recit-4

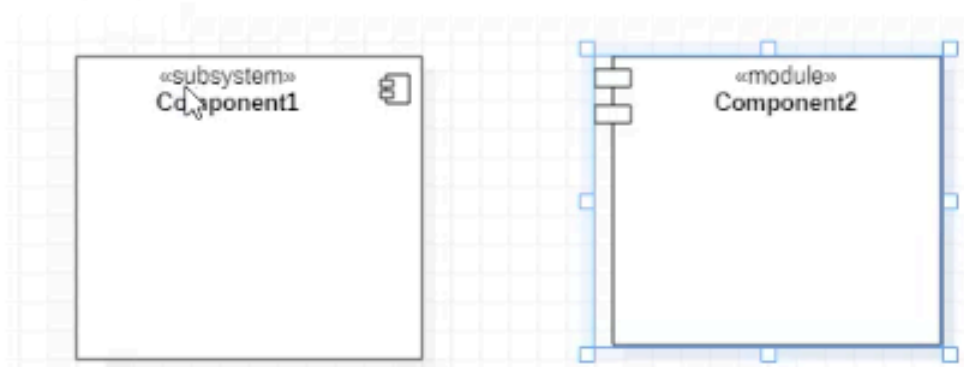
component diagram:

structural
how components work together
component diagram for whole system in YOLO
components: independent and encapsulated units in the system
-provides interfaces to each other to interact with each other

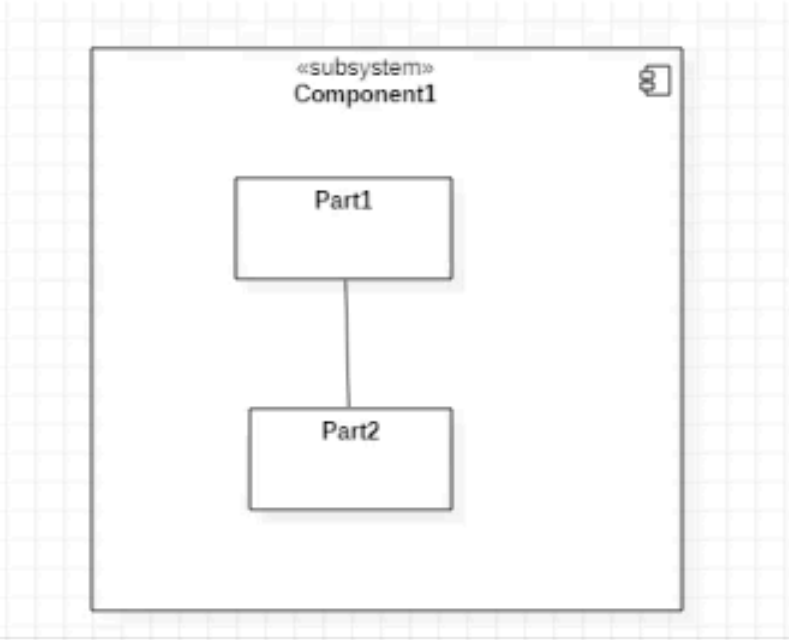
interfaces:
1-required
2-provided

components
database
package
library
application
*for yolo rapsberry pi unit

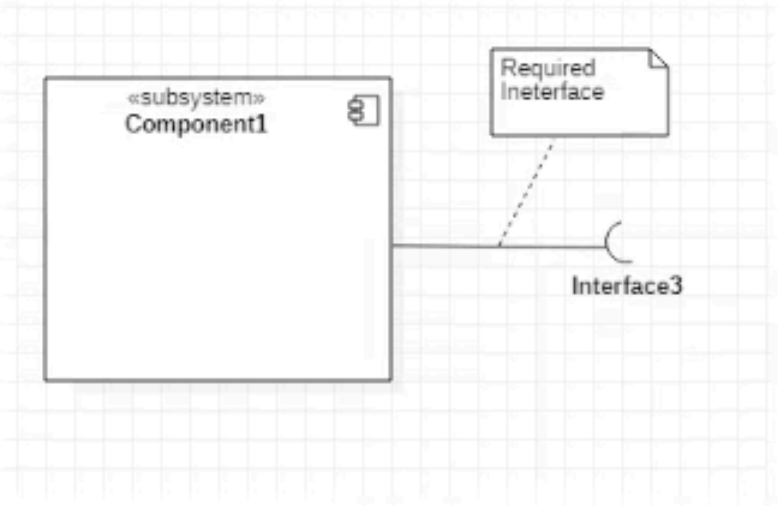
Component:



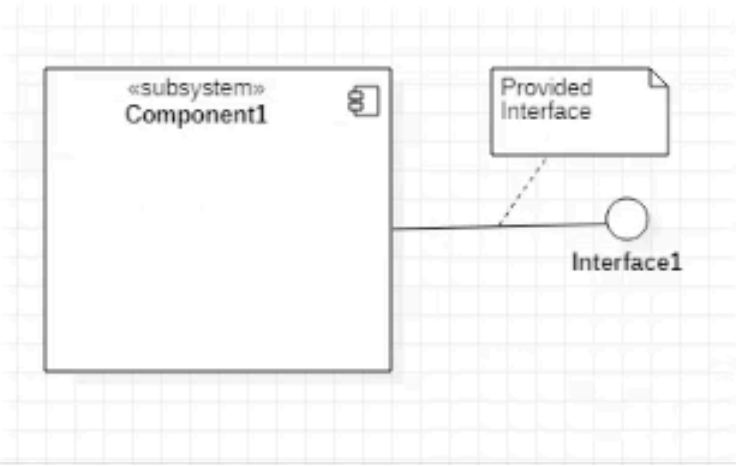
Parts:



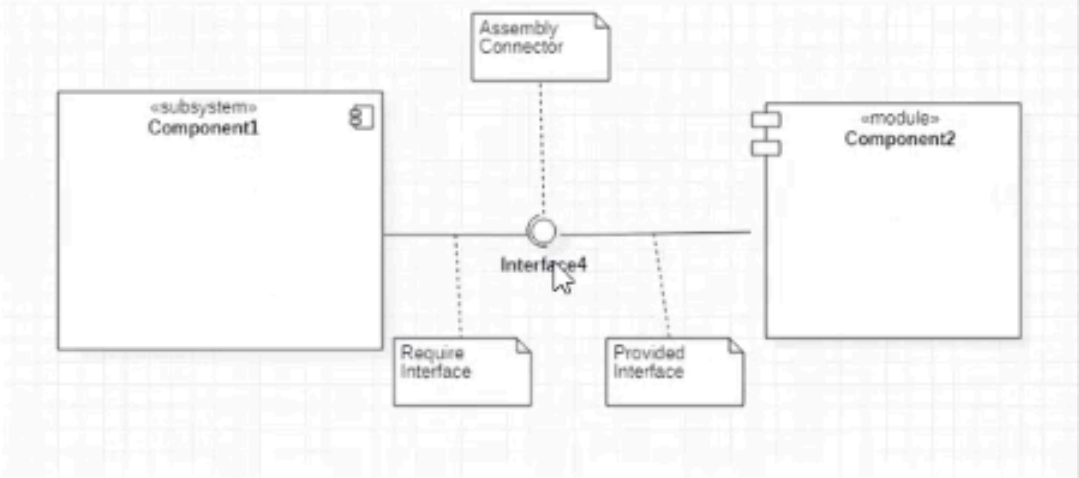
Require Interface:



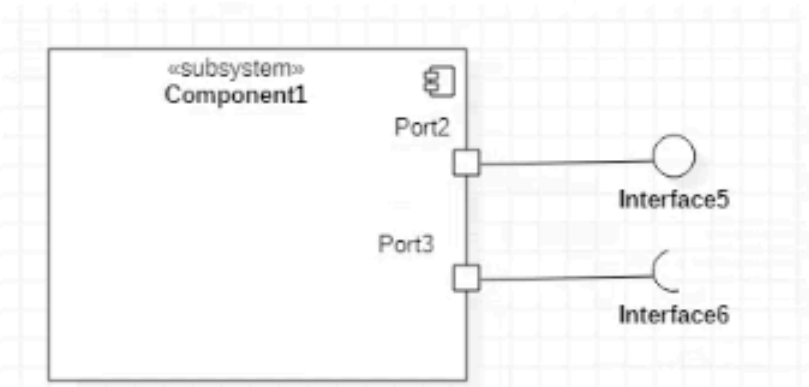
Provided Interface:

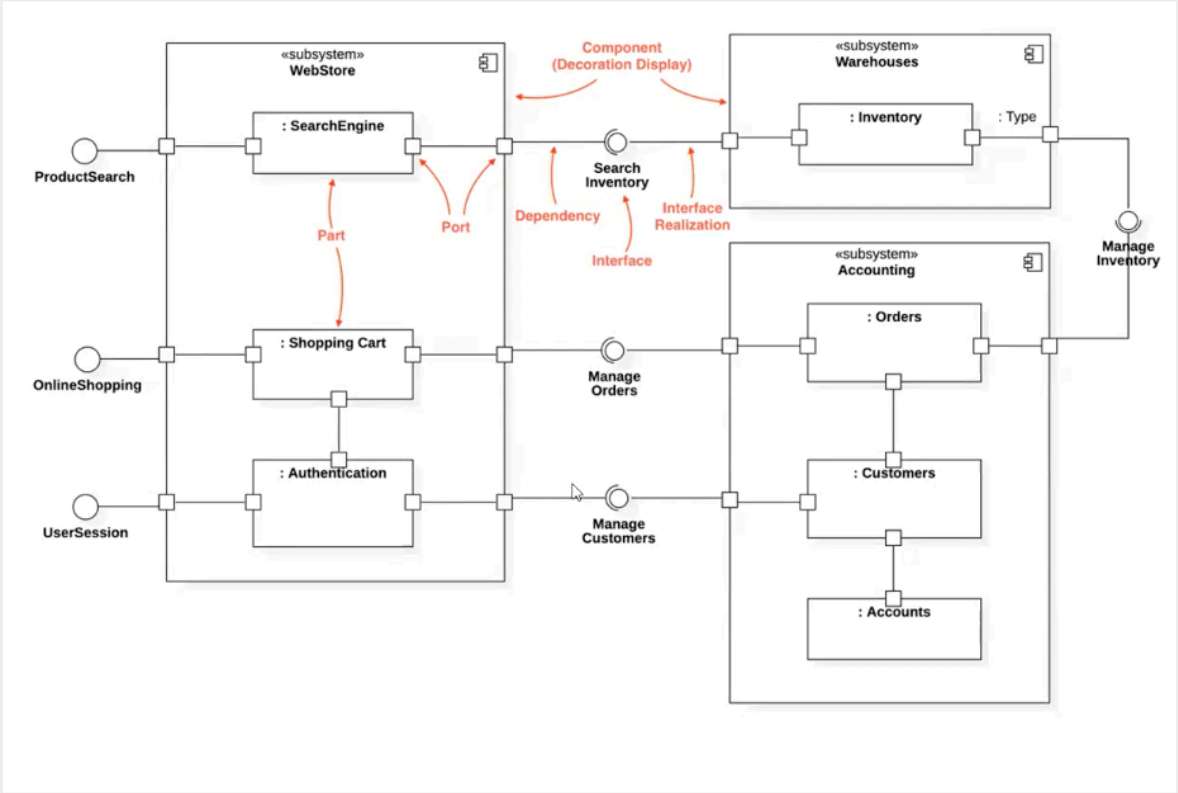


Assembly Interface:



Ports:





deployment diagram:
deployment-> dağıtım
structural

picture of how the physical system will look when it is all together
how the artifacts(eser) are deployed on a system hardware and how pieces of this hardware connect to each other and how they communicate

consists of 3 main notations:

1-node:
computational resource which artifacts are deployed for execution
-devices, execution environment, server, java virtual machine, python environment

2-an artifact:
what the developer for this system developed
-source files, cpp ,c , java.., .exe database files
artifacts should be on the node(device or execution environment)

3-communication:

start with frame

Deployment of YOLOs Components shows how the artifacts are deployed on the YOLO hardware and how pieces of thsi hardware connect to each other and how they communicate with each other. It has three nodes. Sensor node is a device. The developed touch sensor and optical sensor artifacts are on this sensor node. They send data to the Raspberry Pi device node. Actuators node also is a device. The developed wheel actuator and led actuator are on this actuators node. the Raspberry Pi device node sends data to the actuators node. The Raspberry Pi also is a device node. It has Raspbian Stretch Lite OS

execution environment in it. This environment executes python files. It also executes training_data.txt and cached_features.txt files with train_data.py to perform machine learning.