Fassen Sie auf dieser Seite kurz zusammen, welche Aufgabe Ihr Projekt erfüllen soll. Beschreiben Sie dazu, welche grundsätzliche Visualisierungsstruktur eingesetzt wird, und welche Datentypen in welcher Form visualisiert werden.

Projektsteckbrief

Name

Erdbeben und Erdbebenängste in den USA

Domäne

Geoinformationssystem

Aufgabe

Visualisierung von Erbebenvorkommen am Beispiel der USA und Vergleich dessen mit den Ängsten der Amerikanischen Bevölkerung.

Struktur

Geoinformationssystem ( Karte der USA).

Datentypen und visuelle Kodierung

Quantitativ: Umfragedaten zum Thema Erdbebenängste der Amerikanischen Bevölkerung  
Kodierung: Geographisch

Quantitativ: Datenerhebung zur Stärke und Position auftretender Erdbeben der letzten 30 Jahre in den USA  
Kodierung: Geographisch

Ziel

Als Ziel haben wir uns gesetzt, eine übersichtliche und verständliche Visualisierung der Erdbebenproblematik in den USA umzusetzen. Hierbei möchten wir unter Zuhilfenahme einer Karte der USA und unter Verwendung mehrerer Layer eine interaktive Karte erstellen, auf der verschiedene Eigenschaften unserer Datensätze miteinander verglichen werden können. Hierbei steht die Angst vor Erdbeben in einzelnen Regionen der USA im Vordergrund und wird mit der tatsächlichen Vorkommens Häufigkeit der Erdbeben verglichen, um festzustellen ob die Ängste in Erdbeben gefährdeten Regionen größer sind, als in ungefährdeten. Wir wollen die Angst vor einem fatalen Erdbeben visualisieren. Außerdem wollen wir die Angst vor Erdbeben anhand von Alter tiefergehend analysieren um zu zeigen, ob es signifikante Unterschiede gibt.

Software-Aufbau

Wir werden unsere Anwendung nach dem MVC-Pattern konstruieren. Es wird eine Modelklasse geben (EarthquakeDataProcesser), die die Umfragedaten aus einer CSV-Datei ausliest, aufbereitet und der Viewklasse (MapView) zur Verfügung stellt. Dort werden die übergebenen Daten Visualisiert und in Form von einer Map und verschiedenen Layern dargestellt. Benutzerinteraktion wird über die UserInputController-Klasse geregelt, in der auf Nutzereingaben gehorcht wird und diese weiter verarbeitet werden. Je nach Input des Users reagiert der MapView und passt die Darstellung an.

