Wednesday: Add Boostrap

We have been using Bootstrap to style our applications and website ever since we learnt in Prep. It's worth mentioning that there are other alternatives to Bootstrap such as Material UI, Materialize, Foundation and many others(you can read about them **here (https://classpert.com/blog/top-bootstrap-alternatives)**). You might ask, then why use Bootstrap if we have so many alternatives?

So let's look at some of the pros and cons of Bootstrap to a developer and designer.

Pros

- **Good Documentation and Community Support**

  Bootstrap has amazing documentation for every document, hence it's easy to follow and understand the documentation whenever you're stuck. Moreover, it has one of the largest and most active community. If you encounter a problem with bootstrap, most probably someone else has already found a solution to it and the easiest way to find this solution is through the Bootstrap community. Bootstrap community include; the StackOverflow community, **official bootstrap blog (https://blog.getbootstrap.com/)** , **bootstrap twitter account, (https://twitter.com/getbootstrap)** and **slack. (https://bootstrap-slack.herokuapp.com/)**

- **Bootstrap Offers a Great Grid System**

  A good grid is essential if you are looking to create a great page layout. Bootstrap has one of the best responsive grid system. It divides content into 12 fluid and responsive columns. This makes hiding certain elements that can only be visible on desktop (because you don’t need them on mobile devices) easy.

- **Responsiveness**

  As a developer, when creating a website or application, it's very important to always think about how the website will look on a mobile device. This is because in most cases, the user will access the website or application on their mobile devices. Bootstrap is all about creating mobile-friendly websites thanks to the fluid grid we mentioned above. You just need to decide how many squares each column occupies and whether your columns are going to stack vertically or horizontally to display as intended on mobile devices.

- **It Helps Avoid Cross-Browser Bugs**

Cons

- **Without customization, Bootstrap websites look the same**

  The main drawback with bootstrap is that if you use bootstrap in projects and do little to no customization, then all your projects will look the same. Hence, you will be forced to override and modify styles manually.

- **It has a slight learning curve**

In as much as it's easy to learn bootstrap, you still have to invest some time into it. You'll need to familiarize yourself with all the available Bootstrap classes and know how the Bootstrap component access these classes. Moreover, you also need to experiment and get used to the grid system.

Adding Bootstrap

Our application is looking very plain and stale at the moment. We can beautify it using the frontend web component library we covered in prep, bootstrap. We will add bootstrap to Angular using the command line. The boostrap4 documentation also clarifies that some components in bootstrap, like modals, need jquery and popper to work so let's install all of them in case we need to use such components.

```
$ npm install --save bootstrap popper
```

This command will add bootstrap to our node modules. We need to make it available inside the application at the root level for us to use it. We'll do this in our *styles.css* file available at root.

src/app/styles.css

```
@import "~bootstrap/dist/css/bootstrap.css"
```

We have done this by simply importing bootstrap in the root CSS file. We can now use bootstrap in our application. At this point, if we take a look at our application after starting our local server, we can spot a change in the font already since bootstrap is installed.

Let's use a simple layout for our application.

src/app/goal/goal.component.html

```
<div class="container">
  <h1>My Goals</h1>
  <hr>

  <div class="row">
    <div class="col-md-6">
      <div *ngFor='let goal of goals;let i = index'>
        <div>
          <h4 id={{i}} appStrikethrough>{{goal.name}} due on {{goal.completeDate|date|uppercase}}</h4>
          <button (click)='toggleDetails(i)' class="btn btn-primary">Toggle Details</button>
        </div>

        <div *ngIf='goal.showDescription'>
          <app-goal-detail  [goal]='goal' (isComplete) = 'deleteGoal($event,i)'></app-goal-detail>
        </div>
      </div>
    </div>
  </div>

  <p *ngIf='goals.length > 5'>You have too many goals</p>
</div>
```

src/app/goal-detail/goal-detail.component.html

```

<p>{{goal.description}}</p>
<p>This goal will be complete in {{goal.completeDate|dateCount}} days</p>
<button (click)= 'goalDelete(true)' class="btn btn-outline-danger">Delete Goal</button>
```

We have used boostrap classes in our HTML templates tags and the bootstrap grid layout also. Feel free to make adjustments to your application to make it more visually appealing.