

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA "TULLIO LEVI-CIVITA"

CORSO DI LAUREA IN INFORMATICA



**Analisi, progettazione e sviluppo del backend
di un'applicazione web per la gestione di
eventi**

Tesi di laurea

Relatore

Prof.Davide Bresolin

Laureando

Alberto Lazari

ANNO ACCADEMICO 2021-2022

Sommario

La tesi descrive il lavoro svolto durante il periodo di stage, della durata di circa trecento ore, presso la sede di Treviso di Moku S.r.l., il cui obiettivo era la reimplementazione del backend di una piattaforma di gestione di eventi, sfruttando gli strumenti tipicamente utilizzati nei progetti dell'azienda.

In particolare, i seguenti capitoli tratteranno del contesto lavorativo dell'azienda, dell'analisi svolta sullo stato della piattaforma ad inizio stage, della progettazione e successiva implementazione iniziale del nuovo backend, focalizzando l'attenzione sulle scelte stilistiche e architetturelle perseguite.

Ringraziamenti

Voglio ringraziare il Prof. Davide Bresolin, per l'interesse, il supporto e l'aiuto fornito durante il periodo di stage e di stesura di questa tesi.

Ringrazio i miei genitori e tutti i miei famigliari per il supporto e l'affetto che mi hanno donato durante questi anni di studio.

Ringrazio i colleghi di Moku che mi hanno accolto calorosamente tra loro durante la mia esperienza di stage, in particolare Riccardo e Nicolò, per avermi sempre fornito l'aiuto che cercavo durante il mio lavoro.

Ringrazio la Comunità Capi del Mestre 2, per avermi accompagnato fin qui, infondendo in me una maturità e una consapevolezza che mi hanno permesso di raggiungere questo traguardo.

Ringrazio i miei colleghi studenti, stagisti e gli amici dei gruppi di progetto, con cui ho condiviso le difficoltà e le soddisfazioni di quest'anno.

Infine ringrazio i miei amici, che mi sono sempre stati vicini e con cui ho condiviso esperienze indimenticabili.

Padova, Luglio 2022

Alberto Lazari

Indice

1	L'azienda	1
1.1	Descrizione generale	1
1.2	Modello di sviluppo	1
2	Descrizione dello stage	3
2.1	Introduzione al progetto	3
2.2	Requisiti	3
2.3	Pianificazione	3
2.4	Tecnologie utilizzate	3
3	Analisi e refactor dei modelli	5
3.1	Introduzione	5
3.2	Modifiche significative effettuate	5
3.2.1	Utenti	5
4	Progettazione della API	7
4.1	Introduzione	7
4.2	Notazione adottata	7
4.3	Descrizione delle funzionalità esposte	7
4.3.1	Lista delle risorse	7
4.3.2	Dettagli di una risorsa	7
4.3.3	Creazione di una risorsa	7
4.3.4	Modifica di una risorsa	7
4.3.5	Eliminazione di una risorsa	7
4.4	Gestione dei permessi	8
5	Codifica	9
5.1	Modelli	9
5.1.1	Migrazioni del database	9
5.1.2	Associazioni a modelli e file	9
5.1.3	Validazioni	9
5.2	Controller	9
5.2.1	APIController	9
5.2.2	Implementazione delle action	9
5.3	Gestione dei permessi	9
5.4	Test di unità	9
6	Conclusioni	11
6.1	Raggiungimento dei requisiti	11

6.2 Valutazione personale	11
A Diagramma ER del sistema	13
Bibliografia	17

Elenco delle figure

A.1	Sezione sinistra del dettaglio del diagramma ER	14
A.2	Sezione centrale del dettaglio del diagramma ER	15
A.3	Sezione destra del dettaglio del diagramma ER	16

Elenco delle tabelle

Elenco degli esempi di codice

Capitolo 1

L'azienda

1.1 Descrizione generale



Descrizione dell'azienda: brevissima storia, divisione dei ruoli, spazi e luoghi di lavoro.

1.2 Modello di sviluppo

Modello agile, Scrum, organizzazione dei team.

Capitolo 2

Descrizione dello stage

2.1 Introduzione al progetto

Storia del progetto prima del mio arrivo, azienda che ha commissionato il progetto, descrizione dello scopo della piattaforma e del suo funzionamento, motivazioni alla base della scelta di riscrittura del backend.

2.2 Requisiti

Requisiti obbligatori, desiderabili e opzionali previsti.

2.3 Pianificazione

Divisione settimanale del lavoro dal piano di lavoro, incluse correzioni.

2.4 Tecnologie utilizzate

Descrizione della configurazione del framework Ruby on Rails utilizzata: librerie utilizzate, Postgres, AWS, API REST .

Capitolo 3

Analisi e refactor dei modelli

3.1 Introduzione

La prima attività che ho dovuto svolgere è stata la definizione dei modelli del nuovo backend. Purtroppo l'architettura già presente nel vecchio backend, soprattutto lato database, aveva diversi difetti e andava rivista interamente, quindi è stato necessario partire da un'analisi di quanto già fatto, per capire se fosse necessario effettuare del refactor nella struttura dei dati. In molti casi le entità contenevano molti attributi superflui o non adatti ad essere associati alla specifica entità, perché più adatti ad essere associati attraverso una relazione a un'altra entità.

In questo capitolo vengono riportati i cambiamenti e le scelte più significative che sono state fatte durante questa fase.

3.2 Modifiche significative effettuate

3.2.1 Utenti

La prima tabella analizzata è stata quella degli utenti, essendo

Capitolo 4

Progettazione della API

4.1 Introduzione

Spiegazione del lavoro svolto in questa fase.

4.2 Notazione adottata

Spiegazione convenzioni adottate nella descrizione degli endpoint.

4.3 Descrizione delle funzionalità esposte

Descrizione degli endpoint esposti dalla API, in generale per ogni modello e nello specifico per le eccezioni.

4.3.1 Lista delle risorse

Route index, attributi mostrati per ogni modello implementato.

4.3.2 Dettagli di una risorsa

Route show, attributi mostrati per ogni modello implementato.

4.3.3 Creazione di una risorsa

Route create.

4.3.4 Modifica di una risorsa

Route update.

4.3.5 Eliminazione di una risorsa

Route delete.

4.4 Gestione dei permessi

Permessi per le categorie di utenti per ogni controller.

Capitolo 5

Codifica

5.1 Modelli

5.1.1 Migrazioni del database

Comando rails generate e migrazione prodotta.

5.1.2 Associazioni a modelli e file

Associazioni di Active Record e Active Storage.

5.1.3 Validazioni

Validazioni sugli attributi del modello e le associazioni.

5.2 Controller

5.2.1 APIController

Descrizione dei metodi di utilità ereditati dai controller dell'API.

5.2.2 Implementazione delle action

Descrizione ed esempio di action tipiche dei controller.

5.3 Gestione dei permessi

Funzionamento e uso della gemma “Pundit” per la gestione dei permessi relativi agli endpoint dell'API, scope e metodi relativi alle action, esempio di gestione della gerarchia che andrà rivisto

5.4 Test di unità

Descrizione della gemma “RSpec”, che fornisce strumenti per lo sviluppo guidato dal comportamento (behaviour-driven development), esempi di modelli testati.

Capitolo 6

Conclusioni

6.1 Raggiungimento dei requisiti

Tabella con stato di completamento dei requisiti, con commento (dove necessario)

6.2 Valutazione personale

Messe alla prova le competenze fornite dal corso di laurea, verificata l'efficacia dei corsi e dei progetti svolti, imparato un nuovo linguaggio e framework con filosofia di sviluppo a me nuova, scoperto ambiente lavorativo aziendale con i ruoli e le dinamiche interne.

Appendice A

Diagramma ER del sistema

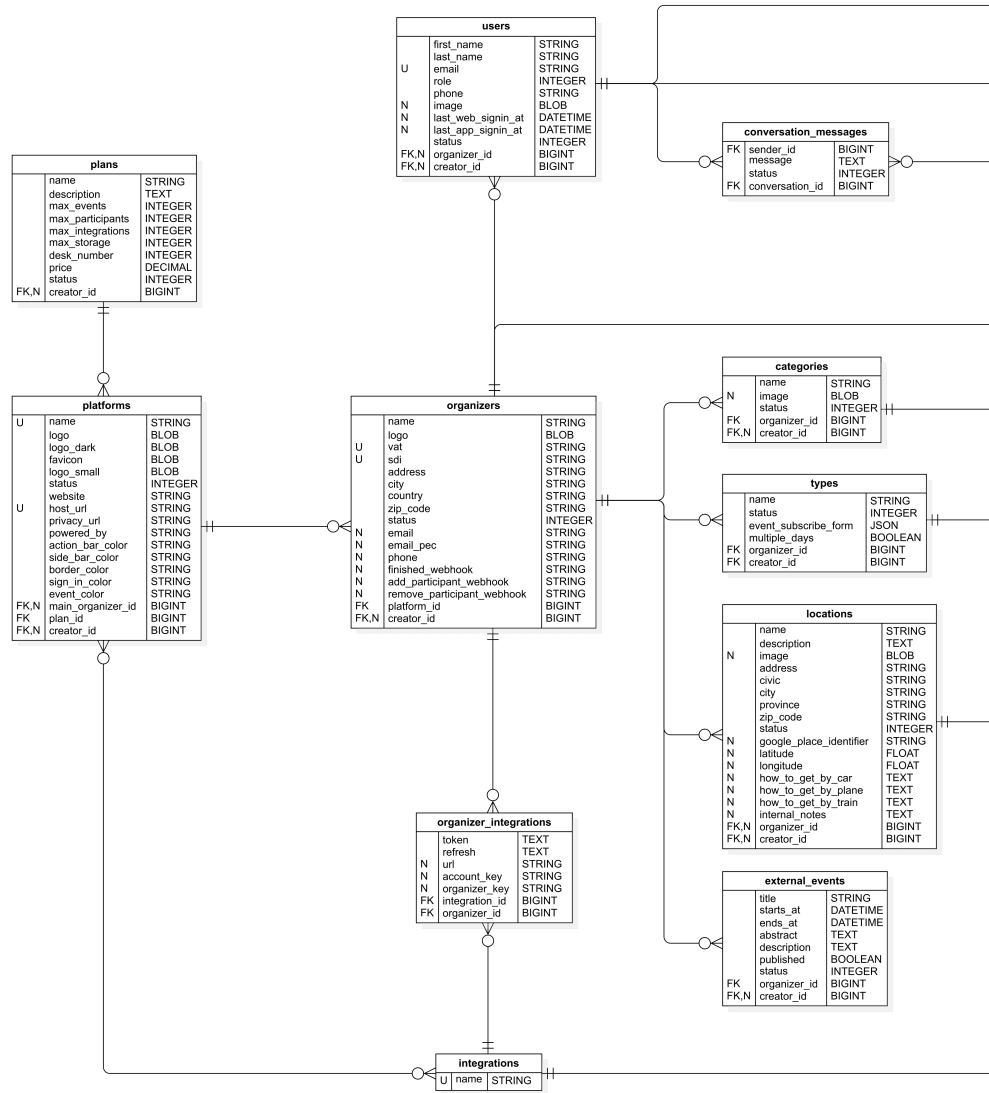


Figura A.1: Sezione sinistra del dettaglio del diagramma ER

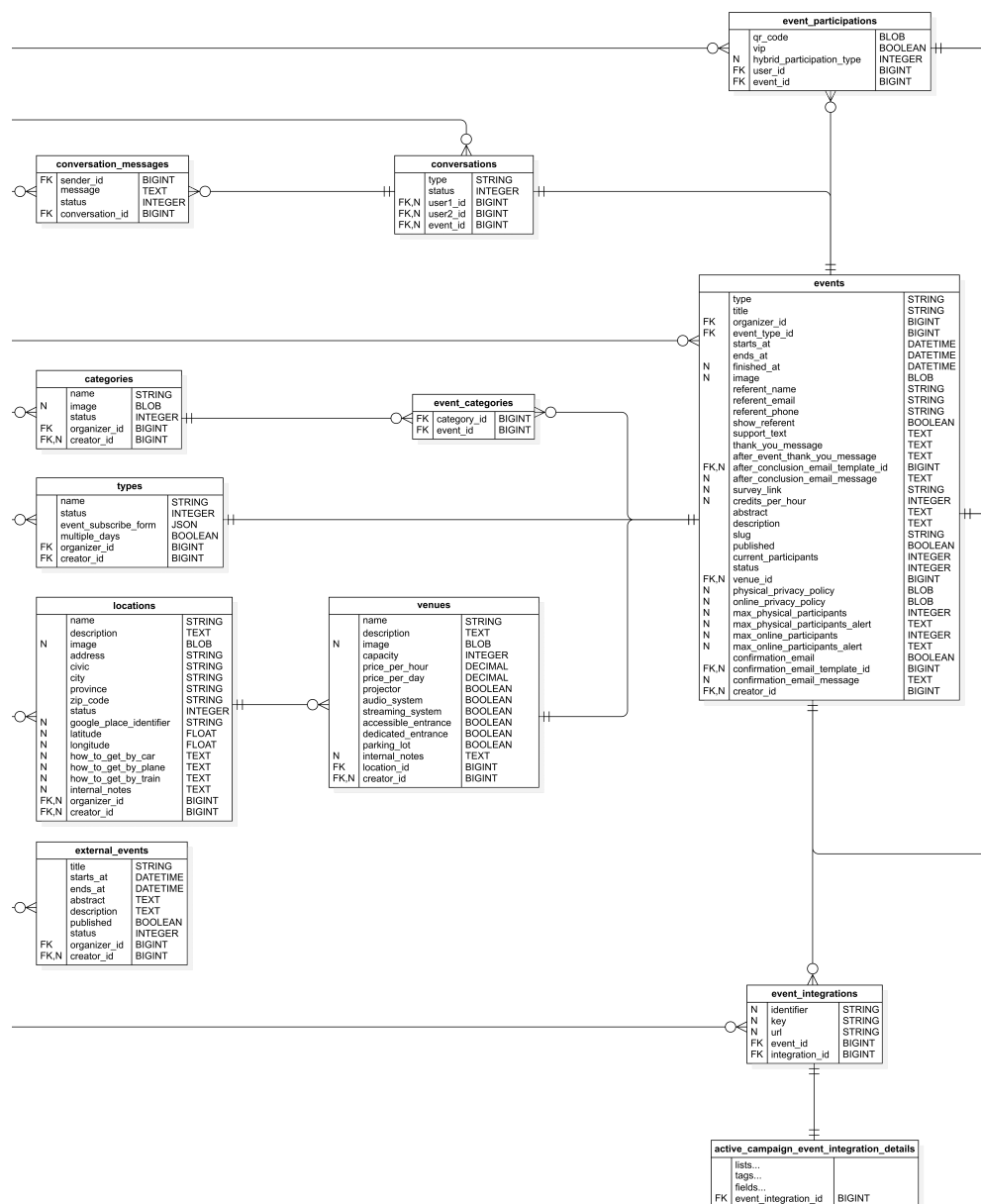


Figura A.2: Sezione centrale del dettaglio del diagramma ER

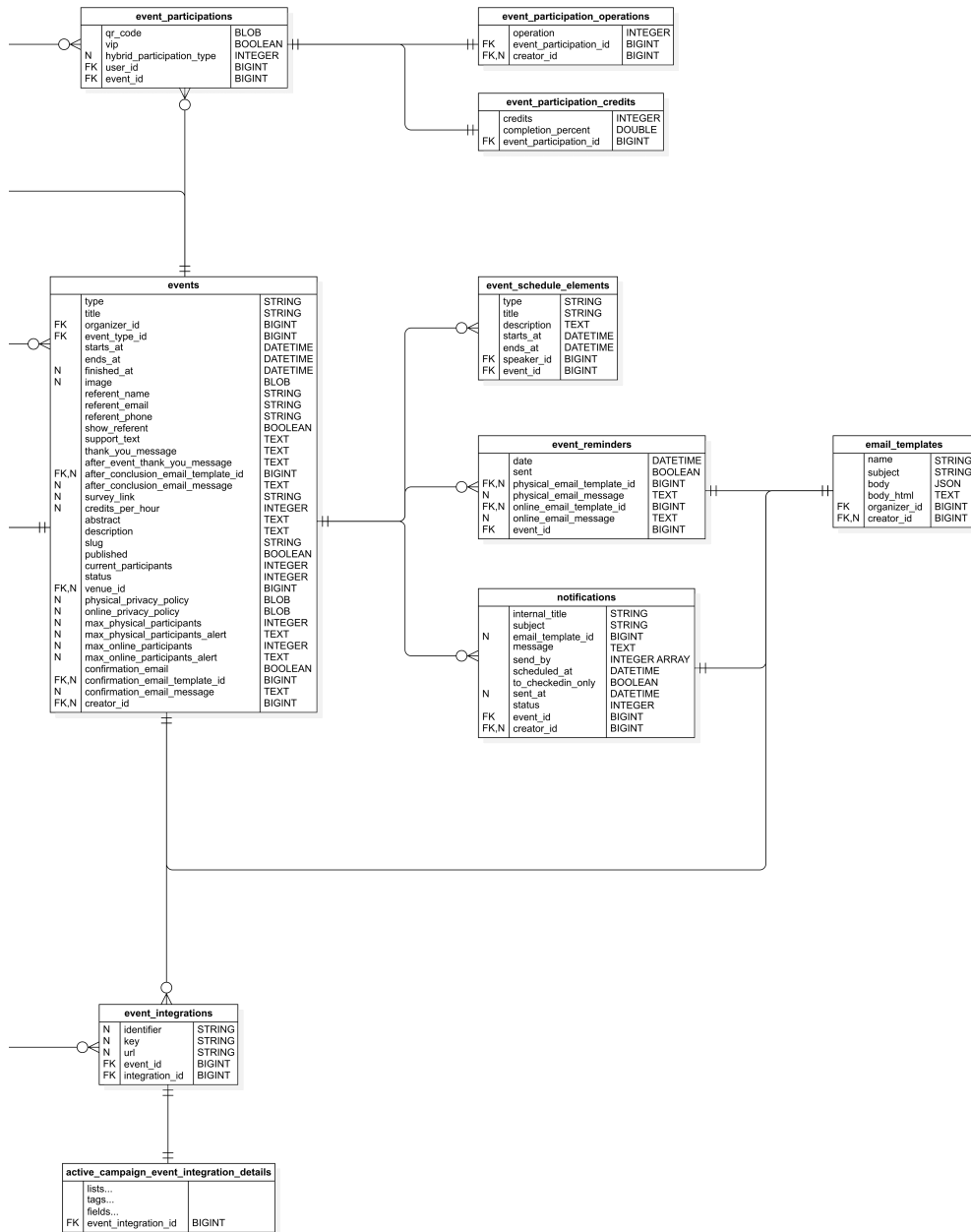


Figura A.3: Sezione destra del dettaglio del diagramma ER

Bibliografia