Location Alert System

An application of the future by Brandon Frankenfield, Nicholas Escalona, and Sofani Mesfun.

The Location Alert System is an app that aims to provide customers with realtime, pressing updates about weather, traffic, and news. Unlike typical apps that push alerts to their users, the Location Alert System allows customers to specify a radius of interest using a simple web-based map of the United States.

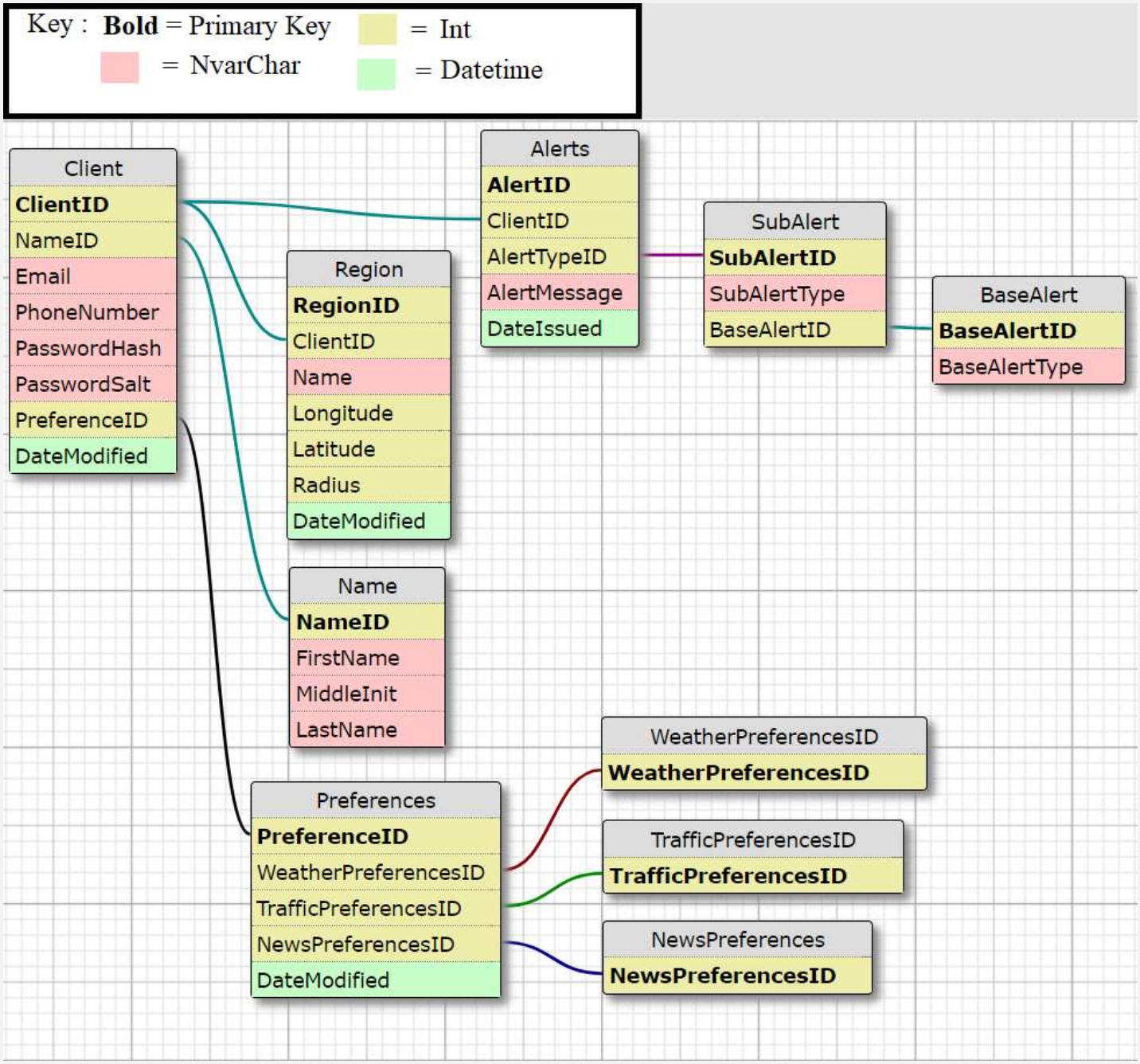
Customers sign up and subscribe to a service that warns them via selected communication channels (email, text messages, browser-desktop notifications) of important updates pertaining to their selected locations throughout the United States.

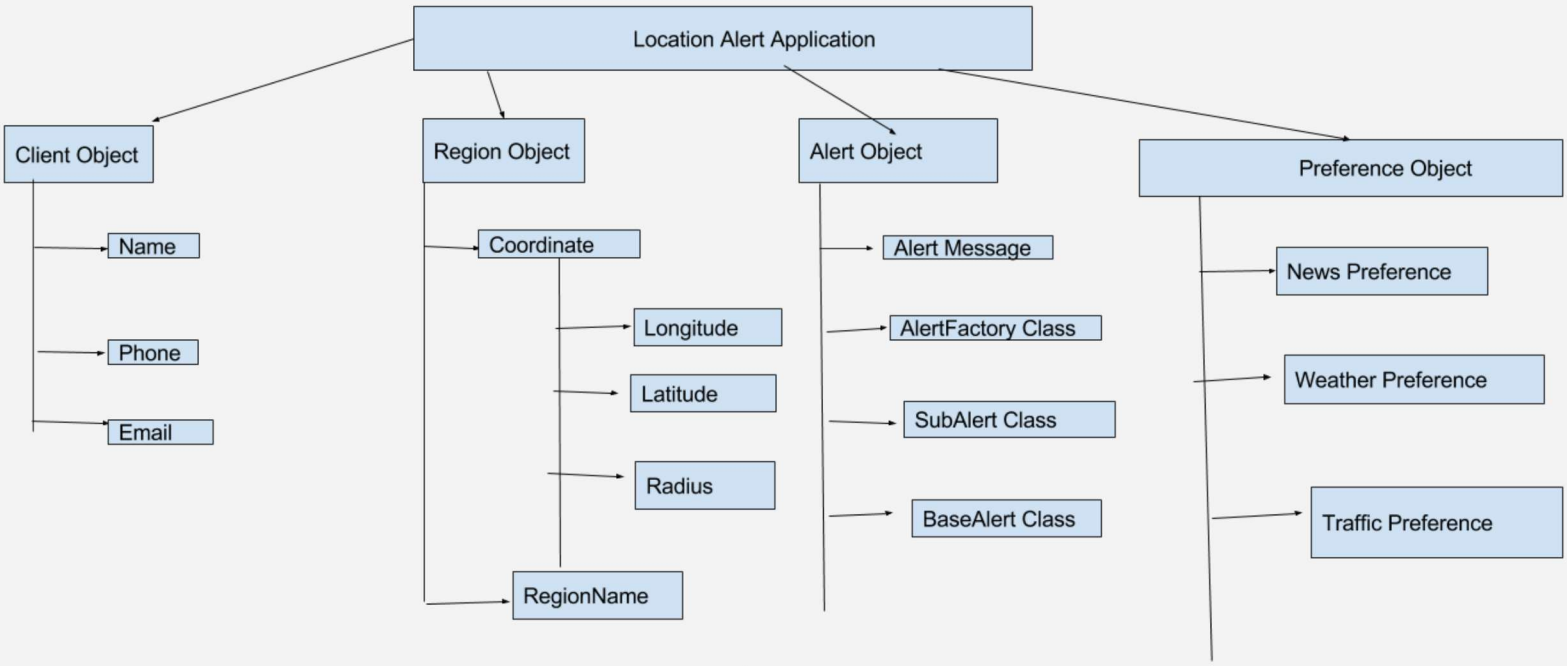
A web portal for clients allows them to select an area-radius of the locations as well as severity preferences. Updates concerning events use the severity filter to ensure only the updates clients are interested in are pushed to their channels.

Technology Stack :

SQL Database,
Transact-SQL,
MVC,
Google Maps API,
Yahoo Weather API,
Bing Traffic API,
JQuery,
Javascript,
Entity Framework,
Jenkins Server,
GitHub Source Control,
xUnit Testing

Data Entity Relationship Diagram





Object Model

LocationAlertUpdater is the manager of all other objects.

The **Client** object represents the user, containing their preferences, regions, and credentials.

Name class contains FirstName, MiddleInit, and LastName properties

Email class contains **MailAddress EmailAddress** property with proper validation methods

Phone class contains **string PhoneNumber** property with proper validation methods

IPreferences interface

⇒ **WeatherPreference -** object (UI Properties) is for UI purposes and push alerts, having weather Alerts(Storms, Temperature, Rain) based on location

⇒ **TrafficPreference -** object (UI Properties) : is for UI purposes and push alerts, having current Traffic alerts based on location

⇒ **NewsPreference -** object (UI Properties) : is for UI purposes and push alerts, having current news based on location

The **Region** object will use Google Maps Drawing API to allow Client to specify locations for updates.

Coordinate Struct (A struct with longitude, latitude and radius properties)
- ⇒ Longitude (Longitude for region center based on google map API)
- ⇒ Latitude (Latitude for region center based on google map API)
- ⇒ Radius (Radius of region based on google map API)

RegionName (String) Name of location supplied by client

The **Alert** object (will Alert to our client based on their needs (Weather, Traffic, News)
- ⇒ Alert Message containing information pushed to client
- ⇒ SubAlert property ID
- ⇒ BaseAlert property ID