

TERM PROJECT PART 2: Design with UML

The purpose of this exercise is to practice modeling with UML. The assignment builds on last week’s scenario, utilizing a system you are familiar with so that you can concentrate on UML



February 8, 2024

Boston university

Mahim Choudhury

Table of Contents

[4.1 Selected Use Case, Revised 2](#_Toc158386283)

[A 4.1 (ChatGPT regarding Selected Use Case – Revised) 3](#_Toc158386285)

[4.2 Sequence Diagram 5](#_Toc158386286)

[Figure 1: Sequence Diagram 5](#_Toc158386288)

[4.3 Class Model 6](#_Toc158386289)

[Figure 2: Class Model Diagram 6](#_Toc158386290)

[4.4 Documenting Classes and Relationships 7](#_Toc158386291)

[4.4.1 First Business Class Selected: Your class name replaces this. 7](#_Toc158386292)

[4.4.2 Second Business Class Selected: Your class name replaces this. 7](#_Toc158386293)

[4.4.3 Third Business Class Selected: Your class name replaces this. 7](#_Toc158386294)

[4.4.4 Non-Business Class Selected: Your class name replaces this. 7](#_Toc158386295)

[Appendix 8](#_Toc158386296)

[References 9](#_Toc158386297)

[Evaluation 10](#_Toc158386298)

[Hints When you submit, please replace these with a copy of your Assignment 3 (Term Project Part 1). 11](#_Toc158386299)

[1.1 Overall Assignment Notes 11](#_Toc158386300)

[1.2 Use Cases 11](#_Toc158386301)

[1.3 Class and Sequence diagrams 11](#_Toc158386302)

[1.4 Specific to Class Model 12](#_Toc158386303)

[1.5 Specific to Sequence Diagram 12](#_Toc158386304)

MET CS682 ASSIGNMENT 4

TERM PROJECT PART 2: Design with UML

*Mahim Choudhury*

The purpose of this exercise is to give you practice modeling with UML. The assignment builds on last week’s scenario, utilizing a system you are familiar with so that you can concentrate on UML. USE THIS TEMPLATE: Please respond by using this Word template, including the text boxes, leaving the headings and the gray text unchanged—but exclude the *hints* section (at the end) from your solution.

1. OBSERVE LIMITS; 2. USE AI GENERATION: as before but only on Question 4.1. 3. OUTSIDE RESEARCH; 4. NAME YOUR FILE; 5. EVALUATION; 6. HINTS: as before

You are to specify a UML design for *EmbeddedAI*. The characteristics listed in Assignment 3 (Part 1 of the term project) apply. *EmbeddedAI* has a potentially large scope, but your solution should focus on aspects that you select. These are assumed to include your requirements for Assignment 3 (Part 1 of the term project), except where you indicate otherwise.

# Selected Use Case, Revised

## Select Improve and update the one of the two use cases from Term Project Part 1. This will be used for the rest of the assignment, specifically the Sequence Diagram in section 4.

| Step # | Actor | System |
| --- | --- | --- |
| 1 | Pet Owner activates pet monitoring mode | System initiates AI-enabled pet monitoring functionalities, including smart cameras, motion sensors, and environmental sensors, to track pet movements and behaviors. |
| 2 | Pet Owner receives activity alert | System utilizes AI algorithms to detect pet movements, sounds, or irregular behaviors, and sends real-time alerts and notifications to the pet owner's mobile device. |
| 3 | Pet Owner views live monitoring feed | System provides access to live video streams and environmental data through the mobile app, allowing the pet owner to observe their pets' activities and surroundings remotely. |
| 4 | Pet Owner interacts with pets remotely | System enables the pet owner to engage with their pets through interactive features, such as remotely operated toys, treat dispensers, or two-way audio communication channels. |
| 5 | Pet Owner ensures pet safety | System allows the pet owner to monitor environmental conditions, such as temperature, humidity, and air quality, and take necessary actions to ensure the safety and comfort of their pets. |
| 6 | Pet Owner receives guidance on mitigating risks or contacting emergency services (Alt 6) | If the system detects unusual behavior or potential hazards, it alerts the pet owner and provides guidance on mitigating risks or contacting emergency services. |
| 8 | System activates pre-configured safety protocols (Alt 8) | If the pet owner is unavailable to respond to alerts, the system can activate pre-configured safety protocols, such as contacting designated emergency contacts or veterinary services. |

The AI response was not useful at all as it failed to provide any new useful extension to my current use case. After carefully observing the use case myself, I am convinced it has enough functionalities and features to make out a complex system design. It has many classes and objects with dependencies which would be beneficial when designing the sequence and class diagram.

## A 4.1 (ChatGPT regarding Selected Use Case – Revised)

Show your most relevant prompt to chatGPT and the response which was most relevant to your solution.

Prompt: Please refine, improve, and extend the pet monitoring use case I defined in part 1 to if necessary.

Response: To extend and improve the pet activity monitoring use case, we can introduce additional functionalities and interactions between the Pet Owner and the Smart Home System. Here's the revised and extended version of the use case:

Use case Name: Pet Activity Monitoring Actor: Pet Owner Description: The integrated EmbeddedAI system empowers pet owners to monitor their pets remotely, ensuring their safety and well-being even when the owners are away from home. Pre-condition: The system is online and connected to IoT devices.

| Step # | Actor | System |
| --- | --- | --- |
| 1 | Pet Owner activates pet monitoring mode | System initiates AI-enabled pet monitoring functionalities, including smart cameras, motion sensors, and environmental sensors, to track pet movements and behaviors. |
| 2 | Pet Owner receives activity alert | System utilizes AI algorithms to detect pet movements, sounds, or irregular behaviors, and sends real-time alerts and notifications to the pet owner's mobile device. |
| 3 | Pet Owner views live monitoring feed | System provides access to live video streams and environmental data through the mobile app, allowing the pet owner to observe their pets' activities and surroundings remotely. |
| 4 | Pet Owner interacts with pets remotely | System enables the pet owner to engage with their pets through interactive features, such as remotely operated toys, treat dispensers, or two-way audio communication channels. |
| 5 | Pet Owner ensures pet safety | System allows the pet owner to monitor environmental conditions, such as temperature, humidity, and air quality, and take necessary actions to ensure the safety and comfort of their pets. |
| 6 | Pet Owner receives guidance on mitigating risks or contacting emergency services (Alt 6) | If the system detects unusual behavior or potential hazards, it alerts the pet owner and provides guidance on mitigating risks or contacting emergency services. |
| 7 | System activates pre-configured safety protocols (Alt 7) | If the pet owner is unavailable to respond to alerts, the system can activate pre-configured safety protocols, such as contacting designated emergency contacts or veterinary services. |

# Sequence Diagram

## Provide a detailed sequence diagram for one of your use cases above. Your solution should have at least five objects. Make sure to represent these correctly. For example, if you were to create an instance of an *VideoChat* class (i.e., to model a new video chat instance) it may be called *newVideo:Video* or just *:Video* if the instance name is irrelevant.

A diagram with green and blue squares

Description automatically generated

### Figure 1: Sequence Diagram

Sequence Diagram depicts the interaction between an actor and the system for a use-case scenario [1]. The sequence diagram above includes everything, but the live monitoring feature described in the use case (to avoid diagram complexity). It starts with the pet owner clicking the pet-monitoring tab in the smart home system’s home screen (non-entity class). It enables the system to activate sensors and start monitoring. The detection class is an instance of the sensors class. The AI will help the system to detect pet movements continuously. If it detects unusual activities, it automatically goes to the Activity function and enables the alert which triggers a notification to be sent to the owner. The alt course to send the notification to emergency contact as mentioned in the use case is again avoided in the sequence diagram to reduce complexity. The user can click interaction and safety options from the pet monitoring display and involve them in the respective features as depicted in the diagram. Notice the loops, it indicates the sensor detection and interaction is a continuous action unless the user breaks out of it. Another key aspect to keep in mind here is that I made all the returns to the pet monitoring screen assuming the user will use it for the interaction except the notification.

# Class Model

Provide a class model for the system, maintaining the system scope you determined. Your solution should have roughly 12 classes. (When complete, a real design typically may contain hundreds of classes, but your submission must focus on the scope you have chosen.) Your class model should show classes and their relationships. To add clarity to your diagram, provide key attributes and methods. (You do not have to list every attribute and method, just the most important ones.) Label clearly.

A diagram of a computer

Description automatically generated

### Figure 2: Class Model Diagram

The class diagram illustrates the pet monitoring aspect of the system, where EmbeddedAI enhances functionality. It begins with a non-entity stereotype class named Design, conforming to the UML User Guide's definition of stereotypes as labels for similar classes within a specific domain, essentially creating a metatype [2]. The model showcases aggregation, composition, and inheritance relationships among its classes. The Detection class serves as the integration point for EmbeddedAI to analyze data. Further elaboration on key classes and their relationships is provided in the subsequent section.

# Documenting Classes and Relationships

Using the format below, select three most important nontrivial business[[1]](#footnote-1) classes and one non-business design class from the class model and the format below, explain its importance for your design, and its relationships with other classes. Select classes which are complex enough to require explanation.

## 4.4.1 First Business Class Selected: Your class name replaces this.

**Importance for the Design**: Your response replaces this.

**Relationship with other classes**: Your response replaces this.

## 4.4.2 Second Business Class Selected: Your class name replaces this.

**Importance for the Design**: Your response replaces this.

**Relationship with other classes**: Your response replaces this.

## 4.4.3 Third Business Class Selected: Your class name replaces this.

**Importance for the Design**: Your response replaces this.

**Relationship with other classes**: Your response replaces this.

## 4.4.4 Non-Business Class Selected: Your class name replaces this.

**Importance for the Design**: Your response replaces this.

**Relationship with other classes**: Your response replaces this.

# Appendix

Since Term Project Part 2 builds on Term Project Part 1, Provide requirements you have outlined last week in Term Project Part 1. Not everything needs to be included, only the parts you feel are helpful in supporting the UML design. This section is not graded.

Scope:

Scenario:

Imagine a Smart Home Automation System designed to manage various aspects of home living, such as lighting, temperature control, security, and appliance automation. This system utilizes IoT devices and sensors to enable remote monitoring and control via mobile apps or voice commands.

Scope: The scope entails integrating EmbeddedAI into an existing Smart Home Automation System to enhance user experience, efficiency, and security. This integration will leverage generative AI like ChatGPT and IoT elements to provide personalized assistance, intelligent automation, and proactive home management capabilities. Key focus areas include personalized home assistance, natural language interaction, predictive home automation, and enhanced security and safety measures.

The project will involve analyzing user behavior, preferences, and environmental data collected by IoT sensors to offer tailored recommendations and automate routine tasks. It will enable natural language interactions with EmbeddedAI, allowing users to communicate with their smart home system using voice commands or text input. Predictive capabilities will anticipate user needs and potential issues, while enhanced security measures will detect anomalies and trigger automated responses to ensure the safety of the home environment.

Initial Use Case:

|  |  |  |
| --- | --- | --- |
| **Use case Name** | Pet activity monitoring | |
| **Actor:** | Pet owner | |
| **Description:** | The integrated EmbeddedAI system can aid the smart home system to empower the pet owners with the ability to leave their pets home without worry. AI-enabled monitoring system enhances pet safety and allows pet owners to stay connected with their pets, even when they are not physically present at home. | |
| **Pre-condition:** | The system is online and connected to IoT devices. | |
| **Step #** | **Actor** | **System** |
| **1** | Pet Owner activates pet monitoring mode | System initiates AI-enabled pet monitoring functionalities, including smart cameras, motion sensors, and environmental sensors, to track pet movements and behaviors. |
| **2** | Pet Owner receives activity alert | System utilizes AI algorithms to detect pet movements, sounds, or irregular behaviors, and sends real-time alerts and notifications to the pet owner's mobile device. |
| **3** | Pet Owner views live monitoring feed | System provides access to live video streams and environmental data through the mobile app, allowing the pet owner to observe their pets' activities and surroundings remotely. |
| **4** | Pet Owner interacts with pets remotely | System enables the pet owner to engage with their pets through interactive features, such as remotely operated toys, treat dispensers, or two-way audio communication channels. |
| **5** | Pet Owner ensures pet safety | System allows the pet owner to monitor environmental conditions, such as temperature, humidity, and air quality, and take necessary actions to ensure the safety and comfort of their pets. |
| **Alternate Courses:** | [System Alt 6] If the system detects unusual behavior or potential hazards, it alerts the pet owner and provides guidance on mitigating risks or contacting emergency services.  [System Alt 7] If the pet owner is unavailable to respond to alerts, the system can activate pre-configured safety protocols, such as contacting designated emergency contacts or veterinary services. |  |

# References

Show that you used a wide variety of resources by listing them below and clearly indicating in the body above where you used each one (inserting [1], [2] etc.). We suggest using APA format, but other formats are fine as long as your paper clearly distinguishes your work from work of others, being mindful of plagiarism rules.

[1] Whitten, B. (2007). *Systems analysis & design methods*. (7th ed.). New York, NY: McGraw-Hill Irwin.

[2] Booch G, Rumbaugh, J, Jacobson I (2014). The Unified Modeling Language User Guide. Upper Saddle River, NJ: Addison-Wesley.

# Evaluation

A picture containing text, screenshot, number, parallel

Description automatically generated

**Please do not include Hints section in your solution.**

# Hints When you submit, please replace these with a copy of your Assignment 3 (Term Project Part 1).

## Overall Assignment Notes

* You may use Draw.io, Visio, LucidChart, or another design tool of your choice (please check with your facilitator in advance if the latter).
* In order to make your design clear to your reader, accompany each requested section with a brief description that explains your thinking and the choices that you made.
  + This is not required for the Use Case (section 1) but is required for all other sections.
* As usual, the notes are a primary source for explanations and examples; we also encourage you to do outside reading and research to gain additional perspective.
* Apply the usual editing process. It is tempting to repeat important points but this often weakens your work and feels to the reader like padding.
* Explain use of references in the appendix. Don't list sites that are not specifically referred to in the body of your paper.
* It is important that the entire solution is consistent. You will want to iteratively review all of the four sections to make sure that together they are consistent as a single solution.

## Use Cases

* Review Assignment 3,and select the use case that has complexity – we model what is complex.

**Clarity**:

* Look to refine scope and clarify the steps (i.e. if someone else read it - would they understand this clearly?)

**Technical soundness:**

* The use case needs to show appropriate sequence (actor/system), branching needs to be minimal within alternate steps. Clear understanding in difference between constraints and pre-conditions. The use case itself should follow one path as best as possible.

**Relevance:**

* Are the use cases relevant and consistent to your class model and sequence diagram?

**Thoroughness and coverage:**

* Did you link to your requirements/user story (place in appendix) did you think through preconditions, alternate steps, and constraints.

## Class and Sequence diagrams

* Note that some students tend to identify classes by first writing sequence diagrams while others prefer to first identify class models, thus you are free to try different approaches.

**Relevance:**

* Are the classes relevant and consistent to your use case, class model, and sequence diagram? Is the scope of your design relevant to the solution? For example, choosing a use case, classes and methods that are secondary (for example, authentication).
* Focus on more business classes vs. non-business classes

**Technical Soundness:**

* A good place to start in identifying classes is to review Module 4 “Introduction to UML.” You will need to understand how to identify business objects and their relationships. Note how examples in the module start with the use case; this is where going back to the use case you developed is important. Take a use case and break it down into verbs and nouns.
* Solution needs to show that you understand the difference between business and non-business classes and that you have identified and organized these correctly. (an example is <<design>> for a GUI-related class).
* Review checklist for Sequence and Class diagrams—these will identify areas where many miss parts on technical soundness, clarity, and possibly thoroughness and coverage

**Clarity:**

* Explain your diagrams, especially parts that are not self-evident. For example, why is a specific relationship aggregation or composition? A good way to do this is to provide a short-bulleted list under the diagram itself.
* Are diagrams clear to read (i.e. no overlapping lines, no non-polished designs). In sequence diagrams, return messages should only be added as needed, for clarity.
* Make sure that you have a consistent design throughout the solution. Are all the classes and messages in the sequence diagram represented in the class diagram? Are these explained in your classes overview. Methods in the sequence diagram should be in the class diagram? All the class names should be consistent.
* Make sure that your sequence diagram is consistent between the use case and the classes which you have identified in previous sections.
* Try to balance clarity and thoroughness, as an example in sequence diagrams return messages may clutter your design.

**Thoroughness and coverage:**

* Flush out your diagrams with details where you can. For example, perhaps in a class Search, we have methods for detectingSearchCriteria() and performSearch(). All methods in the Sequence diagram should also be listed in the method section of the class.

## Specific to Class Model

**Technical Soundness:**

* Please see Module 4 primary readings as the textbook provides much useful detail around object-oriented design and class modeling- You will want to understand concepts such as—but not limited to—inheritance, generalization, specialization, aggregation, composition, and association.
* Show understanding between different types of relationships, for example, aggregation vs. inheritance. In situations where it is a design choice (i.e. aggregation vs. composition)
* Consider the use of dependency as a relationship between business and non-business classes. Dependency between business classes should be explained.
* Multiplicity should be shown on aggregation and composition and no other type of relationship.

**Thoroughness and coverage:**

* It is helpful to add attributes and methods to your class diagram where appropriate, it will help you think through the classes and create more thoroughness and coverage within your solution.

## Specific to Sequence Diagram

**Technical Soundness:**

* Start by reading the Module 4 “Sequence Diagrams” section, as it has several examples and approaches. Module 6 has additional examples of detailed sequence diagrams if you need more examples.
* The textbook goes into some detail on sequence diagrams on pages 202-210.
* The function/method belongs in the class that the arrow points to—this should be represented in the class diagram.
* Functions (solid line, method name) and return messages (dashed line, description) correctly labeled. These should only be used for clarification.
* Objects involved in a sequence diagram are all instances of classes. In some situations (i.e., if in a sequence diagram there are multiple instances of the same class shown) it may be appropriate to label each individual instance of the class (e.g., NewCart:Cart and OpenCart:Cart)
* Classes correctly labeled—including stereotypes where appropriate (CamelCase, vs. camelCase, no spaces, etc)

1. also called “domain” or “entity” class [↑](#footnote-ref-1)