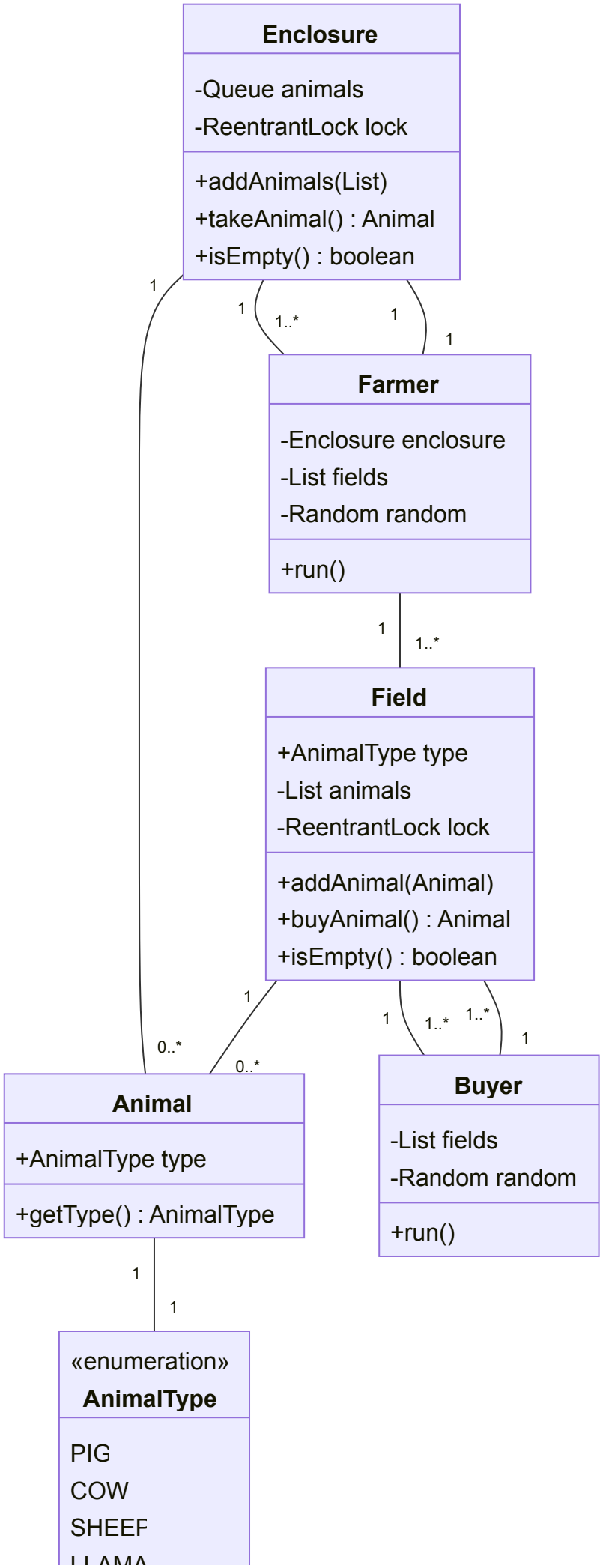
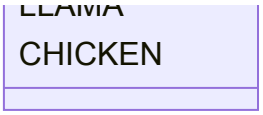
Farm simulation

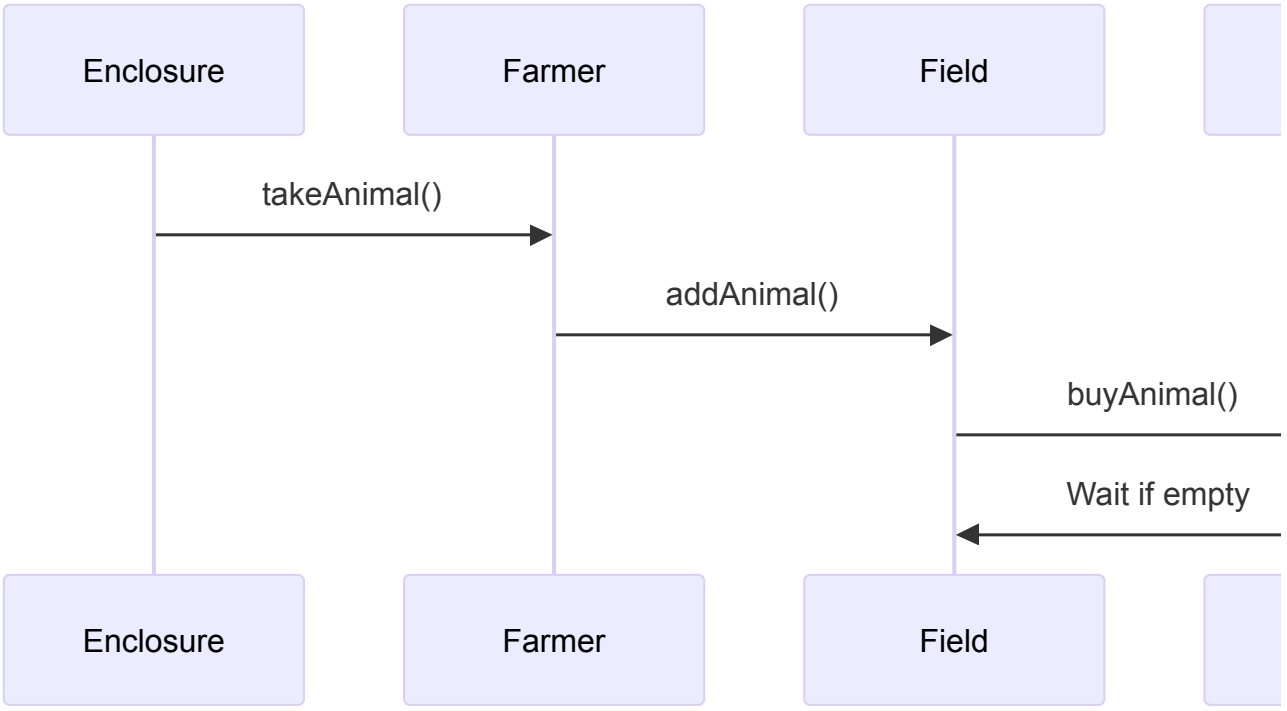

Enclosure
-Queue animals
-ReentrantLock lock
+addAnimals(List)
+takeAnimal() : Animal
+isEmpty() : boolean
Farmer
-Enclosure enclosure
-List fields
-Random random
+run()
Field
+AnimalType type
-List animals
-ReentrantLock lock
+addAnimal(Animal)
+buyAnimal() : Animal
+isEmpty() : boolean
Animal
+AnimalType type
+getType() : AnimalType
Buyer
-List fields
-Random random
+run()
«enumeration»
AnimalType
PIG
COW
SHEEP
LLAMA
1
1..*
0..*



Syncronisation Strategy

`ReentrantLock` : used in enclosure and field to ensure only 1 thread can moidyf them at a time
`synchonized` methods: fine grain control over critical section access
`condition` variables:

- **Buyers wait** when fields are empty
- **Farmers wait** when enclosure is empty



Syncrhonisation rules

1. Farmers and enclosure
   - only **one farmer** at a time can take animals from enclosure
   - `ReentrantLock` to control access
2. Farms and fields
   - Farmer stocks animals in fields
   - Only one farmer can stock a field at a time
3. Buyer and fields
   - Buyer waits if field is empty

- Uses condition variables to notify waiting buyers when an animal is available

Configurable parameters

Parameter	Default Value	Description
`NUM_FARMERS`	2	Number of farmers moving animals from enclosure to fields
`NUM_BUYERS`	3	Number of buyers purchasing animals
`TICK_DURATION_MS`	10 ms	Real-time duration of a single tick
`DAY_TICKS`	1000	Number of ticks per day
`DELIVERY_INTERVAL`	100 ticks	Frequency of new animal deliveries
`BUY_INTERVAL`	10 ticks	Frequency of buyers attempting purchases
`FIELD_CAPACITY`	Unlimited	Max number of animals per field (can be limited for complexity)

Logging

- Thread ID
- tick count
- action performed
- waiting times

```
[TICK 5] [FARMER-1] Taking animal from enclosure...
[TICK 6] [FARMER-1] Moving COW to field...
[TICK 8] [FARMER-1] Stocked 1 COW in Field (COW).
[TICK 9] [BUYER-2] Attempting to buy an animal from Field (COW)...
[TICK 10] [BUYER-2] Purchased 1 COW.
```