GuessBid

Testing

Milica Shulevska
POLITECNICO DI MILANO | SOFTWARE ENGINEERING 2
30.06.2015

CONTENTS

1 INTRODUCTION

1.1 PURPOSE

This document describes the general test cases for the project GuessBid, developed under the course of Software Engineering 2 at Politecnico di Milano. The project is created by the student Milica Shulevska and the Git repository of the project is found at this url: https://github.com/mshulevska/guess-bid-se2015.

1.2 SCOPE

The scope of this document is to test the application developed. By analyzing the documents (Requirements Analysis and Specification and Design documents), these are functionalities that the system should have:

The software will have the following goals:

[G1] Users can register on the system.

[G2] Users can create an auction.

[G3] Users can browse existing auctions.

[G4] Users can bid on existing auctions.

[G5] Users can provide more than one bid before closing time of the auction.

[G6] User with a lowest unique bid is the winner.

[G7] Issuing bids has a cost of 2 per bid.

[G8] Users are informed by the system about the current status of his/her biddings on the open auctions at each bid.

[G9] Users are notified about the outcome of the auctions they have a connection to.

[G10] Users can login.

[G11] Users can logout.

1.3 REFERENCES FOR AVAILABLE MATERIALS

In order to test the application the following tools have been used:

- RASD: Requirements Analysis and Specification Document
- Design Document

- Deploy of the web application

1.4 Overview

The testing cases will be defined according to the following characteristics:

- Objective: The main objective of the test case
- Environment: The page in which the test case is initiated
- Input: The set of data used to validate this test case
- Expected output: After the execution of the test case, this describes the outcome that should be obtained
- Final result: After the execution of the test case, the final page in which the user is redirected
- Observations: The errors observed or the inconsistencies regarding the business logic

2 Test Cases

This section provides the main test cases for the functionalities of the GuessBid project.

2.1 Register

Register	
Objective	Register onto the system
Environment	Register page
Input	Email, password, name, surname, date of birth, address and phone number.
Expected output	The register completes and the user is registered on the system.
Obtained output	Output same as the expected output
Final result	The system is redirected to the homepage where the login needs to be completed.

Observations	If one leaves some fields empty, the system is giving notifications that the fields cannot be blank.

2.2 LOGIN

Login	
Objective	Accessing the system as a user.
Environment	Homepage of the system
Input	Data existing in the database (user's email and the password)
Expected output	The login completes and the user starts browsing from the homepage
Obtained output	Output same as the expected output
Final result	The system is redirected to the home of the user. You can see the public auctions and the auctions the user created.
Observations	If one enters incorrect data, the system reacts in a correct way by informing about the errors and gives a link to the homepage for trying again.

2.3 ASSIGNING VIRTUAL CREDIT

Virtual Credit	
Objective	Assigning virtual credit to the user
Environment	Profile page of the user
Input	The user needs to be logged in.
Expected output	The virtual credit is assigned.
Obtained output	Output same as the expected output
Final result	The user has an initial credit of 100.
Observations	/

2.4 Browse Auctions

Browse auctions	
Objective	Browsing the auctions created by the users.
Environment	Auction feed of the user.
Input	/
Expected output	The system provides the active auctions.
Obtained output	Output same as the expected output
Final result	Auctions are visible to the user.
Observations	/

2.5 Create Auction

Create Auction	
Objective	Creating and managing of an auction
Environment	Auction feed of the system.
Input	Title, description, start date and end date.
Expected output	The auction is created with the given parameters.
Obtained output	Output same as the expected output
Final result	The system is redirected to the auction feed page.
Observations	Some errors in the design. Nothing serious.

2.6 Bid on Auctions

Bid	
Objective	Bid on an existing auction
Environment	Auction page
Input	Bid amount
Expected output	The bid is registered.
Obtained output	Output same as the expected output
Final result	There is a message appearing, saying that the bid has been successful.

Observations	If one enters letters or not acceptable amount, the system is giving errors.

2.7 NOTIFICATIONS

Notification	
Objective	The user is notified about the outcome of the auction.
Environment	Home of the system
Input	/
Expected output	A notification about the outcome of the auction.
Obtained output	/
Final result	There isn't a classic notification, but when the auction ends, the winner is shown on the auction page.
Observations	/

3 CONCLUSION

The application does not give a notification to the user.Instead, there is just a message on the auction page that the auction is closed and who is the winner.

Overall, there are some issues that need to be taken care of and maybe some better notification system can be implemented.