Park Entry Test Plan

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Rev** | **Date** | **Author** | **Description** |
| 3.0 | 7/10/2013 | Francisco Gonzalez | Added Section 15 (Release 1.7 – July 2013) |
| 2.0 | 3/19/2013 | Francisco Gonzalez | Added Section 14 (Release 1.6 – March 2013) |
| 1.0 | 1/16/2013 | Francisco Gonzalez | Release Version |

**Document Approvers & Sign-Off**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Approver** | **Role** | **Document Accept/Reject** |
| 5/21/2013 | Mark Mecham | Synapse Test Manager |  |

**Contacts**

|  |  |
| --- | --- |
| Glenn Curtis | Disney Relations Manager |
| Tony Cowan  Travis Pruitt | xGreeter Application Developer |
| Mike Broz  Imran Sheikh | Project Management |
| Arek Glabek | xBRC Developer |
| Slava Minyailov | xAG Developer |
| Iwona Glabek | xBRMS Developer |
| Greg Strange | Reader code |
| Corey Wharton | Reader code |
| Adam Parish | Disney |
| John McLean | QA |
| Michael Lampi | QA |
| Jason Olmstead | QA |
| Rob Silvernagel | QA |
| Francisco Gonzalez | QA |

**Related Documents**

|  |  |
| --- | --- |
| 900-0063 IDMS Test Plan | The ID Management System (IDMS) stores and maintains information about park guests, making it possible to link external systems with the xConnect components. |
| 900-0071 xBRC Test Plan | xBand Reader Controller |
| 900-0072 xBRMS Test Plan | xBand Reader Management System |
| Park Entry Test cases | And Excel document Including xGreeter and High Availability (HA) test cases. |
| Events and Audit – System Design Document | Describes an enterprise database of security and system health events for all xConnect components. |
| Events And Audit – Test Cases | An excel document with Events and Audit test cases to be executed. |

**Table of Contents**

[1 Background 6](#_Toc356988921)

[2 Objective 6](#_Toc356988922)

[3 UX Elements 7](#_Toc356988923)

[4 In Scope 7](#_Toc356988924)

[5 Schedule and Timeline 7](#_Toc356988925)

[6 Testing Resources 8](#_Toc356988926)

[7 Testing Environment 8](#_Toc356988927)

[8 Test Metrics 8](#_Toc356988928)

[9 Test Deliverables 9](#_Toc356988929)

[10 Test Scenarios 9](#_Toc356988930)

[11 Performance Testing 10](#_Toc356988931)

[12 High Availability 12](#_Toc356988932)

[12.1 Background 12](#_Toc356988933)

[12.2 Objective 12](#_Toc356988934)

[12.3 Network Architecture 12](#_Toc356988935)

[12.4 In Scope 12](#_Toc356988936)

[12.5 Schedule and Timeline 13](#_Toc356988937)

[12.6 Setup instructions on xBRMS 13](#_Toc356988938)

[12.7 Set up the xGreeter App on the device (iPOD) 14](#_Toc356988939)

[13 Test Scenarios 15](#_Toc356988940)

[13.1 Pre-requisites (applicable to all scenarios): 15](#_Toc356988941)

[13.2 Scenario 1 – Master unavailable when events are happening 16](#_Toc356988942)

[13.3 Scenario 2 - Master unavailable before logging in 17](#_Toc356988943)

[13.4 Scenario 3 – xBRMS unavailable 18](#_Toc356988944)

[13.5 Scenario 4 – Device (iPOD) cannot reach the Wi-fi 19](#_Toc356988945)

[13.6 Scenario 5 – a switch from Master to Slave happens when the device (iPOD) cannot reach the Wi-fi 20](#_Toc356988946)

[14 Release 1.6 (March 2013) 20](#_Toc356988947)

[15 Release 1.7 (June 2013) 21](#_Toc356988948)

[15.1 Scope 21](#_Toc356988949)

[15.2 Test Resources 23](#_Toc356988950)

[15.3 Test Deliverables 23](#_Toc356988951)

[15.4 Timeline 24](#_Toc356988952)

# Background

The xGreeter App will be used by Disney cast members who are employed as either “greeters” or “greeter coordinators”. The greeters will be responsible for overseeing guests entering Disney properties such as Magic Kingdom or Epcot. Guests will possess a device (key card) that contains an RFID tag, which they will tap on a reader to identify themselves. The reader will flash green or blue depending on whether the guest entitlement check succeeded or failed.

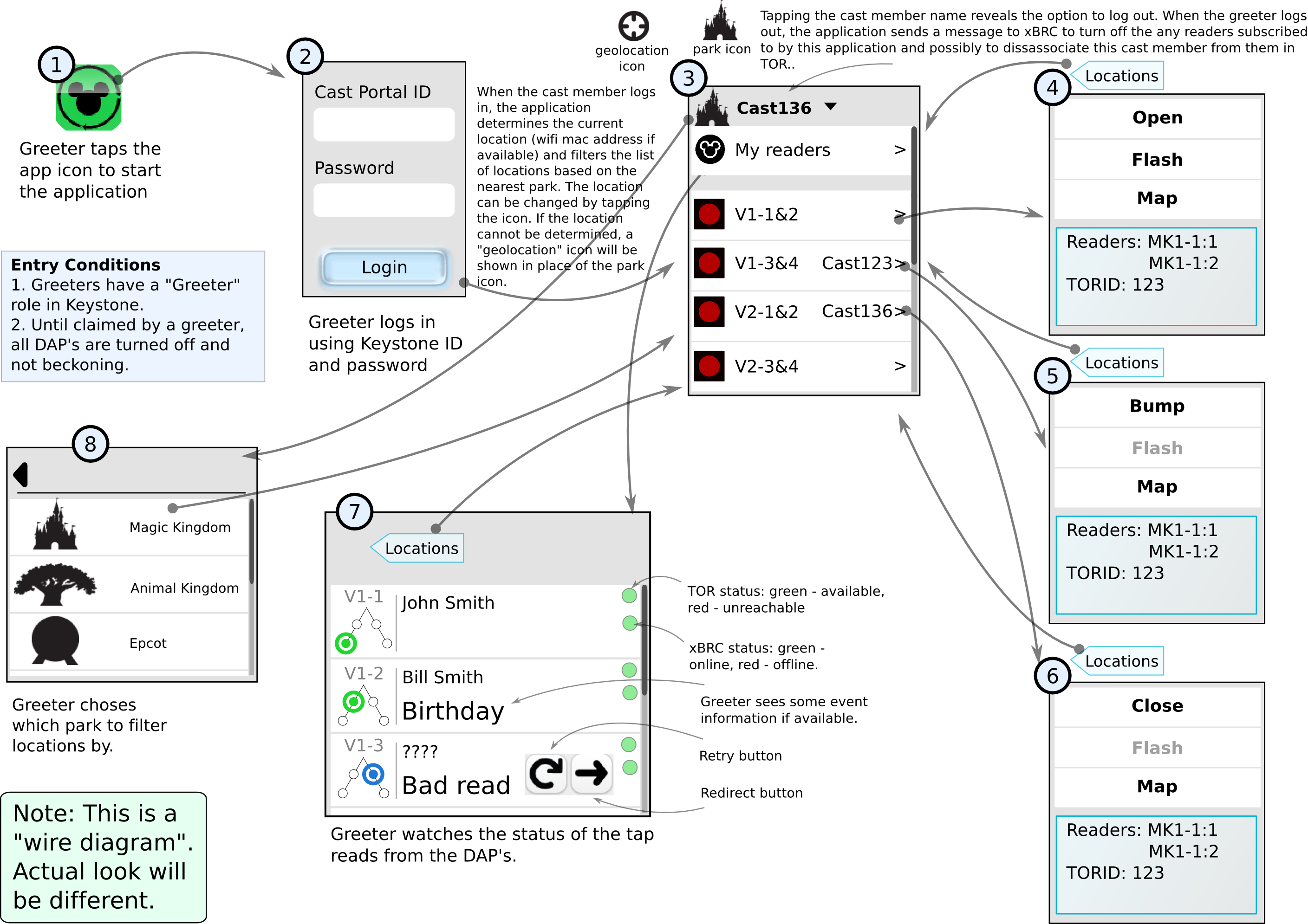
Readers are assembled into locations. Typically, a greeter will be responsible for monitoring a single location that will typically consist of two readers.

Typically, a guest coordinator will be responsible for overseeing up to eight readers at a given time. The readers a given coordinator will be responsible for will be installed contiguously.

# Objective

The objective of Park Entry integration testing is to guarantee all components involved are functioning; guests have the best experience when tapping their cards on the reader, Greeters are able to manage locations, readers, and events and Greeter Coordinators are able to manage readers and Greeters.

# UX Elements



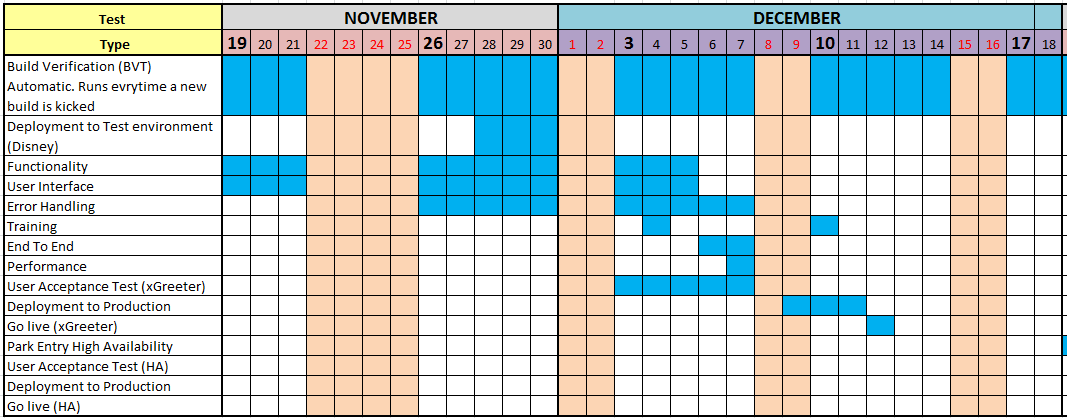
# In Scope

* xGreeter App
* As part of the Integration testing the following components will be tested along:
  + xTP – The device where guest tap the Fast pass card.
  + xBIO - the biometric (fingerprint) reader associated with the DAP reader.
  + xBRC – xBand Reader Controller (manages a set of readers.)
  + XBRMS – xBand Reader Management System (manages a set of xBRC’s.)
  + IDMS – ID Management System (handle the xBand metadata.)
  + Omni (The Park Entry authentication mechanism)

# Schedule and Timeline

**Tasks**

* Build Verification Test (BVT)
* Deployment to Test environment (Disney)
* Functionality
* User Interface
* Error Handling
* Training
* End To End
* Performance
* User Acceptance Test (xGreeter)
* Deployment to Production
* Go Live (xGreeter)



# Testing Resources

* Synapse
  + Rob Silvernagel
  + Francisco Gonzalez

# Testing Environment

* SIT in Seattle – A testing bench with virtual machines hosted in the Hyperchicken vmware server.
* Alpha Lab in Orlando (an environment with virtual machines managed by Adam Parish.)
  + xBRCs: nl-flfa-00160, nl-flfa-00161

# Test Metrics

* Defects found by priority
  + Critical – Blocking, Major impact, must fix.
  + High - Impact on business, functionality broken.
  + Medium – Low impact on guest or client.
  + Low – Mostly cosmetic, little or no impact on business.

# Test Deliverables

* Test Plan (this doc)
* Test Cases (and Excel document including xGreeter and High Availability test cases)

# Test Scenarios

* Scenario 1
  + Configure two xTPs to point to xBRC in test environment
  + Configure xBRC to point to xBRMS in test environment
  + Configure Omni (or simulator) in test environment
  + Point to Keystone with real like users with different profiles (Greeter and Greeter Coordinator among them)
  + Download latest Park xGreeter App to iPOD with iOS 6.0.1
  + Configure iPOD app to specific venue and xBRC configured above
  + Login as Greeter1
  + Open location with readers
  + Have 10 guests (fast pass cards) to tap on reader1 causing green event
  + Have 1 guest (fast pass card) to tap on reader2 and cause blue event
  + Greeter1 clears blue event
  + Greeter1 logs out
* Scenario 2
  + Configure two xTPs to point to xBRC in test environment
  + Configure xBRC to point to xBRMS in test environment
  + Configure Omni (not simulator) in test environment
  + Point to Keystone with real like users with different profiles (Greeter and Greeter Coordinator among them)
  + Download latest Park xGreeter app to iPOD with iOS 6.0.1
  + Configure iPOD app to specific venue and xBRC configured above
  + Login as Greeter1
  + Open location with readers
  + Login as Greeter2
  + Greeter2 bumps Greeter1
  + Greeter1 logs out.
* Scenario 3
  + Configure three xTPs to point to xBRC in test environment
  + Configure xBRC to point to xBRMS in test environment
  + Configure Omni (not simulator) in test environment
  + Point to Keystone with real like users with different profiles (Greeter and Greeter Coordinator among them)
  + Download latest Park xGreeter App to iPOD with iOS 6.0.1
  + Configure iPOD app to specific venue and xBRC configured above
  + Login as Greeter Coordinator
  + Greeter Coordinator Schedules Maintenance for reader1
  + Greeter Coordinator Schedules Bio maintenance for reader2
  + Greeter Coordinator Shuts Down reader3
  + Greeter Coordinator logs out.
* Scenario 4
  + Configure two xTPs to point to xBRC in test environment
  + Configure xBRC to point to xBRMS in test environment
  + Configure Omni (not simulator) in test environment
  + Point to Keystone with real like users with different profiles (Greeter and Greeter Coordinator among them)
  + Download latest Park xGreeter App to iPOD with iOS 6.0.1
  + Configure iPOD app to specific venue and xBRC configured above
  + Login as Greeter Coordinator
  + Greeter Coordinator switches to Greeter role
  + Open location with readers
  + Guest causes a blue event
  + Clear blue event
  + Switches to Greeter Coordinator role
  + Log out
* Scenario 5
  + Attempt to login to Park xGreeter App with not existing user in Keystone
  + Attempt to login to Park xGreeter App with existing user in Keystone but with the wrong password
  + Attempt to login to Park xGreeter App with user and password in Keystone but with role different that Greeter or Greeter Coordinator

# Performance Testing

Park xGreeter App response times will be dependent on a number of external dependencies and on the task being performed. The table below lists target response times for different features of the Park xGreeter App.

| **Feature** | **Maximum Acceptable Average Response Time (in ms)** |
| --- | --- |
| Login – the time since Cast Greeter taps on the Login button until he/she can perform another action (e.g. Open Location) | 5000 |
| Render latest DAP/reader event – the time that takes after a card tap and until the green / blue event is displayed on the device. | 1000 |
| Schedule reader/bio maintenance – the time that takes since Coordinator hits the Schedule reader/bio maintenance button until he/she can perform another action. | 2000 |
| Shutdown DAP/reader - the time that takes since Coordinator hits the Shutdown button until he/she can perform another action. | 2000 |
| Logout greeter – the time that takes since a greeter taps logout until the Login screen is rendered. | 2000 |
| Reset blue lane event – the time that takes since the Reset blue lane is tapped until user can perform another action. | 2000 |
| Render event detail display | 1000 |
| Application startup – the time since user taps on the xGreeter logo on the device until the Login screen renders. | 1000 |
| Back end system (xBRC) failover – the time that takes for the xGreeter App to determine that a problem happened and until user receives the error that location has a problem. | 30000 |

# High Availability

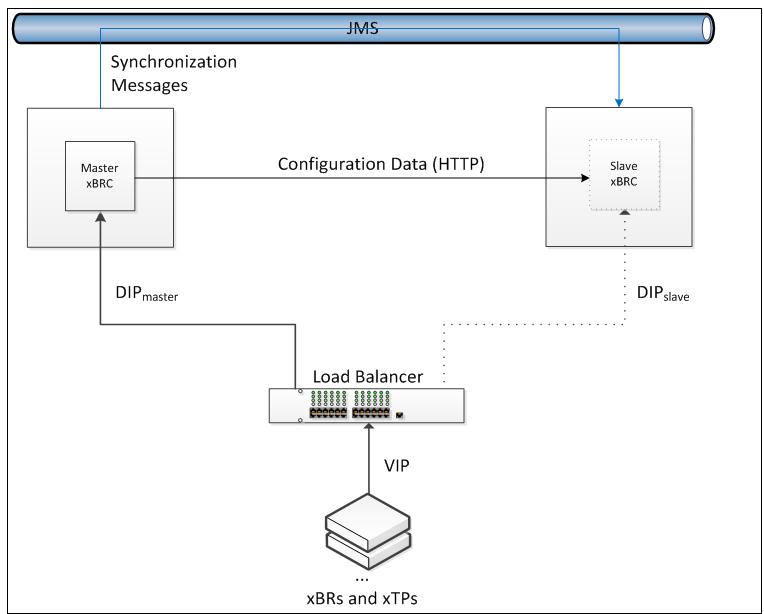
## Background

Version 1.5 of the Park Entry software introduces support for high availability (HA). The testing of the HA code and its rollout in FPT3 have demonstrated the importance of understanding how HA works and the importance of correctly configuring the xConnect system to assure that it works properly. This section provides instructions on how to setup HA in the Alpha Lab and provides test script that can be executed by users in Orlando.

## Objective

This document provides a high-level description of the Park Entry test strategy, setting up the test environment and provides a set of test scenarios that may be executed by users in the park as part of the User Acceptance Test (UAT)

## Network Architecture



## In Scope

xBRC – xBand Reader Controller (Master and Slave)

xBRMS – xBand Reader Management System (to set up xBRC HA properties)

VIP – Virtual IP address – the address that will be used by the xGreeter App

## Schedule and Timeline

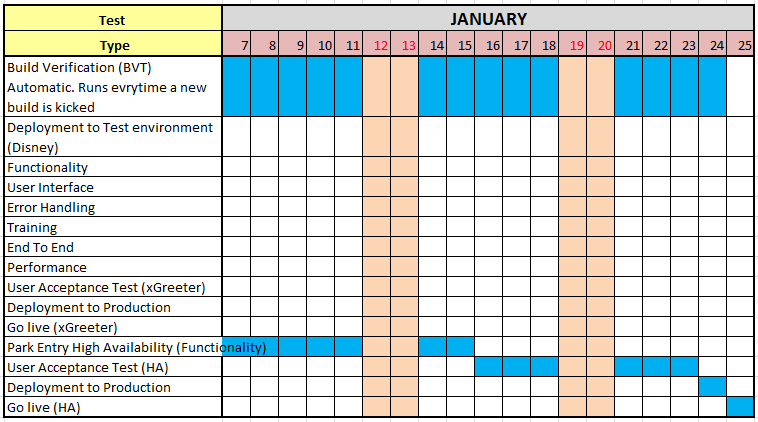
**Tasks**

Park Entry High Availability (Functionality)

User Acceptance Test (HA)

Deployment to Production

Go Live (HA)



## Setup instructions on xBRMS

* Browse to xBRMS:

<http://nl-flfa-00095.wdw.disney.com:8080/XBRMS/>

* Browse to “Edit xBRC Properties”
* Click on the “PARKENTRY” tab
* Expand PE-Alpha Lab(10.92.65.171)
* Make sure following properties are set up as follows:

Enable HA = true



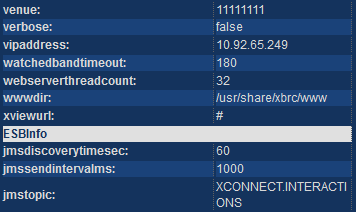
Venue = 11111111

Vipaddress = 10.92.65.249

Jmsdiscoverytimesec = 60

Jmssendintervalms: 1000

Jmstopic: XCONNECT.INTERACTIONS



* Expand PE-Alpha Lab(10.92.65.172)
* Make sure following properties are set up as follows:

Enable HA = true



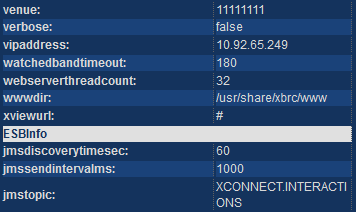
Venue = 11111111

Vipaddress = 10.92.65.249

Jmsdiscoverytimesec = 60

Jmssendintervalms: 1000

Jmstopic: XCONNECT.INTERACTIONS



* Restart xBRC 10.92.65.171
* Restart xBRC 10.92.65.172
* Refresh the xBRMS UI (<http://nl-flfa-00095.wdw.disney.com:8080/XBRMS/editxbrc>)
* Take a look of xBRCs in the right pane under PARKENTRY tab and verify:

One of the xBRCs configured above (nl-flfa-00160 or nl-flfa-00161) becomes the Master, the other becomes the Slave.

## Set up the xGreeter App on the device (iPOD)

* Browse to xGreeter > Settings
* Make sure you have:
* Version: Latest (as of today we have 1.5 (33). If you do not have latest then go to Apps@Work and download it.
* Mode: Failover.
* Venue: the correct one (for instance Magic Kingdom or Alpha Lab).
* xBRC1 = the VIP (for Alpha Lab is http:// nge-rl-xbrc-pe1.wdw.disney.com:8080)
* User Directory: ed

# Test Scenarios

## Pre-requisites (applicable to all scenarios):

Setup xGreeter settings in the iPOD:

Mode: Failover

Venue: Alpha Lab

xBRC1: http:// nge-rl-xbrc-pe1.wdw.disney.com:8080

User Directory: ed

Under xBRMS have the following:

xBRC1=10.92.65.171 HA enabled and with VIP=10.110.65.249

xBRC2=10.92.65.172 HA enabled and with VIP=10.110.65.249

xBRC1 the Master

xBRC2 the Slave

## Scenario 1 – Master unavailable when events are happening

Login to the xGreeter App in the iPOD  
Have guests actively tapping on the reader  
Have cast greeters monitoring taps (green/blue)   
Bring the master xBRC1 down

**Expected:**

Readers’ idle sequence is faster.

Greeter receives a message about location unavailable

After some time (about 15 seconds) xBRC2 becomes the master

Open location can be done (now BIP IP is talking to xBRC2)

Guests able to tap

Greeter able to monitor events

Bring xBRC1 up

**Expected:**

xBRC2 is the Master

xBRC1 is the Slave

## Scenario 2 - Master unavailable before logging in

Bring the master (xBRC1) down

**Expected:**

xBRC2 becomes the Master

xBRC1 is not available

Login to the xGreeter App in the iPOD  
Have guests actively tapping on the reader  
Have cast greeters monitoring taps (green/blue)

**Expected:**

Guests able to get green and blue events

Cast greeter able to open locations and to monitor readers

Bring xBRC1 up

**Expected:**

xBRC2 is the Master

xBRC1 is the Slave

## Scenario 3 – xBRMS unavailable

Login to the xGreeter App in the iPOD  
Have guests actively tapping on the reader  
Have cast greeters monitoring taps (green/blue)   
Bring the xbrms down

**Expected:**

xGreeter App looks in the cache and finds master and slave (no switching)   
Guests able to tap

Cast greeter able to monitor readers

Bring the xbrms up

Wait for about 1 minute

**Expected:**

Guests able to tap

Cast greeter able to monitor readers

xBRC1 is the Master

xBRC2 is the Slave

## Scenario 4 – Device (iPOD) cannot reach the Wi-fi

Login to the xGreeter App in the iPOD  
Have guests actively tapping on the reader  
Have cast greeters monitoring taps (green/blue)   
Have the iPOD w/o wireless (walk to a point where the Wi-fi is lost)

Attempt to Open/Close/Flash location

**Expected:**

Network Error

Return back to the place where Wi-fi is available

Wait until the iPOD connects to the Wi-fi

**Expected:**

Guests able to tap

Cast greeter able to monitor readers

## Scenario 5 – a switch from Master to Slave happens when the device (iPOD) cannot reach the Wi-fi

Login to the xGreeter App in the iPOD  
Have guests actively tapping on the reader  
have cast greeters monitoring taps (green/blue)   
Have the iPOD w/o wireless (walk to a point where the Wi-fi is lost)

Attempt to Open/Close/Flash location

**Expected:**

Network Error

Bring the master (xBRC1) down

xBRC2 becomes the Master

xBRC1 is not available  
Return back to the place where Wi-fi is available

Wait until the iPOD connects to the Wi-fi

**Expected:**

Cast greeter able to open location (now BIP IP is talking to xBRC2)

Guests able to tap

Cast greeter able to monitor readers

Bring xBRC1 up

**Expected:**

xBRC2 is the Master

xBRC1 is the Slave

# Release 1.6 (March 2013)

This release is intended to fix xGreeter App and xBRC Park Entry defects found in previous versions; no new functionality was added. The following is a list of fixed items:

|  |
| --- |
| App able to handle only one blue event under My Touch Points |
| Associate cast member with devices |
| Back icon not consistent among pages |
| Cannot contact any of the 1 configured Omni servers |
| Cannot open location. xBRC claims that already opened, but xGreeter App shows that it is not. |
| Check for xfp+xbio reader type in the Park Entry Model |
| Clicking on the Geolocation icon opens a dialog, rather than reset |
| Close location having an xTP with a bad Bio: xGreeter App spins and ultimately returns a timeout |
| Coordinator: Show the "person" icon for locations that are open in their location list view. |
| Fortify: OminServerService |
| Fortify: sendGetGuestInfo() sometimes fails to release a system resource |
| Geolocation icon is always a Compass |
| Greeter App doesn't refresh its reader list when a reader is moved/unlinked |
| Greeter role and UX persisted after log out for Greeter Coordinator |
| Installer for xBRC-parkentry does not remove/replace older version |
| IO error communicating with xBio causes xTP to play thinking sequence until abandoment occurs |
| It looks like both users have the same location for about 5 seconds or so until the notification comes through |
| Log out is not clearing Blue events |
| Not able to login at the first attempt after application is launched |
| omniPassword missing in the closeLocation request |
| Replace "Shutdown" for "Out of Service" wherever appears |
| Schedule Bio Maintenance persisted after canceling dialog |
| Schedule Maintenance comments persisted after canceling the dialog |
| secure ID should be masked when logging TOR responses |
| Venue map is not accurate |
| When Slave xBRC transitions to Master, ignore reader events |
| Wrong message when trying to log in when the device is not connected to the Wi-fi |
| xBRC - idleSequence does not cause xTPv2 to enter idle on park entry |
| xBRC is not sending cast app notice of unresponsive reader in timely matter |
| xBRC should include 'location' object in JSON to xGreeter App even if no location exist |
| xBRC should tell xBio to cancel biometric scans when guest abandons |
| xGreeter - add "reconnecting..." screen |
| xGreeter - app timeout |
| xGreeter - need xBRC to report hello when back online |
| xGreeter - No Wi-fi mode |
| xGreeter - support more than 3 xBRCs in the cache |
| xGreeter - visual indicator if no connection to xBRC |
| xGreeter improved user experience for poor Wi-fi coverage |

# Release 1.7 (July 2013)

## Scope

This release is intended to fix xGreeter and xBRC Park Entry defects found in previous versions; new functionality was added to the xGreeter App to improve xGreeter App – xConnect connectivity, Wi-Fi issues, and for debugging purposes. xBRMS UI and xBRC UI were modified to substitute the navigation breadcrumb with navigation menu – submenu drop down.

**Events and Audit Subsystem** is a new feature that will be released as part of 1.7. The goal is to create an enterprise database of security and system health events for all xConnect components. The database will enable the maintenance personal to examine functionality of the system over a period of time and find answers to such questions as who did what and when, as well as what happened to the system and why. It will be possible to query the database either directly or via specialized reporting tools that can be developed over time.

Besides providing the historical data the system will be able to take real-time actions in response to critical systems events. For example, it can send an SMS or email to an administrator when a certain event occurs.

In the scope of this document are included events and audit entries generated by the xGreeter App. Currently this component neither pushes nor caches audit events for later collection by xBRMS. xBRMS itself generates events on its behalf when xGreeter App requests a secure token for user authentication from xBRMS.

The following is a list of items addressed:

|  |
| --- |
| xGreeter: **Null Pointer Exception when connecting to an xBRC without any locations.**  Test Cases executed:   * Login to xGreeter App when xBRC under settings is an xBRC without locations. |
| xGreeter: **The greeter app losing its state after xBRC restarts.**  Test cases executed:   * In an xBRC HA Configuration, stop both Master and Slave xBRCs, then bring them both back. |
| xGreeter: **Added native iOS background ping (in attempt to keep iOS network alive).**  Regression:   * xGreeter App login, open location, cause green and blue events, monitor events, clear blue events, close location. Log out from xGreeter App. |
| xGreeter: **Added additional debug information for diagnosing Wi-fi network issues.**  Regression:   * xGreeter App login, open location, cause green and blue events, monitor events, clear blue events, close location. Log out from xGreeter App. |
| xGreeter: **Added additional debug settings for Apple use.**  Regression:   * xGreeter App login, open location, cause green and blue events, monitor events, clear blue events, close location. Log out from xGreeter App. |
| xGreeter: **Added background thread that "pings" heartbeat endpoint on xBRC.**  Regression:   * xGreeter App login, open location, cause green and blue events, monitor events, clear blue events, close location. Log out from xGreeter App. |
| xGreeter: **Added functionality to enable/disable UI based on heartbeat results (i.e., " xBRC reachability", a preferred Apple term).**  Regression:   * xGreeter App login, open location, cause green and blue events, monitor events, clear blue events, close location. Log out from xGreeter App. |
| xGreeter: **Added/updated hi-res images for MK, HS, and TL venues.**  Test cases executed:  Verified venue maps orientation, and look and feel. |
| xGreeter: **Added pinch-to-zoom functionality on venue map display.**  Test cases executed:   * Verified zoom in and zoom out by touching and moving fingers. |
| xGreeter: **Bug fix for connection check interval setting.**  Test cases executed:   * Monitoring green end blue events, clearing blue events after setting ‘Connection Check Interval’ to different values such as ‘0’, ‘3000’, ‘30000’. |
| xGreeter: **Added long polling interval setting. Used to change the time-out for the long-polling. At request of Apple.**  Test cases executed:   * Monitoring green end blue events, clearing blue events after setting ‘Polling Interval’ to different values such as ‘0’, ‘3000’, ‘30000’. |
| xGreeter: **Added back in location status checks even when an invalid reader/tor is detected.**  Test cases executed:   * Monitor locations and readers when communication between reader and tor is invalid for different reasons such as invalid device id reader not communicating with tor. |
| xBRC: **Add dependency on the reader RPMs to the xBRC RPM. This way any installation of the xBRC will ensure that the appropriate reader packages are also installed on the system.** |
| xBRC: **Open location contains 'closed' reader.**  Test cases executed: |
| xBRC: **PETransaction table's 'omniEntitlementErrorDesc' length is too short.**  Test cases executed: |
| xBRC/xBRMS: **Remove breadcrumb model for navigation from all UI screens.**  Test cases executed:  xBRC and xBRMS UI browsing using the new menu with the following Internet Browsers: Firefox, Chrome, Internet Explorer 9. |
|  |

## Test Resources

Rob Silvernagel

Jason Olmstead

Michael Lampi

Francisco Gonzalez

## Test Deliverables

Test Plan (this doc)

Test Cases (Excel document including xGreeter, Park Entry, and Audit System test cases.)

Test cases to verify the fixed bugs and new features to be executed. Besides a comprehensive regression will be performed as well with the purpose of ensuring both the xGreeter Application and xBRC have not been broken.

## Exit Criteria

* 100% of planned test cases have been executed and test results have been recorded with a pass or fail.
* Overall test pass rate of 90% is accomplished once Triage decides which bugs can be postponed, if any.
* All Test Cases have been successfully executed.
* All the defects have been appropriately documented in FogBugz with correct description, status, and are appropriately assigned to relevant team / individual.
* All test cases are updated as necessary and all expected and actual results related to each test case are documented within FogBugz.
* All known “critical” (Pri 1) and “high” (Pri 2) defects have been resolved and closed.
* All known defects have either been Resolved and Closed during Integration testing.
* All postponed defects have been reviewed, and signed-off by the Program Lead, Program Manager, Development Manager, QA Test Manager.
* Workarounds have been identified and documented for all defects that are postponed.

## Timeline

