



# JustCook

03/12/22

**SVILUPPO APPLICAZIONE**

---

# Indice

## SCOPO DEL DOCUMENTO

1. **USER FLOWS**
2. **APPLICATION IMPLEMENTATION AND DOCUMENTATION**

### 2.1. **PROJECT STRUCTURE**

La struttura del progetto si divide in tre cartelle: api, ui, Photos. La cartella api serve per gestire le API locali, la cartella ui contiene tutto il materiale per il front-end, la cartella Photos memorizza le immagini di supporto necessarie. Questa suddivisione è visibile in figura 3.

### 2.2. **PROJECT DEPENDENCIES**

Oltre ai moduli Node standard che vengono scaricati al momento dell'installazione, sono stati utilizzati altri moduli e aggiunti al file Package.Json. Quest'ultimi sono:

- Express
- MongoDB

### 2.3. **PROJECT DATA OR DB**

---

## 2.4. PROJECT APIs

In questo paragrafo vengono presentate le API dell'applicazione JustCook. Prima vengono descritte le risorse API attraverso la lingua italiana e dei digrammi in Unified Modeling Language (UML). Quest'ultimi sono il Resources Extraction Diagram (dal diagramma delle classi) e il Resources Model. Successivamente viene mostrato il codice con cui vengono implementate alcune delle API descritte.

### 2.4.1. RESOURCES EXTRACTION

In questa sezione vengono presentate e descritte le risorse API ideate a partire dai metodi delle classi del Class Diagram (analizzato nel documento precedente). Successivamente viene mostrato il Resources Extraction Diagram.

#### Descrizione risorse estratte

##### 1. PAGINA REGISTRAZIONE - METODO GET CREDENZIALI

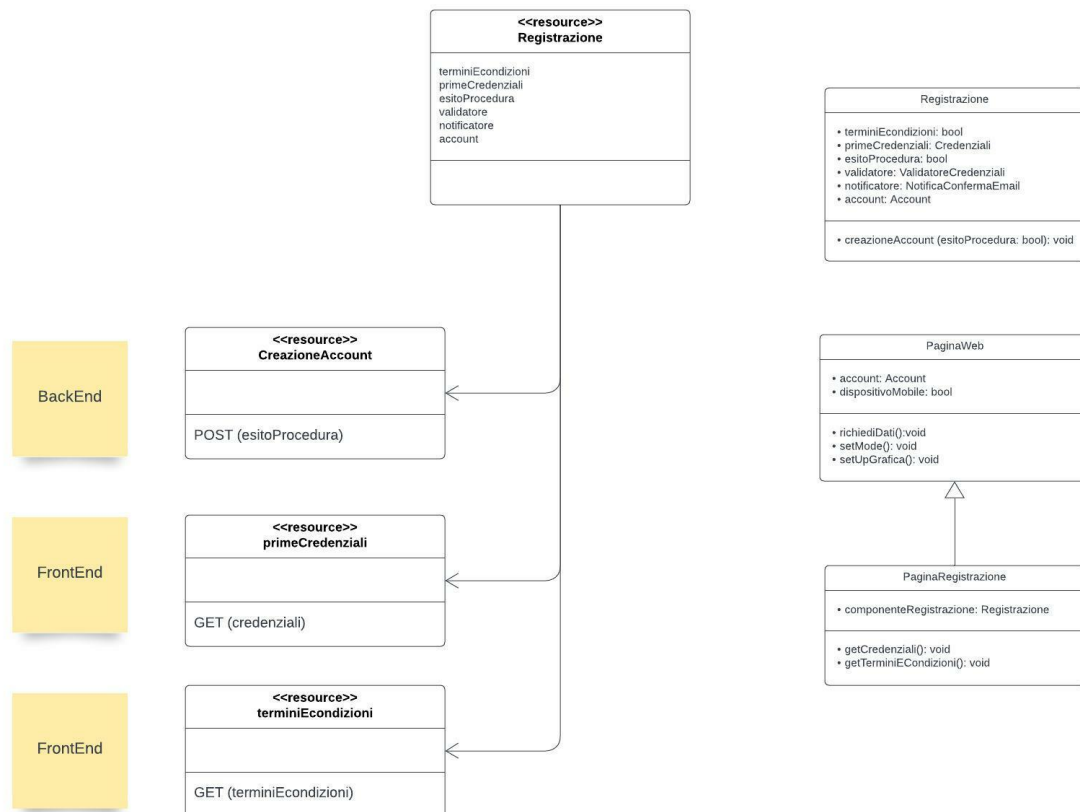
Analizzando la classe `PaginaRegistrazione` e il metodo `getCredenziali` è stata ideata l'API **Credenziali iniziali**. L'utente nella fase di creazione del proprio account deve inserire l'username, la password e l'indirizzo email associato in delle apposite barre. Per ottenere le prime credenziali appena inserite, l'API analizzata possiede il metodo HTTP GET.

##### 2. PAGINA REGISTRAZIONE - METODO GET TERMINI E CONDIZIONI

Analizzando la classe `PaginaRegistrazione` e il metodo `getTerminiECondizioni` è stata ideata l'API **Termini e condizioni**. L'utente per creare un proprio account deve accettare i termini e le condizione per il trattamento dei dati nella pagina della registrazione. Per ottenere questa informazione, l'API in considerazione utilizza il metodo HTTP GET.

##### 3. REGISTRAZIONE - METODO CREAZIONE ACCOUNT

Analizzando il metodo `creazioneAccount` della classe `Registrazione` è stata ideata l'API **Creazione account**. Quest'ultima possiede il metodo HTTP POST perchè deve creare un nuovo account per l'utente che ha effettuato la registrazione correttamente.



**Figura:** Classi e Resources Extraction Diagram fase registrazione

#### 4. ACCOUNT - GET CREDENZIALI

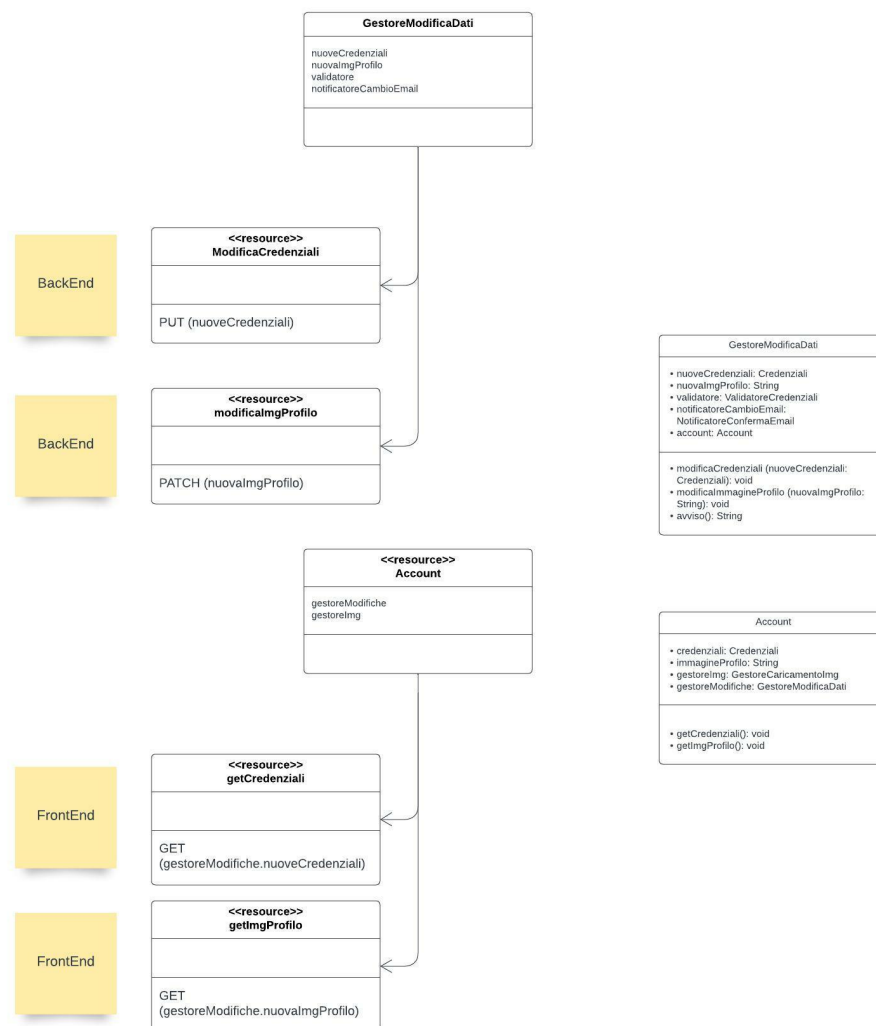
Dalla pagina dell'account è possibile modificare le credenziali (username, password, indirizzo email). Si possono cambiare inserendo i nuovi dati nelle apposite barre. Per questo è stato creato il metodo `getCredenziali` della classe `Account`. Di conseguenza è stata ideata l'API **Nuove credenziali** con il metodo HTTP `GET` per ottenere i dati.

#### 5. ACCOUNT - GET IMG PROFILO

L'utente può modificare la propria immagine profilo dalla pagina dell'account. La nuova foto viene ottenuta tramite il metodo `getImgProfilo` della classe `Account`. Per questo è stata ideata l'API **Nuova img profilo** con il metodo HTTP `GET`.

## 7. GESTORE MODIFICA DATI - MODIFICA IMG PROFILO

Analizzando la classe `GestoreModificaDati` e il metodo `modificaImgProfilo` è stata ideata l'API **Modifica Immagine Profilo**. L'utente nella propria pagina dell'account può modificare la foto in questione. Per cambiarla l'API considerata utilizza il metodo `HTTP PATCH`.



**Figura:** Classi e Resources Extraction Diagram fase modifica dati pagina account

- 
3. **API DOCUMENTATION**
  4. **FRONTEND IMPLEMENTATION**
  5. **GITHUB REPOSITORY AND DEPLOYMENT INFO**
  6. **TESTING**