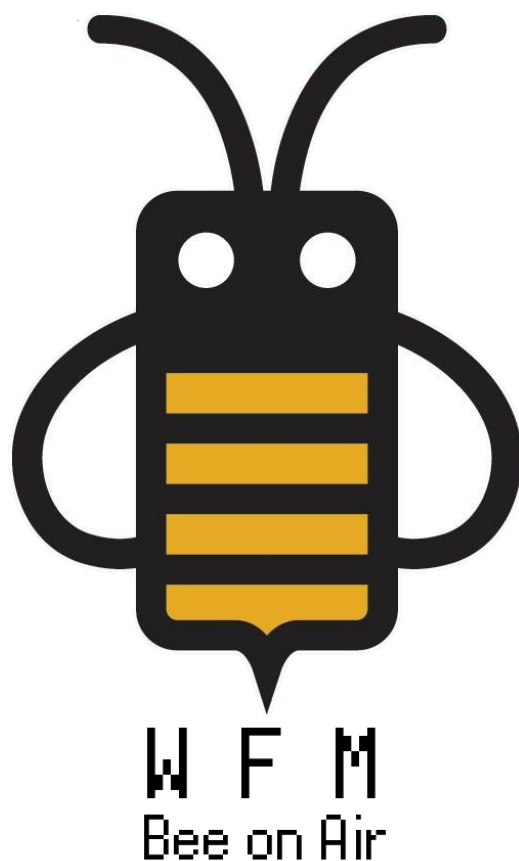


Documento di design



Introduzione	2
Architettura del sistema	2
Il modello three-tier	2
ServerAPI	3
Database	3
Cloud computing	4
Server	4
Firewall	5
Database	5
Storage	5
Media Services	5
Content delivery network	5
Advertising manager	6
Streaming	6
On demand	6
Live	6
Digital Rights Management	7
Automazione	7
Diagrammi di package	7
Packages utente non registrato	7
Packages Creator	8
Packages Administrator	9
Diagramma delle classi di progetto	10
Classi di progetto Core	10
Classi di progetto Strutture dati	11
Classi di progetto Server API lato client	11
Classi di progetto Server API lato server	12
Diagrammi di sequenza	12
Login	12
Registrazione	13
Modifica dati personali	14
Regular streaming	14
Seguire stazione	15
Ricerca stazione	15
Revisioni	15

Introduzione

Nel presente documento viene proposta la descrizione della fase di progettazione.

La prima sezione tratta le tecnologie usate per lo sviluppo del sistema, in essa viene analizzata l'architettura che verrà utilizzata.

Nella seconda sezione del documento sono definite le classi di progetto e le loro componenti, queste sono stabilite a partire dalle classi di analisi per poi sviluppare queste in classi di design con un annesso processo di raffinazione.

Alcuni dei casi d'uso presenti nel *Documento dei requisiti* e nel *Modello dei casi d'uso*, sono analizzati tramite il diagramma di sequenza, fornendo uno studio ancora più approfondito del loro funzionamento.

Architettura del sistema

L'architettura di WFM getta le sue basi sul modello client-server, con due diverse applicazioni a seconda della modalità di accesso al sistema: la versione mobile ovvero un sito web raggiungibile tramite la maggior parte dei browser, e un'applicazione per accedere tramite dispositivi iOS e Android compatibili.

Come descritto precedentemente all'interno del *Documento di visione*, l'architettura è ospitata da una infrastruttura cloud di Amazon, grazie al servizio di cloud computing Amazon Web Services.

Il modello three-tier

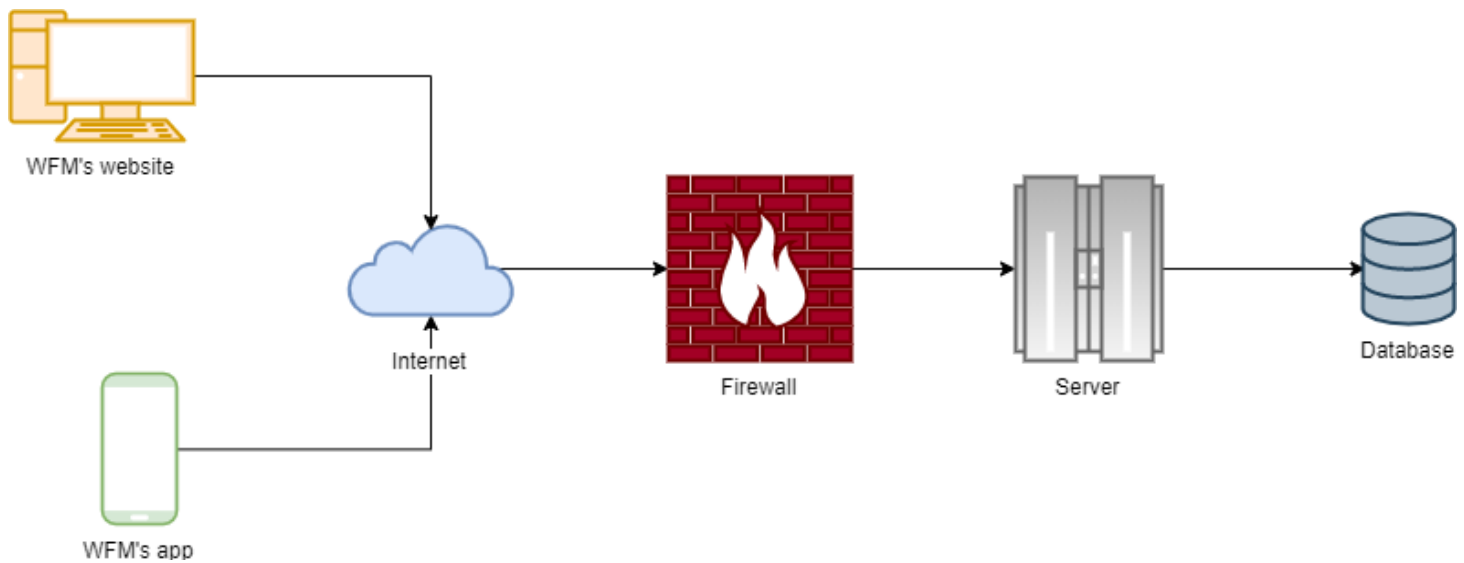
Trattandosi WFM di un'applicazione web è stato valutato il modello three-tier, come il modello di architettura più adatto al progetto, trattandosi esso di un punto di riferimento delle architetture delle tecnologie più moderne.

L'architettura sarà formata da tre strati principali:

- Il primo strato è il terminale adoperato dall'utente, esso fornisce un'interfaccia grafica tramite la quale è possibile interagire con il sistema e fruire dei servizi offerti dall'applicazione.
- Il secondo strato è rappresentato da un application server, qui è dove prende corpo la business logic, grazie a questo vengono eseguite le interazioni tramite l'utente e le operazioni interne all'applicazione
- Il terzo strato riguarda la gestione dei dati, essi vengono conservati da WFM in un database MySQL.

Le interazioni tra il terminale utilizzato dall'utente e l'application server saranno veicolate tramite internet, utilizzando il protocollo per la comunicazione https con TLS 1.2 o superiore, tramite una REST API, che chiameremo nelle pagine a seguire *ServerAPI*.

La comunicazione tra l'utente il server verrà filtrata da un firewall, al fine di proteggere i dati contenuti all'interno del server e la sua operatività.



ServerAPI

Tali ServerAPI rappresentano un collegamento fondamentale tra client e server, essa è implementata su entrambi i fronti: lato client e lato server.

La ServerAPI si occuperà, per quanto riguarda il lato client, di gestire e formulare correttamente le richieste da inviare al server, fornendo dei metodi predefiniti e funzionali per lo sviluppo, e rispetto al lato server, di prelevare le richieste esterne, di effettuare ricerche, validazioni e se necessario di interagire con servizi forniti da terzi.

La chiave dietro la ServerAPI è la modularità, essa viene ingegnerizzata per facilitare il loro l'aggiornamento in futuro.

Secondo il piano progettuale, le applicazioni client possederanno un pacchetto core contenente la ServerAPI e le strutture dati. Questo modello facilita in futuro la modifica delle strutture dati e l'aggiornamento della REST API a una nuova versione.

La ServerAPI rappresenta, all'interno delle applicazioni, l'unica chiave per l'istanziamento di queste strutture, garantendo così che la loro generazione risulti totalmente autonoma.

Database

Per la gestione dei dati persistenti è stata svolta un'attenta analisi e sono stati messi a confronto due dei servizi maggiormente diffusi: PostgreSQL e MySQL.

La scelta PostgreSQL porterebbe a una produzione più rapida del software necessario e pertanto a un abbassamento dei costi e un lancio anticipato della piattaforma, ma sono stati presi in analisi i vantaggi di adoperare MySQL nel lungo periodo, considerando che un ipotetico passaggio da PostgreSQL a MySQL successivo al lancio della piattaforma risulterebbe complesso e molto dispendioso.

Il confronto fra i due sistemi di database management ha portato alla scelta di MySQL per realizzare il DBMS necessario a WFM.

Cloud computing

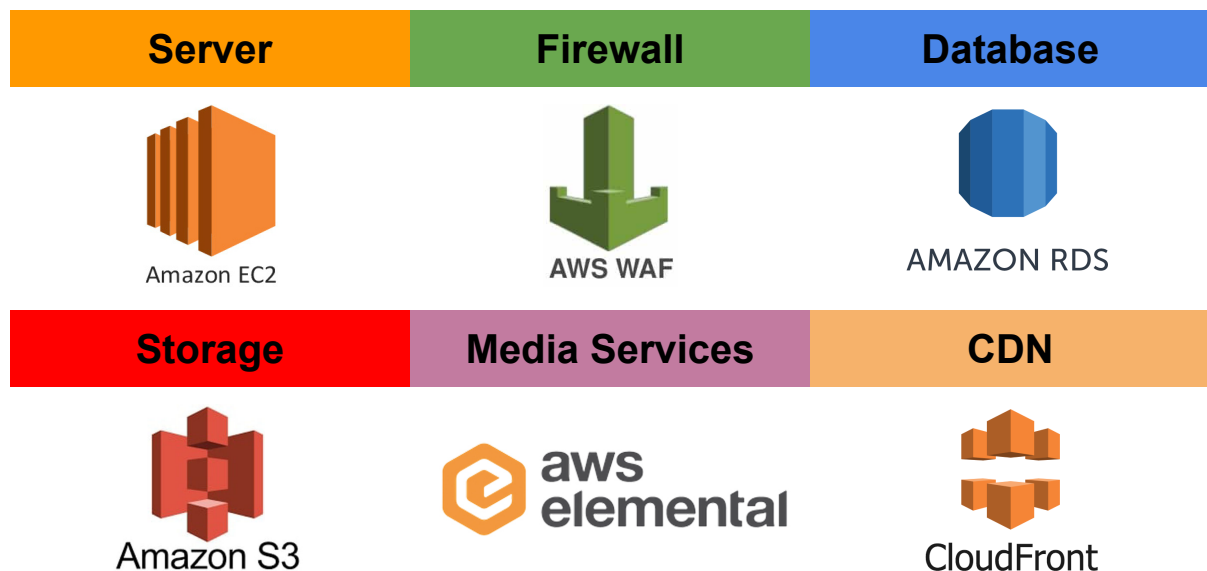
Al fine di rendere scalabile l'architettura, riducendo i costi e ottimizzando la qualità dei servizi, WFM si appoggerà a un sistema di cloud computing, ovvero dei servizi on demand offerti da terzi per la gestione delle proprie architetture.

Questo tipo di sistema infatti permette di creare un sistema sostenibile anche in caso di picchi di richiesta.

Al fine di favorire l'interazione fra le diverse componenti del sistema grazie a una maggiore compatibilità, è stato deciso di affidarsi dove possibile a un'unica compagnia, ovvero Amazon.com, la quale offre un servizio di cloud computing detto Amazon Web Service (AWS).

AWS offre diverse tipologie di servizio, quelle sfruttate da WFM sono schematizzate nella tabella sottostante, e descritte più nel dettaglio nei successivi paragrafi.

Per la gestione degli annunci pubblicitari WFM adopererà il servizio di Google LLC Google AdSense, e per la gestione della live chat



Server

Per la gestione del server, WFM farà affidamento all'applicazione di Amazon Web Services Amazon Elastic Computing Cloud (EC2).

Firewall

Il sistema di firewall verrà gestito attraverso Amazon Web Services Web Application Firewall. AWS WAF è una componente di difesa perimetrale per applicazioni web. Essa permette di difendere l'applicazione e le sue api da eventuali exploit web, evitando il rischio di compromettere la sicurezza del sistema o consumare un quantitativo eccessivo di risorse. Grazie a AWS WAF è possibile bloccare i pattern comuni di attacco e creare delle regole che filtrino pattern di traffico specifici definiti dal sistema.

Database

Per la gestione del database si farà riferimento a Amazon Relational Database Service, il quale è compatibile con il motore di database scelto, questo semplificherà l'impostazione il funzionalmente e il dimensionamento del database di WFM.

Questo servizio permetterà inoltre di aumentare il grado di sicurezza del sistema, potendo salvare i propri dati attraverso delle chiavi di crittografia gestite da Amazon stesso.

Storage

Il sistema di storage verrà gestito attraverso il servizio Amazon S3, ovvero Amazon Simple Storage Services, un servizio web di memorizzazione e archiviazione messo a disposizione da Amazon Web Service.

Media Services

Il sistema di AWS farà affidamento a Amazon Web Services Elemental, un pacchetto di servizi per la gestione dei file multimediali che verrà adoperato per l'elaborazione dei contenuti audio prima della loro distribuzione.

Tra i servizi che verranno usati dal sistema ci sono:

- Amazon Web Services Elemental Media Convert;
- Amazon Web Services Elemental MediaLive;
- Amazon Web Services Elemental MediaPackage.

Content delivery network

Il servizio a cui si fa riferimento all'interno di questo paragrafo è Amazon CloudFront, un content delivery network (ovvero una rete di distribuzione dei contenuti) utilizzabile insieme ad Amazon Web Services per implementare il sistema di streaming.

Esso permette sia la realizzazione di uno streaming on demand che live.

Tra le ragioni che ci hanno portato alla scelta di questo specifico servizio ci sono la velocità e l'efficienza di esso, che permetteranno a WFM di ottimizzare la qualità delle proprie trasmissioni, il fatto che questo sia un sistema altamente programmabile, un attributo molto vantaggioso sia in fase di progettazione che in future fasi di aggiornamento.

Amazon CloudFront è un servizio diffuso già all'interno di molte delle più famose piattaforme di streaming di media come Amazon Prime Video, Spotify.

Advertising manager

Non essendo presente, all'interno del pacchetto Amazon Web Service alcun servizio di gestione di spazi pubblicitari idoneo al sistema di WFM, il team di progettazione ha deciso di fare affidamento a Google AdSense, un servizio offerto da Google LLC, il quale permette di inserire annunci pubblicitari all'interno del media player dello show.

La gestione e selezione degli annunci verrà quindi completamente delegata a Google.

Streaming

On demand

Per offrire un servizio di streaming on demand, il contenuto viene memorizzato all'interno di Amazon S3, dove avviene l'archiviazione. Al fine offrire una soluzione completa, viene adoperato Amazon Web Service Elemental Media Convert per l'elaborazione dei file audio, e successivamente Amazon CloudFront per la sua distribuzione.

Amazon EMC prende il contenuto da Amazon S3, questo viene transcodificato in base a eventuali richieste e il risultato di questa operazione viene memorizzato in S3.

La transcodifica genera una elaborazione dei file audio, generando una versione compressa del contenuto originale, così da ridurre il peso in memoria, modificarne il formato e aumentare la compatibilità con i diversi dispositivi

Conclusa questa operazione i contenuti sono archiviati all'interno di Amazon S3 e disponibili per lo streaming, la loro distribuzione verrà svolta da Amazon CloudFront.

Il network per il content delivery memorizza il contenuto all'interno dell'edge per minimizzare la latenza e ottenere una velocità di trasmissione più elevata.

Una raccolta di protocolli di streaming, tra cui sono presenti il Dynamic Adaptive Streaming over HTTP (DASH), il Live Streaming HTTP di Apple (HLS), il Smooth Streaming di Microsoft (MSS) e il Dynamic Streaming HTTP di Adobe (HDS), vengono utilizzati per migliorare l'esperienza dell'utente, trasmettendo il video nello stesso momento in cui è in corso la visione dello stesso.

Viene utilizzato ancora MediaConvert per convertire i file audio in formato HLS, esso rappresenta il formato di streaming più diffuso e ampiamente supportato, in tal modo esso viene frazionato in segmenti di minori dimensioni e verrà generato un file manifest, al quale il sistema indirizzerà il CloudFront.

In conclusione, per riprodurre le tracce audio verrà incorporato l'URL manifest nei dispositivi di riproduzione utilizzati dagli utenti per guardare la diretta streaming.

Live

AWS Elemental MediaLive prende una sorgente audio in diretta di grandi dimensioni, questa viene compressa in una versione di minore dimensione da distribuire agli utenti.

AWS Elemental MediaPackage si occupa a questo punto della conversione del contenuto in diversi formati per ottimizzare la visione e funzioni secondarie sui diversi dispositivi.

Viene generato un manifest ai quali è indirizzato Amazon CloudFront, quest'ultimo si occuperà della distribuzione della diretta.
 Conclusa la registrazione questa viene salvata per essere convertita in un onDemandShow.

Digital Rights Management

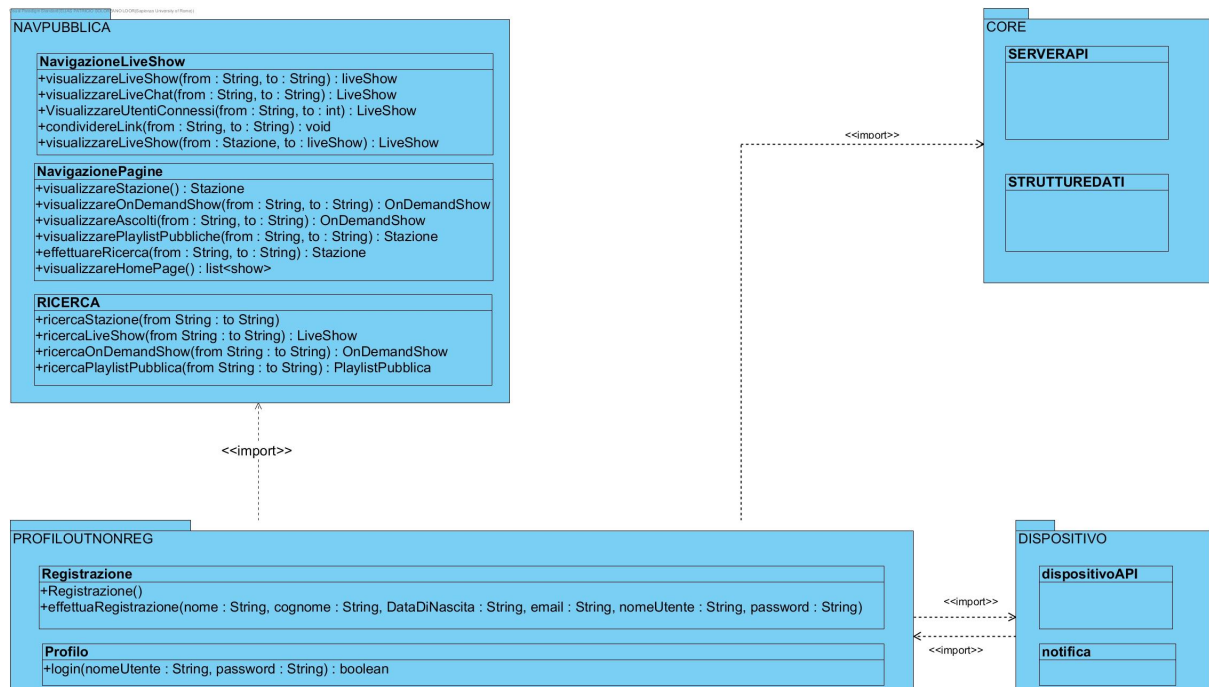
Amazon Web Services comprende all'interno del proprio servizio di supporto streaming dei sistemi di Digital rights management, ovvero dei sistemi di gestione dei diritti digitali, i quali saranno in grado di impedire la registrazione dei contenuti sulla piattaforma, sia in live streaming che in on demand streaming.

Automazione

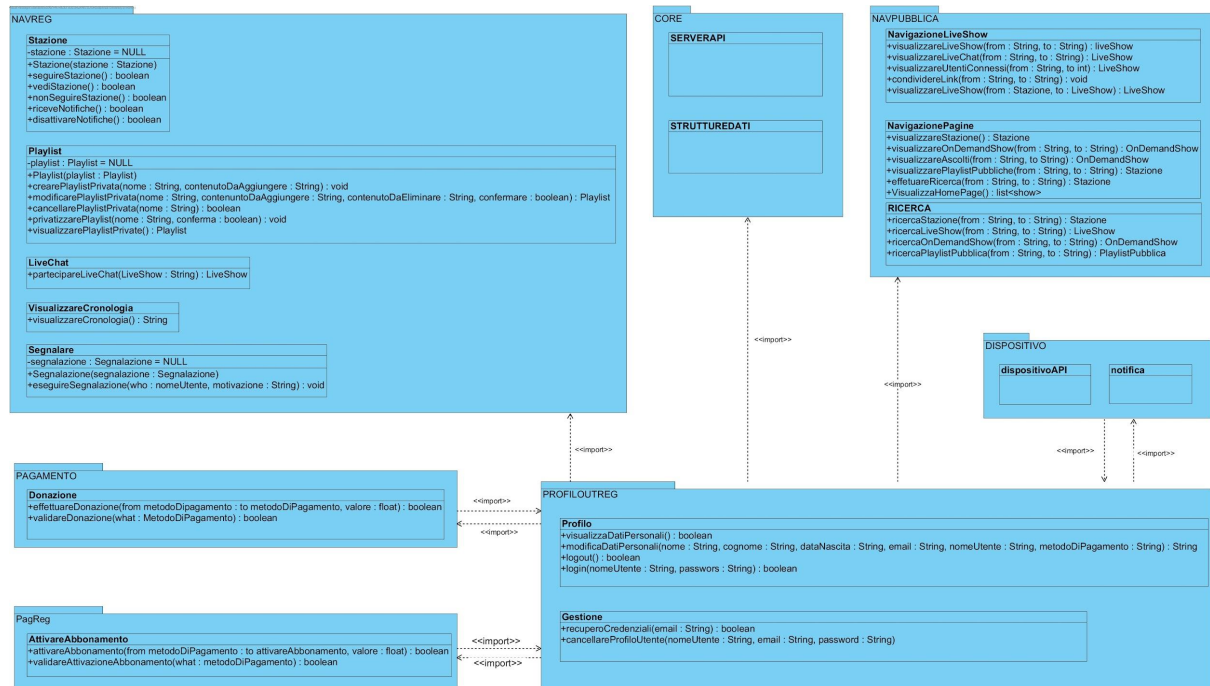
Amazon Web Services offre un servizio di automazione di queste due tipologie di streaming, tramite il quale WFM distribuirà i contenuti caricati dai creator sulla piattaforma ai propri utenti.

Diagrammi di package

Packages utente non registrato



Packages Utente registrato



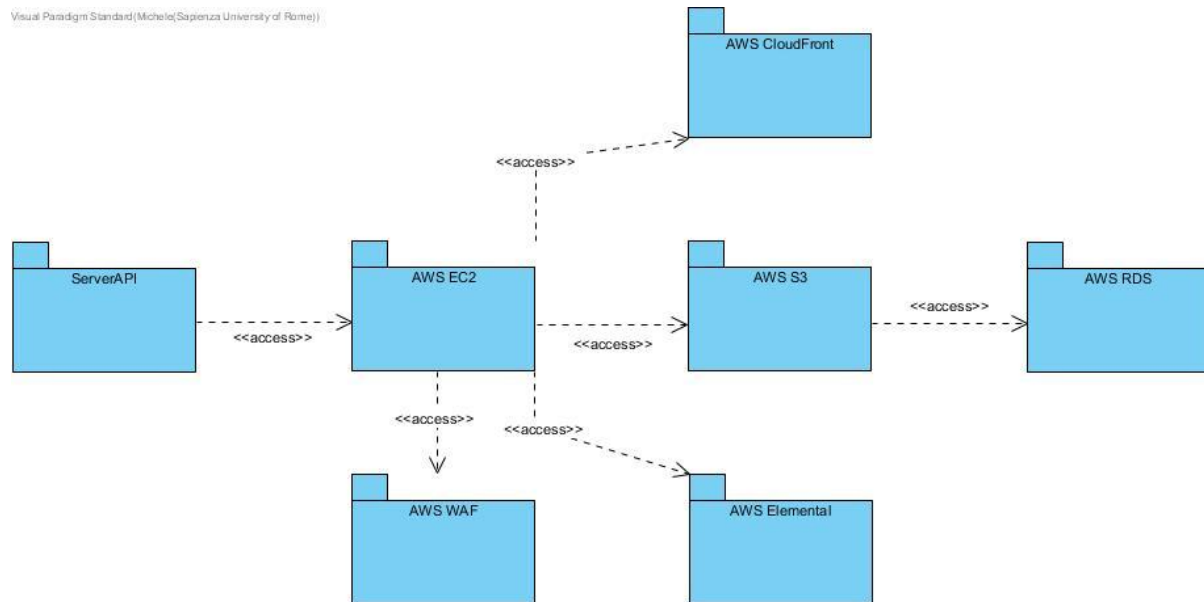
Packages Creator



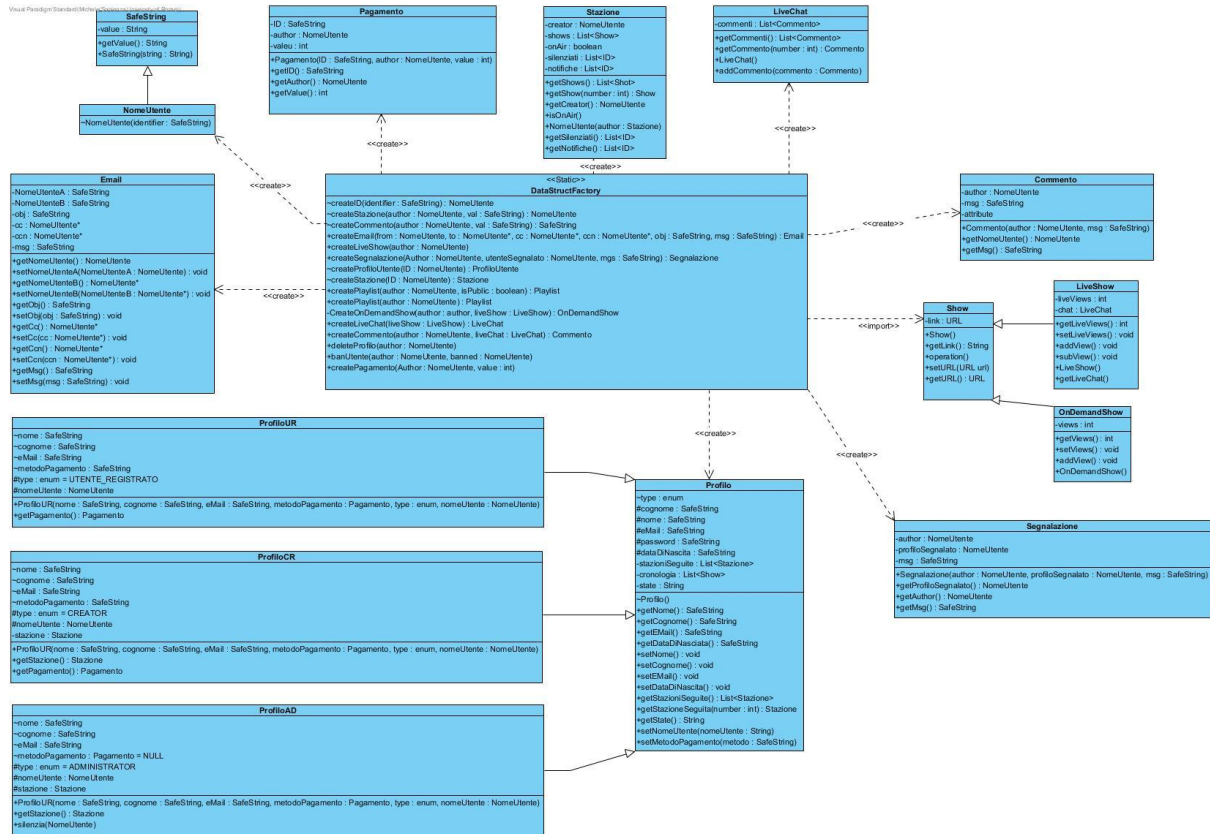


Diagramma delle classi di progetto

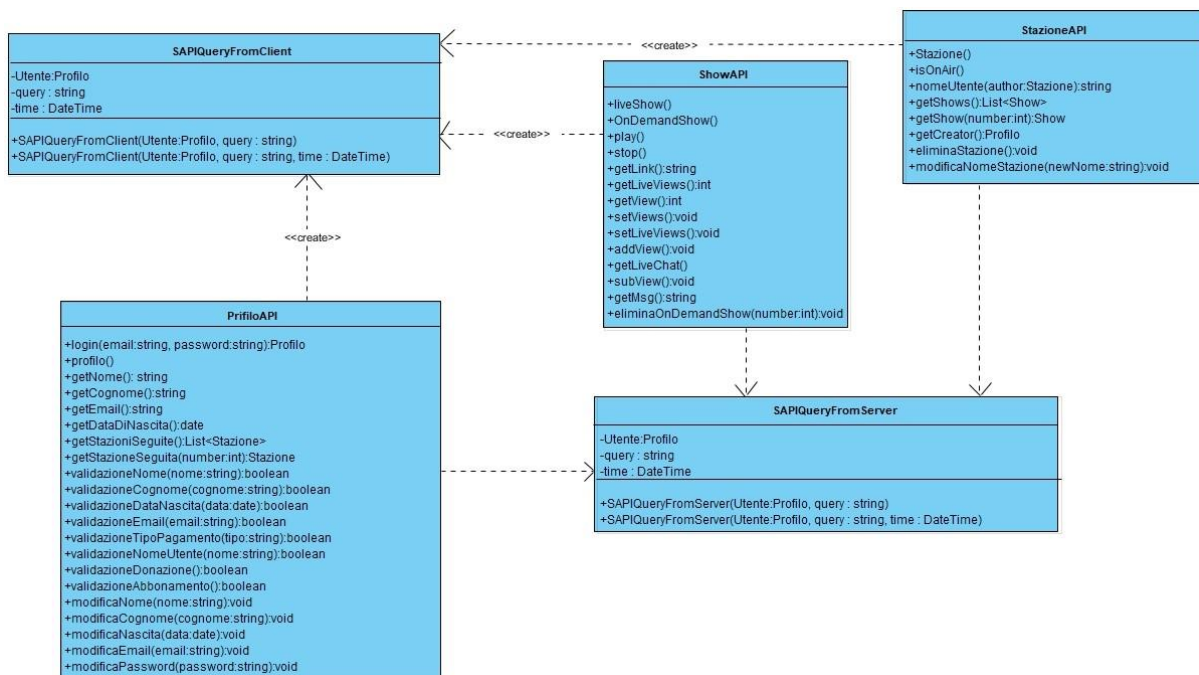
Classi di progetto Core



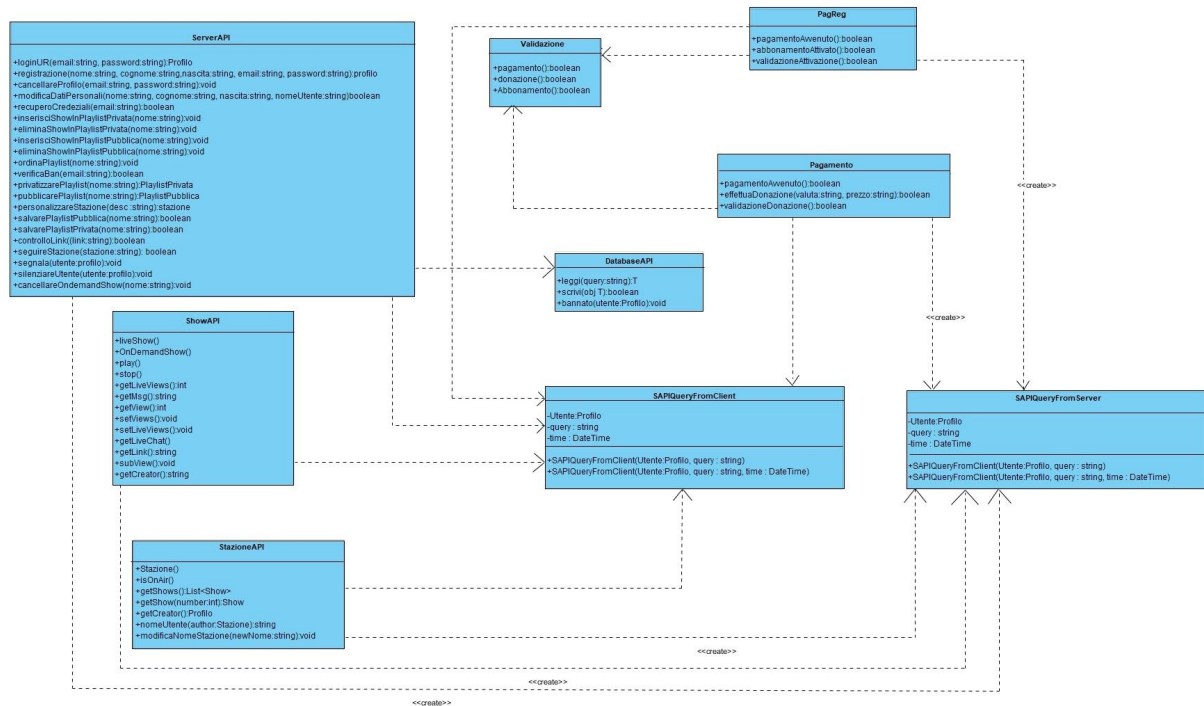
Classi di progetto Strutture dati



Classi di progetto Server API lato client



Classi di progetto Server API lato server

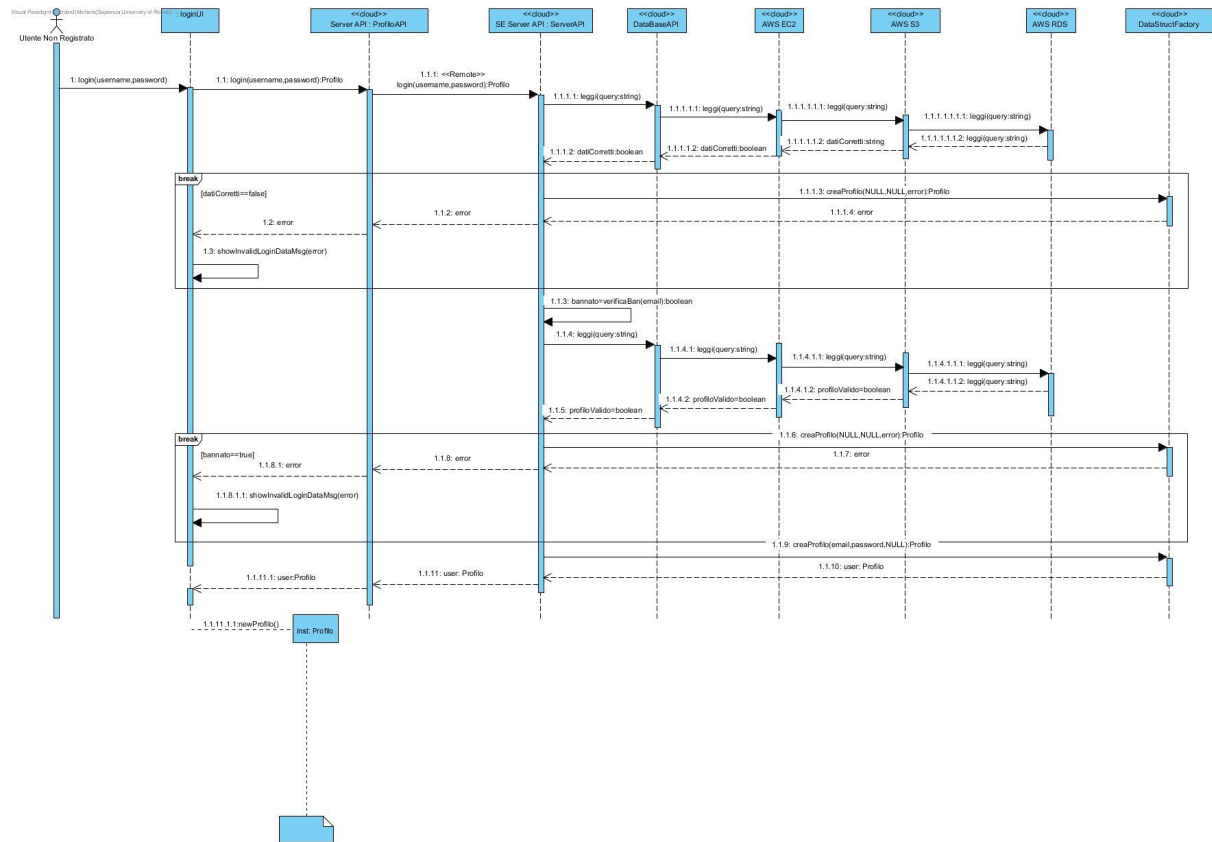


Diagrammi di sequenza

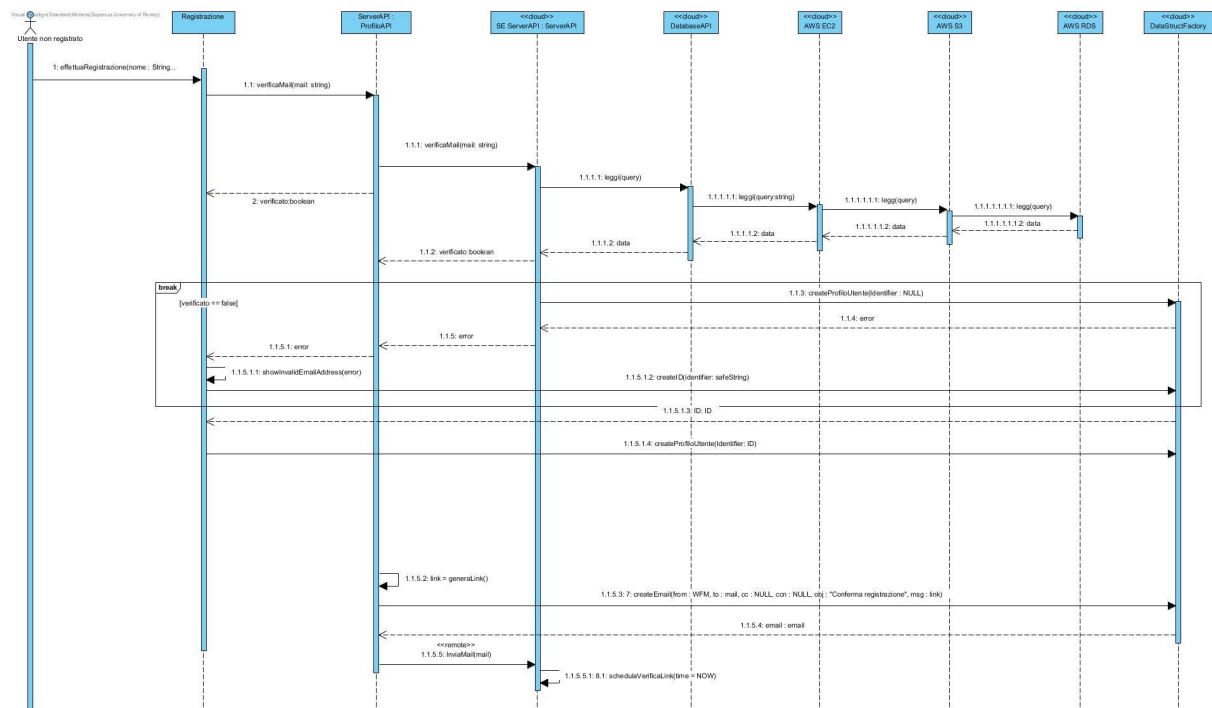
In questa sezione viene mostrata la dinamica di interazione tra i diversi package e le classi di progetto per l'esecuzione di alcuni task.

Al fine di avere una copertura più completa sono stati presi in analisi almeno un task corrispondente ad use case per ogni classe di quelle elencati all'interno del *Documento dei casi d'uso*.

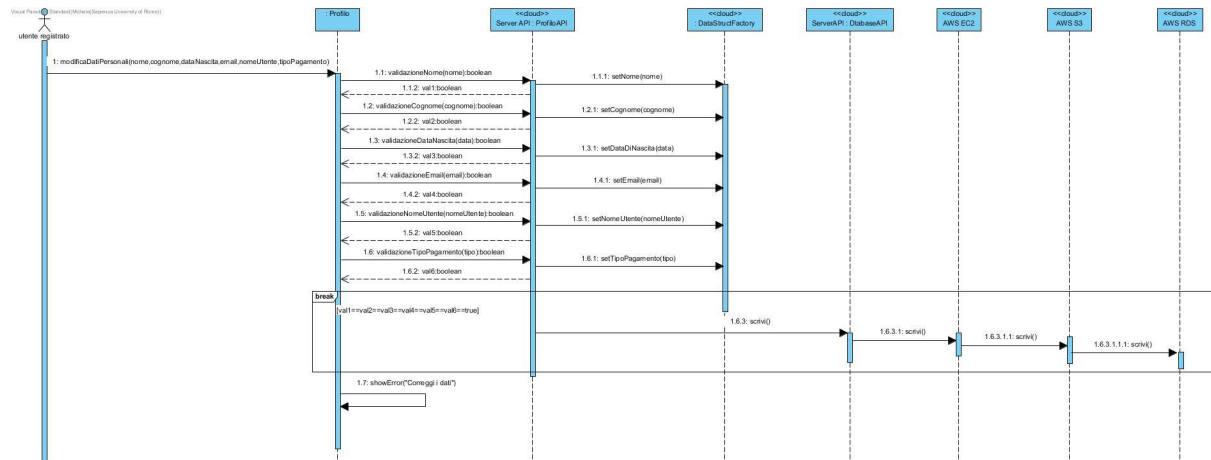
Login



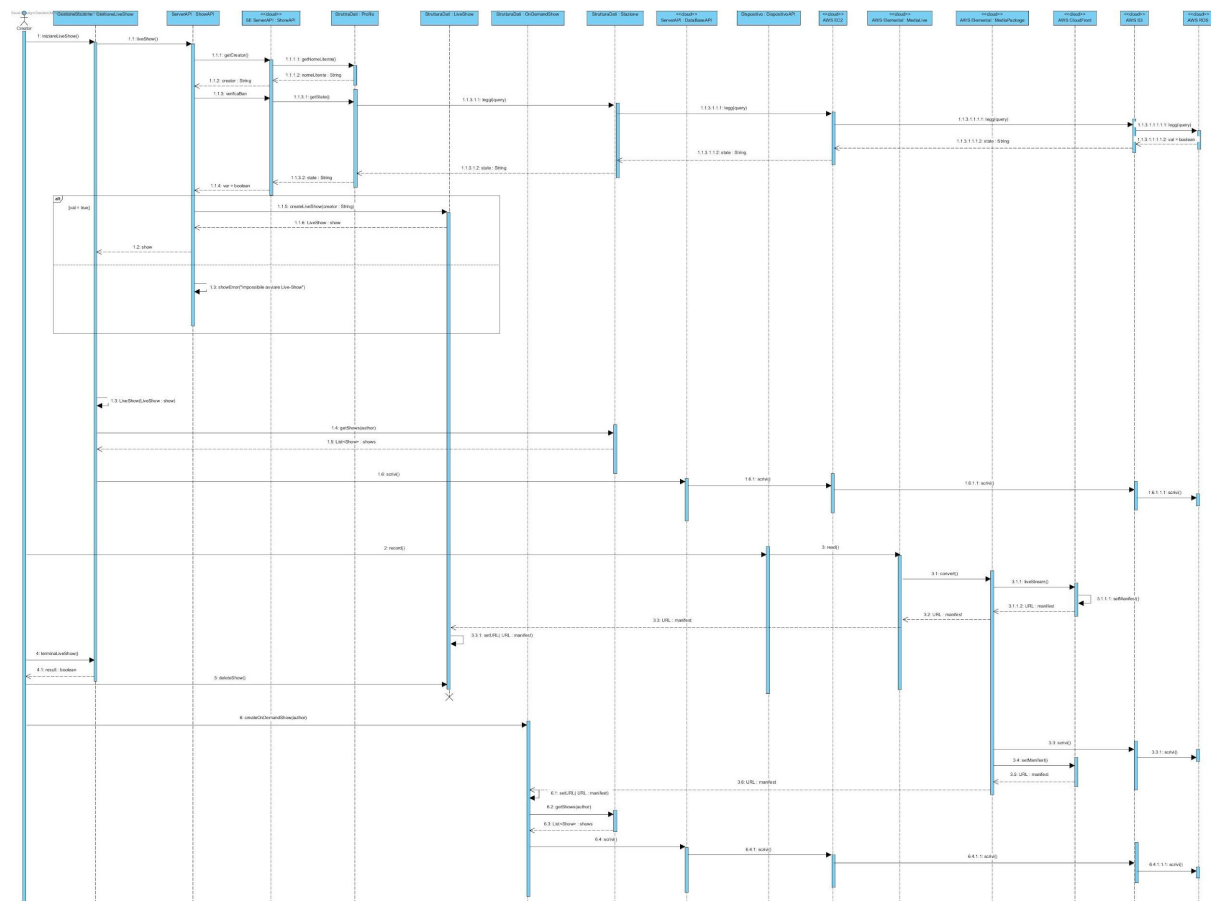
Registrazione



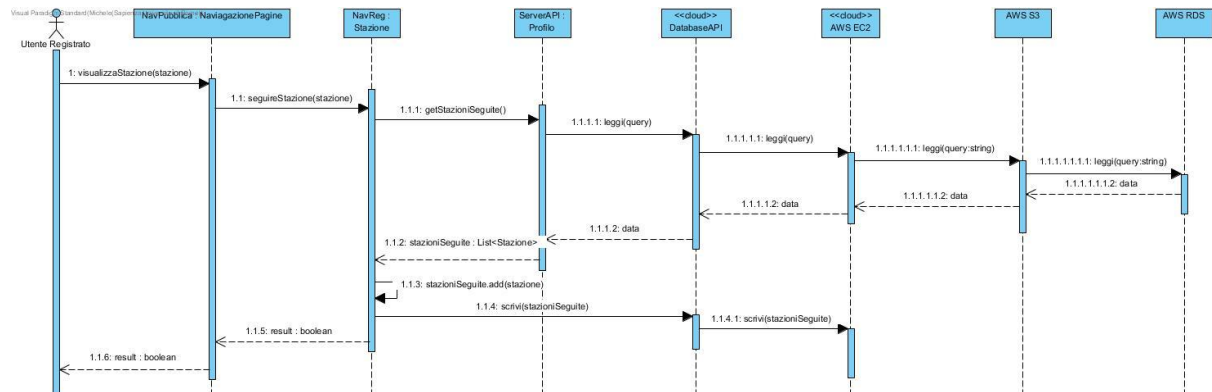
Modifica dati personali



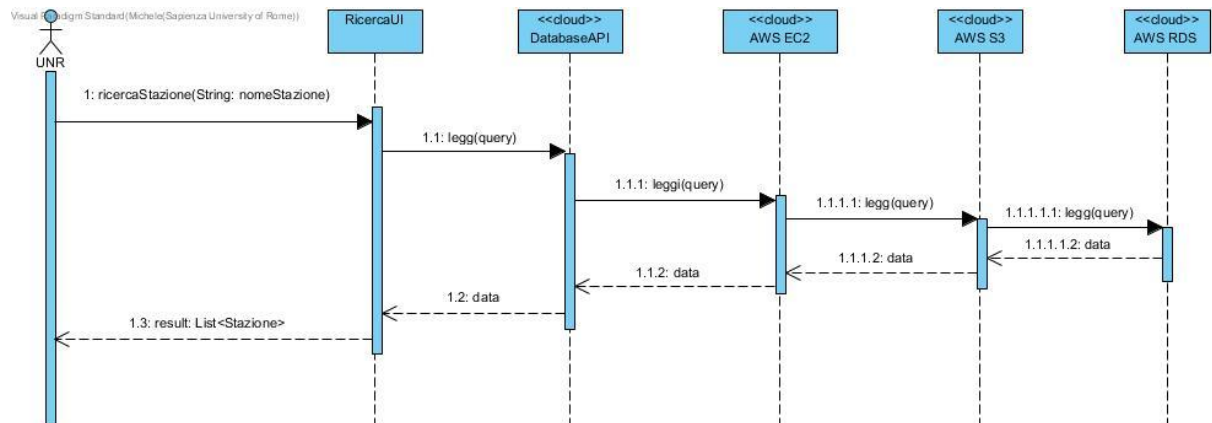
Regular streaming



Seguire stazione



Ricerca stazione



Revisioni

Numero	Descrizione	Data
1	Stesura iniziale	17/12/2019
2	Inserimento sistema di streaming AWS	08/01/2020
3	Conclusione documento	30/01/2020