

# TEST PLAN

Version 4.0  
April 21, 2011



## Revision Sign-off

By signing the following, the team member asserts that he has read the entire document and has, to the best of his knowledge, found the information contained herein to be accurate, relevant, and free of typographical error.

Name	Signature	Date
Michael Fore		
Sean Owen		
Anh Pham		
Jeffrey Regan		
Alex Welsh		
Matthew Williams		

## Revision History

The following is a history of document revisions.

<b>Version</b>	<b>Changes</b>	<b>Edited</b>
<b>Version 1.0</b>	Initial draft.	<b>2/28/2011</b>
<b>Version 1.1</b>	Various revisions. Added Meteor Defense and Surface Hockey to Unit Testing section.	<b>3/22/2011</b>
<b>Version 4.0</b>	Minor revisions, update to version 4.0.	<b>4/21/2011</b>

---

## Contents

Revision Sign-off .....	ii
Revision History .....	iii
1. Introduction.....	1
1.1 Purpose.....	1
1.2 Overview.....	1
2. Resource Requirements .....	2
2.1 Hardware.....	2
2.2 Software .....	2
3. Unit Testing .....	3
3.1 <i>Healing Vision</i> .....	3
3.2 <i>Healing Touch</i> Framework - Surface.....	3
3.3 <i>Healing Touch</i> Framework – iPad .....	4
3.4 Game Testing.....	4
3.4.1 Froggie Says.....	4
3.4.2 Touchdown! .....	5
3.4.3 Meteor Defense.....	5
3.4.4 Air Hockey.....	5
4. Integration and System Testing .....	6
4.1 Requirements Traceability Matrix .....	6
4.2 Definition of Test Cases.....	8
4.2.1 Test Case #1 .....	8
4.2.2 Test Case #2.....	8
4.2.3 Test Case #3 .....	9
4.2.4 Test Case #4.....	10
4.2.5 Test Case #5.....	10
4.2.6 Test Case #6.....	11

# 1. Introduction

## 1.1 Purpose

The purpose of this document is to outline how testing on the *Healing Touch* system will be performed. It describes how the testing procedure will differ on both the Surface and iPad platforms. These test cases are designed to ensure that all aspects of the *Healing Touch* project meet system requirements.

## 1.2 Overview

**Section 2 – Resource Requirements** – Defines the necessary hardware and software components for the testing of the *Healing Touch* system.

**Section 3 – Unit Testing** – Describes the unit testing of specific *Healing Touch* components.

**Section 4 – Integration and System Testing** – Description of the system testing approach taken in testing the *Healing Touch* system and its various components including Requirements Traceability Matrix and Test Case definitions.

## 2. Resource Requirements

### 2.1 Hardware

The hardware that will be required for the *Healing Touch* project testing includes:

- Microsoft Surface
- Apple iPad
- Machine with Microsoft Server 2008 R2 running SQL Server 2008
- Workstation running the *Healing Touch* standalone application, *Healing Vision*
- Wireless access point with wired connections to workstation, database, and Surfaces.

### 2.2 Software

The software required to be installed on the testing computer for the *Healing Touch* project testing includes:

- *Healing Vision* application

The software required to be installed on the Microsoft Surface and Apple iPad platforms for *Healing Touch* project testing includes:

- *Healing Touch* Framework v1.0

## 3. Unit Testing

Unit Testing of *Healing Touch* is being divided up into four parts: functionality of *Healing Vision*, the *Healing Touch* Surface framework, the *Healing Touch* iPad framework, and the individual game testing.

- ***Healing Vision***: The testing of all aspects of the database, including the functionality of the *Healing Vision* application and communication between all platforms and the database.
- ***Healing Touch framework - Surface***: Functionality of the main framework on the Surface including database communication, tag recognition, and menu testing.
- ***Healing Touch framework - iPad***: Functionality of the main framework on the iPad including local database storage and communication with the main database.
- ***Individual games***: The overall functionality of each game created for the *Healing Touch* system, including proper response to user input, animations, and loading times.

### 3.1 *Healing Vision*

Determine that:

- Only registered administrators can login.
- Users can add new clinicians.
- Pages are dynamically generated from tables and columns in the database.
- Add and edit patient are functioning. Clinicians can generate and print reports as well as create and store queries.
- User can print patient profiles.
- Correct information is being sent to and retrieved from the database.
- Errors such as loss of connection to the database are handled.
- Users can set game options for patients.

### 3.2 *Healing Touch Framework - Surface*

Determine that:

- Framework starts and stops correctly.
- Login sequence and tag recognition works only for registered patients.
- Free Play mode is available.
- Menu path and GUI navigation is all correct.
- Patients can store a preferred menu screen.
- Information sent to and retrieved from the database is correct.

- Errors such as loss of connection to the database are handled.

### **3.3 Healing Touch Framework – iPad**

Determine that:

- Application starts and stops correctly
- Configuration of iPad for a patient retrieves the correct information and stores it in the local database.
- Menu path and GUI navigation is all correct.
- Patients can set and store preferred menu screen.

### **3.4 Game Testing**

Determine that:

- Surface – Test Mode:
  - Game options are retrieved from the database on application start.
  - Clinician can set temporary game options by dropping his/her ID card onto the in-game menu screen.
  - Free Play mode enables guests to choose their own game options. Otherwise, default options are used.
  - Correct game options for the patient are read from the database.
  - All game data is sent to the database at the end of a game.
  - Disconnection from the database in the middle of a game is handled correctly.
- Surface – Free Play
  - Free Play mode enables guests to choose their own game options. Otherwise, default options are used.
- iPad
  - Correct game options are loaded on game start from the local database.
  - Game data is sent to the local database at the end of a game.
  - Validate that all game data in the local database is sent to the main database when uploaded by the clinician.

#### **3.4.1 Froggie Says**

Determine that:

- Game waits the correct amount of time before allowing user input.
- Only menu buttons are enabled before a game begins. Game play input buttons do not accept input until the patient turn begins.



- Sequence sent to the database matches actual game sequence.
- Delay time and countdown timer function properly.
- All game types (color, shapes, etc.) work correctly.

### **3.4.2 Touchdown!**

Determine that:

- Number of footballs on the screen does not affect game play or frame rate.
- Camera correctly follows ball in flight.
- Target is always set at an achievable distance from the thrower.
- Wind speed never makes it impossible for a ball to hit the target.
- Game ends on target hit.

### **3.4.3 Meteor Defense**

Determine that:

- Accuracy tracker updates correctly.
- Meteor speed and frequency related to cannon heat bar is reasonable.
- Game difficulty progresses smoothly.
- Number of meteors and amount of action on the screen does not affect the frame rate or responsiveness of the game.
- Meteor frequency and speed stays within the bounds set by the clinician.
- Meteors correctly target cities and cities are destroyed in one hit.
- Game ends correctly on both conditions: time runs out is a win, all cities destroyed is a loss.

### **3.4.4 Air Hockey**

Determine that:

- Game options screen correctly configures the game.
- Game ends when correct conditions are met (timer or goal limit).
- Air Hockey paddles are responsive to user input and puck is affected realistically by paddle and wall hits.

## 4. Integration and System Testing

### 4.1 Requirements Traceability Matrix

System testing will be based on a series of test cases that cover all systems of the *Healing Touch* project. The Requirements Traceability Matrix shows what requirements each test case covers. Details of each test case are given in section 4.2.

Healing Touch Framework Requirements	Test Cases					
	1	2	3	4	5	6
GEN-01 - Patients shall be able to play a variety of games.	X	X	X	X	X	X
GEN-02 - Patient profile and gaming results shall be stored.	X	X	X		X	X
GEN-03 - Database access limited to Surface, iPad, and Healing Vision.	X	X	X	X	X	X
SUR-01 - Users can choose between Test Mode and Free Play.	X			X	X	
SUR-02 - Surface shall have a wired connection to the database.				X		
SUR-03 - The user shall choose between grid menu and album menu.	X				X	
SUR-04 - Clinicians shall be able to choose temporary game options.	X				X	
SUR-05 - Guests shall be able to select game options from a game menu.				X	X	
IPD-01 - Clinicians shall configure the iPad for a specific patient.		X	X			X
IPD-02 - Game data shall be stored locally on the iPad			X			X
IPD-03 - The user shall choose between grid menu and album menu.		X	X			X

Healing Vision Requirements	Test Cases					
	1	2	3	4	5	6
HVN-01 - Clinicians shall use <i>Healing Vision</i> to access the database.	X	X	X		X	X
HVN-02 - Clinicians shall create, edit, and print patient profiles.	X	X	X		X	X
HVN-03 - Clinicians shall be able to generate reports.		X			X	X
HVN-04 - Clinicians shall be able to select game options.		X	X		X	X
HVN-05 - <i>Healing Vision</i> forms shall be dynamically created.		X	X		X	X
HVN-06 - <i>Healing Vision</i> shall display a disconnection error if lost.			X			X

Froggie Says Requirements	Test Cases					
	1	2	3	4	5	6
FSR-01 - The computer shall display a sequence for the player to repeat.	X	X			X	X
FSR-02 - The game shall build a random sequence of button presses.	X	X			X	X
FSR-03 - The game shall play two sounds.	X	X			X	X
FSR-04 - The game shall end on an incorrect button press.	X	X			X	X
FSR-05 - The game shall send relevant information to the database.	X	X			X	X
FSR-06 - The clinician shall be able to set temporary game options.	X				X	

FSR-07 - There shall be a delay between the computer and player turn.	X	X			X	X
---	---	---	--	--	---	---

Touchdown! Requirements	Test Cases					
	1	2	3	4	5	6
TDN-01 - The player shall throw footballs at a target.			X	X	X	X
TDN-02 - The player shall throw until the target is hit.			X	X	X	X
TDN-03 - The screen shall pan to follow the football in flight.			X	X	X	X
TDN-04 - Target distance shall be random between two values.			X	X	X	X
TDN-05 - Information shall be collected at the end of the game.			X	X	X	X
TDN-06 - Options shall be available for clinicians to set.				X	X	

Meteor Defense Requirements	Test Cases					
	1	2	3	4	5	6
MTR-01 - The player shall defend four cities against meteors.					X	X
MTR-02 - Difficulty shall increase as the game progresses.					X	X
MTR-03 - The cannon shall have a heat bar.					X	X
MTR-04 - Heat gauge, accuracy, and game timer shall be shown.					X	X
MTR-05 - Cannon heat bar shall be derived from the rate of spawn.					X	X
MTR-06 - The cities shall be destroyed on one impact from a meteor.					X	X
MTR-07 - The game shall end when cities are destroyed or time ends.					X	X
MTR-08 - The clinician shall set the frequency and speed of meteors.					X	
MTR-09 - The clinician shall be able to set temporary game options.					X	

Surface Hockey Requirements						
	1	2	3	4	5	6
HOC-01 - The game shall be playable by two users in Free Play mode.					X	X
HOC-02 - Players shall hit a puck with virtual paddles to score goals.					X	X
HOC-03 - Surface hockey shall not communicate with the database.					X	X
HOC-04 - Users shall be able to set game options.					X	X
HOC-05 - The game shall end when correct conditions are met.					X	X

## 4.2 Definition of Test Cases

### 4.2.1 Test Case #1

#### *Healing Vision/framework/Froggie Says on Surface*

- Primary items tested:
  - *Healing Vision*
  - Surface framework and database connectivity
  - Froggie Says functionality
- Test Operations:
  - Create test patient using *Healing Vision*
  - Edit patient information
  - Print Patient ID tag and login to *Healing Touch* on Microsoft Surface
  - Set preferred home screen to album view
  - Play game of Froggie Says
  - On game end, check game data in main database
  - Set temporary game options using clinician ID tag
  - Play game of Froggie Says
  - On game end, check game data in main database
  - Logout of *Healing Vision*
  - Log back in as test patient
  - Change preferred home screen to grid view
  - Play game of Froggie Says – game should revert back to original options
  - On game end, check game data in main database
  - Exit *Healing Touch*

### 4.2.2 Test Case #2

#### *Healing Vision/framework/Froggie Says on iPad*

- Primary items tested:
  - *Healing Vision*
  - iPad framework and database connectivity
  - Froggie Says functionality
- Test Operations:
  - Login to *Healing Vision* as administrator
  - Check to see that forms are dynamically generated
    - Add new patient attribute row in main database
    - Validate that *Healing Vision* forms are updated
  - Create test patient using *Healing Vision*

- Configure iPad for test patient use
- Start *Healing Touch* application on iPad
- Set the game menu screen to album view
- Play a game of Froggie Says
- On game end, check game data in main database
- Set new game options for test patient using *Healing Vision*
- Set the game menu screen to grid view
- Play a game of Froggie Says – ensure that new game options are set
- On game end, check game data in main database, including the preferred menu screen
- Exit *Healing Touch*
- In *Healing Vision*, generate report of all Froggie Says game data
- Print Froggie Says report
- Exit *Healing Vision*

### 4.2.3 Test Case #3

#### **Connection handling/Local iPad database storage/Touchdown!**

- Primary items tested:
  - Disconnection error handling
  - iPad local database storage and upload function
  - Touchdown! on iPad
- Test Operations:
  - Login to *Healing Vision* as an administrator
  - Create a new test patient
  - Unplug *Healing Vision* computer from the network
  - Check for disconnection notice/error handling
  - Reconnect *Healing Vision*. Attempt to edit patient information and game attributes for Touchdown!
  - Verify edited information in the database
  - Configure iPad for test patient use
  - Leave network range and start *Healing Touch* on iPad
  - Play a game of Touchdown!
  - On game end, return to network and upload the game data from the local database to the main database
  - Verify that the game data was correctly sent to the database
  - Exit *Healing Touch* and *Healing Vision*

#### 4.2.4 Test Case #4

##### Free Play/Surface error handling/Touchdown! on Surface

- Primary items tested:
  - Free Play on Surface
  - Surface disconnection handling
  - Touchdown! on Surface
- Test Operations:
  - Login as guest on Surface by selecting Free Play
  - Disconnect Surface from network
  - Check for error message and instructions
  - Reconnect Surface to network
  - Start Touchdown! game
  - Check guest options menu and choose new game options
  - Play a game of Touchdown!
  - On game end, select new game options from the guest menu
  - Play a game of Touchdown!
  - On game end, verify that no game data is sent to the database
  - Exit *Healing Touch*

#### 4.2.5 Test Case #5

##### Healing Vision/Healing Touch on Surface/All games on Surface

- Primary items tested:
  - Healing Vision
  - Healing Touch – Surface
  - All games on Surface
- Test Operations:
  - Create test patient using Healing Vision
  - Print Patient ID card
  - Login to Healing Touch on Surface
  - Play a game of Froggie Says
  - Set new game options using clinician ID card
  - Play a game of Froggie Says
  - Change Froggie Says game options using Healing Vision
  - Logout and login to reload new game options
  - Play a game of Froggie Says
  - Verify all game data is correct in the database
  - Logout and enter Free Play Mode

- Test game menu views(album and grid)
- Start Air Hockey
- Set game options for Air Hockey
- Play a game of Air Hockey
- Play a game of Touchdown!
- Set new game options on guest options page
- Play a game of Touchdown!
- Logout
- Use Healing Vision to set options for Touchdown!
- Login as test patient
- Play a game of Touchdown!
- Set new game options using clinician ID card
- Play a game of Touchdown!
- Play a game of Meteor Defense
- Disconnect Surface from network in the middle of the game
- Reconnect Surface and set new options using clinician ID card
- Logout and set new Meteor Defense options using Healing Vision
- Login
- Play a game of Meteor Defense
- Verify game data is correct in the database
- Generate report for all games played on current date on the Surface
- Print report

#### 4.2.6 Test Case #6

##### Healing Vision/Healing Touch on iPad/All games on iPad

- Primary items tested:
  - Healing Vision
  - Healing Touch on iPad
  - All games on iPad
- Test operations:
  - Create test patient using Healing Vision
  - Edit patient information
  - Unplug Healing Vision – check disconnection handling
  - Set game options for Froggie Says, Touchdown!, and Meteor Defense ( there are no options for Air Hockey)
  - Configure iPad for test patient use
  - Leave range of network
  - Start Healing Touch application

- Play one instance of each game (Froggie Says, Touchdown!, Meteor Defense, Air Hockey)
- Change game menu style from album to grid
- Return to network and upload data as clinician
- Verify game data was uploaded to the database
- Set new game options for all games using Healing Vision
- Reconfigure iPad for test patient use
- Leave range of network
- Play one instance of each game
- Verify default menu style is now grid layout. Change to album view.
- Return to network and upload game data
- Verify game data was uploaded to the database
- Set new options, leave network range, and play one instance of each game
- Return to the network and upload game data to the database
- Check all game data in the database for accuracy
- Generate a report for the game data played on current date on the iPad
- Print report