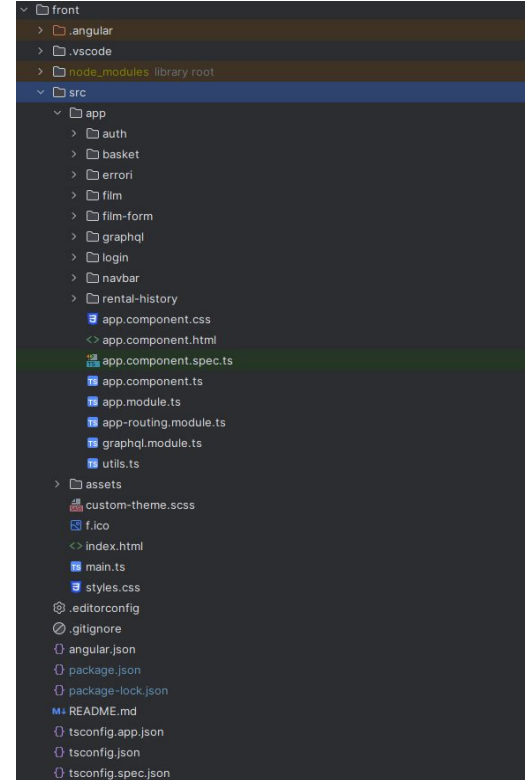
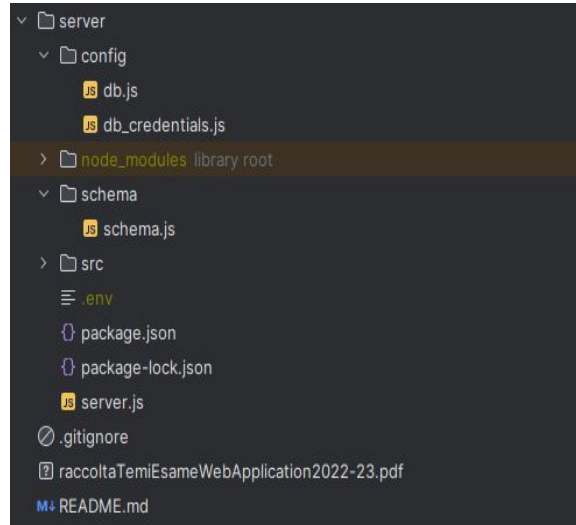


# FilmFinder

**Progetto di:**  
L.Canaia, E.Lonardi

# Sviluppo/Progettazione

- sviluppata secondo il modello SPA 2 tiers
- alla base vige il principio “implementation over configuration”
- divisione dei compiti anche in base al modello adottato
- metodo di sviluppo “scrum impuro”
- utilizzo iniziale di mock dell’api per velocizzare lo sviluppo indipendente
- utilizzo strategia GitFlow per versionamento



# Strumenti utilizzati per lo sviluppo

- IDE utilizzati : WebStorm, VisualStudioCode
- Test API : PostMan
- Strumenti versionamento : GitKraken, GitQlient
- Database : Dbeaver, PgAdmin
- Testing Accessibilità : WAVE Evaluation Tool, webaim.org, ChromeVox, w3.org

# Fasi di progettazione

Fasi adottate ad ogni aggiunta di una nuova feature:

- Apprendimento di tecnologie
- implementazione base
- sviluppo dell'implementazione base
- refactoring
- testing manuale
- fixing di eventuali bug

# Modelli Resi Disponibili da GraphQL - Server

QUERY:

movies(film\_title: String, film\_category: [ID], only\_available: Boolean, limit: Int, offset: Int): [Movie]

movie(film\_id: ID): Movie

categories: [Category]

pecunia\_pagata: [Payment]

basket: Basket

MUTATION:

login(email: String!, password: String!): String

register(email: String!, password: String!, customer\_id: ID!): Boolean

addToBasket(film\_id: ID!): Boolean

removeFromBasket(film\_id: ID): Boolean

rentMovies(rentObj: [RentInput!]): Boolean

# Modelli utilizzati da GraphQL - Front

```
const BASKET : TypedDocumentNode<?, ?> = gql`query Basket {
  basket {
    film {
      title
      film_id
      rental_rate
      rental_duration
      store_availability {
        address {
          district
          postal_code
        }
        store_id
      }
    }
  }
}`;

2 usages  A enricolonardi@gmail.com <enricolonardi@gmail.com>
const FILMINBASKET : TypedDocumentNode<?, ?> = gql`query FILMINBASKET {
  basket {
    film {
      film_id
    }
  }
}`;

export {BASKET, FILMINBASKET};
```

```
const CATEGORIES : TypedDocumentNode<?, ?> = gql`query{
  categories{
    category_id,
    name
  }
}`;

export {CATEGORIES};
```

```
const MOVIES : TypedDocumentNode<?, ?> = gql`query($film_title : String, $film_category : [ID], $only_available:Boolean,
$limit : Int, $offset : Int){
  movies(
    film_title : $film_title,
    film_category : $film_category,
    only_available:$only_available,
    limit: $limit,
    offset : $offset
  ){
    film_id,
    title,
    description,
    release_year,
    rating,
    rental_duration,
    rental_rate,
    length,
    category{
      name
    },
    actor{
      first_name,
      last_name
    },
    language{
      name
    },
    store_availability{
      address{
        address
      }
    }
  }
}`;

export {MOVIES};
```

```
const LOGIN : TypedDocumentNode<?, ?> = gql`mutation Login($email : String!, $password : String!){
  login(email : $email, password : $password)
}`;

2 usages  A enricolonardi@gmail.com <enricolonardi@gmail.com>
const REMOVEFROMBASKET : TypedDocumentNode<?, ?> = gql`mutation RemoveFromBasket($film_id : ID){
  removeFromBasket(film_id: $film_id)
}`;

2 usages  A enricolonardi@gmail.com <enricolonardi@gmail.com>
const ADDTOBASKET : TypedDocumentNode<?, ?> = gql`mutation AddToBasket($film_id : ID){
  addToBasket(film_id: $film_id)
}`;

2 usages  A enricolonardi@gmail.com <enricolonardi@gmail.com>
const RENTMOVIES : TypedDocumentNode<?, ?> = gql`mutation RentMovies($rentInput : [RentInput!] ) {
  rentMovies(rentObj: $rentInput)
}`;

export {LOGIN, REMOVEFROMBASKET, ADDTOBASKET, RENTMOVIES};
```

```
const RENTAL_HISTORY : TypedDocumentNode<?, ?> = gql`query{
  pecunia_pagate{
    rental{
      rental_date,
      inventory{
        film{
          title
        }
      }
      return_date
    },
    amount,
    payment_date
  }
}`;

export {RENTAL_HISTORY};
```

# Accessibilità - AA WCAG 2.1

Processo di  
validazione  
tramite  
controllo di  
tutte le  
norme vigenti  
nel sito  
ufficiale  
w3.org

Aa Name	films	basket	login	navbar	rental-history
<a href="#">Non-text Content : aria-label</a>	Done	Done	Done	Done	Done
<a href="#">Audio-only/video-only</a>	Done	Done	Done	Done	Done
<a href="#">Captions (Prerecorded) Level A</a>	Done	Done	Done	Done	Done
<a href="#">1.2.3Audio Description or Media Alternative (Prerecorded) Level A</a>	Done	Done	Done	Done	Done
<a href="#">Captions (Live) Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Audio Description (Prerecorded) Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Info and Relationships Level A</a>	Done	Done	Done	Done	Done
<a href="#">Meaningful Sequence Level A</a>	Done	Done	Done	Done	Done
<a href="#">Sensory Characteristics Level A</a>	Done	Done	Done	Done	Done
<a href="#">Orientation Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Identify Input Purpose Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Use of Color Level A</a>	Done	Done	Done	Done	Done
<a href="#">Audio Control Level A</a>	Done	Done	Done	Done	Done
<a href="#">Contrast (Minimum) Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Resize text Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Images of Text Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Reflow Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Non-text Contrast Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Text Spacing Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Content on Hover or Focus Level AA ... chromevox</a>	Done	Done	Done	Done	Done
<a href="#">Keyboard Level A</a>	Done	Done	Done	Done	Done
<a href="#">No Keyboard Trap Level A</a>	Done	Done	Done	Done	Done
<a href="#">Character Key Shortcuts Level A</a>	Done	Done	Done	Done	Done
<a href="#">Timing Adjustable Level A</a>	Done	Done	Done	Done	Done

Aa Name	films	basket	login	navbar	rental-history
<a href="#">Pause, Stop, Hide Level A</a>	Done	Done	Done	Done	Done
<a href="#">Three Flashes or Below Threshold Level A</a>	Done	Done	Done	Done	Done
<a href="#">Bypass Blocks Level A</a>	Done	Done	Done	Done	Done
<a href="#">Page Titled Level A</a>	Done	Done	Done	Done	Done
<a href="#">Focus Order Level A</a>	Done	Done	Done	Done	Done
<a href="#">Link Purpose (In Context) Level A</a>	Done	Done	Done	Done	Done
<a href="#">Multiple Ways Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Headings and Labels Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Focus Visible Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Pointer Gestures Level A</a>	Done	Done	Done	Done	Done
<a href="#">Pointer Cancellation Level A</a>	Done	Done	Done	Done	Done
<a href="#">Label in Name Level A</a>	Done	Done	Done	Done	Done
<a href="#">Motion Actuation Level A</a>	Done	Done	Done	Done	Done
<a href="#">Language of Page Level A</a>	Done	Done	Done	Done	Done
<a href="#">Language of Parts Level AA</a>	Done	Done	Done	Done	Done
<a href="#">On Focus Level A</a>	Done	Done	Done	Done	Done
<a href="#">On Input Level A</a>	Done	Done	Done	Done	Done
<a href="#">Consistent Navigation Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Consistent Identification Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Error Identification Level A</a>	Done	Done	Done	Done	Done
<a href="#">Labels or Instructions Level A</a>	Done	Done	Done	Done	Done
<a href="#">Error Suggestion Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Error Prevention (Legal, Financial, Data) Level AA</a>	Done	Done	Done	Done	Done
<a href="#">Parsing Level A</a>	Done	Done	Done	Done	Done
<a href="#">Name, Role, Value Level A</a>	Done	Done	Done	Done	Done
<a href="#">Status Messages Level AA</a>	Done	Done	Done	Done	Done

# Tool Utilizzati

- testing con la colorblind mode tramite strumenti sviluppatore firefox
- testing aria-labels tramite chromevox
- testing dei contrasti tramite tool citato prima
- controllo eventuali errori/contrast problems tramite il tool WAVE di chrome