Guest House Management

Test report

Version1.0

04/08/2017

VERSION HISTORY

|  |  |  |
| --- | --- | --- |
| **Version #** | **Implemented By** | **Revision Date** |
| 1.0 | Group-11 (Harshit Singh and Nikhil Bhamare) | 04/08/2017 |

**Table of Contents**

[**1.0 INTRODUCTION**](#_1fob9te) **4**

[1.1 Purpose](#_3znysh7) 5

[**2.0 TEST PLAN**](#_2et92p0) **5**

[**3.0 TEST CASE SELECTION**](#_tyjcwt) **5**

[**4.0 Test Results**](#_3dy6vkm) **6**

[4.1 FUNCTIONALITY 3: ADDING AND DELETING A GUEST](#_21rpq8afoa1n) 6

[4.2 FUNCTIONALITY 2: ADDING AND DELETING A ROOM](#_dkbuzl9zzsc) 6

[4.3 FUNCTIONALITY 3: BOOKING A ROOM](#_u9yooc8xb6ur) 6

[4.4 FUNCTIONALITY 4: BOOKING A ROOM AFTER ALL ROOMS ARE OCCUPIED](#_m96t2in4sj9e) 8

[**5.0 HEURISTICS TESTING**](#_1t3h5sf) **9**

[**6.0 Test Instances**](#_2s8eyo1) **10**

[Resolved Test Incidents](#_17dp8vu) 10

[Unresolved Test Incidents](#_3rdcrjn) 10

# 

# 1.0 INTRODUCTION

## 1.1 Purpose

This Guest House Management Software Test Report provides a summary of the results of test performed as outlined within this document.

# 2.0 TEST PLAN

1. Goal 1: To check the functionality of the administrator to add and delete a user.
2. Goal 2: To check the functionality of the administrator to add and delete a room.
3. Goal 3: To check the functionality of the guest user to book a room.
4. Goal 4: To test the User Interface design.

# 3.0 TEST CASE SELECTION

1.) Functionality 1: Administrator adds a user.

This functionality is Important because there is no other way the user can interact with the system as there is no option for the guest user to sign up.

The test values for this test case are chosen according to the text fields provided by the developers.

2.) Functionality 2: Administrator adds a room.

This functionality is important because the number of the rooms have to increase as the number of people increase.

The test values for this test case are chosen as per the text fields and scroll down options provided by the developers which are 1) Cost 2) Registered or Unregistered or Not available.

3.) Functionality 3: Guest User Books a Room.

This is the most important functionality as the complete web app revolves around booking rooms in the guest house.

The test values for this test case are chosen as per the text fields and scroll down options provided by the developers which are 1) Start Date 2) End Date 3) Veg or Non-veg.

4.) Functionality 4: Book a room after all rooms are occupied.

# 4.0 Test Results

## 4.1 FUNCTIONALITY 3: ADDING AND DELETING A GUEST

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1. Test Case ID | 1. Intended Result | 1. Intended Result | 1. Pass/Fail | 1. Severity of Defect [Low/ Medium/ High] | 1. Summary of Defect | 1. Comments |
| 1. T1.1 | The password of users should not be visible to anyone | Password is visible | Fail | High | Passwords should not be visible | Hide or encrypt the password |

## 4.2 FUNCTIONALITY 2: ADDING AND DELETING A ROOM

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Intended Result | Intended Result | Pass/Fail | Severity of Defect [Low/ Medium/ High] | Summary of Defect | Comments |
| 1. T2.1 | Administrator should be able to add rooms | Able to add rooms | Pass |  |  | There is no use of adding reserved rooms if one cannot book them |
| 1. T2.2 | Administrator should be able to delete a room | Deletes even the rooms which have been booked | Fail | Medium | Should not be able to delete a booked room | The administrator must be informed that a particular room has already been booked |

## 

## 4.3 FUNCTIONALITY 3: BOOKING A ROOM

|  |  |  |
| --- | --- | --- |
| **Test Case ID** | **Start Date** | **End Date** |
| T3.1 | Past Date | Past Date |
| T3.2 | Past Date | Present Date |
| T3.3 | Past Date | Future Date |
| T3.4 | Present Date | Past Date |
| T3.5 | Present Date | Present Date |
| T3.6 | Present Date | Future Date |
| T3.7 | Future Date | Past Date |
| T3.8 | Future Date | Present Date |
| T3.9 | Future Date | Future Date |

## 

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Intended Result | Intended Result | Pass/Fail | Severity of Defect [Low/ Medium/ High] | Summary of Defect | Comments |
| T3.1 | Should not Book Past Date | Cannot book Past Date | Pass |  |  |  |
| T3.2 | Should not Book Past Date | Cannot book Past Date | Pass |  |  |  |
| T3.3 | Should not Book Past Date | Cannot book Past Date | Pass |  |  |  |
| T3.4 | Should not Book Past Date | Cannot book Past Date | Pass |  |  |  |
| T3.5 | Should be able to book | Able to book | Pass |  |  |  |
| T3.6 | Should be able to book | Able to book | Pass |  |  |  |
| T3.7 | Should Check that the start Date should be before end date | Checking start Date should be before end date | Pass |  |  |  |
| T3.8 | Should check that the start Date should be before end date | Checking the start Date should be before end date | Pass |  |  |  |
| T3.9 | Should be able to book | Able to book | Pass |  |  |  |

## 

## 4.4 FUNCTIONALITY 4: BOOKING A ROOM AFTER ALL ROOMS ARE OCCUPIED

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1. Test Case ID | 1. Intended Result | 1. Intended Result | 1. Pass/Fail | 1. Severity of Defect [Low/ Medium/ High] | 1. Summary of Defect | 1. Comments |
| 1. T4.1 | Should not be able to book | Unable to book a room | Pass |  |  |  |
| 1. T4.2 | After a room becomes available WL should automatically get to CNF status | Stays WL. | Fail | High | Stays WL.  No option to confirm booking of a WL.  WL never converts to CNF | There should be some function so that WL automatically gets confirmed as soon as room becomes available |
| 1. T4.3 | A user should not be able book room until all WL's have been confirmed for those particular ranges of dates | User is able to book rooms | Fail | High | No option to confirm booking of a WL.  WL never converts to CNF | There should be some function so that WL automatically gets confirmed as soon as room becomes available |

# 

# 5.0 HEURISTICS TESTING

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Issue | Heuristics | Severity |
| UT1 | Login form difficult to locate due to heavy use of the cover image | General UI problem | Medium |
| UT2 | Unclear area of clicking after the user has logged in. We have to scroll down to see the text fields | Flexibility and efficiency in use | Medium |
| UT3 | The user is not sure where to click in order to select the start and end dates of his stay at the guest house. | General UI problem | Medium |
| UT4 | When the administrator adds a user the page displays prior entries along with the password. | General UI problem | high |
| UT5 | The user is not always sure if what he sees on the page is all that is to see. | Flexibility and efficiency in use | low |
| UT6 | No details of rooms shown | Visibility of system status | Medium |
| UT7 | When there is a cancellation of the room the people in waiting list do not get the priority , a new entry can get the room | Flexibility and efficiency of use | high |
| UT8 | The search button does not do what it suggests, instead automatically books the room | Consistency and Standards | high |
| UT9 | In the add a user functionality the 2nd option is given as USER which should be GUEST | Consistency and Standards | Low |
| UT10 | The division for giving start date and end date uses the minimalistic logos of the Login division | Aesthetics and minimalistic design | Low |

# 

# 6.0 Test Instances

## Resolved Test Incidents

|  |  |  |
| --- | --- | --- |
| Test Case ID | Summary of Defect | Resolution Taken |
|  |  |  |
|  |  |  |
|  |  |  |

## Unresolved Test Incidents

|  |  |  |
| --- | --- | --- |
| Test Case ID | Summary of Defect | Why not resolved |
|  |  |  |
|  |  |  |
|  |  |  |