Code Test Explanation

I have produced a small app which meets the requirements of all the stories, except partially the encryption story.

The Corona simulator was set up to use an iPhone X, and the app is arranged to be optimal on this layout.

I designed the application to work on a single screen made up of several elements:

* A top bar to display the app logo,
* A text field to display the status of the transaction,
* A progress bar used during the bank-acquisition phase of the transaction,
* A text field in which to display the transaction amount,
* A section on which to display the separate card digits entered by the user,
* And a numeric keypad with cancel, clear, and enter buttons, as seen on most card machines.

The states the application moves through are:

* Prompt user to enter card details,
* If user presses enter, check for valid number of digits entered (must be eight, not possible to enter more as the application does not allow the number to grow past eight digits),
* Ask for a valid number if less than eight digits entered, otherwise hash the number to provide some level of security in the absence of encryption, prompt to return terminal to merchant (it is the user’s phone in this case, but that is not important) and lock user input so that the number cannot be altered,
* Animate a sequence depicting the app contacting the bank for approval, animate the progress bar in order to display the status to the merchant,
* Confirm that the transaction is approved to the merchant.

If the user presses clear at any time before pressing enter, the numbers entered will be wiped and they can start again, also clear resets the application after a successful transaction.

If the user presses cancel, the numbers entered will be wiped and they may not enter any more until the approval stage is finished and the merchant presses clear.

I struggled to load a suitable library to obtain an encryption function. The Lua version on my system is 5.4 and most, if not all encryption libraries require an older version such as 5.1. I spent some time trying to install and switch the interpreter to 5.1, but using windows there is currently no way to install luarocks to manage libraries/versions. I switched to a mac to remedy this, but it was still far from trivial to switch versions on. The Solar2D documentation referred to the Crypto library, so I used the digest function to hash the card details so that the signature would at least be invalidated if the data was compromised.

I produced a small unit test to validate the length of the card details and assert if the length was not 8, this functionality is disabled by default using a flag.

Please see the pictures folder for a demonstration of the app in action.