



Laurea Triennale in informatica-Università di Salerno
Corso di *Ingegneria del Software*- Prof.ssa F. Ferrucci, Prof F.Palomba



SDD System Design Document Environmental Intelligence for Agriculture

Riferimento	
Versione	1.0
Data	07/12/2022
Destinatario	Prof.ssa Filomena Ferrucci Prof. Fabio Palomba
Presentato da	C04
Approvato da	Carmine Laudato, Pierluigi Lambiase



Revision History

Data	Versione	Descrizione	Autori
25/11/2022	0.1	Aggiunta Architettura del sistema attuale	Benedetto Scala Francesco Maria Puca
26/11/2022	0.2	Aggiunta Introduzione dell'SDD	Francesco Fattorusso Gerardo Frino
27/11/2022	0.3	Aggiunta Design Goals	Francesco Fattorusso Maria Lombardi Gerardo Frino Benedetto Scala Francesco Maria Puca
27/11/2022	0.4	Aggiunta Panoramica della sezione	Francesco Fattorusso Gerardo Frino
27/11/2022	0.5	Aggiunta Decomposizione in sottosistemi	Francesco Maria Puca Benedetto Scala
28/11/2022	0.6	Aggiunta Diagramma Architetturale	Benedetto Scala Francesco Maria Puca
29/11/2022	0.6.5	Revisione Diagramma Architetturale	Benedetto Scala Francesco Maria Puca
30/11/2022	0.7	Aggiunta Mapping hardware/software	Francesco Maria Puca Francesco Fattorusso Maria Lombardi
2/12/2022	0.8	Aggiunta Gestione Dati Persistenti	Francesco Maria Puca Francesco Fattorusso Maria Lombardi Benedetto Scala Gerardo Frino
3/12/2022	0.9	Aggiunta Controllo globale del software	Francesco Maria Puca Gerardo Frino
3/12/2022	0.9.3	Aggiunta Condizioni limite	Francesco Maria Puca Gerardo Frino
4/12/2022	0.9.5	Aggiunta Controlli accesso e sicurezza	Francesco Fattorusso Benedetto Scala
6/12/2022	0.9.8	Aggiunta Servizi dei sottosistemi	Francesco Fattorusso Maria Lombardi Benedetto Scala Gerardo Frino



Laurea Triennale in informatica-Università di Salerno
Corso di *Ingegneria del Software*- Prof.ssa F. Ferrucci, Prof F.Palomba

			Francesco Maria Puca
6/12/2022	0.9.9	Aggiunta Glossario	Maria Lombardi Francesco Fattorusso
7/12/2022	1.0	Stesura e revisione completa	Maria Lombardi Francesco Fattorusso Gerardo Frino Francesco Maria Puca Benedetto Scala



Sommario

Revision History	2
1. Introduzione	5
1.1. Scopo Del Sistema	5
1.2. Obiettivi di Design (Design Goals)	5
1.3. Definizioni, acronimi e abbreviazioni	9
1.4. Riferimenti	9
1.5. Organizzazione del documento	9
2. Architettura del sistema attuale	9
3. Architettura del sistema proposto.....	10
3.1. Panoramica della sezione.....	10
3.2. Decomposizione in sottosistemi	10
3.3. Mapping Hardware/Software	13
3.4. Gestione dati persistenti	13
3.5. Controllo degli accessi e sicurezza	16
3.6. Controllo Globale Del Software	18
3.7. Condizioni Limite.....	18
3.7.1. Start-Up	18
3.7.2. Fallimento Durante Lo Start-Up	18
3.7.3. Shut-Down	18
3.7.4. Altri Casi Di Fallimento	18
4. Servizi Dei Sottosistemi.....	19
5. Glossario	23



1. Introduzione

1.1. Scopo Del Sistema

Lo scopo di ENIA è quello di semplificare le varie gestioni presenti all'interno di una azienda agricola, che in genere risultano macchinose e molto laboriose, grazie all'utilizzo di una piattaforma web.

Il sistema è gestito dal proprietario dell'azienda agricola e da uno o più suoi dipendenti, grazie all'assegnazione di ruoli specifici.

La componente principale del sistema consiste in una piattaforma che consenta alle aziende di gestire i terreni, eventuali impianti di irrigazione e tutto ciò che è connesso all'azienda stessa, infine abbiamo una componente aggiuntiva (IAgricoltura), messa a disposizione delle aziende che acquisteranno la licenza premium, basata su intelligenza artificiale che aiuterà nella gestione ottimale del proprio terreno e delle proprie colture.

1.2. Obiettivi di Design (Design Goals)

Nella presente sezione si andranno a presentare i Design Goals, ovvero le qualità sulle quali il sistema deve essere focalizzato, formalizzati esplicitamente così che qualsiasi importante decisione di design può essere fatta consistentemente seguendo lo stesso insieme di design goal. Seguendo le linee guida del libro Bernd Bruegge – Object Oriented Software Engineering i design goal sono stati suddivisi nelle seguenti categorie:

- **Performance:** includono i requisiti di spazio e velocità imposti sul sistema.
- **Dependability:** determinano quanto sforzo deve essere speso per minimizzare i fallimenti del sistema (crash, falle di sicurezza) e le loro conseguenze.
- **Maintenance:** determina quanto sforzo è necessario per modificare il sistema dopo il suo rilascio.
- **End User:** includono qualità che sono desiderabili dal punto di vista dell'utente, ma che non sono state coperte dai criteri di Performance e Dependability.

Ciascun design goal è descritto da:

- Rank, che ne specifica un valore di priorità.
- ID Design Goal, un identificatore univoco e un nome esplicativo.
- Descrizione, una descrizione del design goal.
- Categoria, ovvero la categoria di appartenenza del design goal.
- RNF di origine, ovvero il requisito non funzionale che lo ha generato.

Design Goals

Rank	ID Design Goal	Descrizione design goal	Categoria	Origine	Trade off
1	DG_1	La GUI da sviluppare dovrà permettere	End User	RNF_P_1	Avere un'interfaccia



	Interfaccia Grafica Responsive	all'utente finale di poter utilizzare il sistema su dispositivi con aspect ratio e risoluzioni diverse, mantenendo il layout grafico inteso dagli sviluppatori.			grafica responsive è di fondamentale importanza per l'utilizzo che l'utente andrà a fare del sistema. Dato che bisognerà essere in grado di utilizzare la piattaforma sia su dispositivi mobili sia su personal computer.
2	DG_7 Tempo di risposta	Il sistema dovrà garantire un tempo di risposta non superiore a 5 secondi	Performance	RNF_P_5	È molto importante far comprendere all'utente se il sistema risponde ai suoi input o meno: perciò, ha una priorità alta, a discapito del maggiore impegno di tempo lato UI necessario.
3	DG_4 Facilitazione lettura del codice	Il codice sorgente del sistema avrà un commento per ogni funzione presente esclusivamente in esso, al fine di facilitarne la comprensione per programmatori esterni al team di sviluppo	Maintenance	RNF_S_2	Quest'obiettivo comporta un impiego maggiore di lavoro per il team, che potrebbe invece essere usato per migliorare le funzioni del sistema. Tuttavia, dandogli alta priorità è possibile facilitare il raggiungimento di tutti gli altri design goals
4	DG_12 Facilità d'uso	Il sistema deve risultare facilmente comprensibile ed utilizzabile anche da un'utenza meno	End User	RNF_U_1	Ciò comporta che il sistema deve essere facile da usare, ed intuitivo da utilizzare per tutti gli utenti



		esperta, facendo uso delle “8 regole d’oro di Shneiderman” per il design delle interfacce grafiche.			
5	DG_5 Protezione dati utente	Il sistema garantirà che, nell’eventualità in cui si verifichi un fallimento dello stesso, i dati utente gestiti non verranno modificati irregolarmente	Dependability	RNF_A_2	Garantire la protezione dei dati utente comporta adoperare operazioni di persistenza a più passaggi, e perciò, un’interazione più lenta con il sistema. Perciò, nonostante l’elevata importanza, si dà priorità agli obiettivi prestazionali, in modo da non far percepire alcuna differenza all’utente.
6	DG_3 Performance Su Ambienti Differenti	Il sistema dovrà essere in grado di offrire lo stesso livello di prestazioni su macchine con hardware differenti	Performance	RNF_PE	Le performance, sono altresì importanti, ma sono posti subito dopo la facilità d’uso del sistema e l’intuitività dell’interfaccia grafica
7	DG_6 Velocità di registrazione	Un utente che non abbia mai usato il sistema prima d’ora dovrà essere in grado di registrare un account utente in meno di 15 minuti	End User	RNF_1	È necessario implementare prima le specifiche seguite di semplicità e facilità d’uso dell’interfaccia grafica, che comportano un aumento del lavoro effettuato dal team su di essa



					non insignificante. Ha di conseguenza priorità bassa.
8	DG_8 Gestione permessi del sistema	Il sistema dovrà separare gli utenti in diversi ruoli, al fine di consentire l'accesso alle sole funzionalità concesse per quel ruolo.	Dependability	RNF_A_3	Ciò comporta rallentamenti nelle operazioni utente, dovuti a più controlli. Tuttavia, è molto importante separare i ruoli.
9	DG_9 Disponibilità	Il sistema dovrà restare online per il 99,5% del tempo in un anno, quindi avrà un massimo di 44 ore di down	Performance	RNF_P_4	Ciò comporta che il sistema deve essere strutturato in modo essere fault tolerant ed essere anche resiliente ad attacchi. Quindi aumenta la complessità del sistema
10	DG_10 Manutenibilità	Il sistema dovrà essere programmato secondo la manutenibilità dello standard ISO/IEC 25010	Maintenance	RNF_S_1	Ciò comporta un aumento dei costi per quanto riguarda la manutenzione del sistema.
11	DG_2 Localizzazione	Il sistema dovrà adattarsi alla valuta e al fuso orario utilizzati dall'utente, in modo tale da facilitare non solo l'utilizzo delle funzionalità offerte (irrigation schedule, meteo), ma permettere anche un acquisto più consapevole delle licenze disponibili (standard e premium).	End User	RNF_L_1 RNF_L_2	Il raggiungimento di questo design goal è senza dubbio importante, dato che facilita l'utilizzo del sistema da parte dell'utente, ma non fondamentale come possono essere altri design goal.



1.3. Definizioni, acronimi e abbreviazioni

Sottosistema: un sottoinsieme dei servizi del dominio applicativo, formato da servizi legati da una relazione funzionale.

Design Goal: le qualità sulle quali il sistema deve essere focalizzato.

Dati Persistenti: dati che sopravvivono all'esecuzione del programma che li ha creati e che dunque vengono salvati.

Mapping Hardware/Software: studio della connessione tra parti fisiche e logiche di cui si compongono il sistema.

SDD: System Design Document

RAD: Requirements Analysis Document

1.4. Riferimenti

Questo documento ha come riferimenti i seguenti artefatti finalizzati precedentemente:

- Statement of work
- Requirements Analysis Document

1.5. Organizzazione del documento

Introduzione: Viene descritto in generale lo scopo del sistema, gli obiettivi di design che il sistema propone di raggiungere.

Architettura software corrente: Viene descritto lo stato attuale dell'architettura del software già presente.

Architettura software proposta: Viene descritto come il sistema sarà definito e partizionato in sottosistemi, il loro mapping Hardware/Software, la gestione dei dati persistenti. Verranno poi presentate la struttura dei singoli sottosistemi e le boundary conditions riguardanti l'intero sistema.

Glossario: Contiene la lista dei termini usati nel documento con annessa spiegazione.

2. Architettura del sistema attuale

Attualmente non esiste alcun sistema con un'architettura tale da permettere la distribuzione delle funzionalità offerte da EnIA. Sono disponibili, tuttavia, alternative che gestiscono singolarmente le funzioni di visualizzazione di dati meteo o di agenti inquinanti. Ciononostante, questi competitor non offrono un'architettura coesa in grado di supportare entrambe le funzionalità, oltre che ad estenderle con servizi che permettano la gestione dei terreni. Nel panorama attuale, inoltre, non esistono sistemi che consentano all'agricoltore di ricevere consigli inerenti all'amministrazione dei terreni, sfruttando le capacità dell'Intelligenza Artificiale.



3. Architettura del sistema proposto

3.1. Panoramica della sezione

Il sistema proposto è basato sullo stile architetturale Three Tier. Abbiamo scelto di definire il sistema in questo modo poiché quest'architettura e design pattern sono ottimi per lo sviluppo di un'applicazione di questo tipo, in particolare la suddivisione della logica applicativa da quella di presentazione migliora il riuso, la manutenzione e la leggibilità del codice.

Per lo sviluppo dell'applicazione lato front-end saranno utilizzati JavaScript per la gestione delle view e CSS e Bootstrap per l'aspetto grafico delle pagine.

Per il lato back-end sarà utilizzato Python. Per la gestione del database saranno usati:

- PyMongo per il collegamento al database.
- MongoDB per il database.

3.2. Decomposizione in sottosistemi

I sottosistemi individuati sono:

- **Registrazione:** Si occupa di gestire i vari tipi di registrazione:
- **Registrazione Farmer:** La registrazione di un utente di tipo farmer al sistema con acquisto di licenza (standard o premium).
- **Registrazione Con Codice Di accesso:** La registrazione di un utente di tipo Air Pollution Analyst o Irrigation Manager.
- **Autenticazione:** è responsabile delle funzionalità di login e logout
- **Ambiente Agricolo:** Si occupa della visualizzazione e della gestione dei dettagli di un ambiente agricolo e della relativa strumentazione (irrigatore) collegata ad esso.
- **Gestione Utente:** Permette ad un utente visualizzare e modificare i dati relativi al proprio account. Inoltre, consente l'eliminazione dell'account e l'assegnazione, la modifica e la revoca dei ruoli (Air Pollution Analyst e Irrigation Manager) da parte del farmer.
- **Gestione Pagamento:** Si occupa delle funzionalità di inserimento, modifica e rimozione di un metodo di pagamento di un farmer
- **Decision Intelligence:** Concede le funzionalità di inserimento, modifica, rimozione e visualizzazione suggerimenti dello scheduling di irrigazione
- **Gestione Eventi:** è responsabile della gestione degli alert e delle notifiche da inviare agli utenti, oltre che alla visualizzazione dello storico degli eventi estremi o non passati.
- **PyMongo/Flask:** Gestisce il layer di comunicazione tra il database e gli altri sottosistemi
- **Persistenza:** Gestisce la persistenza attraverso un database NoSQL (MongoDB)

Sono mostrate di seguito le dipendenze tra i sottosistemi attraverso un component diagram UML.

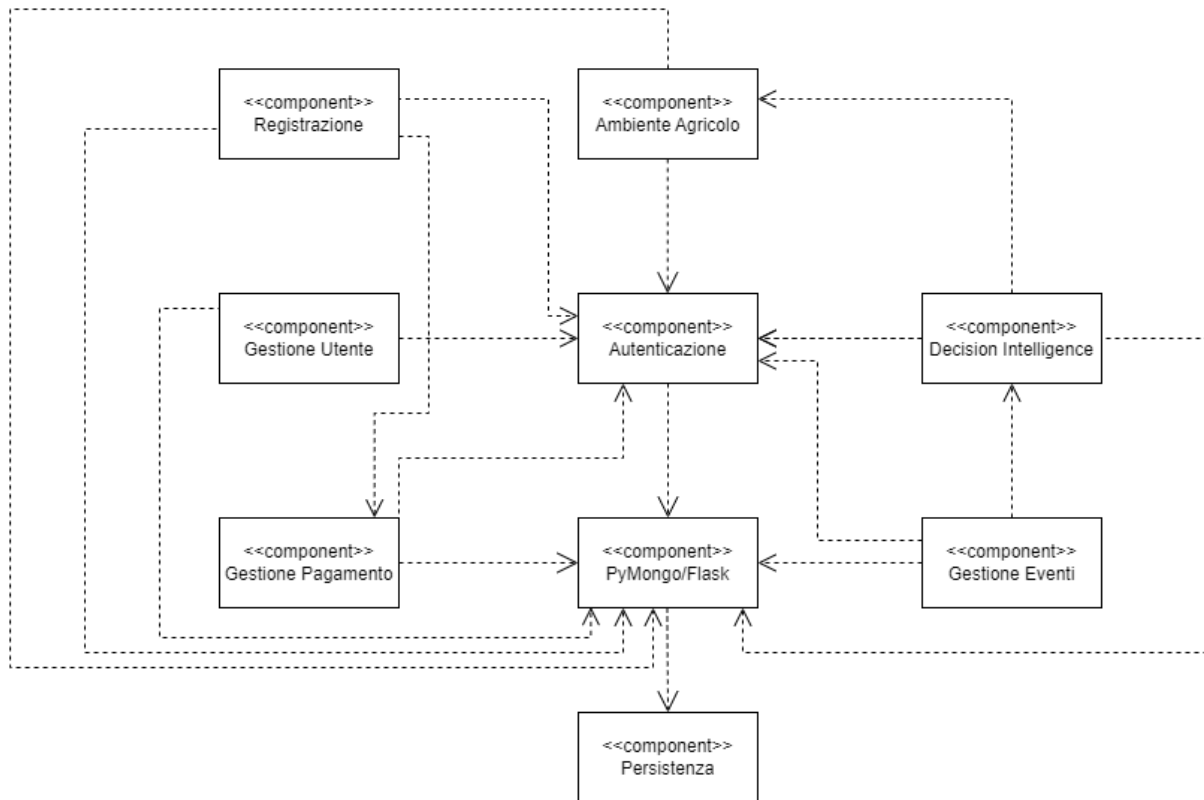
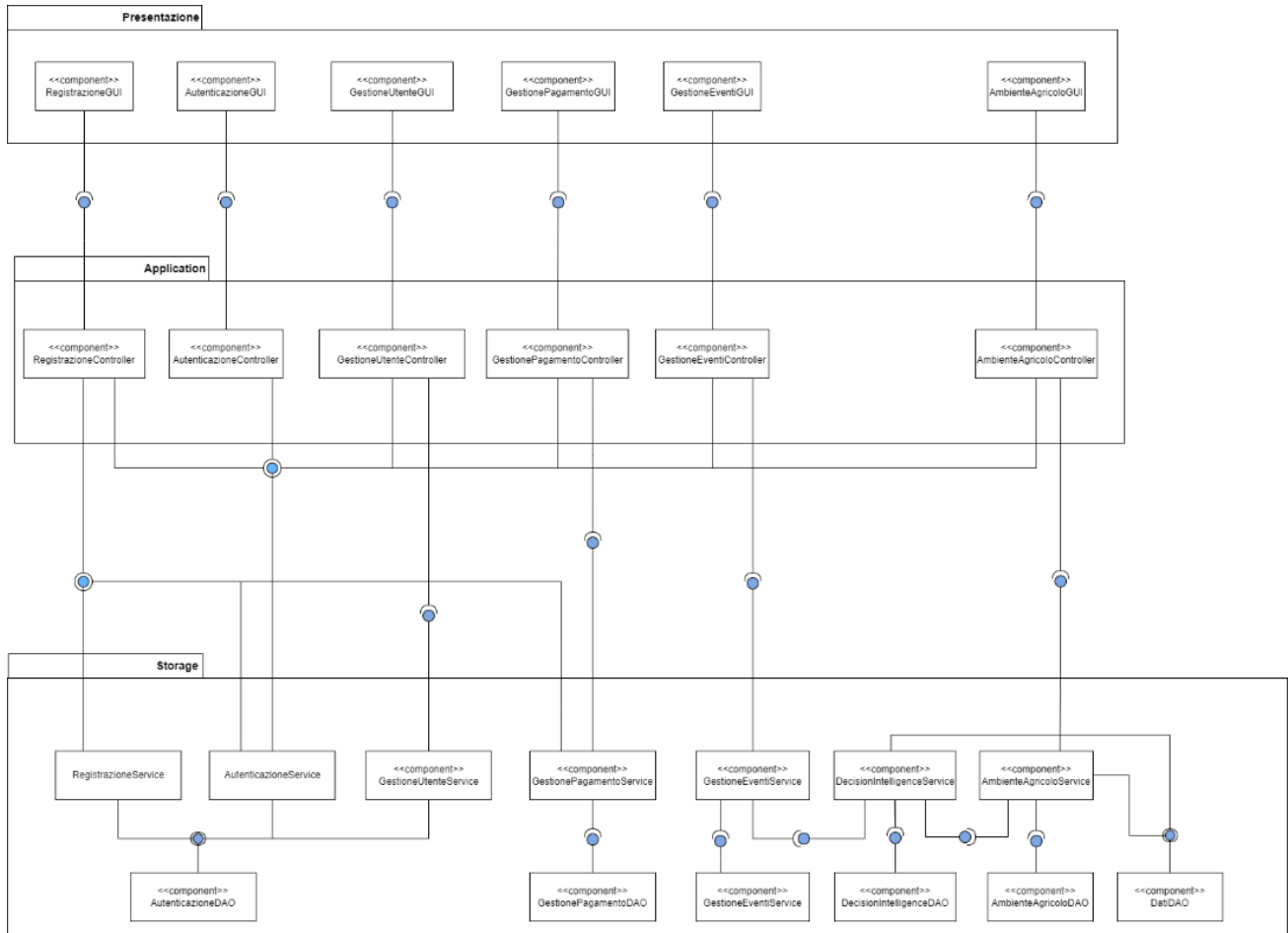


Diagramma Architettura



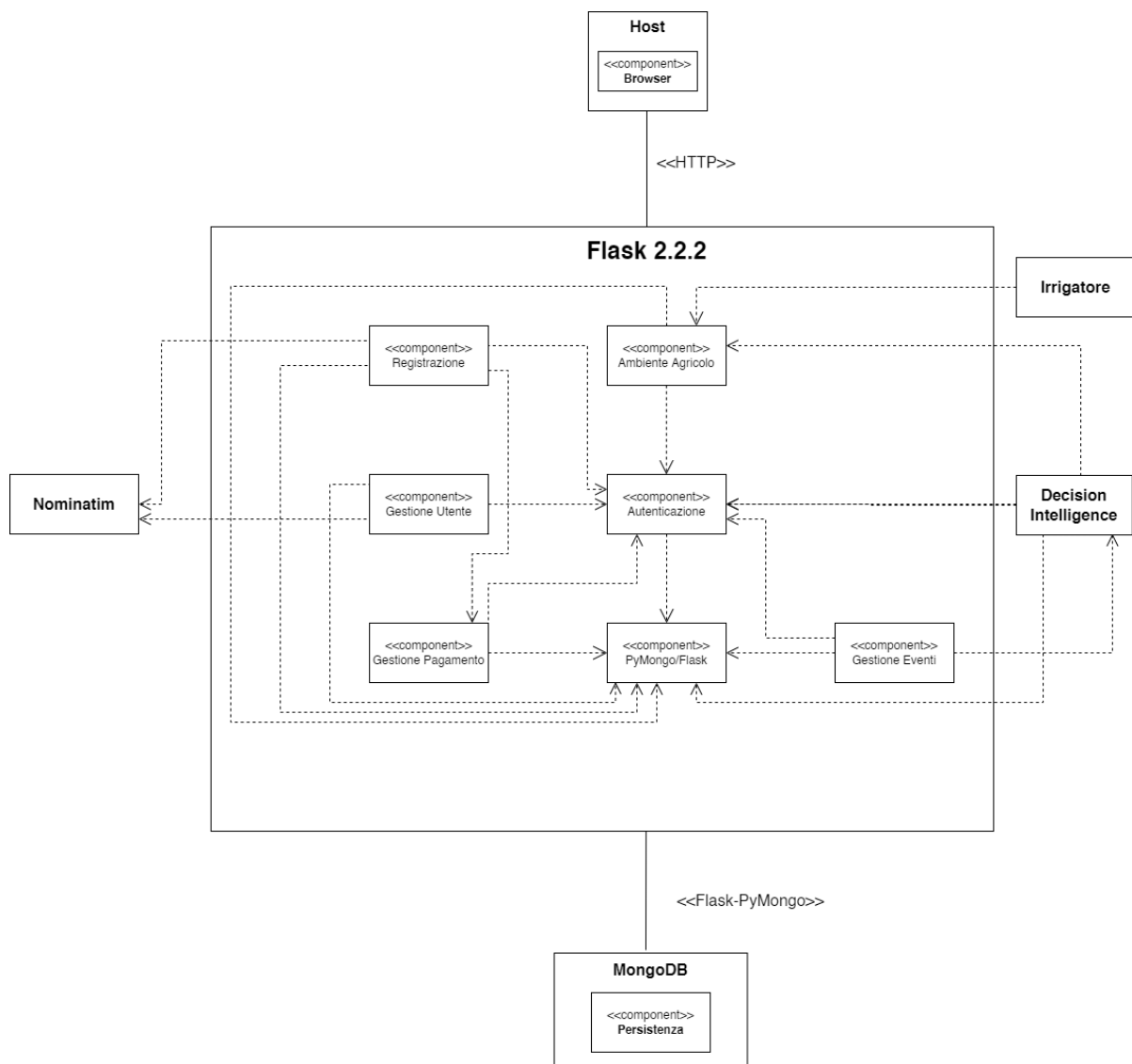
3.3. Mapping Hardware/Software

La piattaforma EnIA sarà sviluppata in modo da permettere a più macchine utente di connettersi attraverso il loro Browser ad una singola macchina server, la quale ospita la piattaforma.

Per la gestione dell'irrigazione, viene prevista la connessione di più irrigatori "Smart", che sono in grado di ricevere e mandare informazioni con il nodo su cui è ospitata la piattaforma attraverso il protocollo HTTP.

Per l'individuazione degli indirizzi esatti verrà utilizzato il servizio gratuito di OpenStreetMap Nominatim

Si trova in questa sezione un Deployment Diagram UML che illustra il mapping effettuato.
Gestione dati persistenti



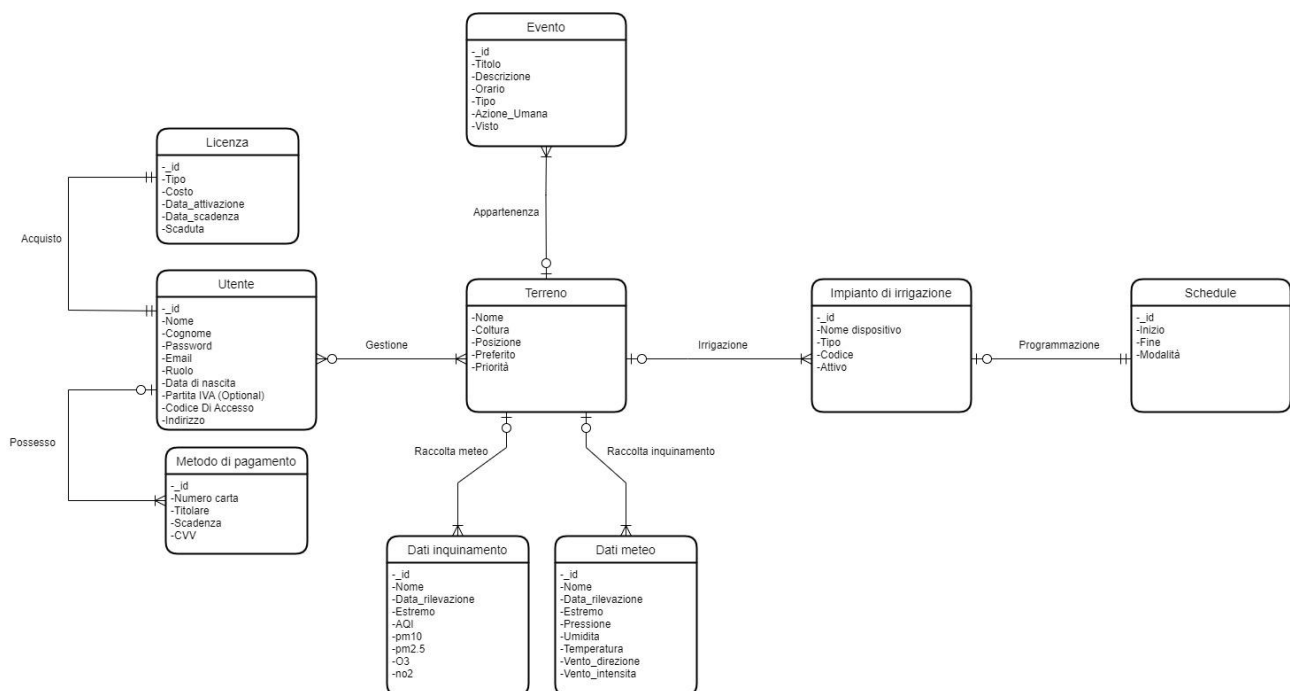
Si è scelto di realizzare la gestione dei dati persistenti attraverso l'uso di un DBMS non relazionale (NoSQL) orientato a documenti JSON-Like, quale MongoDB.

La principale motivazione per questa scelta è stata la necessità di un database ben integrato con il principale linguaggio di programmazione usato, Python, attraverso la libreria PyMongo.

Oltre a ciò, l'utilizzo di questi tool comporta numerosi vantaggi, tra cui:

- Rapida creazione di schemi, che permette il confronto tra varie soluzioni rilevanti al dominio applicativo in poco tempo.
- Velocità di query, fondamentale per soddisfare i design goals espressi precedentemente.
- Gestione integrata di transazioni ACID, la quale permette di far fronte ai requisiti espressi di affidabilità del sistema.

Lo schema logico seguente ed il dizionario dei dati relativo ad esso illustrano l'organizzazione della base di dati realizzata per il sistema.



Entità	Descrizione	Attributi	Identificatore
Licenza	È il tipo di licenza che un utente di tipo farmer può acquistare	_id, Tipo, Data_attivazione, Data_scadenza, Scaduta	_id
Utente	È l'attore che agisce sul sistema, può avere tre ruoli: farmer, air pollution analyst e irrigation manager	_id, Nome, Cognome, Password, E-mail, Ruolo, Data Di Nascita, Partita Iva	_id
Metodo Di Pagamento	Descrive gli attributi della carta di credito che un utente di tipo	_id, Numero Carta, Titolare, Scadenza CVV	_id



	farmer utilizza per acquistare una licenza		
Terreno	Un terreno è una località sulla mappa posseduta da un utente di tipo farmer, che richiede il monitoraggio in tempo reali dei dati delle proprie colture	_id, Nome, Coltura, Posizione, Preferito, Priorità	_id
Evento	Identificano gli avvenimenti di particolare importanza relativi ad un determinato terreno, che può richiedere o meno un'azione umana	_id, Nome_dispositivo, Tipo, Codice, Azione Umana, Visto	_id
Dati Inquinamento	Sono i dati degli agenti inquinanti relativi ad un determinato terreno	_id, Nome, Data_rilevazione, Estremo, AQI, pm10, pm2.5, O3, no2	_id
Dati Meteo	Sono i dati meteo relativi ad un determinato terreno	_id, Nome, Data_Rilevazione, Estremo, Pressione, Umidità, Temperatura, Vento_direzione, Vento_intensità	_id
Impianto di Irrigazione	Individua un impianto di irrigazione relativo ad un particolare terreno	_id, Nome Dispositivo, Tipo, Codice, Attivo	_id
Schedule	Individua ad un planning suggerito relativo ad un impianto di irrigazione	_id, Inizio, Fine, Modalità	_id

Relazione	Descrizione	Entità coinvolte	Attributi
-----------	-------------	------------------	-----------



Acquisto	Associa una licenza all'utente che l'ha acquistata.	Licenza, Utente	
Possesso	Associa un utente ai metodi di pagamento che ha usato.	Utente, Metodo di pagamento	
Appartenenza	Associa uno o più eventi all'ambiente agricolo a cui si riferiscono.	Terreno, Evento	
Gestione	Associa gli ambienti agricoli agli utenti che li gestiscono.	Terreno, Utente	
Raccolta Meteo	Associa un ambiente agricolo ai dati relativi alle sue condizioni metereologiche.	Terreno, Dati meteo	
Raccolta Inquinamento	Associa un ambiente agricolo ai dati relativi alla presenza di agenti inquinanti in esso.	Terreno, Dati inquinamento	
Irrigazione	Associa un ambiente agricolo agli impianti di irrigazione presenti in esso.	Terreno, Impianto di irrigazione	
Programmazione	Associa un impianto di irrigazione all'attività pianificata suggerita in un certo periodo di tempo, generata dal Sistema.	Impianto di irrigazione, Schedule	

3.4. Controllo degli accessi e sicurezza

In questa sezione è stata inserita la matrice degli accessi che tiene traccia degli oggetti a cui possono accedere gli attori.

	Farmer	Air Pollution Analyst	Irrigation Manager
Attori			
Oggetti			
Registrazione	RegistrazioneFarmer	RegistrazioneUtenteConCodiceDiAccesso	RegistrazioneUtenteConCodiceDiAccesso
Autenticazione	Login Logout	Login Logout	Login Logout



Laurea Triennale in informatica-Università di Salerno
Corso di *Ingegneria del Software*- Prof.ssa F. Ferrucci, Prof F.Palomba

Ambiente Agricolo	VisualizzaAmbienteAgricolo InserisciAmbienteAgricolo ModificaAmbienteAgricolo EliminaAmbienteAgricolo VisualizzaMeteo VisualizzaAgentiInquinanti InserisciIrrigatore ModificaIrrigatore EliminaIrrigatore VisualizzaListaIrrigatori VisualizzaIrrigatore AccendiIrrigatore SpegniIrrigatore StampaReport	VisualizzaAmbienteAgricolo VisualizzaAgentiInquinanti StampaReport	VisualizzaAmbienteAgricolo VisualizzaMeteo InserisciIrrigatore ModificaIrrigatore EliminaIrrigatore VisualizzaListaIrrigatori VisualizzaIrrigatore AccendiIrrigatore SpegniIrrigatore
Gestione Utente	VisualizzaAreaUtente ModificaAccount EliminaAccount VisualizzaLicenza VisualizzaListaUtenti AssegnaRuolo RimuoviRuolo	VisualizzaAreaUtente Modifica Account	VisualizzaAreaUtente Modifica Account
Gestione Pagamento	Visualizza Metodi Di Pagamento InserisciMetodoDiPagamento EliminaMetodoDiPagamento ModificaMetodoDiPagamento		
Gestione Eventi	VisualizzazioneStoricoEventi VisualizzazioneAlert VisualizzaEventoEstramo	VisualizzazioneAlert	VisualizzazioneAlert
Decision Intelligence	VisualizzaSuggerimentiIrrigazione VisualizzaSchedulingIrrigazione ModificaSchedulingIrrigazione Analisi Meteo	Analisi Meteo	VisualizzaSuggerimentiIrrigazione VisualizzaSchedulingIrrigazione ModificaSchedulingIrrigazione



3.5. Controllo Globale Del Software

Per la piattaforma EnIA, si è pensato di scegliere un flusso di controllo tale da permettere ad un utente di eseguire numerose operazioni attraverso un'interfaccia grafica localizzata nella pagina web ad esso servita. Si vuole inoltre prevedere la separazione delle responsabilità tra più componenti. La richiesta dell'utente viene processata come un evento che viene ascoltato dal corrispondente metodo della GUI, la quale delega l'esecuzione della funzionalità richiesta alla parte Service del sottosistema che la offre. Il sistema utilizza perciò un controllo di flusso globale di tipo Event-Driven.

3.6. Condizioni Limite

3.6.1. Start-Up

Al primo start-up della piattaforma EnIA, viene avviato il web server gestito da Flask, il quale fa partire il Servizio del database MongoDB. Da questo momento, il web server incomincia ad accettare connessioni da utenti, che accedono ad esso attraverso la pagina web principale, dalla quale possono effettuare il Login, inserendo le proprie credenziali (e-mail e password). Una volta effettuato l'accesso, l'utente può incominciare ad utilizzare le funzionalità messe a sua disposizione dalla piattaforma.

3.6.2. Fallimento Durante Lo Start-Up

Nel caso di un avvenuto errore durante la fase di avvio, il sistema salva una descrizione dello stesso in un file di log, assieme alla data e ora in cui si è verificato, in modo da facilitare per gli sviluppatori la risoluzione del fault che l'ha causato. Successivamente, interrompe le operazioni di avvio in corso, e ripete la procedura di avvio daccapo. Se viene rilevato un errore in fase di avvio del database, per evitare corruzione e/o inconsistenza dei dati, vengono eseguiti dei file per rigenerare il database a partire dall'ultimo backup effettuato.

3.6.3. Shut-Down

Nel momento che l'utente chiude la pagina della piattaforma sul suo browser, il sistema effettua automaticamente il Logout, ed annulla tutte le operazioni non ancora terminate effettuate dall'utente, in modo da garantire la consistenza dei dati.

3.6.4. Altri Casi Di Fallimento

Altri casi di fallimento da tenere in considerazione sono:

- Malfunzionamento del database, viene risolto attraverso backup periodici dello stesso, e l'utilizzo di transazioni per effettuare rollback in caso di incongruenze dei dati.
- Spegnimento improvviso della macchina server, in modo da garantire la consistenza dei dati a seguito del riavvio, verrà ripristinata una copia della base di dati effettuata precedentemente, ad intervalli periodici. Eventuali transazioni in corso saranno interrotte ed annullate automaticamente.
- Errori hardware, non vengono gestiti dal sistema in alcun modo.



- Malfunzionamento del software, qualunque operazione effettuata nella quale si sia rilevato un fallimento viene annullata e viene effettuato rollback della base di dati se esse ne è interessata.

4. Servizi Dei Sottosistemi

Si trovano in questa sezione i vari servizi dei sottosistemi finora elencati, le loro descrizioni e l'interfaccia a cui fanno riferimento.

Sottosistema Registrazione

Servizio	Descrizione	Interfaccia
Registrazione Farmer	Questa funzionalità permette ad un utente di registrarsi come farmer ed acquistare una licenza.	RegistrazioneService
Registrazione Con Codice Di Accesso	Questa funzionalità permette ad un utente di registrarsi con il codice di accesso fornito da un farmer.	RegistrazioneService

Sottosistema Ambiente Agricolo

Servizio	Descrizione	Interfaccia
Visualizza Ambiente Agricolo	Questa funzionalità permette ad un utente di visualizzare i dettagli di un ambiente agricolo, tra cui nome, posizione, tipo di coltura, priorità e se è tra i preferiti.	AmbienteAgricoloService
Inserisci Ambiente Agricolo	Questa funzionalità permette ad un farmer di inserire un nuovo ambiente agricolo sulla piattaforma.	AmbienteAgricoloService
Modifica Ambiente Agricolo	Questa funzionalità permette ad un farmer di modificare i dettagli di un ambiente agricolo.	AmbienteAgricoloService
Elimina Ambiente Agricolo	Questa funzionalità permette ad un farmer di eliminare un ambiente agricolo.	AmbienteAgricoloService
Visualizza Meteo	Questa funzionalità permette di visualizzare i dati meteo di un particolare ambiente agricolo.	AmbienteAgricoloService
Visualizza Agenti Inquinanti	Questa funzionalità permette di visualizzare un grafico degli	AmbienteAgricoloservice



	agenti inquinanti relativi ad un ambiente agricolo.	
InserisciIrrigatore	Questa funzionalità permette di inserire un Irrigatore in un ambiente agricolo.	AmbienteAgricoloService
ModificaIrrigatore	Questa funzionalità permette di modificare i dettagli un Irrigatore in un ambiente agricolo.	AmbienteAgricoloService
EliminaIrrigatore	Questa funzionalità permette di eliminare un irrigatore di un ambiente agricolo.	AmbienteAgricoloService
VisualizzaListIrrigatori	Questa funzionalità permette di visualizzare un alista degli irrigatori relativi ad un ambiente agricolo.	AmbienteAgricoloService
VisualizzaIrrigatore	Questa funzionalità permette di visualizzare i dettagli di un particolare irrigatore.	AmbienteAgricoloService
AccendiIrrigatore	Questa funzionalità permette di accendere un irrigatore.	AmbienteAgricoloService
SpegniIrrigatore	Questa funzionalità permette di spegnere un irrigatore.	AmbienteAgricoloService
StampaReport	Questa funzionalità permette di stampare un report degli agenti inquinanti relativi ad un ambiente agricolo.	AmbienteAgricoloService

Sottosistema Gestione Pagamento

Servizio	Descrizione	Interfaccia
VisualizzaMetodiDiPagamento	Questa funzionalità permette la visualizzazione dei metodi di pagamento.	GestionePagamentoService
InserisciMetodoDiPagamento	Questa funzionalità permette di inserire un nuovo metodo di pagamento.	GestionePagamentoService
ModificaMetodoDiPagamento	Questa funzionalità permette di modificare un metodo di pagamento già esistente.	GestionePagamentoService
EliminaMetodoDiPagamento	Questa funzionalità permette di eliminare un metodo di pagamento già esistente.	GestionePagamentoService

Sottosistema Gestione Eventi



Servizio	Descrizione	Interfaccia
VisualizzazioneStoricoEventi	Questa funzionalità permette di visualizzare lo storico di tutti gli eventi avvenuti nel sistema.	GestioneEventiService
VisualizzazioneAlert	Questa funzionalità permette di visualizzare tutti gli alert.	GestioneEventiService
VisualizzaEventoEstremo	Questa funzionalità permette di visualizzare tutti gli eventi estremi.	GestioneEventiService

Sottosistema Autenticazione

Servizio	Descrizione	Interfaccia
Login	Questa funzionalità permette di effettuare l'accesso al sistema tramite le proprie credenziali per sfruttare tutte le funzionalità che offre.	AutenticazioneService
Logout	Questa funzionalità permette di disconnettersi dal sistema.	AutenticazioneService

Sottosistema Gestione Utente

Servizio	Descrizione	Interfaccia
Visualizza Area Utente	Questa funzionalità permette ad un utente di visualizzare la propria area utenti.	GestioneUtenteService
Elimina Account	Questa funzionalità permette ad un utente di registrarsi con il codice di accesso fornito da un farmer.	<u>GestioneUtenteService</u>
Visualizza Licenza	Questa funzionalità permette ad un utente di visualizzare i dettagli della licenza acquistata.	GestioneUtenteService
Visualizza Lista Utenti	Questa funzionalità permette ad un utente di visualizzare una lista contenente tutti i dipendenti.	GestioneUtenteService



Assegna Ruolo	Questa funzionalità permette ad un utente di assegnare un ruolo ad un dipendente.	GestioneUtenteService
Rimuovi Ruolo	Questa funzionalità permette ad un utente di rimuovere il ruolo ad un dipendente.	GestioneUtenteService
Modifica Account	Permette di modificare i dati relativi alla propria area utente	GestioneUtenteService

Sottosistema Decision Intelligence

Servizio	Descrizione	Interfaccia
AnalisiMeteo	Permette ad un utente autorizzato di effettuare un'analisi dei dati meteo relativi ad un ambiente agricolo.	DecisionIntelligenceService
VisualizzaSuggerimentiIrrigazione	Permette ad un utente autorizzato di ottenere uno schedule per l'irrigazione di un ambiente agricolo, a seguito di un'analisi dei dati meteo rilevati.	DecisionIntelligenceService
VisualizzaSchedulingIrrigazione	Permette ad un utente autorizzato di visualizzare uno scheduling dell'irrigazione per determinato ambiente agricolo	DecisionIntelligenceService
ModificaSchedule	Permette ad un utente autorizzato di modificare le schedule generato.	DecisionIntelligenceService



5. Glossario

Questa sezione contiene le definizioni di tutte le sigle, termini, acronimi ed abbreviazioni usate nel RAD.

Termine	Definizione
Mock-up	Illustrazione grafica dell'interfaccia del sistema, durante un uso regolare da parte dell'utente delle sue funzionalità. Usato per dare al committente un'idea del prodotto desiderato.
Ambiente Agricolo	Qualsiasi terreno, posseduto da un proprietario, che sia usato per coltivazione agricola. Oggetto principale delle funzionalità del Sistema.
Inquinamento	Qualsiasi alterazione da parte di elementi inquinanti, aerei o terrestri, di un Ambiente Agricolo.
Azienda Agricola	Insieme di Ambienti Agricoli creati da un Farmer, i quali sono gestiti dal sistema attraverso le sue funzionalità.
Insight	Evento creato dal Decision Module del Sistema, per un Registered User, associato ad un Ambiente Agricolo. Può rappresentare un suggerimento di azione da parte del Registered User, oppure il risultato di un'analisi dei dati rilevati, richiesta dal Registered User.
Farmer	Responsabile di un'Azienda Agricola, in rispetto al Sistema. È il creatore degli Ambienti Agricoli, e può associare dei Dipendenti ad essi. Ha accesso a tutte le funzionalità del Sistema.
Air Pollution Analyst	Un tipo di Dipendente, associato ad un'Azienda Agricola da un Farmer. Si occupa del monitoraggio dei livelli di inquinamento, relativi ad uno o più Ambienti Agricoli.
Irrigation Manager	Un tipo di Dipendente, associato ad un'Azienda Agricola da un Farmer. Si occupa del monitoraggio delle risorse idriche e delle previsioni meteo, relative ad uno o più Ambienti Agricoli.
Registered User	Uno qualsiasi tra Farmer, Air Pollution Analyst o Irrigation Manager, che abbia effettuato la registrazione al Sistema. Possiede un Account Utente.
Priorità Ambiente Agricolo	Preferenza che può essere espressa da un Registered User, rispetto ad un Ambiente Agricolo che viene indicato come "Prioritario" (o "Preferito"). In tal caso, quando l'utente effettuerà l'accesso, verrà mostrata come prima pagina quella di dettaglio di quell'Ambiente Agricolo.
Report	Documento che contiene vari dati di un Ambiente Agricolo, idrici o di inquinamento, di cui è possibile generare un codice QR per scaricarlo successivamente.
Account Utente	Informazioni memorizzate su di un Registered User, contenenti sia quelle fornite al momento della registrazione, sia quelle inserite in



Laurea Triennale in informatica-Università di Salerno
Corso di Ingegneria del Software- Prof.ssa F. Ferrucci, Prof F.Palomba

	momenti successivi. Viene creato alla registrazione, ed è poi possibile, per il Registered User che lo possiede, cancellarlo o modificarlo.
Ruolo	Dato che indica il tipo di Registered User che si sta considerando (Farmer, Air Pollution Analyst o Irrigation Manager).
Alert	Evento che avvisa un Registered User di un avvenimento in un Ambiente Agricolo che richiede la sua attenzione, oppure il suo intervento diretto. Rappresentato graficamente in modo diverso da un Evento.
Evento	Messaggio inviato dal Sistema ad un Registered User, che lo informa dell'avvenimento di un processo in un Ambiente Agricolo, iniziato dal Sistema oppure dallo stesso Registered User. Viene conservato nello Storico Eventi, associato allo stesso Ambiente Agricolo.
Evento estremo	Fenomeno esterno al Sistema particolarmente violento o intenso, che può influenzare negativamente lo stato di uno o più Ambienti Agricoli.
Storico eventi	Tabella che contiene tutti gli Eventi associati ad un'Ambiente Agricolo dall'istante della sua creazione, e le loro proprietà.
Licenza	Contratto legale che, a seguito di pagamento, concede ad un Farmer l'utilizzo del Sistema per il periodo di tempo specificato dalla stessa. Specifica, inoltre, le funzionalità del Sistema a cui il Farmer potrà accedere.
Dipendente	Qualsiasi Registered User che non sia un Farmer. Effettua la registrazione al Sistema attraverso un Farmer, il quale lo associa ad un'Azienda Agricola.
Notifica	Consegna un messaggio inviato dal Sistema ad un Registered User, che abbia lo scopo di informarlo di un Evento, Alert o Insight. Viene rappresentata graficamente al momento del suo arrivo, e rimane visibile finché il Registered User non ne prende visione.
Schedule	Calendario che stabilisce, dato un determinato Ambiente Agricolo, gli orari in cui sarà accesa o meno l'irrigazione dello stesso. Generata automaticamente dal modulo di Decision Intelligence del Sistema.