

COVID-19 Analysis Model

CS2212-W20

created by Group-3

1.1 Purpose

This document details the requirements of the system COVID-19 Analysis Model.

This document details the requirements of a system that allows for retrieving COVID-19 infection related data for a selected set of countries. It details the use case scenarios and the actors present in the system that is to be developed. It also addresses non-functional requirements of the system, and the distribution of activities required to complete development of the system.

1.2 Overview

Provide a brief overview of the problem to be solved and the requirements of the system. This section should be a brief executive summary.

There are several requirements that need to be met for the system to function. The system must be able to retrieve data from outside sources for the metadata and compile the information that will be used in the analysis tasks, outlined in the requirements. The system also needs to have a security measure when logging in and out of the application as well as freedom for the users to modify the analysis algorithm for their specific needs. The system does require its code to be modular for it to be used in many scenarios where it may retrieve documents from an unknown API or a local text file depending on the situation. As for the finalization, the code must be simplistic and easy to understand, examples from the course lectures should be implemented in order to provide sound understanding of the course material if possible.

1.3 References

Include references to other documents that may assist in the understanding of this document. (ie. Project Plan)

Project Description - <https://owl.uwo.ca/access/lessonbuilder/item/147446358/group/c9cc47fe-7c6c-49b0-9dd5-247a2977f123/Project%20Resources/CS2212A-Project-Description.pdf>

Covid 19 confirmed cases API - <https://api.covid19api.com/total/dayone/country/%25s/status/confirmed>

GANTT: <https://www.lucidchart.com/pages/examples/gantt-chart-maker>

Domain Model: <https://www.lucidchart.com/>

2 Business Scenario Model

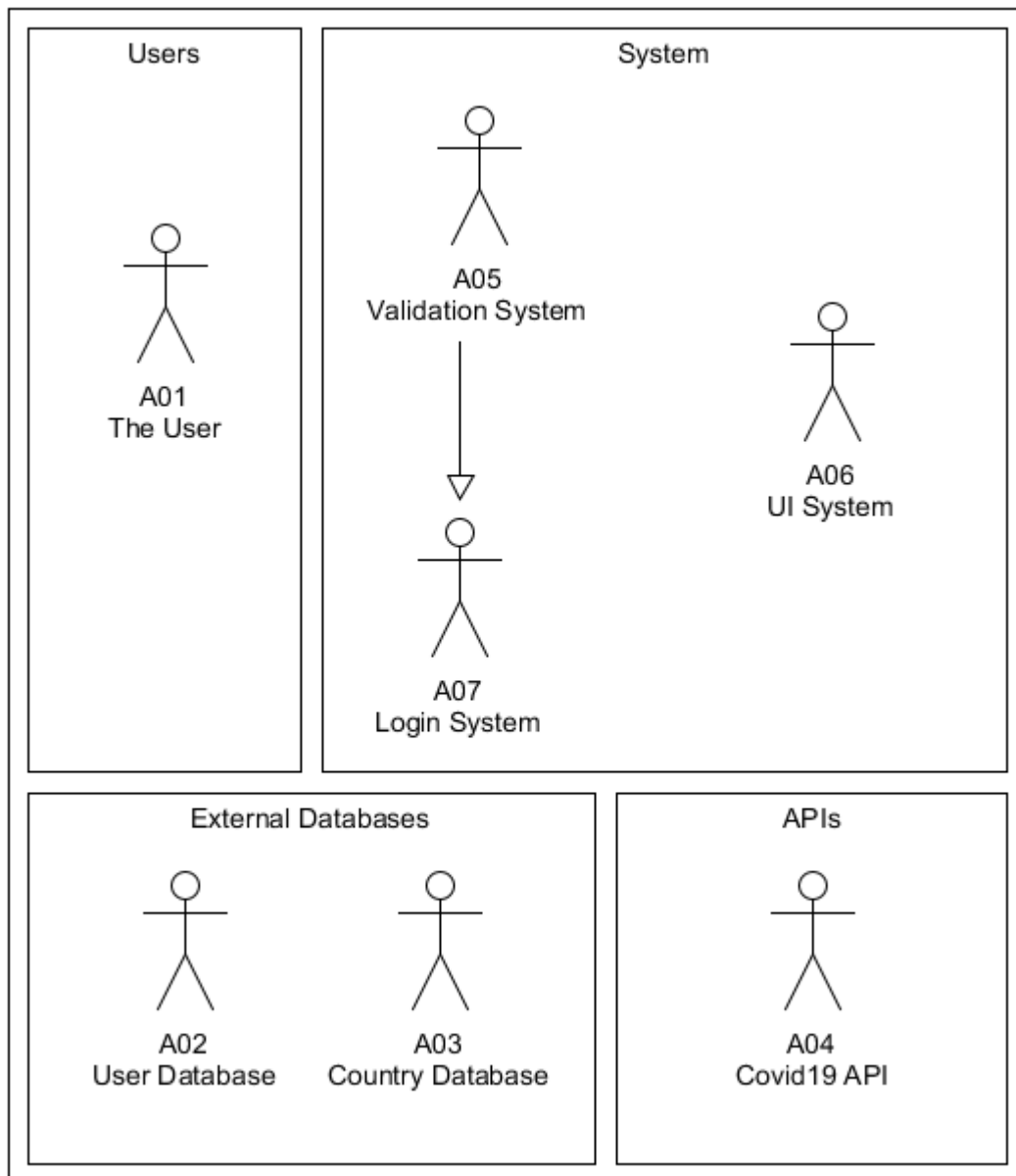
2.1 Actors

2.1.1 Overview

The actors in our system include the user of the application, the userdata base, the country database, and the Covid19 API. The user is able to interact with the system if they have valid login credentials. The User database is an external entity that consists of the login credentials of users. The Country database is an external entity that contains data on the countries in it. The Covid19 API is an interface that allows the system to get country specific covid cases data.

2.1.2 Actor Diagram

The figure below represents the actors of our system. They are grouped based on their interactions with the system. These groups are: Users, Databases, and APIs.



2.1.3 Actor Descriptions

A01: The User

Description	The User actively interacts with the system. The user is able to: log in to the system, add and remove countries, select a type of analysis, and to initiate the analysis on the added countries.
Aliases	None.
Inherits	None.
Actor Type	This is a mainly active human actor. (Active when interfacing with system UI)
Relationships	None.

A02: User Database

Description	The user database system stores information pertaining to the login credentials of users of the system. The system validates with the user database that the credentials are correct to allow the user access to the main application.
Aliases	User database.
Inherits	None.
Actor Type	Passive external system.
Relationships	None.

A03: Country Database

Description	The Country Database System stores the countries that are available for the user to add and information about those countries.
Aliases	None.
Inherits	None.
Actor Type	Passive external system.
Relationships	None.

A04: Covid19 API

Description	The Covid19 API provides covid-19 country data and statistics which is used for analysis
Aliases	None.
Inherits	None.
Actor Type	Passive/active external system (active when using API to get data).
Relationships	None.

A05: Validation System

Description	The validation system is to scan and determine whether the input data coming into the system matches with the accessible databases that hold the user, country, COVID-19 spread/impact information.
Aliases	None.
Inherits	None.
Actor Type	Passive/active external system (active when using API to get data).
Relationships	Login System.

A07: Login System

Description	The login system is a sub-system of the validation system that used to verify credentials of users trying to access to the COVID-19 UI
Aliases	None.
Inherits	Inherits from validation system
Actor Type	Passive/active external system (active when using API to get data, when a user enters their credentials).
Relationships	Validation system

A06: UI System

Description	The Covid19 API provides covid-19 country data and statistics which is used for analysis
Aliases	None.
Inherits	None.
Actor Type	Passive/active external system (active when using API to get data).
Relationships	None.

2.2 Use Case Descriptions

2.2.1 The user logs into the system

Name	UC1: The user logs into the system
Goal in Context	This use case is related to the user logging in to the system through a login UI using a username name password.
Primary actor	The user
Secondary actors	Login System, User Database
Preconditions	The User has access to the system, and the system must be running.
Trigger	When the system is started.
Scenario text	<ol style="list-style-type: none">1. User inputs their username and password<ol style="list-style-type: none">1.1. Username and password are validated with User Database by the Login System2. If username and password are correct the user is granted access to the UI and transferred from the login panel.<ol style="list-style-type: none">2.1. The main UI of the application is displayed
Alternative scenario courses	Following from step 1.1: If the user does not exist or the username and password are incorrect then a popup window will notify the user there was an error and the application will terminate.
Constraints	User credentials must be correct.
Questions	none.

2.2.2 Adding new countries to the list of countries

Name	UC2: Adding new countries to the list of countries to perform analysis of
Goal in Context	This use case is related to the user adding a country to the country list
Primary actor	The user
Secondary actors	Country database, System(Specifically the subsystem in charge of country database)
Preconditions	User has successfully logged into the main application UI.
Trigger	User wants to add a country to the analysis list.
Scenario text	<ol style="list-style-type: none">1. User Types in country name / selects country from dropdown list2. User selects '+' button to add country<ol style="list-style-type: none">2.1. The user input is validated by checking if the country is in the Country Database3. The country is displayed in a panel of the list of countries to be analyzed
Alternative scenario courses	Following step 2.1, if the country is not in the database of valid countries then an error message is displayed.
Constraints	For a regular course of action, the country added by the user must be in the database of known countries.

Questions	Is there any limitations for duplication of the countries, also is there a size limit for the list and if not will there be used a recursive sizeof(example) to multiply the size of the list.
------------------	--

2.2.3 Removing countries from the list

Name	UC3: Removing countries from the list of countries to perform analysis of
Goal in Context	This use case refers to removing countries from the list of selected countries for analysis
Primary actor	The user
Secondary actors	Country Database, System (Specifically the subsystem in charge of country database)
Preconditions	User has successfully added a country to the list of countries to analyze
Trigger	User wants to remove a country from the analysis list
Scenario text	<ol style="list-style-type: none"> 1. User Types in country name / selects country from dropdown list 2. User selects '-' button to remove country <ol style="list-style-type: none"> 2.1 The user selection is validated by checking that the country is in the list of selected countries 2.2 The user input is validated by checking that the country is in the country database 3. The country is removed from the list of selected countries for analysis
Alternative scenario courses	<p>Following step 2.1, if the country is not in the selected list of countries then an error message is displayed.</p> <p>Following step 2.2, if the country is not in the database of valid countries then an error message is displayed.</p>
Constraints	For a regular course of action, the country removed by the user must be in the database of known countries and a country in the list of selected countries
Questions	Will there be an error once the list of countries are empty.

2.2.4 Selecting the analysis type

Name	UC4: Selecting the analysis type to be performed
Goal in Context	This use case refers to selecting an analysis type for a list of one or more countries
Primary actor	The user
Secondary actors	Analysis system
Preconditions	User has added countries to the list of countries selected for analysis
Trigger	User wants to select which analysis type will be executed.
Scenario text	<ol style="list-style-type: none"> 1. User clicks on the arrow expanding the dropdown list <ol style="list-style-type: none"> 1.1 User chooses analysis type to be performed from the available options 2. The system receives the choice of analysis <ol style="list-style-type: none"> 2.1. The system validates the request

	<p>2.1.1. If the request returns false, the request is terminated and the user is returned back to choosing from the dropdown list.</p> <p>2.2. The decision is finalized using a setter method</p> <p>3. The analysis type choice is completed</p>
Alternative scenario courses	None.
Constraints	User will be granted to perform only one of the 4 analysis types, if the user wants to perform a different/repetition of the analysis they would need to re-select their option.
Questions	If the country list is empty, will the user still have freedom to choose an option.

2.2.5 Performing the analysis

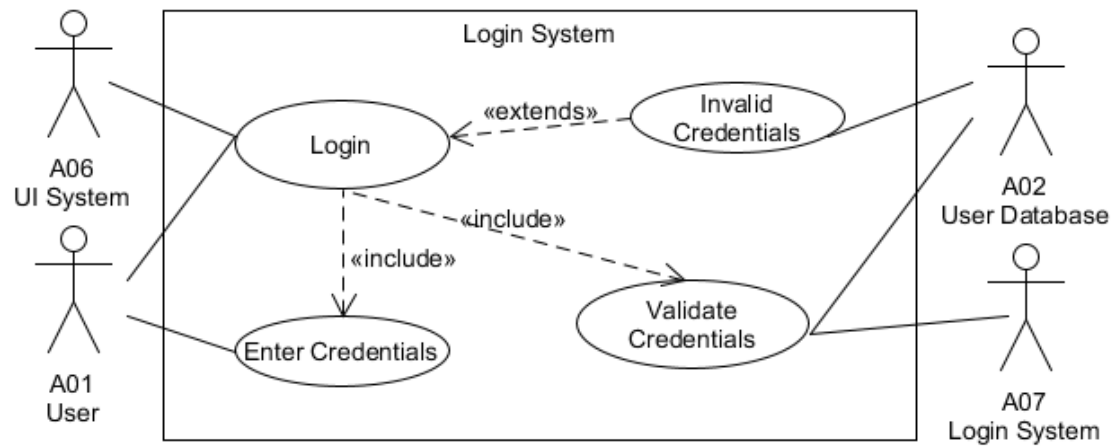
Name	UC5: Performing the analysis
Goal in Context	This use case refers to performing a user specified analysis on a list of countries to compile usable data regarding COVID-19 data for specified countries demographic information.
Primary actor	The User
Secondary actors	Covid19 API, Country Database, UI
Preconditions	User has selected the type of analysis they want to perform.
Trigger	The user wants to initiate an analysis.
Scenario text	<ol style="list-style-type: none"> 1. The user presses “Recalculate” button <ol style="list-style-type: none"> 1.1. Call to Analysis System 1.2. Initiate selected analysis function 1.3. Analysis System accesses Covid19 API I to retrieve COVID-19 data for the list of countries <ol style="list-style-type: none"> 1.3.1. The system will initiate a validation for the accuracy of the COVID-19 data that is retrieved, whether it can be used for the analysis. 1.4. Analysis System accesses Country database to retrieve for information regarding age/population/sexuality etc. <ol style="list-style-type: none"> 1.4.1. The system will initiate a validation for the accuracy of the data that is retrieved, whether it can be used for the analysis. 1.5. analysis on data is performed by analysis function
Alternative scenario courses	Following from step 1.3.1 & 1.4.1: If some or all of the data needed to compute the analysis is unavailable a message will be displayed to the user.
Constraints	The used Database information must be accurate, and be accessible in any IP location.
Questions	

2.2.6 Displaying the results

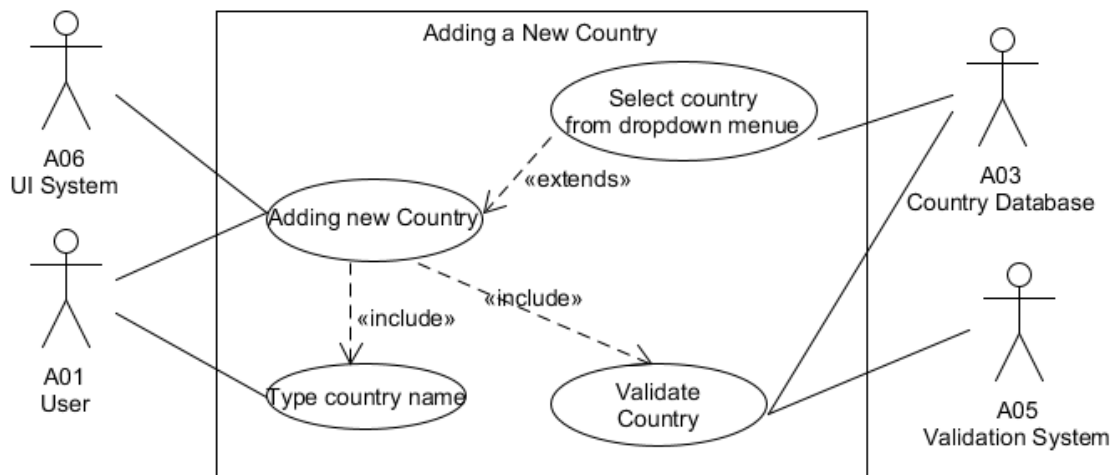
Name	Displaying the results
Goal in Context	To display the results of the analysis in the output as well as a visual representation of the COVID-19 spread data.
Primary actor	UI System
Secondary actors	N/A
Preconditions	The calculations of the database information are compiled properly with accurate data
Trigger	As soon as the analysis functions are completed with their results. The result display is invoked.
Scenario text	<ol style="list-style-type: none">1. After the results are fed into the system, the UI displays results<ol style="list-style-type: none">1.1. Visual Map information is displayed regarding the chosen countries1.2. Output menu is filled with the logistic information regarding the chosen countries
Alternative scenario courses	<p>If the database information was empty regarding the chosen countries there would be no data displayed on both of the parameters of the use case(Map/Output menu).</p> <p>If there were no countries chosen, the display would result in no information being displayed similar to the previous alternative scenario</p>
Constraints	The only information displayed can only be dependent on the user choice of countries as well as the analysis type.
Questions	What would happen if the databases used did not match with the countries that were mentioned? Would there be a failsafe for that possibility.

2.3 Use Case Diagrams

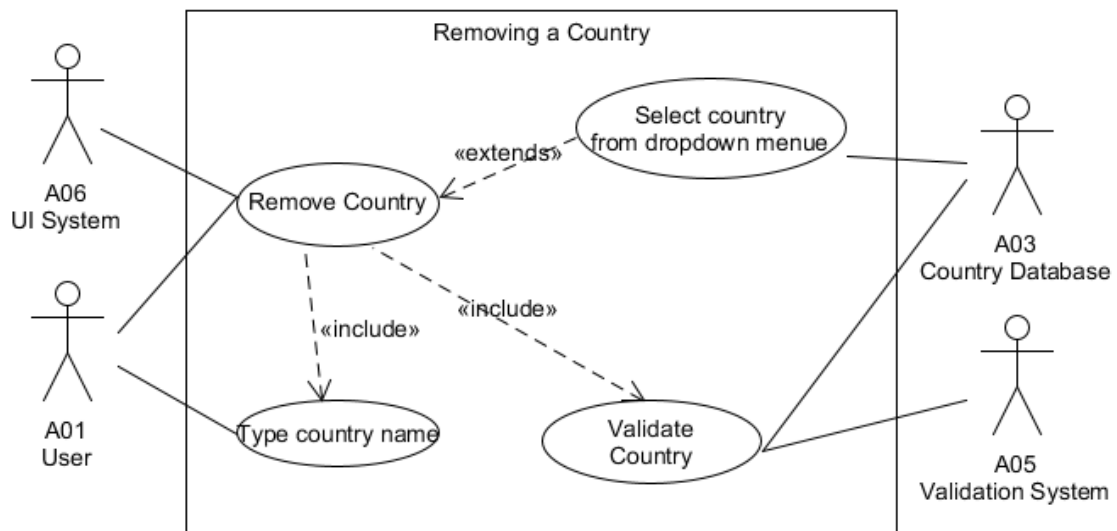
2.3.1 The user logs into the system



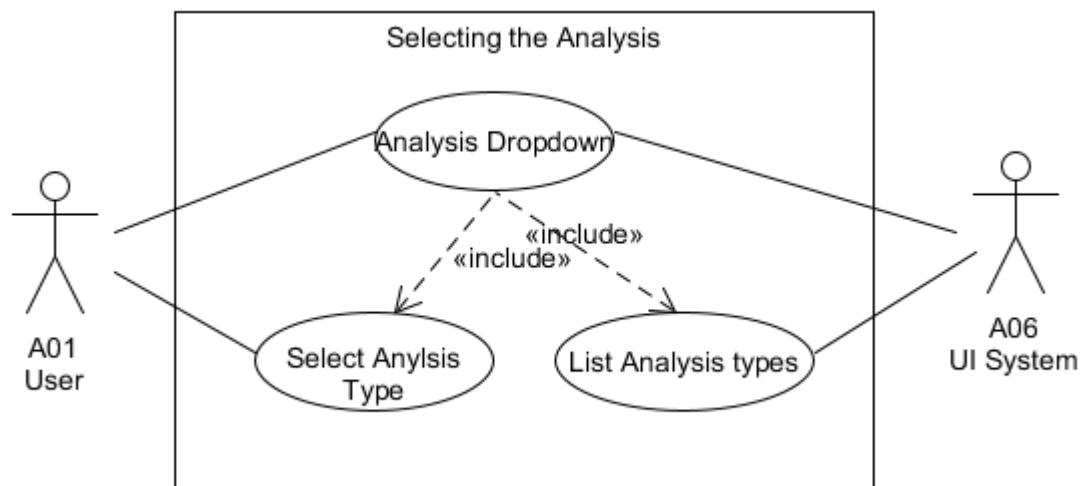
2.3.2 Adding new countries to the list of countries



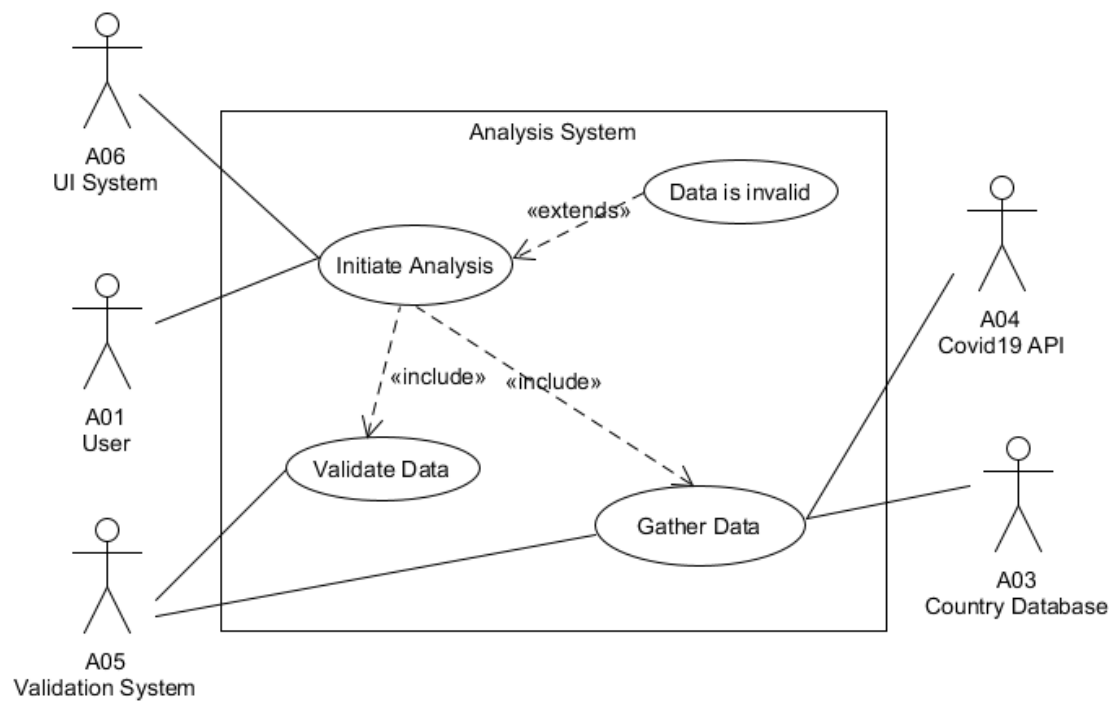
2.3.3 Removing countries from the list



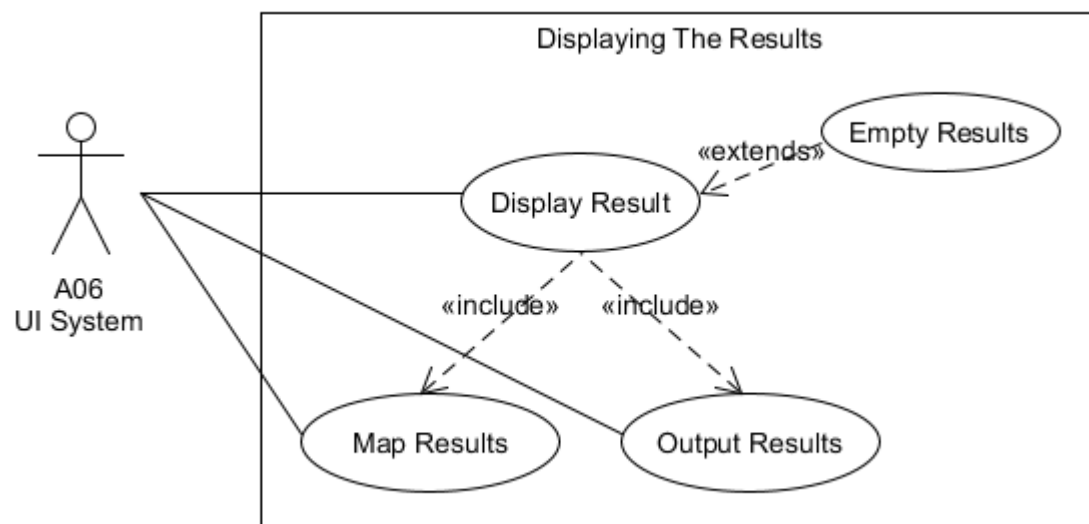
2.3.4 Selecting the analysis type



2.3.5 Performing the analysis

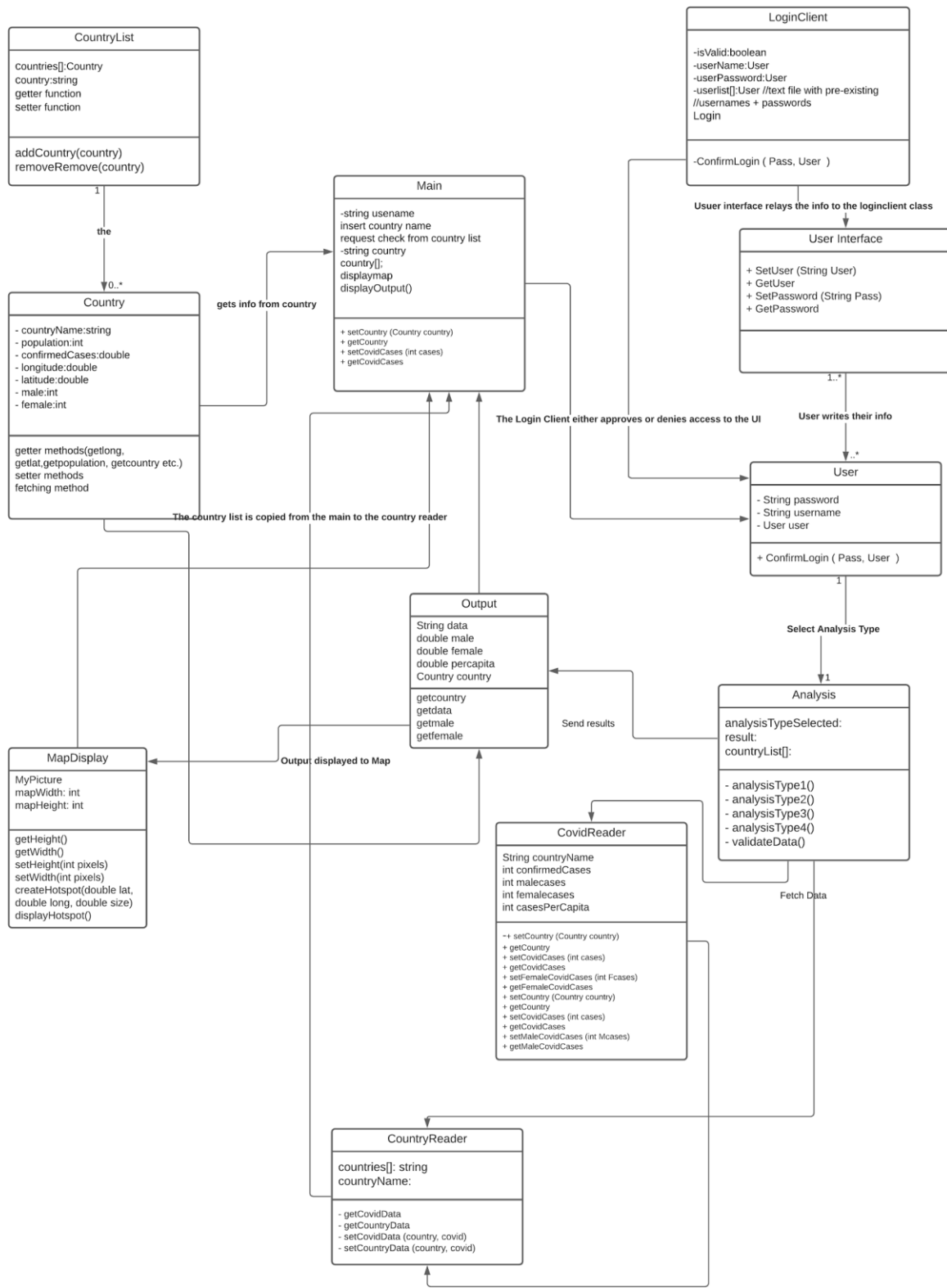


2.2.6 Displaying the results



3 Domain Model

3.1 Domain Model Class Diagram



3.2 Domain Model Class Descriptions

Class name	User
Description	A class to hold user information to verify
Attributes	user:User
Responsibilities	A class to represent the current user, that will hold information. The user attribute will be linked to the login class for its required information to have access to the UI.
Business Rules	

Class name	Country
Description	
Attributes	countryName:string population:int confirmedCases:double longitude:double latitude:double male:int female:int getter methods(getlong, getlat,getpopulation, getcountry etc.) setter methods fetching method
Responsibilities	Assigning attributes to a specific country name that matches with country name with the country databases to fill up the information to locate it on the Map as well as have demographic information to relay back into the visual as well as mathematical data in the UI
Business Rules	

Class name	Analysis
Description	This class connects with reader classes to get data required to perform different analysis types on the, it then calculates the analysis and returns the result to the mapDisplay class.
Attributes	analysisTypeSelected: result: countryList[]:
Responsibilities	analysisType1 analysisType2 analysisType3 analysisType4 getCountryList validateData
Business Rules	

Class name	LoginClient
Description	The login client validates the user's credentials.
Attributes	isValid:boolean userName:User userPassword:User userlist[]:User //text file with pre-existing usernames + passwords
Responsibilities	validateCredentials, allows users to gain access to the UI
Business Rules	

Class name	CovidReader
Description	Uses Covid19 API to fetch number of cases for a particular country
Attributes	countryName: confirmedCases: maleCases: femaleCases: casesPerCapita:
Responsibilities	storing covid-19 cases for the specific name, it will be used to calculate the analysis and to display the information. Will require a fetch method to scan through the country list to mirror the information into the attributes mentioned above.
Business Rules	

Class name	CountryList
Description	Class to store information about a specific country.
Attributes	countries[]:Country country:string getter function setter function
Responsibilities	The purpose of this class is to visually show the chosen countries that will be used in the specified analysis type. The countries names will be displayed on their own specialized infobox
Business Rules	

Class name	CountryReader
Description	Class that stores every valid country names in the provided databases to validate the added country names for the analysis list of countries
Attributes	countries[]:string countryName:Country return T/F getName(string);
Responsibilities	Verifies the chosen country to know whether it is valid from the country database, the class is used for getting the information from the API/databases using a fetch method to find the coinciding name from the list. Once found it will return true, if not false
Business Rules	

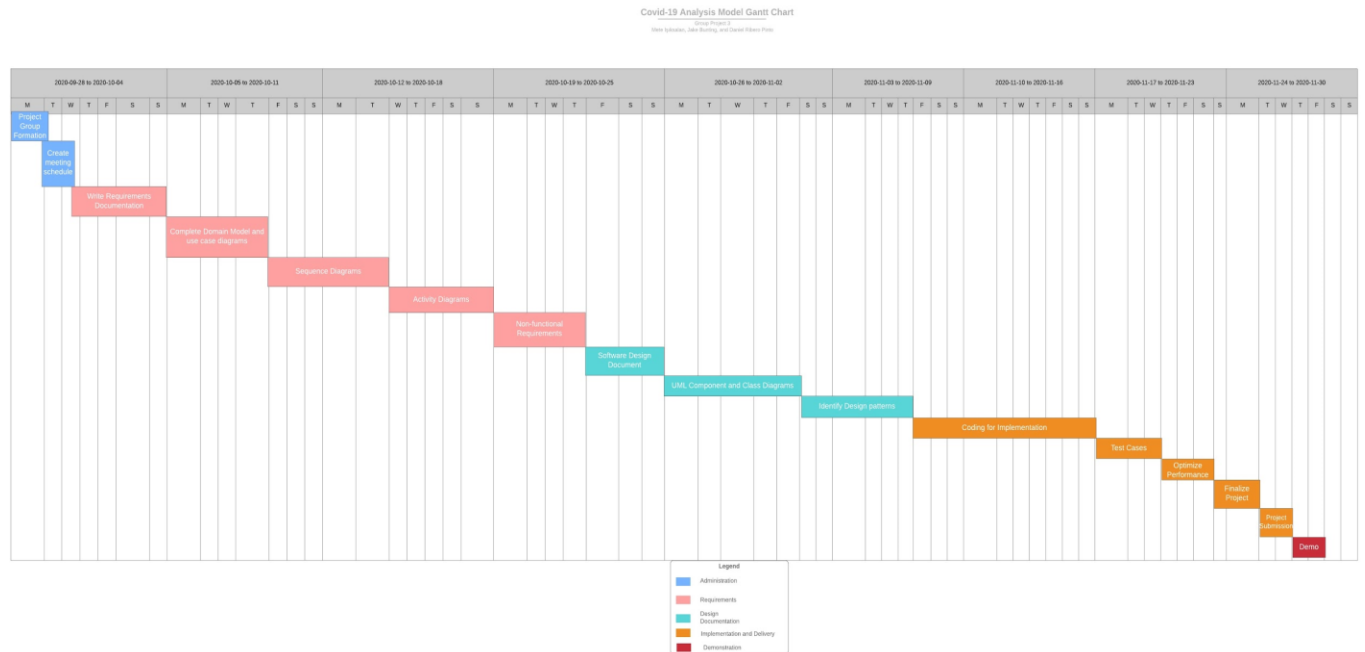
Class name	MapDisplay
Description	Class to display information to the user onto the world map
Attributes	myPicture: mapWidth: int mapHeight: int getHeight(): function getWidth(): function setHeight(): function setWidth(): function createHotspot(): function displayHotspot(): function
Responsibilities	The class is to visually show the hotspots of the COVID-19 of the pre-chosen countries from the available list. It will display information in coloured circles on countries that were chosen, the size of the circular representation will demonstrate the amount of confirmed COVID-19 cases
Business Rules	

Class name	Output
Description	Class to display information to the user on the output window
Attributes	displayOutput: function data: string male: double female: double percaita: double country: Country getCountry(): function getData(): function getMale(): function getFemale(): function
Responsibilities	The output infobox will display demographic information regarding each of the countries depending on the type of analysis performed with the UI. It can display data for confirmed cases in males/females from every country that was included in the analysis etc.
Business Rules	

Class name	Main
Description	Class to implements all the fetching information and performs the required initilization of all the classes, the main body of the UI
Attributes	
Responsibilities	To initialize the country list, to call class functions to validate info, save, execute and initiate every aspect of the overall classes to display the required data into the UI.
Business Rules	

6 Activities Plan

Gantt Chart



Product Backlog

Task	Estimated Hours
Create UI	4
Create Login system	6
Create analysis functions	4
Make Test cases	3
Create Reader classes	1

6.3 Group Meeting Logs

In this Section you write minutes of each meeting, listing the attendance, what the topics of discussion in the meeting were, any decisions that were made, and which team members were assigned which tasks. These minutes must be submitted with the project report in each deliverable and will provide input to be used for the overall assessment of the project.

Present Group Members	Meeting date/length	Issues Discussed/Resolved
All current members	Oct 4th , 2020/90 minutes	split up work for first deliverable
All current members	Oct 5th, 2020/45 minutes	went over use case descriptions
All current members	Oct 7th, 2020/60 minutes	went over actor descriptions, use case diagrams, and gantt chart
All current members	Oct 8th, 2020/	went over class definition, creation of domain model, and product backlog