

PROGETTO LAB-AIR – seconda prova

Funzionalità dinamiche dell'interfaccia basate su Angular

In questa seconda prova dovrete rendere dinamico il lavoro che avete già realizzato in modalità statica nella prima prova di progetto, utilizzando il framework **Angular** per automatizzare tutte le funzionalità richieste dell'applicazione.

Ricordiamo (vedi traccia relativa alla prima prova del progetto) che la finalità è quella di realizzare un sito web dedicato al settore della vendita online di articoli sportivi (noi abbiamo preso spunto da nike.com ma sentitevi liberi di farvi ispirare da qualunque sito simile vi piaccia).

N.B. In questo caso però dovrete implementare in maniera simile **TUTTE** le funzionalità (obbligatorie e opzionali) qui descritte.

STRUTTURA DELL'APPLICAZIONE

Il sito è costituito dalla parte front-end, dalla parte back-end e dai dati che rappresentano un campione rappresentativo di scarpe Nike.

FRONT END

Per il Front End si richiede che venga utilizzato codice HTML, CSS, Javascript / Typescript e Angular; dovrà essere realizzato in modalità Single Page, in maniera conforme alla logica delle applicazioni del framework.

In questa seconda fase il layout dovrà essere totalmente **responsive**.

Si richiede che vengano usate le seguenti funzionalità salienti del framework:

- **Binding** (interpolazione, property binding, event binding e two way binding)
- **Direttive strutturali** (ngFor, ngIf, ngSwitch) o di **attributo** (ngStyle, ngClass)
- **Data sharing** tra Components (@Input() e @Output())
- **Routing** per gestire il passaggio tra le varie rotte (voci di menu e funzionalità)
- **Angular Services**, le classi condivise per gestire task che possono essere comuni a più Component
- **Chiamate HTTP Ajax**

I DATI

Nel file allegato **nike.json** ci sono 50 modelli di scarpe Nike che abbiamo estrapolato.

N.B. a completamento di questi dati, dovrete aggiungere **tutte** le immagini cercandole su Google tramite il nome della scarpa e altre due proprietà che si rendono necessarie ai nostri scopi:

- **nuovo_arrivo: boolean (true/false)**
- **best_seller: number (1 ... 5)**

BACKEND

Il **Back End** è costituito da un'applicazione lato server (Json Server) che simula una presentazione di dati "fake", residenti direttamente sul vostro server locale e inerenti ad un campione rappresentativo di scarpe Nike (il file di dati di cui sopra).

In allegato troverete un file pdf che spiega come configurare ed utilizzare il **Json Server**.

Inoltre, al seguente indirizzo - <https://labforweb-vod.b-cdn.net/video/InstallazioneJsonServer.mp4> troverete un video che spiega l'utilizzo del Json Server.

1. Home

La homepage del sito dovrà visualizzare i prodotti con eventuali banner, lista dei prodotti, slider sulla linea del vostro primo progetto.

Solo le seguenti voci della sezione Header dovranno essere dinamiche:

- **Tutti i nuovi arrivi:** mostrerà tutti i dati filtrati sulla proprietà **nuovo_arrivo**
- **Best seller:** mostrerà tutti i dati filtrati sulla proprietà **best_seller** con valore maggiore o uguale a 4
- Sempre nella sezione Header, sotto la voce **Scopri lo sport**, saranno elencate le **Categorie** (Basket, Running etc), ed a partire da queste sarà possibile filtrare i risultati per Categoria
- **La Search Bar** permetterà di filtrare i dati facendo una ricerca libera sul **nome della scarpa**
- **All'interno della Home Page** come potete notare sono presenti alcuni pulsanti (o link) **Acquista**. Devono puntare tutti ad una pagina che permette di visualizzare **tutti i prodotti**.

OPZIONALE

Per facilitare il caricamento degli articoli, chi vorrà cimentarsi potrà scegliere di presentarne solo 20 per volta e di caricare i restanti o con la tecnica della **paginazione** oppure con la tecnica dell'**Infinite Scrolling**.

Sempre nella pagina di **Tutti i prodotti**, bisognerà implementare dinamicamente, tra le funzionalità presenti sul lato sinistro:

- **Categorie** (funzionalità già descritta)
- **Colore** (con la possibilità di scegliere le scarpe per il colore, con i colori disponibili nei dati Json)
- **Acquista per prezzo** (mostrare i risultati per le tre fasce di prezzo descritte sul sito Nike)

1.2 Dettaglio

Ogni volta che clicchiamo su una delle scarpe presenti in un elenco filtrato accediamo alla singola pagina del prodotto che rappresenta il dettaglio di una singola scarpa.

Questa **Scheda Prodotto** conterrà le caratteristiche elencate nel file di dati Json. Il layout della **Scheda Prodotto** dovete implementarlo voi prendendo spunto da ciò che c'è sul sito e da ciò che avete già implementato nella prima parte del progetto.

Non potrà mancare nella scheda prodotto:

- La possibilità di vedere almeno una foto della scarpa (da cercare in rete).
- La possibilità di scegliere la taglia nelle informazioni presenti nei dati e di sceglierne il colore, sempre presente nei dati
- Il pulsante **Aggiungi al carrello**

L'aggiunta del prodotto al carrello

Facendo click sul pulsante **Aggiungi al carrello** comparirà un box in sovraimpressione (come già implementato nella prima parte del progetto) che riporti la scelta fatta (nome prodotto, immagine, costo, numero e colore scelto).

Da questa sezione sarà possibile:

- Visualizzare il Carrello
- Procedere al pagamento

Visualizza il carrello

La pagina del carrello conterrà tutti prodotti inseriti nel carrello.

Del carrello dovreste gestire le seguenti funzionalità:

- Possibilità di aggiungere più prodotti diversi tra loro
- Possibilità di aggiungere un prodotto già presente aumentandone la quantità
- Possibilità di rimuovere un prodotto singolo / rimuovere un articolo di un prodotto presente in più esemplari
- Gestire dinamicamente il costo totale e parziale

Checkout

È il modulo wizard (a step) che compare per completare l'acquisto e che già avete implementato nella prima parte. Selezionate la modalità ospite per espletare le operazioni senza effettuare login e registrazione come avete già fatto nella prima parte.

Il modulo dovrà gestire anche i controlli dei campi tramite le Regular Expression. Dovranno quindi essere mostrati a video gli eventuali errori.

Il modulo di checkout è a step.

Primo step

Solo al completamento dei dati del primo step diventerà attivo il pulsante **Salva e continua** che permetterà l'accesso al secondo step.

Secondo step

Nel secondo step dovreste **controllare i dati di pagamento**

Terzo step: La thank you page

Il messaggio di conferma acquisto fatelo a vostro piacimento

PARTI OPZIONALI

1. Sarà possibile, **in modo facoltativo**, utilizzare altre funzionalità apprese o nel modulo di Angular Avanzato oppure acquisite in autonomia:

- Le Angular Guards per proteggere le rotte
- Le Custom Pipes
- Le rotte Children
- Le reactive Forms

2. SIMULIAMO L'ACQUISTO NELL'AREA RISERVATA

Per chi vorrà cimentarsi, potete effettuare l'acquisto **con spedizione gratuita** effettuando la **registrazione** ed il **login**

Dovrete quindi implementare le funzionalità di:

- **Login**
- **Registrazione**
- **Acquisto prodotto come user autenticato**

Questo tramite l'utilizzo del Json Server con la modalità autenticazione come spiegato nel video e nel documento allegato.

Una volta autenticati verrete indirizzati nella pagina iniziale (dashboard) dell'area riservata dell'utente e potrete:

- visualizzare la lista degli ordini
- procedere all'inserimento di un ordine

Valutazione

La valutazione sarà a discrezione del corpo docente e si baserà sulla seguente griglia di abilità:

- Funzionamento dell'applicativo e dei suoi task
- Funzionamento del codice
- Gradevolezza della Struttura grafica e della User Experience
- Funzionamento Responsive
- Esposizione del progetto

La votazione assegnata sarà espressa in 30-esimi e con il seguente criterio:

- Fino a 27 per chi implementa correttamente tutte le funzionalità obbligatorie
- Fino a 30 con possibilità di lode per chi implementa correttamente anche le funzionalità facoltative

Buon lavoro a tutti

Lo staff della scuola Labforweb ☺