Receipt Scanner Testing Plan

1. Black box testing

The anticipated functionality of this component is an ability to collect a receipt file either from the gallery, or from a photo from the camera and then extract data from the receipt. Our target is to accurately extract a product name/description, and its quantity for the preliminary version.

Planned testing

Input

i). Receipt image quality based on size - assert a standard minimum (about 500KB ) for clear image to ensure meaningful data extraction.

ii). Image file type - strip the image uri for image type and assert all the veryfi approved image types

iii). Multiple image type and quality availability - attempt to parse in different types images of different quality sizes to check if each time we successfully extract a json file. Using a unit test we’ll set a list with various images and parse them into the module iteratively.

Output

i). Json file check for output - assert an output of type json file from the whole module this will intrinsically test the output of the veryfi API call.

1. White box testing

After the initial specifications, this module has morphed into two main classes ReceiptFileCollector and the ReceiptDataExtractor. My goal is that the two classes do exactly what is needed and that the methods will return anticipated values.

Planned testing

ReceiptFileCollector

i). Run different cases for selecting an image from the gallery or from the camera to ascertain the right constructor path. I’ll run a unit test setting the UIListener differently each time and assert that the correct path was taken.

ii). Correctness test for pickFromGallery() constructor - assert the receiptFileData object has image URI and an image type property in a select from gallery situation.

iii). Correctness test for pickFromCamera - assert the receiptFileData object has image URI and an image type property in a scan with Camera situation.

ReceiptDataExtractor

i). Run a unit test on the client call to ascertain the status code (201) for the veryfi API endpoint. (I am still researching more ways to test the client API call, but I am hoping to do a better time out test as well)

ii). Correctness test for the processReceipt routine - assert the properties item/name/description and quantity.