* Impresa
* Impresa può lavorare su più cantieri contemporaneamente
* Impresa ha:
* Reparto -> amministrativo, operativo
* Reparto amministrativo interagisce con:
* Fornitore[[1]](#footnote-1)(frame a parte per simulare il fornitore)
* Ente locale[[2]](#footnote-2)(frame a parte per simulare l’ente)
* Cliente (frame a parte per simulare il cliente)
* Fornitore fornisce:
* Prodotto
* Servizio
* Fornitore riceve:
* Prodotti e servizi necessari all’impresa (forniti dall’impresa)
* Ente locale rilascia:
* Permesso (non modificabile, final)
* Ente locale riceve:
* Richieste di permesso dall’impresa
* Cliente rilascia:
* Richiesta di un cantiere con valore del cantiere(un cantiere superclasse del cantiere dell’impresa)
* Reparto amministrativo gestisce:
  + Assunzione/licenziamento personale
  + Pagamento stipendio personale
  + Assegnamento/rimozione personale al cantiere
  + Controllo permessi cantiere
* Reparto operativo si occupa delle attività del cantiere
* Personale -> impiegato, operaio, quadro, dirigente <schema retributivo diverso>
* CantiereImpresa
* CantiereImpresa sottoclasse di Cantiere(preso dal Cliente) con fields come elenco lavoratori, dirigente(se c’è), permesso ente
* Cantiere personale necessario:
  + Responsabile[[3]](#footnote-3) = (valore cantiere > 500.000 ? dirigente : quadro)
  + Caposquadra = quadro
  + Operai
  + Squadra = n.1 Caposquadra, n.x operai
  + Lavoratori[[4]](#footnote-4) non impiegati in altri cantieri
* Report
* Report tipi:
  + Lista lavoratori non impiegati[[5]](#footnote-5)
  + Lista cantieri, x<=valore Cantiere <=y[[6]](#footnote-6)
  + Elenco fornitori di un determinato servizio o prodotto
* Report ordinamento:
  + Alfabetico
  + Stipendio crescente
  + Valore cantiere crescente
  + In base al ruolo + (alfabetico || stipendio crescente)
* Eccezioni controllate:
* LavoratoreNonDisponibileException (quando si cerca di aggiungere a una squadra un lavoratore già impegnato in un altro cantiere)
* LavoratoreNonEsistenteException (quando si cerca di licenziare un lavoratore non esistente)
* LavoratoreEsistenteException (quando si cerca di assumere un lavoratore già esistente)
* Eccezioni non controllate:
* PermessiEnteNonRilasciatiException (quando si tenta di rendere operativo un cantiere senza i permessi dell’ente locale)
* PermessiEnteNegatiException (quando si tenta di rendere operativo un cantiere dopo che i permessi sono stati negati)
* NumeroMassimoCantieriException (quando si tenta di assegnare il ruolo di responsabile ad un dirigente e tale dirigente è già responsabile di x cantieri)
* Tutto l’output del sistema deve essere su interfaccia grafica
* Salvare le info dell’impresa tra un’esecuzione e l’altra

1. La classe del fornitore avrà una lista in cui ci saranno i prodotti forniti all’impresa e un’altra lista per i servizi forniti. [↑](#footnote-ref-1)
2. L’ente locale è l’organo che rilascia i permessi per poter rendere operativo un cantiere, senza i permessi non si può lavorare nel cantiere. Assegnare il personale a un cantiere prima del rilascio dei permessi è una scelta soggettiva, tuttavia, il personale sarà considerato non disponibile per gli altri cantieri. [↑](#footnote-ref-2)
3. Il responsabile è un dirigente se valore cantiere >= 500.000. Se valore cantiere < 500.000 il responsabile è un quadro. Se non vi è un quadro disponibile, un dirigente sarà il responsabile. Verrà scelto il primo dirigente con il minor numero di cantieri assegnati. [↑](#footnote-ref-3)
4. I lavoratori, e quindi coloro che possono lavorare su un solo cantiere, sono: operai e quadri. Ciò significa che un dirigente può essere il responsabile di più cantieri, massimo x cantieri. [↑](#footnote-ref-4)
5. Nel caso dei dirigenti verrà indicato anche il numero di quanti cantieri possono essere ancora responsabili. [↑](#footnote-ref-5)
6. L’utente potrà scegliere se includere anche i cantieri non ancora operativi (senza i permessi dell’ente locale). [↑](#footnote-ref-6)