# Software Test Report

## CS 3300-002 Spring 2025

Jack Dodge, West Manison, George Morales, Sebastian Sanchez

## Project Title:

John Doe Portfolio

## Test Report Scope:

Individual tests were done by each tester and feedback was directly given. Testing was done incrementally as the deadlines were met. Full team testing was done when the website hit key milestones.

## Test Team:

Each member was a tester for another. And all updates to the webpage were inspected by each member for quality assurance.

Test Schedule:

Below is a screen capture of our updated Requirements Spreadsheet. This sheet condensed our testing requirements as well as our long-term schedule into one document.

A spreadsheet with numbers and text

AI-generated content may be incorrect.

## Test Description:

Upon completion of the Blackjack game, it was submitted for team review. The expected outcomes were taken, and the program was run multiple times. The expected outcomes were if the player’s hand size is greater than 21, they lose, if the dealer’s hand size is greater than 21, he loses, if the player’s hand size is 21 or less and larger than the dealer then they win. Upon repeated attempts running the game, it was discovered that the dealer was winning even if his hand went above 21, the bug was quickly pinpointed and fixed. By noting what the expected results were and which of them were not being reached the possible problems were tremendously decreased and the process of bug fixing became easy. The discovered problem was that the dealer had a gun (if statement) and was threatening the player. The gun was summarily taken away (if turned into elif statement) and everything worked fine.

## Final product features and quality:

The result of our efforts is a website with multiple pages, each containing a different demonstration of someone’s coding abilities. It was built with expandability in mind and is easily capable of scaling into a larger form with just a few changes.   
7. Test Flags and Risk Management The biggest flaw of the website was the fact that python programs were not as easily embeddable into the website as we expected which caused large changes to be necessary to properly fit.

## Flags from individual games:

**Blackjack**:

The font keeps going crooked, no risk.

**Snake**:

Trying to start the game can sometimes lead to a black screen and a stuck cursor. Moving the cursor off the game fixes it, small risk.

Game skips inputs during certain frames of the gameplay, high risk of raging, little risk otherwise

**Tic-Tac-Toe**:

No way to stop game once you have started, no risk.

**Encoder**:

No way to Unencode after encoding, no risk.

## Lessons Learned from Testing:

A couple of major lessons we learned from our software testing include:

* The more eyes on a piece of code, the easier it is to find problems with it.
* Communication is key to test in-development software.