

School of Electrical Engineering and Computer Science  
The University of Newcastle  
SENG2050 Introduction to Web Engineering  
Assignment 2 (15%) – Semester 1, 2022

Assignment 2 Specification

## 1. Introduction

The main objective of this assignment is to implement an online game – Avoid the Secrete Number. Following are the game rules.

At the start of the game, the system will randomly generate a secret number which is between 1 and 11 (including 1 and 11). In each round, the player is asked to nominate some numbers (these numbers are between 1 and 11 inclusive), the aim of the player is to hopefully not reveal the secrete number.

There are totally 4 rounds, the player is asked to nominate 1 number for the first round, 2 numbers for the second round, 3 numbers in the third round, 4 numbers for round 4. If the player has unfortunately revealed the secrete number at any round, then the game terminates. If the player has not revealed the secrete number at the end of round 4, then the player will be rewarded with a monetary prize (the secrete number x \$100). At the end of each round, the game will offer the player a monetary amount to convince them to quit the game at that point. The amount offered is based on the following formula: the smallest un-revealed number x \$100.

The first page that is presented to the player will allow them to start a new game or load an existing game (see below for more details). When a player starts a new game, the website will indicate that the secrete number is generated and ask player to input a number (for round 1 or numbers for the other rounds). After these numbers been submitted, the system will check if the secrete number has been revealed, if yes, the game terminates. Otherwise, the player will be shown a page displaying the amount the game is offering together with all the unrevealed numbers. There should be an option of “Accept” or “Continue” on the page. If the player chooses “Accept”, the game finishes. If the player chooses “Continue”, they will move onto the next round.

After selecting “Continue” the user will be given the option to save their game. The game’s state should be saved on the server (to prevent cheating) and referenced by a username provided by the user. Each user have a single saved game, i.e. starting a new game with an existing username will replace the save game data. Lastly, saving the game should end the current game and loading it should clear the saved data. This is to prevent a user from reloading their game if they are unhappy with their selection.

Remembering all the numbers that has been nominated could be difficult, also some typos could be made. So validations (both on client side and server side) should be implemented to check if

- Input is valid,
- Inputted number has already been nominated in previous round,
- There are duplicates in the inputted numbers in the current round.

To make this game practical, you will need to make sure:

- It is multi-user safe
- Clicking the browser's "refresh" button will not result in any inconsistencies such as a number been revealed twice, the user is asked to rethink the bank's offer, etc.
- Clicking the browser's "back" button will not roll back the game.
- You don't need to implement authentication (i.e. username/password).

In this assignment, you should use JSPs and Java Servlets together with JavaBeans. It is up to you how you implement session tracking.

## 2. Submission

You should create a zip file called cXXXXXXXX\_assignment2 (where XXXXXXXX is your student number) with the following files and submit it through blackboard:

1. Assignment Cover Sheet.
2. A brief explanation of your program (readme.txt).
  - The application's structure, i.e. relationships among objects etc.
  - What is the purpose of each of your objects?
  - How did you implement session tracking?
  - How did you implement game saving?
  - The URL the marker needs to visit to start the application.
3. Your entire application. This should include all the files that contribute to your web pages, including HTML, java source code, and images, etc.

Your application will be tested in Tomcat 9 and compiled with JDK 11, please use relative addresses (no host name) in your program rather than absolute addresses. Your application should work when deployed to Tomcat's webapps directory with the name cXXXXXXXX\_assignment2 (where XXXXXXXX is your student number) i.e. apache-tomcat/webapps/cXXXXXXXX\_assignment2/ and [http://localhost:8080/cXXXXXXXX\\_assignment2/](http://localhost:8080/cXXXXXXXX_assignment2/)

## 3. Marking Guidelines

The following is a general guide on how the marks are allocated to each part of the assignment and some common mistakes where you might lose marks. Total marks: 90

HTML and CSS: 10

- Is your HTML syntax valid (use a validator)?
- Do your tags make semantic sense?
- Are you using tags just for their visual effect?
- Are you using CSS to advise the browser about visual formatting?
- Are the inputs validated before submitting to the server?

JSP: 20

- Is your JSP syntax valid?
- Are you overloading your JSP with too much Java code?
- Is your JSP nicely formatted and commented?
- Can your JSP handle errors properly?
- Do you have server-side validation, does it work properly?

Java Beans: 20

- Does your "bean" conform to the Java Beans standard?
- Is your bean shared among the pages correctly?
- Does your bean contain HTML output?

- Is your Java nicely formatted and commented?

Requirements: 30

- Does your assignment meet the specifications above?

Session Tracking and Thread Safe: 10

Usability: 10