TRACK ME developer’s guide:

There will be 3 levels:

1. Trial (free)
2. Transition (secret)
3. Professional (secured)

Structure is based on device.

DB – SERVER SIDE

Date format: char 6 - MMDD (PHP : ‘ymd’)

Time format: char 13 - YYMMDD-HHMMSS (PHP : ‘ymd-his’)

JS: var d = new Date();

var yr = d.getFullYear().toString();

var cDate = yr.substr(2, 3) + d.getMonth() + d.getDate();

var cTime = yr.substr(2, 3) + d.getMonth() + d.getDate() + '-' + d.getHours() + d.getMinutes() + d.getSeconds();

Lat, lng: float 10,6

|  |  |  |
| --- | --- | --- |
| REFERENCES |  |  |
| countries | Id, name, abbr, telcode, nativename | Use native name too |
| carriers | Id, countryid, name, smsurl, tested , update, | Tested : each test will be added (or deducted if fails) |
| languages | Id, name, nativename | Use native name too |
|  |  |  |
| english | Id, label, webpage, update | Labels. Other languages will use id here as foreign key.  Webpage is id of table webpages. |
| Turkish | Englishid, turkish, update |  |
| xcountrylanguage | Countryid, langid |  |
| webpages | Id, path (path+name of the page) |  |
|  |  |  |
|  |  |  |
| DEVICES |  |  |
| mobile | countryId, number, carrierId, dateAdded (date), deviceid, (countryName local only) |  |
| device | id(deviceid), dateAdded, platform, height, width, | * countryId+ deviceId combination will be unique |
|  |  |  |
| xdevicemember | deviceid, memberid, |  |
|  |  |  |
|  |  |  |
| PEOPLE |  |  |
| members | Id (memberid), email, password |  |
| memberDetails | Memberid, legalname, age, language, countryid, telnumber, dateAdded (date), status, notes, joker, settings |  |
| attorneys | Memberid(requester), name, countryid, mobilenumber, email, pass, relation, dateAdded, languages, note | Languages: comma seperated |
| Users (local storage) | numerickey, memberid |  |
|  |  |  |
|  |  |  |
| ACTIONS |  |  |
| visitors | deviceid, time, lat, lng |  |
|  |  |  |
| orders | Id(Orderid ), deviceid, memberid, duration, interval, startTime, startLat, startLng, endTime, endLat, endLng | Duration (max **65535** second ( 18 hrs)),  Interval |
| tracking | OrderId, lat, lng, timePassed |  |
|  |  |  |
| failedtokens | Time, deviceid, lat, lng | this table is for security |
|  |  |  |

Ipv4: varchar 19+

IPv6:varchar 39+

Lattitude, longtitude,

**float(10,6)** is just fine.

Any other convoluted storage schemes will require more translation in and out, and floating-point math is plenty fast.

|  |  |
| --- | --- |
|  | The distance between each degree of **latitude** varies because of the shape of the earth and distance between each degree of **longitude** gets smaller as you get closer to the poles. So let's talk about the equator, where the [distance between each degree](http://en.wikipedia.org/wiki/Longitude#Degree_length) is 110.574km for latitude and 111.320km for longitude.  50ft is 0.01524km, so:   * 0.01524 / 110.574 = 1/7255 of a degree of latitude * 0.01524 / 111.320 = 1/7304 of a degree of longitude   You need four digits of scale, enough to go down to ten-thousandths of a degree, with a total of seven digits of precision.  DECIMAL(7,4) should be plenty for your needs. |

Accuracy versus decimal places at the equator

decimal degrees distance

places

-------------------------------

0 1.0 111 km

1 0.1 11.1 km

2 0.01 1.11 km

3 0.001 111 m

4 0.0001 11.1 m

5 0.00001 1.11 m

6 0.000001 0.111 m

7 0.0000001 1.11 cm

8 0.00000001 1.11 mm

LOCAL STORAGE:

|  |  |  |
| --- | --- | --- |
|  |  |  |
| DEVICES |  |  |
| mobile | countryId, number, carrierId, dateAdded (date), deviceid, (countryName local only) |  |
| device | id(deviceid), dateAdded, platform, height, width, | * countryId+ deviceId combination will be unique |
|  |  |  |
| xdevicemember | deviceid, memberid, |  |
|  |  |  |
|  |  |  |