

D3.B - Proof of Concept della piattaforma

Autori: Marialuisa Mongelli (ENEA), Andrea Fantini, Giacomo Napoli (Tecnostudi Ambiente S.r.l.), Mauro Saccone (Roma TRE).

Altri contribuenti al lavoro riportato nel deliverable: Giovanna Spadafora, Mauro Saccone, Marco Puccini.

Revisori: Marco Canciani (Roma Tre)

Sommario

Introduzione	2
1. Descrizione generale della piattaforma	2
1.1 Schema logico funzionale piattaforma	4
2. Descrizione ruoli e permessi utenti	5
3. Descrizione orchestratore di microservizi	6
4. Descrizione moduli software	6
4.1 Modalità di autenticazione	6
4.2 Directus	6
4.3 Aton	7
4.4 Archiviazione dati e metadati	8
4.5 Architettura Database	8
5. Struttura repository	10
6. Servizio di Backup	10
7. Modalità di Deploy	10
8. Descrizione interfaccia piattaforma	11
Bene digitale	12
Bene culturale	12
Organizzazione	15
Persone	16

Introduzione

La piattaforma “KORE” del progetto D-TECH, *Digital Twin Environment for Cultural Heritage*, si basa su una infrastruttura a micro-servizi strutturata su Kubernetes. L’architettura è stata concepita su un’architettura di tipo Open Source, scalabile e modulabile alle esigenze del progetto, al fine di una maggiore flessibilità e adattabilità con gli altri componenti del progetto, quali ad esempio il front end di “Aton” per la gestione e la visualizzazione evoluta dei dati tridimensionali (3d).

In particolare, ogni servizio implementato ha un’architettura a sé costituita da una logica ben definita e da interfacce per la comunicazione con gli altri servizi per mezzo di API.

La parte centrale della piattaforma “Kore” è costituita dal “data storage” e dal relativo database relazionale per la meta-datazione dei dati, volto alla raccolta di tutti e metadati prodotti dai partner del progetto e dai potenziali fruitori così come previsto nei task T 2.2, T2.3 e T2.4.

1. Descrizione generale della piattaforma

Nel dettaglio, la realizzazione della piattaforma “Kore” segue uno stack su base open source strutturato in micro-servizi, così da garantire una maggiore flessibilità e adattabilità, si riporta nell’immagine seguente lo schema semplificato della piattaforma:

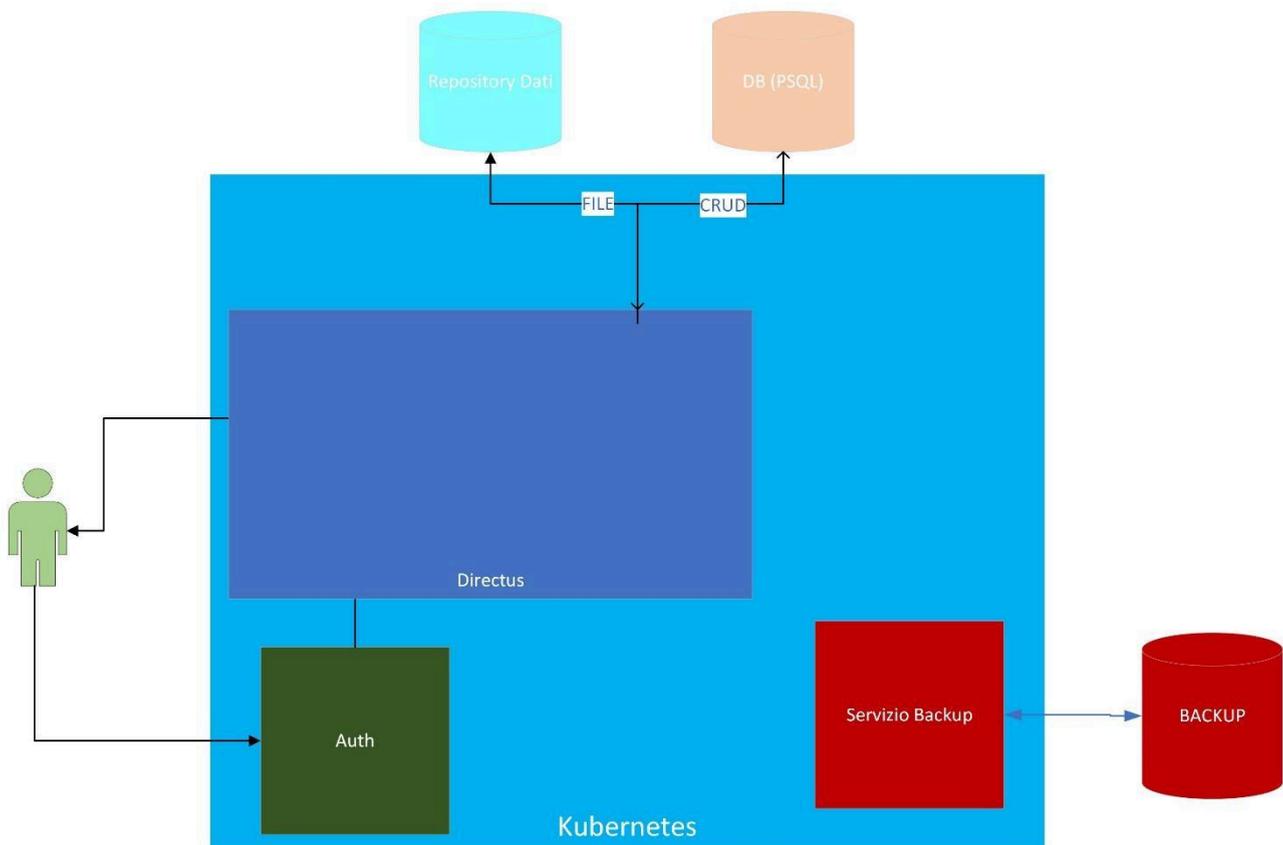


Fig.1- Schema semplificato stack tecnologico

La piattaforma segue la logica composta da un backend per l’acquisizione e la gestione dei dati con metadatezione e un frontend per la visualizzazione e l’interazione base con i dati.

L’intero stack è composto dai seguenti servizi su base Kubernetes (K8s):

- Modulo autenticazione

- Gestione dei dati
- API di comunicazione
- Integrazioni con Aton
- Piattaforma di backup

Inoltre, lo stack è costituito da servizi esterni, alla piattaforma k8s, di gestione dei dati semplici di tipo “anagrafici” tramite piattaforma Database PSQL (*Postgres*) e repository su base S3 o similari.

Lo stack fa uso del progetto open-source Directus come framework di partenza per la gestione e visualizzazione dei dati. Nel dettaglio Directus è basato interamente in *Typescript*, su *Node.js* e *Vue.js* con una struttura modulare ed estensibile basata su diversi moduli personalizzabili che consentono di interagire con i dati in un modo specifico.

Le tecnologie per il momento coinvolte nello stack saranno:

- Orchestratore: Kubernetes
- Autenticazione: JWT
- Backend: Typescript/nodejs
- Frontend: Typescript/vueJs
- Cache (eventuale): Redis/memcache
- Directus
- Database: PostgreSQL
 - o Postgis-psql
- Datastore
 - o Bucket / S3 (o piattaforme similari)

Inoltre, è stata prevista una funzione facoltativa di distribuzione in sola lettura da parte dei stakeholders di dati e relative informazioni presenti nelle piattaforme singole distribuite così da poter centralizzare nella versione KORE di ENEA tutte le informazioni di pubblicazioni.

Al fine di finalizzare la piattaforma KORE in relazione alle esigenze intercorse durante lo svolgimento del progetto, l’infrastruttura cloud a microservizi è stata adeguata ottimizzando le risorse messe a disposizione da parte di ENEA con l’ampliamento, fino alla massima capienza disponibile, dello storage messo a disposizione, pari a 200GB.

Il database ha subito delle modifiche al fine di recepire quanto emerso nelle riunioni di progetto, in particolare le anagrafiche dei beni sono state modificate per accogliere le esigenze di nomenclatura e catalogazione. Sono state anche ottimizzate le query e i collegamenti relazionali tra le tabelle che compongono il database al fine di aumentare le performance di lettura e scrittura dello stesso. Infine, per migliorare la risposta dell’intera piattaforma e dei relativi servizi erogati, quali gestione anagrafica dei metadati relativi ai “beni” e dei visualizzatori 3D (base ed evoluto) è stato utile suddividere le chiamate esterne su indirizzi differenti al fine di instradare al meglio le chiamate per singolo servizio visto che non tutti i fruitori presentano le stesse necessità di utilizzo della piattaforma.

I DNS finali definiti sono:

- dtech.enea.it: Riservato alla gestione di base dell’anagrafica beni e dei loro metadati;
- viewer.dtech.enea.it: Riservato alla visualizzazione di base per i modelli 3D caricati dalla piattaforma dtech.enea;
- model.dtech.enea.it: Riservato alla visualizzazione evoluta dei modelli 3D caricati dalla piattaforma dtech.enea

Al fine di generare tutto il codice sorgente conforme con gli ultimi sviluppi e il relativo porting sulla piattaforma GitHub di ENEA, tutti gli script e le relative procedure di installazione sono state aggiornate con

l'ultima versione in produzione, 1.0.2, che è possibile scaricare dal github del progetto ENEAICT4CH¹ (github.com)

1.1 Schema logico funzionale piattaforma

La piattaforma “kore” si basa su una logica funzionale come rappresentato nelle successive figure (Fig. 2 e Fig. 3), incentrata e semplificata sull’utente finale sia per le fasi di “data entry” che di visualizzazione dei dati presenti nella piattaforma.

Nel dettaglio nella figura sottostante (Fig.2) è riportato uno schema semplificato del flusso inerente all’inserimento di un nuovo dato nel “repository” della piattaforma “Kore” del cliente. L’intero flusso dovrà essere governato dal “frontend” dell’applicativo personalizzato di Directus che tramite API comunica con la parte di “backend” e di “repository” dei dati e dei relativi metadati anagrafici/descrittivi.

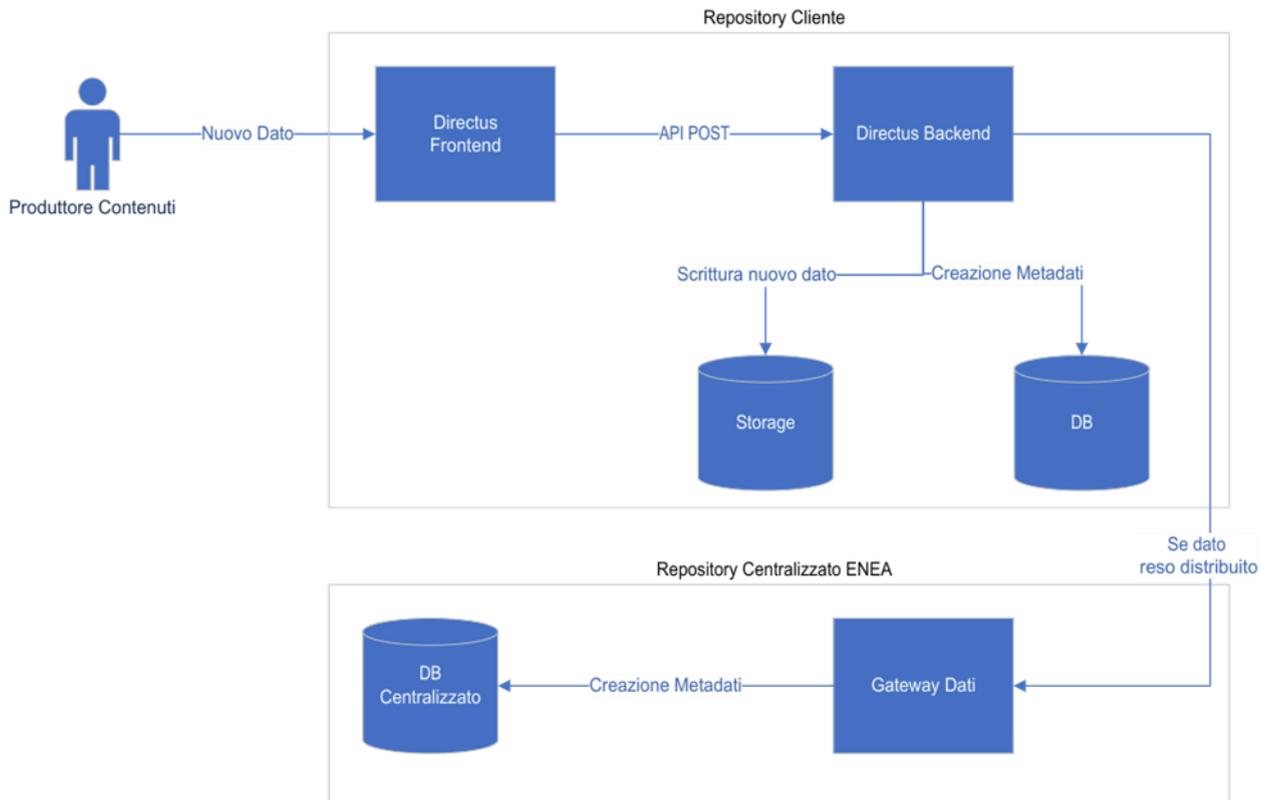


Fig.2- Schema semplificato di creazione e aggiornamento dati nella piattaforma

Come rappresentato in figura 2, i potenziali utilizzatori/clienti, hanno la possibilità di selezionare parte o l’intero dataset di loro competenza, al fine di autorizzare la condivisione delle informazioni con il repository centralizzato di ENEA. Tale funzionalità comporta solo la copia dei dati anagrafici / metadati (DB) nel dataset centralizzato della piattaforma “Kore” di Enea (DB Centralizzato), rimandando l’eventuale visualizzazione dei dati nella piattaforma del cliente.

La visualizzazione dei dati 3d presenti nella piattaforma “Kore” segue la logica rappresentata nella figura 3 sia per la visualizzazione in modalità base che per la visualizzazione “evoluta” tramite l’integrazione nella piattaforma del modulo Aton.

¹ <https://github.com/ENEAICT4CH/D-TECH>

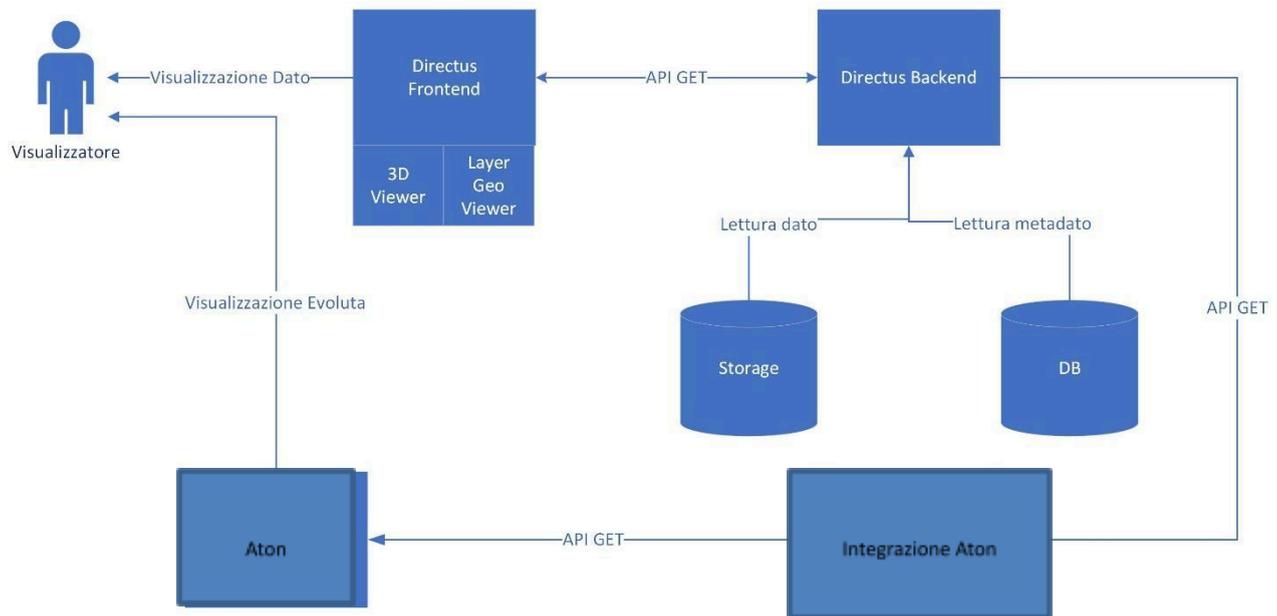


Fig.3- Schema semplificato di visualizzazione dei dati nella piattaforma

Il visualizzatore, tramite il “frontend” di Directus ha l’autonomia di gestire e visualizzare i dati e/o i relativi metadati sia geografici che tridimensionali. La visualizzazione evoluta attraverso il motore 3D Aton con possibilità di modifica della copia digitale è consentita solo ad una categoria di utenti, di seguito descritti come “creatori di contenuti”.

2. Descrizione ruoli e permessi utenti

Gli accessi alla piattaforma sono stati divisi in tre categorie di utenza:

1. Amministratori di Sistema
2. Creatori di contenuti
3. Visualizzatori

Gli Amministratori di sistema sono coloro che hanno accesso alle varie componenti dell’infrastruttura informatica e che potranno agire su di essa. All’interno di questa categoria ci sono vari sotto ruoli; al momento sono stati previsti i seguenti:

1. Gestore Infrastruttura: saranno le persone addette al deploy delle singole componenti e alla sua manutenzione.
2. Gestore Database/Datastore: saranno le persone che potranno agire sui dati direttamente senza dover passare dalla piattaforma
3. Gestore Piattaforma Directus: si occuperà della modifica/gestione della piattaforma directus

Potranno essere aggiunti in fase implementativa ulteriori ruoli qui non presenti.

I creatori di contenuti hanno accesso alla piattaforma directus e da essa possono caricare/modificare/eliminare contenuti e utilizzare la visualizzazione evoluta attraverso il motore 3D Aton. La piattaforma permette una granularizzazione dei ruoli al fine di permettere agli amministratori di sistema una accurata definizione dei permessi per ogni tipologia di richiesta, così da tutelare la consistenza dei dati inseriti da parte degli operatori.

3. Descrizione orchestratore di microservizi

Le tecnologie per la creazione dei servizi sono di tipo Docker per la creazione delle immagini e Kubernetes per l'orchestrazione. Di seguito una breve descrizione delle due tecnologie.

Docker è una tecnologia di containerizzazione che consente la creazione e l'utilizzo dei container Linux.

Docker, considera i container come macchine virtuali modulari estremamente leggere, offrendo la flessibilità di creare, distribuire, copiare e spostare i container da un ambiente all'altro, ottimizzando così le app per il cloud.

La tecnologia Docker utilizza il kernel di Linux e le sue funzionalità, come Cgroups e namespace, per isolare i processi in modo da poterli eseguire in maniera indipendente. Questa indipendenza è l'obiettivo dei container: la capacità di eseguire più processi e applicazioni in modo separato per sfruttare al meglio l'infrastruttura esistente pur conservando il livello di sicurezza che sarebbe garantito dalla presenza di sistemi separati.

Gli strumenti per la creazione di container, come Docker, consentono il deployment a partire da un'immagine. Ciò semplifica la condivisione di un'applicazione o di un insieme di servizi, con tutte le loro dipendenze, nei vari ambienti. Docker automatizza anche la distribuzione dell'applicazione (o dei processi che compongono un'applicazione) all'interno dell'ambiente containerizzato.

Gli strumenti sviluppati partendo dai container Linux, responsabili dell'unicità e della semplicità di utilizzo di Docker, offrono agli utenti accesso alle applicazioni, la capacità di eseguire un deployment rapido, e il controllo sulla distribuzione di nuove versioni.

Kubernetes è uno strumento open source di orchestrazione e gestione di container. È stato sviluppato dal team di Google ed è uno dei tool più utilizzati a questo scopo. Kubernetes permette di eliminare molti dei processi manuali coinvolti nel deployment e nella scalabilità di applicazioni containerizzate e di gestire in maniera semplice ed efficiente cluster di host su cui vengono eseguiti i container.

4. Descrizione moduli software

Di seguito vengono descritti i principali moduli software che dovranno essere sviluppati/implementati per la corretta esecuzione del progetto.

4.1 Modalità di autenticazione

L'autenticazione avviene tramite login utente e password, le successive chiamate tra back end e front end saranno gestite, al fine di rendere il sistema il più sicuro possibile, attraverso l'utilizzo di un token JWT e saranno effettuate sotto protocollo HTTPS.

In base alle esigenze applicative verranno predisposti diversi ruoli e livelli di autenticazione come descritto nel paragrafo "Descrizione ruoli e permessi".

4.2 Directus

Directus è una piattaforma Open Data e Open Source creata per democratizzare il database.

Questa piattaforma offre accesso ai dati e alla gestione delle risorse dei file digitali, per qualsiasi modello di dati o progetto. Directus si collega a diversi tipi di database SQL e a diverse tipologie di Storage per l'archiviazione di file. Directus consente di eseguire operazioni CRUD, creare utenti, assegnare ruoli con autorizzazioni completamente configurabili, creare query complesse e granulari, configurare webhook basati su eventi e automazione delle attività.

Le versioni SQL sono tutte dotate di tipi di dati e idiomi leggermente diversi. Directus utilizza Database Introspection per leggere la struttura del database e creare un livello di astrazione per gestire tutti i dettagli essenziali dietro le quinte. Questa tecnica funziona perfettamente con qualsiasi tipologia di database SQL. Al livello più alto, Directus organizza le sue caratteristiche e funzionalità in Moduli.

Costruito interamente in Typescript, principalmente su Node.js e Vue.js, Directus è completamente open source, modulare ed estensibile.

Nel progetto finale la piattaforma Directus dovrà essere customizzata al fine di rispondere alle esigenze di progetto, di seguito descritte:

- Logo del progetto/cliente
- Colori e stile caratterizzanti
- Gestione granulare degli accessi come documentato in precedenza
- Integrazioni con gli altri sistemi
- Installazione e deploy su kubernetes per aumentare affidabilità e scalabilità

Front end

Il front end dell'applicazione sfrutta l'interfaccia messa a disposizione da Directus e ha due sottomoduli per la visualizzazione dei dati 3D, Aton (sezione evoluta) e 3dHop (sezione base).

Qualora durante l'esecuzione del progetto sorgano esigenze particolari non coperte da Directus, il fornitore potrà integrare tale interfaccia con moduli ad hoc sviluppati in VueJs.

VueJS è un framework JavaScript progressivo open source utilizzato per sviluppare interfacce web interattive. È uno dei famosi framework utilizzati per semplificare lo sviluppo web. VueJS si concentra sul livello di visualizzazione. Può essere facilmente integrato in grandi progetti per lo sviluppo front-end senza problemi.

Back end

Il back end dell'applicazione deve sfruttare l'interfaccia messa a disposizione da Directus, per quanto riguarda le operazioni base con database e storage dei dati.

Qualora durante l'esecuzione del progetto dovessero sorgere esigenze particolari non coperte da Directus, il fornitore potrà integrare tale interfaccia con moduli ad hoc sviluppati in NodeJs.

Node.js è una piattaforma di sviluppo open source per l'esecuzione di codice JavaScript/Typescript lato server. Node è utile per lo sviluppo di applicazioni che richiedono una connessione persistente dal browser al server e viene spesso utilizzato per applicazioni in tempo reale come chat, feed di notizie e notifiche push web.

API di comunicazione

Directus genera in modo dinamico gli endpoint API REST e GraphQL per gestire i dati con un controllo granulare. Le API rispecchiano dinamicamente lo schema e i contenuti effettivi in tempo reale. Questa soluzione è simile a come funzionano i client di database tecnici (come phpMyAdmin).

Tuttavia, Database Introspection presenta molti vantaggi esclusivi:

- Controllo assoluto dello schema del database SQL.
- Trasparenza, portabilità e sicurezza per i dati.
- Consente di importare database esistenti, inalterati e senza migrazioni.
- Accesso diretto al database e tutta la potenza di query SQL complesse.
- Miglioramenti significativi delle prestazioni attraverso ottimizzazioni e indicizzazione.

Quindi per tutte le operazioni CRUD (Create, Read, Update, Delete) di base sul database le API messe a disposizione saranno quelle di Directus. Mentre per esigenze particolari verranno sviluppati dei moduli ad hoc, come descritto nel successivo paragrafo.

4.3 Aton

il Visualizzatore dei dati 3D evoluti, open source Aton, è perfettamente integrato nello stack della piattaforma Kore implementata.

4.4 Archiviazione dati e metadati

L'archiviazione dei dati è garantita attraverso database relazionali, per tutti i dati e metadati in formato tabellare e su storage dedicato per tutti i file di dati complessi (i.e. shape file, Immagini, nuvole di punti, etc...).

È stata prevista una progettazione logica dei database e dello storage in relazione ai dati generalizzati che verranno forniti da ENEA in fase successiva.

Inoltre, è stata prevista l'implementazione per i database relazionali di una configurazione federata che accentri i dati di solo tipo tabellare, previa autorizzazione dell'ente, su un database centralizzato di proprietà e gestione di Enea.

4.5 Architettura Database

La struttura del database è di tipo relazionale con chiavi primarie e secondarie ben definite e ottimizzate, con relative policy di cancellazione aggiornamento a cascata. Il database è indicizzato in maniera ottimizzata per i soli campi oggetto di query e di ricerca.

Ogni tabella contiene i seguenti campi base, ti tipo obbligatori:

- **date_created:** tipo TIMESTAMP, riporta la data di creazione dell'entità all'interno del database
- **date_updated:** tipo TIMESTAMP, riporta la data di aggiornamento dell'entità all'interno del database
- **user_created:** tipo String, riferimento all'utente che ha creato l'entità all'interno del database
- **user_updated:** tipo String, riferimento all'utente che ha aggiornato l'entità all'interno del database.

Le tabelle di anagrafica o di rappresentazione del dato (metadattazione) o di dati geografici deve avere i seguenti ulteriori campi obbligatori:

- **data_owner:** tipo String, nominativo del referente del dato
- **data_origin:** tipo String, fonte di provenienza del dato
- **internal_ref_code:** tipo String, riferimento interno dell'entità
- **external_ref_code:** tipo String, riferimento ad eventuali database/sistemi di archiviazione esterni
- **status:** tipo String, visibilità del dato da parte degli utenti visualizzatori (i.e. in bozza, in validazione, pubblicato)
- **shared_enea:** tipo boolean, se vero, il dato viene condiviso con il database centralizzato di Enea

Nella figura sottostante (Fig. 4) il diagramma E-R rappresentativo della piattaforma Kore ad oggi online sull'infrastruttura ENEA.

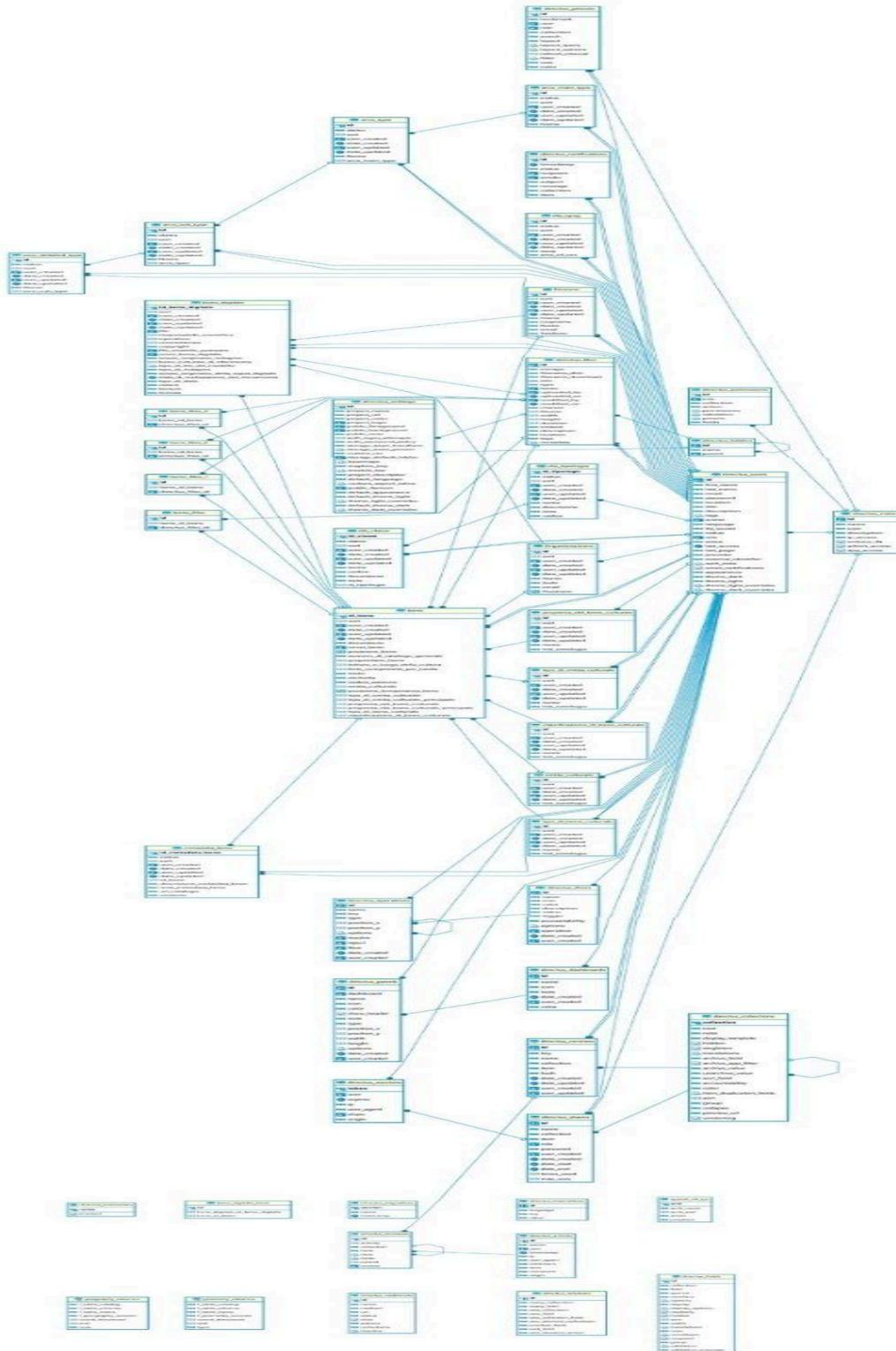


Fig.4- Schema entità e Relazioni², relativo alla piattaforma Kore ad oggi su piattaforma ICT ENEA. Per i dati di tipo geografico è obbligatorio l'uso del tipo di dato "Geometry" così da garantire la lettura corretta del dato geografico.

² ER_DTECH_01032024.pdf

5. Struttura repository

I repository dati sono organizzati per Ente/struttura e, all'interno, sono suddivisi per tipologia del dato (i.e. Shape file, Immagini, Nuvole di punti, etc...), l'accesso agli utenti è garantito attraverso la piattaforma. Il tempo di retain dei dati dovrà essere valutato in funzione delle singole installazioni.

6. Servizio di Backup

La piattaforma "Kore" ha una funzione di backup dei dati e dei metadati presenti sul database e sullo storage S3. Tale funzione in relazione alle modalità di erogazione della piattaforma "Kore" scelta da parte dei diversi clienti/fruitori (on-premise o in-cloud) è centralizzata da un servizio SASS (Software as a service) oppure demandata ai relativi ambienti dove sarà caricata la piattaforma. S Il servizio di backup svolge un ruolo critico nel garantire la resilienza del sistema, assicurando che i dati archiviati nel database siano protetti da perdite accidentali o malfunzionamenti dell'infrastruttura. Utilizzando soluzioni di backup robuste e automatizzate, il servizio di backup della piattaforma Kore assicura la continuità operativa del progetto D-TECH, consentendo un rapido ripristino dei dati in caso di necessità. Inoltre, il processo di backup viene ottimizzato per ridurre al minimo l'impatto sulle prestazioni del sistema, garantendo che l'attività principale della piattaforma non venga compromessa durante le operazioni di backup e ripristino. Il full backup di tutta la cella enea.it (circa 40 TB) viene effettuato ogni 6 mesi, inoltre, vengono effettuati backup incrementali quotidiani fatti la prima volta rispetto al full semestrale, e le volte successive rispetto al backup del primo del mese.

7. Modalità di Deploy

Il processo di deploy dell'infrastruttura della piattaforma Kore è progettato per essere flessibile e adattabile alle diverse esigenze dei clienti. Questo deployment può avvenire sia su piattaforme di Public Cloud, come Google Cloud, Microsoft Azure, Amazon AWS e altre, offrendo agli utenti la possibilità di beneficiare delle infrastrutture altamente scalabili e gestite offerte da questi provider. Allo stesso tempo, il sistema è in grado di adattarsi anche ai requisiti specifici dei clienti che preferiscono mantenere il controllo dei propri dati e delle proprie risorse, attraverso sistemi On Premise, purché soddisfino i requisiti minimi di sistema stabiliti.

Le istruzioni dettagliate per il deploy, insieme al codice sorgente correlato, sono ospitate su repository dedicati su GitHub, forniti direttamente dalla committenza del progetto. Questa scelta consente una maggiore trasparenza e collaborazione tra gli sviluppatori, oltre a facilitare la gestione del codice sorgente e delle versioni. Inoltre, la presenza dei repository su GitHub permette agli sviluppatori di contribuire al miglioramento continuo del sistema attraverso feedback, suggerimenti e correzioni di bug, promuovendo così una cultura di sviluppo open source e collaborativo.

8. Descrizione interfaccia piattaforma



Fig.5- La schermata Home della piattaforma D-TECH

La piattaforma è stata concepita per accogliere utenti registrati, pertanto l'utente dovrà accreditarsi e richiedere accesso³.

Nella home page sono presenti tutti i riferimenti al progetto e ai partner con i diversi loghi.

All'interno della piattaforma la pagina è divisa in due parti, a sinistra un menù mostra le voci che descrivono i contenuti principali:

- Bene Digitale - contiene l'elenco delle copie digitali dei beni culturali;
- Bene - contiene l'elenco dei beni culturali a cui si riferiscono le differenti copie digitali;
- Arco - contiene la struttura dell'ontologia di riferimento (ARCO ONTOLOGY), sulla base della quale ogni copia digitale o bene culturale censito nella piattaforma deve essere classificato.
- Organizzazione - contiene l'elenco di enti, musei, conservatorie e altre tipologie di organizzazioni che a vario titolo sono state censite e inserite nella piattaforma.
- Persone - contiene l'elenco delle persone che, a vario titolo (proprietari, gestori ecc.) sono state censite e inserite nella piattaforma.

A destra vengono visualizzati i contenuti selezionati dal menù sopra citato.

³ D-TECH-digital-twin@outlook.it

Bene digitale

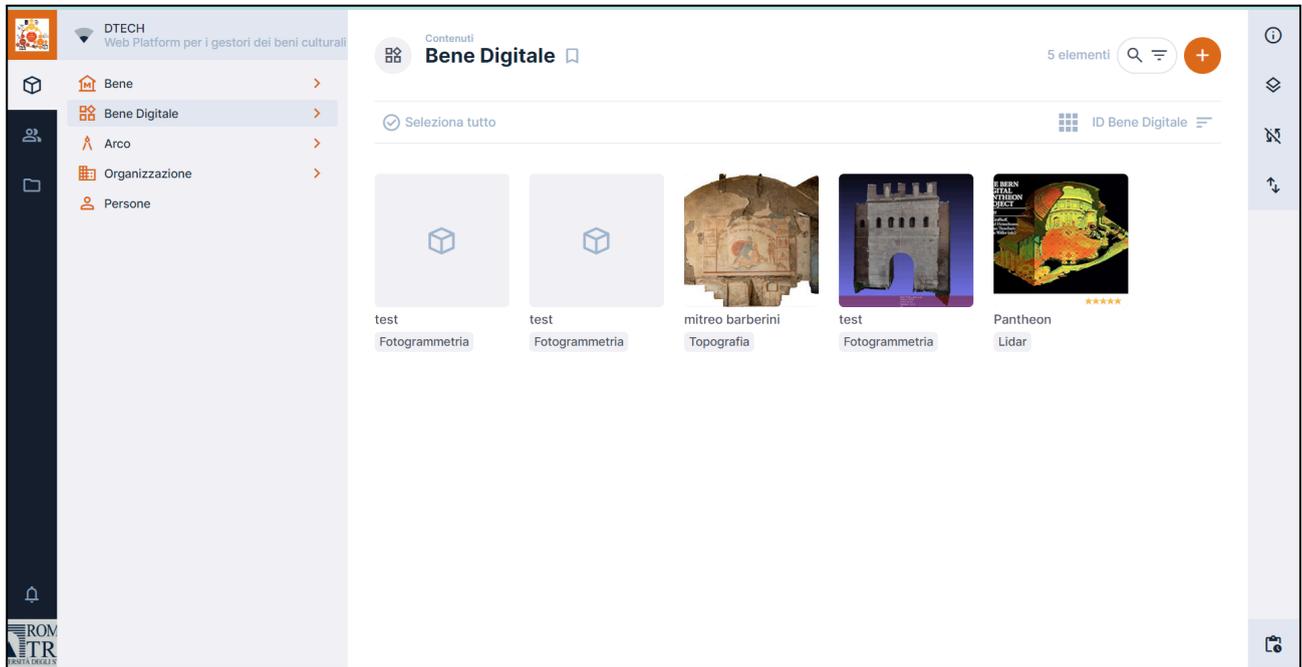


Fig.6- L'elenco dei beni digitali, visualizzato attraverso un'anteprima della copia del bene, mostra due didascalie: il nome della copia digitale e il tipo di indagine che lo ha prodotto (Lidar, Fotogrammetria, etc.)

Bene culturale

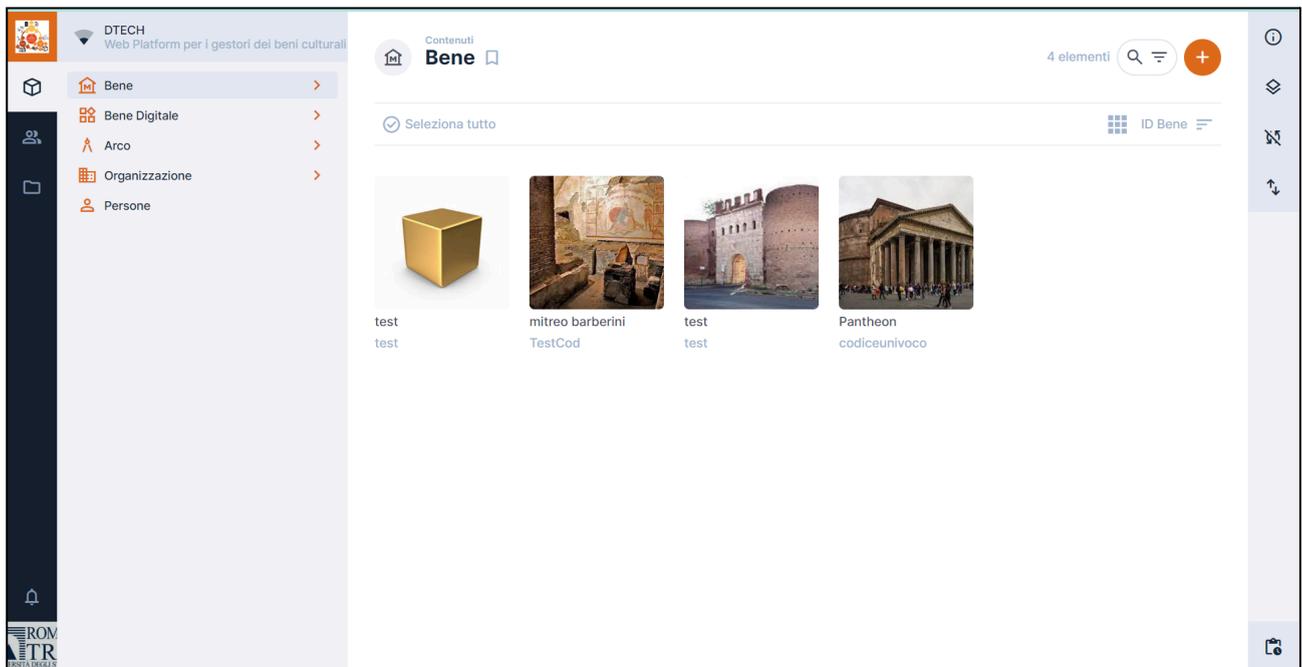


Fig.7- L'elenco dei beni culturali, visualizzato attraverso un'anteprima fotografica, mostra due didascalie, il nome del bene e una sua breve descrizione.

Arco

Nome	Link Ontologia
bene archeologico	https://w3id.org/arco/ontology/arco/ArchaeologicalProperty
bene architettonico paesaggistico	https://w3id.org/arco/ontology/arco/ArchitecturalOrLandscapeHeritage
bene storico artistico	https://w3id.org/arco/ontology/arco/HistoricOrArtisticProperty

Fig.8- Contiene la struttura dell'ontologia di riferimento, ogni oggetto o bene culturale censito nella piattaforma deve essere associato ad alcuni concetti presenti nell'ontologia ARCO (<http://wit.istc.cnr.it/arco>) e nelle ontologie collegate.

I concetti estratti dall'ontologia, utilizzati con i rispettivi link, sono stati descritti nel deliverable D2.B e fanno riferimento sia ai beni culturali sia alle copie digitali degli stessi. Nelle tabelle seguenti il dettaglio dei concetti a cui vengono collegate le singole istanze.

caratteristiche del bene culturale di riferimento

- culturalentity - <http://dati.beniculturali.it/cis/Culturalentity>
- bene culturale - <https://w3id.org/arco/ontology/arco/CulturalProperty>
- Etichetta/label - <https://w3id.org/arco/ontology/movable-property/Label>
- label - <https://www.w3.org/2000/01/rdf-schema#label>
- titolo - literal - <http://purl.org/dc/elements/1.1/title>
- descrizione - <https://w3id.org/arco/ontology/core/description>
- codice univoco - <https://w3id.org/arco/ontology/arco/uniqueIdentifier>
- numero di catalogo generale - <https://w3id.org/arco/ontology/arco/catalogueNumber>
- parte componente di bene culturale <https://w3id.org/arco/ontology/arco/CulturalPropertyComponent>
- bene materiale - <https://w3id.org/arco/ontology/arco/TangibleCulturalProperty>
- bene archeologico - <https://w3id.org/arco/ontology/arco/ArchaeologicalProperty>
- bene mobile - <https://w3id.org/arco/ontology/arco/MovableCulturalProperty>
- bene immobile - <https://w3id.org/arco/ontology/arco/ImmovableCulturalProperty>
- bene architettonico o paesaggistico - <https://w3id.org/arco/ontology/arco/ArchitecturalOrLandscapeHeritage>
- Bene Storico o Artistico - <https://w3id.org/arco/ontology/arco/HistoricOrArtisticProperty>
- Istituto e luogo della cultura - <http://dati.beniculturali.it/cis/CulturalInstituteOrSite>

nome istituzionale luogo cultura - <http://dati.beniculturali.it/cis/institutionalCISName>
 Ente competente per tutela - <https://w3id.org/arco/resource/Role/HeritageProtectionAgency>

caratteristiche della copia digitale del bene culturale

indagine - <https://w3id.org/arco/ontology/context-description/Survey>
 ha un responsabile scientifico - <https://w3id.org/arco/ontology/context-description/hasSurveyScientificDirector>
 ha un operatore - <https://w3id.org/arco/ontology/context-description/hasActivityOperator>
 tipo di indagine - <https://w3id.org/arco/ontology/context-description/SurveyType>
 indagine archeometrica e diagnostica -
<https://w3id.org/arco/ontology/context-description/ArchaeometricDiagnosticSurveyType>
 Committenza - <https://w3id.org/arco/ontology/context-description/Commission>
 Agente - <https://w3id.org/italia/onto/I0/Agent>
 è committenza di (ObjectProperty)
<https://dati.beniculturali.it/lodview-arco-onto/ontology/context-description/isCommissionOf>
 Data - <https://w3id.org/italia/onto/TI/date>
 point cloud geometry - <https://w3id.org/gom#PointCloudGeometry>
 Mesh geometry - <https://w3id.org/gom#MeshGeometry>
 photogrammetry - <http://www.arpenteur.org/ontology/Arpenteur.owl#Photogrammetry>

Organizzazione

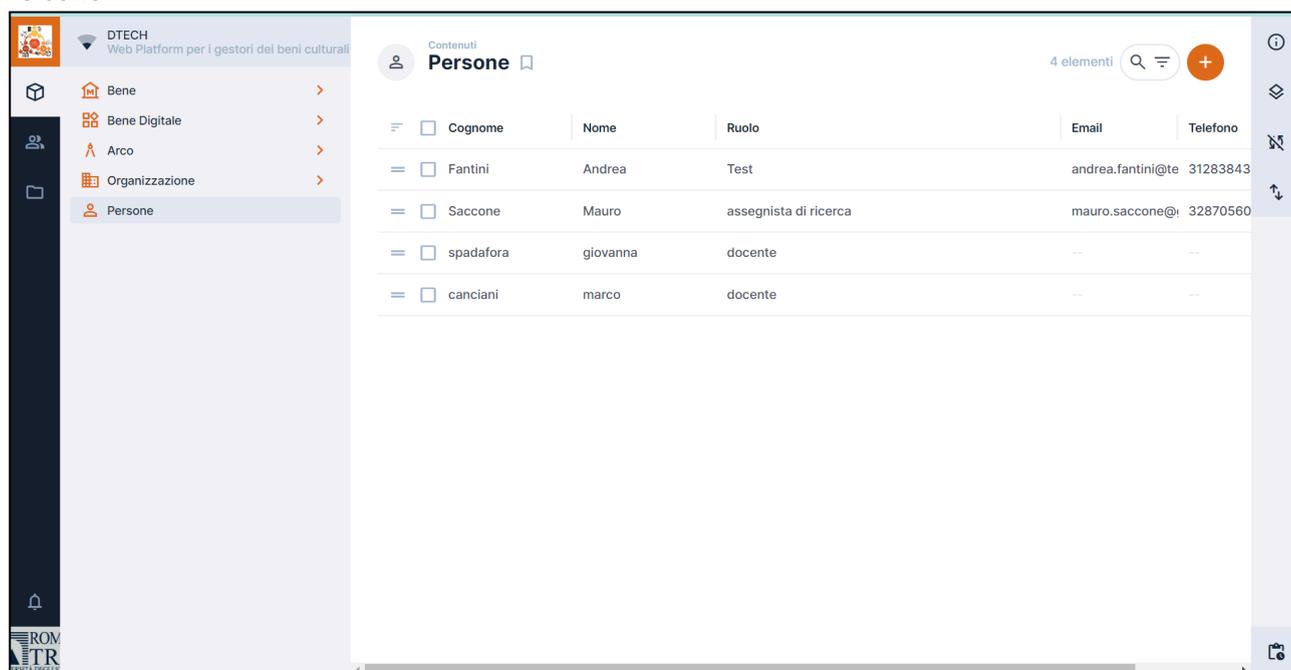
The screenshot shows the 'Organizzazione' section of the DTECH web platform. The interface includes a sidebar with navigation options like 'Bene', 'Bene Digitale', 'Arco', 'Organizzazione', and 'Persone'. The main content area displays a table with 4 elements, showing details for various organizations including their names, email addresses, and addresses.

Nome	Email	Sede
Tecnostudi Ambiente	info@tecnost	Via Test 10
barberini	barberini@gfr	Via delle Quattro Fontane, 13, 00184 Ron
Dipartimento Architettura Università degli studi Roma Tre		via Aldo Manuzio, Roma
Soprintendenza Speciale Archeologia Belle Arti e Paesag		Piazza del Cinquecento, 67 - Roma

Fig.9- L'elenco delle organizzazioni censite contiene una tabella che permette di visualizzare tutti gli enti che si sono accreditati ed hanno inserito dati nella piattaforma o gli enti che sono proprietari, o gestori, o responsabili della tutela del bene che è stato inserito.

L'obiettivo di questa raccolta è quello di costruire un database di soggetti che a vario titolo operano sui beni culturali per associare ad essi, le copie digitali e gli operatori che inseriscono, gestiscono e visualizzano i dati contenuti nella piattaforma.

Persone



Cognome	Nome	Ruolo	Email	Telefono
Fantini	Andrea	Test	andrea.fantini@te	31283843
Saccone	Mauro	assegnista di ricerca	mauro.saccone@	32870560
spadafora	giovanna	docente	--	--
canciani	marco	docente	--	--

Fig.10- L'elenco delle persone censite, il ruolo e i recapiti

L'obiettivo di questa raccolta è quello di costruire un database di soggetti che operano sui beni culturali (siano essi operatori, proprietari o gestori dei dati) in modo da poter associare ad essi, ogni volta che sia necessario, il bene e/o la sua copia digitale.