National University of Computer and Emerging Sciences

FAST School of Computing **Spring-2022** **Islamabad Campus**

SE-2002: Software Design and Architecture

Serial No:

Final Exam

Total Time: 3 Hour

Total Marks: 100

Monday, 27th June, 2022

_____

Signature of Invigilator

Course Instructors

Dr. Zohaib Iqbal, Dr. Uzair Khan,
Dr. Atif Jilani and Ms. Hira Naveed

_____	_____	_____	_____
Student Name	Roll No	Section	Signature

DO NOT OPEN THE QUESTION BOOK OR START UNTIL INSTRUCTED.

Instructions:

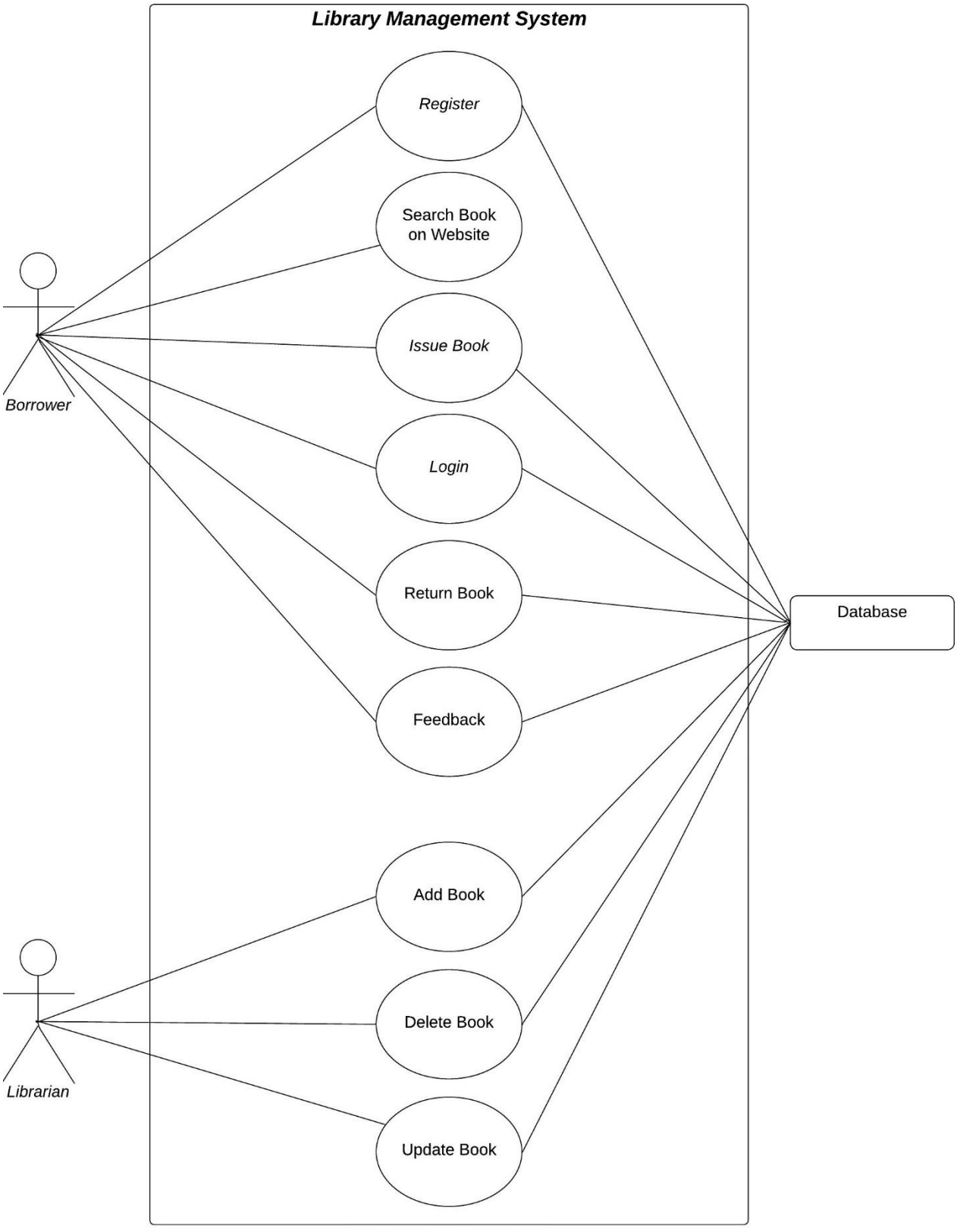
1. Attempt on provided answer sheet. Attempt all questions on the question paper in the given space.
2. After asked to commence the exam, please verify that you have **sixteen (16)** printed pages including this title page. There are total of **three (3)** questions.
3. Use permanent ink pens only. Any part done using soft pencil will not be marked and cannot be claimed for rechecking.
4. Feel free to make any changes in the domain model by drawing additional concepts if needed.

	Q-1	Q-2	Q-3	Total
Marks Obtained				
Total Marks	**25**	**25**	**50**	**100**

Good Luck!

Question 1 [10 + 10 + 5 Marks]

A. Identify the mistakes in the given use case diagram for a library management system.



1. Login is not a use case

2. Database is a part of the system not an actor

3. Add, delete and update books should be one use case i.e., manage books

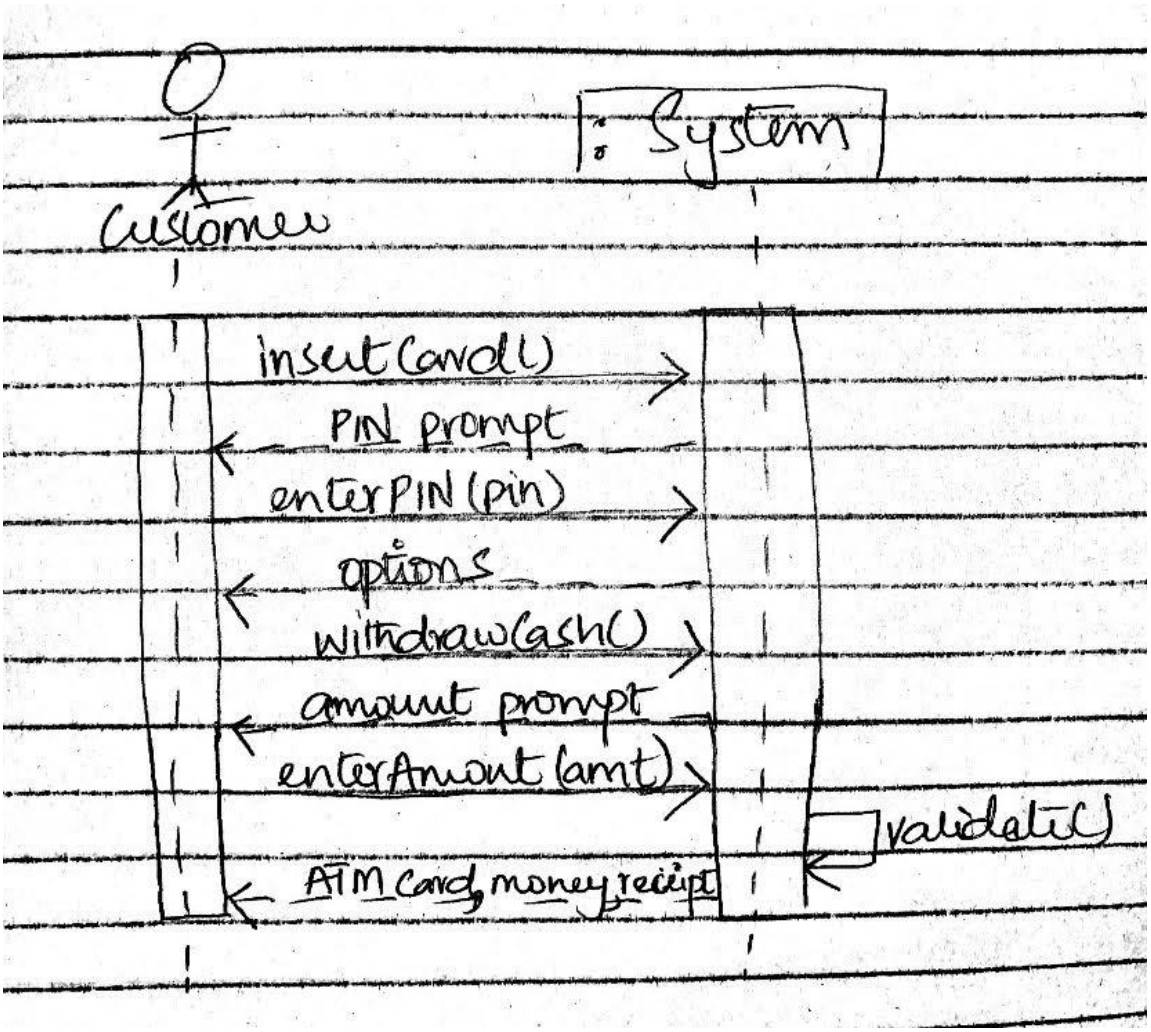
4. Use case must not be implementation specific like search on “WEBSITE”

5. Use case names must always be verbs, feedback is a noun, modify to e.g. give feedback

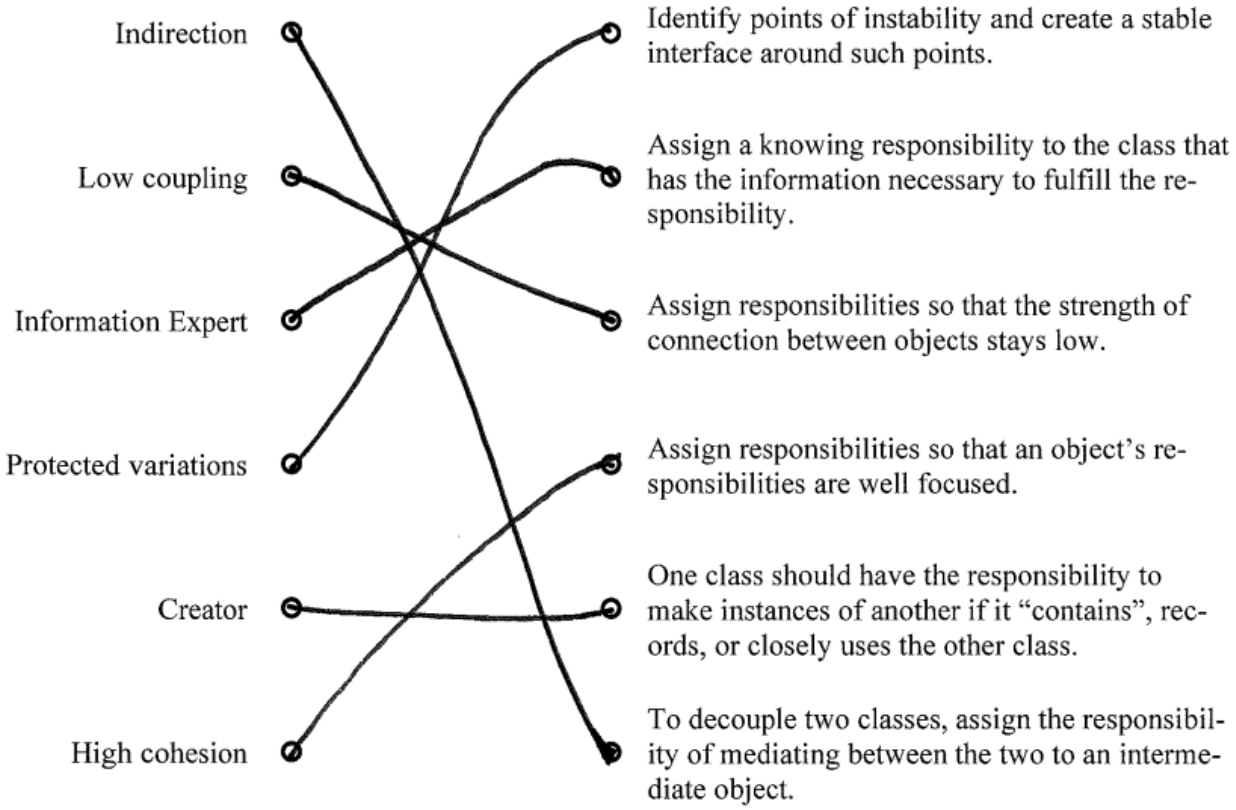
B. Create a System Sequence Diagram (SSD) for the given use case for an ATM system.

Use case Name	Withdraw cash
Actor	Bank Customer
Main Success Scenario	
Actor Actions	**System Response**
1. The bank customer inserts their ATM card 3. The bank customer enters the PIN 5. The bank Customer selects "Withdraw Cash" 7. The bank customer enters an amount	2. The system prompts for a PIN 4. The system displays the different alternatives that are available on this unit 6. The system prompts for an amount 8. The system validates the withdrawal 9. The system records the transaction 10. The ATM card is returned 11. Money is dispensed 12. Receipt is printed
Alternate Scenarios 3a. The bank customer enters incorrect PIN 1. System prompts “Incorrect PIN” 2. System returns ATM card 8a. The amount entered by customer is less than the funds available in the account A. System prompts “Insufficient funds”	

B. System returns ATM card

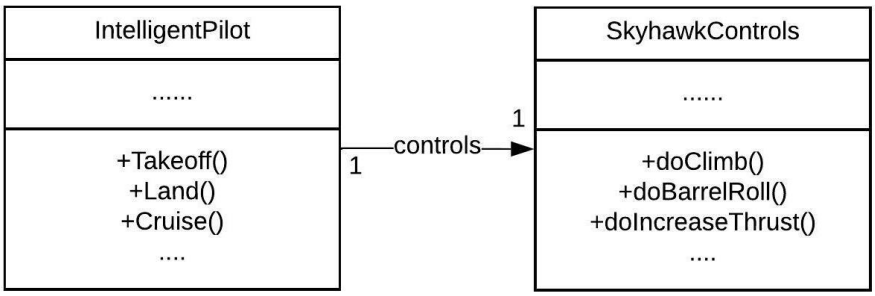


C. For each design pattern below, draw a line from the pattern to the description that best describes it.



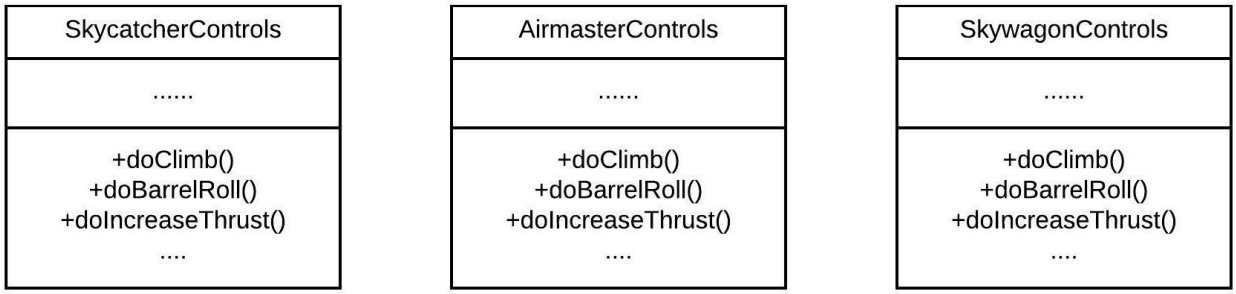
Question 2 [10 +15 Marks]

Imagine you are the creator of an intelligent autopilot system that can actually fly and land real airplanes (wow!). Initially you implemented your system to fly small Cessna Skyhawk airplane with a capacity of 4 passengers. Here is a class diagram of your current software design.



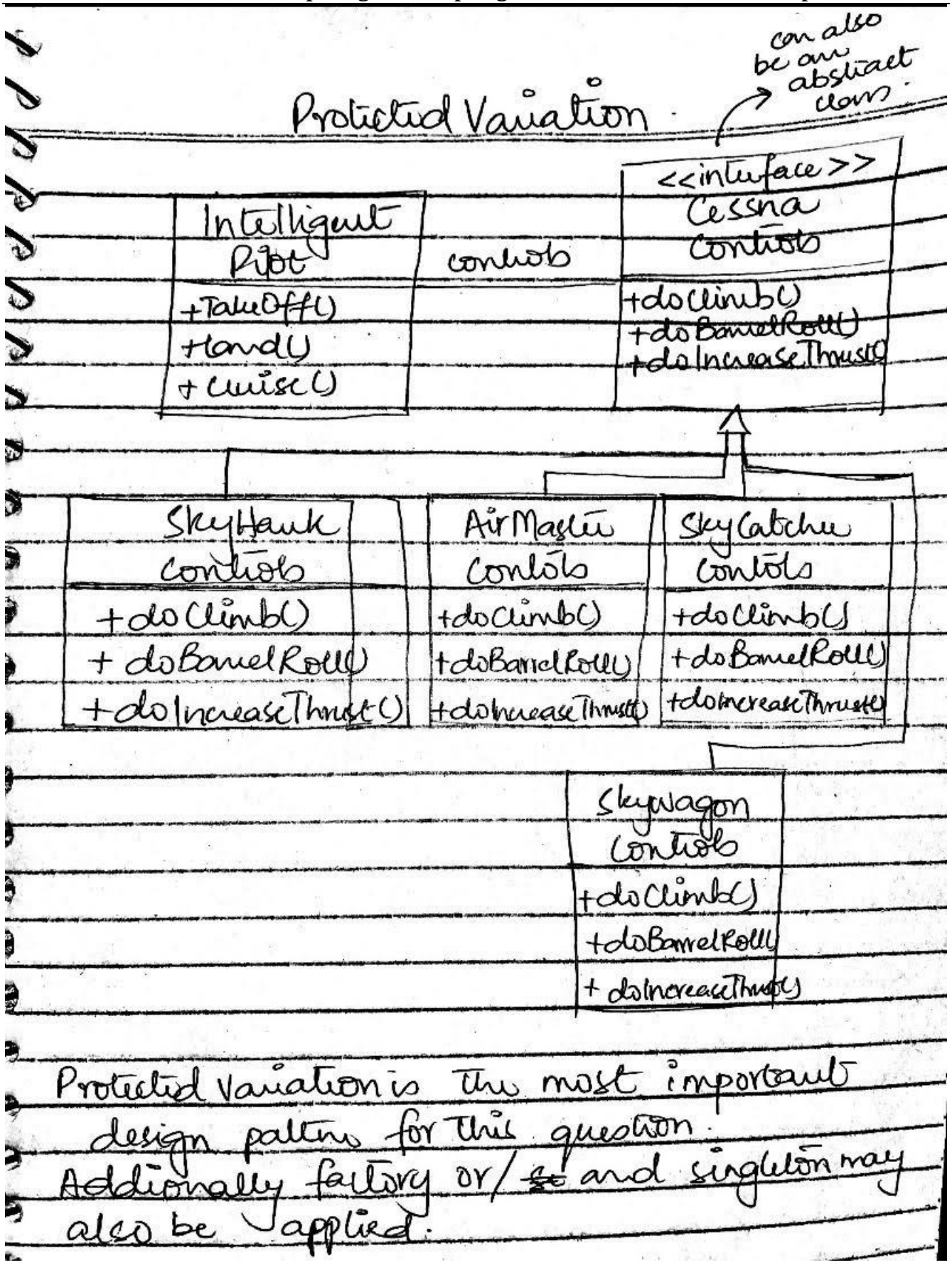
A. Recently there has been a change in requirements and the Cessna Aircraft Company would like to use your intelligent autopilot with other three other airplane models as well, i.e. Skycatcher, Airmaster and

Skywagon. Improve your current design to support this change and allow easy switching between any of the four Cessna airplanes. Keep in mind requirements may change again since this is a *potentially varying functionality*.

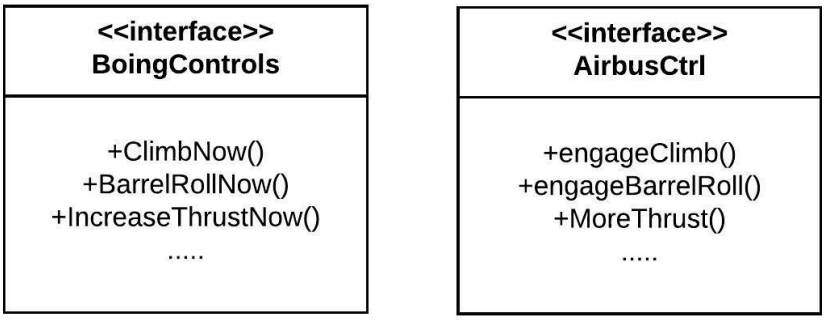


Create a class diagram of your improved design and mention any design patterns you have applied.

Protected variation is necessary for this part

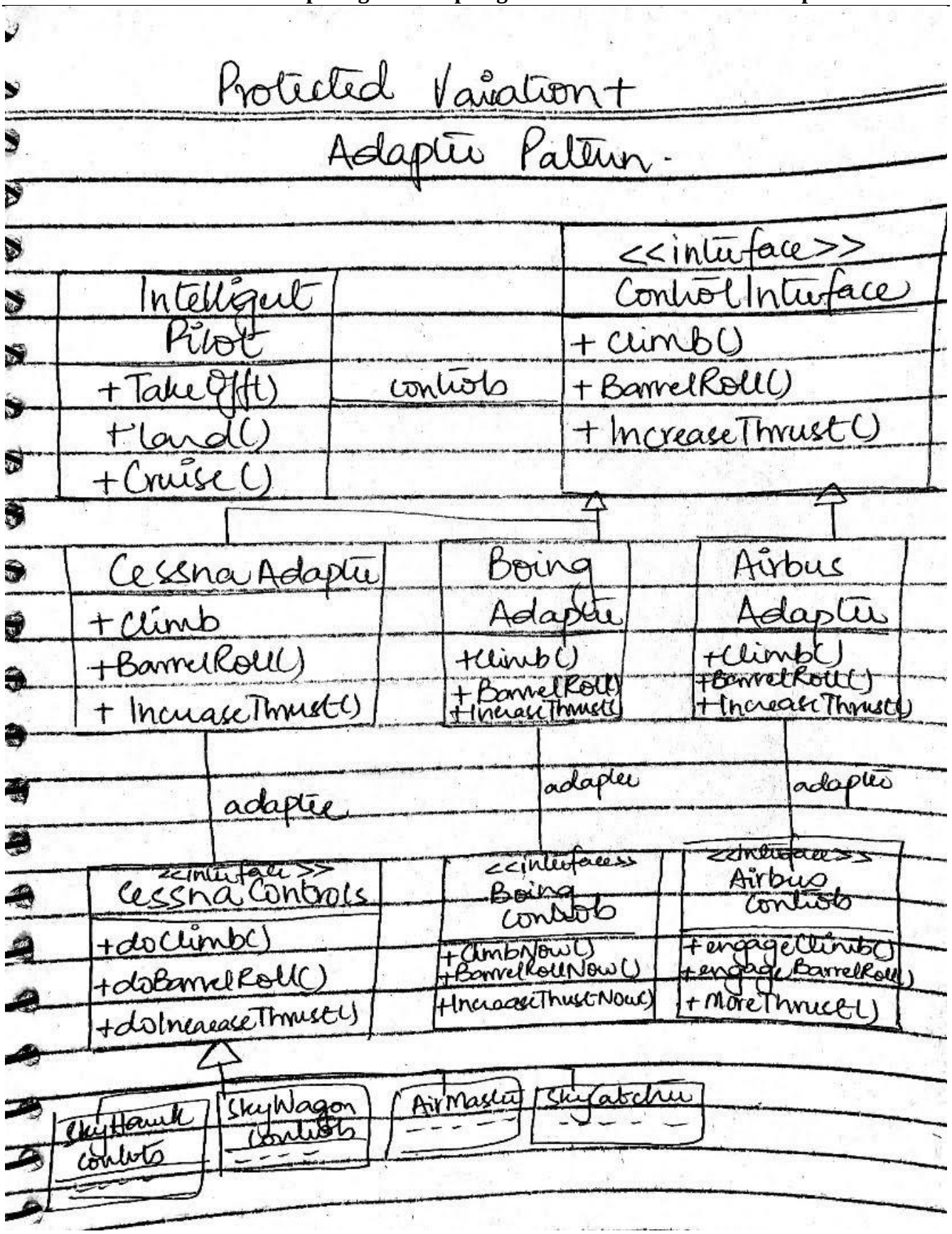

Protected Variation
can also be an abstract class.
<<interface>>
Cessna Controls
+doClimb()
+doBarrelRoll()
+doIncreaseThrust()
IntelligentPilot
controls
+TakeOff()
+Land()
+Cruise()
SkyHawk Controls
+doClimb()
+doBarrelRoll()
+doIncreaseThrust()
AirMaster Controls
+doClimb()
+doBarrelRoll()
+doIncreaseThrust()
SkyCatcher Controls
+doClimb()
+doBarrelRoll()
+doIncreaseThrust()
SkyWagon Controls
+doClimb()
+doBarrelRoll()
+doIncreaseThrust()
Protected Variation is the most important design pattern for this question.
Additionally factory or/ and singleton may also be applied.

B. Since your design is working well with various Cessna airplanes as a next step, you would like your system to support different types of airplanes other than Cessnas. For example, Boing and Airbus both provide their own software control interfaces. Improve your design to enable easy switching between control systems of various types of airplanes.



Create a class diagram of your improved design and mention any design patterns you have applied.

Adapter pattern is necessary for this part


Protected Variation +
Adapter Pattern.
<<interface>>
ControlInterface
+ climb()
+ BarrelRoll()
+ IncreaseThrust()
IntelligentPilot
+ TakeOff()
+ Land()
+ Cruise()
controls
CessnaAdapter
+ Climb
+ BarrelRoll()
+ IncreaseThrust()
BoingAdapter
+ climb()
+ BarrelRoll()
+ IncreaseThrust()
AirbusAdapter
+ climb()
+ BarrelRoll()
+ IncreaseThrust()
adaptee
adaptee
adaptee
<<interface>>
CessnaControls
+ doClimb()
+ doBarrelRoll()
+ doIncreaseThrust()
<<interface>>
BoingControls
+ ClimbNow()
+ BarrelRollNow()
+ IncreaseThrustNow()
<<interface>>
AirbusControls
+ engageClimb()
+ engageBarrelRoll()
+ MoreThrust()
SkyHawk
controls
SkyWagon
controls
AirMaster
SkyLabcher

C. Write code for your improved design in part B. Only implement functions that are necessary, for the rest you can leave function bodies empty.

Question 3 [30 + 20 Marks]

A. You are provided a use case, domain model and system sequence diagram of an asset management system of a vampire hunting company called Stake4Less. Create Sequence Diagrams for all system events shown in the SSD. Make sure your design follows GRASP patterns.

Use case Name	Checkout Weapon
Actor	Hunter
Precondition	Hunter is identified and authenticated
Main Success Scenario	
Actor Actions	**System Response**
1. The hunter starts a new weapon checkout. 3. The hunter enters "Projectile Weapon" with range and ammo rounds 5. The hunter enters number of days 7. The hunter repeats 2-6 until end indicated	2. The system displays the different weapon types available 4. The system prompts for checkout duration 6. The system records the weapon being checked out and presents weapon type, ammo rounds and checkout duration. 8. The system records completed checkout. 9. The system presents confirmation message.

Alternate scenario

3a. The hunter enters "Meele Weapon" with weight

1. Steps 4-9

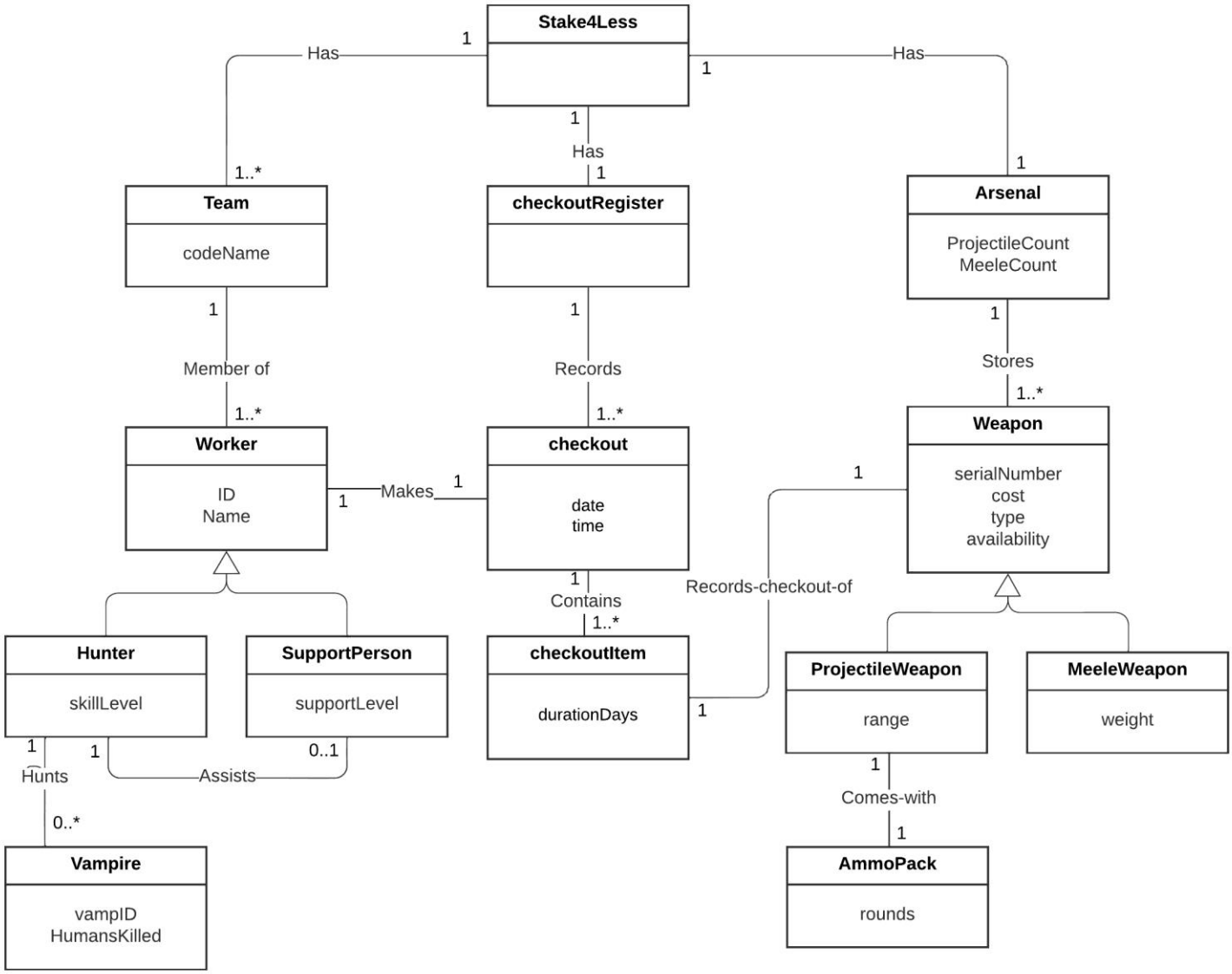
3b. Projectile weapon is not available

1. System prompts "Desired weapon is not available."
2. Repeat 2-9

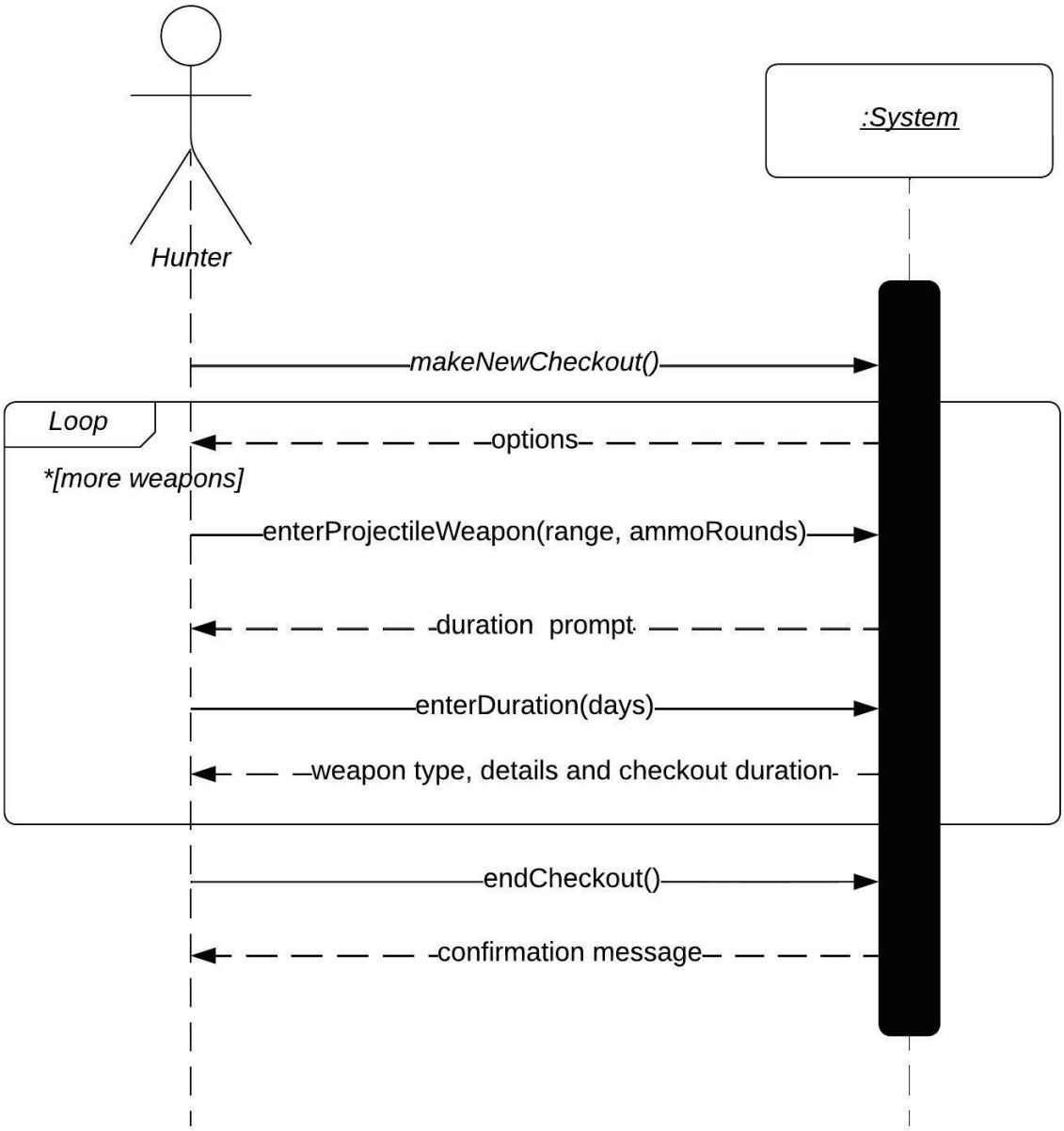
3c. Meele weapon is not available

1. System prompts "Desired weapon is not available."
2. Repeat 2-9

Expanded Use case



Stake4Less Domain Model



Checkout Weapon SSD

B. Convert your sequence diagrams in part A to code. Make sure your code is consistent with your design.

Rough Work