

Project Report: Wildfire Risk Alert System

Prasun Gajurel

1. Description of the Project:

This project is a Wildfire Risk Alert System that is designed to analyze environmental conditions and determine the likelihood of a wildfire happening near you. The objective is to take key weather and environmental inputs, such as temperature, wind speed, humidity, rainfall, and proximity to certain areas and output a categorized risk level of: Low, Moderate, High, or Extreme. This system helps quickly assess wildfire risks for awareness and prevention as well.

2. Significance of the Project:

This project is significant because wildfires have become much more common and dangerous due to climate change. This project addresses a real-world problem by providing a quick way to look into wildfire risk using environmental data that is easy to get. This system is easy to use and can be used locally or a part of a larger safety system. This gives users who are scared of wildfires the chance to check for themselves if they are in danger or not.

3. Installation and Usage Instructions:

First, you will be asked to enter:

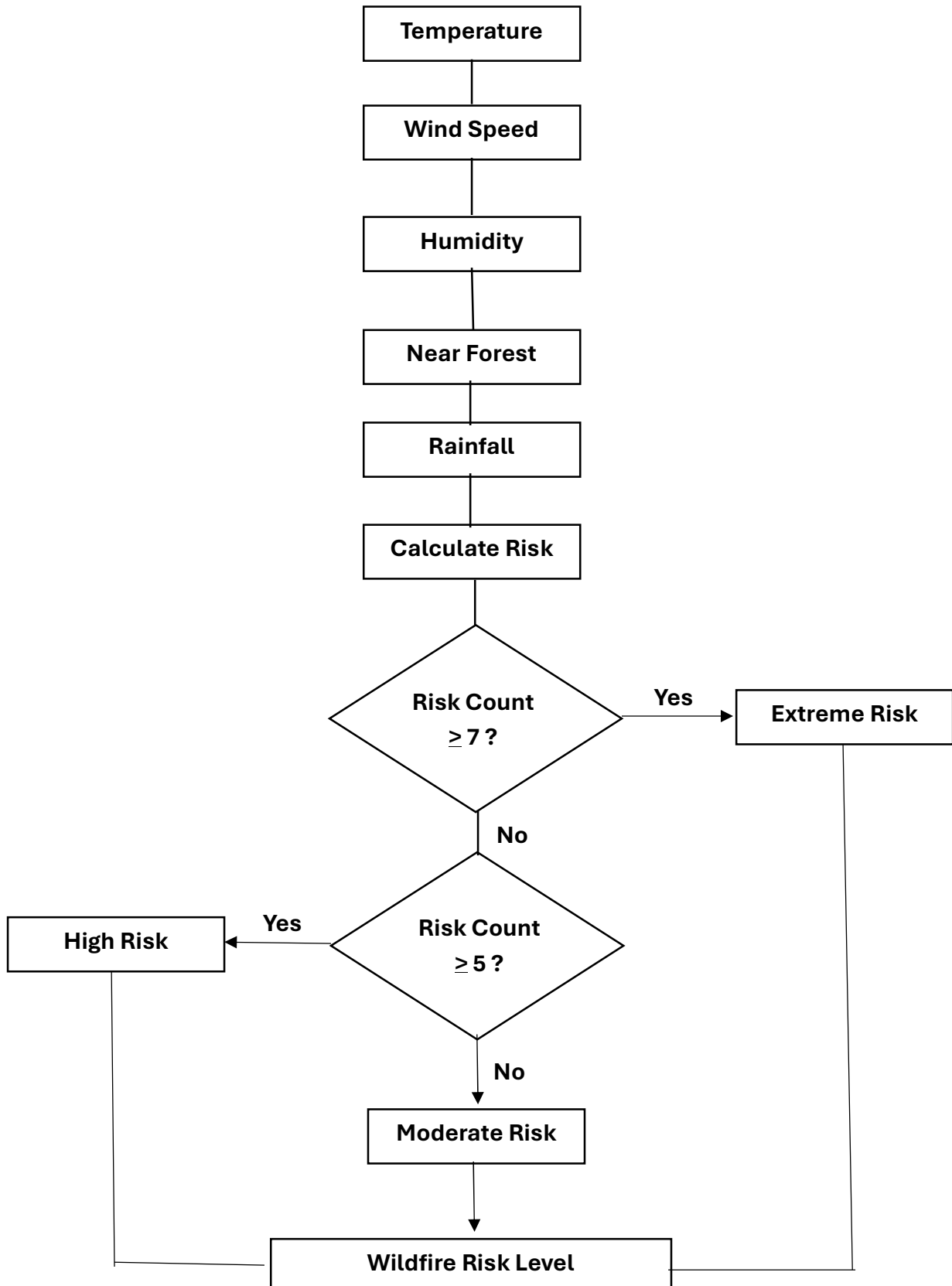
- The temperature in °F
- Wind speed in mph
- Humidity percentage
- If you're near a forest or dry grassland ("yes" or "no")
- How much rain has there been in the last 7 days (in inches)

Then, based on the answers, the program calculates a risk score using a point system. For example:

- High temperature or strong winds = more risk points
- Being near a forest = extra risk

After calculating, it prints a summary with your input and the wildfire risk level

4. Structure of the Code:



5. Functionalities:

- **WildfireRiskSystem**: a class that stores all the information about the weather and surroundings and figures out the wildfire risk level based on these variables.
- **calculate_risk()**: a method that looks at the temperature, wind speed, humidity, proximity to a forest, and rainfall. Based on these, it adds up a "risk count" and then labels the wildfire risk as either Low, Moderate, High, or Extreme risk.
- **display_summary()**: a method that prints a summary showing the weather information and the final wildfire risk level after the risk level is calculated.
- **is_number(s)**: a function that checks if a given input is a number. It helps make sure the user enters real numbers instead of random words.
- **get_valid_input(prompt)**: a function that asks the user for a number (like temperature or rainfall), keeps asking until the user types a valid number, and then returns it as a float.
- **get_yes_no(prompt)**: a function asks the user a yes-or-no question (like "Are you near a forest?") and keeps asking until the user types "yes" or "no".
- **main()**: the function that runs the whole program by asking the user for weather info, calculating the wildfire risk, and showing the results

6. Test Results:

- **Input #1: Extreme**

```
What is the Temperature (°F): 95
What is the Wind Speed (mph): 25
What is the Humidity (%): 15
Are you near a forest or dry grassland? (yes/no): yes
Amount of Rainfall in the past 7 days (inches): 0.1
```

- **Output #1:**

```
Wildfire Risk Level: Extreme Risk
```

- **Input #2: Moderate**

```
What is the Temperature (°F): 80
What is the Wind Speed (mph): 12
What is the Humidity (%): 35
Are you near a forest or dry grassland? (yes/no): no
Amount of Rainfall in the past 7 days (inches): 0.5
```

- **Output #2:**

```
Wildfire Risk Level: Moderate Risk
```

- **Input #3: Low**

```
What is the Temperature (°F): 65
What is the Wind Speed (mph): 5
What is the Humidity (%): 50
Are you near a forest or dry grassland? (yes/no): no
Amount of Rainfall in the past 7 days (inches): 2
```

- **Output #3:**

```
Wildfire Risk Level: Low Risk
```

7. Achievement of Project Goals:

I believe that this project achieves the project goals because the wildfire risk alert system is a simple and effective way for a person to easily assess the risk of wildfire in their area. The code collects user input, checks that the input is valid, applies a clear set of rules to calculate risk, and then shows the results in a summary that shows the risk of a wildfire. This project showed how a simple program can be used to solve a real-world problem like wildfires.

8. Discussion and Conclusion:

Challenges Faced:

A challenge that I faced with this project was making sure the program handled invalid user input and certain symbols that were not needed. I also had to keep the risk calculation system simple but still try to make it useful and effective so that it could still benefit the user.

Limitations:

Some limitations of this system could be that the system does not consider more complex factors like fuel type, land features, or changing weather patterns. It also treats all factors as equally important, which might not be true in real-world wildfire scenarios.

Possible Improvements:

In the future, we could improve the project by:

- Adding a feature to save past risk assessments.
- Building a visual dashboard or map interface for users.
- Connecting the system to real-time weather data for better accuracy

Course Learning Applied:

This project helped me use and get better with key concepts like conditionals, loops, and object-oriented programming. It also showed how lessons from class can come together to solve real-world problems in an impactful and effective way.