MIDDLE EAST TECHNICAL UNIVERSITY
COMPUTER ENGINEERING DEPARTMENT

Software Architecture Desgin

YOLO

Group 71
Mustafa Sezgin - 2380863

Table of Contents

1 Introduction 1

1.1 Purpose and objectives of the YOLO social robot 1

1.2 Scope 1

1.3 Stakeholders and their concerns 2

2 References 3

3 Glossary 4

4 Architectural Views 5

4.1 Context View 5

4.1.1 Stakeholders' use of this view 5

4.1.2 Context Diagram 5

4.1.3 External Interfaces 5

4.1.4 Interaction Scenarios 5

4.2 Functional View 11

4.2.1 Stakeholders' use of this view 11

4.2.2 Component Diagram 11

4.2.3 Internal Interfaces 12

4.2.4 Interaction Patterns 12

4.3 Information View 20

4.3.1 Stakeholders' use of this view 20

4.3.2 Database Class Diagram 20

4.3.3 Operations on Data 20

4.4 Deployment View 25

4.4.1 Stakeholders' use of this view 25

4.4.2 Deployment Diagram 25

4.5 Design Rationale 25

List of Figures

4.1 *Context Diagram* 5
4.2 *External Interfaces Class Diagram* 8
4.3 *Perform Attention Call Behavior Activity Diagram* 9
4.4 *Perform Story Arc Behaviors Activity Diagram* 10
4.5 *Component Diagram* 11
4.6 *Internal Interfaces Class Diagram* 16
4.7 *Determine Movement Shape Sequence Diagram* 17
4.8 *Perform Slow Mirror Behavior Sequence Diagram* 18
4.9 *Perform Fast Contrast Behavior Sequence Diagram* 19
4.10 *Database Class Diagram* 24
4.11 *Deployment Diagram* 25

List of Tables

3.1 *Glossary* 4
4.1 *Perform Attention Call Behavior Function* 6
4.2 *Perform Story Arc Behaviors Function* 7
4.3 *Determine Movement Shape Function* 13
4.4 *Perform Slow Mirror Behavior Function* 14
4.5 *Perform Fast Contrast Behavior Function* 15
4.6 *CRUD Operations* 23

1 Introduction

1.1 Purpose and objectives of the YOLO social robot

YOLO is a social robot that encourages children's creativity. This robot was designed to be utilized by kids during free play, when they may use it as a character in their stories. YOLO employs creative tactics during play to encourage the invention of new story-lines. As a result, throughout the engagement, the robot functions as a tool that has the ability to promote creativity in youngsters. YOLO, in particular, can encourage divergent and convergent thinking when it comes to tale development. YOLO may also take on numerous personas, allowing it to engage in socially sophisticated and engaging actions.

1.2 Scope

- The robot must have a low floor wide walls design principle.
  - The robot must have a few and simple behaviors and promote quick understanding and engagement without any previous experience.
  - The robot behaviours must be non-directional and enable the user to create any story content.
- The robot must have a child-proof body shell that is 3D printable.
- The robot must have a control module that gets information from the sensors and sends information to the actuators.
  - The robot must have capacitive touch sensor to detect whether it is being touched by the user or not. The robot must stop performing any behavior and listen to movements of the user while it is being touched. The robot must return to fully autonomous mode and start performing behaviours when the user stops touching it.
  - The robot must have an optical sensor to detect the directions and the speeds of movements performed by the user on it.
  - The robot must have an LED actuator to control lights that display different colors in different brightness levels to interact with the user.

  - The robot must have a wheel actuator to control the wheel and move the robot in the desired direction at the desired speed.
- The robot must use the AI system interface to send obtained movement information and recognize the shape of the movement.
- The AI system must have a machine learning model that can recognize any shape in the form of a circle, rectangle, loop, curl, spike, and line with at least 80% accuracy.
- The robot must have a planning module to determine its next behavior.
  - The robot must perform mirroring behaviors in the rising action or falling action phases of the storytelling arc.
  - The robot must perform contrasting behaviors in the climax phase of the storytelling arc.

1.3 Stakeholders and their concerns

The stakeholders of the YOLO social robot are children, parents, and teachers.

Children Children are the main actors that interact with the robot. Their concern is just to play with a toy that acts as a human and have fun.

Parents Parents are the parents of the children. Their concern is the development of their children's creativity skills.

Teachers Teachers are the teachers of the children. Their concern is the development of their students' fast learning skills.

2 References

[1] Alves-Oliveira, Patrícia et al. "Children as Robot Designers". In: *Proceedings of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*. HRI '21. Boulder, CO, USA: Association for Computing Machinery, 2021, pp. 399–408. ISBN: 9781450382892. DOI: `https://doi.org/10.1145/3434073.3444650`. URL: `https://doi.org/10.1145/3434073.3444650`.

[2] Alves-Oliveira, Patrícia et al. "Guide to build YOLO, a creativity-stimulating robot for children". In: *HardwareX* 6 (2019), e00074. ISSN: 2468-0672. DOI: `https://doi.org/10.1016/j.ohx.2019.e00074`. URL: `https://www.sciencedirect.com/science/article/pii/S2468067218300890`.

[3] Alves-Oliveira, Patrícia et al. "Software architecture for YOLO, a creativity-stimulating robot". In: *SoftwareX* 11 (2020), p. 100461. ISSN: 2352-7110. DOI: `https://doi.org/10.1016/j.softx.2020.100461`. URL: `https://www.sciencedirect.com/science/article/pii/S2352711019302468`.

3 Glossary

Term	Definition
AI	Artificial Intelligence
Climax	The second phase of the storytelling arc, where the story reaches the point of greatest tension.
Falling Action	The third phase of the storytelling arc, where the story reaches its end.
GitHub	A cloud host for software version control based on Git.
ML	Machine Learning
Rising Action	The first phase of the storytelling arc, where characters are introduced, and the story builds.
Wi-Fi	Wireless Fidelity
YOLO	Your Own Living Object, the name of the robot.

Table 3.1: *Glossary*

4 Architectural Views

4.1 Context View

4.1.1 Stakeholders' use of this view

4.1.2 Context Diagram

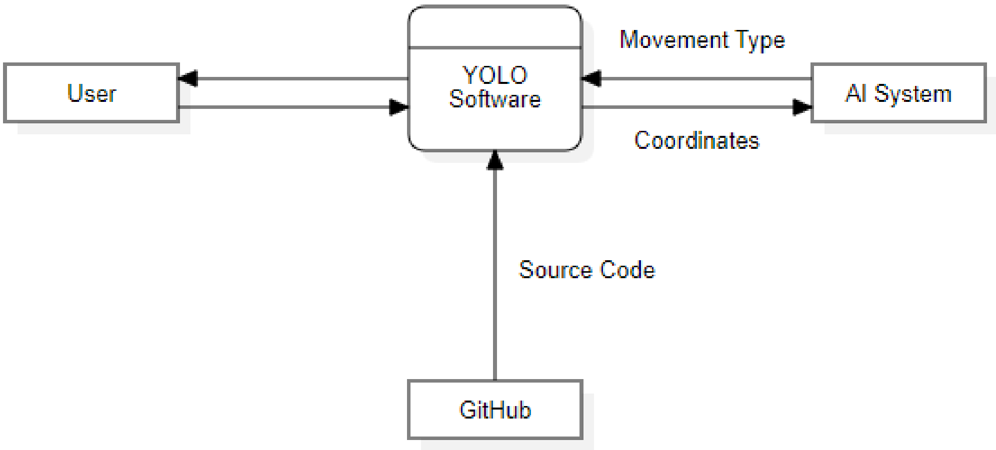

Figure 4.1: *Context Diagram*

User is the children who interact with the robot by touching, moving, and untouching it. AI system is the system that takes coordinates, movement direction and speed from the robot, runs the machine learning algorithm on these inputs, and give the determined movement type to the robot.

4.1.3 External Interfaces

The class diagram for external interfaces is shown in Figure 4.2.

4.1.4 Interaction Scenarios

Use-case name	Perform Attention Call Behavior
Actors	AI System

Description	YOLO is a socially interactive robot. Therefore, it alerts the user by performing attention call behavior if the user is not in contact with it for a while. The minimum time required for this behavior is called attention call threshold.
Data	Touch information, movement speed and direction, LED color and brightness
Preconditions	The user must not be touching the robot for some time that is greater than the attention call threshold.
Stimulus	The time since the last touch passes the attention call threshold.
Basic flow	Step 1: The user stops touching the robot. Step 2: The user leaves the robot untouched for a while. Step 3: If the time without touching goes beyond attention call threshold, attention call behavior is performed.
Alternative flow	-
Exception flow	When the maximum interaction time is achieved, the interaction terminates and the robot turns off.
Postconditions	Idle behavior is performed.

Table 4.1: *Perform Attention Call Behavior Function*

The activity diagram for Perform Attention Call Behavior use-case is shown in Figure 4.3.

Use-case name	Perform Story Arc Behaviors
Actors	AI System

Description	The robot predicts the shape of movement the user makes and imitates it if the current storytelling phase is rising or falling action, or contrasts it if the current storytelling phase is climax. This behavior stimulates convergent or divergent thinking.
Data	Movement shape, speed and direction
Preconditions	The robot must be in any phase of the storytelling arc, and the touch duration must be greater than the shape threshold.
Stimulus	The user stops touching the robot after touching it for a longer time than the shape threshold.
Basic flow	Step 1: The user touches the robot. Step 2: The user moves the robot. Step 3: The user stops touching the robot. Step 4: If the touch duration is greater than the shape threshold, AI System predicts the movement shape. Step 5: If the robot is in the rising or falling stage of the storytelling arc, then the mirroring behavior is performed, else, if the robot is in the climax stage of the storytelling arc, then the contrasting behavior is performed. Step 6: The agent activates actuators via control module to perform the behavior.
Alternative flow #1	Step 4: If the touch duration is smaller than the shape threshold, Idle Behavior is performed.
Alternative flow #2	-
Exception flow	Step 4: If the shape is not recognized by the AI System, the exception is logged and the default shape is returned.
Postconditions	Idle behavior is performed.

Table 4.2: *Perform Story Arc Behaviors Function*

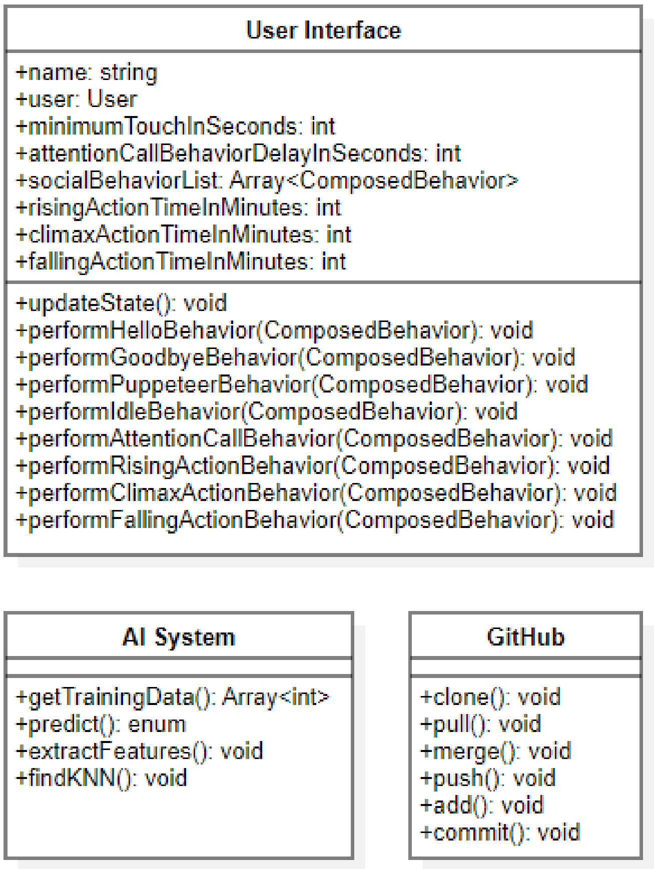

Figure 4.2: *External Interfaces Class Diagram*

The activity diagram for Perform Story Arc Behaviors use-case is shown in Figure 4.4.

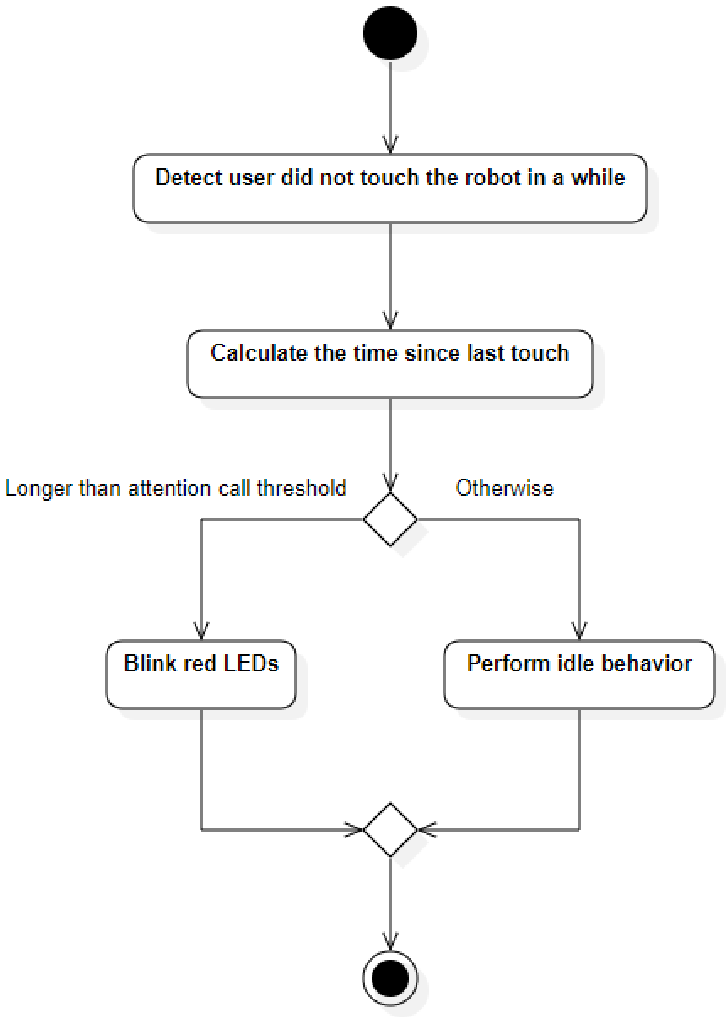

Figure 4.3: *Perform Attention Call Behavior Activity Diagram*

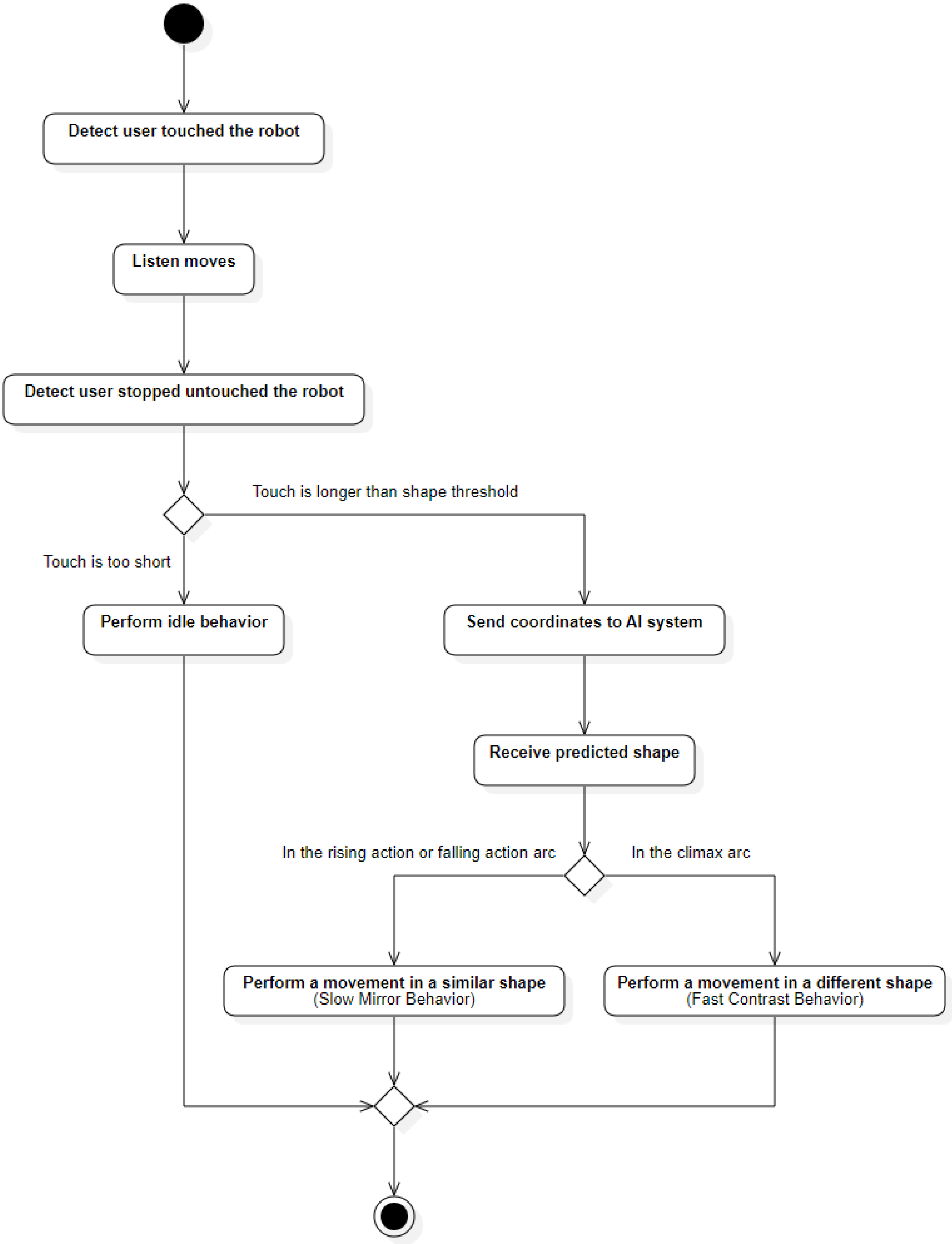

Figure 4.4: *Perform Story Arc Behaviors Activity Diagram*

4.2 Functional View

4.2.1 Stakeholders' use of this view

4.2.2 Component Diagram

The component diagram is shown in Figure 4.5.

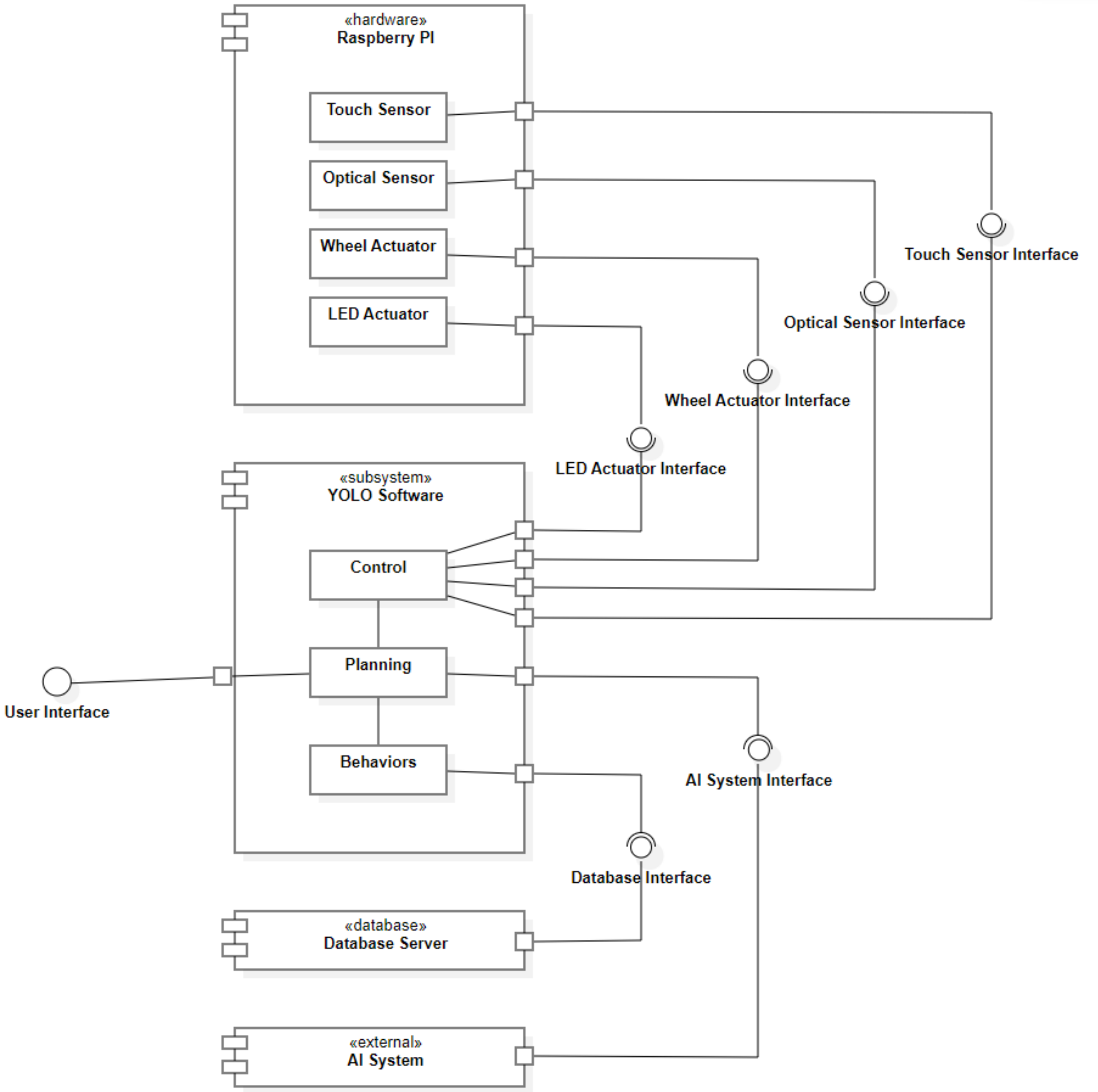

Figure 4.5: *Component Diagram*

4.2.3 Internal Interfaces

The class diagram for internal interfaces is shown in Figure 4.6.

4.2.4 Interaction Patterns

Use-case name	Determine Movement Shape
Actors	AI System
Description	Movement data collected by the optical sensor are sent to the AI System. The shape of movement is predicted using machine learning algorithms.
Data	Movement information, movement shape
Preconditions	The user must move the robot and the optical sensor must collect the movement information.
Stimulus	AI System predicts the shape of movement.
Basic flow	Step 1: The user touches the robot. Step 2: The user applies force on the robot to change its position. Step 3: The optical sensor senses changes in the position and detects the speed and the direction of movement. Step 4: Collected movement information is sent to the AI system by the agent. Step 5: AI System runs machine learning algorithms to predict the movement shape. Step 6: Predicted shape is returned to the agent to be used by the planning module to determine the behavior of the robot.
Alternative flow	-
Exception flow	Step 5: If the shape is not recognized by the AI System, the exception is logged and the default shape is returned.

Postconditions	The planning module determines the robot's behavior depending on the movement shape and the current profile of the robot.

Table 4.3: *Determine Movement Shape Function*

The sequence diagram for Determine Movement Shape use-case is shown in Figure 4.7.

Use-case name	Perform Slow Mirror Behavior
Actors	AI System
Description	The robot predicts the shape of movement the user makes and imitates it if the current storytelling phase is rising action or falling action. This behavior stimulates convergent thinking.
Data	Movement shape, speed and direction
Preconditions	The robot must be in the rising action phase or the falling action phase of the storytelling arc, and the touch duration must be greater than the shape threshold.
Stimulus	The user stops touching the robot after moving it for a longer period of time than the shape threshold.

Basic flow	Step 1: The user touches the robot. Step 2: The user moves the robot. Step 3: The user stops touching the robot. Step 4: If the touch duration is greater than the shape threshold, AI System predicts the movement shape. Step 5: If the robot is in the rising action or falling action stage of the storytelling arc, then the mirroring behavior is performed. Step 6: The agent activates actuators via control module to imitate the shape.
Alternative flow #1	Step 4: If the touch duration is smaller than the shape threshold, Idle Behavior is performed.
Alternative flow #2	Step 5: If the robot is in the climax stage of the storytelling arc, then the contrasting behavior is performed.
Exception flow	Step 4: If the shape is not recognized by the AI System, the exception is logged and the default shape is returned.
Postconditions	Idle behavior is performed.

Table 4.4: *Perform Slow Mirror Behavior Function*

The sequence diagram for Perform Slow Mirror Behavior use-case is shown in Figure 4.8.

Use-case name	Perform Fast Contrast Behavior
Actors	AI System
Description	The robot predicts the shape of movement the user makes and imitates it if the current storytelling phase is climax. This behavior stimulates divergent thinking.
Data	Movement shape, speed and direction

Preconditions	The robot must be in the climax phase of the storytelling arc, and the touch duration must be greater than the shape threshold.
Stimulus	The user stops touching the robot after touching it for a longer time than the shape threshold.
Basic flow	Step 1: The user touches the robot. Step 2: The user moves the robot. Step 3: The user stops touching the robot. Step 4: If the touch duration is greater than the shape threshold, AI System predicts the movement shape. Step 5: If the robot is in the climax stage of the storytelling arc, then the contrasting behavior is performed. Step 6: The agent activates actuators via control module to imitate the shape.
Alternative flow #1	Step 4: If the touch duration is smaller than the shape threshold, Idle Behavior is performed.
Alternative flow #2	Step 5: If the robot is in the rising action or falling action stage of the storytelling arc, then the mirroring behavior is performed.
Exception flow	Step 4: If the shape is not recognized by the AI System, the exception is logged and the default shape is returned.
Postconditions	Idle behavior is performed.

Table 4.5: *Perform Fast Contrast Behavior Function*

The sequence diagram for Perform Fast Contrast Behavior use-case is shown in Figure 4.9.

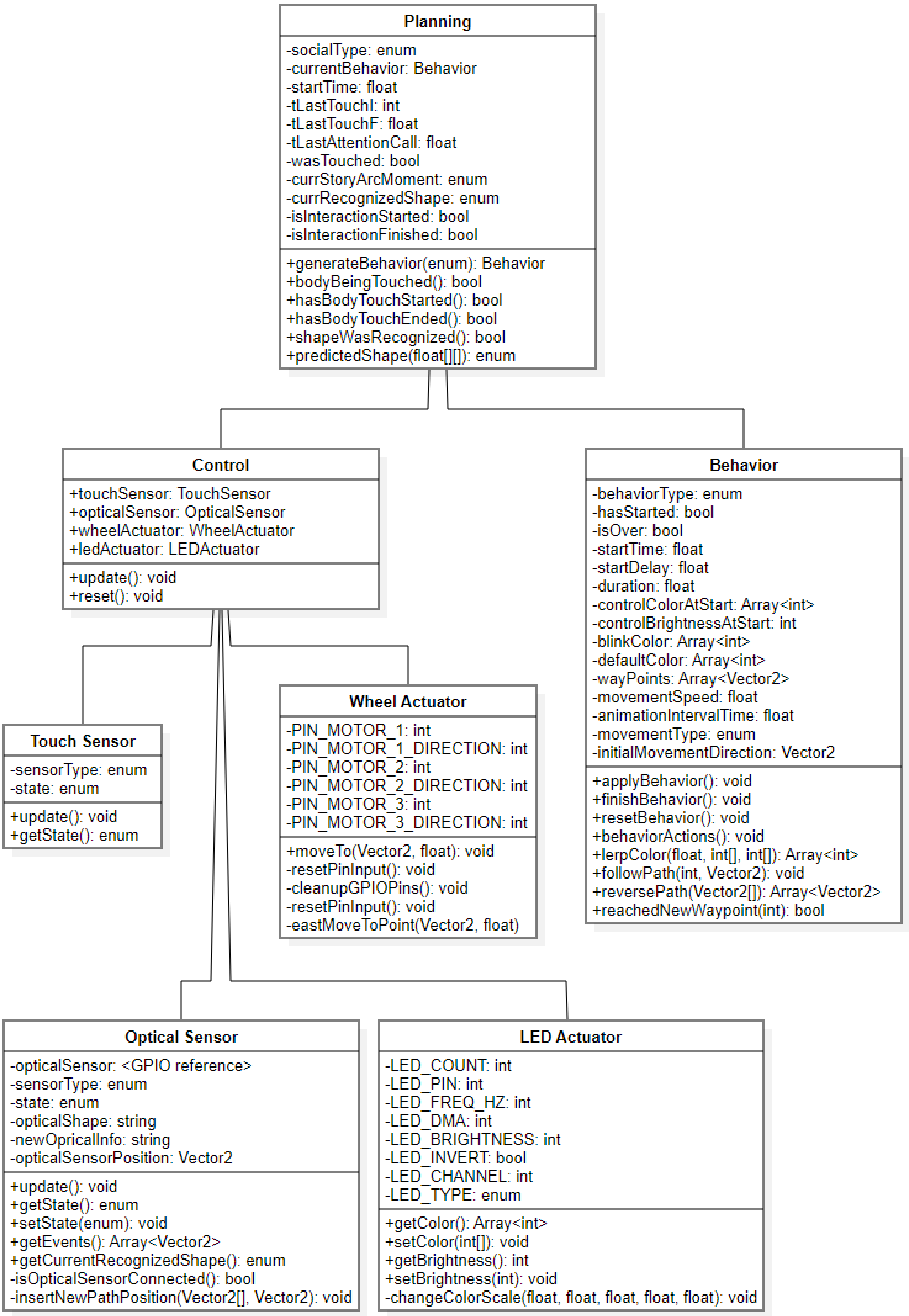

Figure 4.6: *Internal Interfaces Class Diagram*

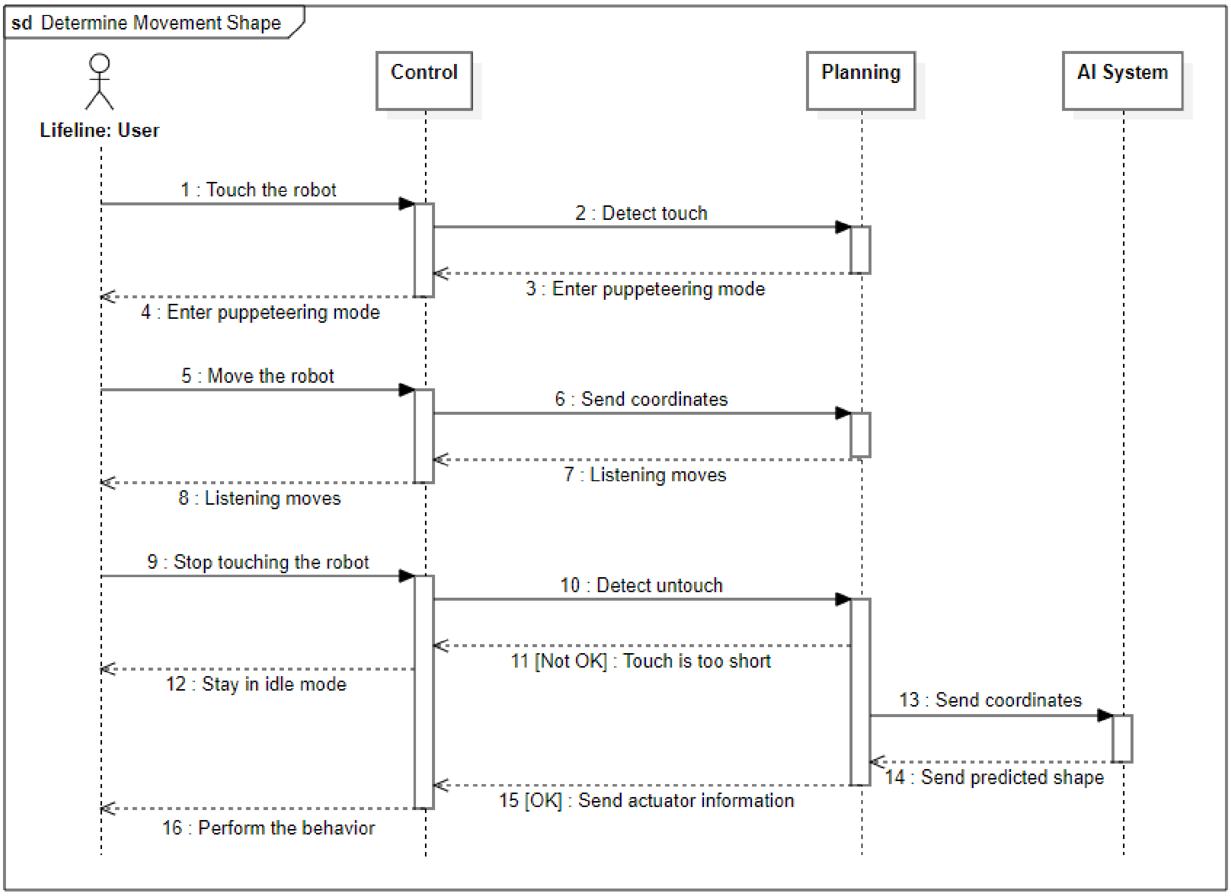

Figure 4.7: *Determine Movement Shape Sequence Diagram*

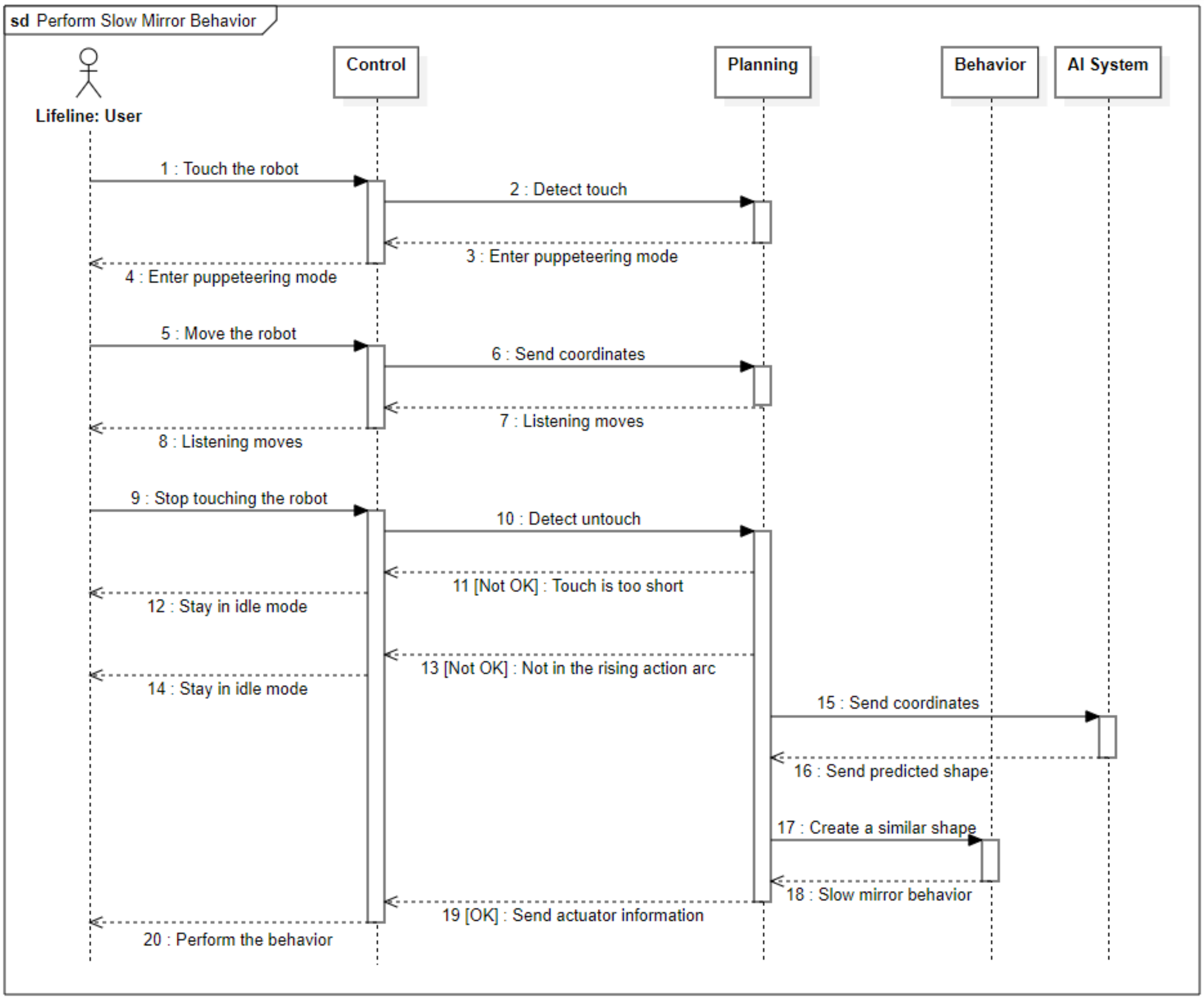

Figure 4.8: *Perform Slow Mirror Behavior Sequence Diagram*

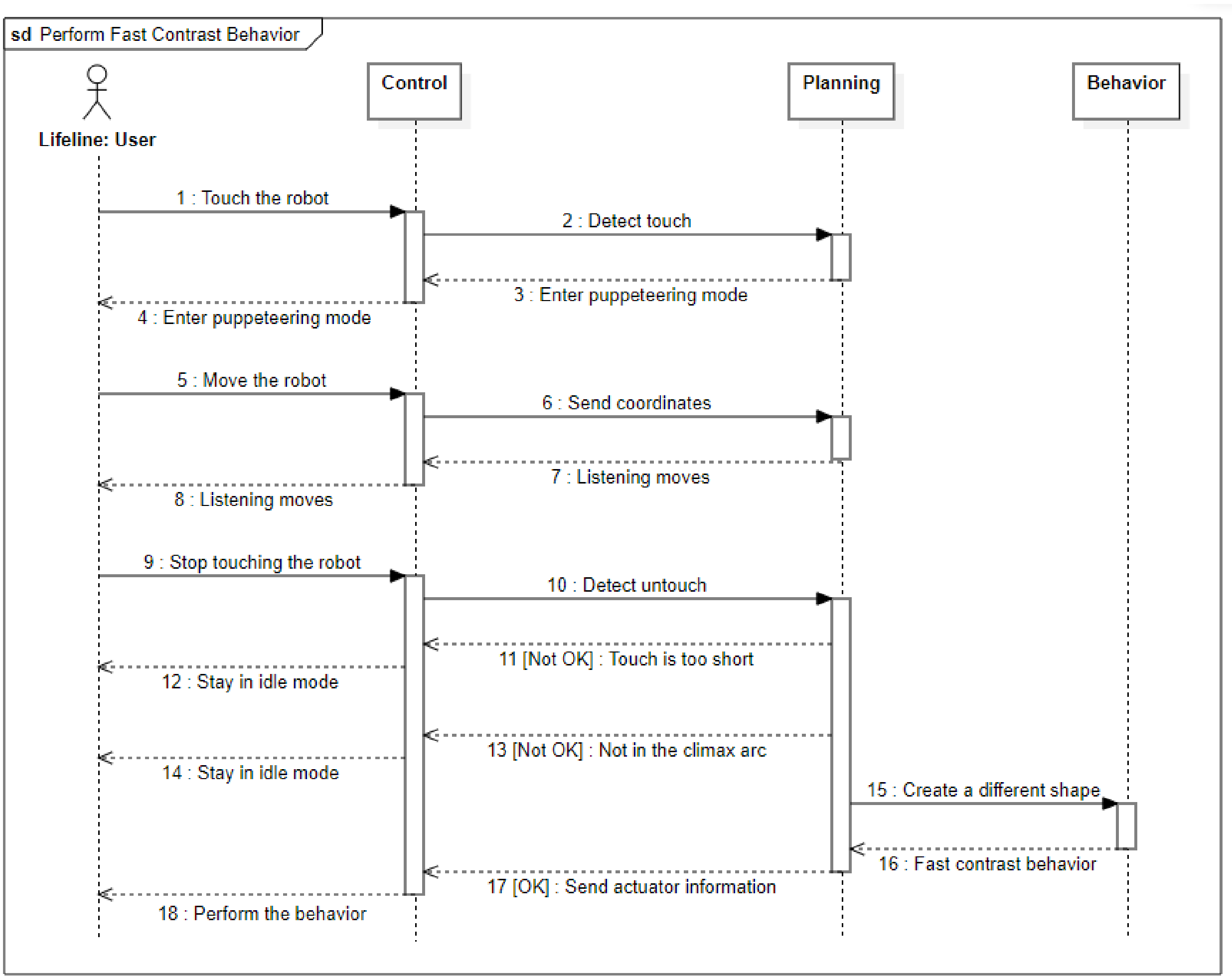

Figure 4.9: *Perform Fast Contrast Behavior Sequence Diagram*

4.3 Information View

4.3.1 Stakeholders' use of this view

4.3.2 Database Class Diagram

The database class diagram is shown in Figure 4.10.

4.3.3 Operations on Data

Operation	Description
GetHelloBehavior	Create: - Read: GeneralProfile Update: - Delete: -
GetGoodbyeBehavior	Create: - Read: GeneralProfile Update: - Delete: -
GetPuppeteerBehavior	Create: - Read: GeneralProfile Update: - Delete: -
GetIdleBehavior	Create: - Read: GeneralProfile Update: - Delete: -
GetMinimumTouchTime	Create: - Read: SocialProfile Update: - Delete: -

GetAttentionCallBehavior	Create: - Read: SocialProfile Update: - Delete: -
GetRisingActionTime	Create: - Read: CreativityProfile Update: - Delete: -
GetClimaxTime	Create: - Read: CreativityProfile Update: - Delete: -
GetFallingActionTime	Create: - Read: CreativityProfile Update: - Delete: -
CreateComposedBehavior	Create: ComposedBehavior Read: SimpleBehavior Update: - Delete: -
GetBehaviorList	Create: - Read: ComposedBehavior Update: - Delete: -
GetStartTime	Create: - Read: SimpleBehavior Update: - Delete: -

GetDuration	Create: - Read: SimpleBehavior Update: - Delete: -
GetBlinkColor	Create: - Read: BlinkBehavior Update: - Delete: -
SetBlinkColor	Create: - Read: - Update: BlinkBehavior Delete: -
GetBlinkBrightness	Create: - Read: BlinkBehavior Update: - Delete: -
SetBlinkBrightness	Create: - Read: - Update: BlinkBehavior Delete: -
GetMovementType	Create: - Read: MoveBehavior Update: - Delete: -
SetInitialMovementDirection	Create: - Read: - Update: MoveBehavior Delete: -

GetMovementSpeed	Create: - Read: MoveBehavior Update: - Delete: -
SetMovementSpeed	Create: - Read: - Update: MoveBehavior Delete: -

Table 4.6: *CRUD Operations*

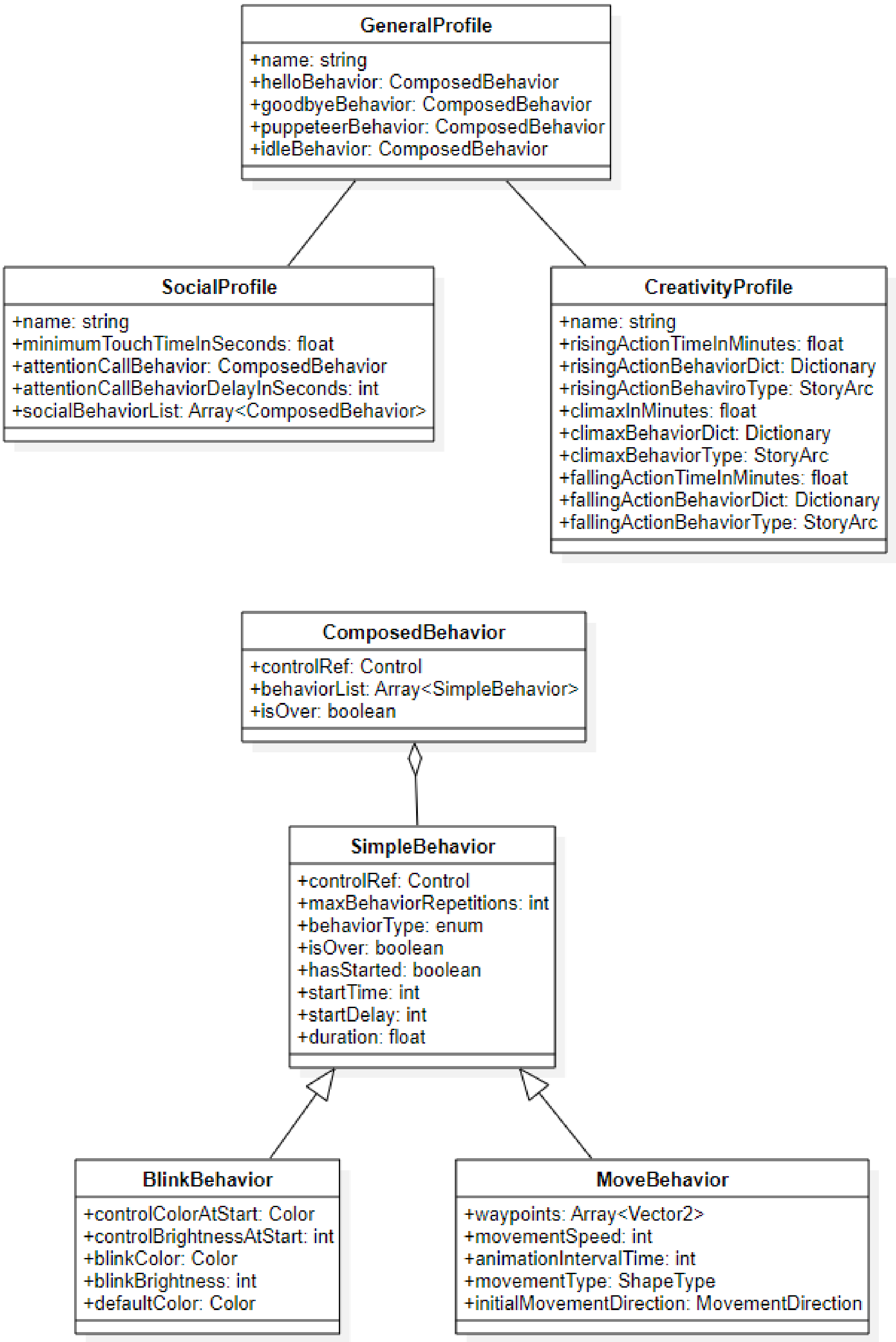

Figure 4.10: *Database Class Diagram*

4.4 Deployment View

4.4.1 Stakeholders' use of this view

4.4.2 Deployment Diagram

The deployment diagram is shown in Figure 4.11.

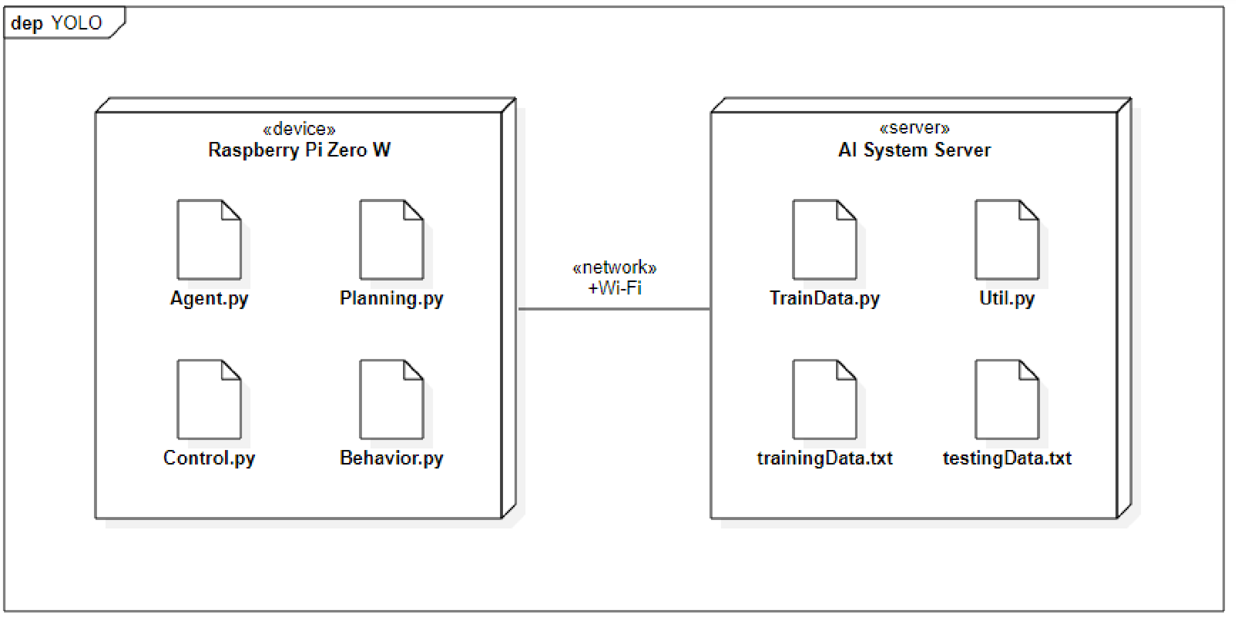

Figure 4.11: *Deployment Diagram*

4.5 Design Rationale