Gebze Teknik Üniversitesi

Bilgisayar Mühendisliği

Object Oriented Analysis and Design

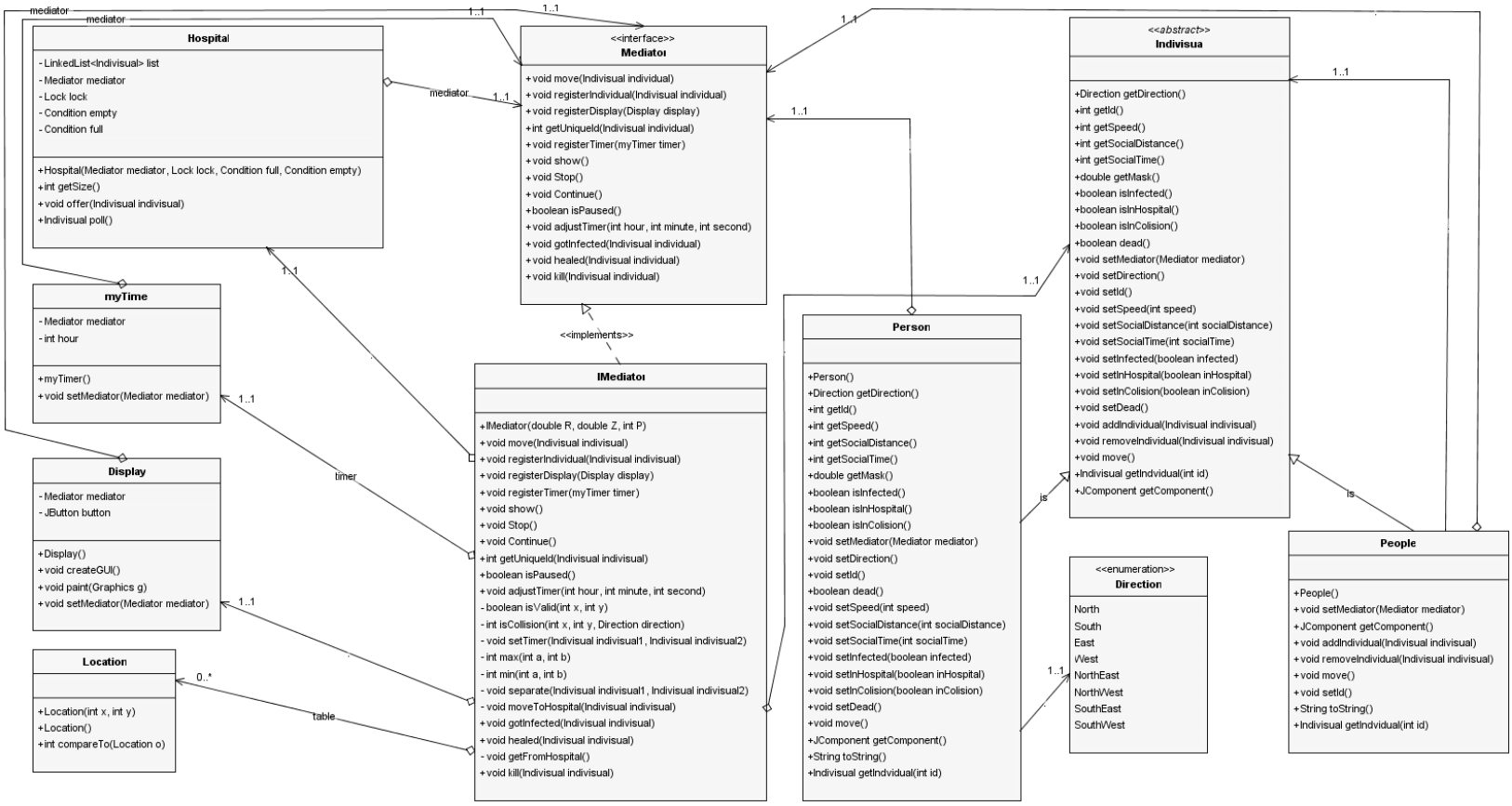
CSE 443

#final project

Jwan hussein

151044078

class diagram:



The individual model was built following the composite pattern.

Since the individual should be able to be added in bulk as well as one by one, and the composite pattern allow us to build structures of object in the form trees that contain both composition of object and individual objects as nodes, furthermore the composite pattern allow us to apply the same operation over both composite and individual object ignoring the actual type of the object.

The **Individual** abstract class represent the ancestor for both **Person** class and **People** class, it provides default implementation for all method .

The **People** class represent the composite .

The Person class represent the leaf.

The **mediator pattern** was used to centralize the communication and control in the simulator, all object used in the program contacts using the mediator class .

Scenario of person movement:

1. The Person object calls move() method of the Individual class.
2. The move() method calls the move() method of the mediator class
3. The move() method of the mediator moves the individual then refresh the frame , in case that the movement caused a collision with other individual then stops the both individuals and sets a timer to separate them after certain amount of time.
4. at the separation phase if one of the individuals caused the other one to get infected, then a timer will be set to move the individual to the hospital after 25 sec and another timer will be set to kill the individual in case the individual was not moved to the hospital.

The **Hospital class:**

Has a Queue (**FIFO**), after 25 seconds of individual got infected and the individual is not already dead the timer calls the moveToHospital() method of the mediator class , which represent the **producer** in its turn, then moveToHospital() method calls the offer() method of the Hospital object , then sets another timer to get the individual out of the hospital after 10 seconds, so after the 10 second the timer calls getFromHospital() method of the mediator class , the last call will cause a call to poll() method of the hospital object which represent the **consumer**.

The **Display class:**

represent the frame.

The **Location class:**

represent 2D point.

Individual class operations:

```
public Direction getDirection();
public int getId();
public int getSpeed();
public int getSocialDistance();
public int getSocialTime();
public double getMask();
public boolean isInfected();
public boolean isInHospital();
public boolean isInColision();
public boolean dead();
public void setMediator(Mediator mediator);
public void setDirection();
public void setId();
public void setSpeed(int speed);
public void setSocialDistance(int socialDistance);
public void setSocialTime(int socialTime);
public void setInfected(boolean infected);
public void setInHospital(boolean inHospital);
public void setInColision(boolean inColision);
public void setDead();
public void addIndividual(Indivisual indivisual);
public void removeIndividual(Indivisual indivisual);
public void move();
public Indivisual getIndvidual(int id);
public JComponent getComponent();
```

Mediator class operations:

```
public void move(Indivisual individual);
public void registerIndividual(Indivisual individual);
public void registerDisplay(Display display);
public int getUniqueId(Indivisual individual);
public void registerTimer(myTimer timer);
public void show();
public void Stop();
public void Continue();
public boolean isPaused();
public void adjustTimer(int hour,int minute , int second);
public void gotInfected(Indivisual individual);
public void healed(Indivisual individual);
public void kill(Indivisual individual);
```