Project – Testing Priceline

Online travel booking and bidding App

Instructor:

Dr Jerry Gao

By:

Rui Zou

Nan Ding

XiaoRui Wang

Nie Chao

Contents

1. Introduction

This test design document is written for mobile application “Priceline”. It contains information on how the testing is divided and how the team approach to design test case around.

Priceline app, while it has a lot of the similar features like other online booking apps, it has its own unique features that makes it stand out. The basics, it allows user to search, book, cancel for hotels, flights as well as rental cars. On top of that, what is special about Priceline is that it gives user the option to bid on reservations they want without revealing the hotel name beforehand. Once a bid is placed and accepted by the hotel, then user can not cancel anymore. It uses a credit card for bidding guarantee.

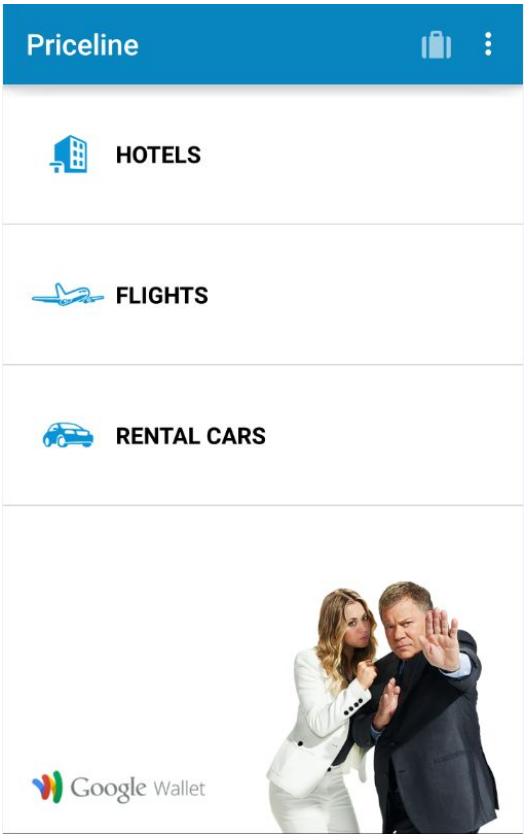


Figure 1 APP Main Page

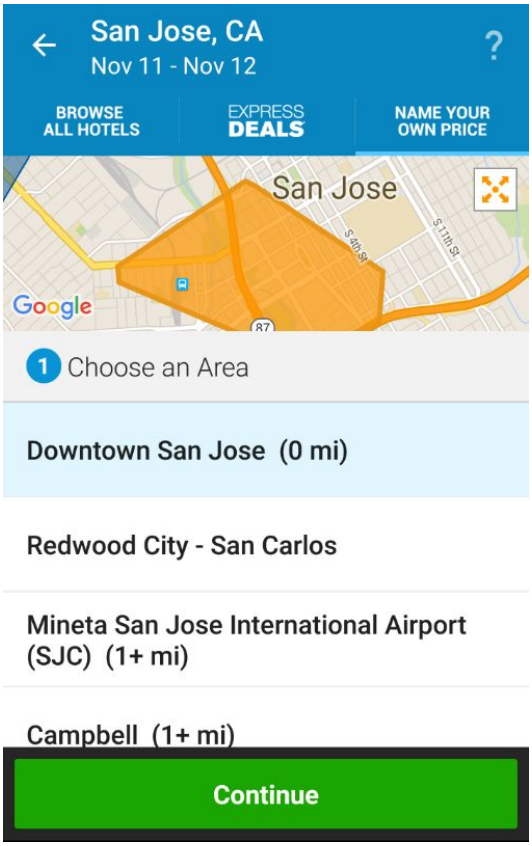


Figure 2: Bidding – Select Hotel Area

TEST PLAN

In this test design document, we will create and document our testing process in detail following our template of record. It has elements like test case number, test environment, version number, test input, test output, date, tester name etc. Various methods are used by providing test cases that covers all possible scenarios following the template below.

Test Case ID:		Test Item:	
Tester Name:		Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	
Product Name:	Priceline	Version #:	Version 3.28.97
Test Case Description:			
Operation Procedure:			
Pre-Condition:		Post Condition:	
Input Data:		Expected Output Data:	
Test Script:			

1.1 Testing Overview

The essential part of software development before release is software testing. The work around software testing is to ensure quality. It is done during software development process and before release as well as post release. In general, software testing can be divided into two categories, white box testing and black box testing. White box testing require access to the inner works of the software including source code. Most times, white box testing is done by software developers during the development process. Black box testing does not require access to inside of the software piece, but rather, to test the software functionalities, and see if it meets the external requirement and is done from an end users' perspective.

Both testing methods serves two purposes, to validate and to verify. Verification purpose is to ensure that software correctly implement a specific function, while validation purpose is to ensure that software that has been built is traceable to customer requirements.[1]

1.2 Document Structure

We will include the following topics in this document:

1. Test Methods: Chapter 1 and chapter 2. Explains our reasons behind why we design testing the way we did and how we approach testing this app.
2. Test Cases: This will include the test cases we come up with following our testing methodologies.
3. Summary: This concludes our testing and provide feedback to the development team about test findings.

1.3 Scope

The scope of a software testing includes but is not limited to the following:

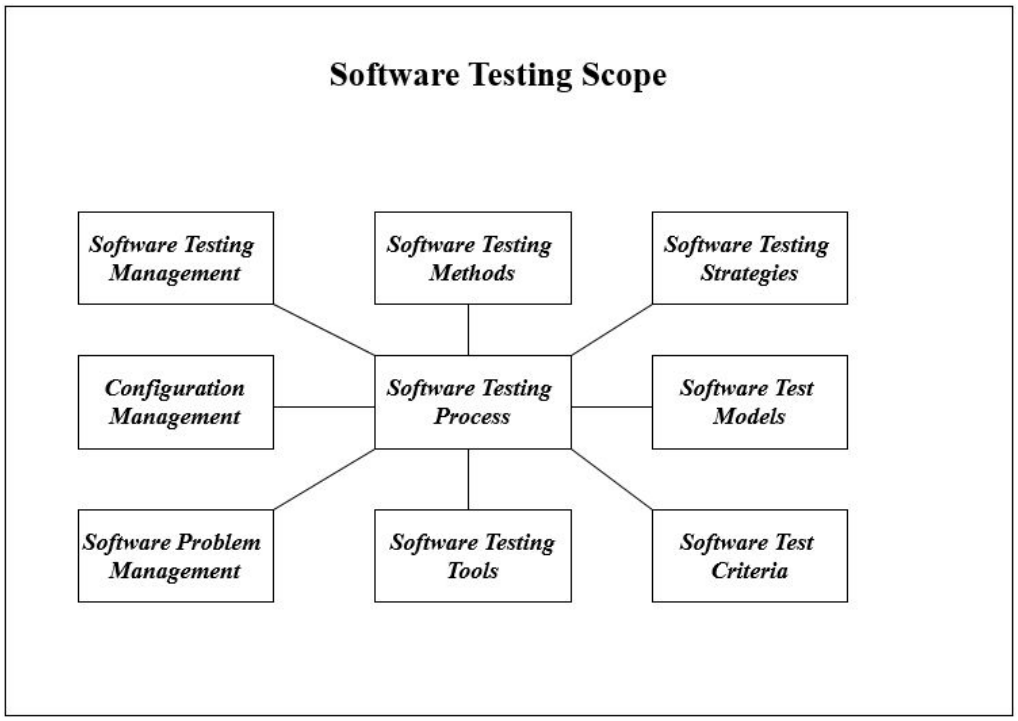


Figure 3: Software Testing Scope – by Dr Jerry Gao

In this project, since we are using a commercial off the shelf app as our testing object and do not have access to source code, we will use black box methodology combined with the following test methods:

- Equivalence Partition Testing
- Category Partition Testing
- Decision Table Testing

2. Testing Type, Design and Methods

2.1 Performance Testing

Software performance is an important aspect of testing. It is the first impression that a user will get about a software. Dull and slow apps will normally lead to bad reviews as well as low user adoption rate.

We will try to do the following:

1. Validate successful hotel search
2. Validate successful flight search
3. Validate successful rental car search
4. All of the above should be completed within a reasonable time frame once a user send the request

2.2 Automation Testing

Software testing automation is for the software tester to design the framework of testing, write testing scripts and allow the system to automatically go through hundreds if not thousands of test cases without human intervention. It also include data logging, sorting and brief data analysis. In some cases, automated testing will compare automatically expected output with actual output and note any abnormally in the test report in a visually easy to read format, in order to make testers job easier. Automating the process also makes eliminate the possibility of human error and speed up testing process. Testing can be done in a very systematic way and ensure maximum coverage.

We have a few objective testing Priceline app:

- To search and make reservation
- Utilize test automation tools and become familiar with the tools
- Draw data driven conclusions from final test data

2.3 Applied tools and usage

All tools used in the project will be open source tools available to us for free. We plan to be using the following, pending trial and error to determine which one suites our need the best:

Selenium IDE

Introduced in one of the lab sessions in the class, selenium seems to be an easy to use tool to navigate through the web. At the time of this writing, we're not sure if selenium can be equally applied to mobile app yet. But it has a few interesting features we like:

- Follow a sample testing procedure, record and repeat it
- Text verification
- Testing result feedback – Success? Failure?
- Easy to use and user friendly interface

Robotium Android Testing Tool

This is one of the first and mostly used automation testing tool for android apps. It is a free UI testing tool and is suitable for a wide variety of android versions. But it seems to lack the capability of taking screenshots.

MonkeyRunner App Testing

This is another popular low-level testing tool that does not require to have access to source code. It can record tests and can be run on real devices connected to a PC or emulator. Its API gains access to the device and controls it during the testing process.

2.4 Installation Testing

Before any testing can take place we first ensures that the app can be installed successfully in devices. We tried with our own devices, both Android and iOS.

Installation testing of app on Android

Test Case ID:	Install_1	Test Item:	Priceline App
Tester Name:	Nan Ding	Documented Date:	11/1/15
Test Type:	Installation Testing	Test Suite:	Google Play Store Download
Product Name:	Priceline app	Version #:	Version 3.28.97
Test Case Description:	To download the official priceline app from Google play store and install on Android device		
Operation Procedure:	Install app ● Search for app ● Click to select ● Click install ● Accept access request ● Open app		
Pre-Condition:	Android Phone Access to Google play store with a registered google account	Post Condition:	Priceline app installed on device
Input Data:	Play store Priceline app for download	Expected Output Data:	Successful installation of Priceline app
Test Script:	N/A		

Installation testing of app on Android device

Test Case ID:	Install_2	Test Item:	Priceline App
Tester Name:	Nan Ding	Documented Date:	11/1/15
Test Type:	Installation Testing	Test Suite:	Google Play Store Download
Product Name:	Priceline app	Version #:	Version 3.28.97
Test Case Description:	To download the official priceline app from Google play store and install on Android device		
Operation Procedure:	Install app ● Search for app ● Click to select ● Click install ● Accept access request ● Open app		
Pre-Condition:	Android Tablet Access to Google play store with a registered google account	Post Condition:	Priceline app installed on device
Input Data:	Play store Priceline app for download	Expected Output Data:	Successful installation of Priceline app
Test Script:	N/A		

Installation testing of app on iOS device

Test Case ID:	Install_3	Test Item:	Priceline App
Tester Name:	Nan Ding	Documented Date:	11/1/15
Test Type:	Installation Testing	Test Suite:	Google Play Store Download
Product Name:	Priceline app	Version #:	Version 3.28.97
Test Case Description:	To download the official priceline app from Google play store and install on Android device		
Operation Procedure:	Install app ● Search for app ● Click to select ● Click install ● Accept access request ● Open app		
Pre-Condition:	iOS phone Access to Google play store with a registered google account	Post Condition:	Priceline app installed on device
Input Data:	Play store Priceline app for download	Expected Output Data:	Successful installation of Priceline app
Test Script:	N/A		

Installation testing of app on iOS device

Test Case ID:	Install_4	Test Item:	Priceline App
Tester Name:	Nan Ding	Documented Date:	11/1/15
Test Type:	Installation Testing	Test Suite:	Google Play Store Download
Product Name:	Priceline app	Version #:	Version 3.28.97
Test Case Description:	To download the official priceline app from Google play store and install on Android device		
Operation Procedure:	Install app ● Search for app ● Click to select ● Click install ● Accept access request ● Open app		
Pre-Condition:	iOS tablet Access to Google play store with a registered google account	Post Condition:	Priceline app installed on device
Input Data:	Play store Priceline app for download	Expected Output Data:	Successful installation of Priceline app
Test Script:	N/A		

2.5 Black box Testing

The two testing principles we plan to use are

- Equivalence Partitioning

  This is a test method is best suited for an app like Priceline. For each state, input is divided into several different partitions and the expected output is divided into several outcomes as well. This allow the testers to achieve almost all input categories to verify expected output. The challenge is to create partition accordingly depend on requirement and not to miss a partition. Test result accuracy heavily depend on how accurately are partitions created.

- Decision-Table Based

Decision-Table as names, mainly tests the app in a way that covers all different combinations of TRUE or FALSE for things like business rules, constraints and conditions. Each condition and each rules has a True or False input, then based on business requirement, tester will come up with expected output being True or False. Tester will then compare system output with expect. This method is mainly used to check logic flow process within the app and ensure it works as expected. A table is constructed in the end to outline all combinations.

A simple example can be if having water is true, and having rice is true, then having cooked rice is also true. But if having water is true and having rice is false, then having cooked rice should also be false. Vice versa, if having water is false, means no water but having rice is true, then having cooked rice is still false because you don't have water to cook it with. So expected output is false, but if the actual output is true, then tester can mark the test case and say developer needs to review this section of the design because logic does not work properly.

2.6 List of Component to be tested

1. Account component

   Equivalence Partition Method

   Decision Based Testing Method

2. Hotel component

   Equivalence Partition Method

   Decision Based Testing Method

3. Flights component

   Equivalence Partition Method

   Decision Based Testing Method

4. Rental Car

   Equivalence Partition Method

   Decision Based Testing Method

In each component, we will use black box testing method to verify and provide various inputs and compare expected outcome against actual outcome.

1. Account component
   1.1 Create
   1.2 Update
   1.3 Login
2. Hotel component
   2.1 Search
   2.2 Browse
   2.3 Express Deals
   2.4 Bid
3. Flights component
   3.1 Search
   3.2 Filter
   3.3 Select
4. Rental Car
   4.1 Search
   4.2 Filter
   4.3 Select

3. Priceline Mobile App based Test Model

3.1 Testing Configuration

On android platforms, Priceline requires android 4.0 and above; on iOS platforms, Priceline only allow iOS 7.0 or later devices to download the app.

3.2 Mobile App Based Testing Environment

With automation testing or cloud platform provided testing environment, test cases can be divided into multiple combinations. However, with limited resource and time, the testing models will be cross tested as follows:

	Android 6.0	iOS 9.1	Hotel Search	Rental Car Search	Flight Search
Android 6.0	X		M1	M1	M1
iOS 9.1		X	M2	M2	M2
Hotel Search	M1	M2			
Rental Car Search	M1	M2			
Flight Search	M1	M2			

Figure 4: Test Environment

3.3 Semantic Tree Model for Priceline

Priceline is an app for searching hotels or rental cars conveniently, the significant feature is that Priceline has a very clear GUI feature trying to save users' effort for booking hotels as much as possible. The following is the semantic tree model of Priceline.

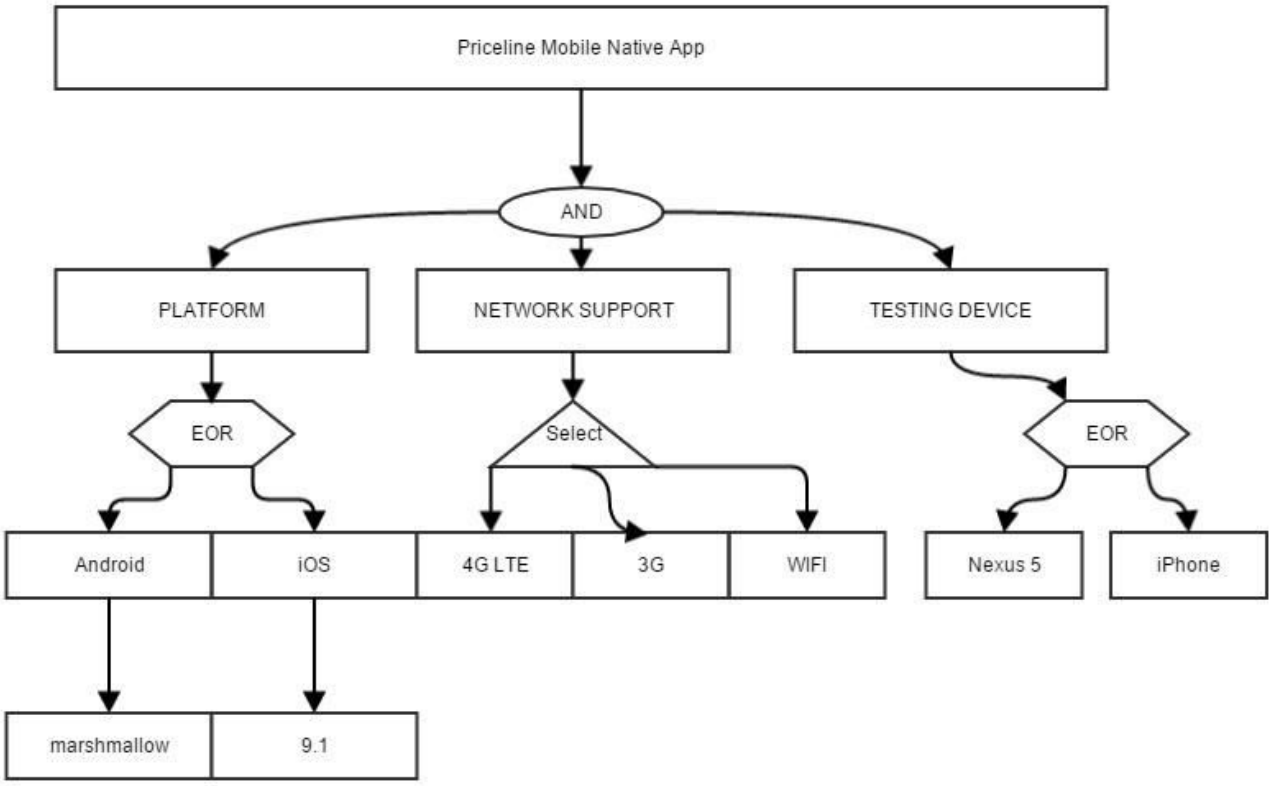
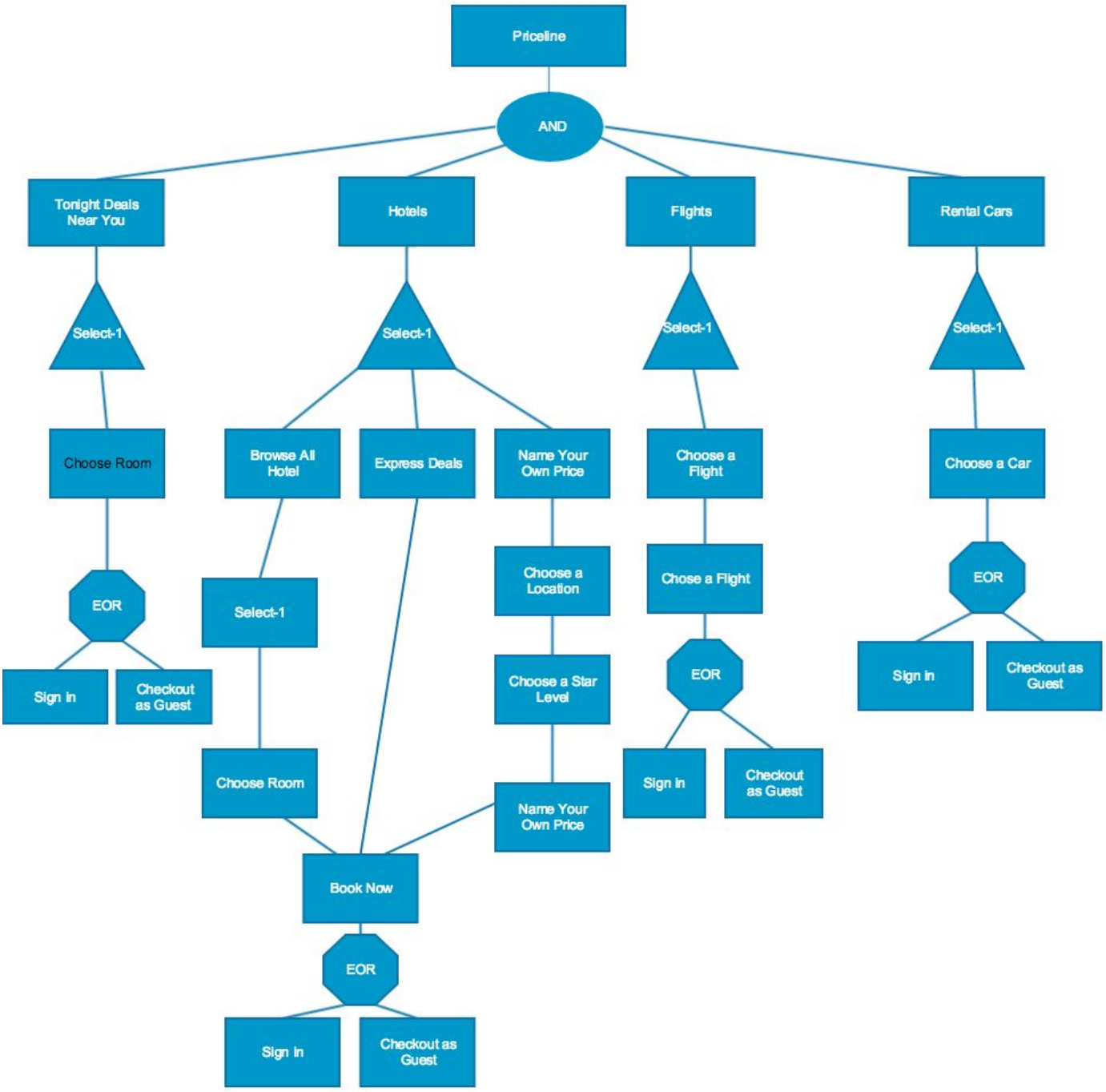


Figure 5: Semantic Tree for Priceline app

3.4 Semantic Tree Model for App Functionalities

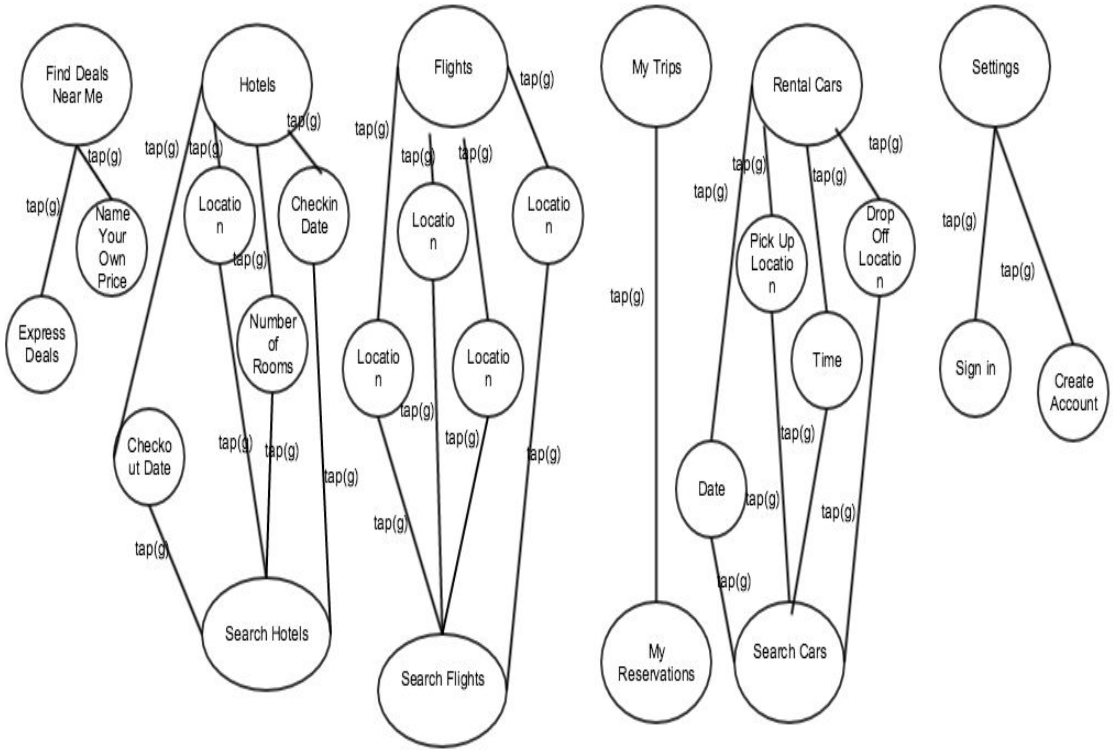

Priceline
AND
Tonight Deals Near You
Hotels
Flights
Rental Cars
Select-1
Select-1
Select-1
Select-1
Choose Room
Browse All Hotel
Express Deals
Name Your Own Price
Choose a Flight
Choose a Car
EOR
Select-1
Choose a Location
Chose a Flight
EOR
Sign in
Checkout as Guest
Choose a Star Level
EOR
Sign in
Checkout as Guest
Sign in
Checkout as Guest
Choose Room
Name Your Own Price
Book Now
EOR
Sign in
Checkout as Guest

Complexity for the above tree model:

Semantic Tree for Organization structure	Number of Leaves	Number Of Links	Max Height	Number of EOR	Number of AND
PriceLine Functionalities	8	36	4	4	0

3.5 GUI model

GUI is a very important part of for mobile application, since it present the actual views and show the content to users. In the following graph, it shows how GUI model works in this mobile application. The major gestures are swipe and tap. All events and contents are shown as icons, when users tap or swipe a specified icon, it will turn out to a new interface for the purpose of demonstration.


Find Deals Near Me
tap(g)
tap(g)
Name Your Own Price
Express Deals
Hotels
tap(g)
tap(g)
tap(g)
Locatio n
Checkin Date
tap(g)
Number of Rooms
tap(g)
tap(g)
tap(g)
Checko ut Date
tap(g)
Search Hotels
Flights
tap(g)
tap(g)
tap(g)
Locatio n
Locatio n
Locatio n
Locatio n
tap(g)
tap(g)
tap(g)
tap(g)
Search Flights
My Trips
tap(g)
My Reservations
Rental Cars
tap(g)
tap(g)
tap(g)
tap(g)
Pick Up Locatio n
Drop Off Locatio n
Time
tap(g)
Date
tap(g)
tap(g)
tap(g)
Search Cars
Settings
tap(g)
tap(g)
Sign in
Create Account

4. Test Case

4.1 Test case for account component

Account Creation

Test Case ID:	AC-1	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		In Priceline registration section, user must provide first name, last name, email address, a password with minimum 8 characters, including one number and one special character	
Operation procedure:		Click on top right corner go to menu, select sign in. Then go to top right corner again and click on "SIGN UP". Verify that the option to sign up with Google+ and sign up with email shows up	
Pre-conditions:	First name is not filled	Post-conditions:	None
Inputs data	No first name	Expected output	Missing first name prompt

Test Case ID:	AC-2	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		In Priceline registration section, user must provide first name, last name, email address, a password with minimum 8 characters, including one number and one special character	
Operation procedure:		Click on top right corner go to menu, select sign in. Then go to top right corner again and click on "SIGN UP". Verify that the option to sign up with Google+ and sign up with email shows up	
Pre-conditions:	Last name is not filled	Post-conditions:	None
Inputs data	No last name	Expected output	Missing first name prompt

Test Case ID:	AC-3	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		In Priceline registration section, user must provide first name, last name, email address, a password with minimum 8 characters, including one number and one special character	
Operation procedure:		Click on top right corner go to menu, select sign in. Then go to top right corner again and click on "SIGN UP". Verify that the option to sign up with Google+ and sign up with email shows up	
Pre-conditions:	Email is not filled	Post-conditions:	None
Inputs data	No email address	Expected output	Missing email prompt

Test Case ID:	AC-4	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		In Priceline registration section, user must provide first name, last name, email address, a password with minimum 8 characters, including one number and one special character	
Operation procedure:		Click on top right corner go to menu, select sign in. Then go to top right corner again and click on “SIGN UP”. Verify that the option to sign up with Google+ and sign up with email shows up	
Pre-conditions:	Password is less than 8 characters	Post-conditions:	None
Inputs data	Less than 8 character small case password	Expected output	Invalid password

Test Case ID:	AC-5	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		In Priceline registration section, user must provide first name, last name, email address, a password with minimum 8 characters, including one number and one special character	
Operation procedure:		Click on top right corner go to menu, select sign in. Then go to top right corner again and click on "SIGN UP". Verify that the option to sign up with Google+ and sign up with email shows up	
Pre-conditions:	Password is more than 8 characters but all small case letters	Post-conditions:	None
Inputs data	Password is more than 8 characters but all small case letters	Expected output	Invalid password

Test Case ID:	AC-6	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		In Priceline registration section, user must provide first name, last name, email address, a password with minimum 8 characters, including one number and one special character	
Operation procedure:		Click on top right corner go to menu, select sign in. Then go to top right corner again and click on “SIGN UP”. Verify that the option to sign up with Google+ and sign up with email shows up	
Pre-conditions:	Password is more than 8 characters but all numbers	Post-conditions:	None
Inputs data	Password is more than 8 characters but all numbers	Expected output	Invalid password

Test Case ID:	AC-7	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		In Priceline registration section, user must provide first name, last name, email address, a password with minimum 8 characters, including one number and one special character	
Operation procedure:		Click on top right corner go to menu, select sign in. Then go to top right corner again and click on "SIGN UP". Verify that the option to sign up with Google+ and sign up with email shows up	
Pre-conditions:	Password is more than 8 characters but only has small case letters and 1 number	Post-conditions:	None
Inputs data	Password is more than 8 characters but only has small case letters and 1 number	Expected output	Invalid password

Test Case ID:	AC-8	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		In Priceline registration section, user must provide first name, last name, email address, a password with minimum 8 characters, including one number and one special character	
Operation procedure:		Click on top right corner go to menu, select sign in. Then go to top right corner again and click on "SIGN UP". Verify that the option to sign up with Google+ and sign up with email shows up	
Pre-conditions:	Password is more than 8 characters, has one number and one special character	Post-conditions:	None
Inputs data	Password is more than 8 characters has one number and one special character	Expected output	Sign up successful, receive confirmation email

Test Case ID:	AC-9	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		In Priceline registration section, user must provide first name, last name, email address, a password with minimum 8 characters, including one number and one special character	
Operation procedure:		Click on top right corner go to menu, select sign in. Then go to top right corner again and click on “SIGN UP”. Verify that the option to sign up with Google+ and sign up with email shows up	
Pre-conditions:	Select the option to sign up with Google+	Post-conditions:	None
Inputs data	Google+ profile is not created	Expected output	Redirect app to Google settings page

Test Case ID:	AC-10	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		In Priceline registration section, user must provide first name, last name, email address, a password with minimum 8 characters, including one number and one special character	
Operation procedure:		Click on top right corner go to menu, select sign in. Then go to top right corner again and click on “SIGN UP”. Verify that the option to sign up with Google+ and sign up with email shows up	
Pre-conditions:	Select the option to sign up with Google+	Post-conditions:	None
Inputs data	Google+ profile was previously created	Expected output	Sign up successful

Account update

Test Case ID:	AC-11	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		Once logged in, users can update account information accordingly	
Operation procedure:		Click on top right corner go to menu Select sign in. Log in to your account, and click on settings	
Pre-conditions:	Logged into app with an account already created	Post-conditions:	None
Inputs data	Change first name, remove all character	Expected output	Invalid name prompt

Test Case ID:	AC-12	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		Once logged in, users can update account information accordingly	
Operation procedure:		Click on top right corner go to menu Select sign in. Log in to your account, and click on settings	
Pre-conditions:	Logged into app with an account already created	Post-conditions:	None
Inputs data	Change last name, remove all character	Expected output	Invalid name prompt

Test Case ID:	AC-13	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		Once logged in, users can update account information accordingly	
Operation procedure:		Click on top right corner go to menu Select sign in. Log in to your account, and click on settings	
Pre-conditions:	Logged into app with an account already created	Post-conditions:	None
Inputs data	Change email, remove all character	Expected output	Invalid email prompt

Account Login

Test Case ID:	AC-13	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		This is to test that once an account is successfully created, can the app authorize users to log in smoothly, if there is an error in this login step or if there is any system processing delay	
Operation procedure:		Go to installed apps on the device Find Priceline app icon and click Go to top right corner and select log in Input data according to input data box below	
Pre-conditions:	At the app login page and ready for login	Post-conditions:	None
Inputs data	Input email address only without providing a password	Expected output	Invalid password prompt

Test Case ID:	AC-14	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		This is to test that once an account is successfully created, can the app authorize users to log in smoothly, if there is an error in this login step or if there is any system processing delay	
Operation procedure:		Go to installed apps on the device Find Priceline app icon and click Go to top right corner and select log in Input data according to input data box below	
Pre-conditions:	At the app login page and ready for login	Post-conditions:	None
Inputs data	Input password only without providing an email	Expected output	Invalid email input prompt

Test Case ID:	AC-15	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		This is to test that once an account is successfully created, can the app authorize users to log in smoothly, if there is an error in this login step or if there is any system processing delay	
Operation procedure:		Go to installed apps on the device Find Priceline app icon and click Go to top right corner and select log in Input data according to input data box below	
Pre-conditions:	At the app login page and ready for login	Post-conditions:	None
Inputs data	Input both email and password but with an incorrect password	Expected output	Invalid password prompt

Test Case ID:	AC-16	Test Item:	Account Creation
Written By:	Nan Ding	Documented Date:	11/1/2015
Test Type:	Black Box Testing	Test Suite#:	4.1
Product Name:	Priceline	Release & Version No.:	Version 3.28.97
Test case description:		This is to test that once an account is successfully created, can the app authorize users to log in smoothly, if there is an error in this login step or if there is any system processing delay	
Operation procedure:		Go to installed apps on the device Find Priceline app icon and click Go to top right corner and select log in Input data according to input data box below	
Pre-conditions:	At the app login page and ready for login	Post-conditions:	None
Inputs data	Input both correct email and correct password	Expected output	Successful login with app main page

1.1.2 Decision Table for Priceline App Sign up

Conditions	Rules							
Provide fist name	F	T	T	T	T	T	T	T
Provide last name	T	F	T	T	T	T	T	T
Provide email address	T	T	F	T	T	T	T	T
Email address is not previously registered	T	T	T	F	T	T	T	T
Password more than 8 characters	T	T	T	T	F	T	T	T
Password has one number	T	T	T	T	T	F	T	T
Password has one special character	T	T	T	T	T	T	F	T
Actions								
Missing first name	X							
Missing last name		X						
Missing email address			X					
Duplicate email address				X				
Invalid password					X	X	X	
Registration Successful								X

4.2 Test case for hotel component

4.2.1 Equivalence Partitioning Testing Method

1.1 Hotels Check In Date Input

Test Case ID:	EP-4.2.1	Test Item:	Hotels
Written By:	Nie, Chao	Documented Date:	11/10/2015
Test Type:	Black Box Testing	Test Suite#:	4.2
Product Name:	Priceline	Release & Version No.:	3.28.97
Test case description:		In hotel search, user can click on “check in date” or “check out date” to select the time duration when they will be living in the hotel. However, there is limitation of the earliest and latest check in/out date respectively.	
Operation procedure:		Under “check in” date, select any date that’s earlier than the current date.	
Pre-conditions:	“location” is filled; “check out” date is filled; “number of rooms” is default 1	Post-conditions:	None
Inputs data	11/10/2015	Expected output	Invalid Input Prompt

Test Case ID:	EP-4.2.2	Test Item:	Hotels
Written By:	Nie, Chao	Documented Date:	11/10/2015
Test Type:	Black Box Testing	Test Suite#:	4.2
Product Name:	Priceline	Release & Version No.:	3.28.97
Test case description:		In hotel search, user can click on “check in date” or “check out date” to select the time duration when they will be living in the hotel. However, there is limitation of the earliest and latest check in/out date respectively.	
Operation procedure:		Under “check in” date, select a date that’s before the default or filled “check out” date and after the current date	
Pre-conditions:	“location” is filled; “check out” date is filled; “number of rooms” is default 1	Post-conditions:	None
Inputs data	11/12/2015	Expected output	Back to previous page with the selected date in the “check in” date placeholder

Test Case ID:	EP-4.2.3	Test Item:	Hotels
Written By:	Nie, Chao	Documented Date:	11/10/2015
Test Type:	Black Box Testing	Test Suite#:	4.2
Product Name:	Priceline	Release & Version No.:	3.28.97
Test case description:		In hotel search, user can click on “check in date” or “check out date” to select the time duration when they will be living in the hotel. However, there is limitation of the earliest and latest check in/out date respectively.	
Operation procedure:		Under “check in” date, select a date that’s after the default or filled “check out” date	
Pre-conditions:	“location” is filled; “check out” date is filled; “number of rooms” is default 1	Post-conditions:	None
Inputs data	11/15/2015	Expected output	Invalid Input Prompt

1.2 Hotel Check out Date Input

Test Case ID:	EP-4.2.4	Test Item:	Hotels
Written By:	Nie, Chao	Documented Date:	11/10/2015
Test Type:	Black Box Testing	Test Suite#:	4.2
Product Name:	Priceline	Release & Version No.:	3.28.97
Test case description:		In hotel search, user can click on “check in date” or “check out date” to select the time duration when they will be living in the hotel. However, there is limitation of the earliest and latest check in/out date respectively.	
Operation procedure:		Under “check out” date, select a date that’s before the default “check in” date	
Pre-conditions:	“location” is filled; “check in” date is filled; “number of rooms” is default 1	Post-conditions:	None
Inputs data	11/1/2015	Expected output	Invalid Input Prompt

Test Case ID:	EP-4.2.5	Test Item:	Hotels
Written By:	Nie, Chao	Documented Date:	11/10/2015
Test Type:	Black Box Testing	Test Suite#:	4.2
Product Name:	Priceline	Release & Version No.:	3.28.97
Test case description:		In hotel search, user can click on “check in date” or “check out date” to select the time duration when they will be living in the hotel. However, there is limitation of the earliest and latest check in/out date respectively.	
Operation procedure:		Under “check out” date, select a date that’s after the default “check in” date, and also after the allowed check out date	
Pre-conditions:	“location” is filled; “check in” date is filled; “number of rooms” is default 1	Post-conditions:	None
Inputs data	12/31/2015	Expected output	Invalid Input Prompt

Test Case ID:	EP-4.2.6	Test Item:	Hotels
Written By:	Nie, Chao	Documented Date:	11/10/2015
Test Type:	Black Box Testing	Test Suite#:	4.2
Product Name:	Priceline	Release & Version No.:	3.28.97
Test case description:		In hotel search, user can click on “check in date” or “check out date” to select the time duration when they will be living in the hotel. However, there is limitation of the earliest and latest check in/out date respectively.	
Operation procedure:		Under “check out” date, select a date that’s after the current date, but before the selected “check in” date	
Pre-conditions:	“location” is filled; “check in” date is filled; “number of rooms” is default 1	Post-conditions:	None
Inputs data	11/13/2015 (check in date is 11/15/2015)	Expected output	Invalid Input Prompt

Test Case ID:	EP-4.2.7	Test Item:	Hotels
Written By:	Nie, Chao	Documented Date:	11/10/2015
Test Type:	Black Box Testing	Test Suite#:	4.2
Product Name:	Priceline	Release & Version No.:	3.28.97
Test case description:		In hotel search, user can click on “check in date” or “check out date” to select the time duration when they will be living in the hotel. However, there is limitation of the earliest and latest check in/out date respectively.	
Operation procedure:		Under “check out” date, select a date that’s after the default “check in” date, and before the allowed check out date	
Pre-conditions:	“location” is filled; “check in” date is filled; “number of rooms” is default 1	Post-conditions:	None
Inputs data	11/15/2015 (check in date is 11/13/2015)	Expected output	Back to previous page with the selected date in the “check out” date placeholder

2 Category Partition Testing Method

Hotel Location Input

<table>
<tr><td>Test Case ID:</td><td>EP-4.2.8</td><td>Test Item:</td><td>Hotels</td></tr>
<tr><td>Written By:</td><td>Nie, Chao</td><td>Documented Date:</td><td>11/10/2015</td></tr>
<tr><td>Test Type:</td><td>Black Box Testing</td><td>Test Suite#:</td><td>4.2</td></tr>
<tr><td>Product Name:</td><td>Priceline</td><td>Release & Version No.:</td><td>3.28.97</td></tr>
<tr><td colspan="2">Test case description:</td><td colspan="2">In hotel search, user can click on “Location” to select the place where the hotel is located. In the placeholder it shows the city name based on the current user location. The input bar also displays the top list of the overall selection; but user can also type in the location they desire, if the name is wrong or doesn’t exist, the list becomes the closest suggestion.</td></tr>
<tr><td colspan="2">Operation procedure:</td><td colspan="2">Click on “Location” input bar, type in parameters with following specification:
Pattern size:
● Empty
● Single Character
● Many Character
● Repetitive characters</td></tr>
<tr><td>Pre-conditions:</td><td>“check in/out” date is properly filled; “number of rooms” is default 1</td><td>Post-conditions:</td><td>None</td></tr>
<tr><td>Inputs data</td><td>● Null
● H
● El Pueblo de Nuestra Señora la Reina de los Ángeles del Río de Porciúncula
● Sannnn Jooose</td><td>Expected output</td><td>Back to previous page with the selected city if input is valid; with default current location if input is invalid.</td></tr>
</table>

Test Case ID:	EP-4.2.9	Test Item:	Hotels
Written By:	Nie, Chao	Documented Date:	11/10/2015
Test Type:	Black Box Testing	Test Suite#:	4.2
Product Name:	Priceline	Release & Version No.:	3.28.97
Test case description:		In hotel search, user can click on “Location” to select the place where the hotel is located. In the placeholder it shows the city name based on the current user location. The input bar also displays the top list of the overall selection; but user can also type in the location they desire, if the name is wrong or doesn’t exist, the list becomes the closest suggestion.	
Operation procedure:		Click on “Location” input bar, type in parameters with following specification: Quoting: ● Pattern is quoted ● Pattern is not quoted	
Pre-conditions:	“check in/out” date is properly filled; “number of rooms” is default 1	Post-conditions:	None
Inputs data	“Los Angeles” Los Angeles	Expected output	Back to previous page with the selected city if input is valid; with default current location if input is invalid.

Test Case ID:	EP-4.2.10	Test Item:	Hotels
Written By:	Nie, Chao	Documented Date:	11/10/2015
Test Type:	Black Box Testing	Test Suite#:	4.2
Product Name:	Priceline	Release & Version No.:	3.28.97
Test case description:		In hotel search, user can click on “Location” to select the place where the hotel is located. In the placeholder it shows the city name based on the current user location. The input bar also displays the top list of the overall selection; but user can also type in the location they desire, if the name is wrong or doesn’t exist, the list becomes the closest suggestion.	
Operation procedure:		Click on “Location” input bar, type in parameters with following specification: Special Characters: ● Pattern with special characters ● Pattern with no special characters	
Pre-conditions:	“check in/out” date is properly filled; “number of rooms” is default 1	Post-conditions:	None
Inputs data	San Fran%cisco S#an Francisco San Francisco	Expected output	Back to previous page with the selected city if input is valid; with default current location if input is invalid.

3 Decision Table Testing Method

1 Decision Table test models

Test Case ID:	EP-4.2.11	Test Item:	Hotels
Written By:	Nie, Chao	Documented Date:	11/10/2015
Test Type:	Black Box Testing	Test Suite#:	4.2
Product Name:	Priceline	Release & Version No.:	3.28.97
Test case description:		In Hotel search, there are four inputs in total, “Location”, “Check in/out” date, and “number of rooms”. The last input has strong constraint that only allows user to tap on “-“or “+” mark to adjust the number, and the range is from 1 to 9; the number will stop incrementing or decreasing when the number reaches the limit.	
Operation procedure:		Condition for Location will be based on the selected city name, which will return the default current location of user if the name does not appear on the available list.	
Pre-conditions:	Application successfully launched	Post-conditions:	None
Inputs data	Three inputs based on condition table	Expected output	Display Available Hotels if Inputs are all valid; display invalid input prompt if any input is invalid.

Test Case ID:	EP-4.2.12	Test Item:	Hotels
Written By:	Nie, Chao	Documented Date:	11/10/2015
Test Type:	Black Box Testing	Test Suite#:	4.2
Product Name:	Priceline	Release & Version No.:	3.28.97
Test case description:		In Hotel search, there are four inputs in total, “Location”, “Check in/out” date, and “number of rooms”. The last input has strong constraint that only allows user to tap on “-“or “+” mark to adjust the number, and the range is from 1 to 9; the number will stop incrementing or decreasing when the number reaches the limit.	
Operation procedure:		Condition for “Check in” date has three regions: ● Before current date ● Valid input ● After allowed long term check in date	
Pre-conditions:	Application successfully launched	Post-conditions:	None
Inputs data	Three inputs based on condition table	Expected output	Display Available Hotels if Inputs are all valid; display invalid input prompt if any input is invalid.

Test Case ID:	EP-4.2.13	Test Item:	Hotels
Written By:	Nie, Chao	Documented Date:	11/10/2015
Test Type:	Black Box Testing	Test Suite#:	4.2
Product Name:	Priceline	Release & Version No.:	3.28.97
Test case description:		In Hotel search, there are four inputs in total, “Location”, “Check in/out” date, and “number of rooms”. The last input has strong constraint that only allows user to tap on “-“or “+” mark to adjust the number, and the range is from 1 to 9; the number will stop incrementing or decreasing when the number reaches the limit.	
Operation procedure:		Condition for “Check out” date has three regions: ● Before check in date ● Valid input ● Check out date is too long than allowed	
Pre-conditions:	Application successfully launched	Post-conditions:	None
Inputs data	Three inputs based on condition table	Expected output	Display Available Hotels if Inputs are all valid; display invalid input prompt if any input is invalid.

4.2.2 Decision Table for Hotel Search

Conditions	Rules							
“Check in” date is before current date	T	F	F	F	F	F	F	T
“Check in” date is after current date	F	T	T	T	T	T	T	F
“Check out” date is before current date	F	T	F	F	F	F	T	F
“Check out” date is before check in date	F	F	T	F	F	F	F	T
“Check out” date is after check in date	F	F	F	T	F	T	F	F
“Check out” date is after allowed date	F	F	F	F	T	F	F	F
“Location” is valid	F	F	F	T	F	F	F	F
Actions								
Display default current location						X		
Display invalid check in/out date input prompt	X	X	X		X			
Return to search page with all valid inputs				X				
Impossible							X	X

4.3 Test Case for Flight Component

As for flight component part, the black box testing method is implemented, to be more specific, in this component testing, we mainly utilize equivalence partition and decision table testing methods. The following tables describe the detailed test cases for above methods into three subcategories for flight component: search, filter, select.

4.3.1 Equivalence Partition Test for Flight Component

Equivalence Partition ID	Partition Category
P1	Select Round Trip and Departure Airport
P2	Select Arrival Airport
P3	Select Departure Dates
P4	Select Number Of Passengers
P5	Select Non-stop Preferred
P6	Search Flights
P7	Select One Way Trip and Select Arrival Airport
P8	Select Departure Dates
P9	Select Number Of Passengers
P10	Select Non-stop Preferred
P11	Search Flights

Test Case ID:	EP-4.3.1	Test Item:	Round Trip and departure Selection
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Flight Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if round trip selection is working or not.		
Operation Procedure:	● Open priceline app ● Tap Flight in the main page ● Tap round trip and select departure airport		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	search a departure airport and add it to departure field	Expected Output Data:	The selected departure location added to the desired field
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.3.2	Test Item:	Arrival Airport
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Flight Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if arrival selection is working or not.		
Operation Procedure:	● Open priceline app ● Tap Flight in the main page ● After tapping round trip and select departure airport, tapping arrival airport to select desired airport		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	search an arrival airport and add it to arrival field	Expected Output Data:	The selected arrival location added to the desired field
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.3.3	Test Item:	Departure Date
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Flight Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if departure date selection is working or not.		
Operation Procedure:	● Open priceline app ● Tap Flight in the main page ● After tapping round trip and select departure airport, tapping arrival airport to select desired airport ● Tap departure date and select desired date		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	search desired date and add it on departure date field	Expected Output Data:	The selected departure date added to the desired field
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.3.4	Test Item:	Passengers
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Flight Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if number of passengers selection is working or not.		
Operation Procedure:	● Open priceline app ● Tap Flight in the main page ● After tapping round trip and select departure airport, tapping arrival airport to select desired airport ● Tap departure date and select desired date ● Add number of passengers as needed		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	Add number of passengers as needed	Expected Output Data:	The added number is shown to the desired field
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.3.5	Test Item:	Non-stop or not
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Flight Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if non-stop selection is working or not.		
Operation Procedure:	● Open priceline app ● Tap Flight in the main page ● After tapping round trip and select departure airport, tapping arrival airport to select desired airport ● Tap departure date and select desired date ● Add number of passengers in this flight ● check if need non-stop or not		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	check non-stop	Expected Output Data:	the result showing all non-stop flights
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.3.6	Test Item:	Search flight
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Flight Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if selections are applied to the result.		
Operation Procedure:	● Open priceline app ● Tap Flight in the main page ● After tapping round trip and select departure airport, tapping arrival airport to select desired airport ● Tap departure date and select desired date ● Add number of passengers ● Check if preferred to non-stop ● Tap search flight		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:		Expected Output Data:	All conditions are included in the results
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.3.7	Test Item:	One-way Trip and departure Selection
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Flight Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if one-way trip selection is working or not.		
Operation Procedure:	● Open priceline app ● Tap Flight in the main page ● Tap one-way trip and select departure airport		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	search a departure airport and add it to departure field	Expected Output Data:	The selected departure location added to the desired field
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.3.8	Test Item:	Arrival Airport
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Flight Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if arrival selection is working or not.		
Operation Procedure:	● Open priceline app ● Tap Flight in the main page ● After tapping one-way trip and select departure airport, tapping arrival airport to select desired airport		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	search an arrival airport and add it to arrival field	Expected Output Data:	The selected arrival location added to the desired field
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.3.8	Test Item:	Departure Date
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Flight Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if departure date selection is working or not for one-way trip.		
Operation Procedure:	● Open priceline app ● Tap Flight in the main page ● After tapping one-way trip and select departure airport, tapping arrival airport to select desired airport ● Tap departure date and select desired date		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	search desired date and add it on departure date field	Expected Output Data:	The selected departure date added to the desired field
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.3.9	Test Item:	Passengers
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Flight Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if number of passengers selection is working or not in a one-way trip.		
Operation Procedure:	● Open priceline app ● Tap Flight in the main page ● After tapping one-way trip and select departure airport, tapping arrival airport to select desired airport ● Tap departure date and select desired date ● Add number of passengers as needed		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	Add number of passengers as needed	Expected Output Data:	The added number is shown to the desired field
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.3.10	Test Item:	Non-stop or not
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Flight Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if non-stop selection is working or not for a one-way trip.		
Operation Procedure:	● Open priceline app ● Tap Flight in the main page ● After tapping one-way trip and select departure airport, tapping arrival airport to select desired airport ● Tap departure date and select desired date ● Add number of passengers in this flight ● check if need non-stop or not		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	check non-stop	Expected Output Data:	the result showing all non-stop flights
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.3.11	Test Item:	Search flight
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Flight Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if selections are applied to the result for a one-way trip.		
Operation Procedure:	● Open priceline app ● Tap Flight in the main page ● After tapping one-way trip and select departure airport, tapping arrival airport to select desired airport ● Tap departure date and select desired date ● Add number of passengers ● Check if preferred non-stop ● Tap search flight		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:		Expected Output Data:	All conditions are covered in the output result
Test Script:	Follow Test Software Requirements.		

4.3.2 Decision Table Testing for Flight Component

Condition ID	Description
C1	Search for departure airport
C2	Search for arrival airport
C3	Search for departure date
C4	Search for arrival date
C5	Add number of passengers
C6	Search a blank entry
C7	Check nonstop preferred
C8	Add invalid location
C9	Add invalid date

Action ID	Description
A1	Show the result
A2	No output displayed
A3	valid input added to the desired field
A4	invalid input didn’t add to the related field
A5	input is not displayed
A6	input displayed

Conditions	1	2	3	4	5	6	7	8	9
C1	T	T	T	T	T	F	T	F	F
C2	T	T	T	T	T	F	T	F	F
C3	F	T	T	F	F	T	T	T	T
C4	F	T	T	T	F	F	T	T	T
C5	T	T	T	T	T	T	T	T	T
C6	T	F	T	F	F	F	T	F	F
C7	T	T	T	T	T	F	F	F	F
C8	T	T	T	F	T	F	F	T	F
C9	T	F	T	T	T	F	F	T	F
Actions									
A1	F	T	F	T	T		F	F	
A2	T	F		F		T	T	F	T
A3	T	T		F	T		F		T
A4	F		F	T		F	T	F	
A5	T			T				T	
A6		F				F			

4.4 Test Case for Rental Car Component

The black box testing method is applied in rental cars component. In other words, we mainly utilize equivalence partition and decision based testing methods. The following tables describe the detailed test cases for above methods into three subcategories for flight component: search, filter, select.

4.4.1 Equivalence Partition Test for Rental Cars Component

Equivalence Partition ID	Partition Category
P1	Select Pick-Up Location
P2	Select Pick-Up Date
P3	Select Pick-Up Time
P4	Select Drop-Off Location
P5	Select Drop-Off Date
P6	Select Drop-Off Time
P7	Search Cars

Test Case ID:	EP-4.4.1	Test Item:	Pick-Up Location Select
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Car Rental Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if pick-up location selection is working or not.		
Operation Procedure:	● Open priceline app ● Tap Rental Cars in the main page ● Tap Pick-Up Location		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	Search desired pick-up location and add it on pick-up location field	Expected Output Data:	The added location name is shown to the desired field
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.4.2	Test Item:	Pick-Up Date Select
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Car Rental Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if pick-up date selection is working or not.		
Operation Procedure:	● Open priceline app ● Tap Rental Cars in the main page ● Tap Pick-Up Date		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	Search desired pick-up date and add it on pick-up date field	Expected Output Data:	The added date is shown to the desired field
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.4.3	Test Item:	Pick-Up Time Select
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Car Rental Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if pick-up time selection is working or not.		
Operation Procedure:	● Open priceline app ● Tap Rental Cars in the main page ● Tap Pick-Up Time		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	Search desired pick-up time and add it on pick-up time field	Expected Output Data:	The added time is shown to the desired field
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.4.4	Test Item:	Drop-Off Location Select
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Car Rental Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if drop-off location selection is working or not.		
Operation Procedure:	● Open priceline app ● Tap Rental Cars in the main page ● Tap Drop-Off Location		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	Search desired drop-off location and add it on drop-off location field	Expected Output Data:	The added location name is shown to the desired field
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.4.5	Test Item:	Drop-Off Date Select
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Car Rental Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if drop-off date selection is working or not.		
Operation Procedure:	● Open priceline app ● Tap Rental Cars in the main page ● Tap Drop-Off Date		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	Search desired drop-off date and add it on drop-off date field	Expected Output Data:	The added date is shown to the desired field
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.4.6	Test Item:	Drop-Off Time Select
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Car Rental Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if drop-off time selection is working or not.		
Operation Procedure:	● Open priceline app ● Tap Rental Cars in the main page ● Tap Drop-Off Time		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	Search desired drop-off time and add it on drop-off time field	Expected Output Data:	The added time is shown to the desired field
Test Script:	Follow Test Software Requirements.		

Test Case ID:	EP-4.4.7	Test Item:	Search Cars
Tester Name:	Xiaorui Wang	Documented Date:	11/1/15
Test Type:	Black Box Testing	Test Suite:	Car Rental Component
Product Name:	Priceline	Version #:	3.28.97
Test Case Description:	This test case is used to check if search cars is working or not.		
Operation Procedure:	● Open priceline app ● Tap Rental Cars in the main page ● Tap Search Cars		
Pre-Condition:	Good	Post-Condition:	Good
Input Data:	Tap search cars button	Expected Output Data:	All conditions are applied in result.
Test Script:	Follow Test Software Requirements.		

4.4.2 Decision Table Testing for Rental Cars Component

Condition ID	Description
C1	Search for pick-up location
C2	Search for pick-up date
C3	Search for pick-up time
C4	Search for drop-off location
C5	Search for drop-off date
C6	Search for drop-off time
C7	Add invalid location
C8	Add invalid date
C9	Add invalid time

Action ID	Description
A1	Show the result
A2	No output displayed
A3	Valid input added to the desired field
A4	Invalid input didn’t add to the related field
A5	Input is not displayed
A6	Input displayed

Conditions	1	2	3	4	5	6	7	8	9
C1	T	T	F	T	F	F	T		T
C2	F	F	T	F	F	T	T	F	T
C3	T		F	F	T	T	F	F	F
C4	T	F	F	T		T	F		T
C5	F	T	T	F	F	T	F	F	T
C6	F	T		T		F	F	T	T
C7	T	T	F	F	T	F	F	T	F
C8	T	T	F	T	F	T		F	F
C9	F	T	F	F	T	T	F	T	F
Actions									
A1	T	F		F		T	F	F	F
A2	T	F	F			T	T	F	T
A3	T		F	F	T	F	F	F	
A4	F		T		F	T		T	
A5	T			T		F	F		
A6	T		F		T			T	

5. Summary

5.1 Summary Test Method

In this project, we mainly implemented equivalence partitioning, and decision table testing methods from black box test for Priceline mobile application. We mainly focused on four components in this application: Account component, Hotel component, Flight component and Rental car component and services associated with them. The following table describes the summary of the test method for each component.

Testing Type: Black Box			Product Name: Priceline	
Testing Method	Testing Component	Test Case	Tester Name	Test Date
Equivalence Partitioning	Account Component	Exploring if different inputs are given, what are the responses of this component	Nan Ding	11/1/2015
Equivalence Partitioning	Hotel Component	Exploring if different inputs are given, what are the responses of this component	Chao Nie	11/1/2015
Equivalence Partitioning	Flights Component	Explore if different inputs are given, what are the responses of this component	Xiaorui Wang	11/1/2015
Equivalence Partitioning	Rental Car Component	Exploring if different inputs are given, what are the responses of this	Xiaorui Wang	11/1/2015

		component		
Decision Table Testing	Account Component	Investigate when the various combination conditions are given, the results of the components' responses.	Nan Ding	11/1/2015
Decision Table Testing	Hotel Component	Investigate when the various combination conditions are given, the results of the components' responses.	Chao Nie	11/1/2015
Decision Table Testing	Flights Component	Investigate when the various combination conditions are given, the results of the components' responses.	Xiaorui Wang	11/1/2015
Decision Table Testing	Rental Car Component	Investigate when the various combination conditions are given, the results of the components' responses.	Xiaorui Wang	11/1/2015

5.2 Summary Test Complexity

Based on the number of function services and testing case number for each component, which defined in previous chapter, we can get the total number of test case. As you can see in the following table.

Components	Total Number of Test Cases	
Installation Testing	6	
	Equivalence Partitioning	Decision Table test
Account	16	8
Hotel	13	8
Flight	11	9
Rental car	7	9
Total:	47	34

6. References

1. Introduction to Software Testing Presentation, class PPT, Dr Jerry Gao

2. Priceline app downloaded from:
iOS: https://itunes.apple.com/us/app/priceline.com-hotels-flights/id336381998?mt=8
Android: https://play.google.com/store/apps/details?id=com.priceline.android.negotiator

3. Gao, J., & Tao, C.Q. (2014) Modeling Mobile Application Test Platform and Environment: testing criteria and complexity.

4. Gao, J., Tsao, H.-S. & Wu, Y. (2003) Testing and Quality Assurance for Component-based Software. Artech House, Boston.