PROVA DI VERIFICA SUL FRAMEWORK ANGULAR

Creare un progetto Angular mediante cli che utilizzi il routing.

Importare i moduli per la gestione del Form (FormsModule) e per la gestione delle comunicazioni http (HttpClientModule). Importare Bootstrap e Jquery.

Realizzare i seguenti punti (difficoltà crescente)

1)Impostare nel componente root un contenitore che utilizzi la class container di bootstrap

2) Creare un componente chiamato header che sia dotato di 3 proprietà (di tipo string,number e boolean). Visualizzare nella view corrispondente i valori assegnati a tali variabili

3) Nel componente del punto 2 aggiungere un bottone che al click modifichi il valore della variabile booleana creata nel punto 2). Sfruttare questa proprietà per gestire la visibilità di un elemento della view

4) Utilizzare ngFor nel componente creato al punto 2) per la visualizzazione di una variabile di tipo array di string,

5) Aggiungere al progetto una classe in un folder model. La classe deve essere dotata di proprietà di vario tipo (string, boolean e number)

6) Creare un servizio che permetta di leggere e scrivere 3 valori di tipo boolean, string e number,

7) Il servizio del punto precedente deve permettere di leggere e scrivere un’istanza della classe generata a punto 5.

8) Utilizzare nel componente header il servizio creato al punto 5) e 6) per valorizzare le proprietà mostate a video.

9) Modificare il servizio creato al punto 5) e il componente in modo da utilizzare gli Observable.

10) Aggiungere un componente chiamato body dotato almeno di una proprietà @input e di una di tipo @Output

11) Visualizzare il nuovo componente e gestire la comunicazione col componente header mediante @Input e @Output

12) Aggiungere un servizio che generi messaggi da propagare nell’applicazione. In ogni componente che è stato creato utilizzare questo servizio nell’OnInit e in tutti i metodi scritti in modo da creare un nuovo messaggio. Ad esempio (“sono nell’onInit di header”),

13) Visualizzare nel componente body i messaggi prodotti da servizio al punto precedente,

14) Simulare una visualizzazione del numero di notifiche come visto nell’applicazione Facebook.

Ad esempio: uno dei servizi trasmetterà un numero che verrà usato dal componente header

E visualizzato in uno span usando la classe badge di bootstrap. Opzionale: usare setInterval per far crescere questo numero ogni 30 secondi.

15) Creare un servizio che comunichi con gli endpoint del sito placeholder.

Nella vostra app assumerete uno userid( ad esempio 1). Questo userId sarà gestito da uno dei servizi che avete creato nell’applicazione.

1. Recuperate il numero di posts pubblicati dallo userid = 1 e visualizzatelo nel componente header mediante badge.
2. Recuperate il numero totale di commenti a post dello userid = 1 e visualizzatelo nel componente header mediante badge
3. Recuperate il numero totale di album pubblicati dallo userId=1 e visualizzatelo nel componente header mediante badge
4. Recuperate il numero totale di foto pubblicate dallo userid = 1 e visualizzatelo nel componente header mediante badge

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

16) Utilizzare il routing come mostrato nelle slide di teoria in modo da associare ad ogni path un componente. Aggiungere i componenti e inserire il <router-outlet> nel componente body.

Almeno uno dei nuovi componenti verrà usato per inserire un form che si occuperà di comunicare in post con le api di placeholder (ad esempio l’aggiunta di un post),

17) A partire dalle api di placeholder e da uno userId, creare una classe “aggregato” che contenga tutte le informazioni di uno user ( i post, e per ogni post i commenti), album ( e per ogni album le foto). Valutare l’impatto di una istanza di tale classe aggregato e il tempo necessario per il caricamento dei dati

18) Implementare il componente modale secondo le istruzioni contenute nella slide. Al suo interno provare a gestire il componente usato per il post nel punto 16

19) Creare un componente che visualizzi tramite le api di geolocation di html 5 le coordinate dell’utente. Gestire l’errore in assenza di autorizzazione

20) Creare un servizio per la gestione di lettura e scrittura su local storage e utilizzarlo in un componente

21) Creare un componente comprendente una canvas, un bottone per aggiungere un personaggio alla canvas e 4 bottoni per far muovere il personaggio nella canvas.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_