# Department of Computing

**CS344: Web Engineering**

**Class: BSCS – 12AB**

# Lab 11: City Search

**Date: 10 May 2024**

# Time: 09:00AM – 11:50AM/ 02:30 PM - 04:50 PM

# Instructor: Dr Farzana Jabeen

# 

# Lab 11: City Search

## Introduction

In this lab, you will create an application that will load data from the provided Geo City Lite CSV file to an appropriately defined NoSQL database. You must clearly model the cities, countries, and location from the GeoCityLite file and store everything as a document. Once the data has been loaded in the DB you should allow the user to search for 5 cities closest to a particular point on earth.

**Description**

The Geo City lite CSV file provides information about locations by name, region and lat/long. You must not change the CSV file or add any additional information in the DB besides a primary key. The application will start by checking the DB for any data and in case the data has not been already loaded (first execution), the CSV file is read from the class path and data is loaded in the DB.

Next allow the user to search for 5 cities closest to a point, identified by its name or a lat/long. For lat/long simply apply the nearby search to locate cities near the lat/long from the stored data. In case of city name, use reverse search to find out the lat/long for that city, before applying the same nearby search algorithm. Remember, since the earth is not flat, you cannot apply simple arithmetic operations to find out the nearby cities, instead use the “The Great Circle Distance Formula” to find out the nearby lat/long.

Latitude defines how far north or south of the equator a point is positioned.  Points alongside the equator have latitudes of zero. The North Pole has positive (north) latitude of 90, and the South Pole negative (south) latitude of -90. Accordingly, northern-hemisphere locations have positive latitude, and southern-hemisphere locations have negative latitude. Longitude describes how far east a point is, from the prime meridian: an arbitrary line on the earth surface running from pole to pole.

Latitudes are values in the range [-90, 90]. Longitudes are values in the range (-180, 180]. These values are sometimes expressed in degrees, minutes, and seconds, rather than degrees and decimals.  If you’re planning to do calculations, convert the minutes and seconds to decimals first.

You are encouraged to do some research on the internet about “The Great Circle Distance Formula” to find out how to do this task.

## Objectives

* Select a NoSQL DBMS.
* Load the data from GeoCityLite file into the selected DBMS.
* Identify an appropriate method to query the selected DBMS for information.
* Create an interface to allow the user to select a point on the map (You may use Google Maps for this).
* Alternatively, the user can first select a city from a drop down. Calculate the lat/lon for the selected city, from the stored data.
* Identify cities closest to the geo location selected by the user.
* List the closest 5 cities, into the same interface (Use AJAX to avoid page loading).
* The application must be built using laravel.

## Tools/Software Requirement

* You must use a NoSQL DBMS to store the data.
* The web application should be built using laravel.

## Pitfalls

* Any exceptions or errors leading to non-execution of submitted code.
* Failure to upload the solution on LMS.
* Failure to explain the submission, during viva.

## Deliverables

* Convert your files to a zip folder and name it as given below and upload the zip folder to LMS.
  + Name – Registration No. – Section
* This lab is graded. Min marks: 0. Max marks: 20.

**Lab Rubrics:**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Item** | **Clo** | **Plo** | **Marks** | | | | | |
| **0** | **1** | **2** | **3** | **4** | **5** |
| R1 | Concepts Related to WWW | CLO-1 | PLO-1 | The student is absent in the lab/ The submission is plagiarised. | The student is unable to understand the given problem within the context of WWW and does not select the relevant method to solve it. | Inbetween | The student requires some guidance to  understand the problem, to select relevant method, and to develop appropriate web driven program flow. | Inbetween | The student fully understands the given problem in context of WWW, is able to select the relevant method to solve it. |
| R2 | Modern Solutions | CLO-3 | PLO-5 | The student is absent in the lab/ The submission is plagiarised. | The student is unable to build the website/webapplication using the specified modern tool or framework | Inbetween | The student is trying to use some of the modern tools but is unable to utilize all the appropriate tools or to build the correct solution. | Inbetween | The student is using all the appropriate modern tools to complete the tasks at hand, and provide a modern solution. |
| R3 | Solution Accuracy | CLO-4 | PLO-3 | The student is absent in the lab/ The submission is plagiarised. | The student is unable to produce any part of the solution, with accurate results. | Inbetween | The student is able to produce accurate results for some parts of the solution. | Inbetween | The student is able to produce accurate results for the complete solution. |
| R5 | Team Response | CLO-5 | PLO-9 | The student is absent. | Student shows a lack of enthusiasm and willingness to contribute to the team's efforts, often disengaging or avoiding teamwork. | Student's contributions to the team are inconsistent, and they may hesitate to actively participate or initiate collaborative efforts. | Student generally participates willingly within the team, showing a proactive approach to collaborative work. | Student is an enthusiastic and active member of the web development team, actively participating, supporting teammates, and showing a positive attitude. | Student is a passionate and influential contributor to the web development team, actively inspiring and motivating teammates to achieve exceptional results. |