Requirements for Fire Spread Simulation Software

Weiliang Zhao

January 28, 2023

Analyze requirements to determine appropriate testing strategies [default 20%]

1. **Range of requirements, functional requirements, measurable quality attributes, qualitative requirements**

   Stakeholders

   (a) Government agencies:
   These organizations may use the software to plan for and respond to fires, and may also be responsible for purchasing and implementing the software.

   (b) Fire departments:
   Fire departments would likely be among the primary users of the software, as it would help them plan and execute their firefighting efforts.

   (c) Insurance companies:
   Insurance companies may be interested in using the software to assess fire risk and make informed decisions about coverage and pricing.

   (d) Property owners:
   Individuals and businesses who own property in areas at risk of fire may be interested in using the software to help them make informed decisions about their properties.

   (e) Environmental organizations:
   Groups concerned with the impact of fires on the environment may be interested in using the software to understand and mitigate the impact of fires.

   (f) Researchers:
   Researchers studying fire behavior and the impacts of fires on people, property, and the environment may use the software to gather data and inform their work.

   Range of requirement

   (a) Accurate prediction of fire behavior:
   The software should be able to accurately predict the spread of a fire based on factors such as wind speed and direction, humidity, and the type of fuel being burned.

   (b) Integration with other systems:
   The software may need to be able to integrate with other systems, such as weather forecasting software or satellite imagery, to gather the necessary input data.

   (c) User-friendly interface:
   The software should have a user-friendly interface that allows users to input and analyze data easily.

   (d) Scalability:
   The software should be able to handle large amounts of data and function effectively in various environments.

   (e) Reliability:
   The software should be reliable and able to function correctly even in challenging or high-stress situations.

(f) Security:
The software should have robust security measures in place to protect sensitive data and prevent unauthorized access.

Functional Requirements

(a) Ability to input and analyze data:
The software should allow users to input data such as wind speed and direction, humidity, and the type of fuel being burned, and use this data to predict the spread of a fire.

(b) Visualization of fire spread:
The software should be able to display the predicted spread of a fire in a visual format, such as a map or graph, to help users understand and analyze the data.

(c) Integration with other systems:
The software should be able to integrate with other systems, such as weather forecasting software or satellite imagery, to gather the necessary input data.

(d) Alerts and notifications:
The software should be able to send alerts and notifications to users in the event of a predicted fire, allowing them to take necessary precautions.

(e) Customization:
The software should allow users to customize their predictions based on specific needs and preferences.

(f) Data storage and management:
The software should be able to store and manage large amounts of data, and allow users to access and analyze this data as needed.

Measurable Quality Attributes

(a) Accuracy:
The software's predictions should be accurate, with a low rate of error.

(b) Speed:
The software should be able to process and analyze data quickly, allowing users to make timely decisions.

(c) Reliability:
The software should be reliable, functioning correctly even in challenging or high-stress situations.

(d) Usability:
The software should have a user-friendly interface that is easy to use and understand.

(e) Scalability:
The software should be able to handle large amounts of data and be able to function effectively in a variety of different environments.

(f) Security: The software should have robust security measures in place to protect sensitive data and prevent unauthorized access.

(g) Maintainability:
The software should be easy to maintain and update as needed.