

# Documentazione di progetto

TRACCIA 2 – GESTIONE PREVENTIVI – RIA

CARLO CHIODAROLI

# Specifica

## Traccia 2 – Gestione preventivi

### Versione pure HTML

Un'applicazione web consente la gestione di richieste di preventivi per prodotti personalizzati. L'applicazione supporta registrazione e login di clienti e impiegati mediante una pagina pubblica con opportune form. La registrazione controlla l'unicità dello username. Un preventivo è associato a un prodotto, al cliente che l'ha richiesto e all'impiegato che l'ha gestito. Il preventivo comprende una o più opzioni per il prodotto a cui è associato, che devono essere tra quelle disponibili per il prodotto. Un prodotto ha un codice, un'immagine e un nome. Un'opzione ha un codice, un tipo ("normale", "in offerta") e un nome. Un preventivo ha un prezzo, definito dall'impiegato. Quando l'utente (cliente o impiegato) accede all'applicazione, appare una LOGIN PAGE, mediante la quale l'utente si autentica con username e password. Quando un cliente fa login, accede a una pagina HOME PAGE CLIENTE che contiene una form per creare un preventivo e l'elenco dei preventivi creati dal cliente. Selezionando uno dei preventivi il cliente ne visualizza i dettagli. Mediante la form di creazione di un preventivo l'utente per prima cosa sceglie il prodotto; scelto il prodotto, la form mostra le opzioni di quel prodotto. L'utente sceglie le opzioni (almeno una) e conferma l'invio del preventivo mediante il bottone INVIA PREVENTIVO. Quando un impiegato effettua il login, accede a una pagina HOME PAGE IMPIEGATO che contiene l'elenco dei preventivi gestiti da lui in precedenza e quello dei preventivi non ancora associati a nessun impiegato. Quando l'impiegato seleziona un elemento dall'elenco dei preventivi non ancora associati a nessuno, compare una pagina PREZZA PREVENTIVO che mostra i dati del cliente (username) e del preventivo e una form per inserire il prezzo del preventivo. Quando l'impiegato inserisce il prezzo e invia i dati con il bottone INVIA PREZZO, compare di nuovo la pagina HOME PAGE IMPIEGATO con gli elenchi dei preventivi aggiornati. Il prezzo definito dall'impiegato risulta visibile al cliente quando questi accede all'elenco dei propri preventivi e visualizza i dettagli del preventivo. La pagina PREZZA PREVENTIVO contiene anche un collegamento per tornare alla HOME PAGE IMPIEGATO. L'applicazione consente il logout dell'utente.

### Versione con JavaScript

Si realizzi un'applicazione client server web che modifica le specifiche precedenti come segue:

- La registrazione controlla la validità sintattica dell'indirizzo di e-mail e l'uguaglianza tra i campi "password" e "ripeti password", anche a lato client.
- Dopo il login, l'intera applicazione è realizzata con un'unica pagina per ciascuno dei ruoli: una pagina singola per il ruolo di cliente e una pagina singola per il ruolo di impiegato.
- Ogni interazione dell'utente è gestita senza ricaricare completamente la pagina, ma produce l'invocazione asincrona del server e l'eventuale modifica del contenuto da aggiornare a seguito dell'evento.
- Nella pagina del cliente, la scelta del prodotto comporta la successiva visualizzazione delle opzioni senza produrre un'ulteriore chiamata al server.
- L'invio del preventivo da parte del cliente deve produrre la verifica dei dati anche a lato client (almeno un'opzione scelta).
- Nella pagina dell'impiegato, il controllo del prezzo (non nullo e maggiore di zero) deve essere fatto anche a lato client.
- Eventuali errori a lato server devono essere segnalati mediante un messaggio di allerta all'interno della pagina del cliente o dell'impiegato.

# Applicazione RIA

## Appunti su specifica

Legenda: **Pagine**, **Componenti vista**, **Eventi**, **Azioni**

### Versione RIA

Un'applicazione web consente la gestione di richieste di preventivi per prodotti personalizzati. L'applicazione supporta **registrazione** e **login** di clienti e impiegati mediante una pagina pubblica con opportune form. La registrazione controlla l'unicità dello username. Un preventivo è associato a un prodotto, al cliente che l'ha richiesto e all'impiegato che l'ha gestito. Il preventivo comprende una o più opzioni per il prodotto a cui è associato, che devono essere tra quelle disponibili per il prodotto. Un prodotto ha un codice, un'immagine e un nome. Un'opzione ha un codice, un tipo ("normale", "in offerta") e un nome. Un preventivo ha un prezzo, definito dall'impiegato. Quando l'**utente** (cliente o impiegato) **accede** all'applicazione, appare una **LOGIN PAGE**, mediante la quale **l'utente si autentica** con **username e password**. Quando un **cliente fa login**, accede a una pagina **HOME PAGE CLIENTE** che contiene una **form per creare un preventivo** e **l'elenco dei preventivi creati dal cliente**. **Selezionando uno dei preventivi** il cliente ne **visualizza i dettagli**. Mediante la **form di creazione** di un preventivo l'utente per prima cosa **sceglie il prodotto**; scelto il prodotto, la **form mostra le opzioni di quel prodotto**. L'utente **sceglie le opzioni (almeno una)** e conferma **l'invio del preventivo** mediante il bottone **INVIA PREVENTIVO**. Quando un **impiegato effettua il login**, accede a una pagina **HOME PAGE IMPIEGATO** che contiene **l'elenco dei preventivi gestiti da lui** in precedenza e **quello dei preventivi non ancora associati a nessun impiegato**. Quando **l'impiegato seleziona un** elemento dall'elenco dei **preventivi non ancora associati** a nessuno, compare una pagina **PREZZA PREVENTIVO** che **mostra i dati del cliente** (username) e **del preventivo** e una **form per inserire il prezzo del preventivo**. Quando l'impiegato inserisce il prezzo e **invia i dati con il bottone INVIA PREZZO**, compare di nuovo la pagina HOME PAGE IMPIEGATO con gli elenchi dei preventivi aggiornati. Il **prezzo definito dall'impiegato** risulta visibile al cliente quando questi accede all'elenco dei propri preventivi e **visualizza i dettagli del preventivo**. La pagina PREZZA PREVENTIVO contiene anche un collegamento per tornare alla HOME PAGE IMPIEGATO. L'applicazione consente il **logout dell'utente**.

### Versione con JavaScript

Si realizzi un'applicazione client server web che modifica le specifiche precedenti come segue:

- La registrazione **controlla la validità sintattica dell'indirizzo di e-mail** e l'uguaglianza tra i **campi "password" e "ripeti password"**, anche a lato client.
- Dopo il login, l'intera applicazione è realizzata con un'unica pagina per ciascuno dei ruoli: **una pagina singola per il ruolo di cliente e una pagina singola per il ruolo di impiegato**.
- Ogni interazione dell'utente è gestita senza ricaricare completamente la pagina, ma **produce l'invocazione asincrona del server** e l'eventuale modifica del contenuto da aggiornare a seguito dell'evento.
- Nella pagina del cliente, la **scelta del prodotto** comporta la successiva visualizzazione delle opzioni senza produrre un'ulteriore chiamata al server.
- L'invio del preventivo da parte del cliente deve produrre la **verifica dei dati** anche a lato client (almeno un'opzione scelta).
- Nella pagina dell'impiegato, **il controllo del prezzo** (non nullo e maggiore di zero) deve essere fatto anche a lato client.
- Eventuali **errori a lato server devono essere segnalati** mediante un messaggio di allerta all'interno della pagina del cliente o dell'impiegato.

## Completamento delle Specifiche

- Form di login e Form di registrazione
- Form di registrazione con campi: Username, e-mail, password e ripeti password; contiene pulsanti "Submit".
- Anche un'impiegato, cliccando su un preventivo gestito in precedenza, può vederne i dettagli
- I dettagli di un preventivo appaiono sotto di esso
- Tasto di Logout in ogni pagina delle zone private
- Mostrando i dettagli del preventivo si mostra anche l'immagine del prodotto
- Username sono unici tra diversi tipi di utenti (Non si può avere un impiegato con uno stesso username di un cliente).
- Form per prezzare il preventivo appare sotto i dettagli del preventivo non prezzato nella employee page

## Sommario

- Pagina di Default
  - Form di accesso con: Username, Password e bottone di accesso
  - Form di registrazione con: Username, e-mail, password, ripeti password, tipo utente e Bottone di registrazione
- Home page cliente
  - Form di richiesta di un nuovo preventivo
    - Scelta del Prodotto, tasto conferma scelta del prodotto
    - Scelta delle opzioni, tasto di conferma scelta preventivo e tasto per annullare
  - Elenco preventivi chiesti in precedenza
  - Dettagli preventivo, una volta che se ne è selezionato uno
- Home page impiegato
  - Elenco preventivi non prezzati
  - Sezione di prezzamento preventivo nei dettagli di preventivo non prezzato
  - Elenco preventivi prezzati da lui
  - Dettagli preventivo prezzato da lui

## Eventi e Azioni

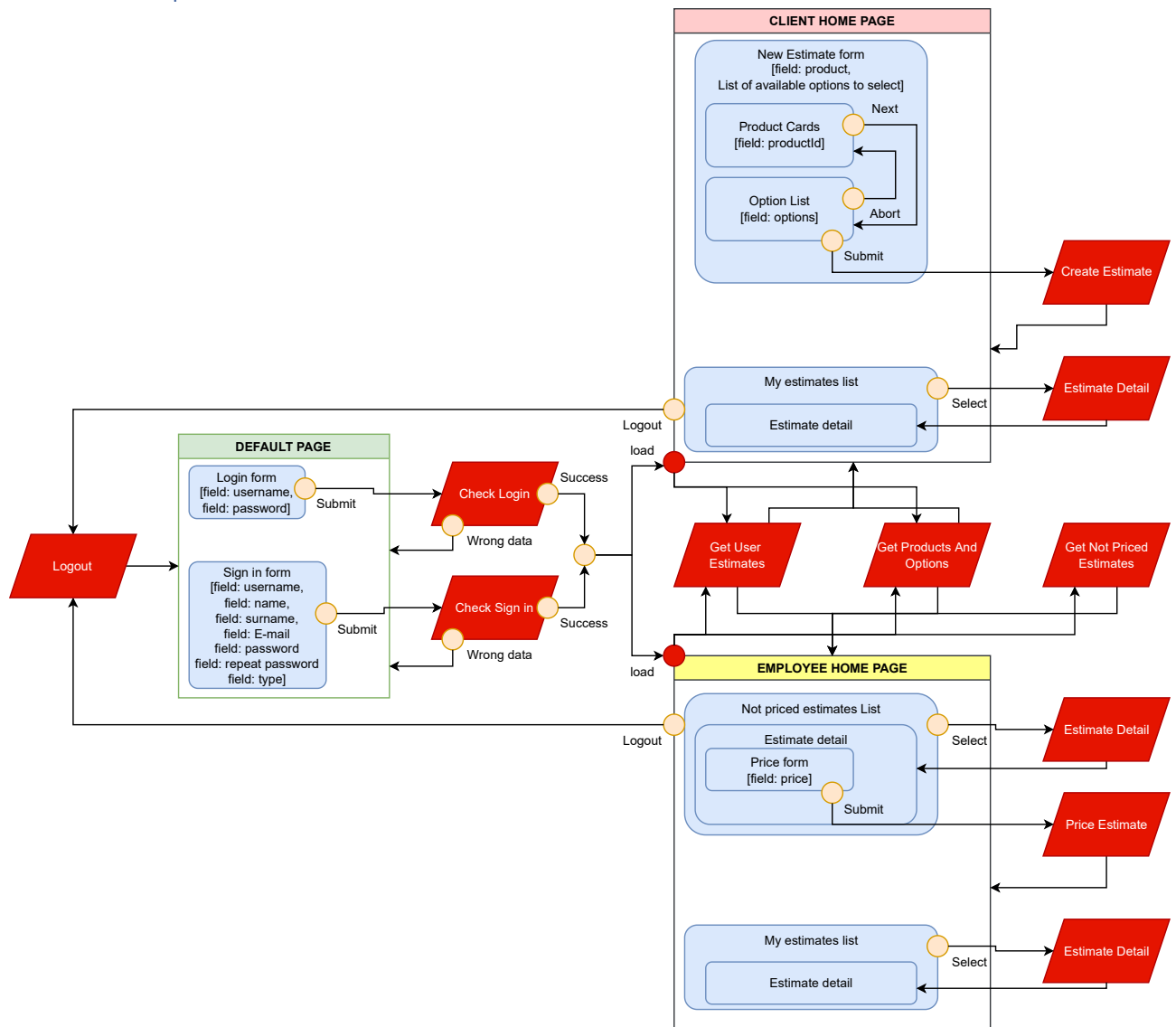
Client side		Server side	
Evento	Azione	Evento	Azione
Index -> form login -> login	Controllo dati	Post: Username, Password	Controllo credenziali
Sign up -> form sign up -> sign up	Controllo dati	Post: Username, e-mail, Password, Ripeti Password, Tipo Utente	Controllo dati e Registrazione delle credenziali
Home Page Cliente -> load	Aggiorna view con dati Form: prodotti e opzioni	Get	Estrazione prodotti e opzioni
	Aggiorna view con preventivi cliente	Get	Estrazione vecchi preventivi cliente
Elenco preventivo -> richiesta dettagli	Aggiorna view con dettagli preventivo	Get id preventivo	Estrazione dettagli del preventivo
Form nuovo preventivo -> scegli prodotto	Passa alla scelta delle opzioni	-	-
Form nuovo preventivo -> submit	Controllo dati	Post: id prodotto, lista id opzioni	Controllo dati e registrazione nuovo preventivo
	Aggiorna view elenco preventivi cliente	Get id cliente	Estrazione preventivi cliente
Form nuovo preventivo -> annulla scelta prodotto	Passa alla scelta del prodotto	-	-
Dettagli preventivo -> chiudi	Chiudi sezione dettagli preventivo	-	-
Home Page Impiegato -> load	Aggiorna view con preventivi non prezzati	Get	Estrazione preventivi non prezzati
	Aggiorna view con preventivi prezzati in precedenza dall'utente	Get id employee	Controllo id utente ed Estrazione preventivi prezzati dall'utente
Elenco Preventivi prezzati -> scelta di un preventivo	Aggiorna view con dettagli preventivo	Get id preventivo	Controllo dati ed estrazione dettagli del preventivo
Dettagli preventivo -> chiudi	Chiudi sezione dettagli preventivo	-	-
Elenco Preventivi non prezzati -> scelta di un preventivo	Aggiorna form prezza preventivo con dettagli preventivo	Get id preventivo	Controllo dati ed estrazione dettagli preventivo
Form prezza preventivo -> submit	Controllo dati	Post: id preventivo, costo	Controllo dati e salvataggio del prezzo
	Aggiorna vista elenco preventivi non prezzati	-	-
	Aggiorna vista elenco preventivi prezzati	-	-
Log out	-	Get	Session.invalidate

## Controller / Event handler

Client side		Server side	
Evento	Controllore	Evento	Controllore
Index -> form login -> login	Function makeCall()	Post: Username, Password	CheckLogin (servlet)
Sign up -> form sign up -> sign up	Function makeCall()	Post: Username, e-mail, Password, Ripeti Password, Tipo Utente	CheckSignUp (servlet)
Home Page Cliente -> load	Function PageOrchestrator	Get	GetPrAndOptList (servlet)
Home Page Cliente -> load	Function EstimateList.update()	Get	GetUserEstimates (servlet)
Elenco preventivo -> richiesta dettagli	Function EstimateDetails()	Get id preventivo	GetEstimateDetails (servlet)
Form nuovo preventivo -> scegli prodotto	Function form.next()	-	-
Form nuovo preventivo -> submit	Function form.submit()	Post: id prodotto, lista id opzioni	CreateEstimate (servlet)
	Function PageOrchestrator.updateUserEstimates	Get id cliente	GetUserEstimates (servlet)
Form nuovo preventivo -> annulla scelta prodotto	Function form.abort()	-	-
Dettagli preventivo -> chiudi	Function EstimateDetails.close()	-	-
Home Page Impiegato -> load	Function EstimateList.update() called by PageOrchestrator.initialize()	Get	GetNotPricedEstimates (servlet)
		Get id employee	GetUserEstimates (servlet)
	Function ProductsAndOptions.getData()	Get	GetProductsAndOptions (servlet)
Elenco Preventivi prezzati -> scelta di un preventivo	Function PageOrchestrator	Get id preventivo	GetEstimateDetails (servlet)
Dettagli preventivo -> chiudi	Function EstimateList.resetView()	-	-
Elenco Preventivi non prezzati -> scelta di un preventivo	Function EstimateDetails()	Get id preventivo	GetEstimateDetails (servlet)
Form prezza preventivo -> submit	Function makeCall()	Post: id preventivo, costo	PriceEstimate (servlet)
	Function PageOrchestrator	-	-
	Function PageOrchestrator	-	-
Log out	-	Get	Session.invalidate

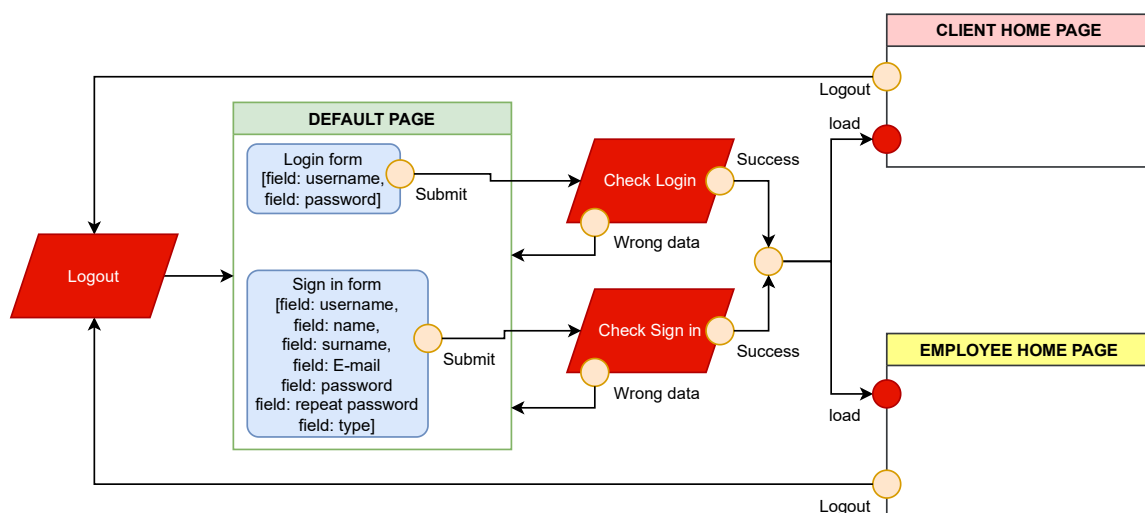
## Schema di applicazione

### Schema Complessivo

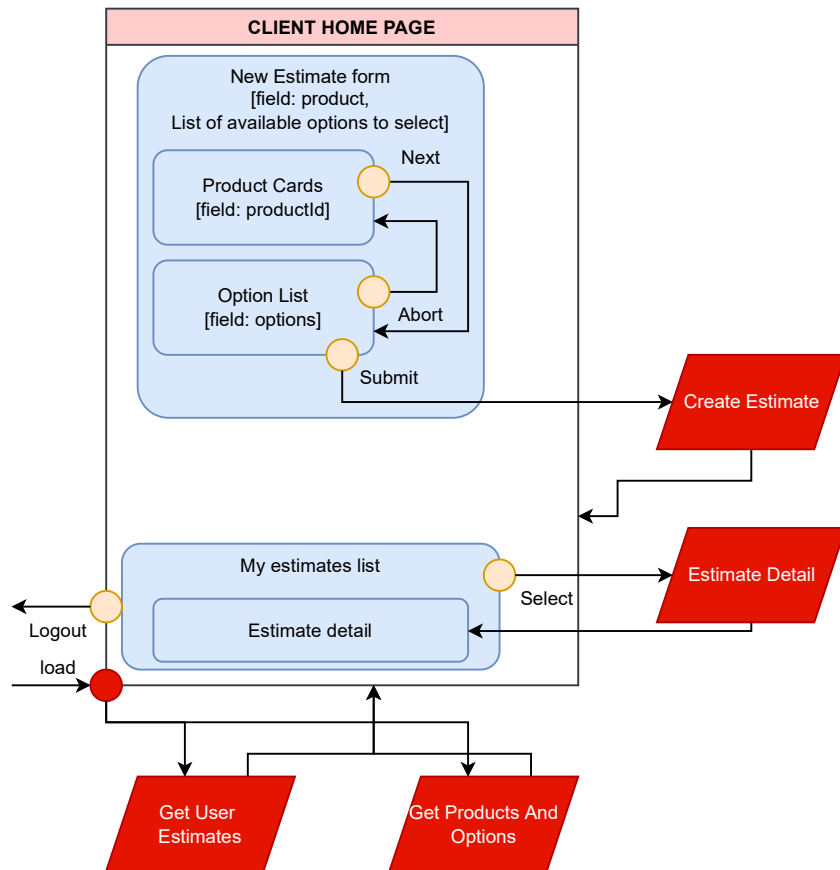


## Zone autorizzate per tipo di utente

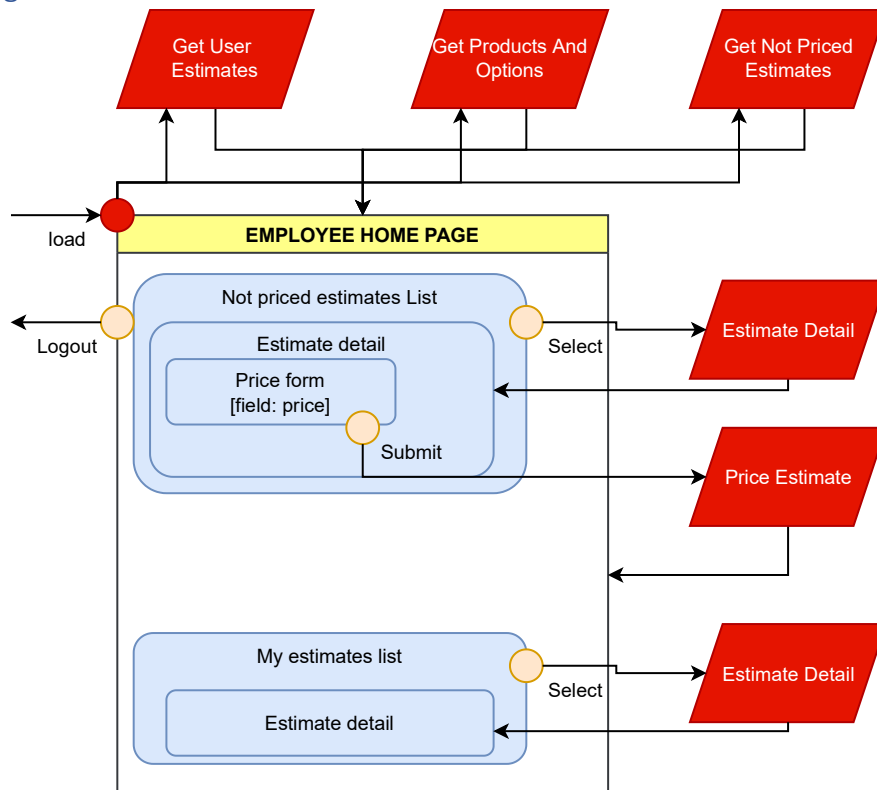
### Accesso Pubblico



## Accesso Cliente



## Accesso Impiegato





## Server Side - Componenti

### DAOs (Data Access Objects)

#### UserDAO

```
checkCredentials(username, password)
registerCredentials(username, e-mail, password, userType)
isUsernamePresent(username)
getById(userId)
```

#### OptionDAO

```
optionsFromProduct(productCode)
optionsFromEstimate(estimateCode)
getFromCode(optionCode)
```

#### ProductDAO

```
getAll()
getByCode(productId)
```

#### EstimateDAO

```
getUser(user)
getByCode(estimateCode)
getNotPriced()
createEstimate(clientId, productCode, optionCodeList)
priceEstimate(estimate, employeeId, price)
```

### Bean

User

Estimate

Option

Product

### Controllori

CheckLogin

CheckSignIn

GoToHome

GetEstimate

GetAvailableOptions

CreateEstimate

Logout

## Client Side - View e View Component

### Index

#### Login Form

Gestione di submit e degli errori

#### SignUp Form

Gestione di submit e degli errori

### Client Home Page

#### ProductsAndOptions

- Show(): chiamata asincrona al server per conoscere: prodotti, opzioni e disponibilità.
- Update(): inizializzazione della visualizzazione del form
  - Assengazione event listener ai bottoni: Next, Submit, Abort;
  - updatePage1(): gestione della visualizzazione della pagina 1
  - updatePage2(): gestione della visualizzazione della pagina 2
- checkDataCorrectness(): Cliccato submit, controlla correttezza dei dati della form.
- Vari semplici getter di specifici prodotti o opzioni a partire dal loro codice

#### EstimateList

- update(): chiamata asincrona al server per conoscere preventivi precedenti.
- Show(): stampa a video dell'elenco dei preventivi
- resetView(): quando si seleziona un nuovo preventivo di cui vedere i dettagli toglie i dettagli visti in precedenza

#### EstimateDetail

- showEstimateDetail(): costruisce e posiziona la visualizzazione dei dettagli del preventivo

### Client Home Page

#### ProductsAndOptions

- getInfo(): chiamata asincrona al server per conoscere: prodotti, opzioni e disponibilità.
- Vari semplici getter di specifici prodotti o opzioni a partire dal loro codice

#### EstimateList

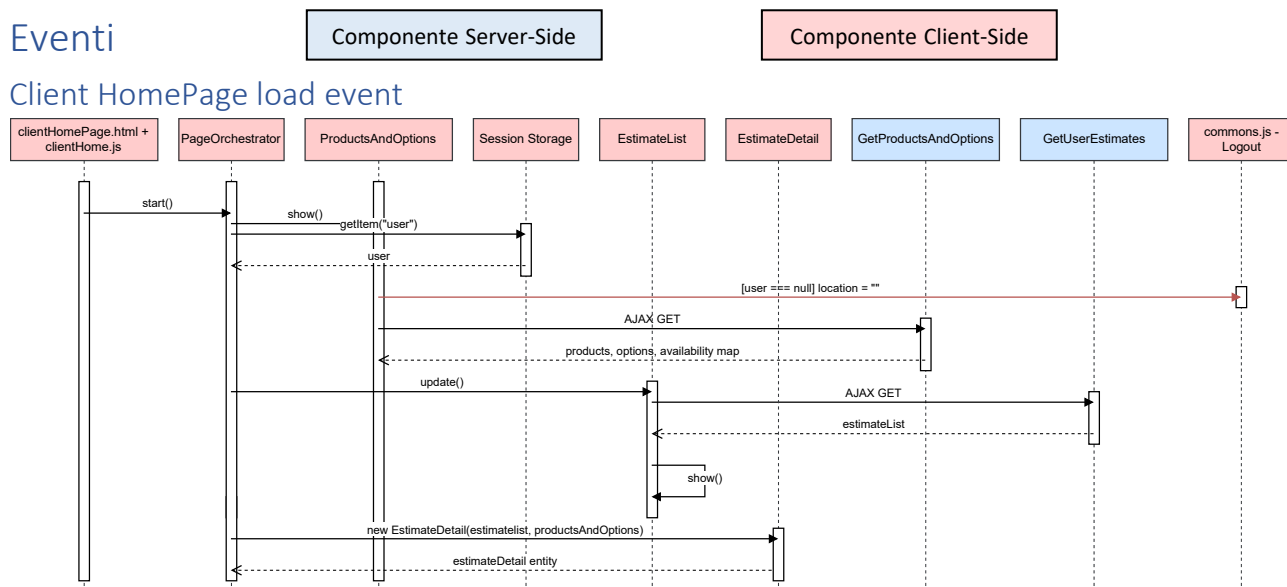
- update(): chiamata asincrona al server per conoscere preventivi precedenti.
- Show(): stampa a video dell'elenco dei preventivi
- resetView(): quando si seleziona un nuovo preventivo di cui vedere i dettagli toglie i dettagli visti in precedenza
- containsCode(codice preventivo): dice se nell'elenco attuale è contenuto un certo preventivo
- removeEstimate(codice preventivo): rimuove un preventivo
- addEstimate(preventivo): aggiunge un preventivo

#### EstimateDetail

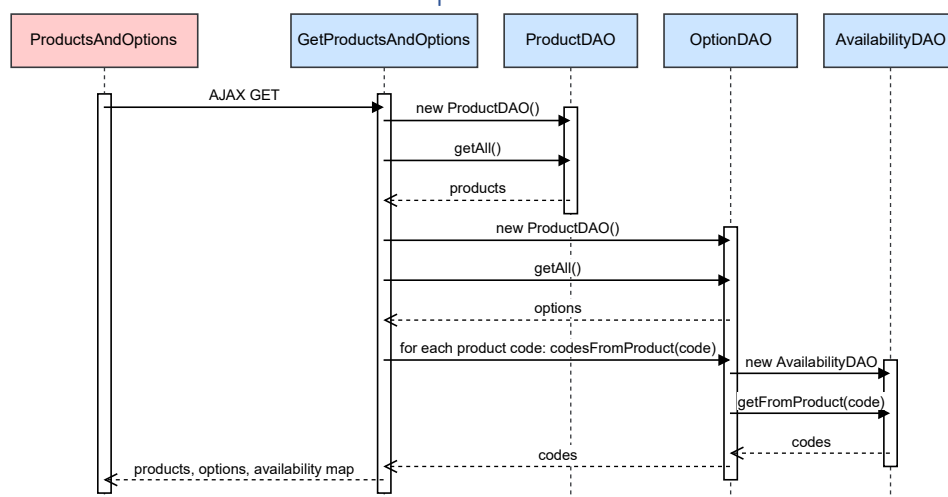
- showEstimateDetail(): costruisce e posiziona la visualizzazione dei dettagli del preventivo

## Eventi

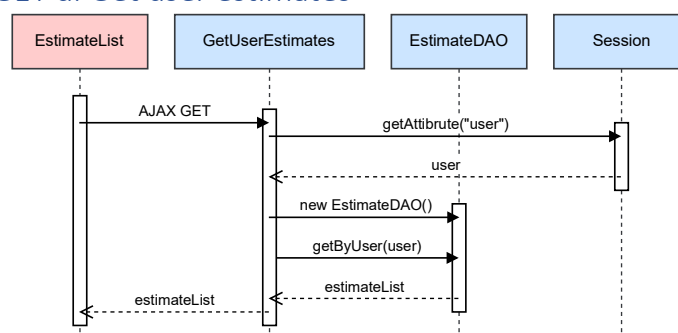
### Client HomePage load event



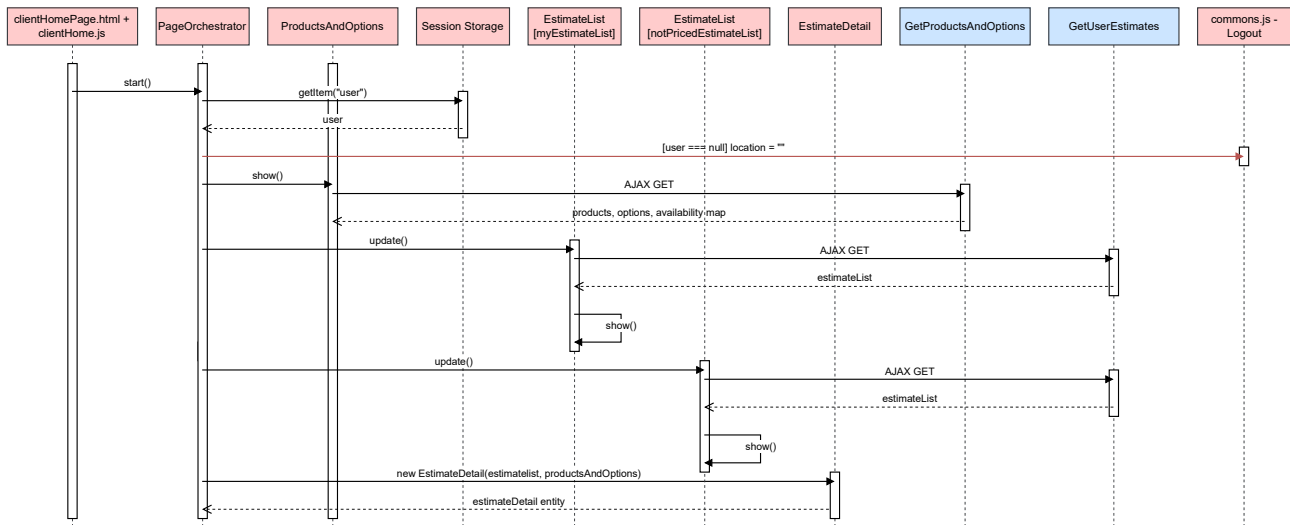
### Dettaglio della AJAX GET di Products and Options



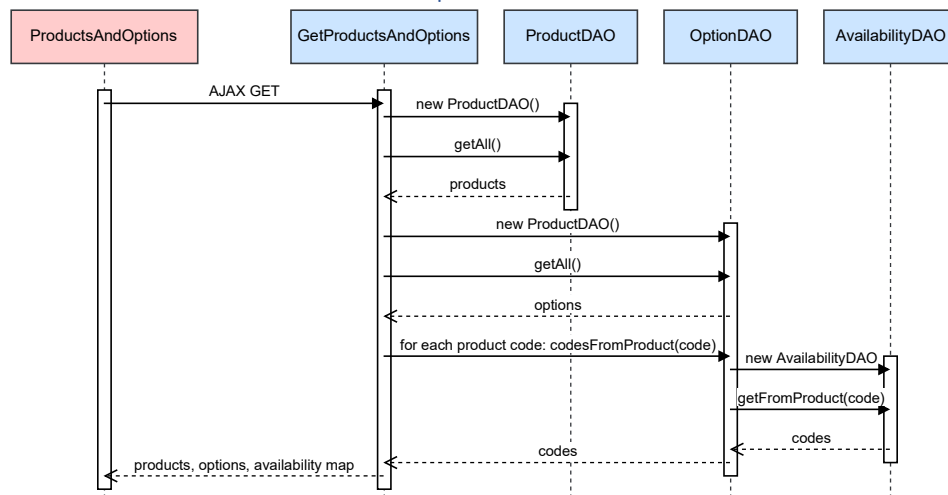
### Dettaglio della AJAX GET di Get user estimates



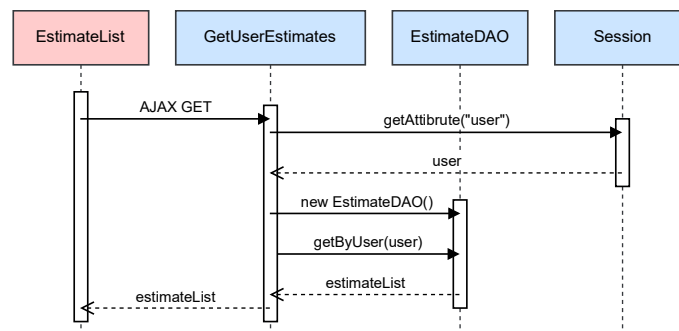
## Estimate HomePage load event



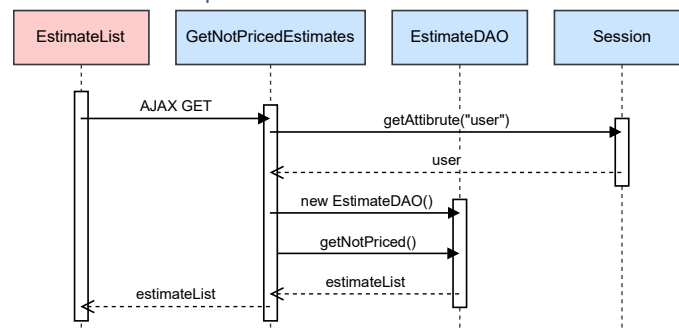
## Dettaglio della AJAX GET di Products and Options



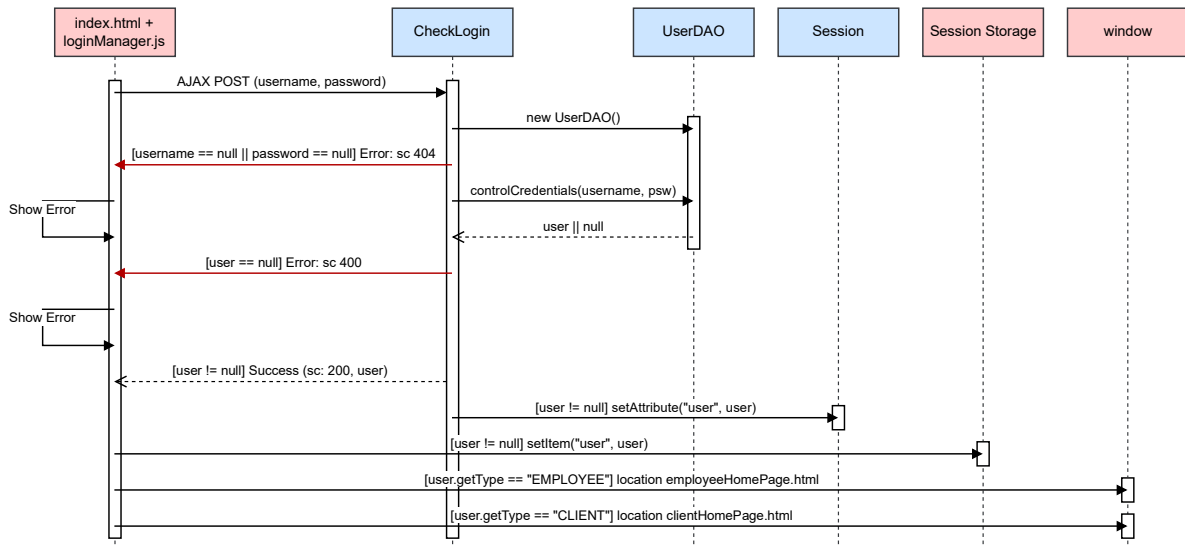
## Dettaglio della AJAX GET di Get user estimates



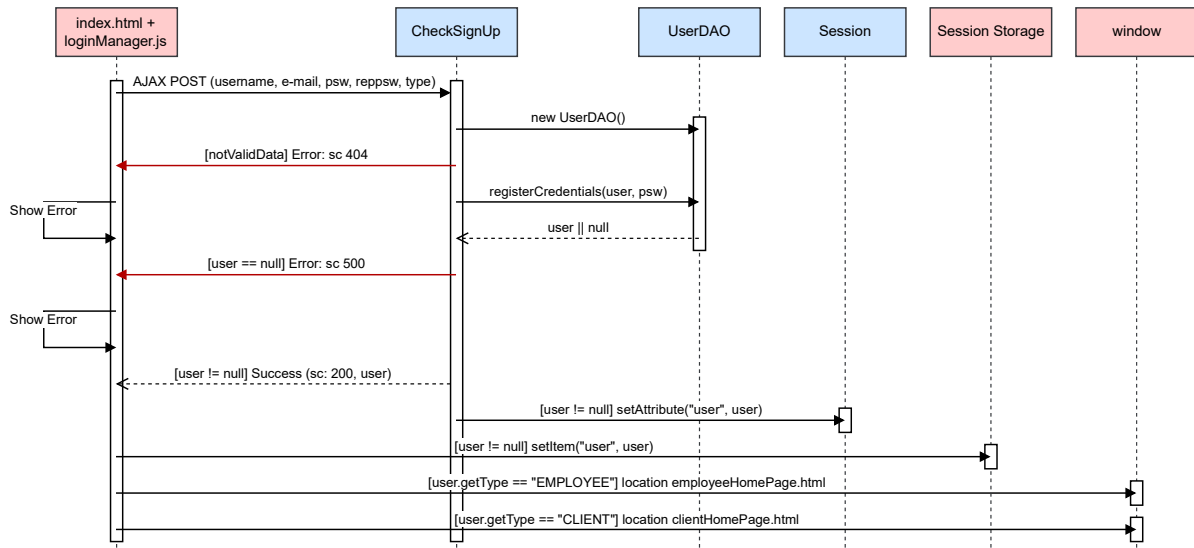
## Dettaglio della AJAX GET di Get not priced estimates



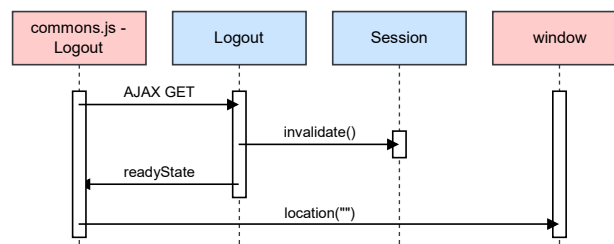
## Log in



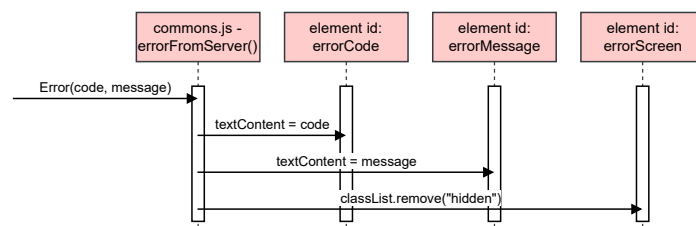
## Sign up



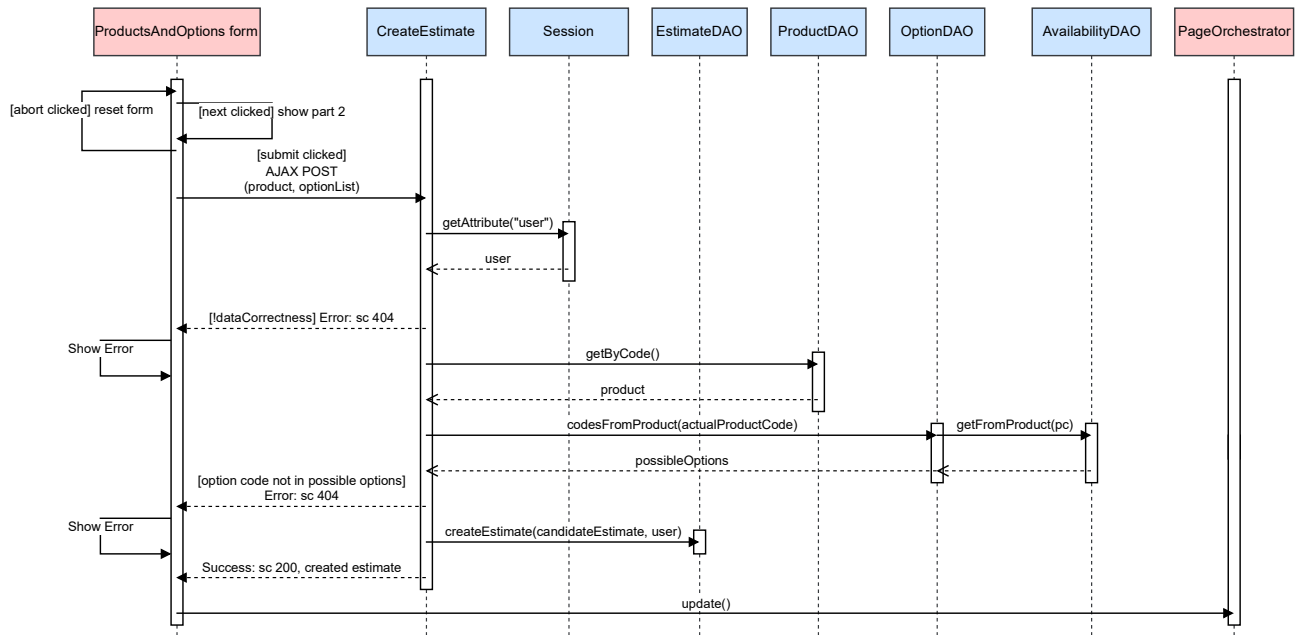
## Logout



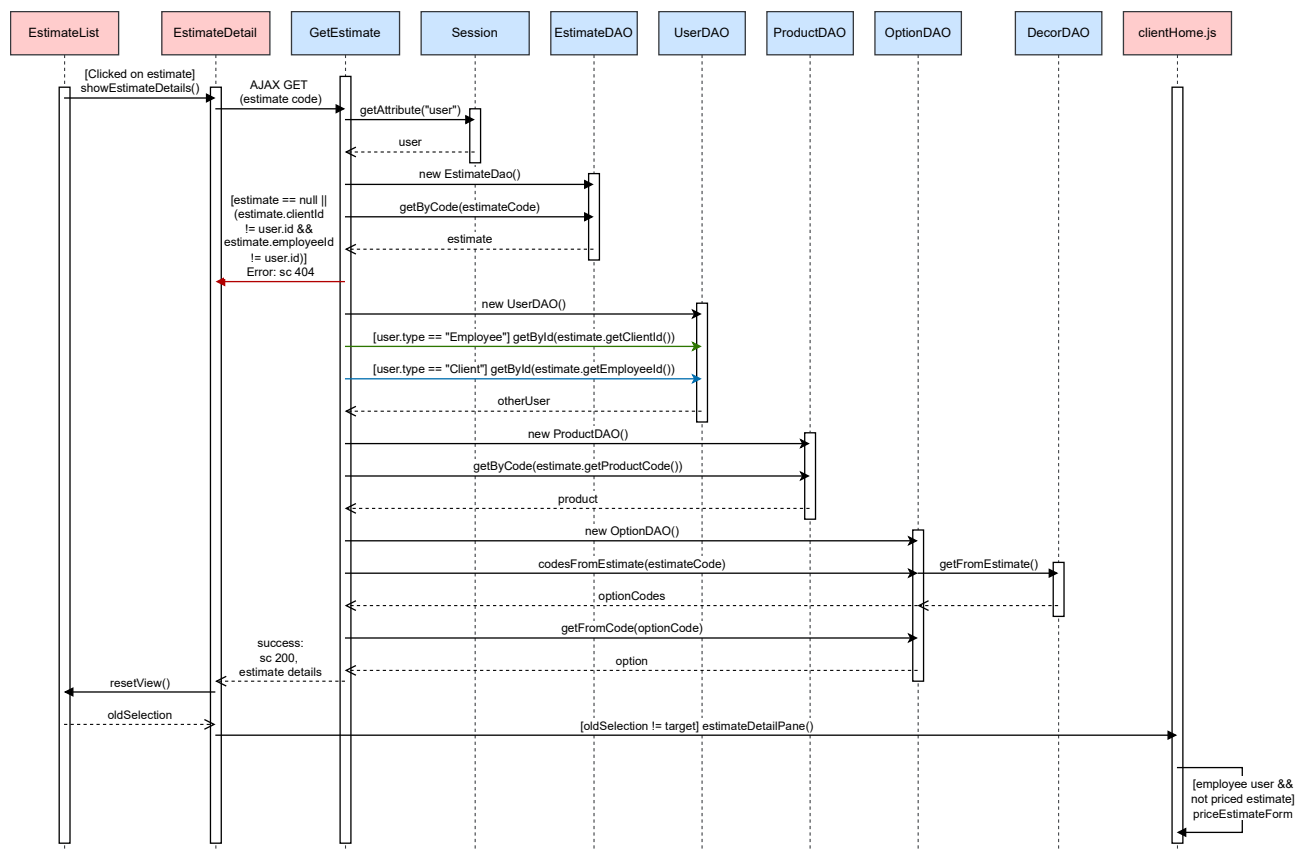
## Show Error



## Create estimate



## Estimate Detail



## Price estimate

