+ GoalNode(MapArea goal, int time)

+ MapArea goal

+int time //time to reach goal

GoalNode

+ Passenger(every important thing)

+ MapArea destination, initial

+ String label

+ int wTime, rTime //wait & ride

Passenger

+ MapArea(int x, int y) //random cost

+ MapArea(int x, int y, int cost)

+ printPosition() : void

+ magWith(MapArea des) : int[]

//magnitudeWith another MapArea

+ distanceWith(MapArea des) : int

+ timeWith(MapArea des) : int

+ directionTo(MapArea next) : String

+ int cost

+ final int X, Y //coordinate

+ MapArea north, south, east, west

//connector, initialize with null

MapArea

+ Taxi(MapArea, mapAndTaxi) //Const

+ closeLog() : void

+ moveNorth() : void

+ moveSouth() : void

+ moveEast() : void

+ moveWest() : void

+ fetch() : void //fetch passenger

+ drop() : void //drop passenger

+ getPath() : void //get GPS path

+ moveToGoal() //trigger all move

+ updateTime(int time) : void

+ dynamicCustomer() : Boolean

+ rest() : void //rest when no order

+ final int space = 4

+ private int availableSpace

+ MapArea currentPosition

+ ArrayList<Passenger> booking

+ ArrayList<Passenger> contains

+ Queue<MapArea> GPS

+ int time

+ PrintWriter write

* Write the log book

+ private mapAndTaxi image;

* Connect to the jPanel

Taxi

+main(String[] args)

>> get size, passenger number from user

>> initialize map with MapArea[][] and link them together

>> initialize f = Frame() and taxi = Taxi()

>> for every passenger, user enter label, then auto generate source and destination, store passenger information in taxi’s passenger booking list

>> set f.Visible(true), now the window is pop up

>> while still have passenger, taxi find the shortest goal, move to the goal, fetch/drop passenger

Taxi\_DS

taxi\_ds

taxi\_ds.images1

Frame

+ mapAndTaxi image //jPanel

+ JPanel panel

+ int column, row //taxi lacation at the panel

+ int counter //for animation

+ ImageIcon backgroundImg //mapImage

+ String pngFile //map source file

+ Taxi[] taxiHolder //hold taxi reference

+ MapArea taxiLocation //get taxi location data like cost and etc

+ MapArea[][] map //for generate map purpose

mapAndTaxi

+ Frame() //initialize component and jPanel(mapAndTaxi)

Taxi\_DS(MainClass)

taxi\_ds

Taxi

+ mapAndTaxi(JPanel panel, MapArea[][] map, Taxi[] taxiHolder)

+ paint(Graphics g) : void //Graphics

+ moveNorth()

+ moveSouth()

+ moveEast()

+ moveWest()

+ updateLocation() : void //get taxi location cost

+ trafficEvent() : Boolean //trigger trafficEvent

Flow

* MainClass(Taxi\_DS) run, ask user for size, number of passenger, then initialize map, frame and taxi, ask for label for every passenger, the passenger will be save into taxi booking list
* Then is a while loop, it will run whenever still have passenger

1. Taxi will getPath() to get nearest goal and best path, then save to to queue GPS
2. moveToGoal() follow the path from GPS until reach goal, when taxi move to next location, it will call moveDirection()
   * \* moveDirection() will trigger the animation of the graphics
   * Will check whether environment change (new dynamic passenger && traffic condition), if environment change recalculated GPS
3. drop() – drop passenger if passenger reach go
4. fetch() – fetch passenger if needed