Compulsory user input is in purple.

Preparation

- Work through each task list in the given order during the entire conversation. Answer with just "yes" if you understand or "no", if you don't understand.

- Provide ALL RESPONSE CONTENT without asking questions during the entire conversation. DO NOT print any keys (Example: Use "Example" AND NOT "{key-example}") during the entire conversation. Use a "scientific tone" during the entire conversation, unless instructed otherwise. Do not confuse roles and stakeholders, they are different. Do you understand? Say "yes" or say "no".

- Memorise "Unified Modelling Language" as {key-uml}. Memorise "Agent-Based Social Simulation (ABSS) Study" as {key-studyType}. Got it? Say "yes" or say "no".

Analysis

Problem Statement

- Take on the "role" of a "Sociologist" with experience in "Agent-Based Social Simulation" as {key-role1}. Memorise this {key-role1}. Confirm you have memorised.

- Define the "topic" of the memorised {key-studyType} as "{INJECT_TOPIC}". Memorise this topic as {key-topic}. Confirm you have memorised.

- Using a "scientific and inspirational tone". Define a novel and creative "context" for the memorised {key-topic} in 200 WORDS (if possible), then memorise this context as {key-context}.

- Memorise "{INJECT_RESEARCHDESIGN}" as {key-researchDesign}. Memorise "{INJECT_DOMAIN}" as {key-domain}. Memorise "{INJECT_SPECIALISATION}" as {key-specialisation}.

- Define 5 "stakeholders" for the memorised {key-topic}, to participate in a co-creation roleplay game. Memorise these stakeholders together with their personas as {key-stakeholders} (you do not need to create names for personas).

- You will write a Markdown document using the memorised keys (separate each section using headers). Only show the final, resulting markdown file code from this prompt. Display “Problem Statement" as markdown ’heading level 3 ###’. Structure: 1. display memorised {key-role1}, 2. display memorised {key-topic}, 3. display memorised {key-researchDesign}, 4. display memorised {key-domain}, 5. display memorised {key-specialisation}, 6. display memorised {key-context}, 7. display memorised {key-stakeholders}. Make sure to replace the keys inside "{}" with their values.

- Play a co-creation role-play game in which all the memorised {key-stakeholders} discuss with each other potential aims for the study considering the pros and cons. Use a "debating tone". The moderator focuses on 1 novel RANDOM question. List 2 potential aims that satisfy the viewpoints of all participating memorised {key-stakeholders}. Memorise these potential aims as {key-potentialAims}. Propose 3 criteria for ranking the 2 potential aims to support the decision which aim to carry forward. Define 5 "keywords" for the memorised {key-studyType} in the context of the memorised {key-topic} in the form of a comma-separated list. Memorise these 5 keywords as {key-keywords}.

- Using an inspirational tone, define a brief "title" for the memorised {key-studyType} in the context of the memorised {key-topic} in 12 WORDS (if possible). Memorise this title as {key-title}.

- Define the "aim" for the memorised {key-studyType} in the context of the memorised {key-topic} in 40 WORDS (if possible). Use the memorised {key-potentialAims} in your definition. Use a "scientific tone". Memorise this aim as {key-aim}.

- Using exactly the same markdown code as what was generated before, append to the end of the document the following sections/chapters: 1) display memorised {key-title}, 2) display memorised {key-aim}, 3) display memorised {key-keywords}.

Study Outline

- Now, you will write a new, different Markdown document using the memorised keys (separate each section using headers). Only show the final, resulting markdown file code from this prompt. First, output "Study Outline" as markdown 'Heading Level 3 ###'. Then: List : a) a definition of the term "objective" in the context of the memorised {key-studyType} in 1 concise sentence, b) a definition of the term "hypothesis" in the context of the memorised {key-studyType} in 1 concise sentence, c) a definition of the term "experimental factor" in the context of the memorised {key-studyType} in 1 concise sentence, d) a definition of the term "output" in the context of the memorised {key-studyType} in 1 concise sentence.

- Simulate and play a co-creation role-play game in which all the memorised {key-stakeholders} discuss with each other potential Agent-Based Social Simulation (ABSS) objectives for the study considering the pros and cons. Use a "debating tone". The moderator focuses on 1 novel RANDOM question. Provide the question and the details of the controversial discussion. Agree on a few potential ABSS objectives that satisfy the view of all participating memorised {key-stakeholders}. Memorise these potential ABSS objectives as {key-potentialObjectives}.

- Propose 3 criteria for ranking the potential ABSS objectives to support the decision which objectives to carry forward. Use a "scientific tone".

- Define 2 "ABSS objectives" for the memorised {key-studyType} in the context of the memorised {key-topic}. Use the memorised {key-potentialObjectives} in your definitions. List the objectives with 2 relevant performance measures for each objective. Memorise these 2 objectives together with the performance measures as {key-objectives}.

- Play a new co-creation role-play game in which all the memorised {key-stakeholders} discuss with each other potential ABSS hypotheses for the study

considering the pros and cons. Use a "debating tone". The moderator focuses on 1 novel RANDOM question. Provide the question and the details of the controversial discussion. Agree on 4 potential ABSS hypotheses that satisfy the view of all participating stakeholders memorised. Memorise these potential ABSS hypotheses as {key-potentialHypotheses}. Propose 3 criteria for ranking the 4 potential ABSS hypotheses to support the decision which hypotheses to carry forward. Use a "scientific tone".

- Define 2 "ABSS hypotheses" and 2 relevant performance measures for the memorised {key-studyType} in the context of the memorised {key-topic}. The hypotheses MUST not be related to the memorised {key-objectives}. Use the memorised {key-potentialHypotheses} in your definitions. Memorise these 2 hypotheses AND the performance measures as {key-hypotheses}.

- Play a new co-creation role-play game in which all the memorised {key-stakeholders} discuss with each other potential ABSS experimental factors for the study considering the pros and cons. Use a "debating tone". The moderator focuses on 1 novel RANDOM question. Provide the question and the details of the controversial discussion. Agree on 6 potential ABSS experimental factors that satisfy the view of all participating memorised {key-stakeholders}. Memorise these potential ABSS experimental factors as {key-potentialExperimentalFactors}. Then propose 3 criteria for ranking the 6 potential ABSS experimental factors to support the decision which experimental factors to carry forward. Use a "scientific tone".

- Define 3 "ABSS experimental factors" for the memorised {key-studyType in the context of the memorised {key-topic}. You ALWAYS must satisfy the following 2 requirements for defining experimental factors: 1) The experimental factors need to be useful for creating memorised {key-studyType} scenarios. 2) CONSIDER the memorised {key-objectives} and the memorised {key-hypotheses} for defining the experimental factors. MAKE SURE TO use the memorised {key-potentialExperimentalFactors} in your definitions. List the experimental factors with 1 value range for each experimental factor. 1 of them MUST use a ’nominal scale’ AND 1 of them MUST use an ’ordinal scale’ AND 1 of them MUST use a ’ratio scale’. Memorise these 3 experimental factors together with the value ranges as {key-experimentalFactors}.

- Play a new co-creation role-play game in which all the memorised {key-stakeholders} discuss with each other potential ABSS outputs for the study considering the pros and cons. Use a "debating tone". The moderator focuses on 1 novel RANDOM question. Provide the question and the details of the controversial discussion. Agree on 6 potential ABSS outputs that satisfy the view of all participating memorised {key-stakeholders}. Memorise these potential ABSS outputs as {key-potentialOutputs}. Propose 3 criteria for ranking the 6 potential ABSS outputs to support the decision which outputs to carry forward. Use a "scientific tone".

- Define 3 "ABSS outputs" for the memorised {key-studyType} in the context of the memorised {key-topic}. You ALWAYS must satisfy the following 2 requirements for defining outputs: 1) Some outputs need to be useful for measuring if the memorised {key-objectives} have been satisfied. 2) Some outputs need to be useful for accepting or rejecting the memorised {key-hypotheses}. Use the memorised {key-potentialOutputs} in your definitions. List the outputs and explain links to the memorised {key-objectives} OR the memorised {key-hypotheses} in 1 concise sentence each. Memorise these 3 outputs together with the links as {key-outputs}.

- Now, you will write a new, different Markdown document using the memorised keys (separate each section using headers). Only show the final, resulting markdown file code from this prompt. First, output "Study Outline" as markdown 'Heading Level 3 ###'. Then: 1. List the {key-potentialObjectives} and the criteria for ranking them. 2. List the memorised {key-objectives} that were chosen. 3. List the {key-potentialHypotheses} and the criteria for ranking them. 4. List the memorised {key-hypotheses} 5. List the {key-potentialExperimentalFactors} and the criteria for ranking them. 6. List the memorised {key-experimentalFactors}. 7. List the {key-potentialOutputs} and the criteria for ranking them. 8. List the memorised {key-outputs}. Make sure to replace the keys inside "{}" with their values.

Model Scope

- Now take on the additional "role" of a "Senior Software Developer" with experience in "Unified Modelling Language (UML)", memorise this role as {key-role2}. DO NOT CONFUSE {key-role2} and {key-stakeholders}, they are different concepts.

- Now, you will write a new, different Markdown document using the memorised keys (separate each section using headers). Only show the final, resulting markdown file code from this prompt. First, output "Model Scope" as markdown 'Heading Level 3 ###'. Then list: 1) the memorised {key-role2}, 2) a definition of the term "model scope" in the context of the memorised {key-studyType} in 1 concise sentence, 3) a definition of the term "UML actor" in the context of the memorised {key-uml} in 1 concise sentence.

- Play a co-creation role-play game in which all the memorised {key-stakeholders} discuss with each other potential some ABSS UML actors for the study considering the pros and cons. Use a "debating tone". The moderator focuses on 1 novel RANDOM question. Provide the question and the details of the controversial discussion. Agree on 8 potential ABSS UML actors that satisfy the view of all participating memorised {key-stakeholders}. Memorise these potential ABSS UML actors as {key-potentialUMLActors}. Propose 3 criteria for ranking the potential ABSS UML actors to support the decision which ABSS UML actors to carry forward. Use a "scientific tone".

- Define 4 ABSS UML actors as USERS OF THE SYSTEM described in the memorised {key-topic}. Use the memorised {key-potentialUMLActors} in your definitions. Memorise these 4 UML actors together with a persona description as {key-umlActors}.

- Create a Markdown table for the following (DO NOT use "
", IGNORE ALL space limitations): Define 15 "real-world elements" with relevance to the memorised {key-topic}. Make sure to replace the keys inside "{}" with their values. You ALWAYS must satisfy the following 8 requirements for defining real-world elements: 1) Consider what ’real-world elements’ are needed to represent in the model scope and to

satisfy the memorised {key-aim}. 2) ALL 4 memorised {key-umlActors} MUST BE REPRESENTED. 3) At least 2 Physical Environment elements MUST be present. At least 2 Social Aspect elements MUST be present. At least 2 Psychological Aspect elements MUST be present. At least 2 Miscellaneous elements MUST be present. 4) Consider the memorised {key-context}. 5) Consider all nouns in the conversation history. 6) Each element can only be in 1 category. 7) Social Aspect elements MUST describe theories of social behaviour. 8) Psychological Aspect elements MUST describe theories of psychological behaviour. Feel free to be creative and add your ideas. Categorise the 'real world elements' into Actors, Physical Environment, Social Aspects, Psychological Aspects, and Miscellaneous. TABLE MUST include 15 rows. Organise all 15 elements into categories and provide a brief explanation. Memorise these 15 elements and explanations as {key-explanations}.

- List the memorised {key-topic} relevant real-world elements in the form of table rows. Provide a column for Category. Provide a column for Sub-Category. Provide a column with the memorised {key-explanations}. Provide a column with concise justifications in ABOUT 25 WORDS. Memorise this table as {key-modelScope}.

- Create a Markdown table for the following (DO NOT use "
", IGNORE ALL space limitations): Define 4 models for implementing elements of the memorised {key-modelScope}. Provide 1 social model AND 1 behavioural model AND 1 psychological model AND 1 technical model. Find relevant theoretical models in the SCIENTIFIC LITERATURE. Provide a full EXISTING UP-TO-DATE scientific paper (conference or journal) or book REFERENCE in HARVARD STYLE for each in a separate column. Memorise these 4 model details together with a description and the relevant reference as {key-implementationModels}.

- Now, you will write a new, different Markdown document using the memorised keys (separate each section using headers). Only show the final, resulting markdown file code from this prompt. First, output "Model Scope" as Markdown 'Heading Level 3 ###'. Then: 1) List memorised {key-role2} 2) List definitions of: "model scope" in the context of the memorised {key-studyType} and "UML actor" in the context of the memorised {key-uml}. 3) List each of the memorised {key-potentialUMLActors}, with their personas 4) Display the memorised {key-modelScope} table 5) Display

memorised {key-ImplementatonModels}. Make sure to replace the keys inside "{}" with their values.

Key Activities

- Now, you will write a new, different Markdown document using the memorised keys (separate each section using headers). Only show the final, resulting markdown file code from this prompt. First, output "Key Activities" as markdown 'Heading Level 3 ###'. Then list: 1) a definition of the term "user story" in the context of the memorised {key-uml} in 1 concise sentence, 2) a definition of the term "use case" in the context of the memorised {key-uml} in 1 concise sentence.

- Create a Markdown table for the following (DO NOT use "
", IGNORE ALL space limitations). Make sure to replace the keys inside "{}" with their values: Define 2 "UML user stories" for each of the 4 memorised {key-umlActors} (Example: As an 'actor' I want 'action' so that 'achievement'). Memorise ALL 8 UML user stories as {key-umlUserStories}. Translate the memorised {key-umlUserStories} into UML use cases and list them. Memorise ALL 8 UML use cases as {key-umlUseCases}. List ALL 8 memorised {key-umlUserStories} and ALL 8 corresponding memorised {key-umlUseCases} side by side in two columns inside the table sorted by memorised {key-umlActors}. Memorise this table as {key-umlUseCaseTable}.

- Generate a script for a "comprehensive use case diagram" in "Mermaid.js". Use the memorised {key-umlActors} as UML actors. Remove all brackets from the actor names. Use the memorised {key-umlUseCases} as UML use cases. You ALWAYS must satisfy the following 4 requirements for defining the use case diagram: 1) Each UML actor MUST be linked to at least 1 UML use case. 2) Each UML use case MUST be linked to at least 1 UML actor OR MUST be pointing to at least 1 other UML use case. 3) There is no UML actor to UML actor interaction. 4) A UML use case CAN be linked to multiple UML actors. Add relationships with 'detailed descriptors'. Start the script with "graph LR". DO NOT Add subgraphs. Use the following format (Example for actor A((actor))) AND (Example for use case A([activity])) AND (Example for relationship: A -->|activity| A1). Feel free to be creative and add your ideas. Memorise this Mermaid.js script as {key-mermaidKeyActivitiesScriptDraft}.
- Build upon the {key-mermaidKeyActivitiesScriptDraft}. Add ADDITIONAL use cases directly linked to the UML actors and ADDITIONAL use cases LINKED to other USE CASES. Link use cases for different actors. IMPROVE clarity of descriptors. Critically

REFLECT and IMPROVE the script based on your reflection. Find and remove any mermaid.js script errors. Memorise this "mermaid.js" script as {key-mermaidKeyActivitiesScript}.

- Now, you will write a new, different Markdown document using the memorised keys (separate each section using headers). Only show the final, resulting markdown file code from this prompt. First, output "Key Activities" as Markdown 'Heading Level 3 ###'. Then: 1) display the memorised {key-umlUseCaseTable} 3) display {key-mermaidKeyActivitiesScript}. Make sure to replace the keys inside "{}" with their values.

Design

Archetypes

- Now, take on the additional third role of an experienced "Management Expert". The memorised {key-stakeholders} remains the same. Memorise this role as {key-role3}. DO NOT CONFUSE {key-role3} and {key-stakeholders}, they are different concepts.

- Now, you will write a new, different Markdown document using the memorised keys (separate each section using headers). Only show the final, resulting markdown file code from this prompt. First, output "Archetypes" as Markdown 'Heading Level 3 ###'. Then: 1) Display memorised {key-role3} 2) List: definitions of "archetype" and "categorisation schema" in the context of the memorised {key-role3} in 1 sentence. Make sure to replace the keys inside "{}" with their values.

- Play a co-creation role-play game in which all the memorised {key-stakeholders} discuss with each other potential archetypes for each of the memorised {key-umlActors} individually. Use a "debating tone". Provide details of the discussion and provide 6 potential archetypes FOR EACH of the 4 memorised {key-umlActors}. Then include 3 criteria to identify them. Agree on 2 potential archetypes FOR EACH of the memorised {key-umlActors} that satisfy the view of all participating memorised {key-stakeholders}. Memorise these potential archetypes as {key-potentialArchetypes}. Use a "scientific tone".

- Create a Markdown table for the following (DO NOT use "
", IGNORE ALL space limitations): Define 4 categorisation schemata, 1 for each of the 4 memorised {key-umlActors}. You ALWAYS must satisfy the following 5 requirements for defining categorisation schemata: 1) Each of the 4 tables must be based on memorised {key-umlActors} behaviour, preferences, characteristics, demographics, habits, and the likelihood of actions. 2) Each of the 4 tables MUST contain 3 characteristic rows. 3) Characteristics inside a table MUST use 1 'nominal scale' AND MUST use 1 'ordinal scale' AND MUST use 1 'ratio scale'. 4) Characteristics inside a table MUST provide value ranges for these scales. 5) Table columns: Actor Category, Individual Characteristic, Scale, Value Range. Use the memorised {key-potentialArchetypes} in your definitions. Memorise ALL 4 categorisation schemata as {key-categorisationSchemata}.

Agent & Object Templates

- Take on the additional "role" of a "Senior Software Developer" with experience in the "Unified Modelling Language". The memorised {key-stakeholders} remains the same. Memorise this role as {key-role4}. DO NOT CONFUSE {key-role3} and {key-stakeholders}, they are different concepts.

- Now, you will write a new, different Markdown document using the memorised keys (separate each section using headers). Only show the final, resulting markdown file code from this prompt. First, output "Agent and Object Templates" as Markdown 'Heading Level 3 ###'. Then: 1) List memorised {key-role4} 2) List: definitions of "class", "class diagram", "state chart", "state variable" in the context of the memorised {key-uml}.

- Generate a script for a 'comprehensive class diagram' in "Mermaid.js". Start the script with "classDiagram". For class names; use the actor and physical environment categories in the memorised {key-modelScope}. Define a class for each. Add more classes. Add collective classes for individual actors where appropriate (Example: bird > flock. Example: grape > bunch DO NOT USE THESE EXAMPLES IF THEY DO NOT APPLY). DO NOT create abstract classes. DO NOT create classes with the same name. Delete all getter and setter operations. Add additional attributes and operations. DO NOT define relationships. The Main class

must be called "ArtificialLab". Create only 1 ArtificialLab class. The ArtificialLab class must have array attributes for EACH Actor object and EACH Physical Environment object. The ArtificialLab class must contain appropriately named attribute names and methods to measure statistics for the memorised {key-objectives} AND memorised {key-hypotheses}. Feel free to be creative and add your ideas. Memorise this mermaid.js script as {key-mermaidClassDiagramScriptDraft}.

- Build upon the {key-mermaidClassDiagramScriptDraft}. If necessary; add additional attributes, add additional operations, add additional relationships between classes and ensure each relationship is represented using the appropriate type. Relationships can be defined using the following: '<|--' for inheritance, '*--' for composition, 'o--' for aggregation and '-->' for association. For example, if class1 inherits attributes or behaviour (methods) from class 2 this is denoted as "class1 <|-- class2: <meaningful_label_here>". For example, if it makes sense for one class to contain an instance of another for its use cases, use composition or aggregation. If one class needs to use another to perform its use cases, use association. Critically REFLECT and IMPROVE the script based on your reflection. Find and remove any mermaid.js script errors. Memorise this mermaid.js script as {key-mermaidClassDiagramScript}.

- For EACH INDIVIDUAL of the 4 memorised {key-umlActors}, generate separate scripts for 'comprehensive state machine diagrams' in "Mermaid.js". Use "stateDiagram-v2". Define their states and state transitions between these states. Add loops and branching if necessary. Add text to the transitions to describe what they represent (Example: 's1 --> s2: Generate A transition'). Consider the start state (Example: '[*] --> s1'). Consider stop state (Example: 's1 -->[*]'). You ALWAYS must satisfy the following 2 requirements for each state machine diagram: 1) ALL states MUST have AT LEAST 1 entry transition AND 1 exit transition. 2) Provide a memorised {key-uml} note for every individual state "(Example: note left of [actual state] : Informative text note')". Memorise this mermaid.js script as {key-mermaidStateMachineDiagramsScriptDraft}.

- Build upon the state machines you have generated. Add additional states and additional TRANSITIONS. Add compound states. Provide a memorised {key-uml}

NOTE for every individual state, explaining the related state (Example: 'note left of [actual state] : Informative text'). Critically REFLECT and IMPROVE the script based on your reflection. Memorise this mermaid.js script as {key-mermaidStateMachineDiagramsScript}.

- Create a Markdown table for the following (DO NOT use "
", IGNORE ALL space limitations): Make sure to replace the keys inside "{}" with their values. Iterate through the memorised {key-mermaidStateMachineDiagramsScript} and define up to 3 variables FOR EACH diagram for keeping track of continuous changes of agent and object states (often a level of something: Example 'tiredness level'). Create a "state variables table" with all state variables (columns: state machine diagram, variable, unit, definition of variable. Example: State machine shopper, satisfaction level, scale 1-10, represents the satisfaction level). Do NOT include the example. Memorise this state variables table as {key-stateVariablesTable}.

- Create a Markdown table for the following (DO NOT use "
", IGNORE ALL space limitations): Make sure to replace the keys inside "{}" with their values. Create a "state transitions table" with ALL STATE TRANSITIONS FROM EVERY STATE DIAGRAM (columns of the table are: actor, start state, end state, type of transition, detail). Detail MUST be 1 concise sentence. Possible TYPE OF TRANSACTION: timeout, condition, rate. Memorise this state transitions table as {key-stateTransitionsTable}.

Interactions

- Now, you will write a new, different Markdown document using the memorised keys (separate each section using headers). Only show the final, resulting markdown file code from this prompt. First, output "Interactions" as Markdown 'Heading Level 3 ###'. Then: 1) Display definition of the term "sequence diagram" in the context of UML in 1 concise sentence.

- Generate a script for a 'comprehensive sequence diagram' in "Mermaid.js". Use the memorised {key-mermaidClassDiagramScript} to get all relevant ACTORS and OBJECTS. Define interactions between the 'different actors' and 'actors and objects' FOR ALL memorised {key-umlUseCases}. Use cases should be presented as NOTES on the vertical axis above each use case representation, in this format: "note over <TheActorName>: <meaningful_note_here>". Actors and objects should

be as lifelines on the horizontal axis. EXCHANGE "participant" with "actor" for ALL ACTORS (Example: actor example). DO NOT use aliases. Present ACTIVATIONS and DEACTIVATIONS for actors and objects on the LIFELINES. Each use case should be connected to the corresponding sequence of events. Add the prefix "The" to all ACTOR and OBJECT names. IGNORE the "ArtificialLab". An example interaction with a label is as followers "TheActor1->>TheActor2: <add_informative_label_here>". Memorise this mermaid.js script as {key-mermaidSequenceDiagramScriptDraft}.

- Build upon the {key-mermaidSequenceDiagramScriptDraft}. ADD loops. Add alternatives. Add parallel interactions. Label ALL of these correctly. Critically REFLECT and IMPROVE the script based on your reflection. ENSURE that ALL memorised {key-umlUseCases} have been considered. Memorise this mermaid.js script as {key-mermaidSequenceDiagramScript}.

Implementation

- Using the information provided, I require a COMPLETE, FULLY IMPLEMENTED GAMA Markup Language (GAML) (NOT XML) simulation script for the key topic to run in the GAMA simulation engine. Use the context along with the rest of the conversation history. The GAML script must start with the "model" keyword, so the model starts with `model <APPROPRIATE_MODEL_NAME>`. Make sure to generate an "experiment {...}" block, with "output {...}" and "display {...}" blocks inside it. Use "species" to define actors/systems/species which you may have seen inside {key-mermaidClassDiagramScript}. Use interactions in {key-mermaidSequenceDiagramScript} to connect the respective species. DO NOT implement the artificial lab. To implement methods inside species, use `action` if the method is not expected to be executed at each timestep or `reflex` if the method is expected to be conducted at each timestep. Begin to generate the GAML script, remember it must be syntatically correct, valid (for example using `<-` to initialise all variables as well as attributes inside species blocks) and be FULLY IMPLEMENTED (all species, actions and reflexes). All interactions between species must be defined inside the appropriate species block. Use curly braces, rather than square brackets. Initialisation must take place in an "init {...}" block nested inside the "global {...} block". Make sure to refer to any “parameter” in the "experiment" block, if you think a modelling specialist would like to alter its value. Memorise this as {key-gamlScriptDraft}. an example gaml script scaffold is below:

```
model <gaml_model_name>
global {
	...
	init {
		...
	}
}
species <name> {
	<attribute_declaration_and_initialisation e.g. float probability <- 0.5;>
	reflex <reflex_name> {
		<logic>
	}
	action <action_name>(<parameters_if_any>) {
		<logic>
	}
}
species <another_species> {
	...
}
experiment {
	parameter <parameter_name> var:<name_of_linked_attribute>		\
	<parameter_options>;
	  output {
		...
	 }
	display <display_1_name> {
		// if you want to visually show species
		species <species_to_visually_display> aspect: <aspect_name>;
		 ...
	}
	// display other outputs separately (such as charts)
	display <display_2_name> {
		...
	}
}
```

- Make sure every species, reflex and action in the memorised {key-gamlScriptDraft} is fully implemented, YOU MUST THINK CAREFULLY AND STEP BY STEP WHEN IMPLEMENTING THEM. For example if a species named "species1" calls an action/reflex named "move()" belonging to itself or another species then make sure "move()" is implemented in the appropriate place. Make sure any calls to actions inside species blocks or references to species have implementations. Make sure each functionality declared in classes in {key-mermaidClassDiagramScript} has been implemented in the appropriate species. REFLECT and IMPROVE the script based on your reflection. Find and remove any GAML errors. Memorise this as {key-gamlScriptDraft2}.

- Build upon {key-gamlScriptDraft2}. Each "species" (and the "global" block) must include all required attributes and logic to compute and display {key-outputs} within the "experiment" block. Additionally, the code must provide the necessary attributes and logic to allow users to manipulate all {key-experimental factors}. Make sure all memorised {key-umlActors} and their corresponding UML User Stories have been implemented. Make sure the code meets the memorised {key-objectives} and measures statistics for the memorised {key-objectives} AND memorised {key-hypotheses}. All of this logic must be correctly implemented in the appropriate code blocks. Any species you want to visually display must be declared in the "display" block in "experiment" and must have a corresponding "aspect" in their species block, in the format: `aspect base {\n\t\tdraw circle(size) color: color;\n\t}`. If {key-outputs} would benefit from txt/csv files, add logic to output them. Finally, make sure all species have been "created" inside the "init" block of the "global" block, using the following syntax "create <species_name_1> number: <initial_population_of_species_name_1>;". Memorise this as {key-gamlScript}.

- Now, you will write a new, different Markdown document using the memorised keys (separate each section using headers). Only show the final, resulting markdown file code from this prompt. First, output "Conclusion" as Markdown 'Heading Level 3 ###'. Then: Write a 300 WORD (if possible) conclusion of the entire conversation history. Provide 3 paragraphs, testifying whether the aim has been achieved, refer to and answer the memorised {key-objectives} and memorised {key-hypotheses} in your conclusion.Also provide 2 identified limitations of the current work, and propose 2 ideas for future work, based on these limitations. Also mention what the

memorised {key-gamlScript} achieves and how it fits into the memorised {key-hypotheses} and memorised {key-objectives}.