Practical No. 1

Question 1:

1. Design and Develop Agent Based Model by:
    - Creating the agent population
    - Defining the agent behaviour
    - Add a chart to visualise the model output

Agent Based Modelling [Market Model], Agent-Based Modelling of Market Dynamics

Agent-based modelling (ABM) provides a powerful framework for understanding complex systems such as financial markets. In this study, we develop an ABM to simulate market dynamics, aiming to analyse the emergence of price trends, trading volume, and market stability.

Introduction:

Financial markets are complex systems characterized by the interaction of heterogeneous agents. Understanding the dynamics of these markets is crucial for investors, policymakers, and regulators. ABM offers a bottom-up approach to modelling market behaviour, where individual agents, representing traders, interact within an environment, leading to emergent market phenomena. The aim of this study is to develop an agent-based model to simulate market dynamics and analyse the emergence of price trends, trading volume, and market stability.

Procedure:

1. Agent Population:

- Agent Attributes: Each trader agent is characterized by attributes such as cash, holdings, trading strategy, risk tolerance, and market sentiment.
- Initialization: Traders are randomly distributed in the market with initial cash and holdings.
- Interaction: Traders interact with each other and the market based on predefined rules governing trading behaviour, such as buying, selling, setting limit orders, and changing trading strategies.

2. Agent Behaviour:

Trading Behaviour: Agents make trading decisions based on their trading strategies, market conditions, and the behaviour of other agents.

Market Impact: Trading decisions by agents affect market prices, trading volume, and market liquidity.

3. Model Output Visualization:

Chart: Line charts are used to visualize various market metrics over time, such as price trends, trading volume, market liquidity, and trader sentiment.

Steps:

Sure, here are the steps to create an agent-based model of financial markets using AnyLogic software:

1. Setting up the Agent Population:

- Open AnyLogic and create a new project.

- Create an Agent Population representing traders:

  - Define agent attributes such as cash, holdings, trading strategy, risk tolerance, and market sentiment.

  - Initialize the agent population by randomly distributing traders in the market with initial cash and holdings.

  - Define interactions between agents and the market based on predefined rules governing trading behaviour:

    - Agents make trading decisions (buying, selling, setting limit orders) based on their trading strategies, market conditions, and the behaviour of other agents.

2. Agent Behaviour:

- Define the trading behaviour of agents:

  - Define trading strategies for agents based on factors such as technical indicators, fundamental analysis, and market sentiment.

  - Implement trading decision-making algorithms for agents.

  - Define how trading decisions by agents affect market prices, trading volume, and market liquidity.

3. Model Output Visualization:

- Create line charts to visualize various market metrics over time:

  - Price trends: Plot market prices over time.

  - Trading volume: Plot trading volume over time.

  - Market liquidity: Plot bid-ask spreads or market depth over time.

  - Trader sentiment: Plot sentiment indicators over time.

4. Simulation Setup and Execution:

- Set simulation parameters such as simulation duration, time step, and initial market conditions.

- Run the simulation to observe the emergence of price trends, trading volume, market liquidity, and trader sentiment over time.

5. Analysis and Interpretation:

- Analyse the simulation results to gain insights into market dynamics:

  - Evaluate the impact of different trading strategies on market behaviour.

  - Examine the relationship between trading volume, market liquidity, and price trends.

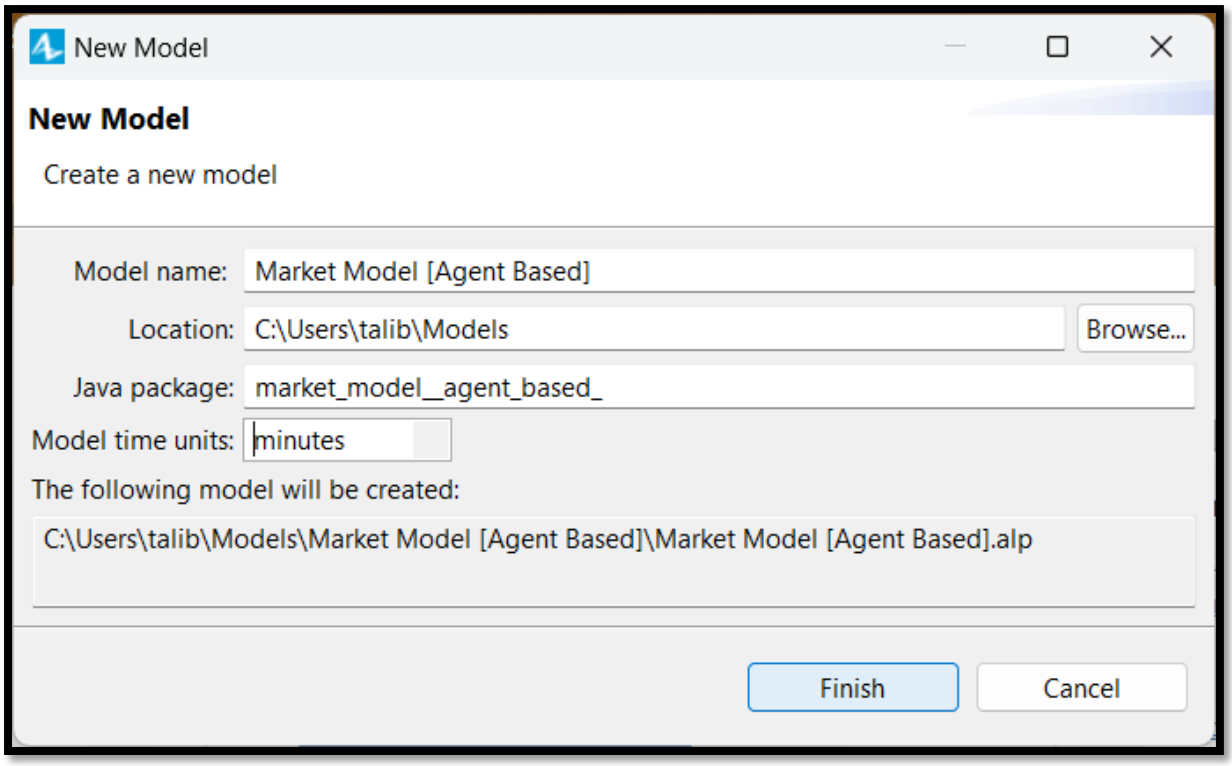
  - Assess the influence of trader sentiment on market outcomes.

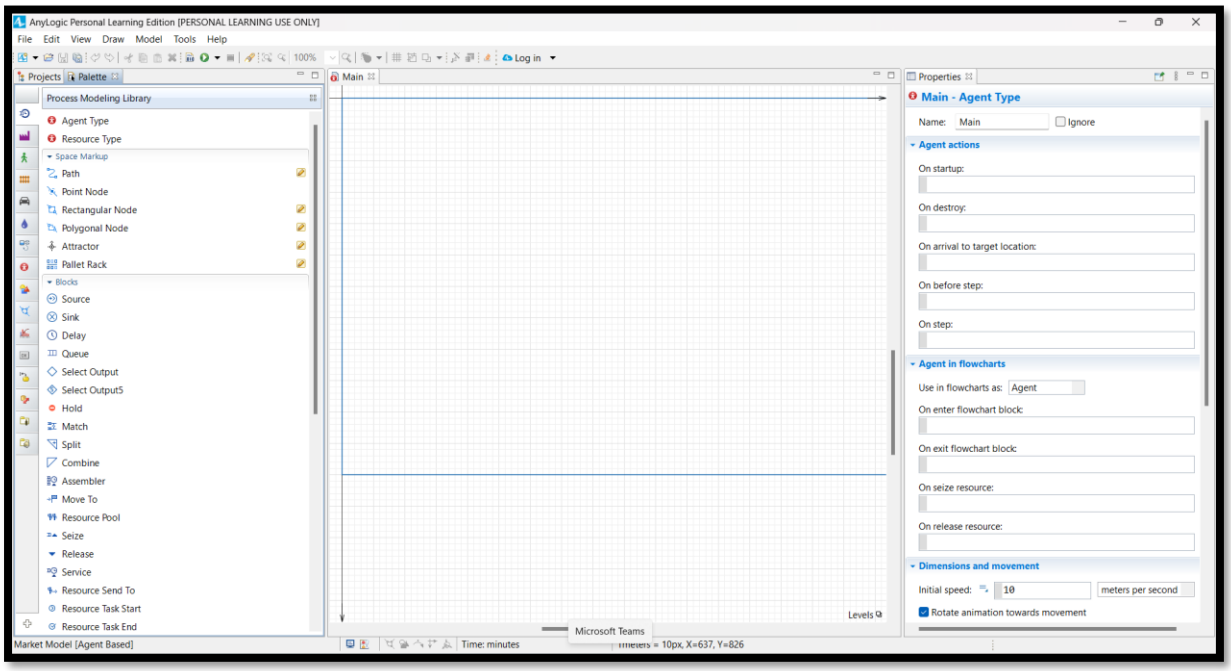
6. Model Validation and Sensitivity Analysis:

- Validate the model by comparing simulation results with real-world market data.

- Perform sensitivity analysis to evaluate the robustness of the model to changes in parameter values and assumptions.

By following these steps, you can develop an agent-based model of financial markets using AnyLogic software and gain valuable insights into market dynamics and trader behaviour.


New Model
New Model
Create a new model
Model name:
Market Model [Agent Based]
Location:
C:\Users\talib\Models
Browse...
Java package:
market_model__agent_based_
Model time units:
minutes
The following model will be created:
C:\Users\talib\Models\Market Model [Agent Based]\Market Model [Agent Based].alp
Finish
Cancel


AnyLogic Personal Learning Edition [PERSONAL LEARNING USE ONLY]
File Edit View Draw Model Tools Help
Projects
Palette
Process Modeling Library
Agent Type
Resource Type
Space Markup
Path
Point Node
Rectangular Node
Polygonal Node
Attractor
Pallet Rack
Blocks
Source
Sink
Delay
Queue
Select Output
Select Output5
Hold
Match
Split
Combine
Assembler
Move To
Resource Pool
Seize
Release
Service
Resource Send To
Resource Task Start
Resource Task End
Main
Properties
Main - Agent Type
Name: Main
Ignore
Agent actions
On startup:
On destroy:
On arrival to target location:
On before step:
On step:
Agent in flowcharts
Use in flowcharts as: Agent
On enter flowchart block:
On exit flowchart block:
On seize resource:
On release resource:
Dimensions and movement
Initial speed: 10
meters per second
Rotate animation towards movement
Levels
Microsoft Teams
Market Model [Agent Based]
Time: minutes

New agent

Step 1. Choose what you want to create

Population of agents

Create a number of agents of the same type living in the same environment in the current agent.

Typical cases:

- People
- Consumers
- Patients
- Trucks
- Projects or products

A single agent

Create a single agent that will always exist within the current agent.

Typical cases:

- Supplier, distributor, producer
- Building
- Factory
- Store

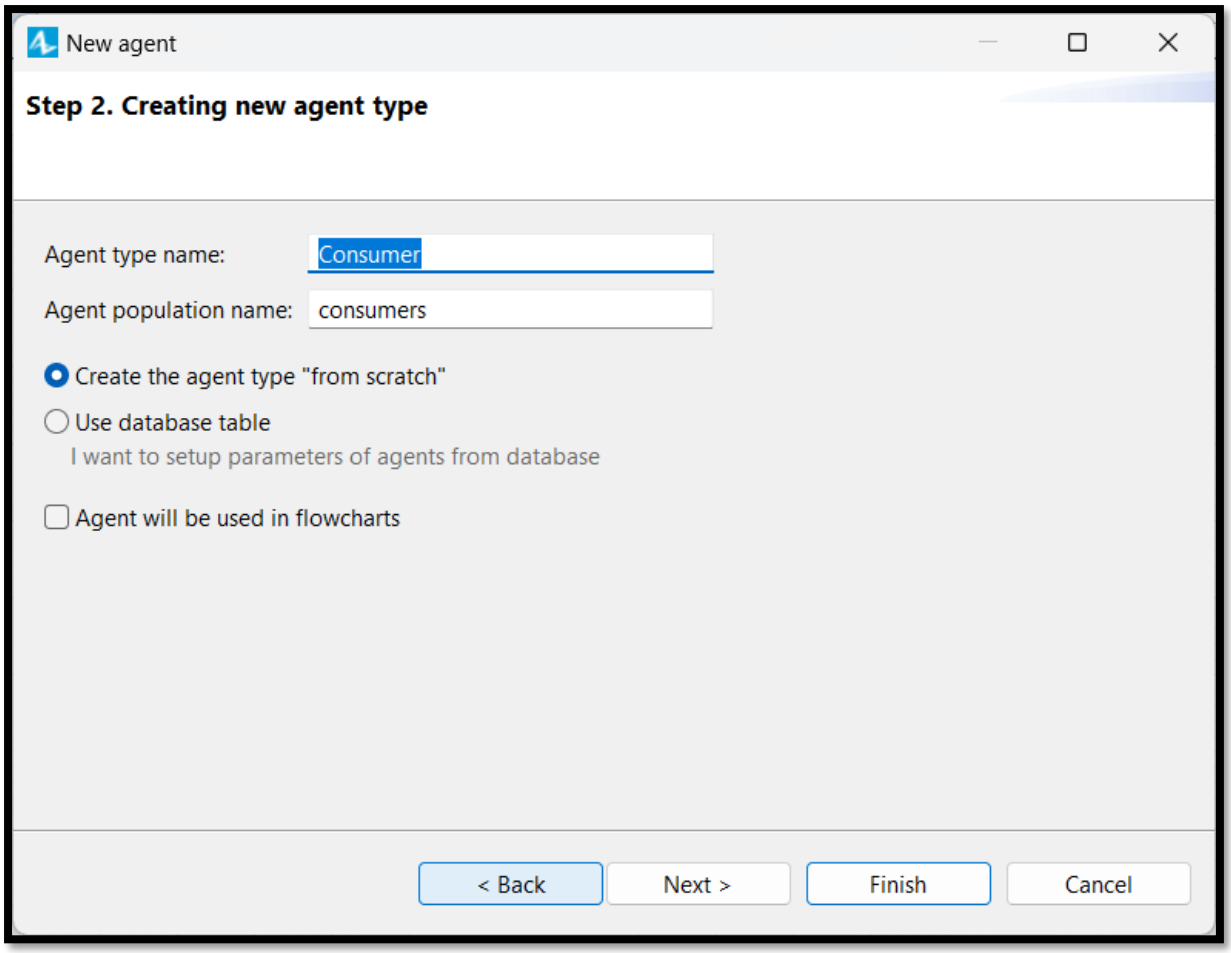
Agent type only

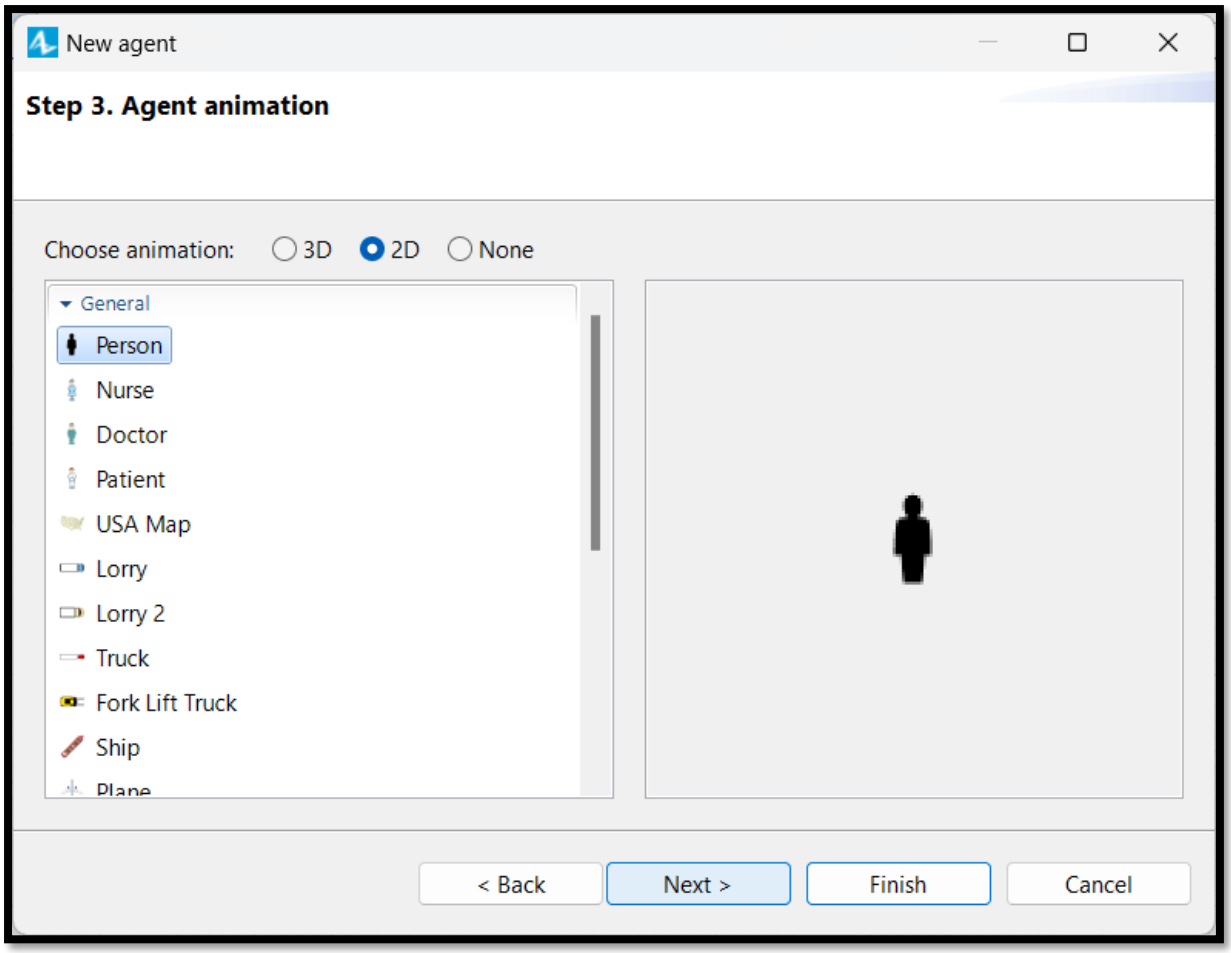
Create an agent type, do not create any agents at this point.

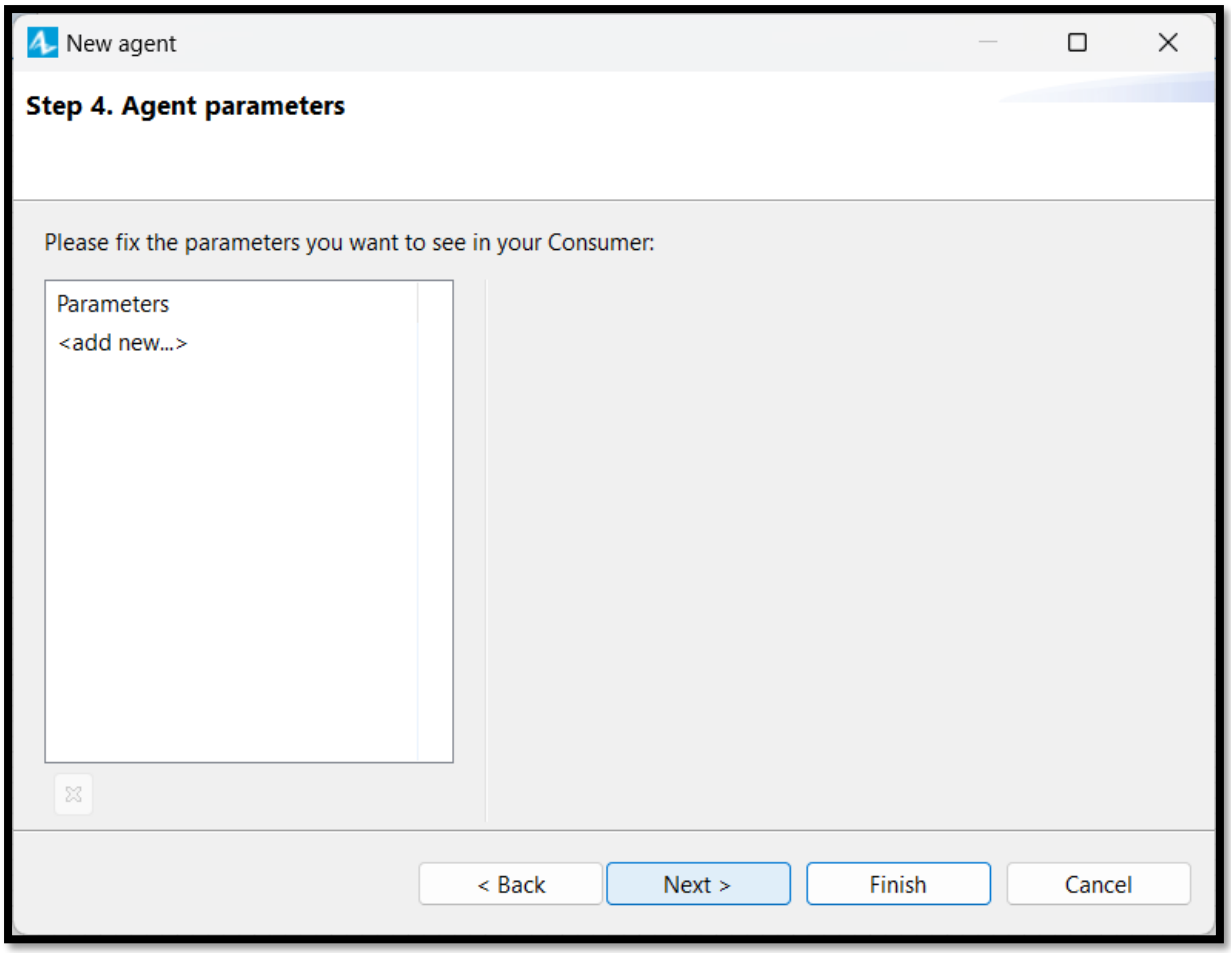
Typical cases:

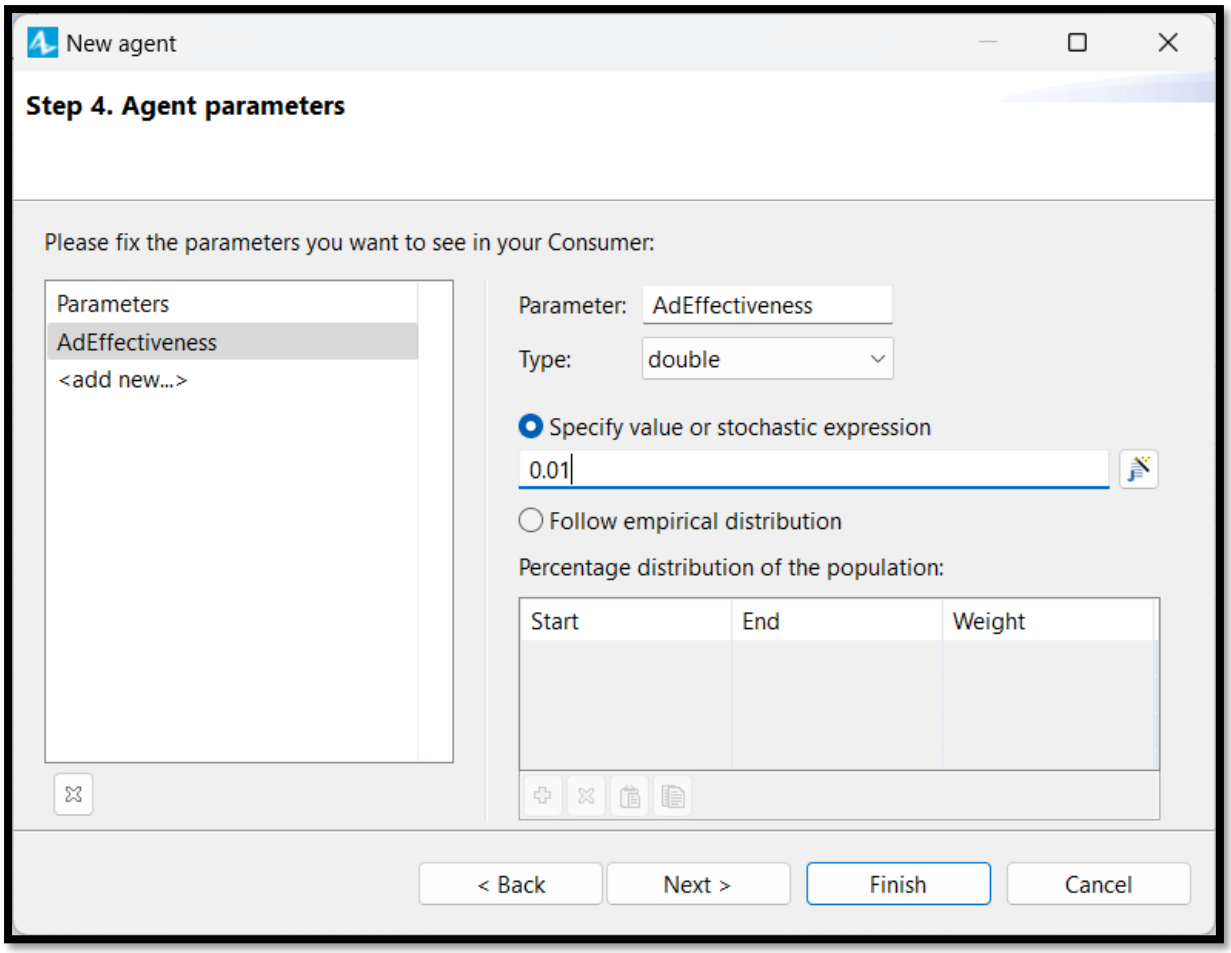
- Agent type: Patient, Customer, Document, Part, Transaction
- Resource type: Doctor, Worker, ForkliftTruck
- Train or rail car type

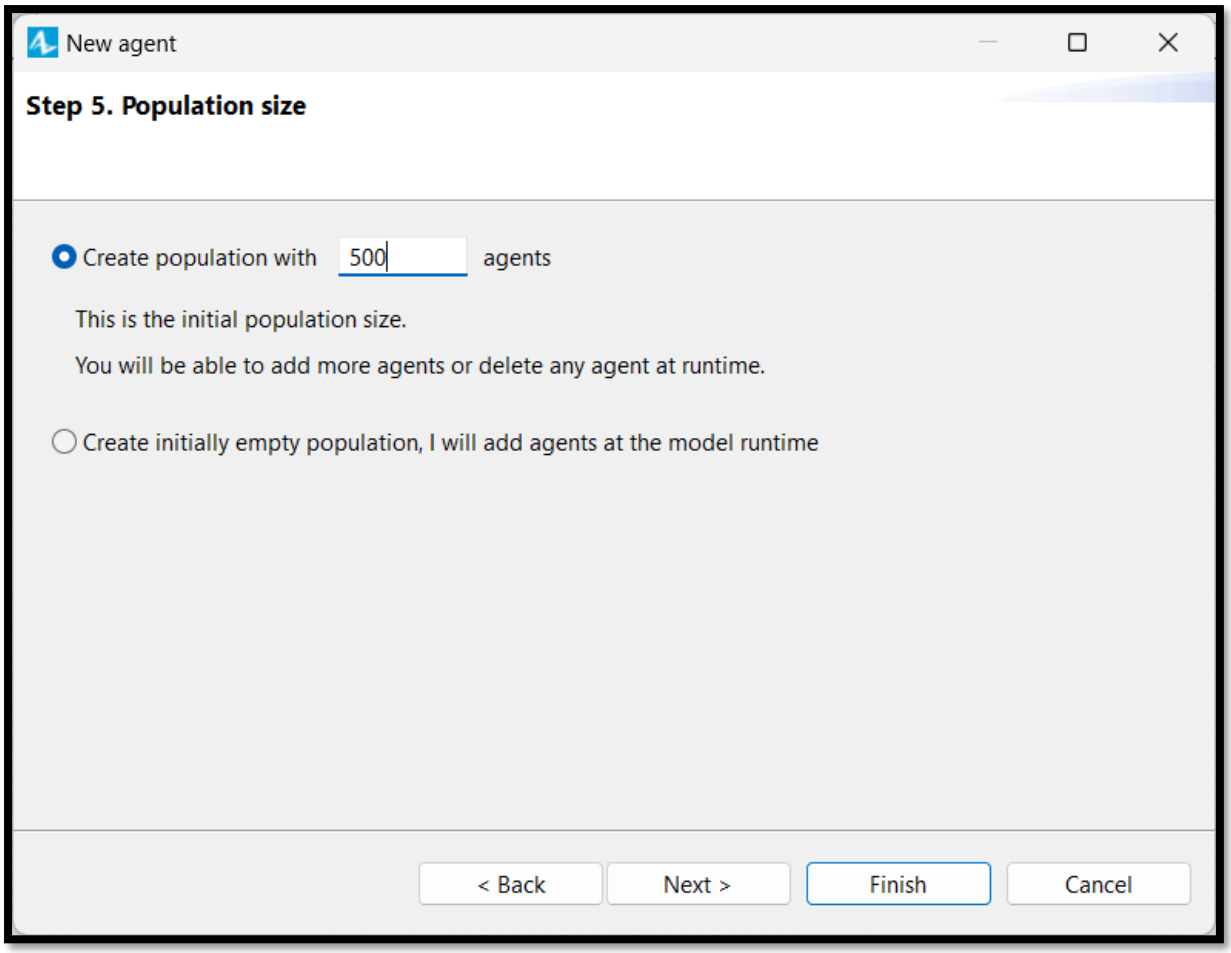
< Back | Next > | Finish | Cancel

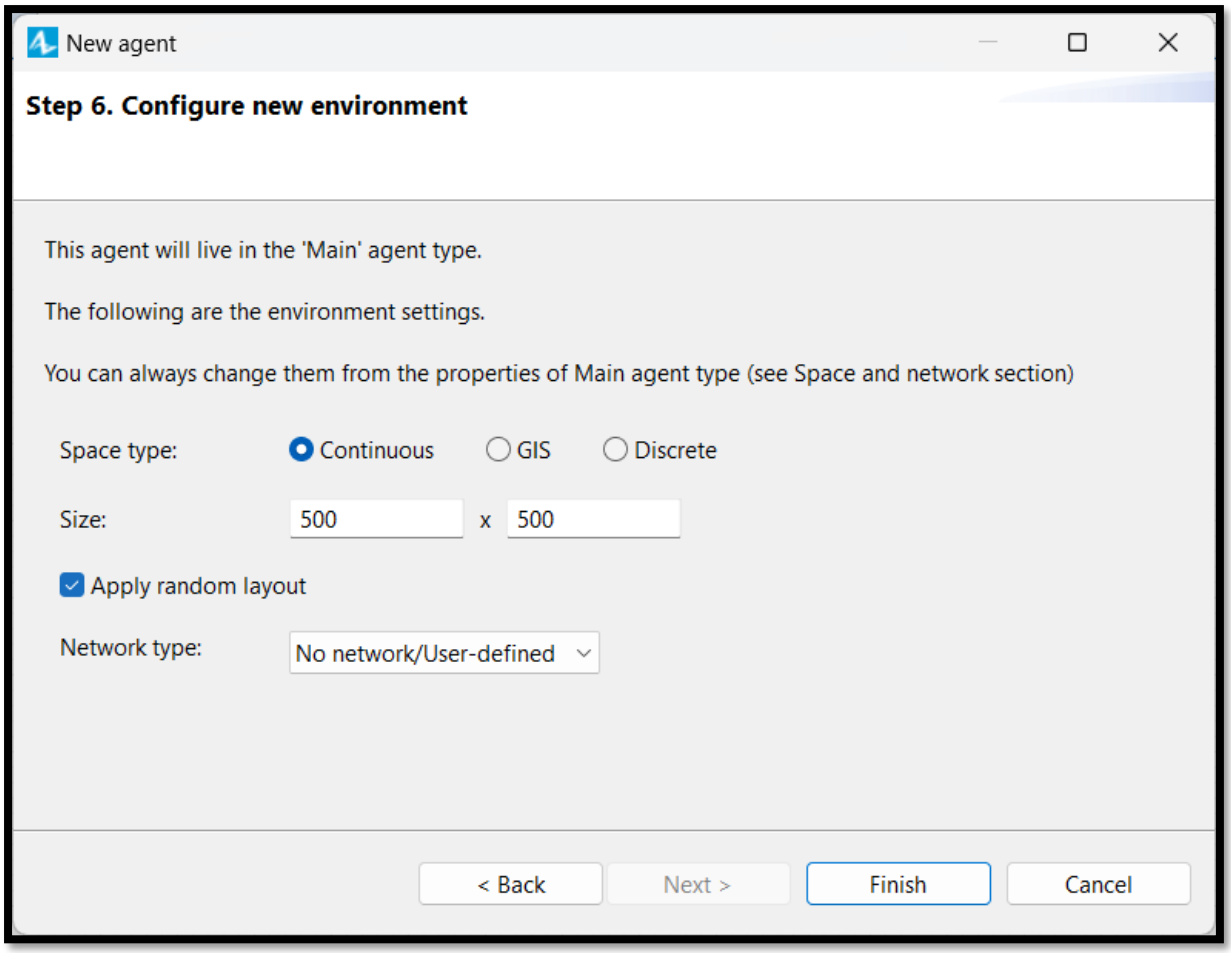

New agent
Step 2. Creating new agent type
Agent type name:
Consumer
Agent population name:
consumers
Create the agent type "from scratch"
Use database table
I want to setup parameters of agents from database
Agent will be used in flowcharts
< Back
Next >
Finish
Cancel

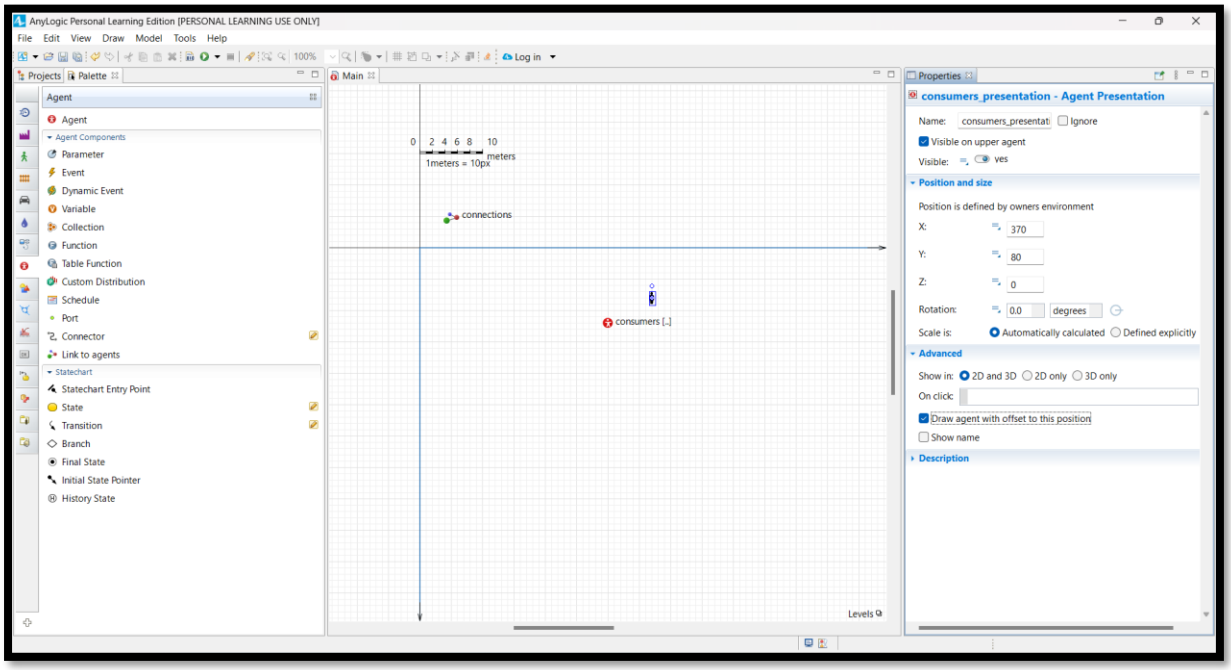

New agent
Step 3. Agent animation
Choose animation:
3D
2D
None
General
Person
Nurse
Doctor
Patient
USA Map
Lorry
Lorry 2
Truck
Fork Lift Truck
Ship
< Back
Next >
Finish
Cancel

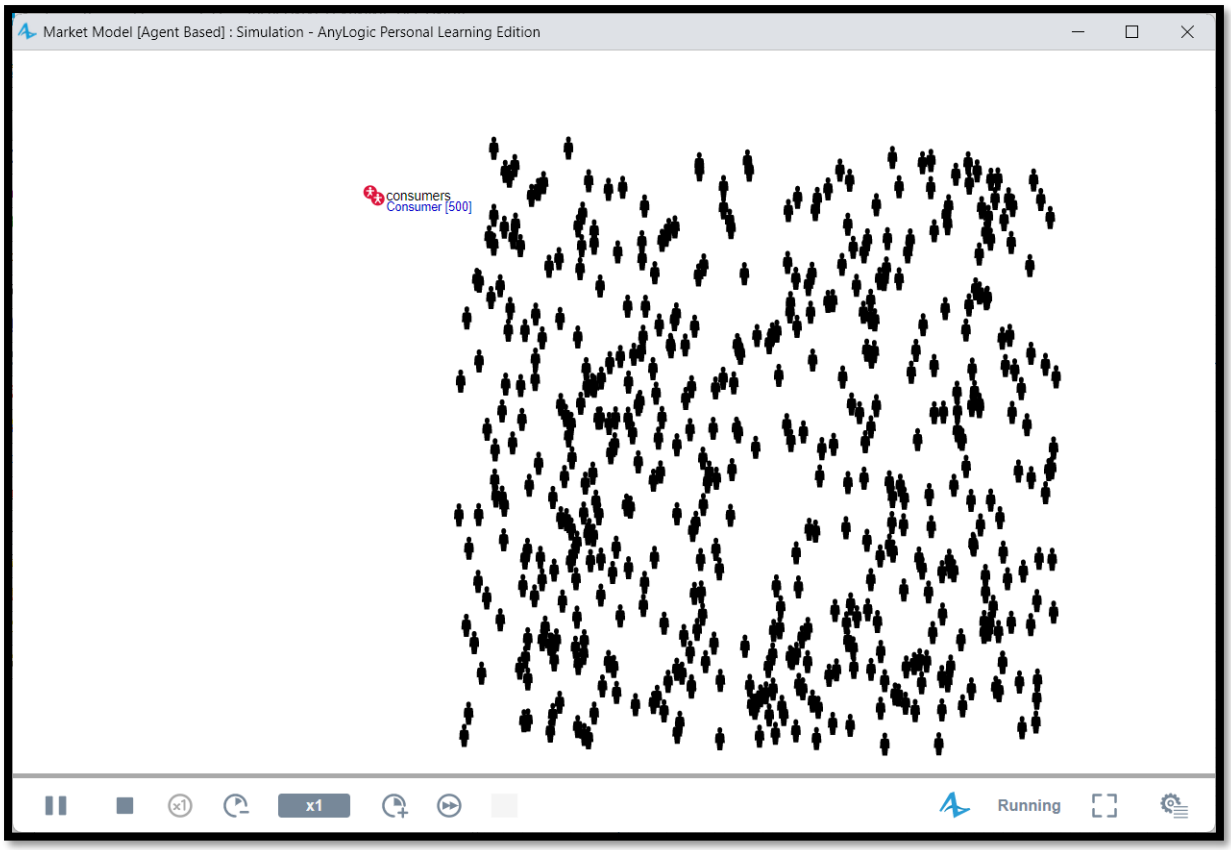

New agent
Step 4. Agent parameters
Please fix the parameters you want to see in your Consumer:
Parameters
<add new...>
< Back
Next >
Finish
Cancel

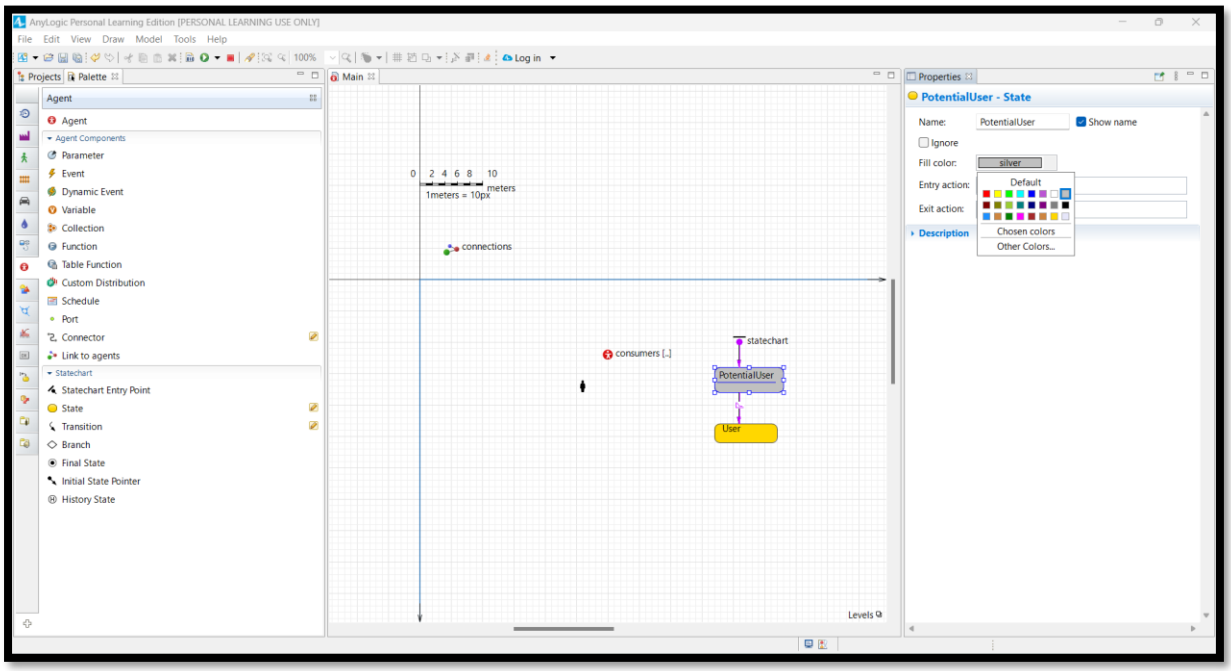

New agent
Step 4. Agent parameters
Please fix the parameters you want to see in your Consumer:
Parameters
AdEffectiveness
<add new...>
Parameter: AdEffectiveness
Type: double
Specify value or stochastic expression
0.01
Follow empirical distribution
Percentage distribution of the population:
Start
End
Weight
< Back
Next >
Finish
Cancel

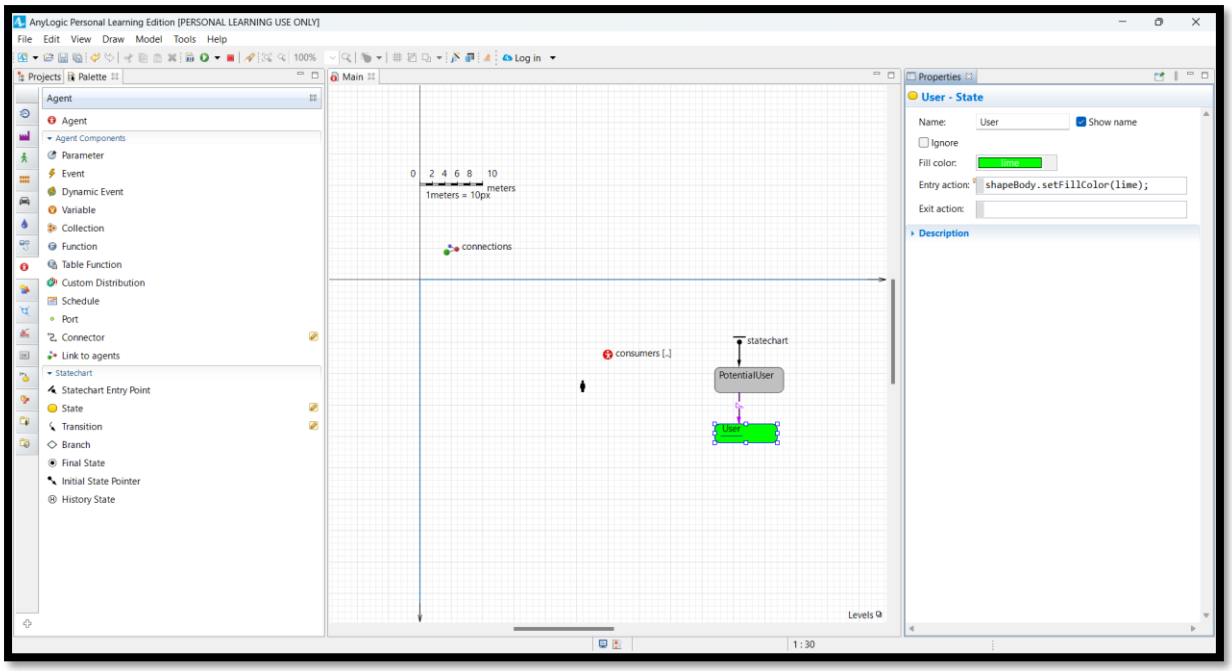

New agent
Step 5. Population size
Create population with 500 agents
This is the initial population size.
You will be able to add more agents or delete any agent at runtime.
Create initially empty population, I will add agents at the model runtime
< Back
Next >
Finish
Cancel

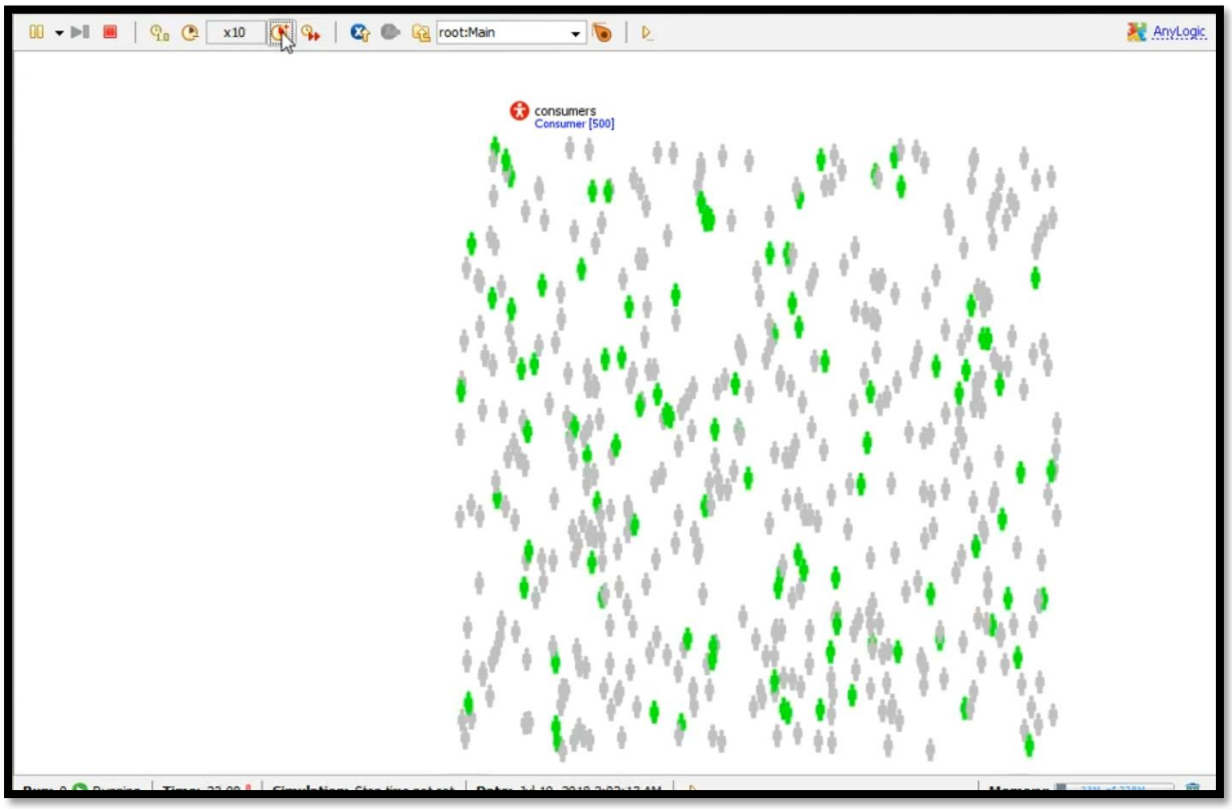

New agent
Step 6. Configure new environment
This agent will live in the 'Main' agent type.
The following are the environment settings.
You can always change them from the properties of Main agent type (see Space and network section)
Space type:
Continuous
GIS
Discrete
Size:
500
x
500
Apply random layout
Network type:
No network/User-defined
< Back
Next >
Finish
Cancel

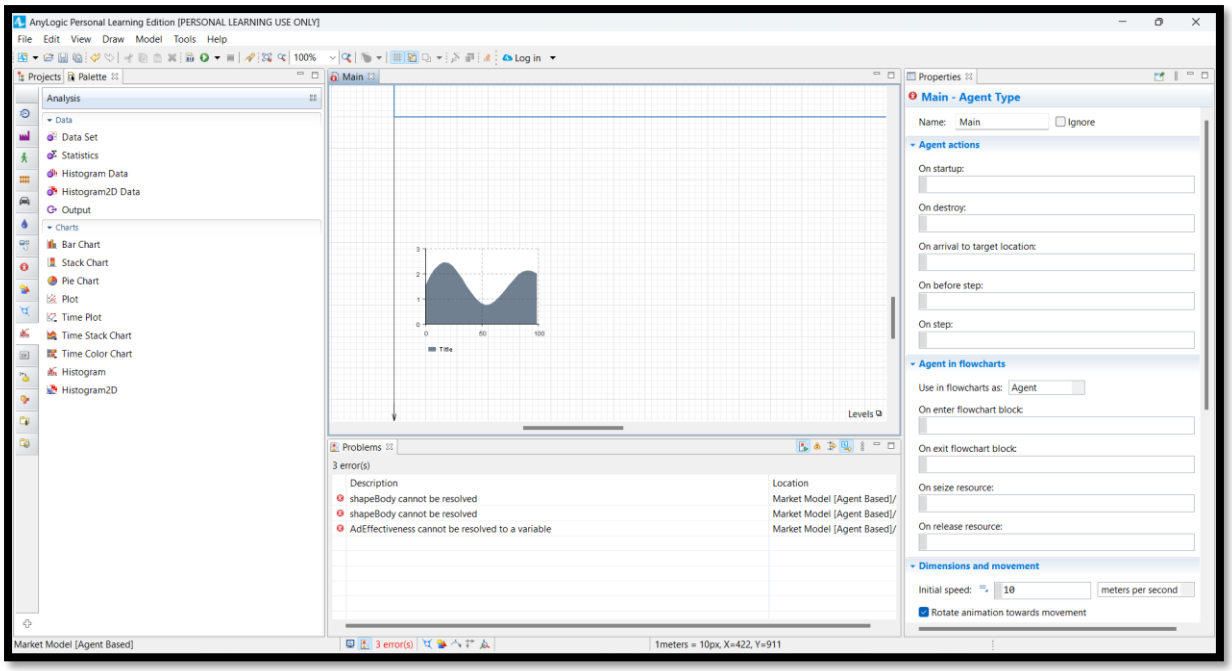

AnyLogic Personal Learning Edition [PERSONAL LEARNING USE ONLY]
File Edit View Draw Model Tools Help
Projects
Palette
Agent
Agent Components
Parameter
Event
Dynamic Event
Variable
Collection
Function
Table Function
Custom Distribution
Schedule
Port
Connector
Link to agents
Statechart
Statechart Entry Point
State
Transition
Branch
Final State
Initial State Pointer
History State
Main
1meters = 10px
meters
connections
consumers [...]
Levels
Properties
consumers_presentation - Agent Presentation
Name:
Ignore
Visible on upper agent
Visible:
Position and size
Position is defined by owners environment
X:
370
Y:
80
Z:
0
Rotation:
degrees
Scale is:
Automatically calculated
Defined explicitly
Advanced
Show in:
2D and 3D
2D only
3D only
On click:
Draw agent with offset to this position
Show name
Description

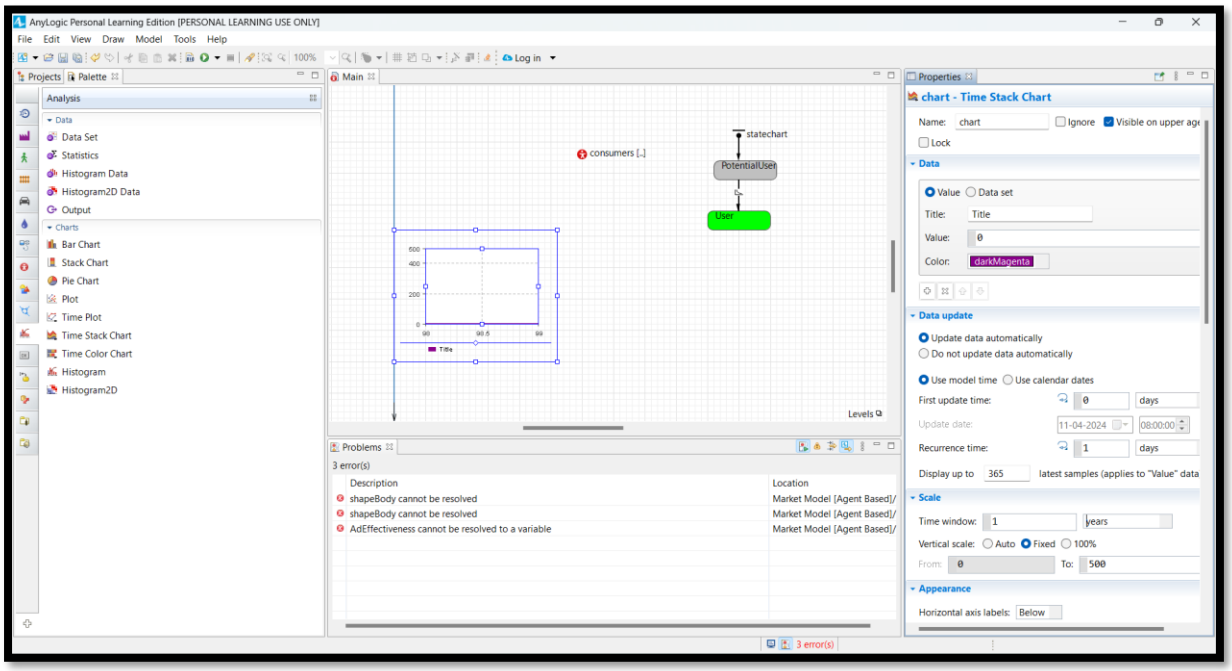

Market Model [Agent Based] : Simulation - AnyLogic Personal Learning Edition
consumers
Consumer [500]
x1
Running

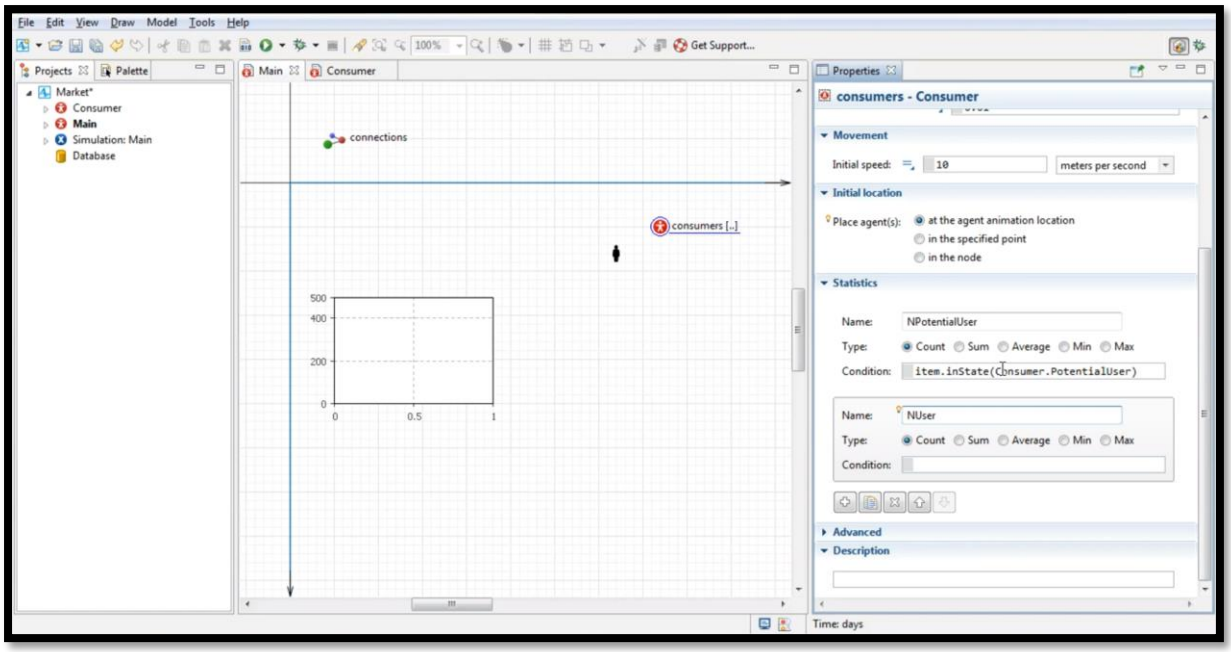

AnyLogic Personal Learning Edition [PERSONAL LEARNING USE ONLY]
File Edit View Draw Model Tools Help
Projects
Palette
Agent
Agent Components
Parameter
Event
Dynamic Event
Variable
Collection
Function
Table Function
Custom Distribution
Schedule
Port
Connector
Link to agents
Statechart
Statechart Entry Point
State
Transition
Branch
Final State
Initial State Pointer
History State
Main
1meters = 10px
connections
consumers [..]
statechart
PotentialUser
User
Properties
PotentialUser - State
Name:
PotentialUser
Show name
Ignore
Fill color:
silver
Entry action:
Exit action:
Default
Chosen colors
Other Colors...
Description
Levels

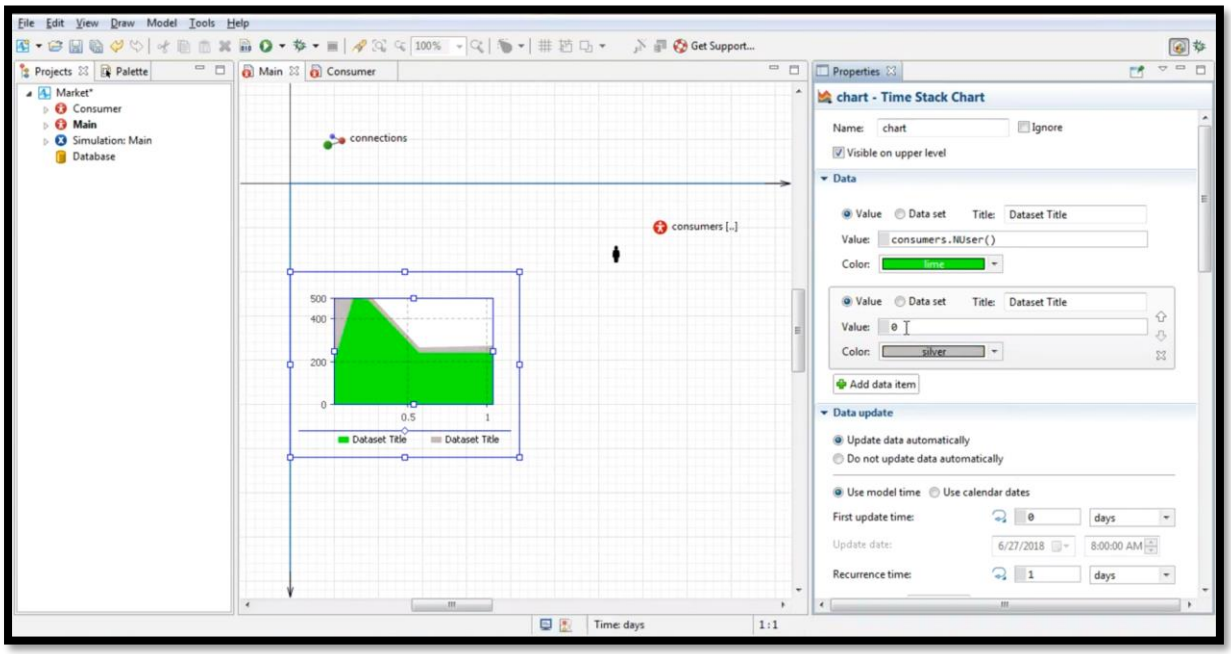

AnyLogic Personal Learning Edition [PERSONAL LEARNING USE ONLY]
File Edit View Draw Model Tools Help
Projects Palette
Agent
Agent
Agent Components
Parameter
Event
Dynamic Event
Variable
Collection
Function
Table Function
Custom Distribution
Schedule
Port
Connector
Link to agents
Statechart
Statechart Entry Point
State
Transition
Branch
Final State
Initial State Pointer
History State
Main
0 2 4 6 8 10
meters
1meters = 10px
connections
consumers [..]
statechart
PotentialUser
User
Levels
1 : 30
Properties
User - State
Name: User
Show name
Ignore
Fill color: lime
Entry action: shapeBody.setFillColor(lime);
Exit action:
Description

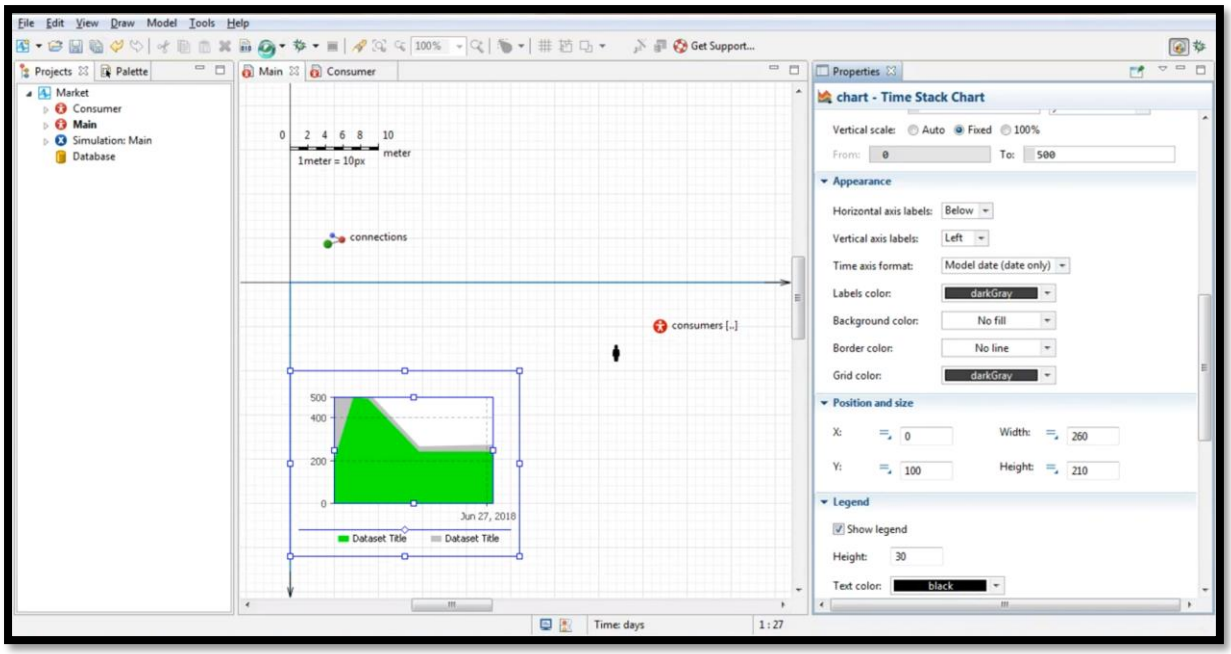

x10
root:Main
AnyLogic
consumers
Consumer [500]

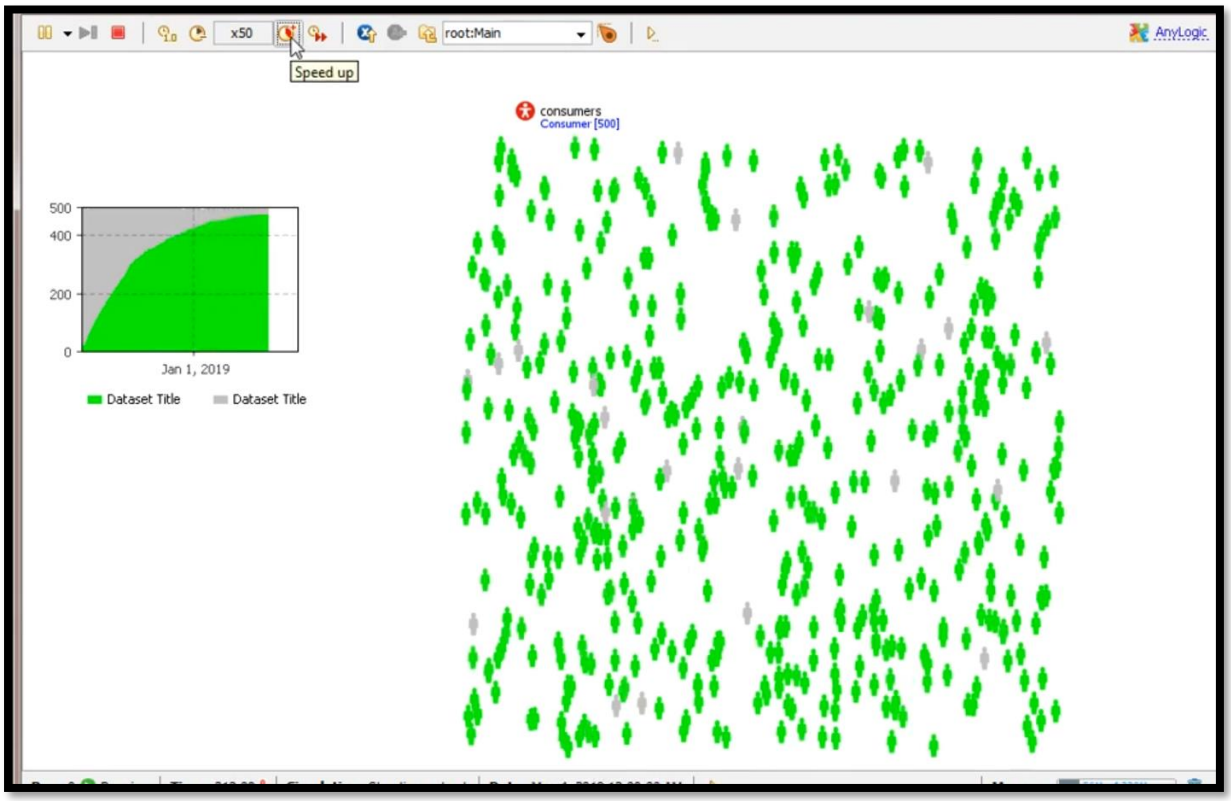

AnyLogic Personal Learning Edition [PERSONAL LEARNING USE ONLY]
File Edit View Draw Model Tools Help
Projects Palette
Analysis
Data
Data Set
Statistics
Histogram Data
Histogram2D Data
Output
Charts
Bar Chart
Stack Chart
Pie Chart
Plot
Time Plot
Time Stack Chart
Time Color Chart
Histogram
Histogram2D
Main
Levels
Problems
3 error(s)
Description
Location
shapeBody cannot be resolved
Market Model [Agent Based]/
shapeBody cannot be resolved
Market Model [Agent Based]/
AdEffectiveness cannot be resolved to a variable
Market Model [Agent Based]/
Properties
Main - Agent Type
Name: Main
Ignore
Agent actions
On startup:
On destroy:
On arrival to target location:
On before step:
On step:
Agent in flowcharts
Use in flowcharts as: Agent
On enter flowchart block:
On exit flowchart block:
On seize resource:
On release resource:
Dimensions and movement
Initial speed: 10
meters per second
Rotate animation towards movement
Market Model [Agent Based]
3 error(s)
1meters = 10px, X=422, Y=911

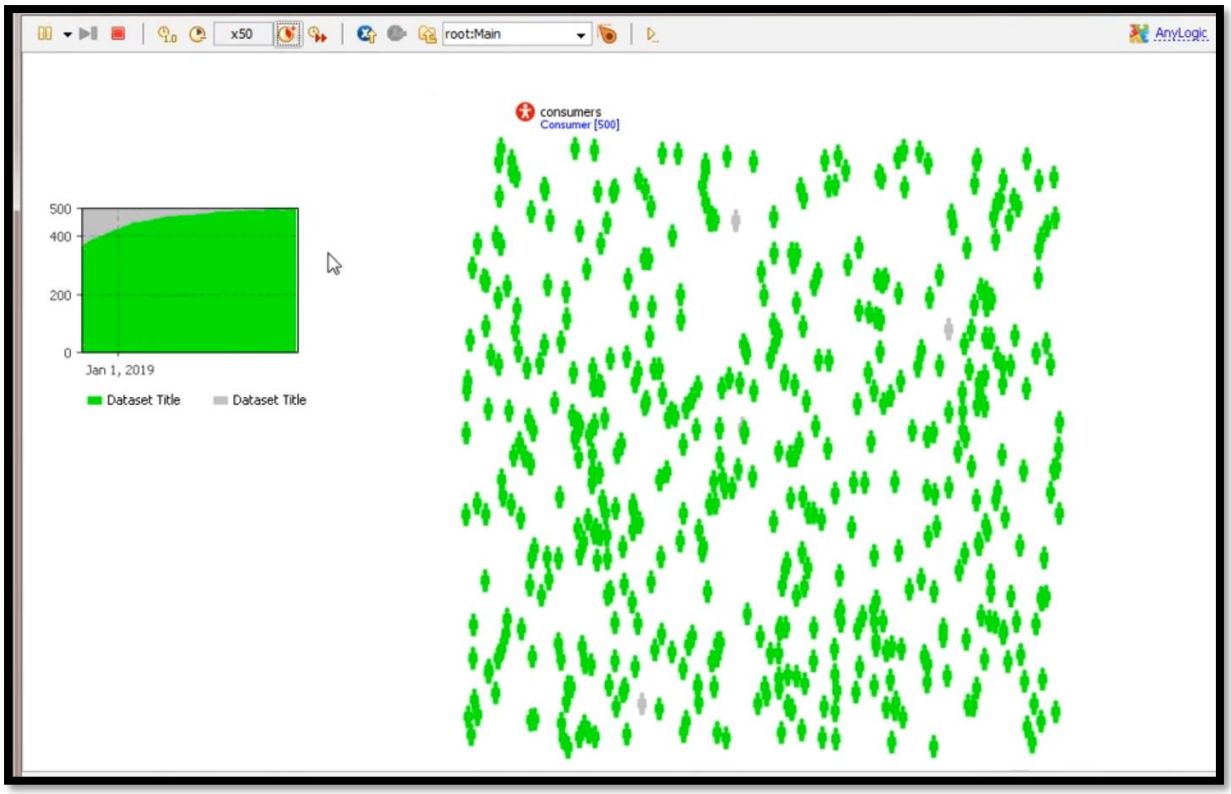

AnyLogic Personal Learning Edition [PERSONAL LEARNING USE ONLY]
File Edit View Draw Model Tools Help
Projects Palette
Analysis
Data
Data Set
Statistics
Histogram Data
Histogram2D Data
Output
Charts
Bar Chart
Stack Chart
Pie Chart
Plot
Time Plot
Time Stack Chart
Time Color Chart
Histogram
Histogram2D
Main
consumers [..]
statechart
PotentialUser
User
Levels
Problems
3 error(s)
Description
Location
shapeBody cannot be resolved
Market Model [Agent Based]/
shapeBody cannot be resolved
Market Model [Agent Based]/
AdEffectiveness cannot be resolved to a variable
Market Model [Agent Based]/
Properties
chart - Time Stack Chart
Name: chart
Ignore
Visible on upper age
Lock
Data
Value
Data set
Title: Title
Value: 0
Color: darkMagenta
Data update
Update data automatically
Do not update data automatically
Use model time
Use calendar dates
First update time: 0 days
Update date: 11-04-2024 08:00:00
Recurrence time: 1 days
Display up to 365 latest samples (applies to "Value" data
Scale
Time window: 1 years
Vertical scale: Auto Fixed 100%
From: 0 To: 500
Appearance
Horizontal axis labels: Below
3 error(s)

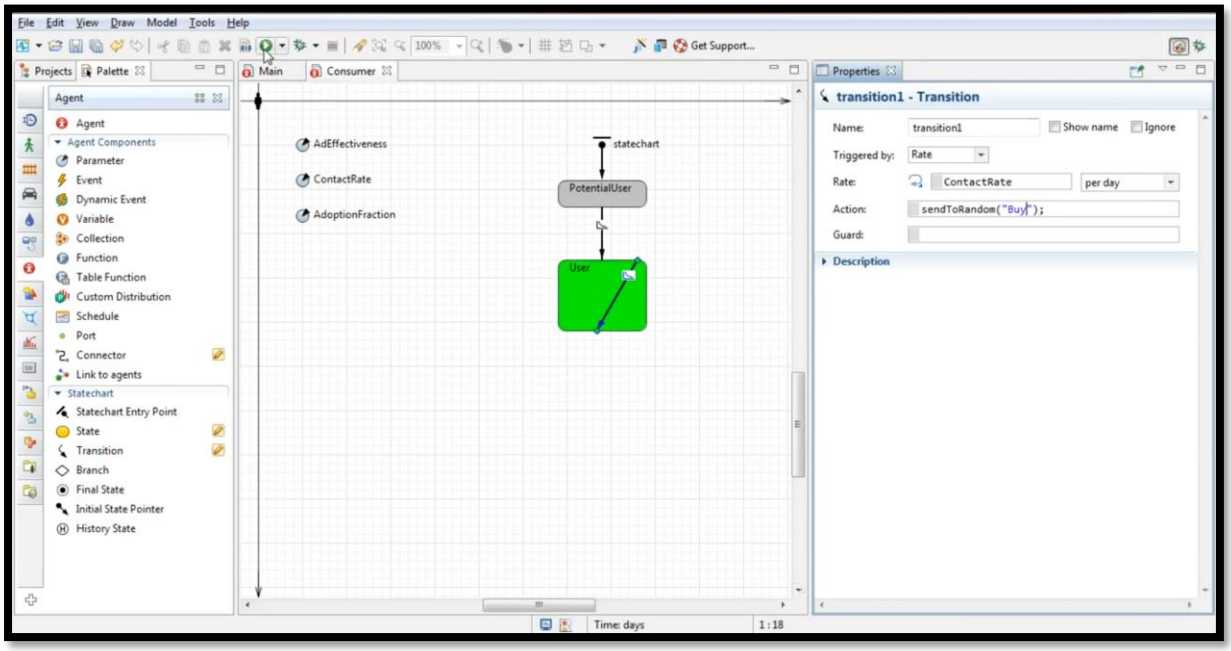

File Edit View Draw Model Tools Help
Get Support...
Projects
Palette
Market*
Consumer
Main
Simulation: Main
Database
Main
Consumer
connections
consumers [..]
500
400
200
0
0
0.5
1
Properties
consumers - Consumer
Movement
Initial speed:
10
meters per second
Initial location
Place agent(s):
at the agent animation location
in the specified point
in the node
Statistics
Name:
NPotentialUser
Type:
Count
Sum
Average
Min
Max
Condition:
item.inState(Consumer.PotentialUser)
Name:
NUser
Type:
Count
Sum
Average
Min
Max
Condition:
Advanced
Description
Time: days

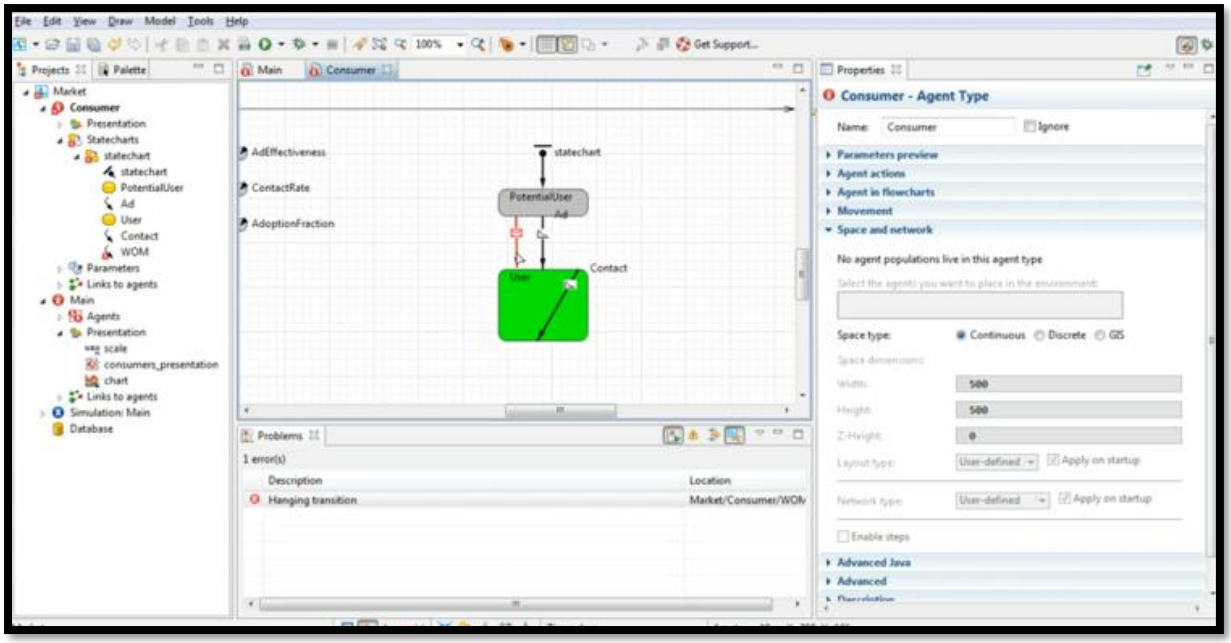

File Edit View Draw Model Tools Help
Get Support...
Projects
Palette
Market*
Consumer
Main
Simulation: Main
Database
Main
Consumer
connections
consumers [..]
500
400
200
0
0.5
1
Dataset Title
Dataset Title
Properties
chart - Time Stack Chart
Name:
chart
Ignore
Visible on upper level
Data
Value
Data set
Title:
Dataset Title
Value:
consumers.NUser()
Color:
lime
Value
Data set
Title:
Dataset Title
Value:
0
Color:
silver
Add data item
Data update
Update data automatically
Do not update data automatically
Use model time
Use calendar dates
First update time:
0
days
Update date:
6/27/2018
8:00:00 AM
Recurrence time:
1
days
Time: days
1 : 1

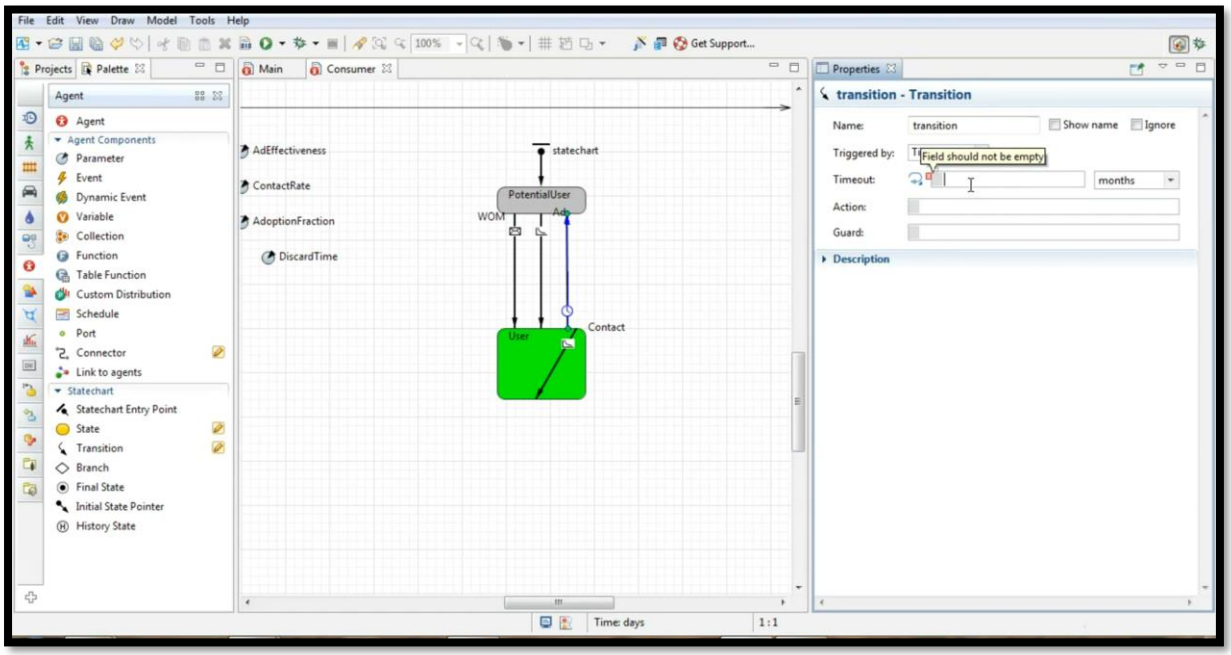

File Edit View Draw Model Tools Help
Get Support...
Projects
Palette
Market
Consumer
Main
Simulation: Main
Database
Main
Consumer
0 2 4 6 8 10
meter
1meter = 10px
connections
consumers [..]
500
400
200
0
Jun 27, 2018
Dataset Title
Dataset Title
Properties
chart - Time Stack Chart
Vertical scale: Auto Fixed 100%
From: 0
To: 500
Appearance
Horizontal axis labels: Below
Vertical axis labels: Left
Time axis format: Model date (date only)
Labels color: darkGray
Background color: No fill
Border color: No line
Grid color: darkGray
Position and size
X: 0
Width: 260
Y: 100
Height: 210
Legend
Show legend
Height: 30
Text color: black
Time: days
1 : 27

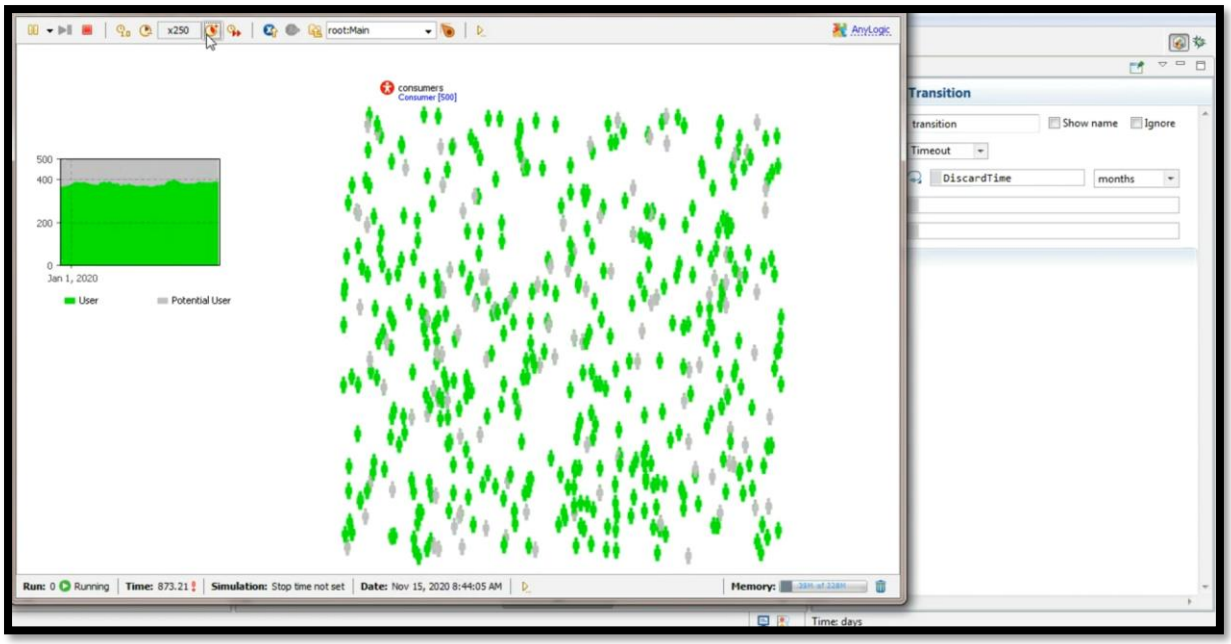

x50
Speed up
root:Main
AnyLogic
consumers
Consumer [500]
500
400
200
0
Jan 1, 2019
Dataset Title
Dataset Title

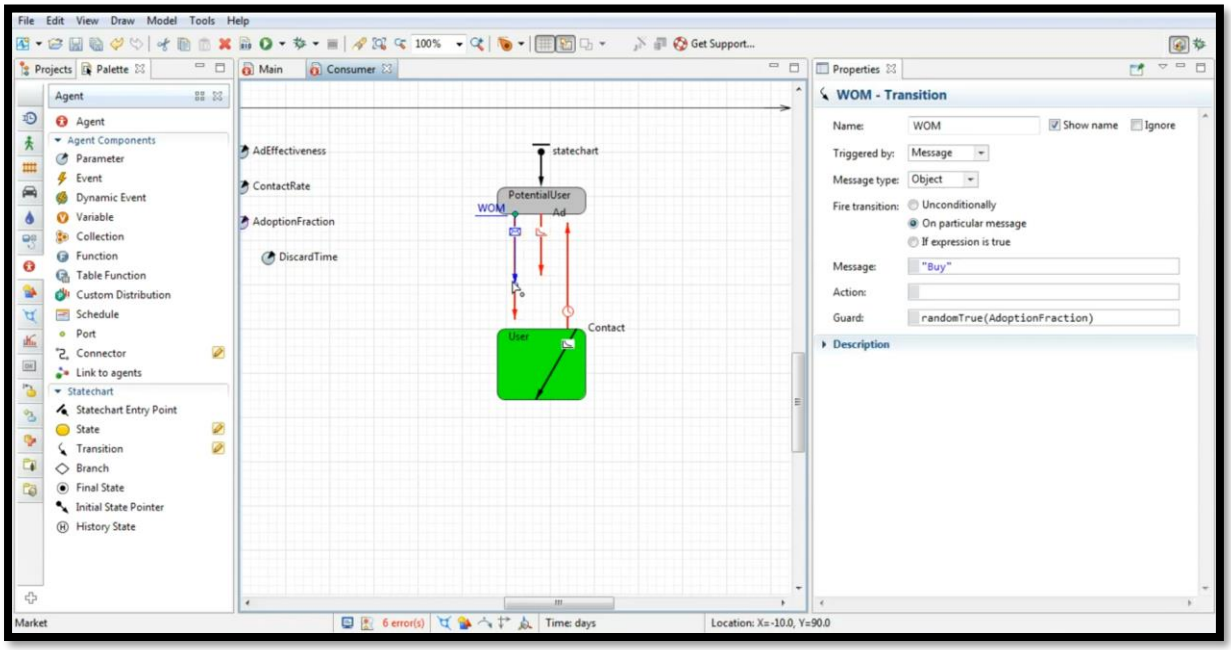

x50
root:Main
AnyLogic
consumers
Consumer [500]
500
400
200
0
Jan 1, 2019
Dataset Title
Dataset Title

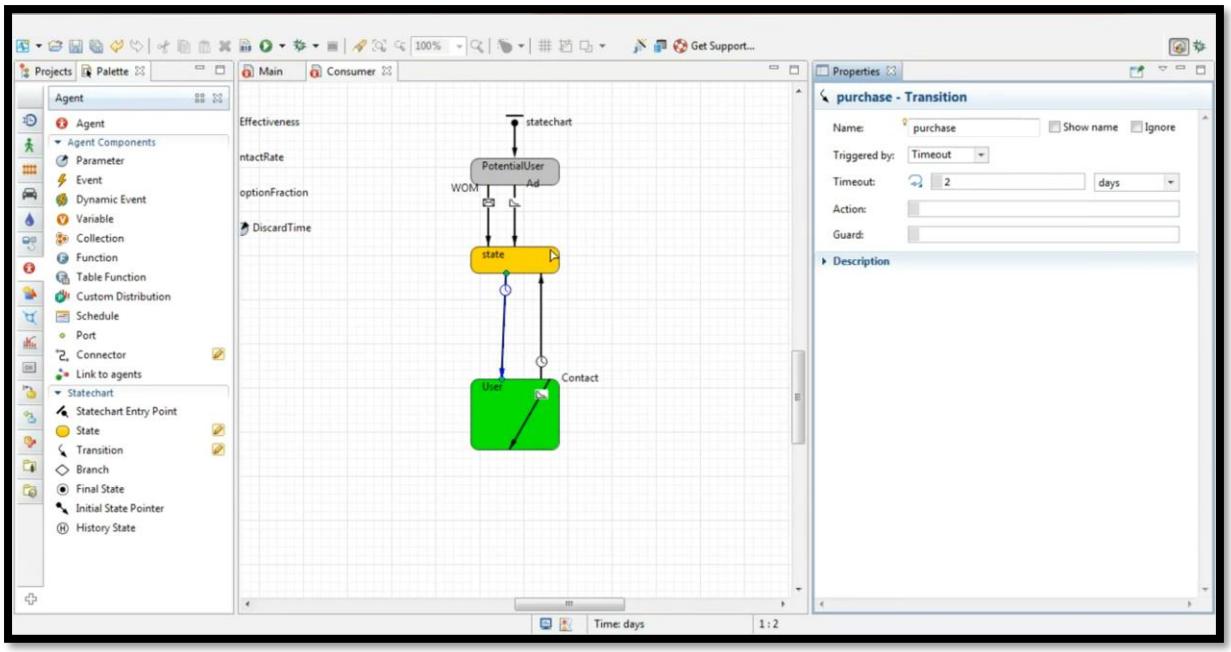

File Edit View Draw Model Tools Help
Get Support...
Projects
Palette
Agent
Agent Components
Parameter
Event
Dynamic Event
Variable
Collection
Function
Table Function
Custom Distribution
Schedule
Port
Connector
Link to agents
Statechart
Statechart Entry Point
State
Transition
Branch
Final State
Initial State Pointer
History State
Main
Consumer
AdEffectiveness
ContactRate
AdoptionFraction
statechart
PotentialUser
User
Properties
transition1 - Transition
Name: transition1
Show name
Ignore
Triggered by: Rate
Rate: ContactRate per day
Action: sendToRandom("Buy");
Guard:
Description
Time: days
1:18

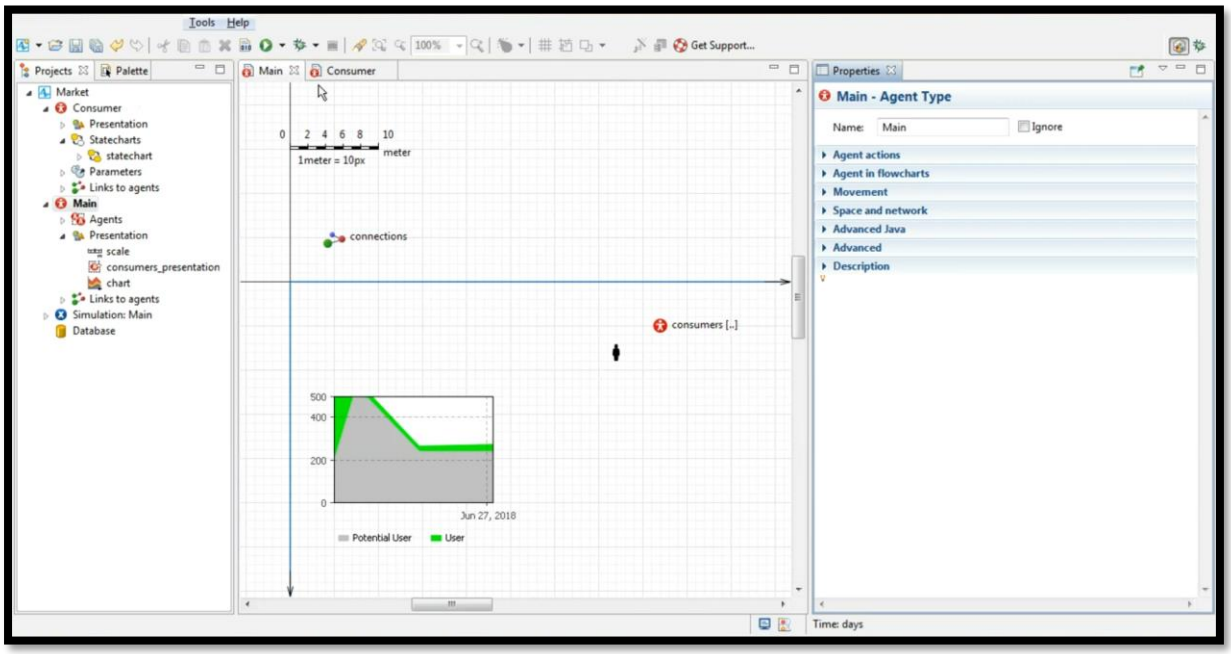

File Edit View Draw Model Tools Help
Projects Palette
Main Consumer
Properties
Consumer - Agent Type
Name: Consumer
Ignore
Parameters preview
Agent actions
Agent in flowcharts
Movement
Space and network
No agent populations live in this agent type
Space type: Continuous Discrete GIS
Market
Consumer
Presentation
Statecharts
statechart
statechart
PotentialUser
Ad
User
Contact
WOM
Parameters
Links to agents
Main
Agents
Presentation
scale
consumers_presentation
chart
Links to agents
Simulation: Main
Database
AdEffectiveness
ContactRate
AdoptionFraction
statechart
PotentialUser
Ad
Contact
User
Problems
1 error(s)
Description
Location
Hanging transition
Market/Consumer/WOM
Advanced Java
Advanced

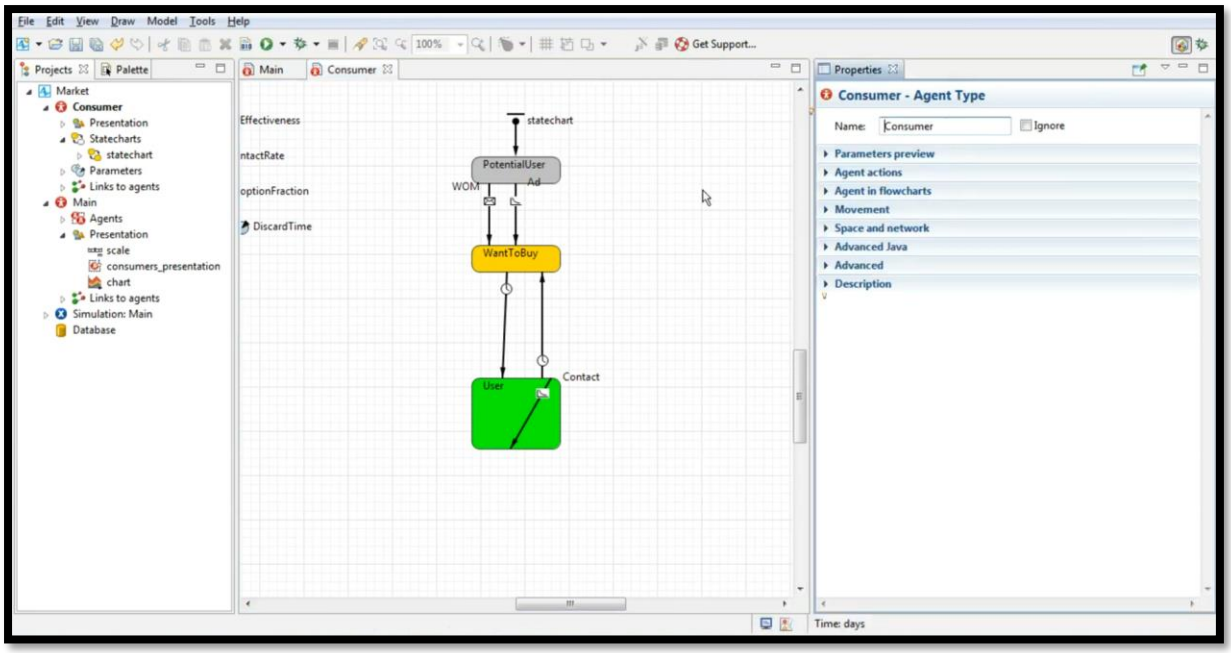

File Edit View Draw Model Tools Help
Projects Palette
Main Consumer
Properties
transition - Transition
Name: transition
Show name
Ignore
Triggered by:
Field should not be empty
Timeout:
months
Action:
Guard:
Description
Agent
Agent Components
Parameter
Event
Dynamic Event
Variable
Collection
Function
Table Function
Custom Distribution
Schedule
Port
Connector
Link to agents
Statechart
Statechart Entry Point
State
Transition
Branch
Final State
Initial State Pointer
History State
AdEffectiveness
ContactRate
AdoptionFraction
DiscardTime
statechart
PotentialUser
WOM
Ad
Contact
User
Time: days
1:1

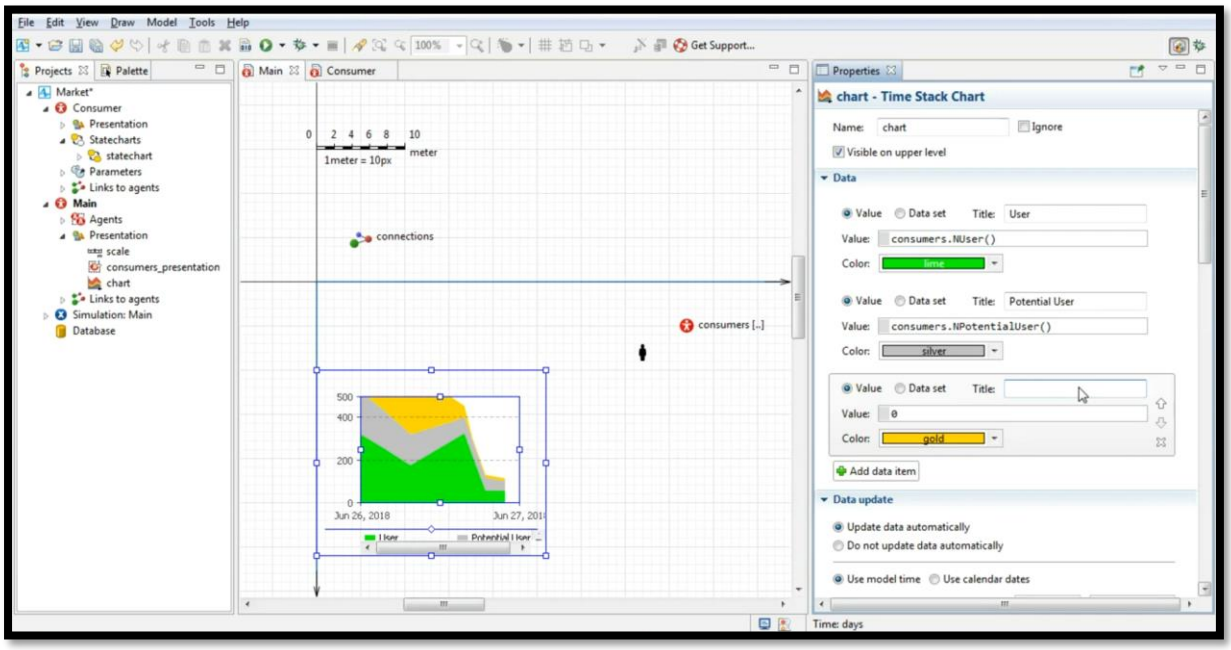

x250
root:Main
AnyLogic
consumers
Consumer [500]
500
400
200
0
Jan 1, 2020
User
Potential User
Transition
transition
Show name
Ignore
Timeout
DiscardTime
months
Run: 0 Running
Time: 873.21
Simulation: Stop time not set
Date: Nov 15, 2020 8:44:05 AM
Memory:
Time: days

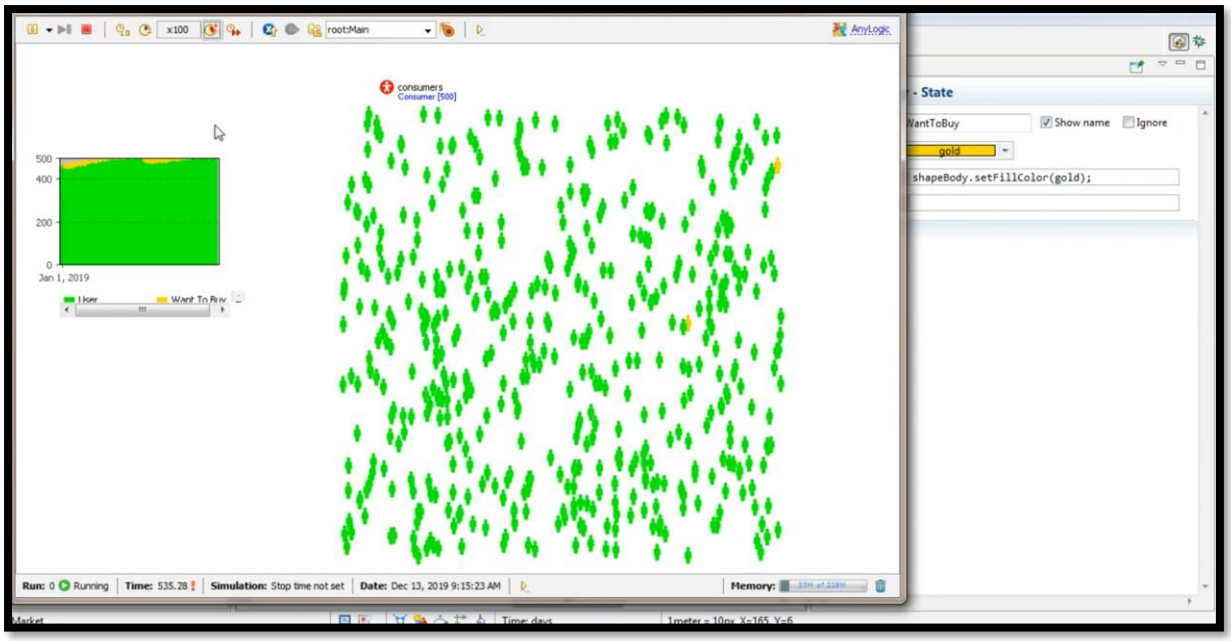

File Edit View Draw Model Tools Help
100%
Get Support...
Projects
Palette
Agent
Agent Components
Parameter
Event
Dynamic Event
Variable
Collection
Function
Table Function
Custom Distribution
Schedule
Port
Connector
Link to agents
Statechart
Statechart Entry Point
State
Transition
Branch
Final State
Initial State Pointer
History State
Main
Consumer
AdEffectiveness
ContactRate
AdoptionFraction
DiscardTime
statechart
PotentialUser
WOM
Ad
Contact
User
Properties
WOM - Transition
Name: WOM
Show name
Ignore
Triggered by: Message
Message type: Object
Fire transition:
Unconditionally
On particular message
If expression is true
Message: "Buy"
Action:
Guard: randomTrue(AdoptionFraction)
Description
Market
6 error(s)
Time: days
Location: X=-10.0, Y=90.0


Projects
Palette
Agent
Agent Components
Parameter
Event
Dynamic Event
Variable
Collection
Function
Table Function
Custom Distribution
Schedule
Port
Connector
Link to agents
Statechart
Statechart Entry Point
State
Transition
Branch
Final State
Initial State Pointer
History State
Main
Consumer
Effectiveness
ntactRate
optionFraction
DiscardTime
statechart
PotentialUser
WOM
Ad
state
User
Contact
Properties
purchase - Transition
Name:
purchase
Show name
Ignore
Triggered by:
Timeout
Timeout:
2
days
Action:
Guard:
Description
Time: days
1 : 2


Tools
Help
Projects
Palette
Market
Consumer
Presentation
Statecharts
statechart
Parameters
Links to agents
Main
Agents
Presentation
scale
consumers_presentation
chart
Links to agents
Simulation: Main
Database
Main
Consumer
0
2
4
6
8
10
1meter = 10px
meter
connections
consumers [..]
500
400
200
0
Jun 27, 2018
Potential User
User
Properties
Main - Agent Type
Name:
Main
Ignore
Agent actions
Agent in flowcharts
Movement
Space and network
Advanced Java
Advanced
Description
Time: days


File Edit View Draw Model Tools Help
Projects Palette
Market
Consumer
Presentation
Statecharts
statechart
Parameters
Links to agents
Main
Agents
Presentation
scale
consumers_presentation
chart
Links to agents
Simulation: Main
Database
Main Consumer
Effectiveness
ntactRate
optionFraction
DiscardTime
statechart
PotentialUser
WOM
Ad
WantToBuy
User
Contact
Properties
Consumer - Agent Type
Name: Consumer
Ignore
Parameters preview
Agent actions
Agent in flowcharts
Movement
Space and network
Advanced Java
Advanced
Description
Time: days


File Edit View Draw Model Tools Help
Projects Palette
Market*
Consumer
Presentation
Statecharts
statechart
Parameters
Links to agents
Main
Agents
Presentation
scale
consumers_presentation
chart
Links to agents
Simulation: Main
Database
Main Consumer
0 2 4 6 8 10
meter
1meter = 10px
connections
consumers [..]
500
400
200
0
Jun 26, 2018
Jun 27, 201
Properties
chart - Time Stack Chart
Name: chart
Ignore
Visible on upper level
Data
Value Data set Title: User
Value: consumers.NUser()
Color: lime
Value Data set Title: Potential User
Value: consumers.NPotentialUser()
Color: silver
Value Data set Title:
Value: 0
Color: gold
Add data item
Data update
Update data automatically
Do not update data automatically
Use model time Use calendar dates
Time: days



Conclusion:

The agent-based model developed in this study successfully simulates market dynamics, allowing us to analyse the emergence of price trends, trading volume, and market stability. By capturing the behaviour of individual traders and their interactions, the model provides insights into the complex dynamics of financial markets, which can inform investment strategies, risk management, and regulatory policies.