CSE 471 - Lab 05

Sequence Diagram

Sequence Diagram

- Most common Interaction Diagram
- Collaboration among objects in a use case
- Vertical dimension : Time ordering of messages
- Horizontal dimension : Objects which participate in the scenario

Calculate Order Price

Let's say, we need to calculate the price for an order. In order to that, first prices for all order items need to be fetched which can be found in the corresponding products' details. After calculating the sub-total, we need to calculate total discount amount which is customer specific and can be fetched from customers' details.

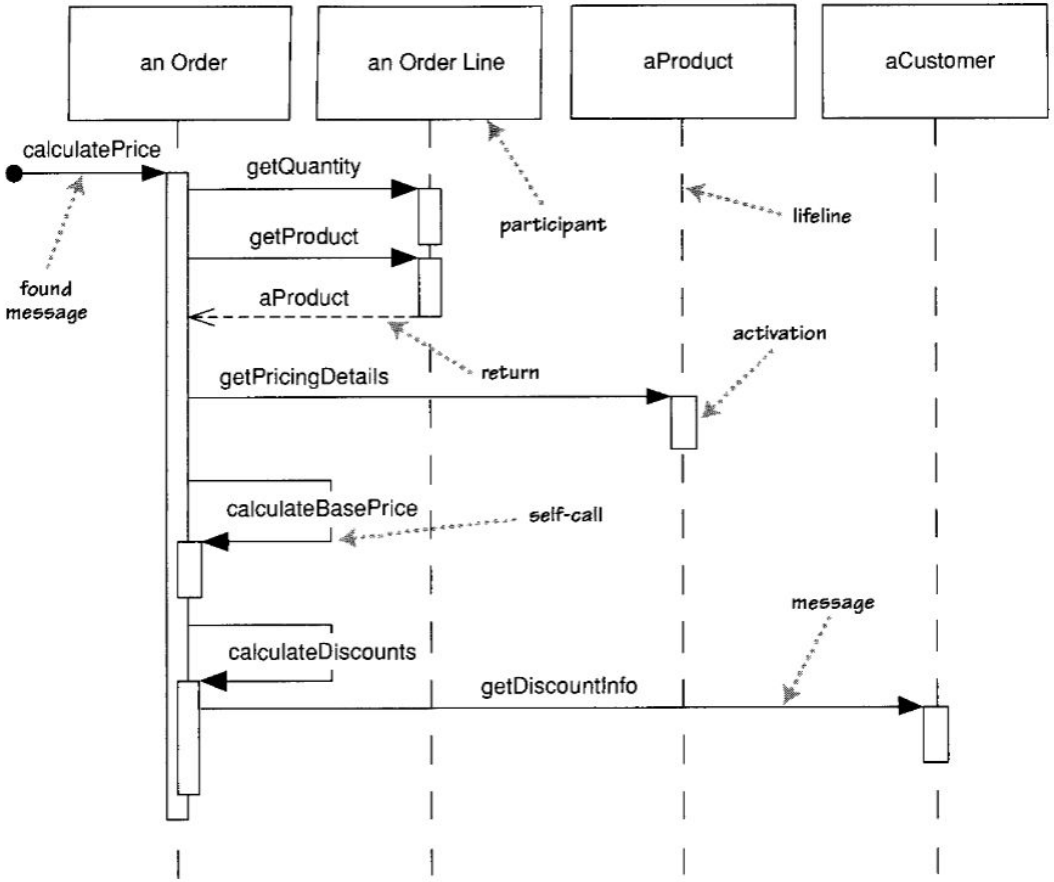


Figure 4.1 *A sequence diagram for centralized control*

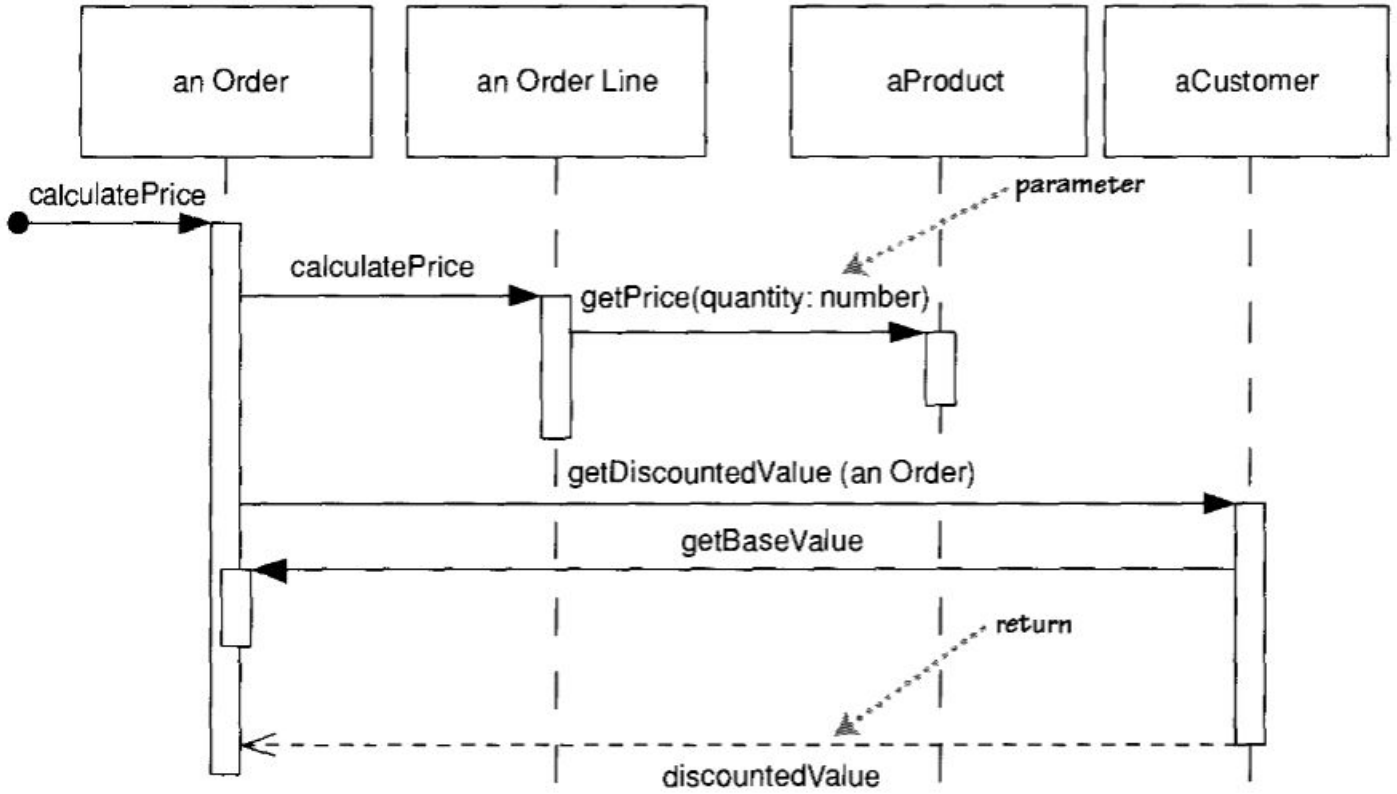


Figure 4.2 *A sequence diagram for distributed control*

Pseudocode

```
procedure dispatch:
    foreach (lineitem)
        if (product.value > $10K)
            careful. dispatch
        else
            regular.dispatch
        end if
    end for
    if (needsConfirmation) messenger.confirm
end procedure
```

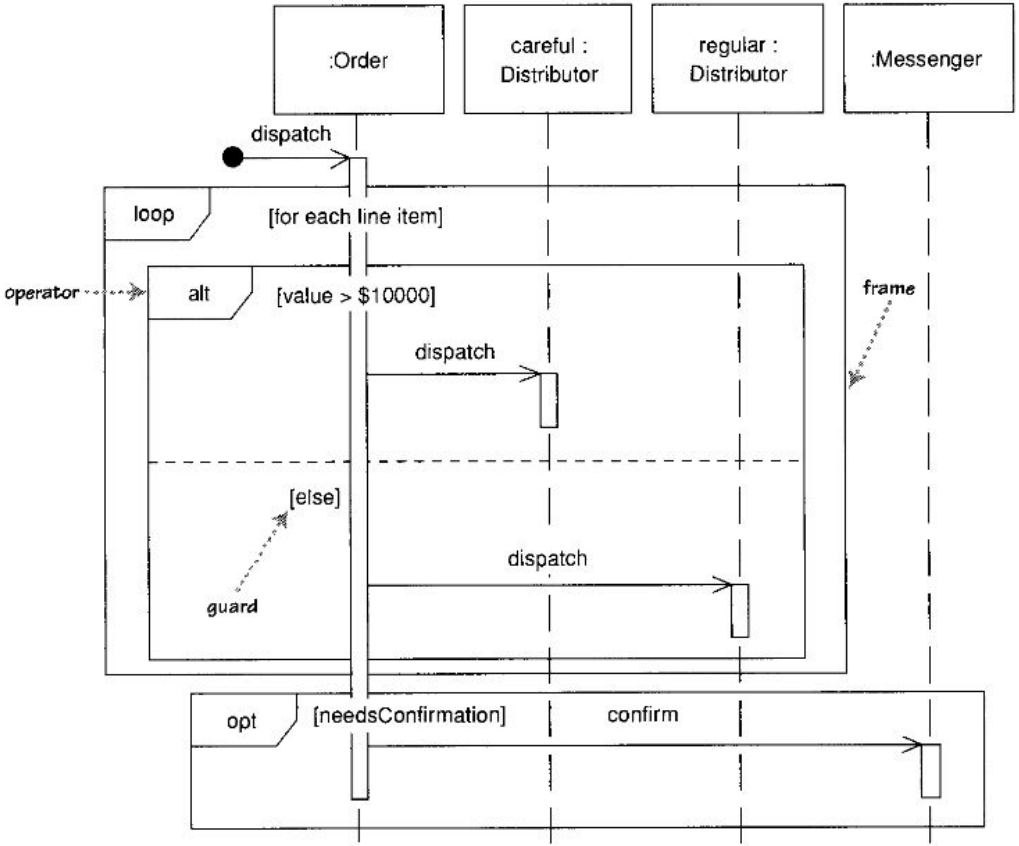


Figure 4.4 *Interaction frames*

Emergency Aircraft Landing

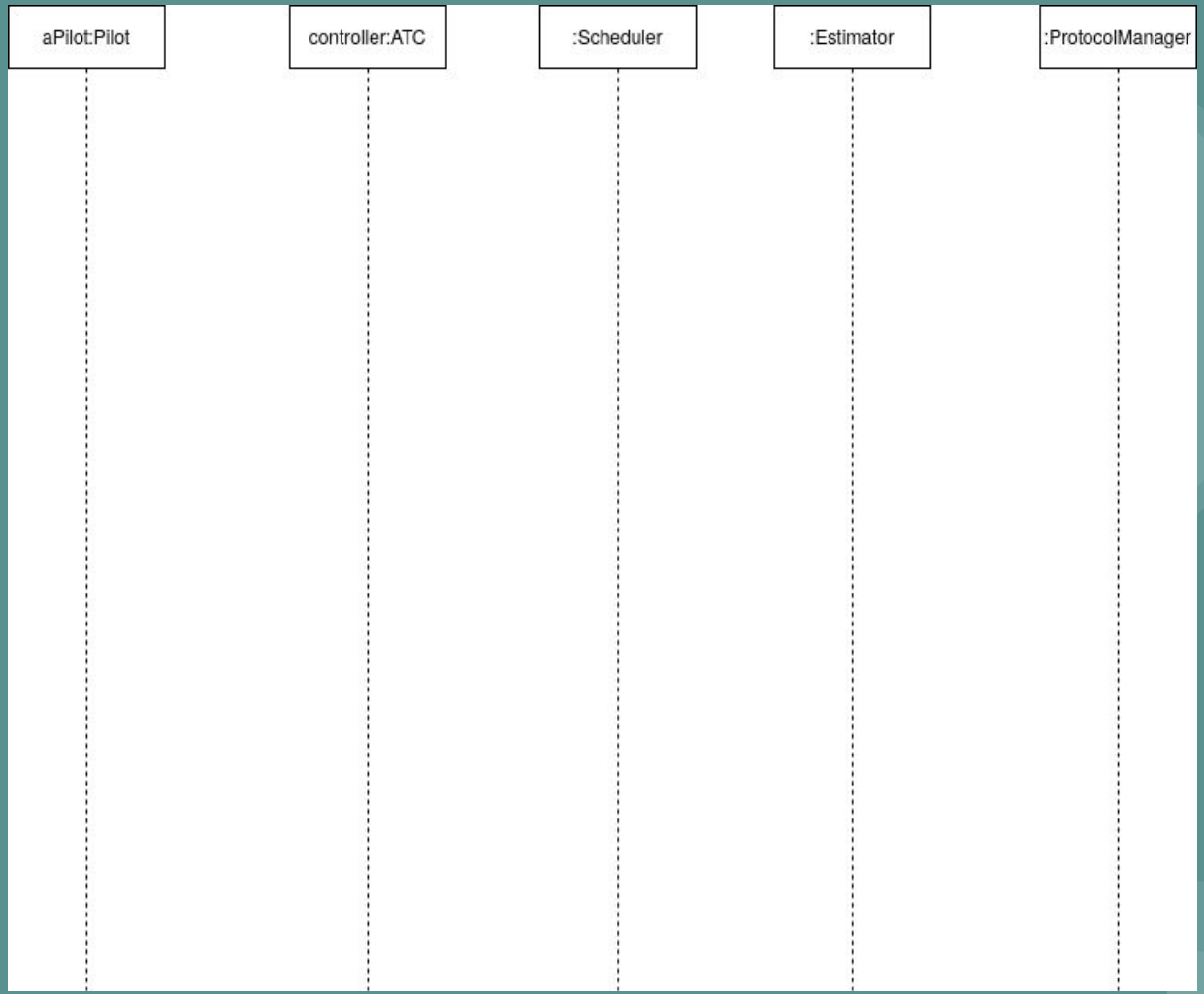
The process starts when the pilot sends a request for landing to Air Traffic Controller. The controller will send new flight landing information to a scheduler and receive a number for landing sequence. Then controller will send this number to an estimator to get the estimated time of landing. After receiving the waiting time from estimator, this information is passed to the pilot from controller.

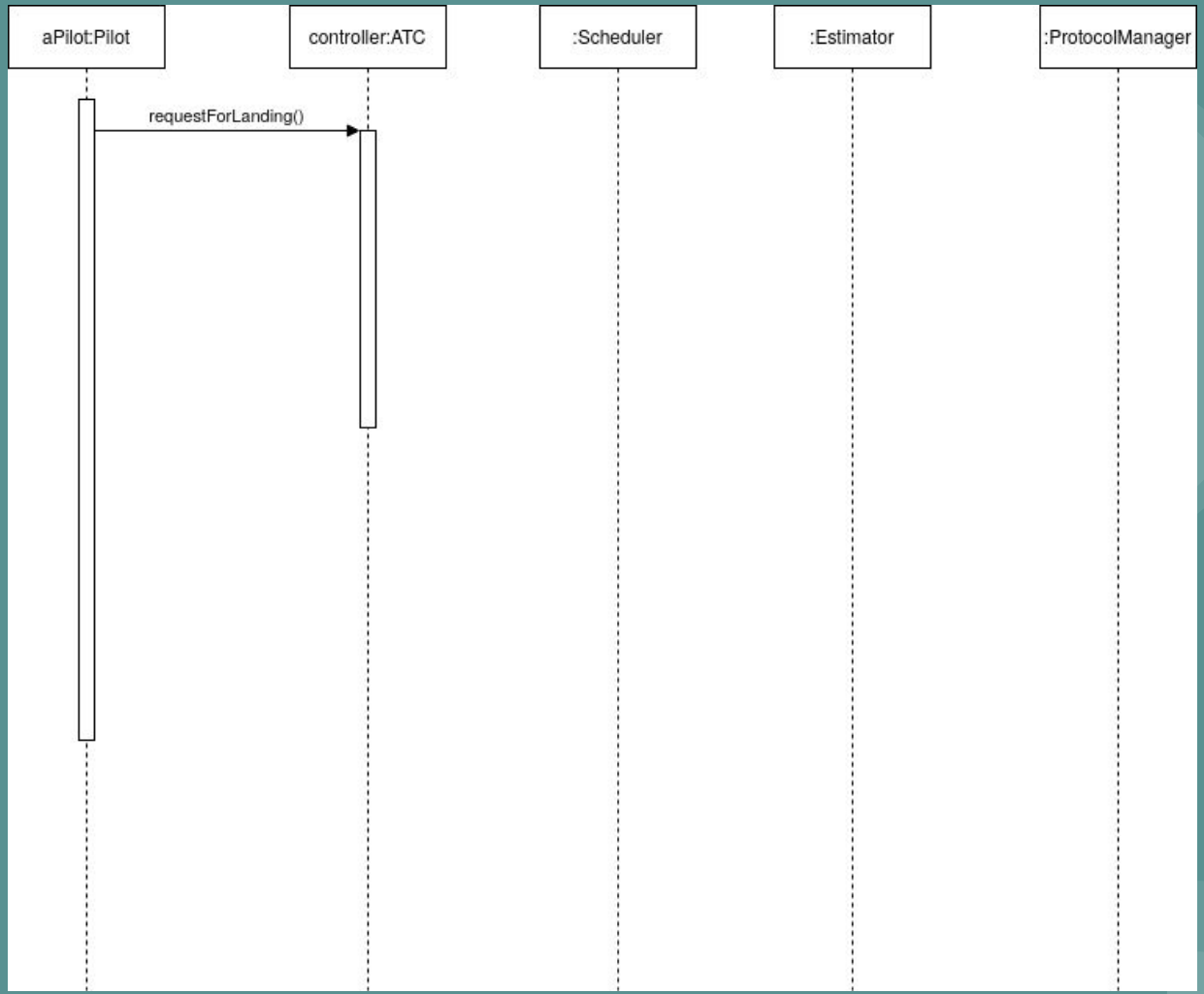
After the estimated time is over, pilot will ask for permission again to land from controller. If permission is given, pilot will ask for runway and weather information from controller and land the aircraft. If, however, permission is denied due to landing sequence congestion, pilot will ask for new waiting time from controller. If the waiting time is over 20 minutes, pilot will declare an emergency through emergency protocol. Once declaration is confirmed from the protocol management, pilot will ask for landing and weather information again and land the aircraft.

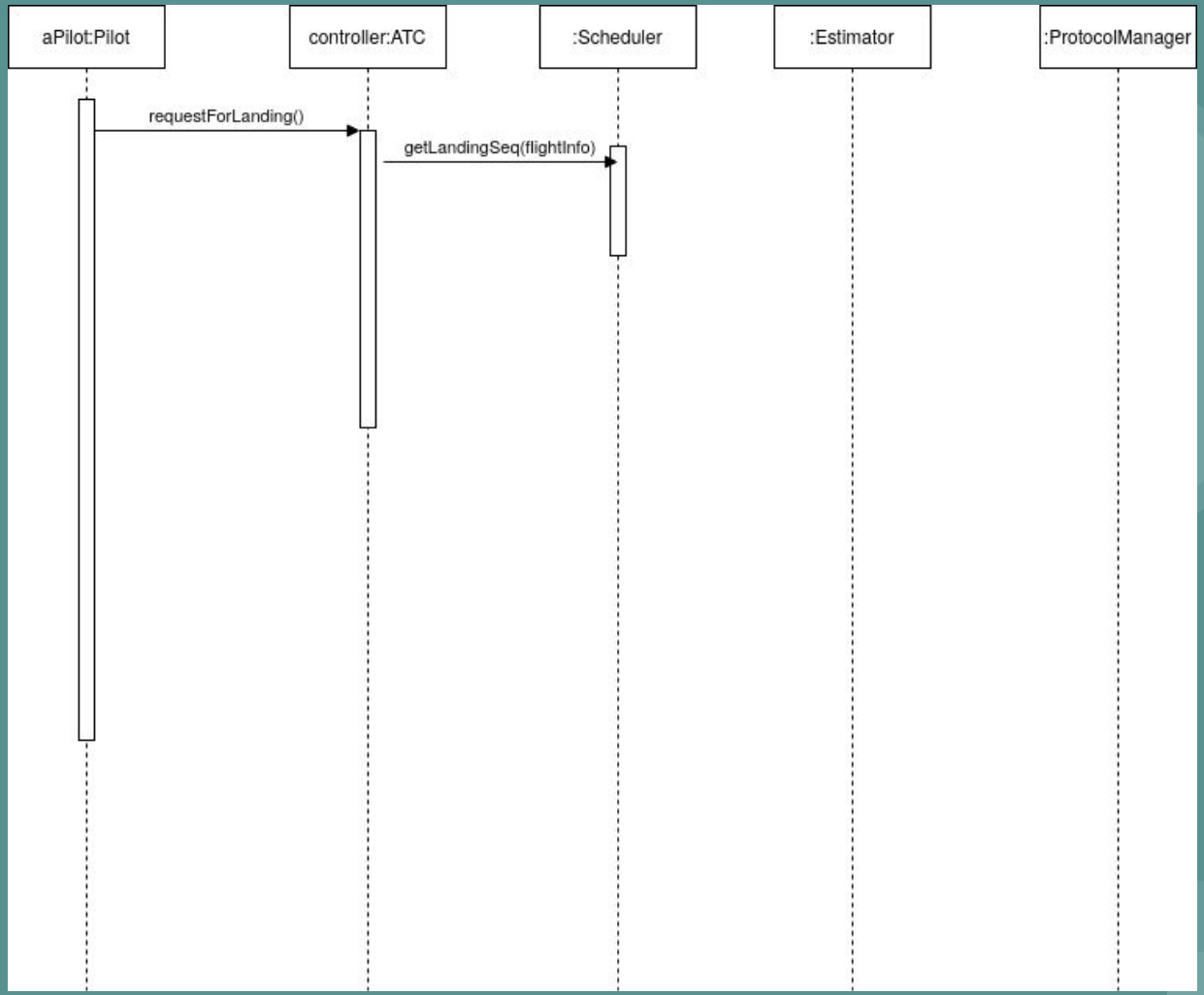
Emergency Aircraft Landing

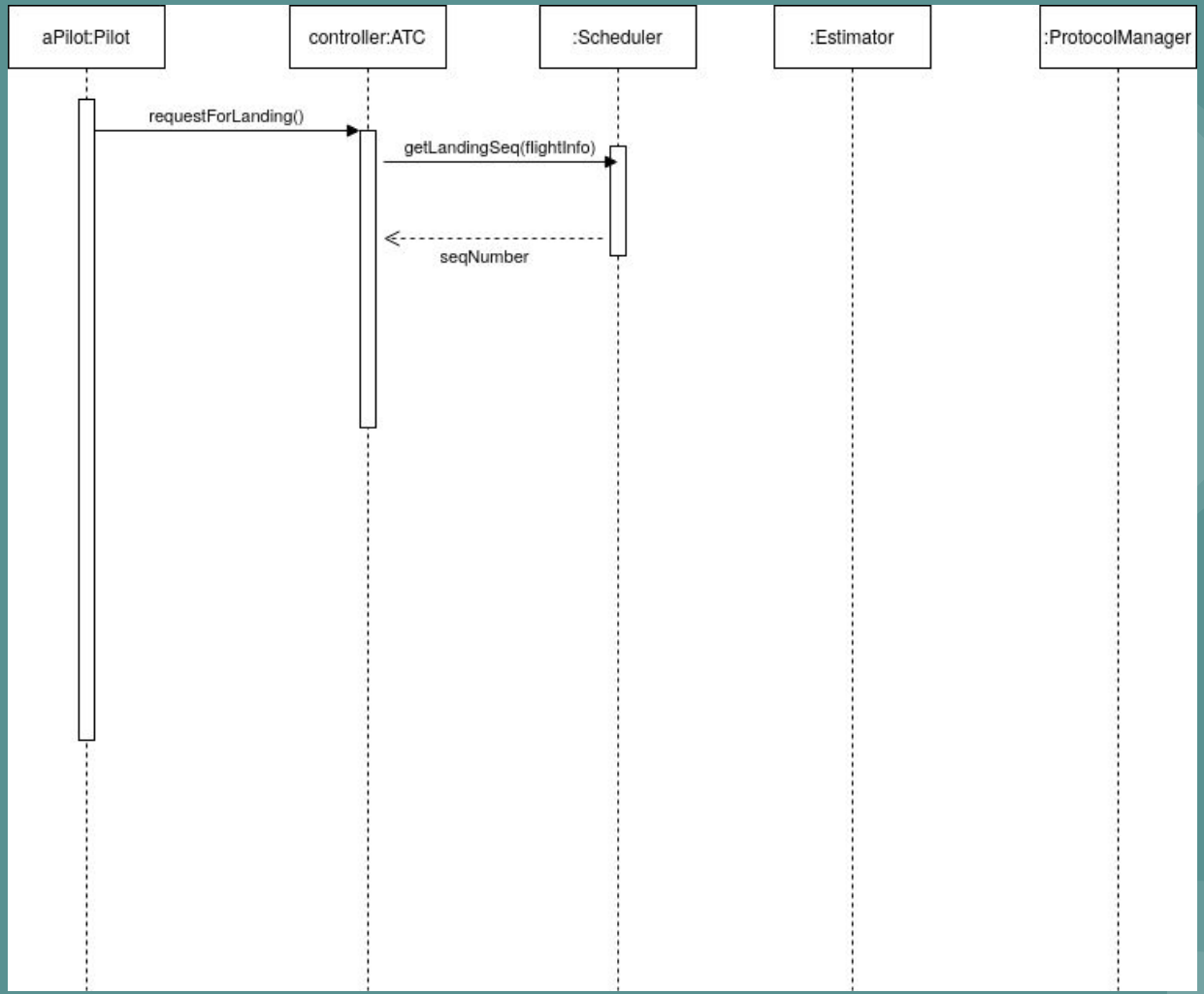
The process starts when the pilot sends a request for landing to Air Traffic Controller. The controller will send new flight landing information to a scheduler and receive a number for landing sequence. Then controller will send this number to an estimator to get the estimated time of landing. After receiving the waiting time from estimator, this information is passed to the pilot from controller.

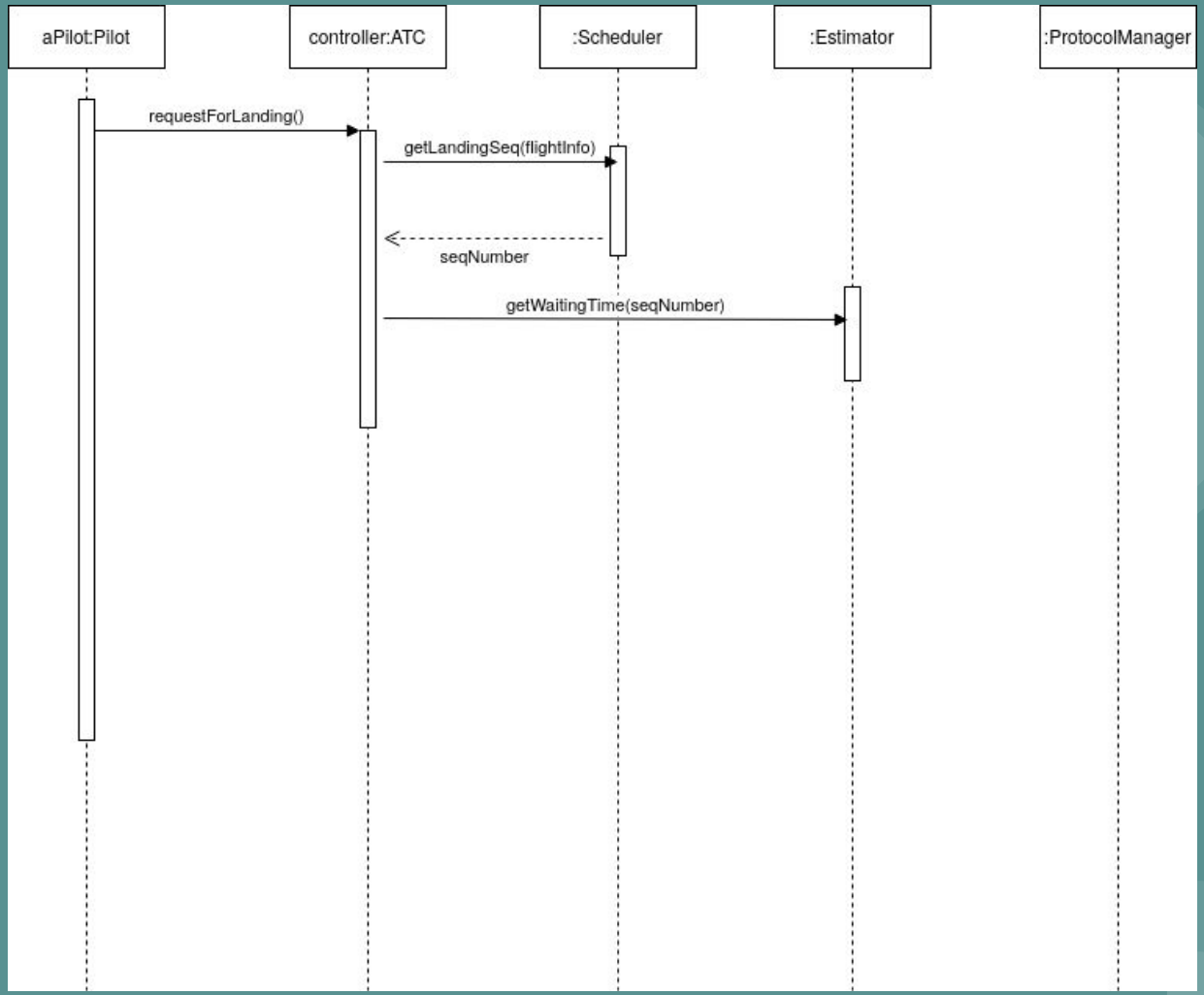
After the estimated time is over, pilot will ask for permission again to land from controller. If permission is given, pilot will ask for runway and weather information from controller and land the aircraft. If, however, permission is denied due to landing sequence congestion, pilot will ask for new waiting time from controller. If the waiting time is over 20 minutes, pilot will declare an emergency through emergency protocol. Once declaration is confirmed from the protocol management, pilot will ask for landing and weather information again and land the aircraft.

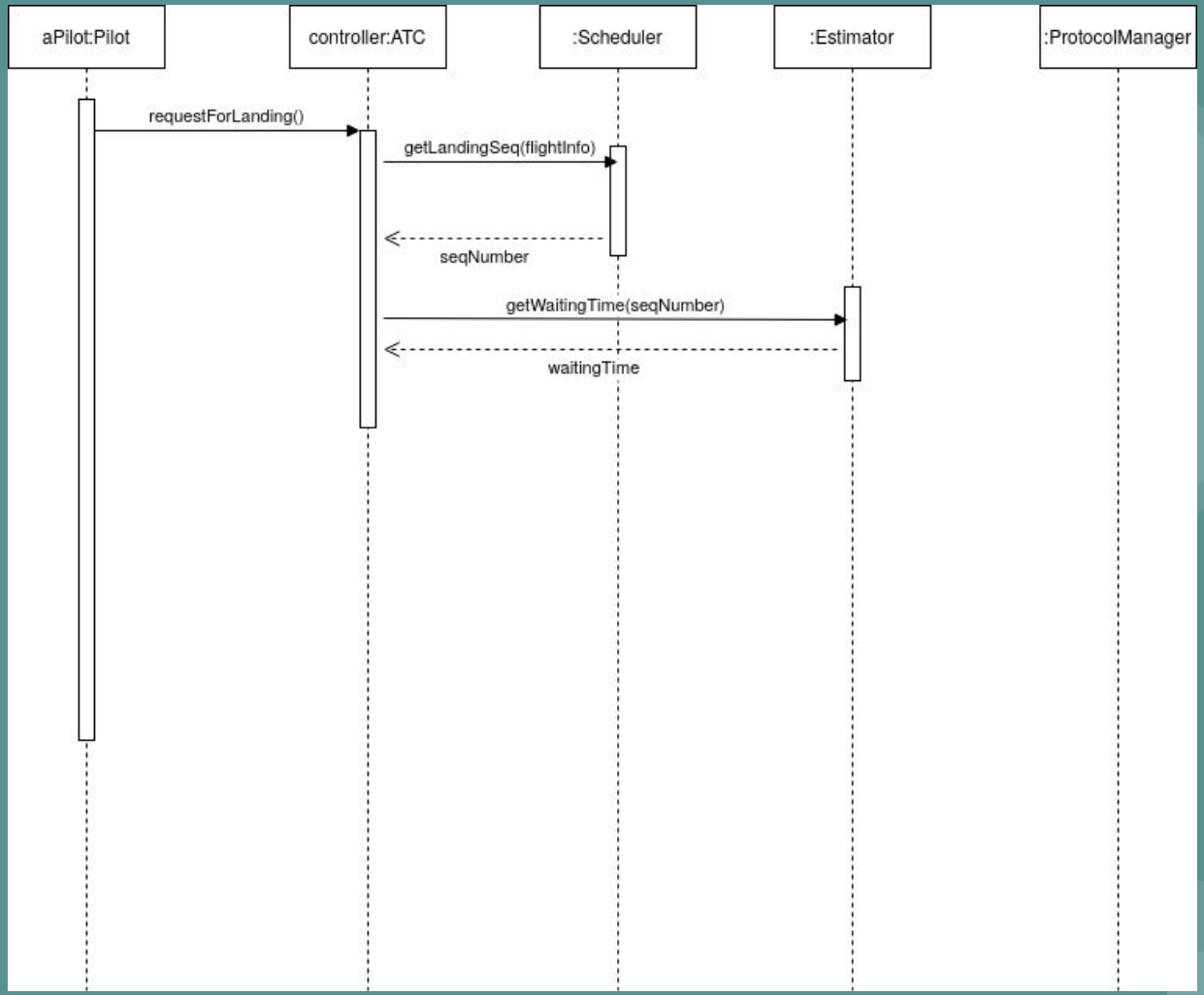

aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager

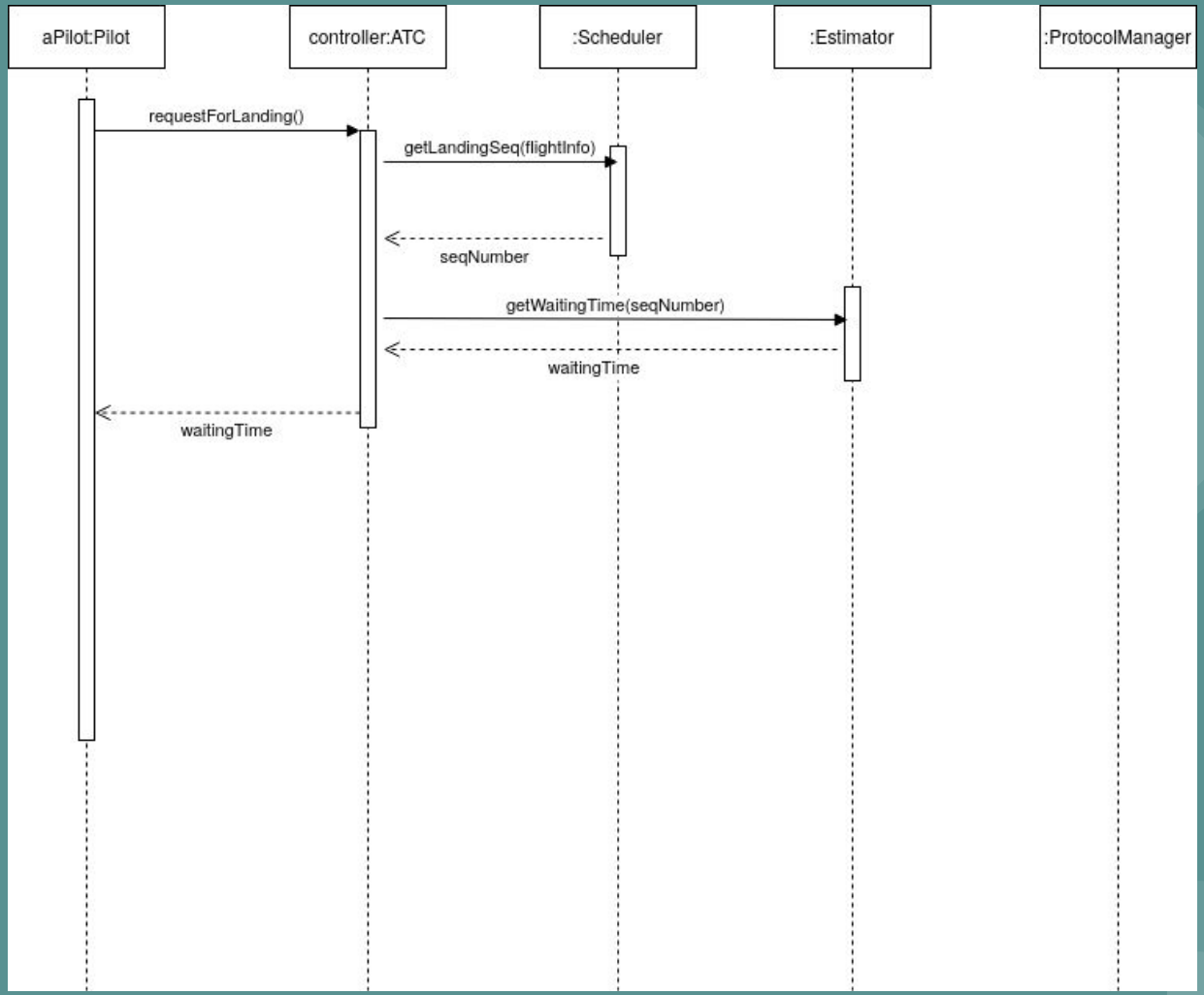

aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager
requestForLanding()

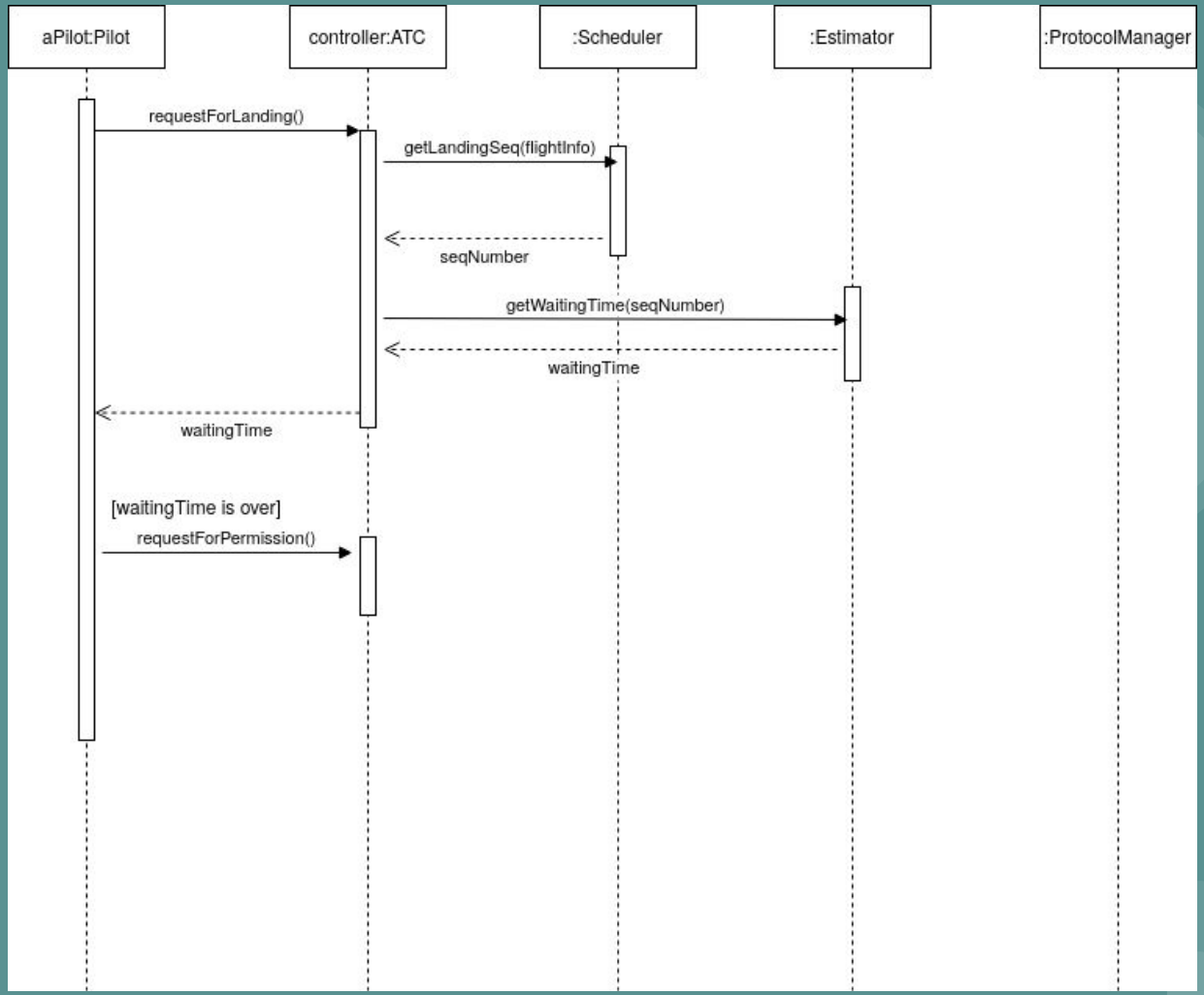

aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager
requestForLanding()
getLandingSeq(flightInfo)

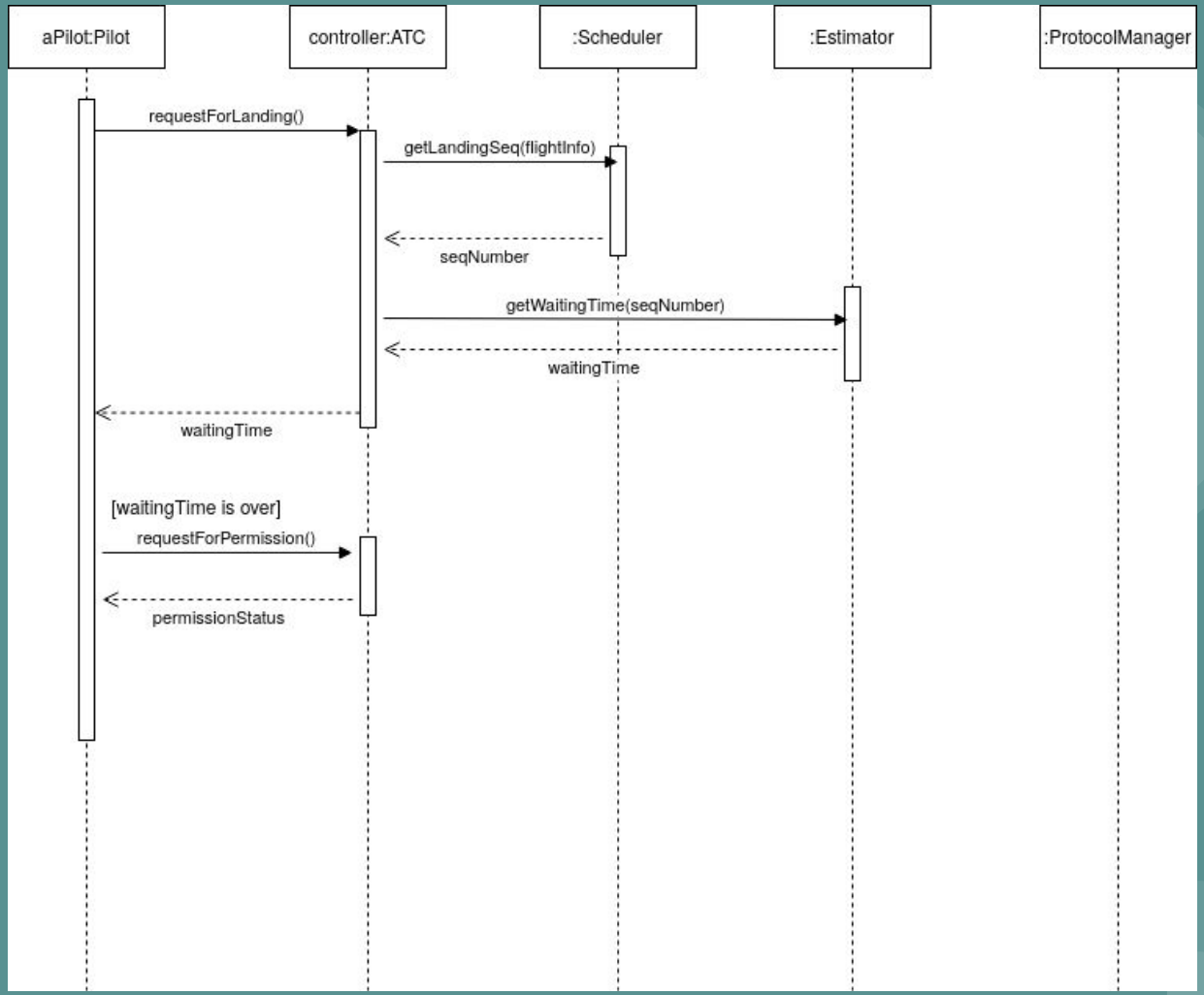

aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager
requestForLanding()
getLandingSeq(flightInfo)
seqNumber

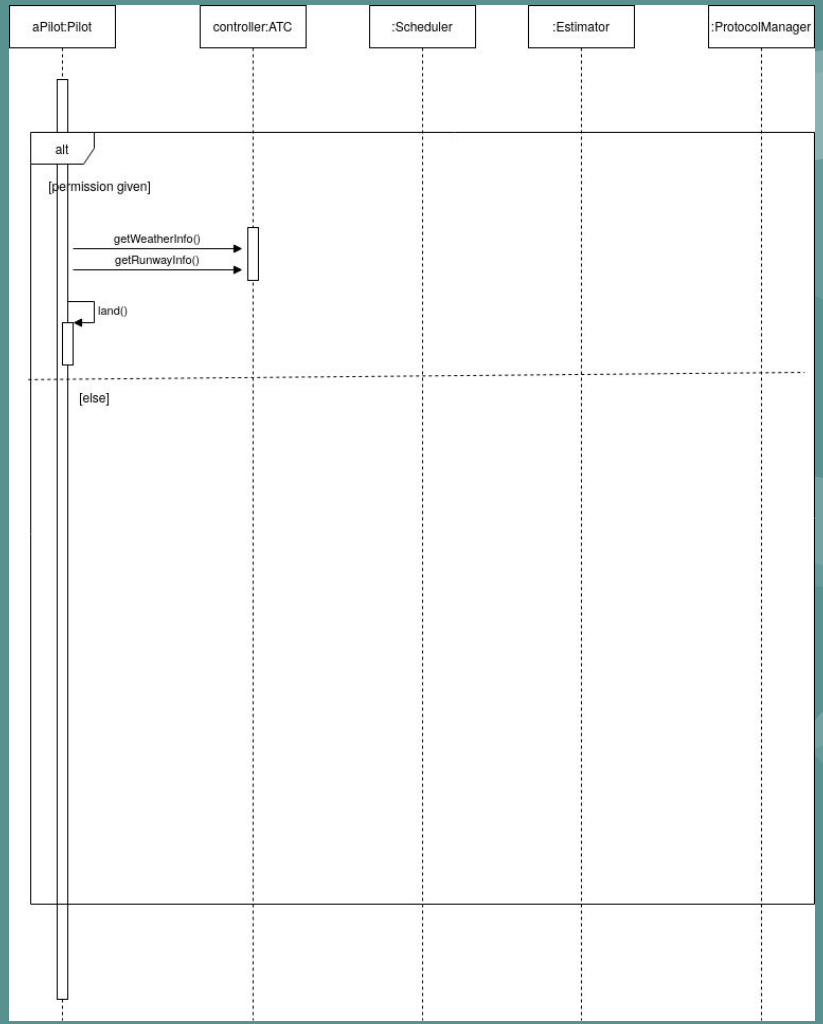

aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager
requestForLanding()
getLandingSeq(flightInfo)
seqNumber
getWaitingTime(seqNumber)

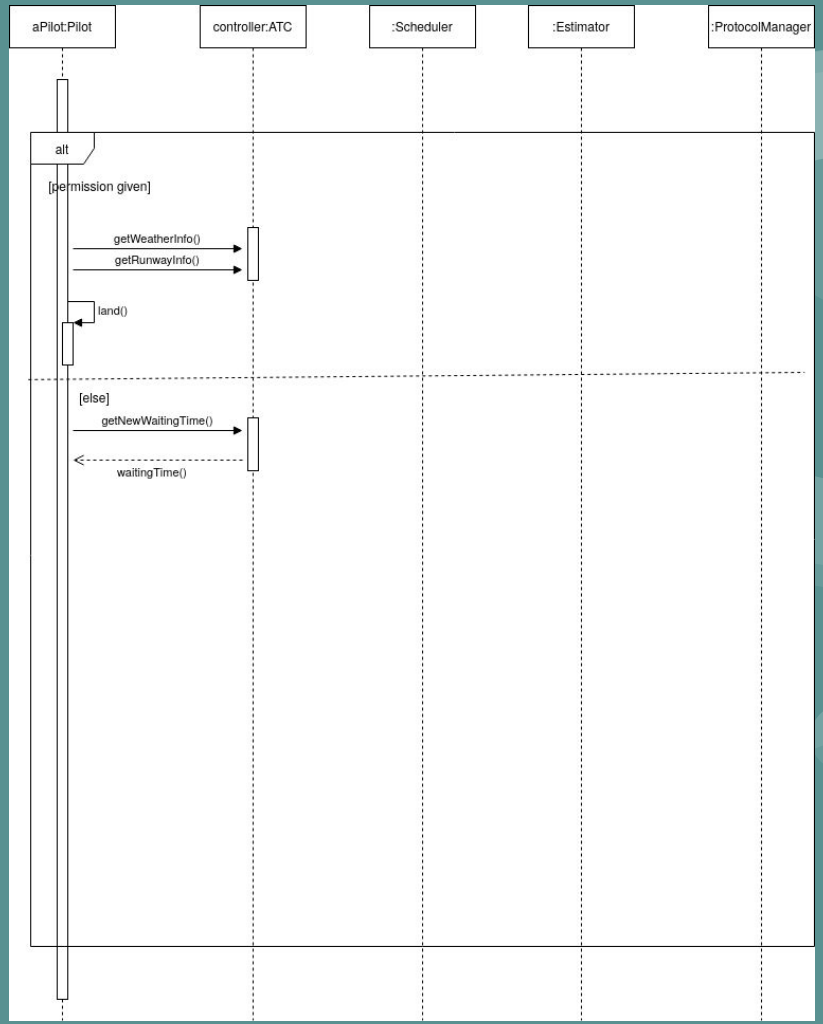

aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager
requestForLanding()
getLandingSeq(flightInfo)
seqNumber
getWaitingTime(seqNumber)
waitingTime

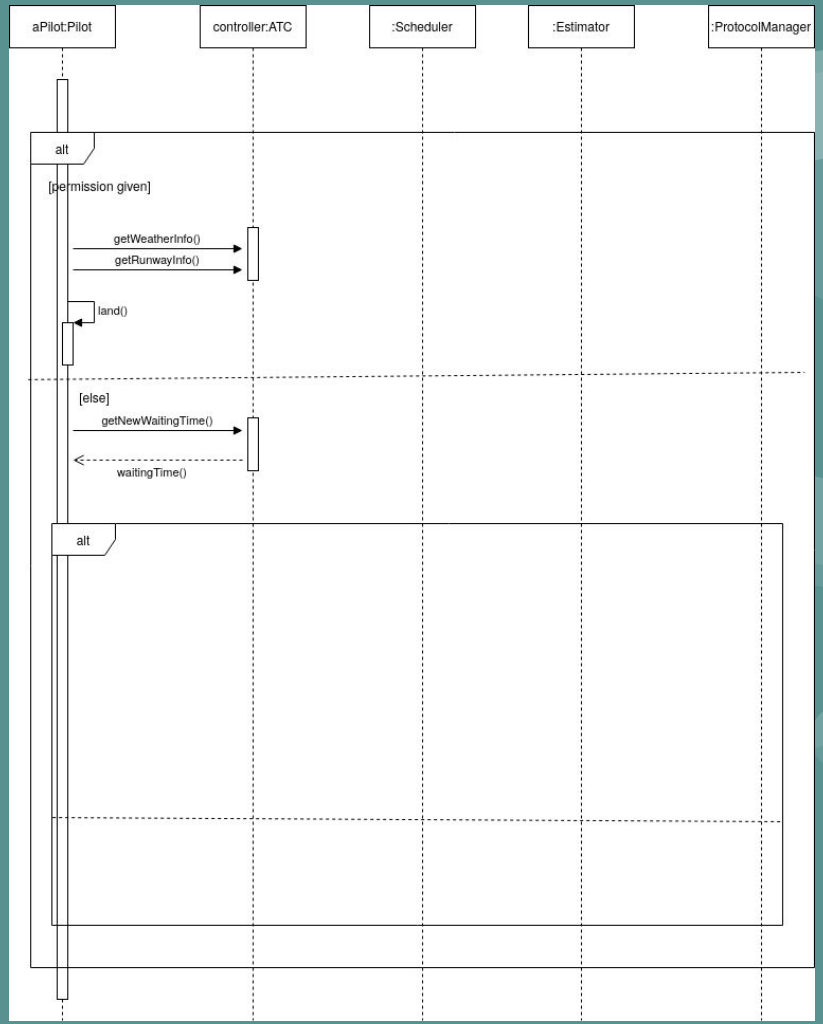

aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager
requestForLanding()
getLandingSeq(flightInfo)
seqNumber
getWaitingTime(seqNumber)
waitingTime
waitingTime

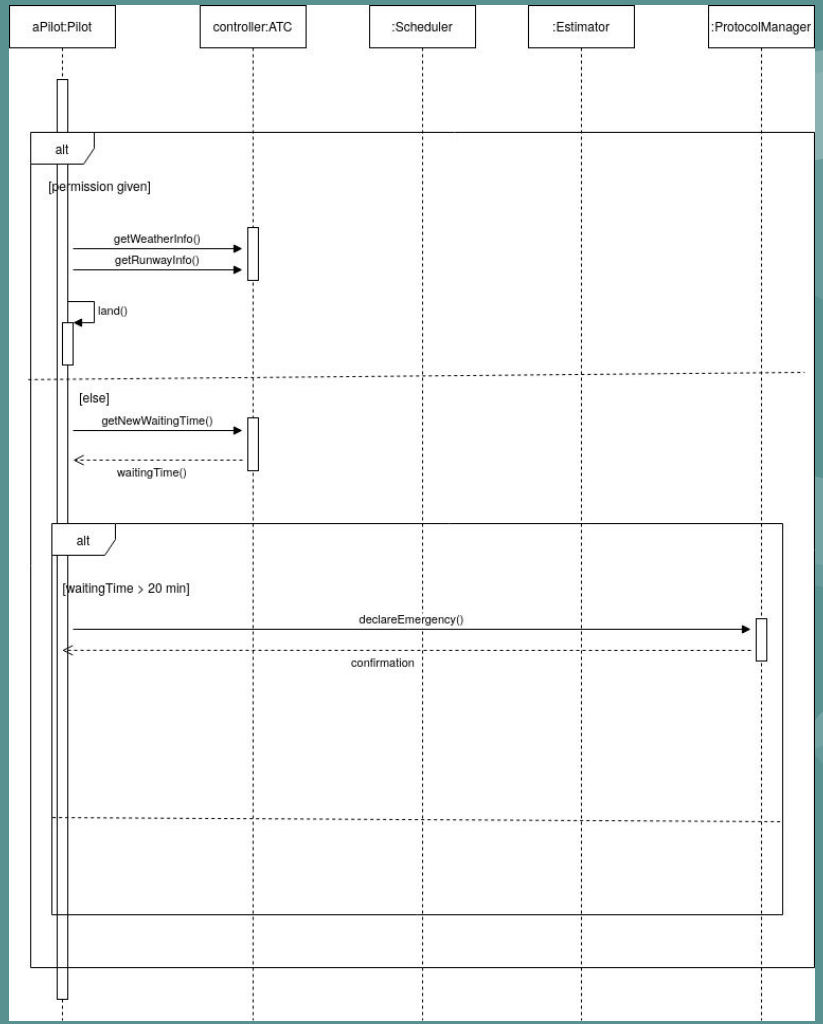

aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager
requestForLanding()
getLandingSeq(flightInfo)
seqNumber
getWaitingTime(seqNumber)
waitingTime
waitingTime
[waitingTime is over]
requestForPermission()

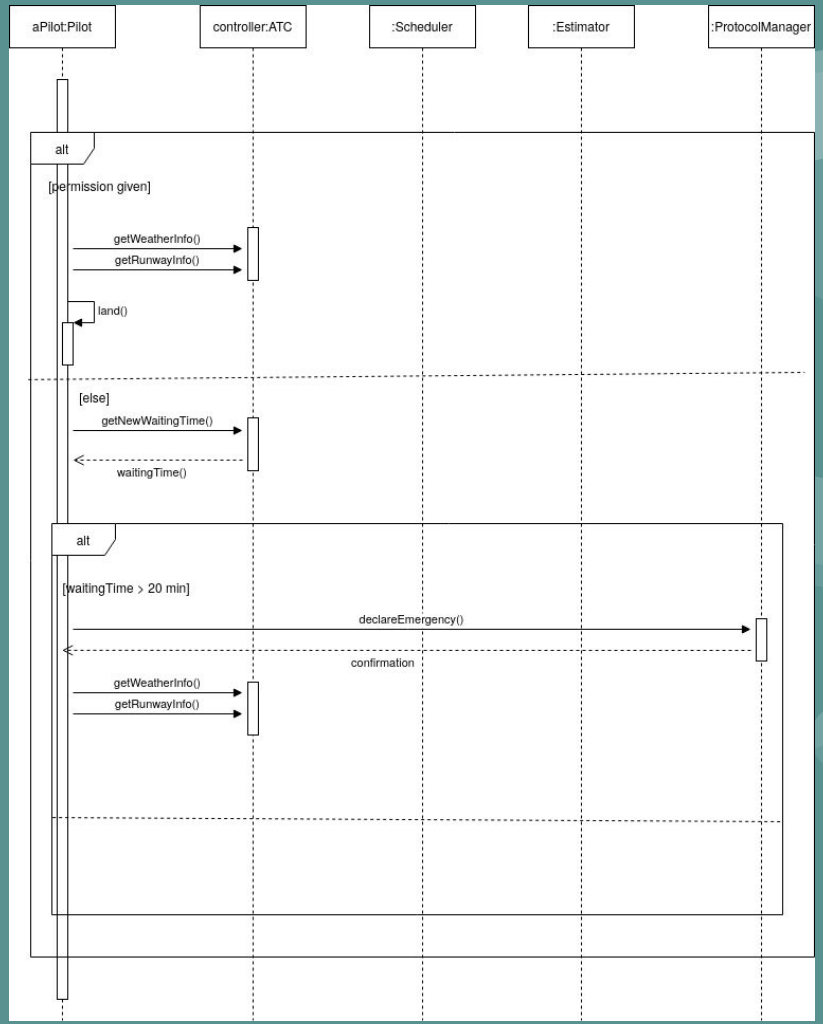

aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager
requestForLanding()
getLandingSeq(flightInfo)
seqNumber
getWaitingTime(seqNumber)
waitingTime
waitingTime
[waitingTime is over]
requestForPermission()
permissionStatus

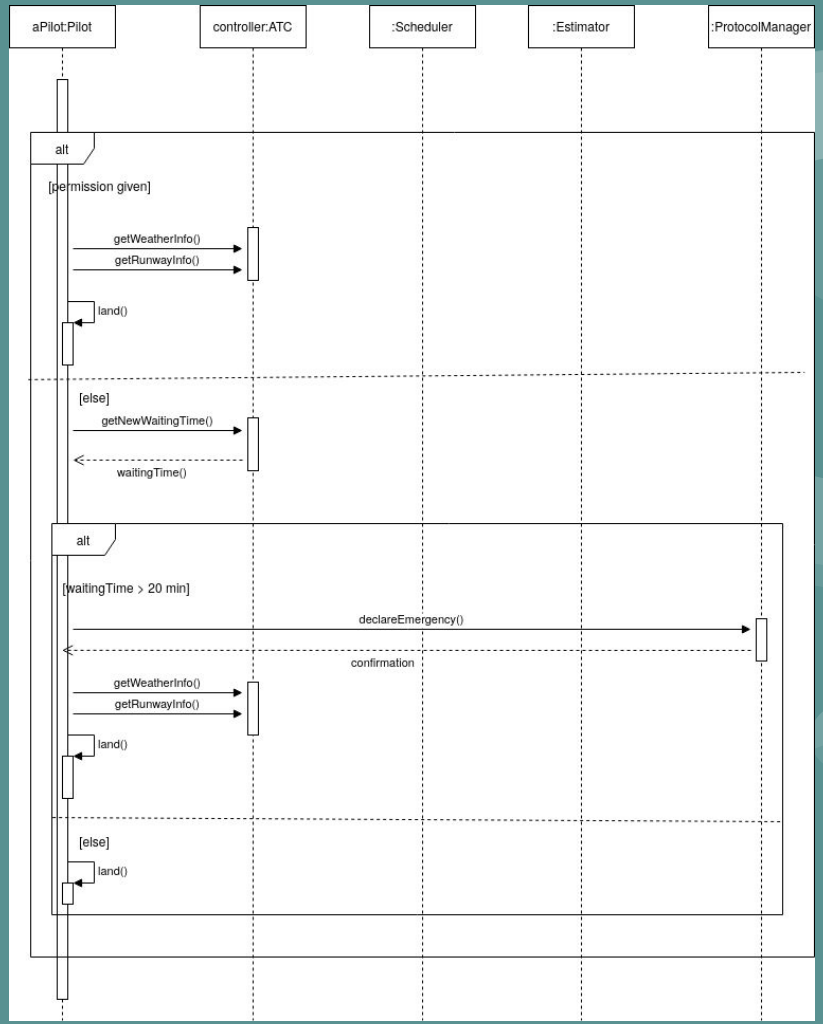

aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager
alt
[permission given]
getWeatherInfo()
getRunwayInfo()
land()
[else]


aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager
alt
[permission given]
getWeatherInfo()
getRunwayInfo()
land()
[else]
getNewWaitingTime()
waitingTime()


aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager
alt
[permission given]
getWeatherInfo()
getRunwayInfo()
land()
[else]
getNewWaitingTime()
waitingTime()
alt


aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager
alt
[permission given]
getWeatherInfo()
getRunwayInfo()
land()
[else]
getNewWaitingTime()
waitingTime()
alt
[waitingTime > 20 min]
declareEmergency()
confirmation


aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager
alt
[permission given]
getWeatherInfo()
getRunwayInfo()
land()
[else]
getNewWaitingTime()
waitingTime()
alt
[waitingTime > 20 min]
declareEmergency()
confirmation
getWeatherInfo()
getRunwayInfo()


aPilot:Pilot
controller:ATC
:Scheduler
:Estimator
:ProtocolManager
alt
[permission given]
getWeatherInfo()
getRunwayInfo()
land()
[else]
getNewWaitingTime()
waitingTime()
alt
[waitingTime > 20 min]
declareEmergency()
confirmation
getWeatherInfo()
getRunwayInfo()
land()
[else]
land()