# CAMBIAMENTI DA FARE ALL’ANALISI DEI REQUISITI

## Use Case Diagram

E niente, semplicemente inserire il nuovo use case diag

## Requisiti Funzionali -> Attori Coinvolti (3.1)

Rimuovere Manutentore e Moderatore. Aggiungere “Sensori Hive” come attore automatizzato.

Descrizione:

I “Sensori Hive” saranno un set di sensori per ogni arnia che monitoreranno le situazioni ambientali all’interno dell’arnia stessa per segnalare criticità. In particolare, saranno composti da 3 sensori: un sensore di temperatura, con l’obiettivo principale di assicurarsi che la temperatura dell’arnia non scenda sotto livelli critici in inverno, un sensore di umidità, per assicurarsi che questa rimanga in un range ben definito (50%-60%) per avere il miele sia della qualità migliore possibile ed infine un sensore interno ai contenitori del nutrimento, che segnali quando il livello (essendo liquido) scenda sotto un livello minimo, segnalando quindi la necessità di riempirlo. I sensori permetteranno all’apicoltore di sapere quali misure adottare per migliorare la condizione delle “arnie intelligenti”, quali modificarne l’insulamento, l’aerazione o dar subito da mangiare alle api

Nel contesto del progetto, i sensori saranno simulati tramite l’inserimento di dati fittizi, aventi valori rispettivamente in celsius a due cifre decimali (eg 35.50), la percentuale in formato da 0 ad 1 (eg 0.505 sarà 50.5%) e un semplice segnale che ritorni 1 quando c’è ancora liquido e 0 quando non lo rileva più.

## Funzionalità richieste dal cliente (2.2)

* Gestione emergenze tramite apicoltura intelligente

Un sistema che permetta al cliente di caricare dei dati ottenuti tramite sensore al server che potrà poi analizzare i dati e mostrare le criticità rilevate, specificando tipologia, i valori e un identificativo

## Elenco funzionalità (3.2)

## FU13 (nuova): Attori coinvolti: Beekeper e Sensori Hive

Nome: Smart Beehive

Descrizione: Una sezione dell’applicazione che, ricevuti i dati dei sensori, mostri a schermo le criticità rilevate dai sensori nelle ultime 48 ore separandole per tipologia.

Se possibile, utilizzare un sistema di API per il caricamento dati dei sensori.

## Requisiti Non Funzionali -> Portabilità (4.4)

PO02 -> “DATA COLLECTION SENSORI” -> I sensori devono produrre un output standard come definito nella descrizione dei sensori

## Descrizione casi d’uso (5.2)

Aggiungere UC24 usando le info aggiunte nell’elenco funzionalità

## Component Diagram

Prima di poter fare il component diagram, è necessario decidere l’architettura che il nostro sistema dovrà avere.

Una overview del sistema attuale è che avremo un *server centrale*, sul quale avremo le basi dati utente e i servizi di analisi, comunicante con due diverse tipologie di client:

* *client utente* dove gli utenti possono collegarsi e accedere alle informazioni e ai servizi di analisi
* *client sensore* che può caricare DATI SENSORIALI legati ad una arnia identificata dal suo nome (Una stringa di caratteri UNIVOCA per ogni apicoltore) assieme all’informazione legata all’utente che invia i dati.

Una opzione potrebbe essere usare l’archiettura SOA (service orchestration) with RESTful services, ovvero l’utilizzo di comunicazioni REST tra applicazione e server di dati in formato JSON o XML