



# SDD - System Design Document

<b>Riferimento</b>	<a href="#">2024 C10 SDD beehAlve V2.0</a> <a href="#">2024 C10 RAD beehAlve V2.0</a>
<b>Versione</b>	2.0
<b>Data</b>	21/01/2024
<b>Destinatario</b>	Prof.ssa Filomena Ferrucci
<b>Presentato da</b>	N. Gallotta, F. Festa, S. Valente, A. De Pasquale, L. Milione, C. Boninfante
<b>Approvato da</b>	Nicolò Delogu, Dario Mazza



Composizione Gruppo	
Francesco Festa	05121-13547
Nicolò Gallotta	05121-14639
Andrea De Pasquale	05121-14909
Sara Valente	05121-14627
Lorenzo Milione	05121-14107
Carmine Boninfante	05121-13309

## Cronologia Revisioni

Data	Versione	Descrizione	Autori
28/11/2023	0.1	Creazione Documento	Tutto il Team
28/11/2023	0.2	Decomposizione in Sottosistemi e Diagramma Architettuale	Francesco Festa Andrea De Pasquale
29/11/2023	0.3	Stesura Condizioni Limite	Lorenzo Milione
29/11/2023	0.4	Gestione dei Dati Persistenti - Schema ER	Sara Valente
29/11/2023	0.5	Controllo degli Accessi e Global Control Flow	Lorenzo Milione
29/11/2023	0.6	Stesura Architettura Software Corrente	Carmine Boninfante



<b>29/11/2023</b>	0.7	Stesura Introduzione	Nicolò Gallotta
<b>29/11/2023</b>	0.8	Servizi dei Sottosistemi	Francesco Festa Andrea De Pasquale
<b>30/11/2023</b>	0.9	Gestione dei Dati Persistenti - Schema Logico	Sara Valente
<b>01/12/2023</b>	0.10	Aggiunta Glossario	Lorenzo Milione
<b>01/11/2023</b>	0.11	Stesura Panoramica Architettura Software Proposta	Nicolò Gallotta
<b>01/12/2023</b>	0.12	Gestione dei Dati Persistenti - Dizionario dei Dati	Sara Valente
<b>01/12/2023</b>	0.13	Mapping HW/SW e Deployment Diagram	Carminé Boninfante
<b>03/12/2023</b>	0.14	Integrazione e Revisione	Tutto il Team
<b>12/12/2023</b>	1.0	Revisione Finale	Nicolò Delogu Dario Mazza
<b>21/01/2024</b>	2.0	Revisione e sistemazione finale	Tutto il Team



## **Sommario**

<b>Cronologia Revisioni.....</b>	<b>2</b>
<b>Sommario.....</b>	<b>4</b>
<b>1. Introduzione.....</b>	<b>5</b>
1.1 Obiettivo del sistema.....	5
1.2 Obiettivi di Design.....	6
1.2.1 Design Goals.....	6
1.2.2 Design Trade-off.....	8
1.3 Definizioni, acronimi e abbreviazioni.....	8
1.4 Riferimenti.....	10
1.5 Panoramica.....	10
<b>2. Architettura del Sistema corrente.....</b>	<b>11</b>
<b>3. Architettura del Sistema proposto.....</b>	<b>12</b>
3.1 Panoramica.....	12
3.2 Decomposizione in sottosistemi.....	13
3.3 Mapping hardware/software.....	17
3.4 Gestione dei dati persistenti.....	18
3.5 Controllo degli accessi e sicurezza.....	24
3.6 Controllo del flusso globale del sistema.....	26
3.7 Condizioni limite.....	27
<b>4. Servizi dei Sottosistemi.....</b>	<b>33</b>
4.1 Gestione Arnie.....	33
4.1.1 Gestione Dashboard.....	33
4.1.2 Gestione Interventi.....	34
4.1.3 Gestione Stato.....	35
4.2 Gestione Utente.....	36
4.2.1 Gestione Profilo.....	36
4.2.2 Gestione Abbonamento.....	37
4.3 Gestione Sensori.....	38
4.4 Previsione Anomalie.....	39
<b>5. Glossario.....</b>	<b>40</b>



## **1. Introduzione**

### **1.1 Obiettivo del sistema**

Il sistema che si vuole realizzare ha come obiettivo primario il supporto agli apicoltori, in maniera tale da svolgere i loro compiti e di effettuare interventi straordinari in modo semplice, rapido ed efficiente. Gli apicoltori che utilizzeranno tale sistema avranno a disposizione funzionalità per la gestione delle proprie arnie, per la pianificazione di interventi su di esse e, in particolare, per la prevenzione o mitigazione del CCD. Quest'ultima è la problematica principale a cui il sistema fornito vuole dare una soluzione, poiché rappresenta uno dei più pericolosi fenomeni che causano lo spopolamento degli alveari fino ad un loro totale abbandono, il che porta non solo ad un enorme danno all'economia dell'industria apistica ma anche ad un forte impatto negativo sull'ambiente circostante.

Il sistema è una web app che permetterà ad un apicoltore di registrare e monitorare ognuna delle sue arnie, e farà uso di un algoritmo di Machine Learning, sfruttando i dati registrati dall'apicoltore e aggiornati in tempo reale con l'aiuto di sensori IoT, per identificare anomalie nella salute delle arnie ed eventuali sintomi del CCD. Per mezzo di ciò, il sistema potrà notificare preventivamente l'apicoltore, fornendogli le funzionalità necessarie per pianificare interventi immediati o futuri.

Per poter usufruire dei servizi forniti dal sistema, l'apicoltore dovrà necessariamente eseguire una procedura di autenticazione, oltre a doversi sottoporre ad una sottoscrizione di un abbonamento mensile.



## 1.2 Obiettivi di Design

### 1.2.1 Design Goals

Rank	ID Design Goal	Descrizione	Categoria	Origine
10	DG_1	Il sistema deve inviare ad un apicoltore una notifica di avvertimento entro 30 secondi dall'individuazione di un problema.	Performance	RNF4
9	DG_2	Il sistema deve registrare con successo una nuova arnia entro 3 secondi.	Performance	RNF5
6	DG_3	I sensori IoT devono aggiornare i parametri delle arnie in tempo reale, con una frequenza di misurazione di 1 ora.	Performance	RNF6
8	DG_4	Il sistema deve garantire al 98% degli apicoltori l'esecuzione delle operazioni senza commettere errori nel flusso di navigazione.	End User	RNF1
12	DG_5	Le interfacce devono garantire uniformità grafica e stilistica utilizzando una palette di colori predeterminata per rendere la navigazione familiare e coerente.	End User	RNF1
7	DG_6	Il sistema deve adottare una metodologia di addestramento per l'algoritmo di Machine Learning in maniera tale da ottenere una percentuale minore del 10% di falsi negativi.	Dependability	RNF2



<b>2</b>	DG_7	I sensori IoT devono fornire un servizio di misurazione dei parametri delle arnie, eccetto quando in manutenzione, continuativamente e senza interruzioni ogni giorno dell'anno, 24 ore su 24.	Dependability	RNF3
<b>3</b>	DG_8	I sensori IoT devono fare affidamento esclusivamente all'energia solare e rimanere attivi anche nei periodi notturni, riducendo al minimo l'impatto ambientale e contribuendo a un funzionamento a lungo termine del sistema.	Maintenance	RNF7
<b>5</b>	DG_9	Il sistema deve presentare un'architettura tale da poter supportare ogni sistema operativo e ogni tipologia di web browser.	Maintenance	RNF8
<b>4</b>	DG_10	Il sistema deve essere concepito in modo tale da agevolare la manutenzione senza richiedere un incremento significativo dei tempi e dei costi associati alla correzione di bug o problemi, in maniera tale da consentire agli sviluppatori di individuare, comprendere e risolvere gli eventuali problemi rilevati dopo il rilascio.	Maintenance	RNF9
<b>1</b>	DG_11	Il costo previsto per lo sviluppo del sistema ammonta a 50 ore/uomo.	Cost	Statement Of Work
<b>11</b>	DG_12	La manutenzione del sistema richiede un costo complessivo di 12700€/anno per la correzione di bug ed errori e per aggiornamenti al sistema.	Cost	Scope Statement Document

## 1.2.2 Design Trade-off

Trade-Off	Descrizione
<b>Scalabilità vs. Performance</b>	Il sistema dovrà prediligere la scalabilità alla performance, in quanto è necessario gestire l'aumento del carico di lavoro o delle risorse in modo efficace, come l'aumento del numero di arnie registrate e salvate nel database. Inoltre, è necessario permettere ad ogni apicoltore di accedere alla piattaforma da qualunque dispositivo e sistema operativo.
<b>Sicurezza vs. Performance</b>	Il sistema dovrà prediligere la sicurezza alla performance poiché, col fine di salvaguardare i dati degli apicoltori, potrebbe richiedere un tempo di risposta maggiore.
<b>Affidabilità vs. Performance</b>	Il sistema dovrà prediligere l'affidabilità alla performance, in quanto dovrà garantire un corretto funzionamento degli algoritmi utilizzati, come il modello di Machine Learning.
<b>Sicurezza vs. Usabilità</b>	Il sistema dovrà prediligere la sicurezza all'usabilità, fornendo ad esempio meccanismi come autenticazione forte e crittografia piuttosto che procedure e tempi di accesso più semplici e veloci possibile.
<b>Costi di sviluppo vs. Usabilità</b>	Il sistema dovrà rientrare nei costi di budget prefissati a discapito dell'usabilità per gli apicoltori, che comporta l'assenza di diverse funzionalità avanzate e la presenza di una piattaforma più semplice.

## 1.3 Definizioni, acronimi e abbreviazioni

- **Definizioni.**

- **Apicoltore non abbonato:** apicoltore registrato che non ha ancora effettuato una sottoscrizione ad uno degli abbonamenti disponibili;
- **Apicoltore:** apicoltore registrato e sottoscritto ad un abbonamento, che ha accesso ai servizi forniti dal sistema;
- **Sensore:** dispositivo IoT utilizzato dall'apicoltore per tenere sotto controllo le arnie, di cui ognuna è provvista;
- **Anomalia:** stato in cui uno o più valori di un'arnia sono fuori scala;





- **Colony Collapse Disorder:** sindrome che comporta lo spopolamento delle arnie, dovute alla morte o all'abbandono delle api che lo abitavano;
- **Intervento:** processo effettuato sulle arnie con lo scopo di migliorare la salute di un'arnia, correggere un'anomalia o prevenire il Colony Collapse Disorder;
- **Design Goals:** obiettivi di design che il sistema proposto deve soddisfare;
- **Design Trade-Off:** preferenze e compromessi tra Design Goals in conflitto;
- **Sottosistema:** una frazione del sistema e dei servizi da esso fornito, costituita da un certo numero di classi del dominio della soluzione;
- **Architettura:** la struttura complessiva del sistema, che definisce la disposizione e le relazioni tra i suoi componenti principali.
- **Acronimi.**
  - **CCD:** Colony Collapse Disorder;
  - **SDD:** System Design Document;
  - **ODD:** Object Design Document;
  - **RAD:** Requirements Analysis Document;
  - **DG:** Design Goal;
  - **DT:** Design Trade-Off;
  - **UC:** Use Case;
  - **SU:** Start Up;
  - **SD:** Shut Down;
  - **FA:** Failure;
  - **IoT:** Internet of Things;
  - **UI:** User Interface;
  - **HTTP:** HyperText Transfer Protocol;
  - **DBMS:** Database Management System;
  - **API:** Application Programming Interface.



## 1.4 Riferimenti

- *Challenges in the development of Precision Beekeeping* - Aleksej Zacepins et al.
- *A metagenomic survey of microbes in honey bee colony collapse disorder* - Cox-Foster et al.
- *A Survey of Honey Bee Colony Losses in the U.S.* - Dennis vanEngelsdorp et al.
- *Object-Oriented Software Engineering (Using UML, Patterns, and Java)* - Bernd Bruegge & Allen H. Dutoit
- RAD beehAlve
- [Io-Bee](#)
- [BeeKeepPal](#)
- [BeeKing](#)
- [osBeehives](#)

## 1.5 Panoramica

Questo documento è composto da quattro sezioni:

- **Introduzione.** Viene brevemente descritto lo scopo del sistema, gli obiettivi di design che esso vuole raggiungere e i relativi trade-off. Si conclude riportando le definizioni e gli acronimi utilizzati all'interno del documento e i vari riferimenti utilizzati.
- **Architettura Software corrente.** Descrive lo stato attuale dell'architettura del software attualmente utilizzata, se esiste.
- **Architettura Software proposta.** Descrive come il sistema verrà sviluppato, definendo come verrà partizionato in sottosistemi e il mapping hardware/software tra essi. Inoltre, descrive come verranno gestiti i dati persistenti, il controllo di accesso ai dati e alle funzionalità, il flusso di controllo delle operazioni dell'intero sistema e le sue condizioni limite.
- **Glossario.** Riporta una lista dei termini utilizzati nel documento con relative descrizioni.



## **2. Architettura del Sistema corrente**

Attualmente non esiste un'architettura software che con un'unica applicazione aiuti gli apicoltori ad effettuare i vari controlli. Infatti, essi sono supportati da architetture software che offrono servizi frammentati e con un basso livello di compatibilità tra di loro. Alcuni di questi offrono solo l'installazione e l'utilizzo di sensori di IoT, mentre altri mettono a disposizione solo un IA.



## **3. Architettura del Sistema proposto**

### **3.1 Panoramica**

Il sistema proposto è una piattaforma web di supporto agli apicoltori, con lo scopo di migliorare e semplificare la gestione totale delle loro arnie e la salvaguardia della salute di queste ultime. Più precisamente, il nostro sistema è messo a disposizione ai singoli apicoltori, dove ognuno di essi potrà effettuare il Log-In sul proprio account e sottoscrivere ad un abbonamento per accedere alle varie funzionalità che il sistema offre. Fatto ciò, l'apicoltore potrà monitorare da remoto le sue arnie avendo a disposizione una serie di parametri.

I parametri in questione sono i seguenti:

- Peso;
- Rumore;
- Temperatura dell'arnia;
- Temperatura dell'ambiente;
- Umidità;
- Umidità dell'ambiente;
- Presenza della regina.

Questi parametri sono messi a disposizione da un sensore IoT, posizionato su ogni arnia, che effettua misurazioni costanti e in tempo reale, che vengono successivamente elaborate da un algoritmo di Intelligenza Artificiale al fine di prevedere possibili anomalie, o peggio il CCD.

Il sistema provvederà a inviare notifiche all'apicoltore nel momento in cui si verifica un'anomalia che possa, a lungo termine, recare un danno alla specifica arnia.

L'apicoltore potrà rispondere al problema pianificando un intervento, al fine di gestire la situazione e di riportare i parametri nella norma.

L'architettura scelta per questo sistema è la Three-Tier, in quanto ottima e vantaggiosa per applicazioni web come la nostra per i seguenti motivi:

- Consente una chiara separazione delle responsabilità tra i diversi livelli. Il livello di interfaccia (front-end), la logica dell'applicazione (backend), e il livello di storage (database) operano in modo indipendente, facilitando la manutenzione e l'aggiornamento del sistema;
- Permette una buona suddivisione del lavoro fra i membri del team che si concentrano su diversi aspetti;



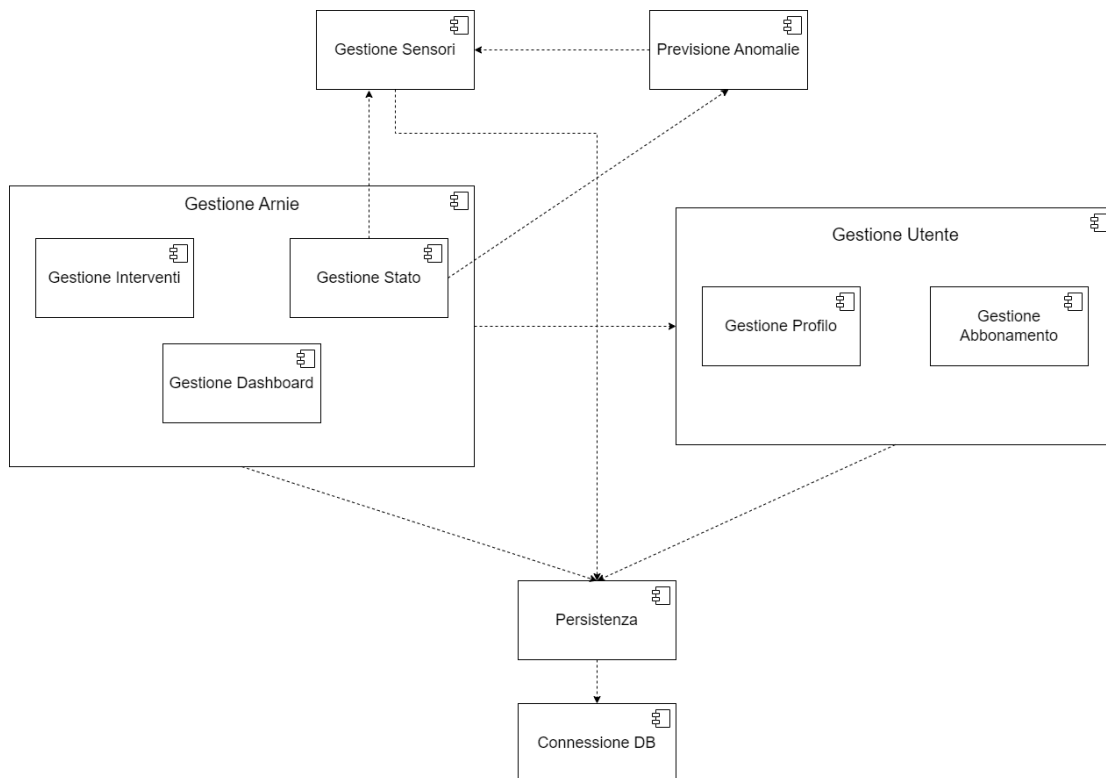
- Contribuisce alla sicurezza dell'applicazione. Il livello di presentazione, il livello di business e il livello di dati possono essere gestiti e protetti in modo indipendente, riducendo la probabilità di vulnerabilità.

### **3.2 Decomposizione in sottosistemi**

Per garantire una corretta suddivisione delle responsabilità e delle competenze, nonché un basso accoppiamento e un'elevata coesione, sono stati individuati i seguenti sottosistemi:

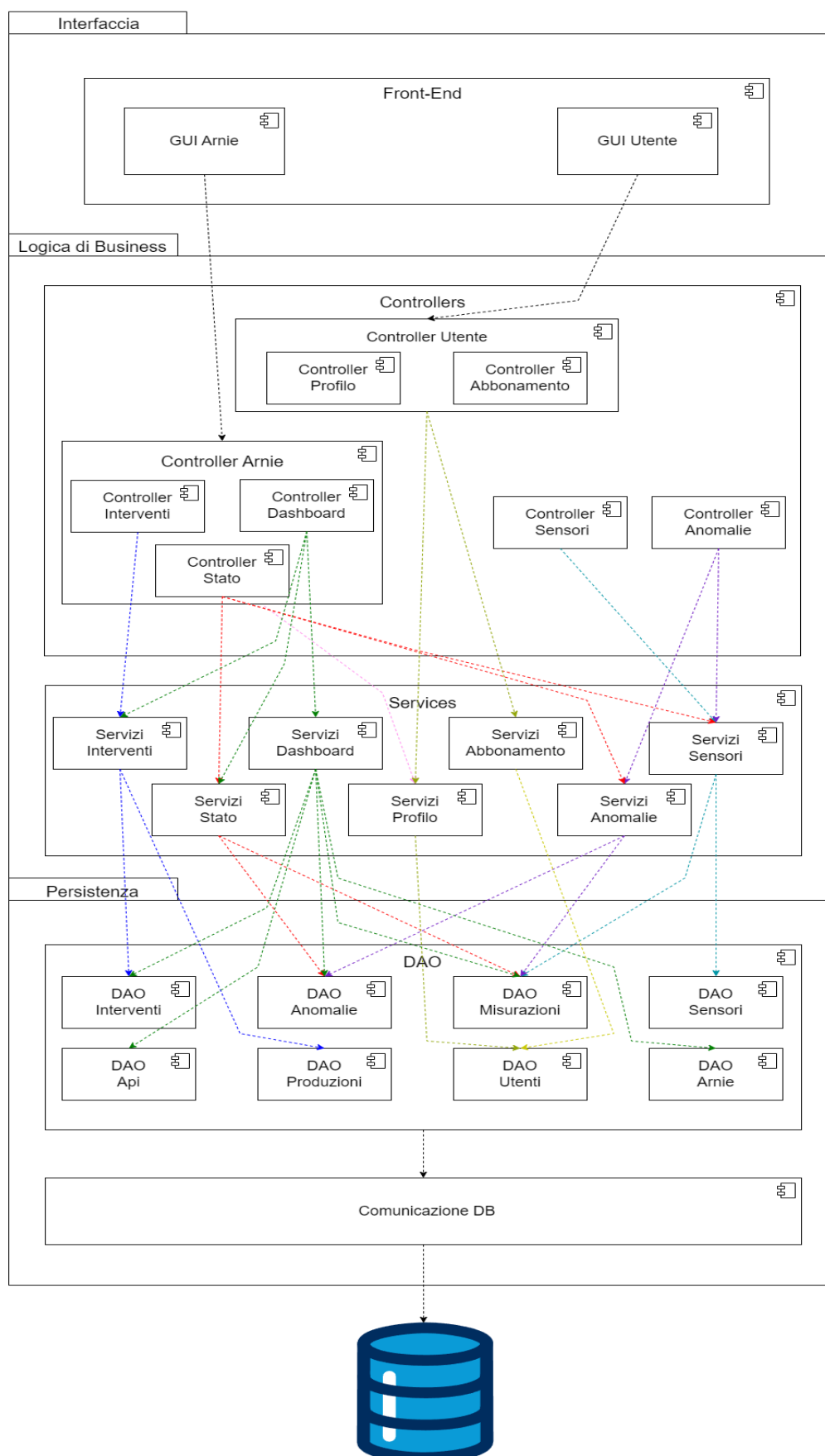


- **Gestione Arnie:** si occupa della gestione delle arnie di un apicoltore ed è diviso nei seguenti sottosistemi:
  - **Gestione Dashboard:** si occupa delle operazioni di inserimento, rimozione e modifica delle arnie nonché della visualizzazione di queste all'interno della GUI.
  - **Gestione Interventi:** si occupa della gestione degli interventi programmati, come la loro calendarizzazione e l'avviso di questi nella data prevista. Inoltre si occupa del caricamento della produzione dell'apicoltore.
  - **Gestione Stato:** si occupa di formattare lo stato di salute delle arnie in un formato human readable e della produzione di report dettagliati su questo. Inoltre avvisa l'apicoltore in caso di anomalie all'interno di un'arnia.
- **Gestione Utente:** si occupa della gestione degli account degli apicoltori all'interno della piattaforma ed è diviso nei seguenti sottosistemi:
  - **Gestione Profilo:** gestisce il profilo degli apicoltori e le operazioni basilari di gestione dell'account, come login e registrazione.
  - **Gestione Abbonamento:** gestisce i pagamenti mensili della sottoscrizione al servizio attraverso il sistema esterno "PayPal" ed eventuali avvisi sulla scadenza.
- **Gestione Sensori:** si occupa di ottenere i dati dai sensori di IoT collegati alle Arnie, controllando eventuali errori e formattandoli per passarli agli altri sottosistemi.
- **Previsione Anomalie:** si occupa di interpretare i dati dei sensori per individuare eventuali anomalie su di essi, come valori fuori dalle metriche di buona salute oppure potenziale presenza di CCD attraverso un modello di Machine Learning.
- **Persistenza:** si occupa di gestire i dati persistenti attraverso un database relazionale MySQL.
- **Connessione DB:** gestisce l'interazione con il database del sottosistema Persistenza e corrisponde a Spring JPA.



I sottosistemi sono organizzati in un'architettura Three-Tier:

- **Layer Interfaccia:** si occupa del front-end dell'applicazione, quindi contiene le varie GUI con cui l'apicoltore interagisce direttamente.
- **Layer Logica di Business:** si occupa della logica del sistema. In particolare è diviso in:
  - **Controllers:** contiene gli oggetti Control per gestire il flusso delle varie operazioni
  - **Services:** contiene gli oggetti Service che implementano la logica di business del sistema
- **Layer Persistenza:** si occupa dell'interazione con il Database. In particolare è diviso in:
  - **DAO:** Contiene gli oggetti DAO che permettono di accedere ed eseguire operazioni all'interno del Database
  - **Comunicazione DB:** è un'interfaccia tra l'applicazione e il database e si occupa della connessione con quest'ultimo





### 3.3 Mapping hardware/software

Il mapping hardware/software descrive come i sottosistemi sono assegnati all'hardware e ai vari componenti hardware standard.

La nostra web application, segue un architettura client-server che permette al client di interagire con il server tramite l'inoltro di richieste, per poi visualizzare le risorse corrispondenti.

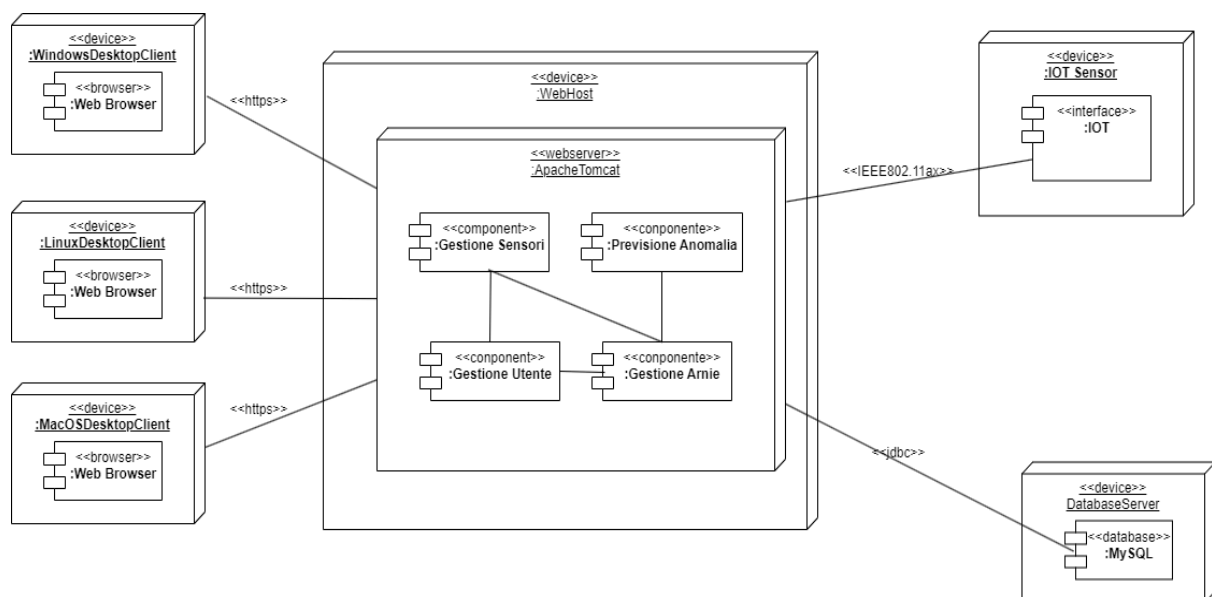
La comunicazione client-server avviene tramite un protocollo di ipertesti HTTP, usato come sistema per la comunicazione di informazioni sul web.

Il client richiede una qualsiasi macchina con un browser e una connessione alla rete internet. Il server richiede un dispositivo connesso alla rete internet e l'applicazione Server Tomcat, il quale comunicherà con il DB server tramite Spring JPA.

I dati persistenti relativi al dominio dell'applicazione verranno gestiti dal DB attraverso l'utilizzo di un Database Management System, che nel nostro caso sarà MySQL.

I sensori IoT comunicheranno con il server attraverso una connessione WiFi, che permetterà ai sensori di trasferire il flusso di dati al server.

Di seguito vi è l'UML Deployment Diagram per descrivere quanto detto in precedenza.





### 3.4 Gestione dei dati persistenti

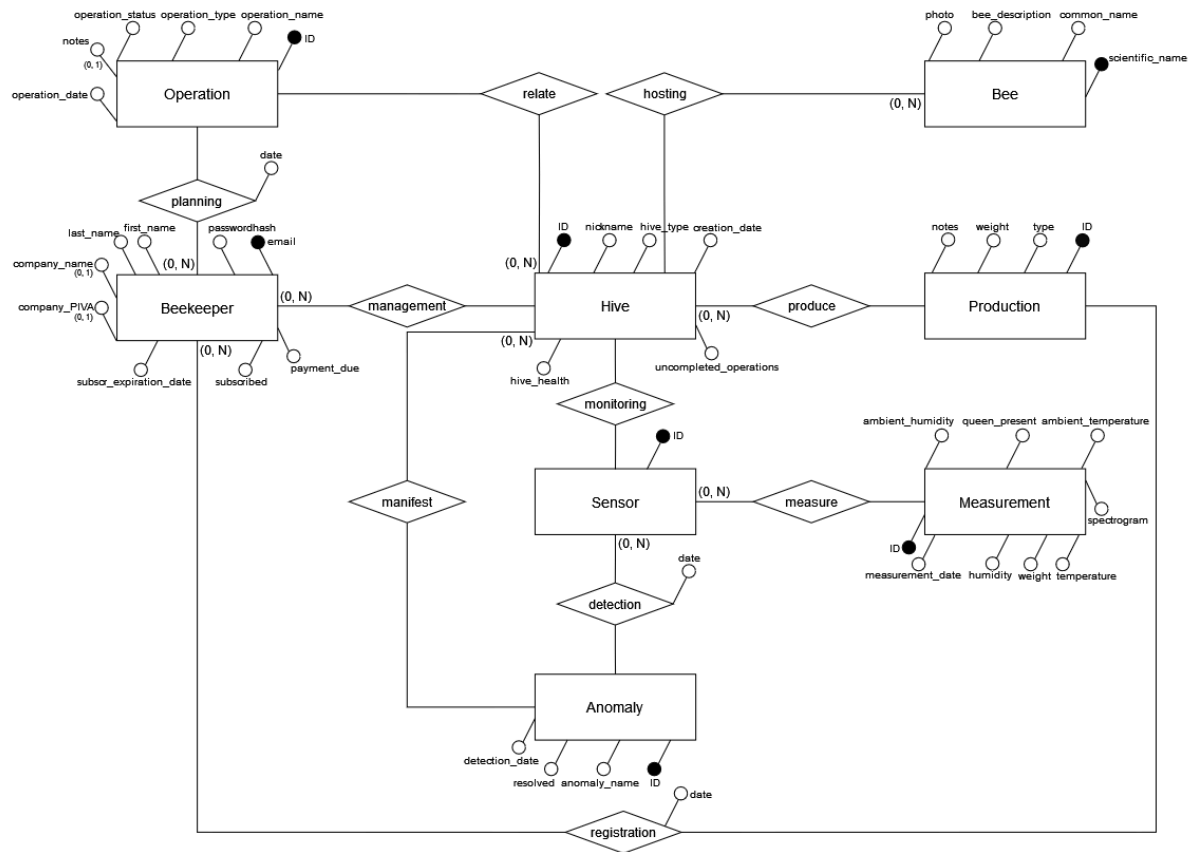
Per la gestione dei dati persistenti è stato utilizzato un database relazionale, affinché venga supportato l'accesso multiplo ai dati da parte di molteplici apicoltori e venga garantita la consistenza dei dati affiancata da una buona scalabilità.

E' stato scelto un DBMS (Database Management System) in quanto il suo utilizzo offre i seguenti vantaggi:

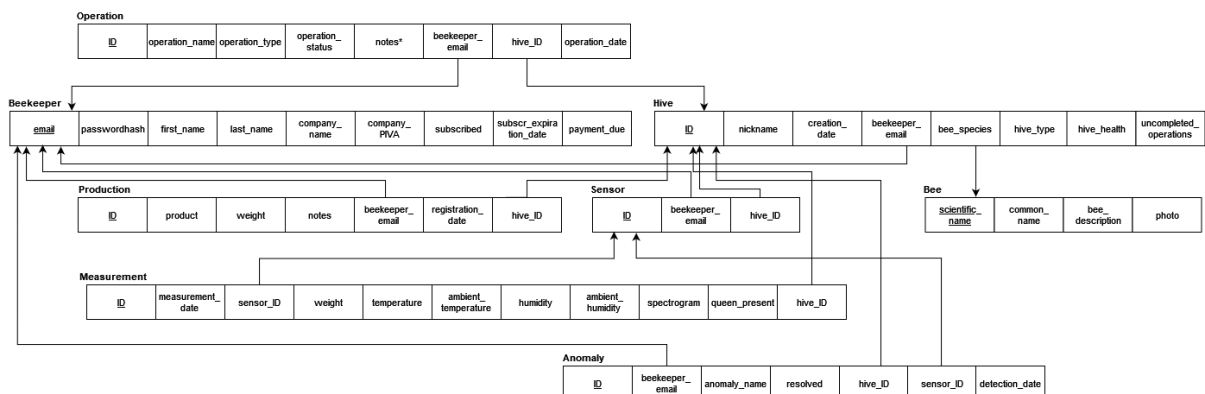
- **Affidabilità.** Un DBMS, nel caso in cui si manifesti un malfunzionamento di qualsiasi tipo (legato al sistema o errori hardware/software), protegge i dati grazie alle funzionalità di backup e ripristino, alle tecniche di ridondanza dei dati e al carattere atomico delle operazioni.
- **Sicurezza.** Un DBMS cura la privacy dei dati, grazie al controllo degli accessi, al meccanismo di autenticazione e alla gestione delle vulnerabilità.
- **Integrità dei dati.** Un DBMS prevede meccanismi che garantiscono la coerenza e l'accuratezza delle informazioni, grazie all'utilizzo delle chiavi primarie e delle chiavi esterne.

Il DBMS che è stato selezionato per il sistema è MySQL, che garantisce i vantaggi citati precedentemente.

## Diagramma Entità-Relazione



## Schema logico





## Dizionario dei dati

Nome entità	Hive		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
ID	Int	primary key	not null
nickname	Varchar(50)		not null
hive_type	Varchar(30)		not null
creation_date	Date		not null
beekeeper_email	Varchar(50)	foreign key(Beekeeper)	not null
bee_species	Varchar(50)	foreign key(Bee)	not null
hive_health	int		not null
uncompleted_operation	boolean		not null



Nome entità	Beekeeper		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
email	Varchar(50)	primary key	not null
passwordhash	Varchar(100)		not null
first_name	Varchar(50)		not null
last_name	Varchar(50)		not null
company_name	Varchar(100)		
company_PIVA	Char(20)		unique
subscribed	Boolean		not null
subscr_expiration_date	Date		
payment_due	Double		not null

Nome entità	Bee		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
scientific_name	Varchar(50)	primary key	not null
common_name	Varchar(50)		not null
bee_description	Varchar(300)		not null
photo	Varchar(50)		not null



Nome entità	Production		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
ID	Int	primary key	not null
product	Varchar(50)		not null
weight	Double		not null
notes	Varchar(300)		not null
beekeeper_email	Varchar(50)	foreign key(Beekeeper)	not null
registration_date	Date		not null
hive_ID	Int	foreign key(Hive)	not null

Nome entità	Anomaly		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
ID	Int	primary key	not null
anomaly_name	Varchar(100)		not null
resolved	Boolean		not null
detection_date	DateTime		not null
sensor_ID	Int	foreign key(Sensor)	not null
hive_ID	Int	foreign key(Hive)	not null
beekeeper_email	Varchar(50)	foreign key(Beekeeper)	not null



Nome entità	Operation		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
ID	Int	primary key	not null
operation_name	Varchar(70)		not null
operation_type	Varchar(50)		not null
operation_status	Varchar(30)		not null
operation_date	DateTime		not null
notes	Varchar(300)		
hive_ID	Int	foreign key(Hive)	not null
beekeeper_email	Varchar(50)	foreign key(Beekeeper)	not null

Nome entità	Sensor		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
ID	Int	primary key	not null
hive_ID	Int	foreign key(Hive)	not null
beekeeper_email	Varchar(50)	foreign key(Beekeeper)	not null



Nome entità	Measurement		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
ID	Int	primary key	not null
sensor_ID	Int	foreign key(Sensor)	not null
hive_ID	Int	foreign key(Hive)	not null
measurement_date	DateTime		not null
weight	Double		not null
spectrogram	Varchar(35)		not null
temperature	Double		not null
ambient_temperature	Double		not null
humidity	Double		not null
ambient_humidity	Double		not null
queenPresent	Boolean		not null

### 3.5 Controllo degli accessi e sicurezza

Il programma "BeehAlve" offre agli apicoltori la possibilità di registrarsi e autenticarsi sulla piattaforma utilizzando un indirizzo e-mail o un nome utente insieme a una password. La sicurezza degli accessi è affidata al controllo della password e dell'indirizzo e-mail. Successivamente, viene visualizzata una tabella degli accessi per registrare e tenere traccia delle operazioni che l'apicoltore può eseguire:





Oggetti	Gestione utente	Gestione arnie
<b>Attori</b>		
<b>Apicoltore non registrato</b>	<ul style="list-style-type: none"> <li>• Registrazione</li> </ul>	
<b>Apicoltore registrato</b>	<ul style="list-style-type: none"> <li>• Login</li> <li>• Logout</li> <li>• Modifica dati personali</li> <li>• Modifica password</li> <li>• Visualizza piani abbonamento</li> <li>• Sottoscrizione abbonamento</li> </ul>	
<b>Apicoltore sottoscritto</b>	<ul style="list-style-type: none"> <li>• Login</li> <li>• Logout</li> <li>• Modifica dati personali</li> <li>• Modifica password</li> <li>• Modifica piano abbonamento</li> <li>• Cancellazione abbonamento</li> <li>• Visualizza piani abbonamento</li> <li>• Rinnova abbonamento</li> </ul>	<ul style="list-style-type: none"> <li>• Creazione arnia</li> <li>• Cancellazione arnia</li> <li>• Modifica arnia</li> <li>• Visualizzazione di tutte le arnie</li> <li>• Visualizzazione singola arnia</li> <li>• Visualizzazione arnie con interventi pianificati</li> <li>• Visualizzazione arnie con anomalie rilevate</li> <li>• Pianificazione intervento</li> <li>• Cancellazione intervento</li> <li>• Modifica intervento</li> <li>• Notifica intervento imminente</li> <li>• Visualizzazione storico interventi</li> <li>• Registrazione produzione</li> <li>• Visualizzazione produzione</li> <li>• Visualizzazione grafico temperatura dell'arnia</li> <li>• Visualizzazione grafico umidità dell'arnia</li> </ul>



		<ul style="list-style-type: none"><li>• Visualizzazione grafico peso dell'arnia</li><li>• Visualizzazione presenza della regina nell'arnia</li><li>• Scaricamento report di salute dell'arnia</li><li>• Notifica anomalia</li><li>• Salvataggio anomalia</li></ul>
--	--	--

### 3.6 Controllo del flusso globale del sistema

Il sistema "BeehAlve" rappresenta un ambiente interattivo in cui ogni funzionalità è avviata mediante un comando impartito dall'apicoltore tramite un'interfaccia grafica. Quando l'apicoltore desidera accedere e utilizzare una specifica funzionalità, può farlo attraverso l'interfaccia grafica, che seleziona il controllo corrispondente. Questa azione genera un evento gestito dal relativo handler. A sua volta, l'handler dirige il flusso degli eventi al sottosistema responsabile della logica di controllo e della gestione, che, successivamente, si collega ai servizi per la logica applicativa.

Per quanto riguarda i controlli sulle arnie e il rilevamento delle anomalie, il sistema utilizza sensori posizionati all'interno delle arnie, consentendo all'apicoltore di monitorare (temporizzando su base giornaliera) le variabili responsabili del CCD (Colony Collapse Disorder).

In virtù di questi requisiti e poiché si tratta di una web application, il sistema impiega un meccanismo di controllo del flusso di tipo procedure-driven, affiancato da meccanismi event-driven.



### 3.7 Condizioni limite

Nel seguente paragrafo vengono presentate le condizioni limite riguardanti l'inizializzazione, la terminazione e i possibili fallimenti del sistema.

Identificativo  UC_CL_1	Inizializzazione del sistema	Data	29/11/2023
		Vers.	0.00.001
		Autore	Lorenzo Milione
Descrizione	Il caso d’uso permette l’avvio del server.		
Attore Principale	Apicoltore  Vuole avviare il sistema.		
Attore Secondari	N/A		
Entry Condition	Il sistema è pronto per essere avviato.		
Exit Condition  On success	Il sistema viene avviato con successo.		
Exit Condition  On failure	Il sistema non viene avviato.		
FLUSSO DI EVENTI PRINCIPALE / MAIN SCENARIO			
1	Apicoltore	Esegue l’inizializzazione del sistema accedendo alla pagina opportuna.	
2	Sistema	Ricevuta la richiesta, la pagina viene avviata.	
Note		N/A	
Special Requirements		N/A	



Identificativo  UC_CL_2	Terminazione del sistema	Data	29/11/2023
		Vers.	0.00.001
		Autore	Lorenzo Milione
Descrizione	Il caso d’uso permette lo spegnimento del sistema.		
Attore Principale	Apicoltore  Vuole terminare l’utilizzo del sistema.		
Attore Secondari	N/A		
Entry Condition	Il sistema è pronto per essere arrestato.		
Exit Condition  On success	Il sistema viene arrestato con successo.		
Exit Condition  On failure	Il sistema non viene arrestato con successo.		
FLUSSO DI EVENTI PRINCIPALE / MAIN SCENARIO			
1	Apicoltore	Esegue l’arresto del sistema chiudendo in modo opportuno la pagina.	
2	Sistema	Ricevuta la richiesta, la pagina viene arrestata.	
Note		N/A	
Special Requirements		N/A	



Identificativo  UC_CL_3	Fallimento:  Fallimento del sistema	Data	29/11/2023
		Vers.	0.00.001
		Autore	Lorenzo Milione
Descrizione	Il caso d’uso definisce il comportamento del Sistema in caso di errore del server.		
Attore Principale	Apicoltore		
Attore Secondari	N/A		
Entry Condition	Il sistema viene terminato senza preavviso.		
Exit Condition On success	Il sistema viene riavviato con successo.		
Exit Condition On failure	Il sistema non viene riavviato con successo.		
FLUSSO DI EVENTI PRINCIPALE / MAIN SCENARIO			
1	Apicoltore	Ripristina la configurazione del sistema ad uno stato precedente a quello di errore.	
2	Sistema	Include UC_CL_1	
Note		N/A	
Special Requirements		N/A	



Identificativo  UC_CL_4	Fallimento:  Errore di accesso ai dati persistenti	Data	29/11/2023
		Vers.	0.00.001
		Autore	Lorenzo Milione
Descrizione	Il caso d’uso descrive il comportamento del sistema in presenza di errori relativi ai dati persistenti.		
Attore Principale	Apicoltore		
Attore Secondari	N/A		
Entry Condition	Il sistema non ha modo di accedere ai dati persistenti.		
Exit Condition  On success	Il sistema risolve i problemi relativi ai dati persistenti.		
Exit Condition  On failure	Il sistema non risolve i problemi relativi ai dati persistenti.		
FLUSSO DI EVENTI PRINCIPALE / MAIN SCENARIO			
1	Sistema	Apicoltore viene avvisato tramite messaggio di errore dell'impossibilità di accesso ai dati persistenti.	
2	Sistema	Ripristina l’accessibilità o la consistenza dei dati.	
3	Apicoltore	Include UC_CL_1	
Note		N/A	
Special Requirements		N/A	



Identificativo  UC_CL_5	Fallimento:  Interruzione Alimentazione	Data	29/11/2023
		Vers.	0.00.001
		Autore	Lorenzo Milione
Descrizione	Si verifica un errore non previsto all'alimentazione.		
Entry Condition	Il sistema viene arrestato senza preavviso a causa di un errore nell'alimentazione.		
FLUSSO DI EVENTI PRINCIPALE / MAIN SCENARIO			
1	Non sono previsti metodi per il ripristino del sistema allo stato precedente al fallimento.		
2	Include UC_CL_1		
Note	N/A		
Special Requirements	N/A		



Identificativo  UC_CL_6	Fallimento:  Errore Critico Sensori	Data	29/11/2023
		Vers.	0.00.001
		Autore	Lorenzo Milione
Descrizione	Si verifica un errore critico dell’hardware.		
Attore Principali	Apicoltore		
Entry Condition	Il sistema ha riscontrato un errore nell’accesso ai sensori e notifica l’apicoltore.		
FLUSSO DI EVENTI PRINCIPALE / MAIN SCENARIO			
1	Non sono previste contromisure.		
Note	N/A		
Special Requirements	N/A		





## 4. Servizi dei Sottosistemi

### 4.1 Gestione Arnie

#### 4.1.1 Gestione Dashboard

Servizio	Descrizione	Interfaccia
<b>Creazione Arnia</b>	Questa funzionalità permette all'apicoltore di creare una nuova arnia.	ServiziDashboard
<b>Cancellazione Arnia</b>	Questa funzionalità permette ad un apicoltore di cancellare un'arnia esistente.	ServiziDashboard
<b>Modifica Arnia</b>	Questa funzionalità permette ad un apicoltore di modificare un'arnia esistente.	ServiziDashboard
<b>Visualizzazione di tutte le Arnie</b>	Questa funzionalità permette di visualizzare tutte le arnie di un apicoltore.	ServiziDashboard
<b>Visualizzazione singola Arnia</b>	Questa funzionalità permette di visualizzare nel dettaglio i parametri e le informazioni di una singola arnia di un apicoltore.	ServiziDashboard
<b>Visualizzazione Arnie con interventi pianificati</b>	Questa funzionalità permette di visualizzare tutte le arnie con interventi pianificati per il futuro di un apicoltore.	ServiziDashboard
<b>Visualizzazione Arnie con anomalie rilevate</b>	Questa funzionalità permette di visualizzare tutte le arnie con anomalie rilevate e non risolte di un apicoltore.	ServiziDashboard



## 4.1.2 Gestione Interventi

Servizio	Descrizione	Interfaccia
<b>Pianificazione Intervento</b>	Questa funzionalità permette ad un apicoltore di pianificare un intervento su un'arnia.	ServiziInterventi
<b>Cancellazione Intervento</b>	Questa funzionalità permette ad un apicoltore di cancellare un intervento pianificato su un'arnia.	ServiziInterventi
<b>Modifica Intervento</b>	Questa funzionalità permette ad un apicoltore di modificare un intervento pianificato su un'arnia.	ServiziInterventi
<b>Notifica Intervento imminente</b>	Questa funzionalità notifica un apicoltore di un intervento imminente.	ServiziInterventi
<b>Visualizzazione storico Interventi</b>	Questa funzionalità permette ad un apicoltore di visualizzare lo storico degli interventi pianificati.	ServiziInterventi
<b>Registrazione produzione</b>	Questa funzionalità permette ad un apicoltore di registrare una produzione di una delle sue arnie.	ServiziInterventi
<b>Visualizzazione produzione</b>	Questa funzionalità permette ad un apicoltore di visualizzare la produzione delle sue arnie.	ServiziInterventi



### 4.1.3 Gestione Stato

Servizio	Descrizione	Interfaccia
<b>Visualizzazione grafico temperatura dell'Arnia</b>	Questa funzionalità permette di visualizzare il grafico delle temperature registrate nel tempo di un'arnia.	ServiziStato
<b>Visualizzazione grafico umidità dell'Arnia</b>	Questa funzionalità permette di visualizzare il grafico dell'umidità registrate nel tempo di un'arnia.	ServiziStato
<b>Visualizzazione grafico peso dell'Arnia</b>	Questa funzionalità permette di visualizzare il grafico del peso registrato nel tempo di un'arnia.	ServiziStato
<b>Visualizzazione presenza della regina nell'Arnia</b>	Questa funzionalità permette di controllare se la regina è presente all'interno di un'arnia.	ServiziStato
<b>Notifica anomalia</b>	Questa funzionalità notifica un apicoltore nel caso in cui venga rilevata un'anomalia all'interno di un'arnia.	ServiziStato
<b>Risoluzione Anomalia</b>	Questa funzionalità permette ad un apicoltore di impostare un'anomalia rilevata in un'arnia come risolta.	ServiziStato
<b>Salvataggio anomalia</b>	Questa funzionalità permette di salvare un'anomalia rilevata in un'arnia.	ServiziStato
<b>Scaricamento report di salute dell'Arnia</b>	Questa funzionalità permette di generare e scaricare un report di salute complessivo di un'arnia.	ServiziStato



## 4.2 Gestione Utente

### 4.2.1 Gestione Profilo

Servizio	Descrizione	Interfaccia
<b>Registrazione</b>	Questa funzionalità permette ad un apicoltore di registrarsi sulla piattaforma.	ServiziProfilo
<b>Login</b>	Questa funzionalità permette ad un apicoltore di poter accedere.	ServiziProfilo
<b>Logout</b>	Questa funzionalità permette ad un apicoltore di uscire dal profilo.	ServiziProfilo
<b>Modifica dati personali</b>	Questa funzionalità permette ad un apicoltore di modificare i dati del suo profilo.	ServiziProfilo
<b>Modifica password</b>	Questa funzionalità permette ad un apicoltore di modificare la propria password.	ServiziProfilo



## 4.2.2 Gestione Abbonamento

Servizio	Descrizione	Interfaccia
<b>Sottoscrizione abbonamento</b>	Questa funzionalità permette ad un apicoltore di sottoscrivere un abbonamento mensile.	ServiziAbbonamento
<b>Modifica piano abbonamento</b>	Questa funzionalità permette ad un apicoltore di cambiare il proprio piano di abbonamento.	ServiziAbbonamento
<b>Cancellazione abbonamento</b>	Questa funzionalità permette ad un apicoltore di cancellare il proprio piano di abbonamento.	ServiziAbbonamento
<b>Visualizza piani abbonamento</b>	Questa funzionalità permette di visualizzare tutti i possibili piani di abbonamento.	ServiziAbbonamento
<b>Rinnova abbonamento</b>	Questa funzionalità permette ad un apicoltore di rinnovare il proprio piano di abbonamento.	ServiziAbbonamento



## 4.3 Gestione Sensori

Servizio	Descrizione	Interfaccia
<b>Misurazione dati arnia</b>	Questa funzionalità permette di misurare a intervalli regolari le metriche di interesse di un'arnia attraverso un sensore	ServiziSensori
<b>Validazione misurazione dati arnia</b>	Questa funzionalità permette di validare i dati di una misurazione di un sensore, controllando eventuali errori di trasmissione	ServiziSensori
<b>Formattazione misurazione dati arnia</b>	Questa funzionalità permette di formattare i dati misurati da un sensore in un formato più facilmente gestibile.	ServiziSensori
<b>Salvataggio misurazione dati arnia</b>	Questa funzionalità permette di salvare i dati di una misurazione	ServiziSensori
<b>Invio misurazione dati arnia</b>	Questa funzionalità permette di inviare i dati di una misurazione	ServiziSensori



## 4.4 Previsione Anomalie

Servizio	Descrizione	Interfaccia
<b>Controllo Temperatura</b>	Questa funzionalità prende i dati di una misurazione e controlla che la temperatura sia nei range di buona salute.	ServiziAnomalie
<b>Controllo Umidità</b>	Questa funzionalità prende i dati di una misurazione e controlla che l'umidità sia nei range di buona salute.	ServiziAnomalie
<b>Controllo Peso</b>	Questa funzionalità prende i dati di una misurazione e controlla che il peso sia nei range di buona salute.	ServiziAnomalie
<b>Controllo presenza regina</b>	Questa funzionalità permette di controllare se è presente l'ape regina attraverso il rumore all'interno di un'arnia	ServiziAnomalie
<b>Controllo CCD</b>	Questa funzionalità permette di calcolare se l'Arnia ha un possibile rischio di CCD attraverso un modulo di Machine Learning che prende i valori di una misurazione.	ServiziAnomalie



## 5. Glossario

<b>Termine</b>	<b>Descrizione</b>
<b>Piattaforma</b>	Tecnologia di base, sulla quale vengono sviluppate nuove tecnologie.
<b>DBMS</b>	Sistema software per consentire creazione, manipolazione e interrogazione di database.
<b>Design Goal</b>	Qualità sulle quali il sistema deve tenere il proprio focus.
<b>Mapping Hardware/Software</b>	Studio della connessione tra parti fisiche e logiche di cui si compone il Sistema.
<b>Sottosistema</b>	Controlla un particolare settore delle funzioni del sistema più complesso, quale, nel nostro caso, il dominio applicativo.
<b>Dati persistenti</b>	Dati che nel caso di modifiche conservano, e salvano, l'ultima versione di essi.
<b>Sensori</b>	Dispositivo utilizzato per tenere sotto controllo le arnie. Ogni arnia che possiede uno.
<b>Anomalia</b>	Stato in cui uno o più valori di un determinato hive sono fuori scala, i dati vengono prelevati da un sensor e analizzati dal sistema.
<b>Colony Collapse Disorder (CCD)</b>	Sindrome che caratterizza la scomparsa delle api da un'arnia di cui alcune morte ed altre che lo abbandonano. Viene predetto dall'algoritmo di Machine Learning controllando i parametri dell'arnia.