Full Use Cases Defined

Sign Up:

Basic Flow:

- Users must create their account before using the population information system of an organization.
- Organization verifies user's account by sending verification security code and then make them members.

Alternate Flow:

- Without creating account users can't access any type of population information.

Login:

Basic Flow:

- User can Login with user id and password after making account.
- Organization will allow Users, and they can access the information by using the system.

Alternate Flow:

- If user doesn't have an account, then he/she must create it before login.

Add Population Info:

Basic Flow:

- An organization admin must login first with their admin account.
- Only Organization's admin can Add population information.

Alternate Flow:

- Admin can perform any other operations as well after getting login.

Remove Population Info:

Basic Flow

- An organization admin must login first with their admin account.
- Only Organization's admin can Remove population information.
- Admin will remove population information if he/she finds mistakes or errors in details.

Alternate Flow:

- Admin can perform any other operations as well after getting login.

Update Population Info:

Basic Flow:

- An organization's admin must login first with their admin account.
- Only Organization's admin can Update population information.
- Admin will update population information if he/she thinks that the system needs to change or to be upgraded in details.

Alternate Flow:

- Admin can perform any other operations as well after getting login.
- Admin can update information by itself or can hire others to do so.

View Population Info:

Basic Flow:

- An organization's admin and user must login first with their account.
- Organization's admin and user can view population information.
- User will be able to view information once organization will allow and share.
- Organization have rights to see the activity of users.
- Both users and organization can view information by

    - Country
    - City
    - Region
    - District
    - Language

Alternate Flow:

- Admin and user can perform any other operations as well after getting login.

- User will not be able to access any information if organization restricted the information.

View Population Info by Country:

Basic Flow:

- An organization’s admin and user must login first with their account.
- Organization’s admin and user can view population information by any country.

Alternate Flow:

- Admin and user can perform any other operations as well after getting login.
- User will not be able to access any information if organization restricted the information.

View Population Info by City:

Basic Flow:

- An organization’s admin and user must login first with their account.
- Organization’s admin and user can view population information by any city.

Alternate Flow:

- Admin and user can perform any other operations as well after getting login.
- User will not be able to access any information if organization restricted the information.

View Population Info by Region:

Basic Flow:

- An organization’s admin and user must login first with their account.
- Organization’s admin and user can view population information by any Region.

Alternate Flow:

- Admin and user can perform any other operations as well after getting login.
- User will not be able to access any information if organization restricted the information.

View Population Info by District:

Basic Flow:

- An organization’s admin and user must login first with their account.
- Organization’s admin and user can view population information by any district.

Alternate Flow:

- Admin and user can perform any other operations as well after getting login.
- User will not be able to access any information if organization restricted the information.

View Population Info by Language:

Basic Flow:

- An organization’s admin and user must login first with their account.
- Organization will provide the facility that user can view population information by desired language like: -
    - Chinese
    - English
    - Hindi
    - Spanish
    - Arabic

Alternate Flow:

- Admin and user can perform any other operations as well after getting login.
- User will not be able to access any information if organization doesn't provide language facility.

Reports:

Basic Flow:

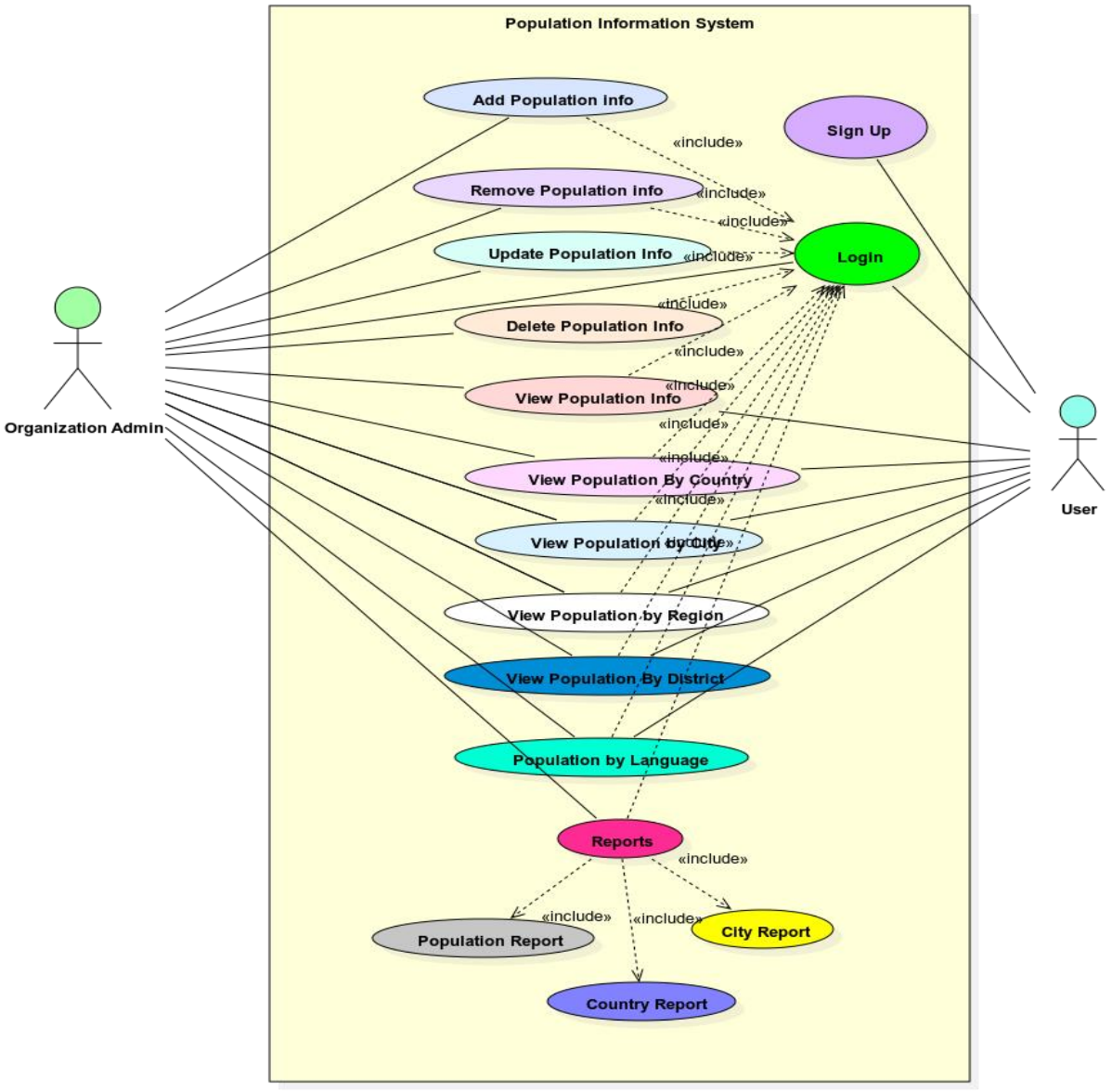
- An organization's admin must login first with their account.
- Admin can make annually, quarterly, and monthly report about population information to check the performance of organization.
- Admin can make report about what manipulations have been done in system.
- Organization's admin will make the report about population information by : -
    - Country
    - City
    - Region

Alternate Flow:

- Admin and user can perform any other operations as well after getting login.
- User will not be able to access any information if organization doesn't provide language facility.

Use Case Diagram

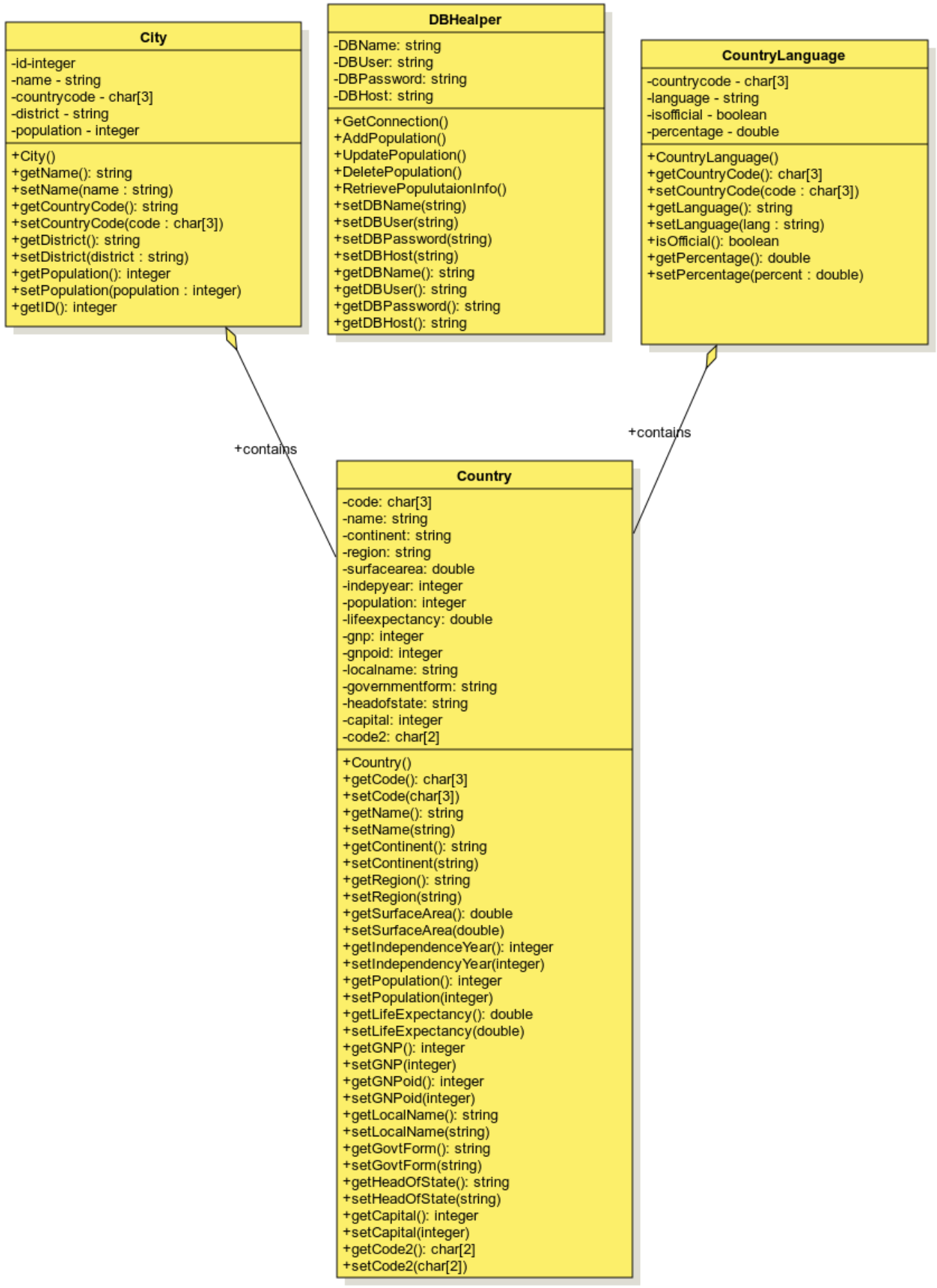
All the use cases are defined for the population information system, the following diagram defines the functional and necessary use cases for the system. System must have functionality of CRUD operations for population information, Login System, Reports, and different views of the population by country, region, continent, city and language. Use cases are performed by the two-stake holder, one organizational admin who has all the permissions and second a user who can view all the information about population for now. Both the stake holders require a login session in order to perform operations.



Class Diagram

Below is a UML class diagram of the basic system. More classes could be added as the system grows in development, but the following classes are necessary in order for the system to work.

- City – class is an object mapping from table ‘city’ and useful in ‘object’ mapping in the system, database and relationships.
- Country Language – similarly, country language is an object mapping from table ‘country language’ and useful in object mapping in the system, database and relationships
- Country – It is also an object mapping from table ‘country’ and is helpful in the system for object mapping and relationship.
- DB Helper – This class manages the connection to database and has all the functions for crud operations on database tables. It is helpful in generating reports and different views.


City
-id-integer
-name - string
-countrycode - char[3]
-district - string
-population - integer
+City()
+getName(): string
+setName(name : string)
+getCountryCode(): string
+setCountryCode(code : char[3])
+getDistrict(): string
+setDistrict(district : string)
+getPopulation(): integer
+setPopulation(population : integer)
+getID(): integer
DBHealper
-DBName: string
-DBUser: string
-DBPassword: string
-DBHost: string
+GetConnection()
+AddPopulation()
+UpdatePopulation()
+DeletePopulation()
+RetrievePopulutaionInfo()
+setDBName(string)
+setDBUser(string)
+setDBPassword(string)
+setDBHost(string)
+getDBName(): string
+getDBUser(): string
+getDBPassword(): string
+getDBHost(): string
CountryLanguage
-countrycode - char[3]
-language - string
-isofficial - boolean
-percentage - double
+CountryLanguage()
+getCountryCode(): char[3]
+setCountryCode(code : char[3])
+getLanguage(): string
+setLanguage(lang : string)
+isOfficial(): boolean
+getPercentage(): double
+setPercentage(percent : double)
+contains
+contains
Country
-code: char[3]
-name: string
-continent: string
-region: string
-surfacearea: double
-indepyear: integer
-population: integer
-lifeexpectancy: double
-gnp: integer
-gnpoid: integer
-localname: string
-governmentform: string
-headofstate: string
-capital: integer
-code2: char[2]
+Country()
+getCode(): char[3]
+setCode(char[3])
+getName(): string
+setName(string)
+getContinent(): string
+setContinent(string)
+getRegion(): string
+setRegion(string)
+getSurfaceArea(): double
+setSurfaceArea(double)
+getIndependenceYear(): integer
+setIndependencyYear(integer)
+getPopulation(): integer
+setPopulation(integer)
+getLifeExpectancy(): double
+setLifeExpectancy(double)
+getGNP(): integer
+setGNP(integer)
+getGNPoid(): integer
+setGNPoid(integer)
+getLocalName(): string
+setLocalName(string)
+getGovtForm(): string
+setGovtForm(string)
+getHeadOfState(): string
+setHeadOfState(string)
+getCapital(): integer
+setCapital(integer)
+getCode2(): char[2]
+setCode2(char[2])