

# Architettura Applicazione Web



**ResumeForge**

**Marzo/Aprile 2024**

## Composizione del Team:

Il progetto è stato realizzato da studenti dell'Istituto di Istruzione Secondaria Superiore "E. Fermi":

- Riccardo William Elisabetti (Project Manager & Programmatore)
- Matteo Martino (Analista)
- Jonathan De Donno (Progettista)
- Christian Sparapane (Commerciale/Designer)

## Sommario

<b>Traccia del Progetto.....</b>	<b>2</b>
<b>Progettazione dell'applicazione.....</b>	<b>4</b>
- Tabella dei requisiti	
- Diagramma di sequenza	
- Diagramma dei Casi d'uso	
<b>Frontend.....</b>	<b>7</b>
Linguaggi di markup e programmazione:	
○ HTML	
○ CSS	
○ JavaScript	
Selezione dei Framework:	
- React	
Selezione delle Librerie:	
○ React Material UI	
○ React Routes	
○ ReactPDF	
<b>Back End.....</b>	<b>9</b>
- Firebase Hosting	
- Firebase Authentication	
- Firebase Storage	
- Firebase Firestore	

## **Traccia del Progetto n.3 - Sviluppo di un Software per la Creazione e Gestione di Curriculum Vitae:**

**Obiettivo:** L'obiettivo di questa esercitazione è sviluppare un'applicazione software che consenta agli utenti di creare, modificare e gestire curriculum vitae in modo efficace e professionale.

**Consegna del Codice:** 2 maggio 2024

**Interfaccia Utente Intuitiva:** Creare un'interfaccia utente intuitiva che guidi gli utenti attraverso il processo di creazione e modifica dei curriculum vitae. L'interfaccia utente deve consentire agli utenti di inserire le informazioni personali, l'esperienza lavorativa, l'istruzione, le competenze e altre sezioni rilevanti di un curriculum vitae.

**Gestione dei Curriculum Vitae:** Implementare le funzionalità per la gestione dei curriculum vitae, inclusa la possibilità di creare nuovi curriculum, modificare quelli esistenti e visualizzare un elenco dei curriculum salvati.

**Personalizzazione del Layout:** Fornire agli utenti la possibilità di personalizzare il layout e il design dei curriculum vitae creati, inclusa la scelta di diversi modelli predefiniti o la possibilità di creare un layout personalizzato.

**Esportazione in Formati Standard:** Implementare la funzionalità di esportazione dei curriculum vitae in formati standard come PDF, Word o file di testo. Gli utenti devono poter scegliere il formato di esportazione e ottenere un curriculum vitae pronto per la stampa o l'invio via email.

### **Relazione sulle Scelte Progettuali:**

Nella relazione, descrivere le scelte progettuali effettuate durante lo sviluppo del programma, inclusi:

**Architettura del Software:** Spiegare come è stata organizzata la struttura del software per gestire la creazione e la gestione dei curriculum vitae, inclusa la gestione dei dati e dell'interfaccia utente.

**Implementazione delle Funzionalità:** Descrivere l'implementazione delle funzionalità principali del software, inclusa la creazione, la modifica e l'esportazione dei curriculum vitae, e le scelte progettuali adottate per garantire un'esperienza utente fluida e intuitiva.

**Presentazione del Lavoro Realizzato:**

Durante la presentazione, mostrare il funzionamento del software e discutere:

**Demo dell'Applicazione:** Illustrare le principali funzionalità del software, inclusa la creazione, la modifica e l'esportazione dei curriculum vitae.

**Scelte Progettuali:** Spiegare le scelte progettuali effettuate durante lo sviluppo del software, inclusa l'architettura del software, l'interfaccia utente e le funzionalità implementate.

**Possibili Miglioramenti:** Discutere delle possibili aree di miglioramento del software e delle idee per estendere le funzionalità esistenti o aggiungere nuove caratteristiche.

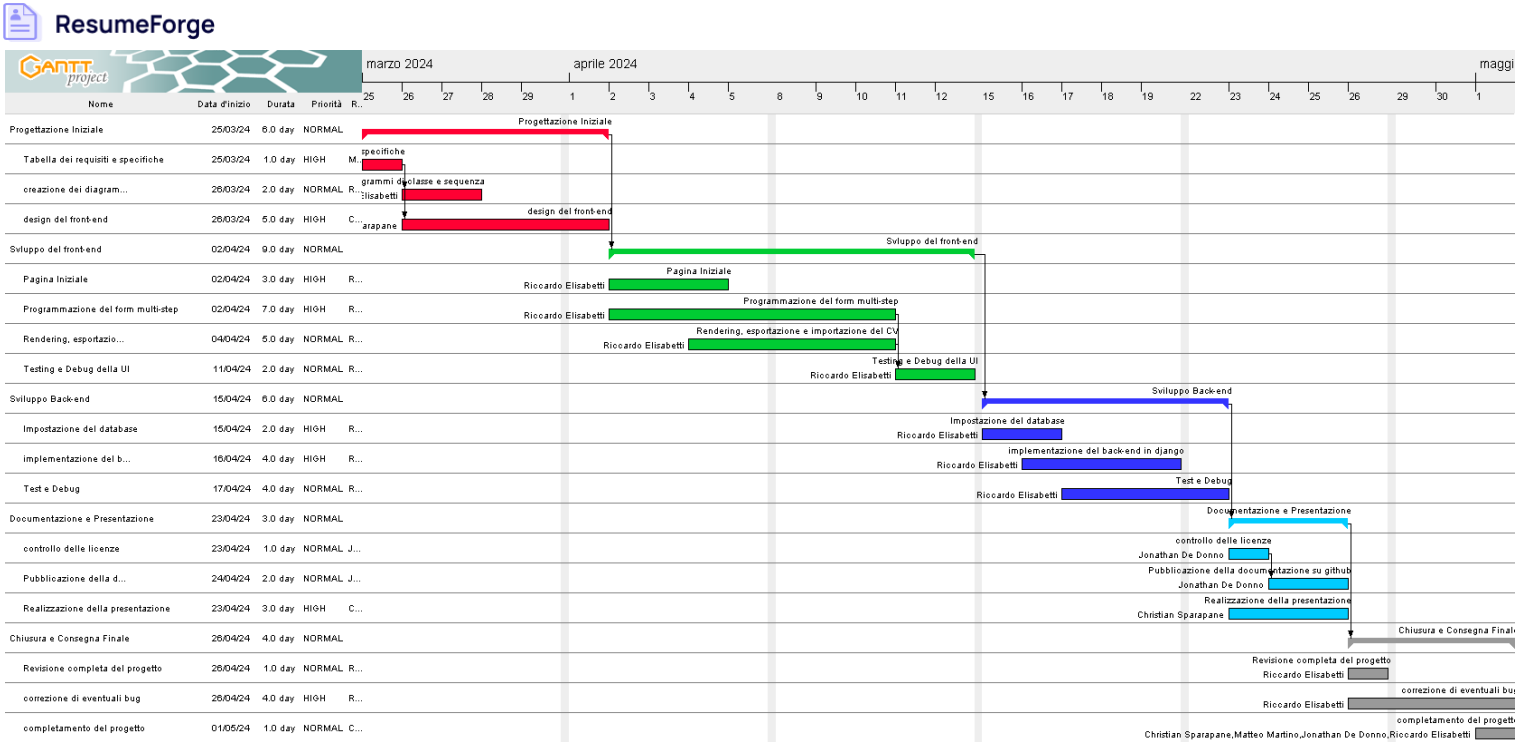
# PROGETTAZIONE DELL'APPLICAZIONE

Tabella dei Requisiti:

Requisito	Tipologia	Priorità	Definizione
1	Funzionale	MUST	L'utente può creare e accedere tramite una pagina di login
2	Funzionale	MUST	L'utente può creare un nuovo CV
3	Funzionale	SHOULD	L'utente può scegliere il layout del CV da una lista di layout predefiniti
4	Funzionale	MUST	L'utente può modificare il colore predefinito del layout del CV
5	Funzionale	MUST	L'utente può salvare il CV sul proprio account
6	Funzionale	MUST	L'utente può modificare un CV salvato sull'account
7	Funzionale	MUST	L'utente può esportare il CV in formato PDF
8	Funzionale	SHOULD	L'utente può esportare il CV in formato TXT/DOCX/Markdown/PNG
9	Funzionale	MAY	L'utente può accedere con il proprio account google
10	Funzionale	MAY	L'utente può selezionare il tema scuro
11	Funzionale	MAY	L'utente può selezionare la lingua predefinita della pagina
12	Non Funzionale	MUST	La repository di github deve contenere il codice sorgente documentato
13	Non Funzionale	SHOULD	La repository di github dovrebbe contenere un file markdown descrittivo del progetto
14	Tecnologico	MUST	Il Front End è basato su React
15	Tecnologico	MUST	Il Front End applica la libreria Material UI di React
16	Tecnologico	MUST	L'ambiente di programmazione è Vite
17	Tecnologico	MUST	Il Back-end deve essere ospitato su Firebase
18	Tecnologico	MUST	Il Front End applica la libreria react-pdf di React

- Martino Matteo: Analista

Gantt Chart:

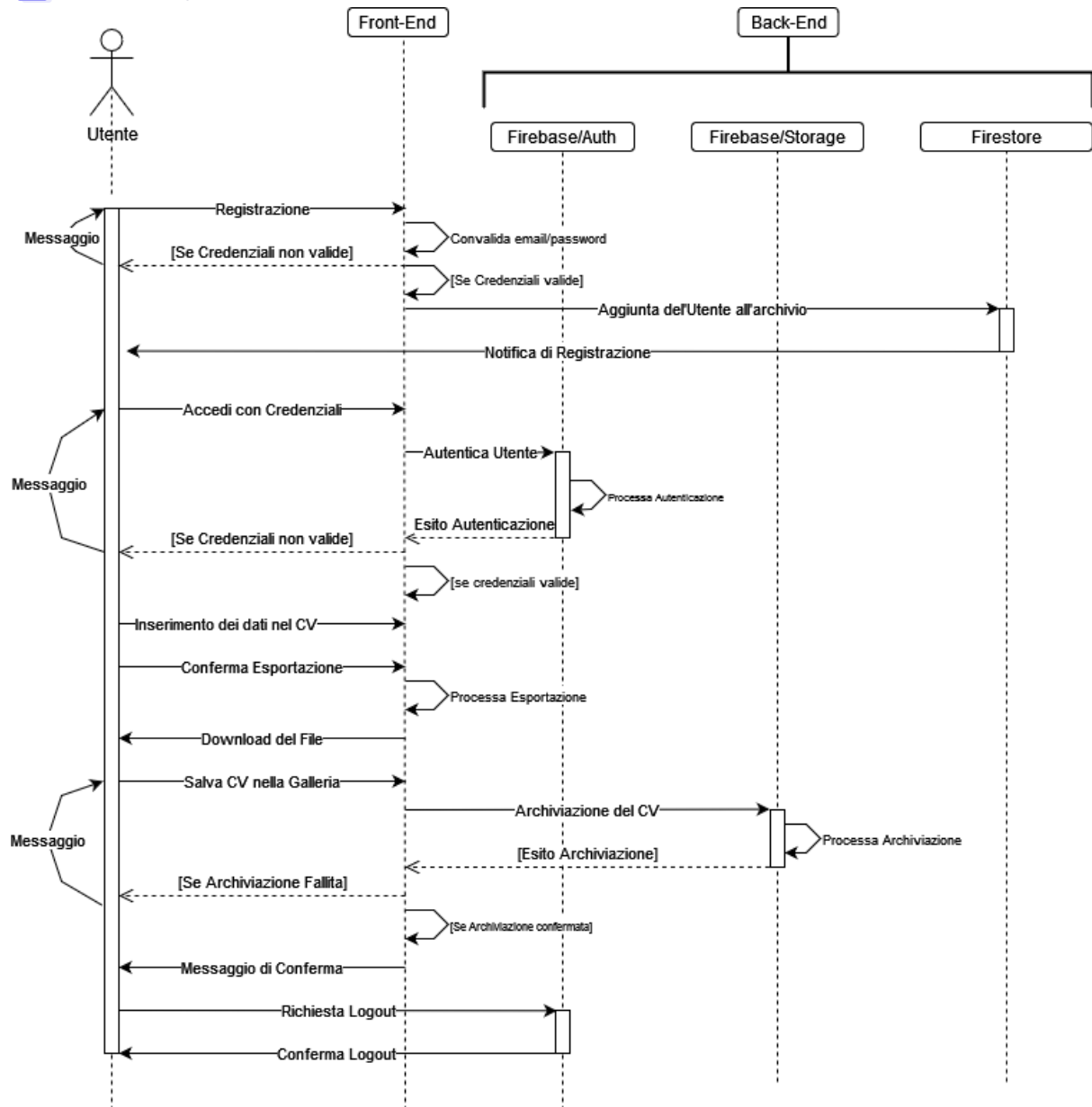


-Riccardo William Elisabetti: Project Manager

## Diagramma di sequenza:

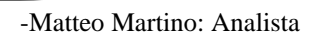


ResumeForge



-Riccardo William Elisabetti: Programmatore

 **ResumeForge**



# FRONTEND

## Strumenti di Sviluppo

- **Vite:** Vite è un server di sviluppo locale scritto da Evan You, il creatore di Vue.js, e utilizzato per impostazione predefinita da Vue e per i modelli di progetto React. Ha il supporto per TypeScript e JSX. Lo abbiamo scelto per la sua efficacia nel gestire la divisione dei progetti nelle sue varie risorse e per allinearci con la tipica suddivisione in cartelle e file di un progetto web standard.

## Linguaggi di Programmazione e Markup

- HTML: un linguaggio di markup semplice da usare ed efficace nel suo uso, ovvero delimitare la struttura generale di una pagina web formando le “colonne” o meglio “i blocchi” del design del progetto
- CSS: un linguaggio di markup estensivo usato per gestire l'estetica di ogni elemento della pagina, assieme al HTML può formare le più semplici ma moderne pagine statiche del web
- TypeScript: TypeScript è un linguaggio di programmazione open source sviluppato da Microsoft. Più nello specifico, TypeScript è un superset di JavaScript, che aggiunge tipi, classi, interfacce e moduli opzionali al JavaScript tradizionale. Si tratta sostanzialmente di una estensione di JavaScript. La differenza principale tra i due è che TypeScript è un linguaggio tipizzato che utilizza tipi statici per trovare errori e controllare i tipi in fase di compilazione. JavaScript è un linguaggio dinamico che trova gli errori in fase di esecuzione

## Selezione del Framework

- React (noto anche come React.js o ReactJS) è una libreria open-source, front-end, JavaScript per la creazione di interfacce utente. È mantenuto da Meta (già Facebook) e da una comunità di singoli sviluppatori e aziende. Abbiamo scelto questo Framework perché grazie alla sua scomposizione in “componenti”, ovvero funzioni che possono contenere degli stati o attributi, rende molto più fluido ed efficace lo sviluppo dell'applicazione web.



## Librerie Utilizzate

- **ReactPDF:** ReactPDF è una libreria che consente di generare documenti PDF usando React. È un wrapper di React intorno alla libreria PDFKit, che offre un modo semplice e dichiarativo per creare documenti PDF dinamici utilizzando componenti React. Con ReactPDF, è possibile creare facilmente documenti PDF personalizzati, inclusi report, fatture, curriculum e molto altro, tutto sfruttando la potenza e la familiarità di React.
- **React Router:** React Router è una libreria che gestisce la navigazione all'interno di un'applicazione React. Fornisce un modo per definire le rotte dell'applicazione e associare ciascuna rotta a un componente specifico. Ciò consente di mantenere lo stato dell'applicazione sincronizzato con l'URL del browser e di renderizzare dinamicamente i componenti corrispondenti in base alla rotta attiva. React Router offre anche funzionalità avanzate come il nesting delle rotte, la gestione dei parametri dinamici e il controllo della navigazione tramite programmazione.
- **React Material UI:** React Material UI è una libreria di componenti React che implementa il design system Material Design di Google. Offre una vasta gamma di componenti UI predefiniti, come bottoni, input, modali, barre di navigazione e molto altro ancora, tutti progettati seguendo le linee guida di Material Design. Questi componenti sono altamente personalizzabili e consentono di creare rapidamente interfacce utente moderne e accattivanti utilizzando React. Oltre ai componenti UI, React Material UI fornisce anche stili CSS predefiniti e temi configurabili per semplificare la gestione dell'aspetto e della sensazione dell'applicazione.

## BACK END

Il back end che noi stiamo usando per ospitare il sito, per l'autenticazione dell'account, per memorizzare i dati e per gestirli è Firebase.

I servizi di firebase che stiamo usando sono le seguenti:

- **Firebase Hosting:** è un servizio che ci permette di lavorare più comodamente con JavaScript essendo progettato per ospitare siti web e web app statiche, inoltre offre funzionalità di sicurezza avanzate come protezione dagli attacchi DDoS e anche integrazione con Google Cloud Identity and Access Management per il controllo degli accessi.
- **Firebase Authentication:** è un servizio offerto da Google Firebase che fornisce funzionalità di autenticazione e gestione delle identità per le app web e mobili, inoltre offre diversi metodi di autenticazione sicura, inclusi l'email/password, l'autenticazione tramite provider di social media come Google, Facebook, Twitter e altri, ci permette anche di offrire funzionalità per la gestione degli utenti, inclusa la possibilità di aggiungere, rimuovere e aggiornare gli account utente.
- **Firestore:** Firestore è un database NoSQL completamente gestito offerto da Google come parte della sua piattaforma di sviluppo Firebase. È progettato per essere scalabile, affidabile e facile da usare, consentendo agli sviluppatori di creare rapidamente e distribuire applicazioni con un backend cloud potente e flessibile.
- **Firebase Storage:** fornisce un sistema di archiviazione di file basato su cloud, progettato per l'archiviazione e la gestione dei dati multimediali come immagini, video, audio, file PDF e altro ancora. Firebase Storage offre funzionalità avanzate di controllo degli accessi e sicurezza dei dati, consentendo agli sviluppatori di impostare facilmente le autorizzazioni di accesso per i singoli file o per gruppi di file.

