**Title:** CST-150 Activity 2 – Seconds Conversion Application

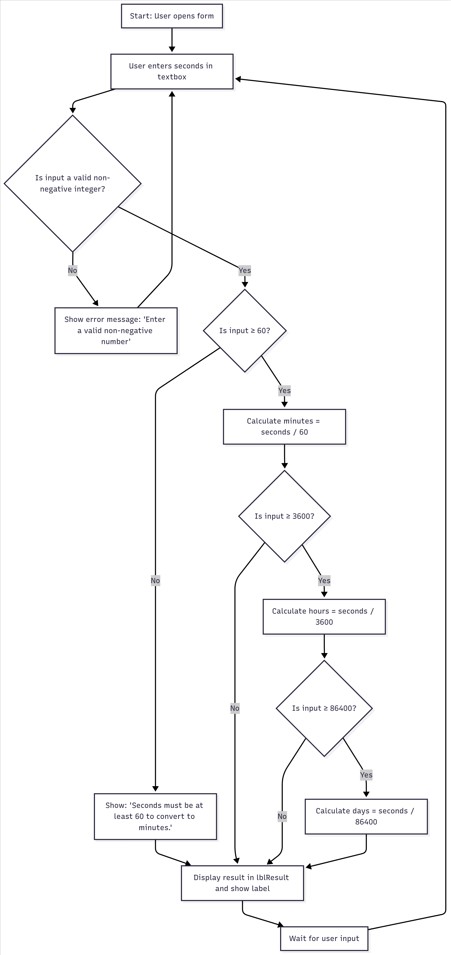
**Name:** Eric Gathinji

**Date:** June 23, 2025

**Course:** CST-150

**Github link:**

**Video link:**

Flowchart of the program Description:

The application begins by prompting the user to enter a number of seconds.

The input is validated using int.TryParse() to ensure it’s a non-negative integer.

If the input is less than 60, a message is displayed stating that at least 60 seconds are needed to convert.

If the input is 60 or more, it calculates the number of **minutes**.

If 3600 or more, it adds **hours**.

If 86400 or more, it adds **days**.

The output is shown in a label.

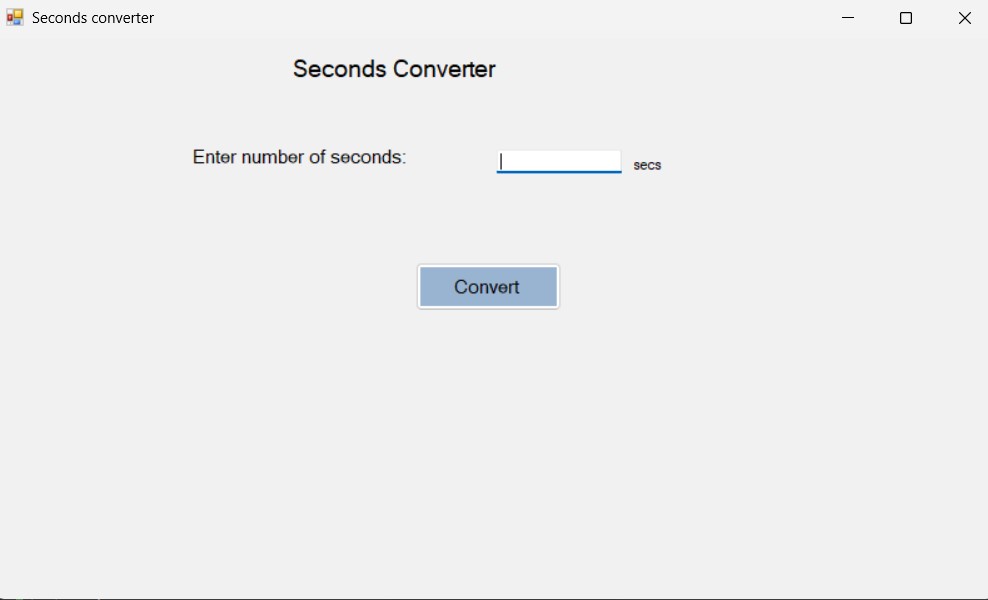
**Logic Used:**

* Constants: SECONDS\_IN\_MINUTE, SECONDS\_IN\_HOUR, SECONDS\_IN\_DAY
* Conditional logic: nested if statements

* Input validation: TryParse

* Output formatting: string output += ...

**Screenshot of the Form Before Being Populated**



Shows the default interface with:

* A textbox labeled “Enter number of seconds”

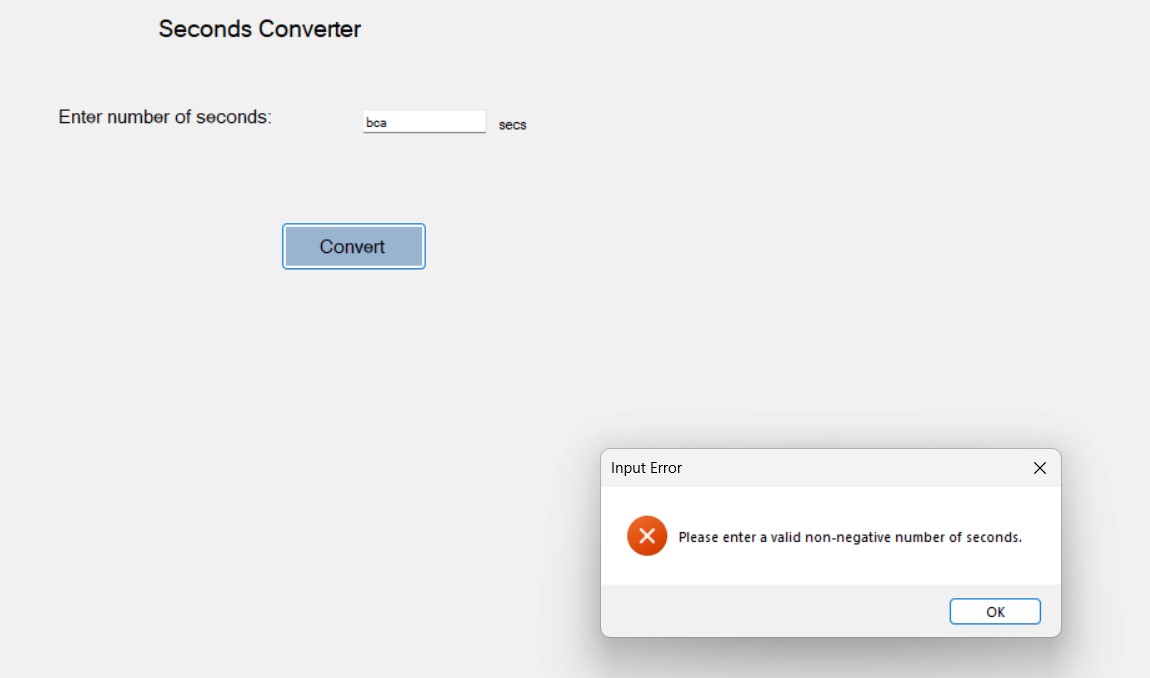
* A button labeled “Convert”

* An output label (initially hidden)

**Explanation:**

This screenshot captures the initial state of the form before user interaction. It reflects the design stage where the user interface is ready for input but no actions have yet occurred.

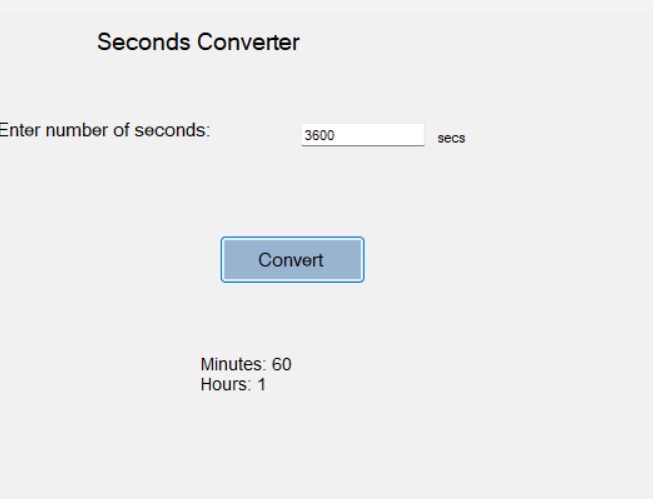
**Screenshots of the Form After Being Populated**



App displays: “Please enter a valid non-negative number of seconds.”

**Explanation:**

This tests the exception handling. The application correctly prevents invalid input from being processed and guides the user through a friendly message.

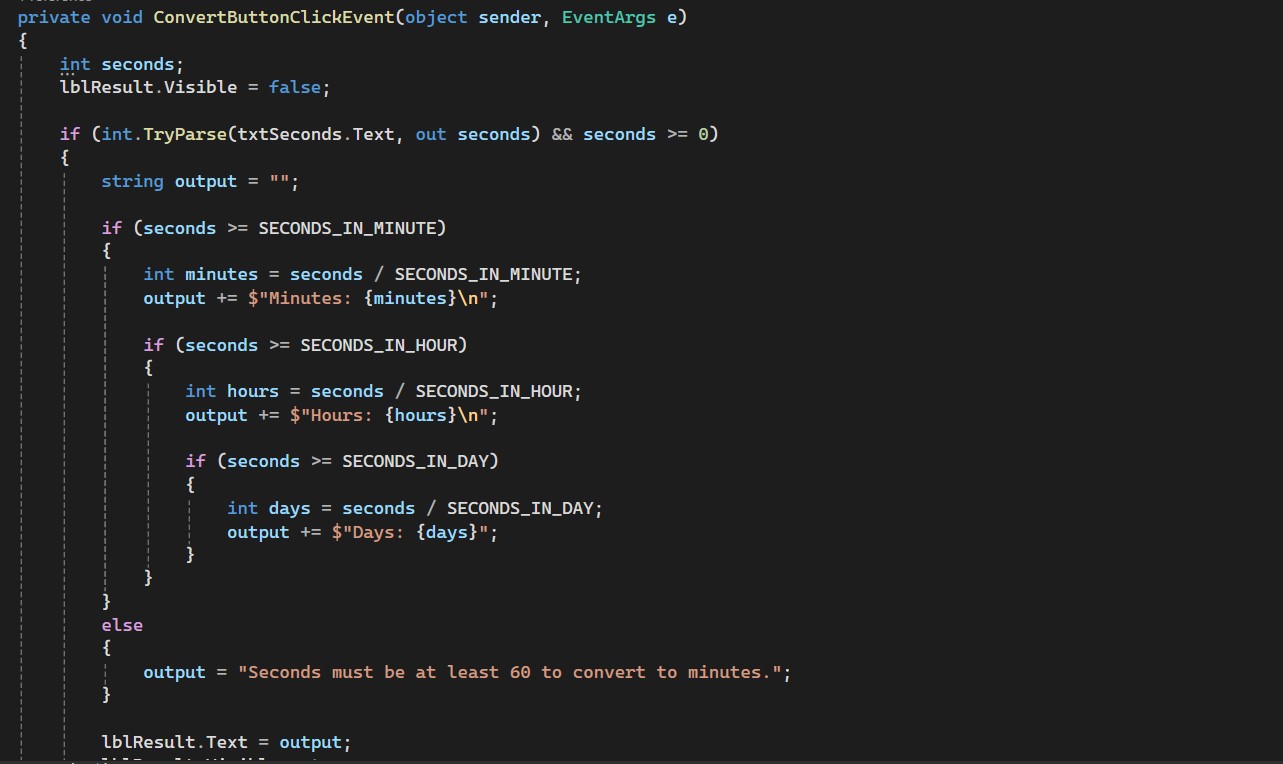
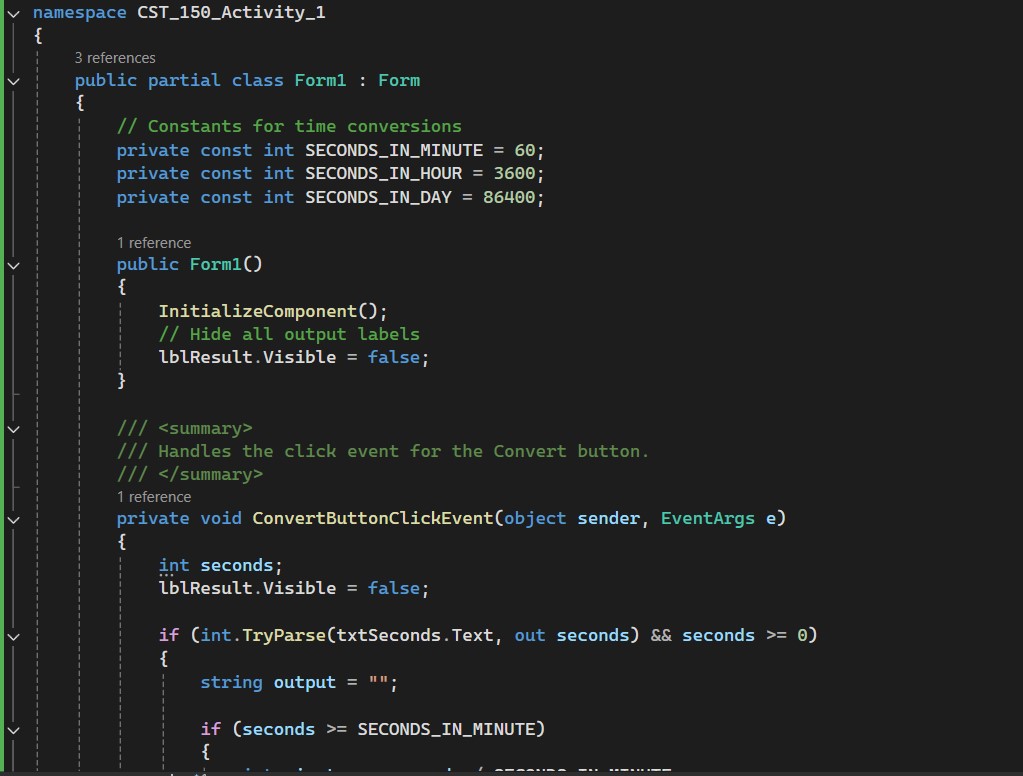


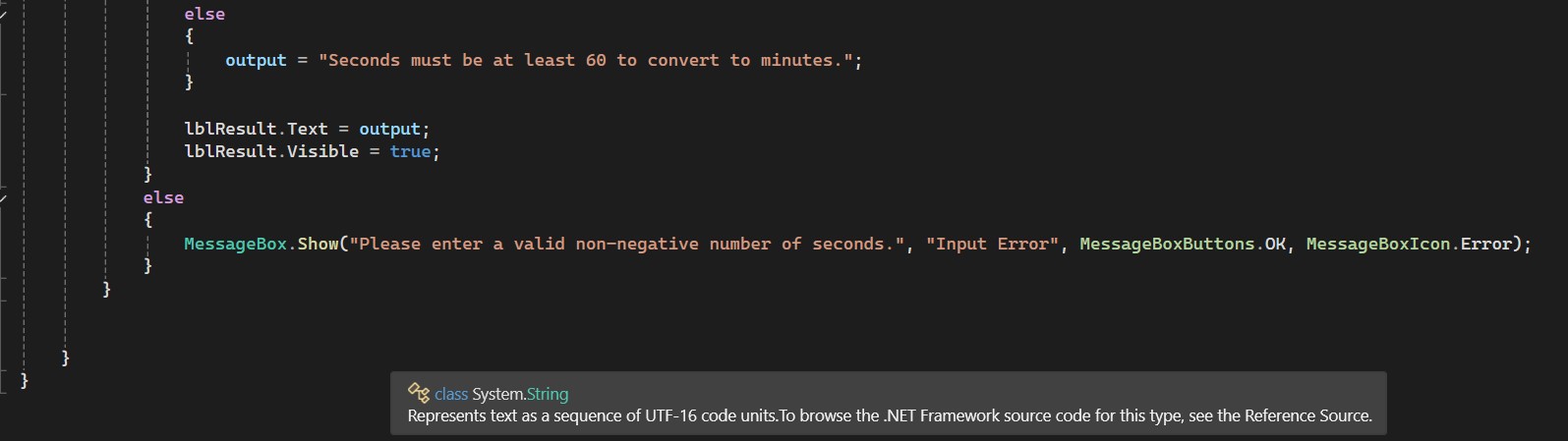
Valid Input

**Explanation:**

This shows the full logic in action. Since the number is over 86400, all conversion levels (minutes, hours, days) are triggered and displayed. The result is formatted clearly in the output label.

**Screenshot(s) of the Code Behind**





Shows the ConvertButtonClickEvent method

Constants are declared at the top of the class

Includes full logic from TryParse → nested ifs → string formatting

The code has been properly commented;

Monday

Start: End: Activity:

Tuesday

Start: End: Activity:

Wednesday

Start: End: Activity:

Thursday

Start: End: Activity:

Friday

Start: End: Activity:

Saturday

Start: End: Activity:

Sunday

Start: End: Activity: