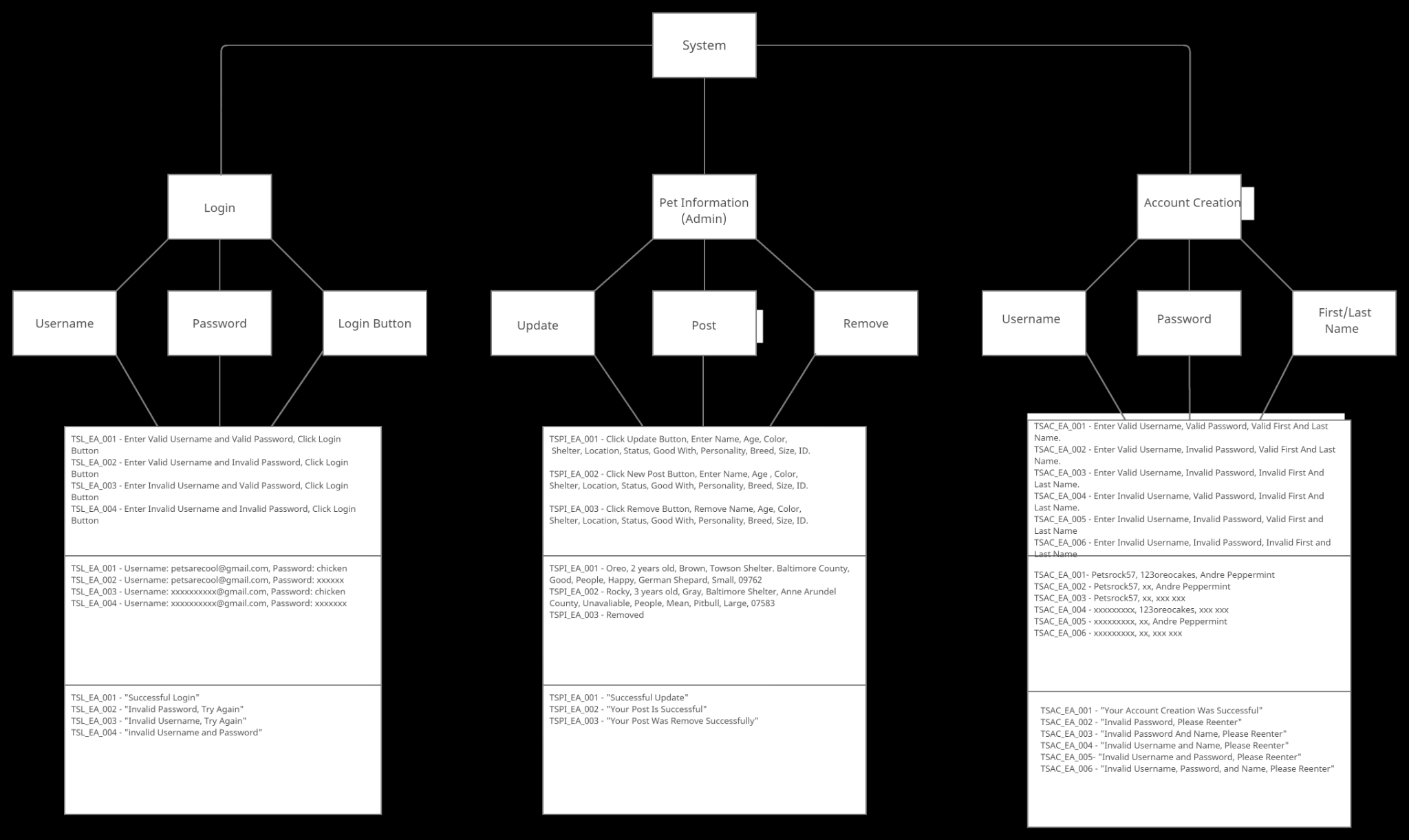
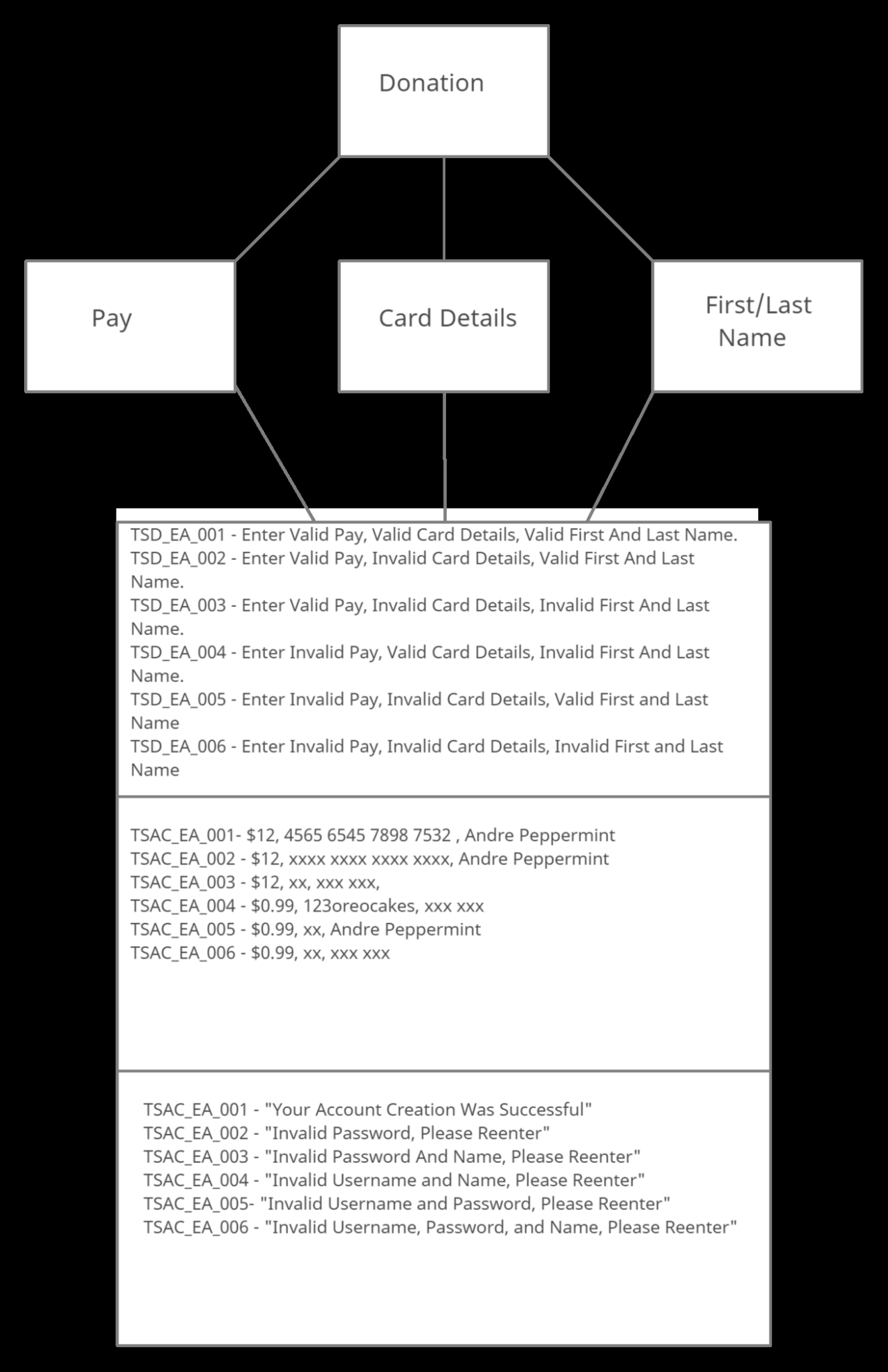
**Testing Documentation**

**Table of Contents**

|  |  |
| --- | --- |
| **Topic:** | **Page Number:** |
| **Skeleton of Integration Tests** | **0** |
| **End-to-End System Test Procedure** | **3** |
| **Regression Test Procedure** | **5** |

**Skeleton Integration Tests: (also in png form in repo docs)**

****

**End-to-End System Test Procedure:**

The End-to-End System Test Procedure we will be conducting on our project follows a seven point checklist to ensure the identification of all system dependencies and to guarantee that data integrity is withheld between all system components. Our End-to-End testing procedure is centered around the fact that we are making an application utilizing wordpress. Because we are using wordpress, tests will be conducted on the functionalities of plugins and whether or not they function, return the correct information, and if they abide by the requirements documents.The seven point checklist is as follows:

1. **Analyze Requirements:**

* Prior to employing the tests themselves we return to our requirements documentation to ensure that we have implemented all necessary functionalities and features.
* After assessing the completeness of our functionalities we will break down the functionalities in terms of their base necessities
* The base necessities are the key components that the various plugins must return or display in order to operate as intended.

1. **Create Test Environment:**

* Once the key components are analyzed and specified we will create a test environment whereby we employ a plugin as a standalone along with any other plugins it may depend on to display or function.
* The test environment is key to isolating plugins so that we may test and ensure that each individual plugin added into our wordpress is functioning as intended and in alignment with all the requirements.

1. **Software and Hardware Requirements:**

* In this phase we must answer the following questions:
* What software is necessary for this plugin to operate properly?
* What hardware is necessary for this plugin to operate properly?
* Are there any additional requirements?

1. **System Response List:**

* In this phase we must compile a list of all scenarios that any user may engage with the plugin and record all variations of how the system should respond to the specified interaction.

1. **Testing Methods List:**

* Once we have listed hw the system should respond we will list the various methods of how we intend to test the application.
* This will include clear descriptions of standards used in the methods, for example: different tools, languages etc.

1. **Test Case Design:**

* After drafting the methodology of the test and defining all ways the system should respond we will create a step-by-step testing case design to mock the user of whichever plugin is being tested.

1. **Test Case Data Storage and History:**

* In this phase the case design has been employed and all tests that are ran will be recorded
* Results will be saved and described as to eliminate the possibility of redundant testing.
* In the event a plugin is not operating properly a re-test will be conducted and the procedure is restarted.

**Regression Test Procedure:**

The Regression Test Procedure we will be conducting on our project follows a four step checklist to ensure any newly implemented plugins don’t interfere with previously existing plugins. The Regression Test Procedure we will be following for this project is based on the prioritization model. Our Regression Test Procedure is centered around the fact that we are making an application utilizing wordpress. Because we are using wordpress, tests will be conducted on the functionalities of plugins and whether or not they operate as intended, return the correct information, and if they abide by the requirements documents.The four step procedure is to be followed after any new plugin is added. The procedure is as follows:

1. **New Plugin Assessment:**

* Now that a new plugin has been added to the wordpress we will assess the plugin by answering the following questions:
* What interdependencies does the new plugin create?
* What types of data does the new plugin utilize?
* How could the new plugin alter how the application operates?
* Lists of the answers to the aforementioned questions will be compiled and reviewed to determine what other plugins are to be retested after the installation of the new plugin.

1. **Plugin Selection:**

* Based on the information recorded in step one we will select only plugins that are the most impactful and critically affected by the **new plugin.**
* Selecting only the critical and most impacted plugins prevents wasted time and money reviewing code that may be unaffected

1. **Plugin Testing and Review:**

* Now that the plugins have been selected the testing phase may begin
* Testing will ensure that the newly installed plugin is not interfering with previously installed plugins.
* In the event that a bug is found, there will be a group discussion and a decision will be made to either replace pre-existing plugins to fix the bug or to scratch the new plugin and find an alternative.

1. **Data Storage and History:**

* Concluding the testing and review process all tests that occurred will be time stamped and recorded for referral in future tests,
* Any and all plausible interdependencies found in testing will be recorded and noted for future reference to increase testing efficiency throughout the construction of the application.

Overall, our regression testing is employed to ensure changes made to the wordpress don’t unintentionally fix one component and break another.