Command:

**build:**

xcodebuild build-for-testing -workspace "/Users/rubhatt/Documents/PayPalApp/consumer-venice-ios/consumervenice.xcworkspace" -scheme "consumerveniceOnboardingUITestsAll" -destination "platform=iOS Simulator,name=iPhone 6,OS=11.0" -derivedDataPath "/Users/rubhatt/Documents/PayPalApp/consumer-venice-ios/build"

**test:**

xcodebuild test-without-building -xctestrun "build/Build/Products/consumerveniceOnboardingUITestsAll\_iphonesimulator11.0-x86\_64.xctestrun" -destination "platform=iOS Simulator,name=iPhone 6,OS=11.0" '-only-testing:consumerveniceOnboardingUITests/consumerveniceOnboardingUITests\_us' -derivedDataPath 'build/reports/Us' | tee xcodebuild.log | xcpretty --report html --screenshot

&

xcodebuild test-without-building -xctestrun "build/Build/Products/consumerveniceOnboardingUITestsAll\_iphonesimulator11.0-x86\_64.xctestrun" -destination "platform=iOS Simulator,name=iPhone 7,OS=11.0" '-only-testing:consumerveniceOnboardingUITests1/consumerveniceOnboardingUITests1\_uk' -derivedDataPath 'build/reports/Uk' | tee xcodebuild1.log | xcpretty --report html --screenshot &

xcodebuild test-without-building -xctestrun "build/Build/Products/consumerveniceOnboardingUITestsAll\_iphonesimulator11.0-x86\_64.xctestrun" -destination "platform=iOS Simulator,name=iPhone 7 Plus,OS=11.0" '-only-testing:consumerveniceOnboardingUITests2/consumerveniceOnboardingUITests2\_pt' -derivedDataPath 'build/reports/Pt' | tee xcodebuild2.log | xcpretty --report html --screenshot

tee and xcpretty can be used for proper error reporting and screenshots. You can remove that pipe if you just want the standard console report generated by xcodebuild.

All the simulators should be triggered beforehand in order to successfully execute the command.