Agenda

Overview
Objectives of the Research Paper
Challenges: Mobile Application Testing
Mobile Automation Testing
Types of Data Centric Applications
Financial Applications for Mobile Devices
Fiserv’s Mobile Applications Testing Approach
Benefits of Implementing Automation Testing
Conclusion
References
Overview

The Mobile Application Boom: Trend and Performance Statistics

- International Data Corporation (IDC) predicts that 182.7 billion mobile apps will be downloaded by 2015.
  - That is a 1600% increase from the 10.7 billion apps downloaded in 2010.
Overview

In Fact…

- Based on trends and forecast, mobile phones have shifted the focus from computers to mobile phones.
- This shift has exponentially increased the development of mobile applications.
- Unfortunately, many of the applications are facing failure post market launch due to insufficient testing methodologies used for mobile application testing.
- Inappropriate testing of mobile applications is majorly due to precise time schedules available; as there are frequent updates received in this field.
- Testing of mobile applications is traditionally done by manual execution of test cases and visual verification of the results.
- In order to meet such challenges, mobile application testing must evolve as separate stream of testing where automating the appropriate areas of a testing program can yield quantifiable benefits.
Objectives of The Research Paper

Automation testing for mobile applications is a boundless subject, this research paper would emphasize on the major key areas and outline information regarding

- Challenges experienced while testing mobile applications
- Measures to overcome those challenges via automation testing and its selection criteria.
- Mobility requirement of a financial applications
- Comparative analysis on various mobile automation tools to procure optimum tool for testing
- Methods to perform automation on mobile apps
- Major benefits drawn out of automation testing
Challenges: Mobile Application Testing

Generic Challenges
- Device/platforms diversity
- Hardware complexity
- Application complexity
- Compressed development schedules

Technical Challenges
- Weaker CPU and small memory
- Varied carriers
- API Level test execution
- Verification of UI control
- Handset specific features
Mobile Automation Testing

Selection Criteria for Mobile Automation Testing

<table>
<thead>
<tr>
<th>Types of Mobile Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Centric Apps</td>
</tr>
<tr>
<td>These mobile applications can be automated. Example of such applications are BFS, CRM, and enterprise applications.</td>
</tr>
<tr>
<td>Device Capability Apps</td>
</tr>
<tr>
<td>These mobile applications cannot be automated. Example of such applications are LBS, Camera, Multimedia, VOIP.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Types of Data Centric Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Native Apps</td>
</tr>
<tr>
<td>➢ Installed and launched on mobile devices.</td>
</tr>
<tr>
<td>➢ Testing of such applications on some mobile devices requires access to a device ID.</td>
</tr>
<tr>
<td>➢ Functionality and Usability of native apps need to be tested on multiple devices.</td>
</tr>
<tr>
<td>Mobile Web</td>
</tr>
<tr>
<td>➢ Do not require installation.</td>
</tr>
<tr>
<td>➢ They are required to be tested on varied mobile browsers</td>
</tr>
<tr>
<td>Hybrid Apps</td>
</tr>
<tr>
<td>Combination of a native app and mobile web, where the icon and interface of the native app is merged with the ease and the rich content of the mobile web.</td>
</tr>
<tr>
<td>Wrapper Webs</td>
</tr>
<tr>
<td>These are websites, packaged as apps. ‘Wrapper’ apps are shells that are downloaded and installed on the device. The shell then accesses mobile formatted web content from a traditional web server</td>
</tr>
</tbody>
</table>

© 2009 Fiserv, Inc. or its affiliates.
Types of Data Centric Applications

Retail

Media & Entertainment

Banking & Finance

Health Care

Games

Travel
# Financial Applications for Mobile Devices

<table>
<thead>
<tr>
<th>Financial Applications Requirement</th>
<th>Financial Application Challenges</th>
</tr>
</thead>
</table>
| From retail banks and insurance brokers to asset management companies and investment banks, financial services firms are looking to mobile to increase internal efficiencies, deliver higher levels of customer satisfaction, build awareness and customer adoption, and increase sales and revenue metrics | ➢ Privacy and Security concerns  
➢ Addressing multiple Platforms and networks to carry finance applications.  
➢ Creating an intuitive and flexible interface based on robust and efficient information architecture |

## Benefits drawn from Financial mobile Applications

- Added Customer convenience to avail banking services.  
- Controlled costs due to less reliance on costly resources to run physical locations.  
- 'Always on, always available' aspect of the mobile channel provides productive benefits to the clients.
# Fiserv’s Mobile Applications Testing Approach

## Business Case
To build a mobile application that generates an extremely large number of customer transactions each day for multiple currencies and displays various graphs and charts with respect to transactions on a periodic basis.

## Business Challenges
- Considering Customer requirements and Market trends; determination of platform.
- Nature of the application
- Device/platforms diversity
- Simultaneous launch for two Platforms
- Cost effective device compatibility testing with a few tools available.
- Ensure quality of service considering the frequent updates
- Minimal CPU Utilization
- Compatibility with various OS versions

## Implementation Approach
- Determination of the Platform, Devices and Nature of the Application
- Intensive Research on Various Mobile Automation Tools
- Creation of the Cross Platform Test Scripts
- Preparation and Execution of Smoke, Sanity and Regression Test Suites.
- Validation of CPU utilization.
- Testing on various OS versions.
Fiserv’s Mobile Applications Testing Approach

Implementation Approach

Step-by-step implementation of the below mentioned various attributes of design approach, resulted in a successful launch of the mobile application.

I. Determination of the Platform, Devices and Nature of the Application

- **Android and iOS platforms** were selected considering the customers’ requirements, their location, and the market trends.
- Mobile devices with **size greater than 3.5 inches** were recommended for the perfect UI view of the application.
- Nature of application was decided as ‘**Native application**, as ‘Native Mobile Apps Offer Advantages over Wrapper Apps for Financial Services’

**How Native Mobile Apps Offer Advantages over Wrapper Apps for Financial Services?**

- Faster and readily available, which is critically important for financial services.
- ‘Re-download’ not required every time it is opened.
- Responsive and powerful in terms of the capabilities it can support.
- Deposit of a check via a camera-equipped smart phone can be enabled.
- Uses advantage of ‘Core code’ that is in the operating system itself.
### Fiserv’s Mobile Applications Testing Approach

#### Implementation Approach

**II. Intensive Research on Important Features of Various Mobile Automation Tools**

A Comparative Analysis on the Important Features of the Tools was performed on six mobile automation tools considering the following important features.

<table>
<thead>
<tr>
<th>Important Features of the Tool</th>
<th>See Test (QTP Plug-in)</th>
<th>Monkey Talk</th>
<th>Roboium</th>
<th>Selenium</th>
<th>Meux tool(QTP Plugin)</th>
<th>EggPlant</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Support of Platforms</strong></td>
<td>iOS and Android</td>
<td>iOS and Android</td>
<td>Android</td>
<td>iOS and Android</td>
<td>All mobile platforms, different licence available for each platform</td>
<td>All mobile platforms</td>
</tr>
<tr>
<td><strong>Support for Native app</strong></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Scripts run on Mobile</strong></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Scripts run on Emulator</strong></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Record and play back</strong></td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td><strong>Same test running on different devices</strong></td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Open Source /Freeware</strong></td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>
### Fiserv’s Mobile Applications Testing Approach

#### Implementation Approach

<table>
<thead>
<tr>
<th>Important Features of the Tool</th>
<th>SeeTest (QTP Plugin)</th>
<th>Monkey Talk</th>
<th>Robotium</th>
<th>Selenium</th>
<th>Meux tool(QTP Plugin)</th>
<th>EggPlant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name of Tool</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Same test running on different mobile OS (cross Platform)</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Test management tool integration</td>
<td>Quality centre</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Quality centre</td>
<td>No</td>
</tr>
<tr>
<td>Scripting Language</td>
<td>Vbscript /java/perl</td>
<td>Monkey talk/Java</td>
<td>Java</td>
<td>Java/C#/Python/ruby</td>
<td>VbScript</td>
<td>Sense Talk</td>
</tr>
<tr>
<td>Code Rebuild is required or Not</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Initial setup efforts( 1-5, 1 is least complicated )</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Usability ( 1-5, 1 is the easiest )</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Support for Native ID recognition</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Support for Image recognition</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Support for Text recognition</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Security levels match with applications</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Fiserv’s Mobile Applications Testing Approach

Implementation Approach

III. Creation of the Cross Platform Test Scripts

Based on the comparative analysis, ‘Monkey Talk’ was the only tool, which supported ‘Cross Platform’ functionality and provided following benefits.

- Reusability of Scripts
- Single Unified IDE
- Reduced time for scripting
Fiserv’s Mobile Applications Testing Approach

Implementation Approach

IV Preparation and Execution of Test Suites.

Mobile Application, being under the testing phase, there were frequent builds, which had to be tested from the core. Thereby different test suites mentioned below were prepared executed several times.

- Smoke Test Suite
- Sanity Test Suite
- Regression Test Suite
Fiserv’s Mobile Applications Testing Approach

Implementation Approach

V. Validation of CPU utilization

Reports generated by running the Test suites provide the execution time of each test script. This helps in keeping account of the time taken for execution of the same functionality and thereby keeping a note of CPU utilization at different instances.

VI. Testing on various OS versions

• Applications were deployed on various OS versions of iOS and Android Emulators/simulators.
• Separate test suites to verify compatibility of application were executed.
Benefits of Implementing Automation Testing

Pursuance of the implementation approach helped the company to ‘Sail through the challenges’ and facilitate a successful launch of product.

- Reduction in resources due to implementation of Automation testing and enhanced quality of the product delivered.

- No investment in launching the automation tool and benefited Savings and ROI.

*Manual vs Automation efforts in hours*

*ROI calculation (Saving per year/Investment)*
Benefits of Implementing Automation Testing

- Reduction of creation and execution time of test scripts, as the same script executed for two different platforms.

- Benefited profit Scales due to selection of appropriate OS with reference to market statistics proved highly beneficial for profit scales.

- Early defect detection and reduced defect leakage from previous testing cycle by 40%. This in turn resulted into early downtime detection to allow remedial action to minimize loss of revenue.

- Reduction in CPU utilization by proper validation of CPU utilization which was tracked from reports generated by the tool which mentions time taken to execute the script.

- Improvement in CSI (customer satisfaction index) due to on-time delivery, reduction in time for execution of UAT and reduced number of post delivery defects.

- Improved user experience through optimizations in infrastructure by gaining early visibility into issues that impact end-user.

- Replication opportunities of other similar applications were achieved looking at the performance of current application.
Conclusion

- Mobile test automation should become imperative because testing mobile financial apps manually takes significantly more time than testing web or Windows applications as the data entry is much slower on mobile keyboards than on regular ones.

- The cost of error in mobile software is high, and verifying that the changes introduced to one of its components are not going to cause problems with the other parts of the application is essential. The time-to-market is ever more important, hence, the need to run the tests quickly is a must.

- Organizations intending to deploy mobile finance applications must plan their testing strategy across both the manual and automation testing approaches for efficient and error-free delivery.

- There should be optimum selection of mobile testing methodology along with the appropriate tool for automation testing of the mobile applications. The tool must be in compliance with the environment for which application is built.

- Last but not the least, there exist a great opportunity for entrepreneurs to bring new tools into market which may use similar automation scripts for all major platforms.
References

• uTest_eBook_Mobile_Testing : eBooks on mobile apps development and testing.
• http://www.sdtimes.com/SURVEY_MORE_THAN_HALF_OF_ORGANIZATIONS_ARE_BUILDING_MOBILE_APPLICATIONS/By_Alan_Zeichick/About_MOBILEDEVELOPMENT/36553
• http://www.mobileapptesting.com/tag/free-mobile-testing-tool/
• http://www.gorillalogic.com/testing-tools/monkeytalk
• http://harshalkharod.blog.com/2011/09/16/mobile-application-testing-guidelines/
• http://www.mobileapptesting.com/tag/mobile-automation-testing/
• http://blog.infostretch.com/meux-automation-tool-for-mobile-devices
• http://experitest.com/support-2/seetest-pricing-mobile/
• http://www.tekritissoftware.com/android-automation-testing-robotium
• http://seleniumtalks.blogspot.in/
Questions
Thank You