

Peyton Wolf

4/20/2025

Cst-250 Programming in C# ||

Mark Smithers

Grand Canyon University

Milestone 5

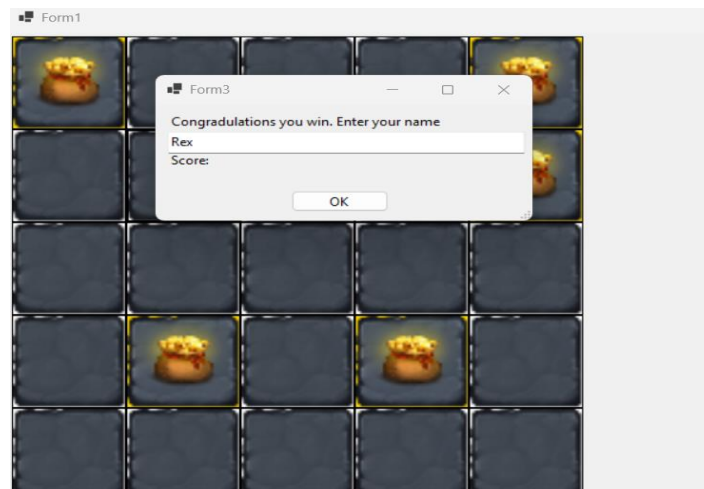
### **Loom Screencast Videos**

1. <https://www.loom.com/share/8fbc9288816547659bc19619bc49f886?sid=3b46196f-b02a-4c60-8259-8d2b6ff3f3cc>
2. <https://www.loom.com/share/ff6acaaf75164583a2fc74437b140d37?sid=039bd700-4f7f-48f2-a2d4-c8c7978d9b40>

### **GitHub Link**

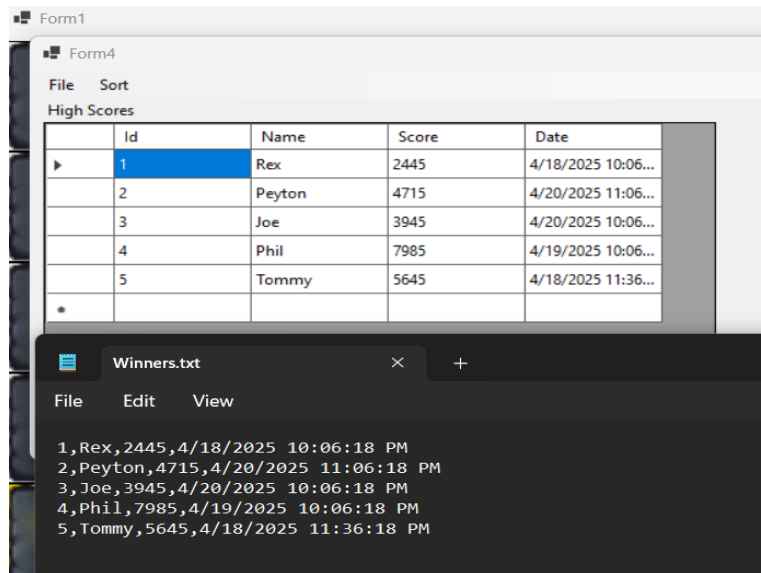
<https://github.com/KnoxHighStax/CST250/tree/main/Milestone5>

### **Form3 Capturing Winner**



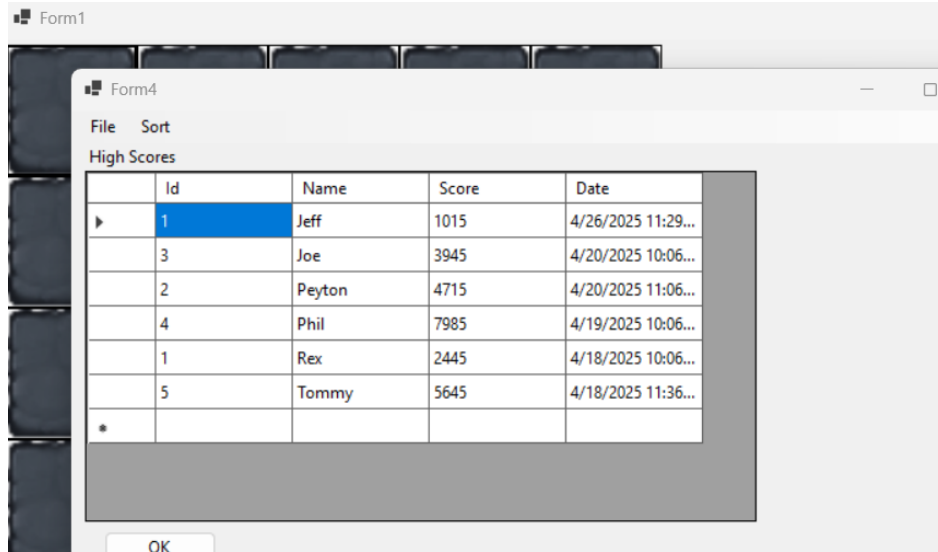
In this screenshot we have created Form3 to capture the winner's name of whoever is playing the game. From here the user can type in their name and click on the “OK” button to then save their game information, such as the name they type in here, their score, their unique identifier and the date of when this occurred.

### **Saving Winners to Document**



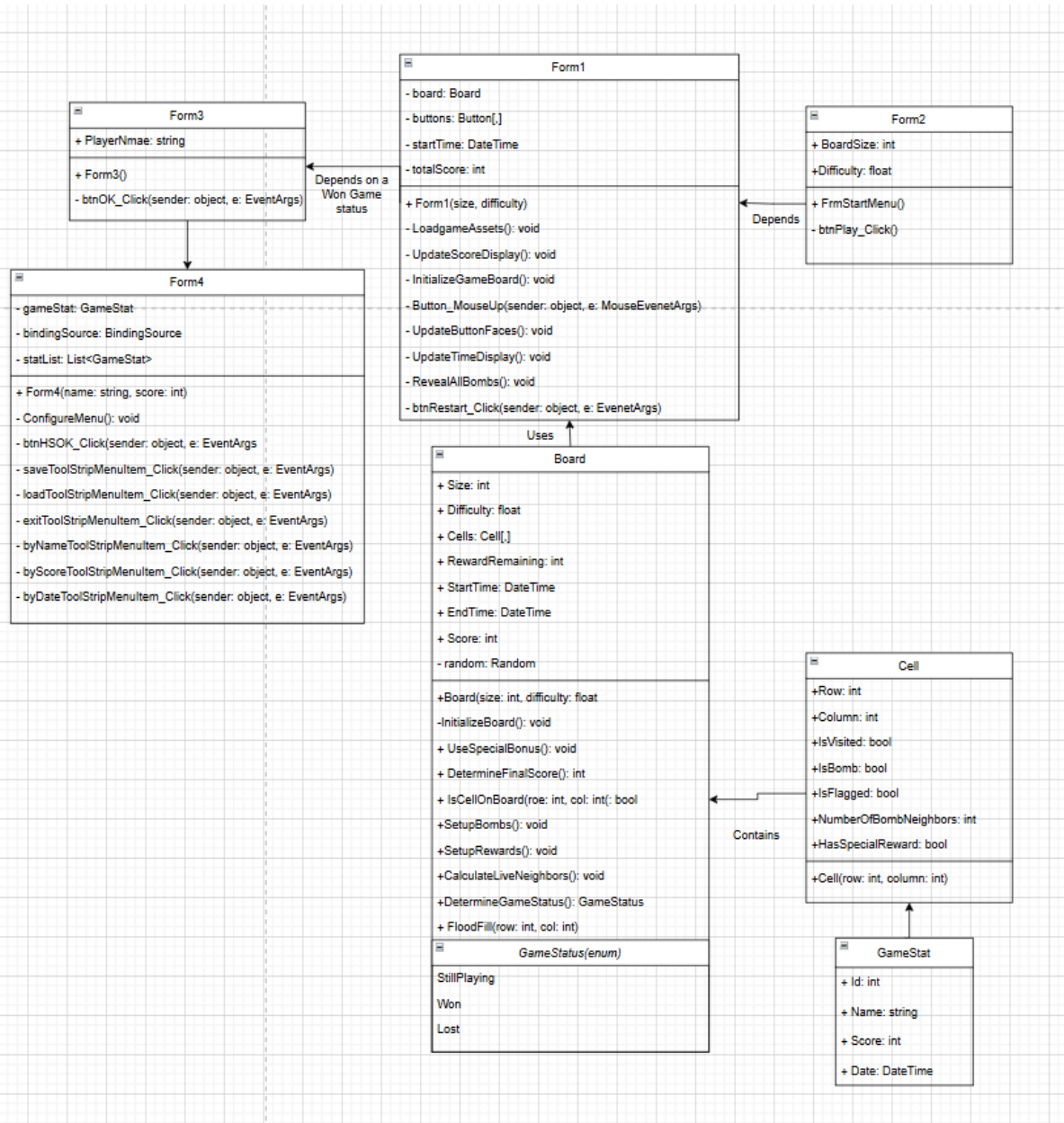
In this screenshot we are just demonstrating how we have added the ability for the user to be able to save and record their game score to a txt file. Here they will be able to keep track of how well they have done or anyone around them that may want to play can play they record their score for some fun competitive family fun.

### **Sorting Function**



In this screenshot we have added the ability for the user to be able to sort the high scores by either name and putting the in alphabetical order, by scores to get the greatest from least or by date to see the scores made a long time ago to the most recent scores.

# UML



## **Questions**

1. What was challenging?

I would have to say that the implementation of Form3 and Form4 in this Milestone presented a few challenges. The first hurdle was ensuring proper data flow between forms; Form3 needed to capture a valid player name before closing, which required input validation and error handling to prevent empty submissions from occurring. Another difficulty was managing the high score system in Form4, particularly the data binding and real-time updates to the “DataGridView”.

2. What did you learn?

With this project I have deepened my understanding of form interactions with WinForms, particularly how to pass data between forms using properties and “DialogResult”. I learned how to implement input validation effectively, ensuring that user-submitted data meets requirements before going on to be processed. Working with data binding in Form4 taught me how to dynamically update a “DataGirdView” and apply sorting without losing data integrity. Also, I gained experience with list manipulation, such as sorting collections with LINQ and resetting data sources for UI updates.

3. How would you improve on the project?

To improve this project, I would first strengthen the validation in Form3 by adding character limits and input sanitation to ensure player names don’t contain any sort of problematic characters they could interfere with file storage system with a lightweight database. This could help to provide a more realiable high-score management system, better query capabilities and improved data integrity. We also could refactor the sorting logic into a dedicated service class to better follow the separations of concerns principles and eliminate code duplication (this is something I am still trying to figure out more and work on).

4. How can you use what you learned on the job?

I would have to say that the skills developed in this Milestone have direct applications in professional software development. My experience with form management and model dialogs translates perfectly to any application requiring user input, which is fundamental in desktop application development. The data binding techniques I implemented for high scores displays are directly applicable to enterprise applications that need to present dynamic data in grids or lists, such as

admin dashboards or reporting tools. The LINQ based sorting and filtering logic I developed can be leveraged in any data intensive application, particularly those involving reports or CRUD interfaces

### **Day-to-Day**

Monday

Start:5pm End: 6:30pm Started on Activity 5

Start: End: Activity:

Start: End: Activity:

Tuesday

Nothing

Wednesday

Start:6pm End:9pm Continued on Activity 5

Start:9pm End:10pm Activity: Activity DQ1

Thursday:

Start:6pm End:9pm Continued on Activity 5

Start: 5pmEnd:6pm Activity: Tutoring session

Friday

Start: 5pmEnd:6pm Activity: Tutoring session

Start:8pm End:10pm Activity: Started on Milestone 5 Project

Start: 10pm End:11pm Activity: Activity DQ2

Saturday

Start: 12pm End:4pm Activity: Continued Activity 5

Sunday

Start:11am End:3pm Activity: Continued Activity 5

Start: 5pmEnd:6pm Activity: Tutoring session

Start:6pm End:8pm Activity: Continued Milestone 5 Project

### Specs

Item	Value
OS Name	Microsoft Windows 11 Home
Version	10.0.26100 Build 26100
Other OS Description	Not Available
OS Manufacturer	Microsoft Corporation
System Name	MSI
System Manufacturer	Micro-Star International Co., Ltd.
System Model	GS65 Stealth 9SD
System Type	x64-based PC
System SKU	16Q4.1
Processor	Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz, 2601 Mhz, 6 Core(s), 12 Logica...
BIOS Version/Date	American Megatrends Inc. E16Q4IMS.10D, 3/12/2019
SMBIOS Version	3.2
Embedded Controller Version	255.255
BIOS Mode	UEFI
BaseBoard Manufacturer	Micro-Star International Co., Ltd.
BaseBoard Product	MS-16Q4
BaseBoard Version	REV:1.0
Platform Role	Mobile
Secure Boot State	On
PCR7 Configuration	Elevation Required to View
Windows Directory	C:\WINDOWS
System Directory	C:\WINDOWS\system32
Boot Device	\Device\HarddiskVolume1
Locale	United States
Hardware Abstraction Layer	Version = "10.0.26100.1"
User Name	MSI\USER

### Test Cases

Test Case Id	Test Case Description	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
Test1	High Score saving	1. Win game 2.Player enters name 3. Save	Name="Player1" Score= 1250	Score appears in Form4 Persists on reload	The system should save the winning players name and score to the	Pass

					DataGrid on Form4	
<b>Test2</b>	Score Sorting	1.Form4 Opens 2.Click “Sort by Score”	Scores: [500,1200,8 00]	Displays as [1200,80 0,500]	The DataGrid will display the scores from greatest to least	Pass
<b>Test3</b>	Name sorting	1.Form4 Opens 2.Click “Sort by Name”	Names: [Zac,Herny, Peyton]	Displays as [Henry, Peyton, Zac]	The DataGrid will display the names in alphabeti cal order	Pass

### **Programming Conventions**

1. We will be using the naming convention of camelCase for variables and parameters, like for “isVisited” and “numberOfBombNeighbors” and PascalCase for methods and classes, like for GameBoard making sure that constants in UPPER\_CASE (MAX\_BOARD\_SIZE).
2. Making sure that we have the correct error handling in place for input validation to check the bounds before accessing cell (if (row < 0 || row >= Size) return; ). Guard clauses can also asset in early returns for invalid states (if (cellsVisited) return; ) and exception handling to user try-catch for critical operations.
3. Ensuring comprehensive documentation practices with comments for all public members to be able to review and making sure we have concise inline comments to explain complex algorithms or non-obvious decisions, while trying to avoid any redundant explanations of simpler parts of the code.

### **Use Case Diagram With Flowchart**



**Description:** The game needs a system to record the player names and scores after winning, then display and manage high scores for competitive play.

**Primary Actor:** Player

**Secondary Actors:** System (Game Logic)

File Storage (High Score Database)

**Goals:**

1. Capture player name after winning
2. Save and load high scores persistently
3. Display and sort high scores

**Stakeholders:**

- Players (Competitive motivation)
- Developer (Game analytics/improvements)

**Pre-conditions:** - Player has won a game (all safe cells revealed)

- Game has calculated the final score

**Post-Conditions:**

- Player's name and score are recorded
- High scores are saved to persistent storage
- Updated high score list is viewable

**Basic Flow:**

1. Player wins the game; System shows victory message.

2. System opens Form3

- Player enters name in text field

- Player clicks "OK" to submit

3. System validates input (non-empty name)

4. System creates GameStat object with:

- Player Name

- Score

- Current time stamp

5. System opens Form4

- Binds high score list to DataGridView
- Shows entries sorted by score (descending)

6. System saves high score to file

**Alternative Paths:**

Path 1: - Invalid name Submission

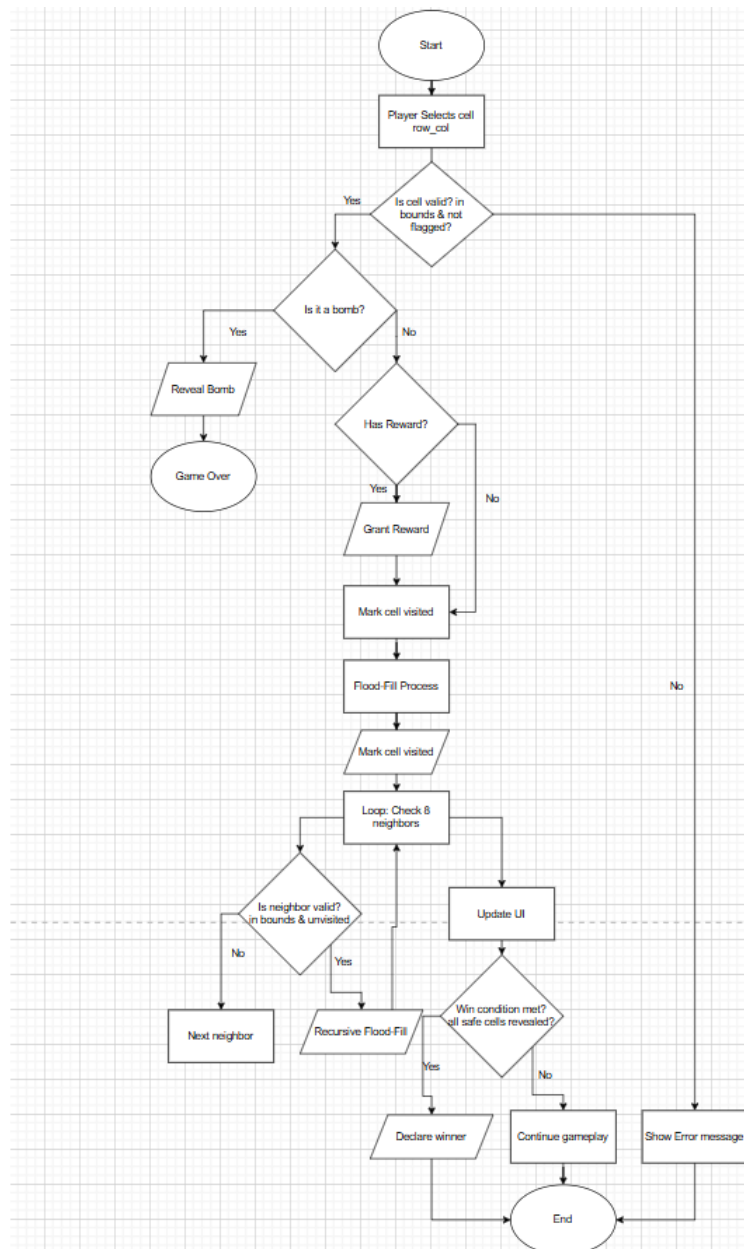
- At Step 3, if player submits empty name:
  - o System shows error: "Please enter the name"
  - o Form3 remains open until valid input

Path2: - Manual High Score Viewing

- Player clicks "High Scores" button in main menu
- System loads saved scores from file
- System opens Form4 with current high scores

Path3:- Score Sorting Options

- In Form4, player can click headers to sort by:
  - o Name (alphabetically)
  - o Score (highest first)
  - o Date (most recent first)



### **Bug Report**

Nothing at this time.

