

From
2015



Thought Frameworks Case Study: Mobile Automation & Functional Testing

Summary of the Product:

CellTrust is a leading provider of traceability and security for mobile communication for highly regulated industries and mobile aggregation across 200+ countries and over 800 carriers and mobile operators. CellTrust Secureline offers archiving and protection for mobile communication content supporting enterprise mobile collaboration, eDiscovery and major global regulatory compliance for financial services, government and healthcare. CellTrust's multi-channel SMS gateway processes half a billion messages annually and hosts some of the most prestigious and high profile mobile brand campaigns.

Technology:

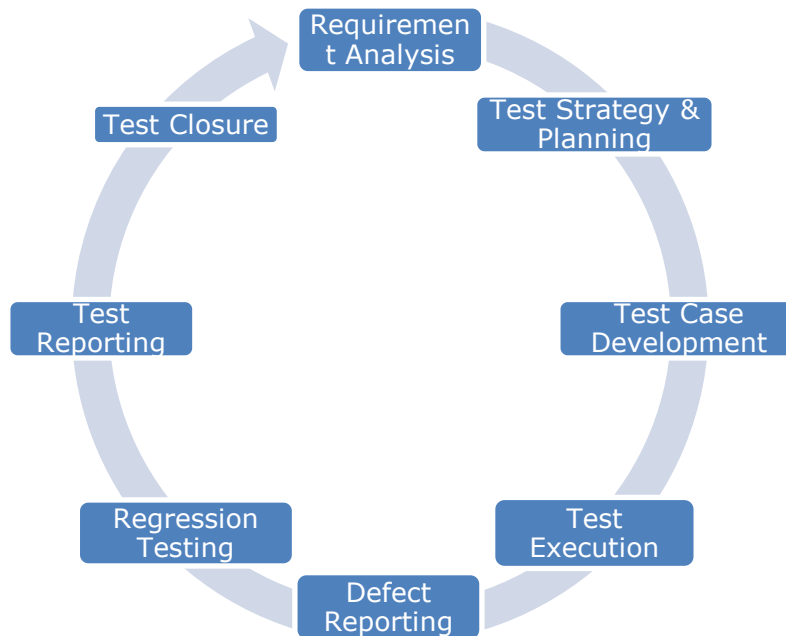
Secure line Android OS compatible app was developed using Android Studio IDE, with Android SDK Version 23 and SQL Lite as a Backend Database.

Business challenges:

CellTrust Secureline Native app was to provide a secured communication (SMS & Voice) for the users using the Secureline application. With People Using Various Mobile devices with Different Mobile operating systems (Android, iOS, Windows) and Service providers, the app was to be developed keeping the OS compatibility and security needs.

Technical challenges:

- Ambiguous Requirements
- Quality of Existing test cases was not up to the standard
Lack of dedicated Subject Matter Expert for the required knowledge transfer
- Testing team had to understand major part of the application by executing the existing test cases.
- Test Coverage focused towards :
 - Functional Testing (Android ,IOS & Web Application)
 - Device Testing – Samsung & Apple iPhones
 - Compatibility Testing(Mobile Browsers & Web browsers)
- Test Phases
 - Execution phase – Android ver, IOS Ver, Web application



➤ **Progress and Coverage:**

- All new feature test case Design, execution and reporting
- All test case status updates via test management tool
- All Defects raised and assigned to developer using a bug tracking tool
- Daily QA status report summarizing the test execution summary, list of bugs raised and their status.
- Cross browser testing of the application in Web(IE, Chrome & FF) Android Mobile(Chrome, FF) iOS Mobile(Safari, Chrome)
- OS Compatibility testing on web(Windows, Mac), on Mobile Devices(iOS, Android)
- Tests done in QA test Env & Production Env.
- Automating the designed web application test cases

➤ **Defect Reporting:**

- Log defects into JIRA - Defect Management Tool.
- Defects were triaged by the triage team and then passed on to the developer for fixing
- All Blocker defects to be communicated to the QA manager as and when the investigation was complete

➤ **Project Challenges Faced:**

- Lack of requirement documents for planned feature
- Lack of access to Project tools
- Several mobile devices had to be procured for testing
- Automation testing tool had to be finalized
- Sprint release information was unavailable(Bug fixes, New Features introduced)
- Frequent changes to device details

➤ **Highlights:**

- 2500 plus feature test cases designed during the project tenure
- 500 plus automation scripts designed using Sahipro for web & 900+ automation scripts designed using Appium/Selenium for mobile.
- 1000 plus defects identified, fixed and retested.
- Cumulative of 5000 plus regression test cases executed so far.

➤ **Accomplishments:**

- Designed high quality test cases for the Secure line app from scratch
- Tested the Native app using all new Android & iOS devices released in the market
- Agile QA Process was put in place and streamlined the release and testing Process.