Giancarlo Sturla

Overview

One of the most difficult jobs for meteorologists is to accurately predict and track storms like tornadoes. It turns out that one of the most useful sources of information for meteorologist when tracking storms is people reporting dangerous weather conditions. So we want to create an application that uses user data to report current weather conditions.

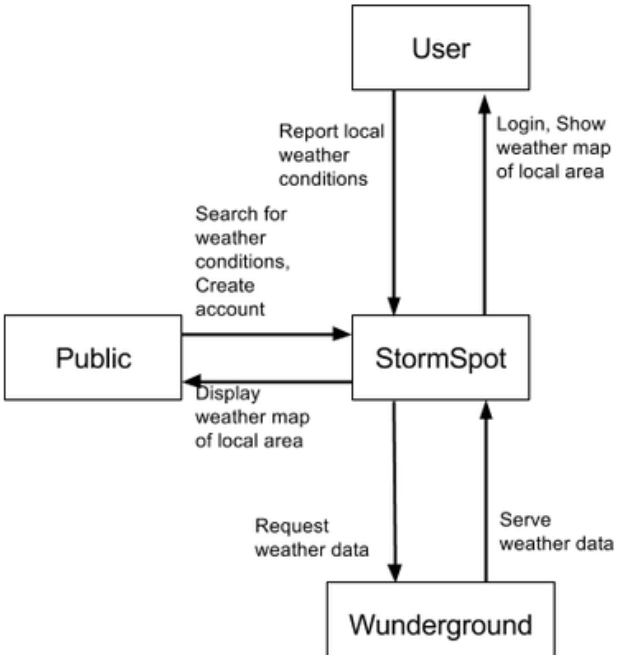
Purposes

StormSpot aims to aggregate user information to increase the accuracy of storm tracking and emergency alerts. This application has three main purposes in order of importance:

1) Show the public current weather conditions (especially severe weather)
   a) The basic use of the application is simply to see the weather conditions around your location. We wants to be especially accurate with severe conditions. My hope is that users would be more inclined to report dangerous storms, so user input should be at a max around those locations that have sever weather.
2) Allow users to report local weather conditions
   a) We want our users to be active in reporting dangerous weather conditions, which could warn the nearby public about the severe storms. One problem that could arise, however, is misinformation. So we must have a good way of tracking accurate data from imprecise data.
3) Help meteorologists track storms that are inherently difficult to track, such as tornadoes.
   a) With the amount of data that we hope to receive from this application, meteorologists could look at data that could possibly help make predictions when tracking future storms.

Context

To the right is the context diagram of the application. It shows two main roles that may interact with StormSpot. The most basic role is the Public. This is how to interact with the basic use of the app: to search for local weather reports and alerts in a particular location. The second role is the User. This is the role that will help the application prosper. The user may report local weather conditions. We hope that the user information will act as a significant source of weather data to show to the public.



Finally, we will rely on Wunderground (Weather Underground) to retrieve radar data and display a map on the application page. We will use information about severe weather conditions; however, we hope that we can increase the amount of information that can be obtained by the radar map by displaying the user information about weather conditions.

Concepts

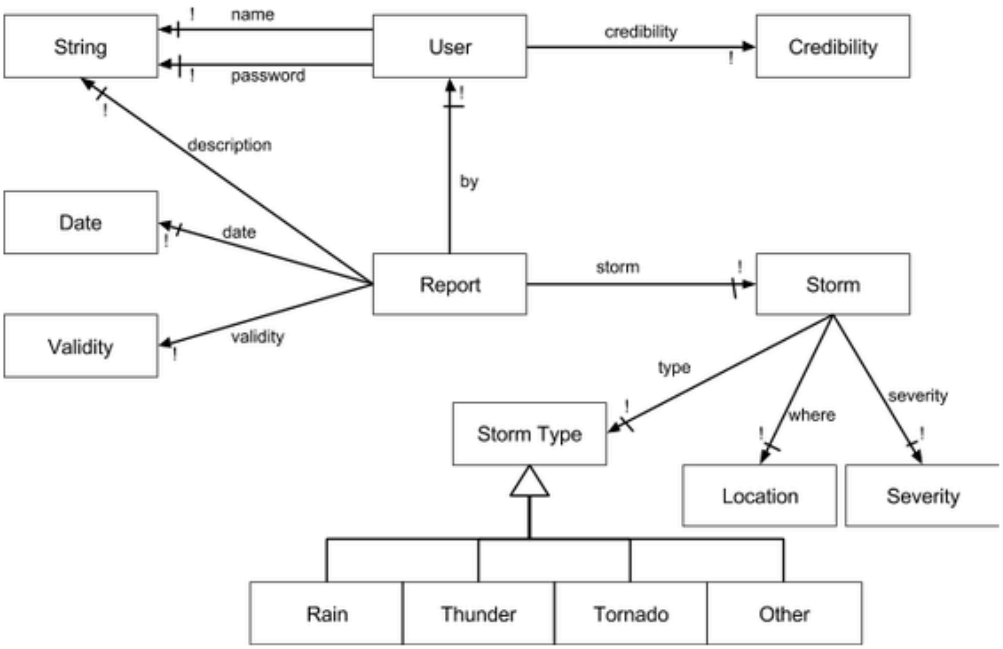
1) Weather Map
    a) Motivation
        i) Show the public current weather conditions
    b) Description
        i) Like many weather applications, we will show the public the weather conditions on a map. What makes this different from other applications is that we also include on the map icons of user input and highlight severe conditions over less severe conditions.
2) Weather Reports
    a) Motivation
        i) Allow users to report local weather conditions
        ii) Help meteorologists track storms that are inherently difficult to track, such as tornadoes.
    b) Description
        i) This application is fueled off of user weather reports. So this is, in fact, a key concept to our application. We want our weather reports to increase the accuracy of storm tracking and prediction. We will allow users to report weather conditions and keep track of the validity of the report. Users will have a credibility rating based on the accuracy of the rating. Severe reports will have a heavier weight that other conditions.

Data Model

The following data model diagram shows how I envision organizing the design. In order to show reports on a weather map we will have to have the concept of a storm. This means that a storm is something that the user can use to describe the current weather conditions in a particular location. By giving the storm a location in the data model, we can effectively portray them on a map.

Weather reports is where the main interest in the application lies. We want to keep track of the validity of the weather report. In order to do this, users can up vote or down vote a report. Also, the validity of a weather report will affect the credibility of a user. We want users to give accurate information, so the credibility attribute on a User will make the people inclined to give true data.

Finally, since we want to help meteorologists use the data to help track current storms or predict future storms, we add some extra attributes to a report such as a data (time). Using this data, meteorologists might be able to find patterns in certain areas and more accurately predict storms.


String
name
password
User
credibility
Credibility
description
by
Date
date
Report
storm
Storm
Validity
validity
type
where
severity
Storm Type
Location
Severity
Rain
Thunder
Tornado
Other