**Social Robotics Use Cases**

**Use Case 1: Load Master Module**

Actors: Robot, Subject

Entry Conditions: The Robot has been turned on.

Exit Conditions: The master module has been loaded.

Flow of Events:

1. Robot Startup
2. Load Activities Master Module
3. Detect Subject

Exceptions:

**Use Case 1.1: Robot Startup**

Actors: Robot

Entry Conditions: The Robot has been turned on.

Exit Conditions: The Robot has completed its startup sequence.

Flow of Events:

Exceptions:

**Use Case 1.2: Load Activities Master Module**

Actors: Robot

Entry Conditions: The robot has been turned on and completed its startup sequence.

Exit Conditions: The activities master module has been loaded.

Flow of Events:

Exceptions:

**Use Case 1.3: Detect Subject**

Actors: Robot, Subject

Entry Conditions: The activities master module has been loaded and its initial sequence begun.

Exit Conditions: The subject has been detected.

Flow of Events:

Exceptions:

**Use Case 2: Initialize Activities Master Module**

Actors: Robot, Subject

Entry Conditions: The activities master module has been loaded and the subject has been detected.

Exit Conditions: The subject has been greeted and all activity modules have been discovered.

Flow of Events:

1. Greet Subject
2. Discover Available Activity Modules

Exceptions:

**Use Case 2.1: Greet Subject**

Actors: Robot, Subject

Entry Conditions: The subject has been detected.

Exit Conditions: The subject has been greeted.

Flow of Events:

Exceptions:

**Use Case 2.2: Discover Available Activity Modules**

Actors: Robot

Entry Conditions: The activities master module has been loaded.

Exit Conditions: All activity modules have been discovered.

Flow of Events:

Exceptions:

**Use Case 3: Start Activity**

Actors: Robot, Subject

Entry Conditions: The activities master module has been initialized.

Exit Conditions: An activity module has been selected, loaded, and its initial operation sequence begun.

Flow of Events:

1. Explain Commands
2. Explain Activities
3. Select Activity
4. Listen for Selection
5. Validate Activity Selection

Exceptions:

**Use Case 3.1: Explain Commands**

Actors: Robot, Subject

Entry Conditions: The activities master module has been initialized.

Exit Conditions: The available master-level commands have been explained to the subject.

Flow of Events:

Exceptions:

**Use Case 3.2: Explain Activities**

Actors: Robot, Subject

Entry Conditions: The activities master module has been initialized and all activity modules discovered.

Exit Conditions: The available activities modules have been explained.

Flow of Events:

Exceptions:

**Use Case 3.3: Select Activity**

Actors: Robot, Subject

Entry Conditions: The available activities have been explained to the subject.

Exit Conditions: The subject has selected a valid activity.

Flow of Events:

Exceptions:

**Use Case 3.4: Listen for Selection**

Actors: Robot, Subject

Entry Conditions: The activities master module has been initialized and the subject identified.

Exit Conditions: The Robot has heard a potential selection or command.

Flow Events:

Exceptions:

**Use Case 3.5: Validate Activity Selection**

Actors: Robot

Entry Conditions: The robot has heard a potential activity selection.

Exit Conditions: The robot has identified a potential activity selection as valid or invalid.

Flow of Events:

Exceptions:

**Use Case 3.6: Handle Invalid Activity Selection**

Actors: Robot, Subject

Entry Conditions: The robot has identified an invalid activity selection.

Exit Conditions: The robot has notified the user of the invalid selection and prompted for the next action to be taken.

Flow of Events:

Exceptions:

**Use Case 3.7: Load Selected Activity Module**

Actors: Robot

Entry Conditions: The robot has identified a valid activity selection.

Exit Conditions: The activity module has been loaded.

Flow Events:

Exceptions:

**Use Case 3.8: Start Activity Module Initialization Sequence**

Actors: Robot

Entry Conditions: An activity module has been loaded.

Exit Conditions: The activity module’s initialization sequence has begun.

Flow Events:

Exceptions:

**Use Case 4: Select Object to Draw**

Actors: Robot, Subject

Entry Conditions: The Social Interactive Drawing Module has been loaded and initialized.

Exit Conditions: A valid object has been selected by the subject for the robot to draw.

Flow Events:

Exceptions:

**Use Case 4.1: List Drawable Objects**

Actors: Robot, Subject

Entry Conditions: The Social Interactive Drawing Module has been loaded and initialized.

Exit Conditions: The available objects that can be drawn have been listed to the subject.

Flow Events:

Exceptions:

**Use Case 4.2: Prompt Object Selection**

Actors: Robot, Subject

Entry Conditions: The robot has listed all available objects which can be drawn to the subject.

Exit Conditions: The robot has prompted the subject to select an object to draw.

Flow Events:

Exceptions:

**Use Case 4.3: Validate Object Selection**

Actors: Robot

Entry Conditions: The robot has heard a potential object selection.

Exit Conditions: The robot has identified whether the object selection is valid or invalid.

Flow Events:

Exceptions:

**Use Case 4.4: Handle Invalid Object Selection**

Actors: Robot, Subject

Entry Conditions: The robot has identified an invalid object selection.

Exit Conditions: The robot has notified the subject of the invalid selection and prompted for the next action to be taken.

Flow Events:

Exceptions:

**Use Case 5: Obtain Writing Implement**

Actors: Robot, Subject

Entry Conditions: The robot has been instructed to draw a valid object.

Exit Conditions: The robot has obtained a writing implement which is oriented correctly for drawing.

Flow Events:

Exceptions:

**Use Case 5.1: Determine if Holding Marker**

Actors: Robot

Entry Conditions: The robot has been instructed to draw a valid object.

Exit Conditions: The robot has determined whether it is already holding a marker or not.

Flow Events:

Exceptions:

**Use Case 5.2: Ask Whether to Use Current Marker**

Actors: Robot, Subject

Entry Conditions: The robot has determined that it is already holding a marker.

Exit Conditions: The robot has determined whether it should continue using the current marker or obtain a new one.

Flow Events:

Exceptions:

**Use Case 5.3: Validate Marker Selection**

Actors: Robot

Entry Conditions: The robot has heard a potential marker selection instruction.

Exit Conditions: The robot has identified whether the selection instruction statement is valid or invalid.

Flow Events:

Exceptions:

**Use Case 5.4: Drop Marker**

Actors: Robot

Entry Conditions: The robot needs to obtain a new marker but is currently holding one.

Exit Conditions: The robot is no longer holding the marker.

Flow Events:

Exceptions:

**Use Case 5.4: Request New Marker**

Actors: Robot, Subject

Entry Conditions: The robot has determined that it needs to obtain a new marker and is not currently holding a marker.

Exit Conditions: The robot has requested the subject to provide a new marker and the subject has placed the marker in its hand.

Flow Events:

Exceptions:

**Use Case 5.5: Evaluate Marker Orientation**

Actors: Robot

Entry Conditions: The robot has a marker in its hand.

Exit Conditions: The robot has determined whether the marker is oriented correctly for drawing.

Flow Events:

Exceptions:

**Use Case 5.6: Evaluate Whether Cap On/Off**

Actors: Robot

Entry Conditions: The robot has a marker in its hand which has been determined to be oriented correctly.

Exit Conditions: The robot has determined whether the marker’s cap is on or off.

Flow Events:

Exceptions:

**Use Case 5.7: Handle Invalid Marker Orientation**

Actors: Robot, Subject

Entry Conditions: The robot has determined that it is holding a marker in an incorrect orientation for drawing.

Exit Conditions: The robot has notified the subject that the marker is incorrectly oriented and prompted for the next action to be taken.

Flow Events:

Exceptions:

**Use Case 5.8: Handle Marker Cap On**

Actors: Robot, Subject

Entry Conditions: The robot has determined that it is holding a marker with its cap on.

Exit Conditions: The robot has prompted the subject to remove the cap and the subject has done so.

Flow Events:

Exceptions:

**Use Case 6: Draw Object**

Actors: Robot, Subject

Entry Conditions: The robot has been instructed to draw a valid object and is holding a writing implement properly for drawing.

Exit Conditions: The robot has drawn the requested object on the drawing surface.

Flow Events:

Exceptions:

**Use Case 6.1: Locate Drawing Surface**

Actors: Robot, Subject

Entry Conditions: The robot has been instructed to draw a valid object and is holding a writing implement properly for drawing.

Exit Conditions: The robot has located a drawing surface.

Flow Events:

Exceptions:

**Use Case 6.2: Determine Drawing Surface Boundaries**

Actors: Robot

Entry Conditions: The robot has located a drawing surface.

Exit Conditions: The robot has identified the boundaries of the drawing surface.

Flow Events:

Exceptions:

**Use Case 6.3: Evaluate Drawing Surface Accessibility**

Actors: Robot

Entry Conditions: The robot has located a drawing surface and identified its boundaries.

Exit Conditions: The robot has evaluated whether a drawing surface is accessible or inaccessible.

Flow Events:

Exceptions:

**Use Case 6.4: Activate Advanced Motor Control Module**

Actors: Robot

Entry Conditions: The robot has identified an accessible drawing surface.

Exit Conditions: The robot has activated the advanced motor control module.

Flow Events:

Exceptions:

**Use Case 6.5: Execute Object Drawing Instructions**

Actors: Robot

Entry Conditions: The robot has been instructed to draw a valid object and has activated the advanced motor control module.

Exit Conditions: The robot has drawn the instructed object on the drawing surface.

Flow Events:

Exceptions:

**Use Case 6.6: Handle Inaccessible Drawing Surface**

Actors: Robot, Subject

Entry Conditions: The robot has located a drawing surface but has determined that is inaccessible.

Exit Conditions: The robot has notified the subject of the inaccessible drawing surface and prompted the subject to position it properly.

Flow Events:

Exceptions:

**Use Case 6.7: Handle No Drawing Surface**

Actors: Robot, Subject

Entry Conditions: The robot has been unable to locate a drawing surface in the environment.

Exit Conditions: The robot has notified the subject that it cannot locate a valid drawing surface and has prompted the subject to select the next action.

Flow Events:

Exceptions:

**Use Case 7: Interact with Subject**

Actors: Robot, Subject

Entry Conditions: The robot has determined that it needs to interact with the subject.

Exit Conditions: The robot has completed the interaction with the subject.

Flow Events:

Exceptions:

**Use Case 7.1: Select Interaction Statement**

Actors: Robot

Entry Conditions: The robot has determined that it needs to speak to the subject.

Exit Conditions: The robot has selected an interaction statement to speak to the subject.

Flow Events:

Exceptions:

**Use Case 7.2: Speak Interaction Statement**

Actors: Robot, Subject

Entry Conditions: The robot has selected an interaction statement to speak to the subject.

Exit Conditions: The robot has spoken the interaction statement to the subject.

Flow Events:

Exceptions:

**Use Case 7.3: Listen for Question Response**

Actors: Robot, Subject

Entry Conditions: The robot has asked the subject a question.

Exit Conditions: The robot has heard a possible response to the question.

Flow Events:

Exceptions:

**Use Case 7.4: Evaluate Question Response**

Actors: Robot

Entry Conditions: The robot has heard a possible response to a question.

Exit Conditions: The robot has evaluated a subject’s response to a question and determined the next action to take.

Flow Events:

Exceptions:

**Use Case 8: Complete Activity**

Actors: Robot, Subject

Entry Conditions: The robot has completed the sequence for an activity.

Exit Conditions: The robot has determined whether to exit or restart the activity.

Flow Events:

Exceptions:

**Use Case 8.1: Prompt Whether to Continue Activity**

Actors: Robot, Subject

Entry Conditions: The robot has recognized some condition indicating that an activity may need to be ended or restarted.

Exit Conditions: The robot has prompted the subject to select the next action.

Flow Events:

Exceptions:

**Use Case 8.2: Restart Activity Module**

Actors: Robot

Entry Conditions: The robot has determined that an activity module needs to be restarted.

Exit Conditions: The robot has started the activity module’s initialization sequence.

Flow Events:

Exceptions:

**Use Case 8.3: Exit Activity Module**

Actors: Robot

Entry Conditions: The robot has determined that it needs to exit the current activity module.

Exit Conditions: The robot has exited the current activity module and returned to the activity selection stage.

Flow Events:

Exceptions: