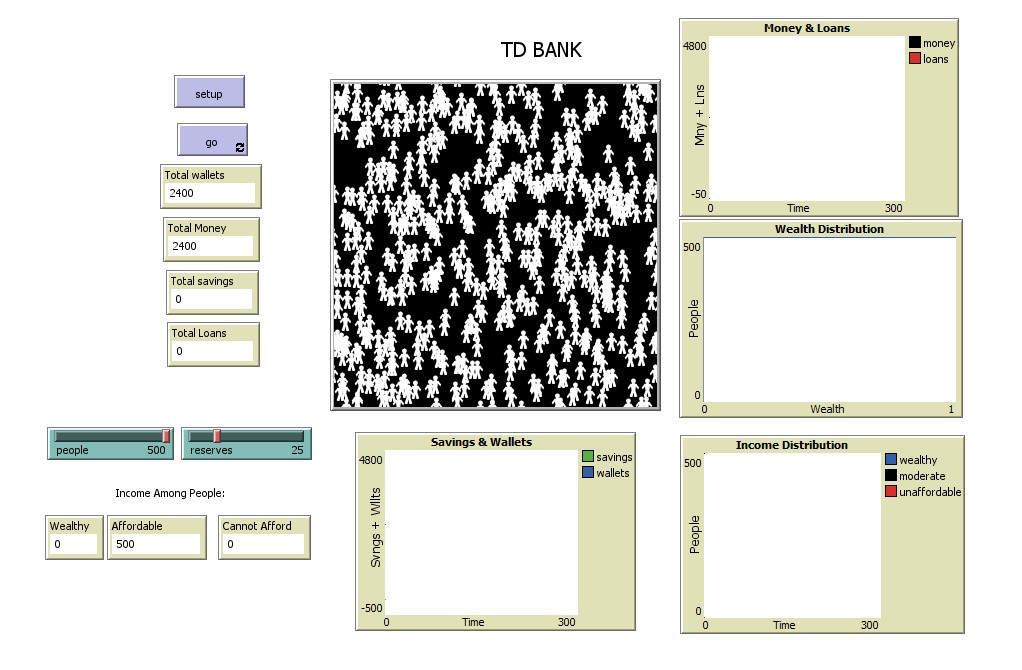
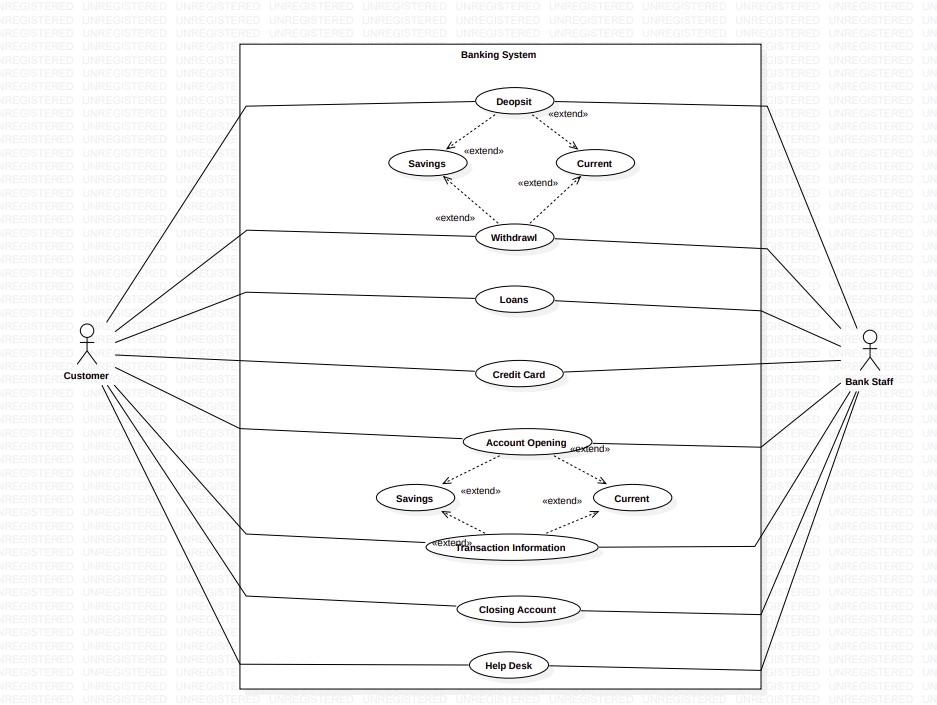
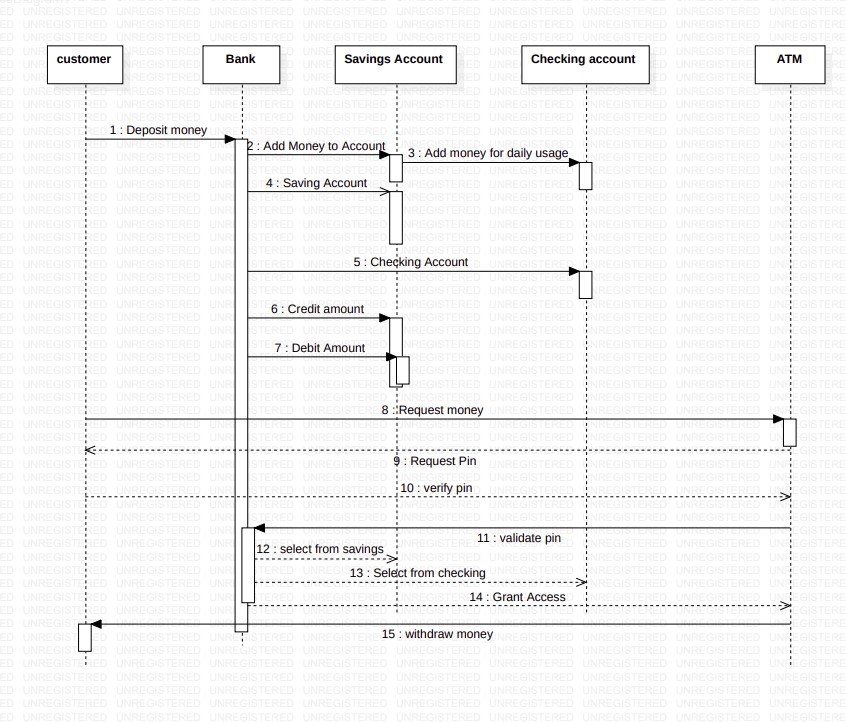
Diagram, schematic

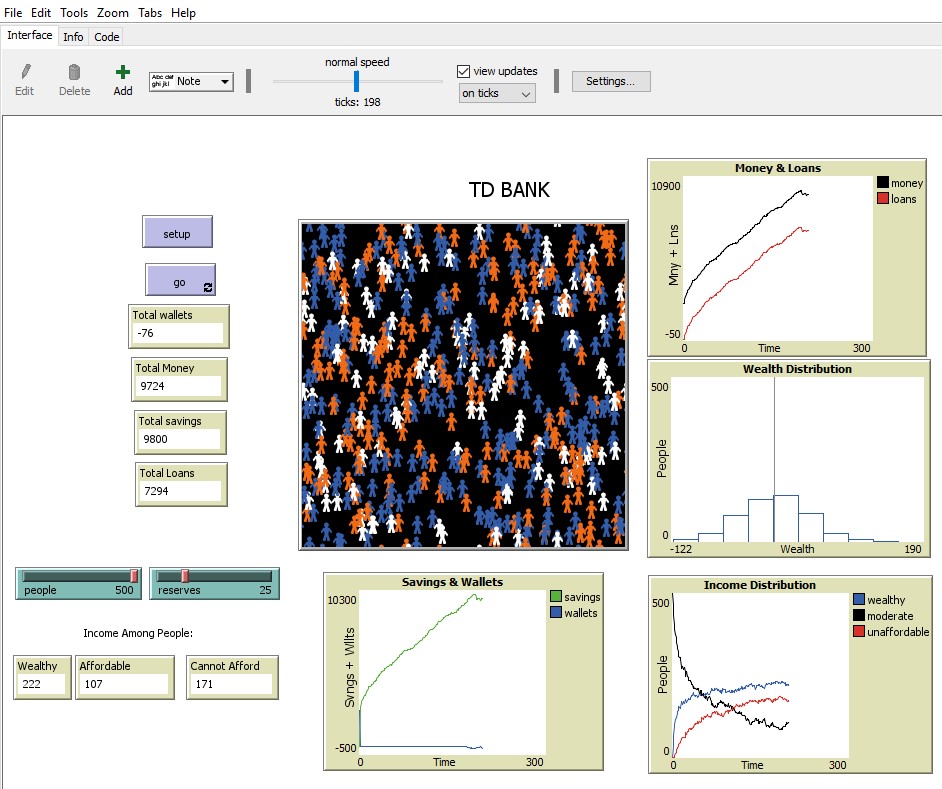
Description automatically generatedDiagram

Description automatically generatedThis model is used to describe the workflow of TD banking system. It is the complete representation of how the cash will be passed through all the stages and project every aspect of the transaction performed by the customer. Graphical user interface, text, application, Word

Description automatically generated

Graphical user interface, application

Description automatically generatedThis model is very simple and easy to understand. It uses basic commands to run the model. It consists of a world and where all the persons and actions can be done. The above model consists of various buttons, sliders and display graphs. These display graphs show the results of the persons activity on how their money is being transferred and how well the customers maintain the bank balances. The SETUP button is used to reset. It returns to the starting state of the model. The GO button is used to start and stop the model and the plotter.



With slider bars we can set total number of people and reserves and compare the model with the other factors to check the income and wealth distribution among the customers with respect to the bank. All these can be done by predefining the commands on to the code and link them to the interface.

The code for the following output is:-

globals [

bankmoney-requested

reservebank

bankdepo

bankloan

xmax

ymax

wealthy

unaffordable

moderate

maxincome

]

turtles-own [

currentlysaving

money-requested

amountinwallet

temporaryloan

wealth

user

]

to setup

clear-all

initialize-settings

create-turtles people [turtlessetup]

classpoll

banksetup

set xmax 500

set ymax (2 \* totalmoney)

reset-ticks

end

to initialize-settings

set wealthy 0

set unaffordable 0

set moderate 0

set maxincome 10

end

to turtlessetup

set shape "person"

setxy random-xcor random-ycor

set amountinwallet (random 2 \* maxincome)

set currentlysaving 0

set money-requested 0

set wealth 0

set user -1

get-color

end

to banksetup

set bankmoney-requested 0

set reservebank 0

set bankdepo 0

set bankloan 0

end

to go

ask turtles [runbusiness]

ask turtles [

remainingbalance-books

get-color

]

summarybalance

classpoll

tick

end

to classpoll

set wealthy (countturtles with [currentlysaving > maxincome])

set unaffordable (countturtles with [money-requested > maxincome])

set moderate (countturtles) - (wealthy + unaffordable)

end

to runbusiness

rt random 360

fd 1

to remainingbalance-books

ifelse (amountinwallet < 0)

[ifelse (currentlysaving >= (- amountinwallet))

[draw-money (- amountinwallet)]

[if (currentlysaving > 0)

[draw-money currentlysaving]

set temporaryloan bankloan

ifelse (temporaryloan >= (- amountinwallet))

[takingloan(- amountinwallet)]

[takingloantemporaryloan]

]

]

[depositiinsavings amountinwallet]

if (money-requested > 0) and (currentlysaving > 0) [

ifelse (currentlysaving >= money-requested)

[draw-money money-requested

payloan money-requested]

[draw-money currentlysaving

payloan amountinwallet]

]

end

to summarybalance

set bankdepo sum [currentlysaving] of turtles

set bankloan sum [money-requested] of turtles

set reservebank ((reserves / 100) \* bankdepo)

set bankloan (bankdepo - (reservebank + bankloan))

end

to depositiinsavings [totalamount]

set amountinwallet (amountinwallet - totalamount)

set currentlysaving (currentlysaving + totalamount)

end

to draw-money [totalamount]

set amountinwallet (amountinwallet + totalamount)

set currentlysaving (currentlysaving - totalamount)

end

to payloan [totalamount]

set money-requested (money-requested - totalamount)

set amountinwallet (amountinwallet - totalamount)

set bankloan (bankloan + totalamount)

end

to takingloan[totalamount]

set money-requested (money-requested + totalamount)

set amountinwallet (amountinwallet + totalamount)

set bankloan (bankloan - totalamount)

end

to get-color

set color white

if (currentlysaving > maxincome) [set color blue]

if (money-requested > maxincome) [set color orange]

set wealth (currentlysaving - money-requested)

end

to-report currentlysaving-total

report sum [currentlysaving] of turtles

end

to-report money-requested-total

report sum [money-requested] of turtles

end

to-report totalamountinwallets

report sum [amountinwallet] of turtles

end

to-report totalmoney

report sum [amountinwallet + currentlysaving] of turtles

end