and inform them about their desire to visit the property.

The functional specifications, as given by the Product Owner, are based on the successful submission of the form that meets the acceptance criteria.

So, the examined form is located in this URL: <http://375.synergatos.gr/en/propertyDetails/13079> and the successful submission include:

* The required First Name field which contains only letters
* The required Last Name field which contains only letters
* The required Telephone Number field which contains only digits, spaces, and symbols: "+", "-", and "/".
* The optional Email field which valid format must be [xxxx@xxxx.xxx](mailto:xxxx@xxxx.xxx)
* The optional Message field
* The required Accept Terms checkbox

**Test Approach**

We define the test prioritization according to the acceptance criteria set by the Product Owner.

* So we begin to test the functionality of the form regarding that all the requirements are fulfilled. If the outcome is positive we try to detect the weaknesses, missing to fill in a value or filling in a value that does not comply with the specifications, for each one of the required fields separately. For example we can include numbers or symbols in the First Name field when all the other values of the fields are complying with the specifications and test if the form is submitted successfully. If the field we want to examine is optional, we can leave it blank or put random text (numbers, symbols, and letters).
* In the case that the outcome is not positive when all the requirements are fulfilled, we can also examine with the same process as previously if the form can be submitted changing the value for each field separately.

Only for this case we can include a limited number of tests, which will change the values of two or more fields when all the others comply with the specifications set by the Product Owner.

**Test Environment**

The test environment which is suitable for our occasion is the Integration Testing Environment.

In this type of environment, we integrate the individual software modules and then verify the behavior of the integrated system. A set of integration tests are used to check that the system behaves as specified in the requirements document. In an integration testing environment, we can integrate one or more modules of our application and verify the functional correctness.

**Testing Tools**

The test management and automation tool to be used for the test execution is Open source Eclipse IDE for Enterprise Java and Web Developers (V. 4.19.0). The Test Results Report and the Extent Report are to be created by this tool.

**Estimation**

The results of the tests will be appeared on the Console Screen of the IDE and will also be created one Extent Report .html and one TestNG .html file.

In the Console Screen the system will return a message for each test if the form was successfully submitted (“Testx Successfully Submitted the Form”) or not (“Testx Can’t Submit the Form”).

However the happy testing path (**PASS)** will be the RIGHT SUBMITTED (submit when all requirements are fulfilled) or the RIGHT NOT\_SUBMITTED (can’t submit when not all requirements are fulfilled).

The unhappy testing path (**FAIL**) will be the WRONG SUBMITTED (submit when not all requirements are fulfilled) or the WRONG NOT\_SUBMIITTED (can’t submit when all requirements are fulfilled).

The happy (**PASS**) or unhappy (**FAIL**) testing path will be available to be examined through the Extent Report.