<https://serversideup.net/displaying-resources-google-map-vue-js/>

$latitude = [current\_latitude];

$longitude = [current\_longitude];

SELECT

((((acos(sin((".$latitude."\*pi()/180)) \* sin((`geo\_lat`\*pi()/180))+cos((".$latitude."\*pi()/180)) \* cos((`geo\_lat`\*pi()/180)) \* cos(((".$longitude."- `geo\_lon`)\* pi()/180))))\*180/pi())\*60\*1.1515) \* 1.609344) as distance

FROM

[table\_name]

WHERE distance

|  |
| --- |
| SELECT zip, primary\_city, |
|  | latitude, longitude, distance |
|  | FROM ( |
|  | SELECT z.zip, |
|  | z.primary\_city, |
|  | z.latitude, z.longitude, |
|  | p.radius, |
|  | p.distance\_unit |
|  | \* DEGREES(ACOS(COS(RADIANS(p.latpoint)) |
|  | \* COS(RADIANS(z.latitude)) |
|  | \* COS(RADIANS(p.longpoint - z.longitude)) |
|  | + SIN(RADIANS(p.latpoint)) |
|  | \* SIN(RADIANS(z.latitude)))) AS distance |
|  | FROM zip AS z |
|  | JOIN ( /\* these are the query parameters \*/ |
|  | SELECT 42.81 AS latpoint, -70.81 AS longpoint, |
|  | 50.0 AS radius, 111.045 AS distance\_unit |
|  | ) AS p ON 1=1 |
|  | WHERE z.latitude |
|  | BETWEEN p.latpoint - (p.radius / p.distance\_unit) |
|  | AND p.latpoint + (p.radius / p.distance\_unit) |
|  | AND z.longitude |
|  | BETWEEN p.longpoint - (p.radius / (p.distance\_unit \* COS(RADIANS(p.latpoint)))) |
|  | AND p.longpoint + (p.radius / (p.distance\_unit \* COS(RADIANS(p.latpoint)))) |
|  | ) AS d |
|  | WHERE distance <= radius |
|  | ORDER BY distance |
|  | LIMIT 15 |

def close\_destinations(latitude, longitude)

distance\_sql = Graticule::Distance::Spherical.to\_sql(:latitude => latitude, :longitude => longitude, :units => :kilometers)

Destination.all(:conditions => [ "#{distance\_sql} <= 10" ])

end