UI/UX Component Design

Customer v/s Consumer
=
Loyalty v/s Experience

What Constitutes an Experience?

=> For the consumer
⇒ For the provider

Are these Related?

Sites or Apps?

Documents:
Content Centric

Applications:
Behavior Centric

Web sites

Web Apps

Progressive Enhancements v/s Graceful Degradation

Is responsiveness the same?

Is everything an app?

Component Design

Key drivers

Modularity - Independent **Evolution**

Testability - Independently Testable

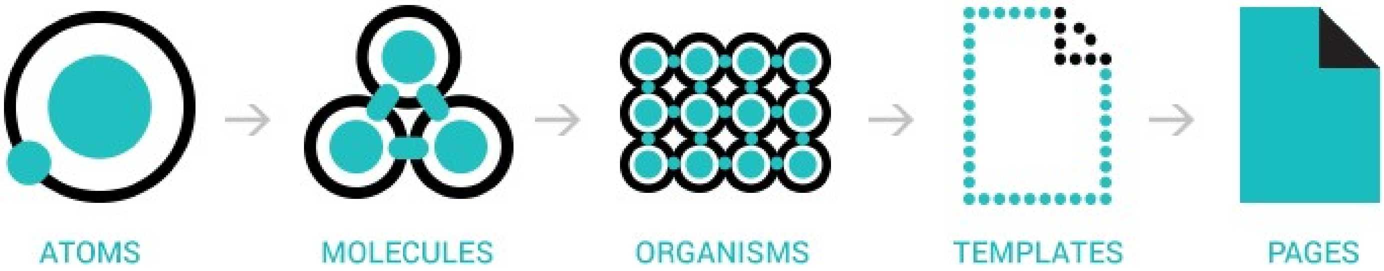
Reusability – Reusable across applications

Responsibility – Single Responsible Principle

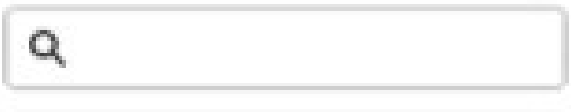
Gather together those things that change for the same reason.
Separate those things that change for different reasons

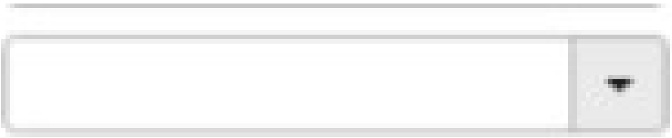
Contentious issues: coupling, orchestration

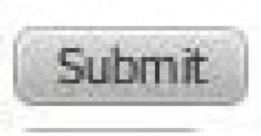
Atomic Design

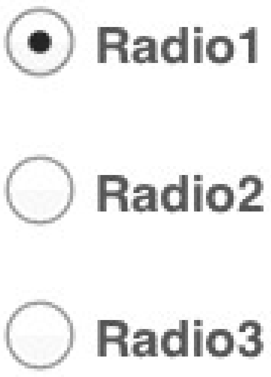


Atomic Design: Atom





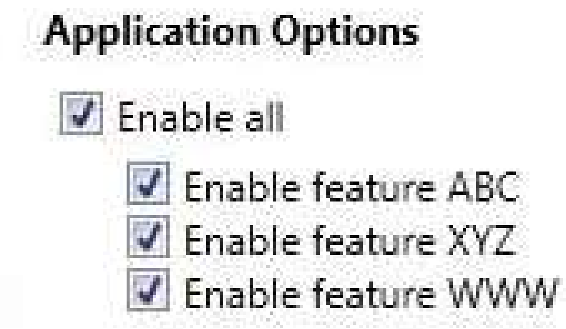




● Radio1

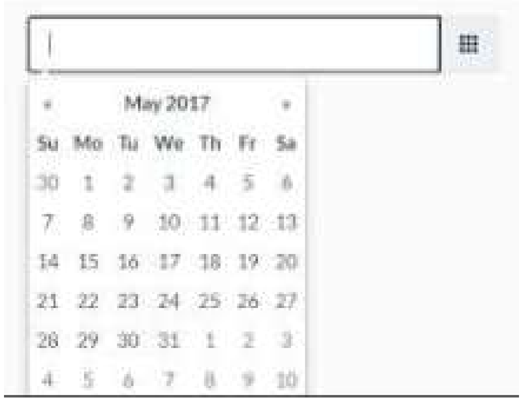
○ Radio2

○ Radio3



Application Options

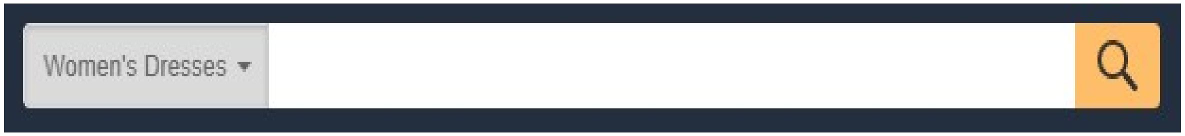
- ☑ Enable all
  - ☑ Enable feature ABC
  - ☑ Enable feature XYZ
  - ☑ Enable feature WWW

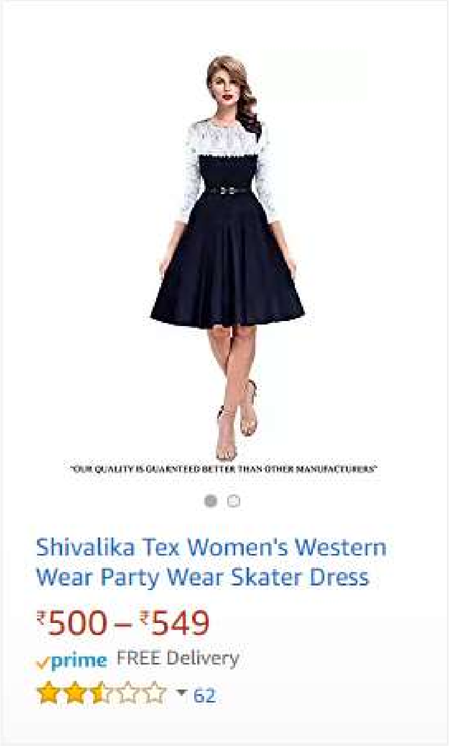


« May 2017 »

Su	Mo	Tu	We	Th	Fr	Sa
30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

Atomic Design: Molecules





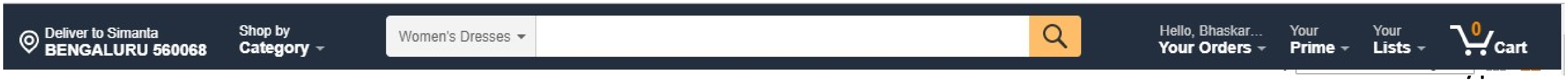
Shivalika Tex Women's Western Wear Party Wear Skater Dress

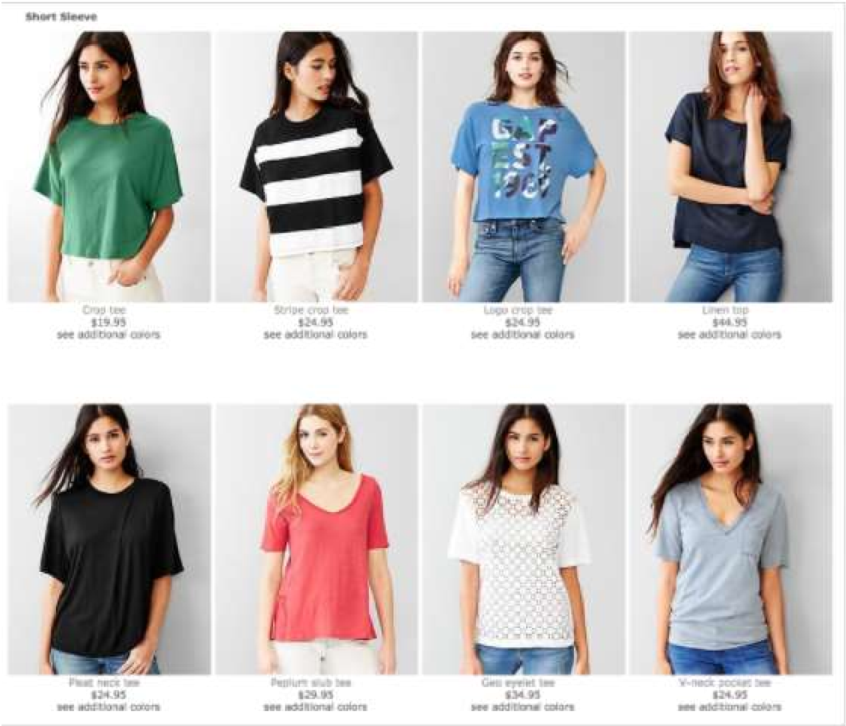
₹500 – ₹549

✓prime FREE Delivery

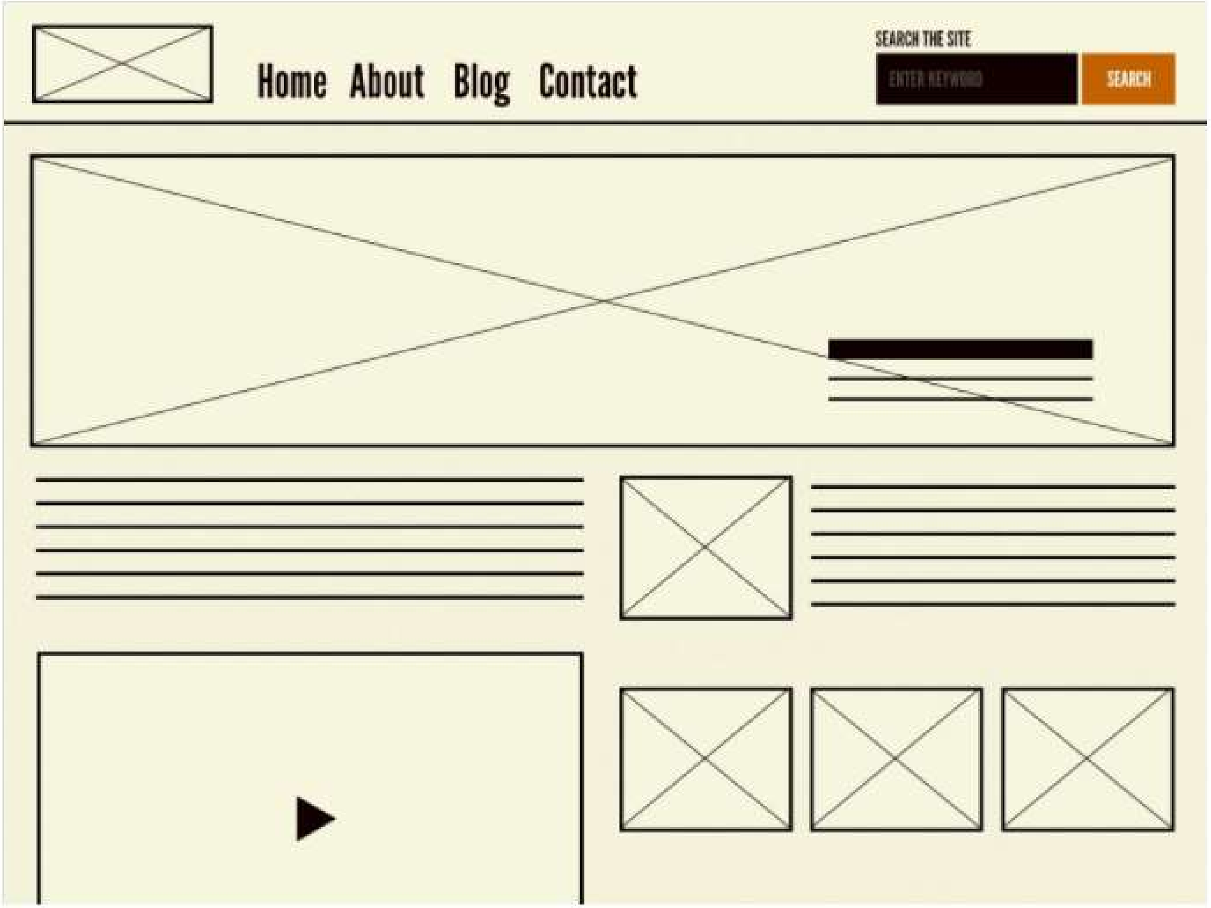
★★⯪☆☆ ▾ 62

Atomic Design: Organism





Atomic Design: Template



Angular Component Design

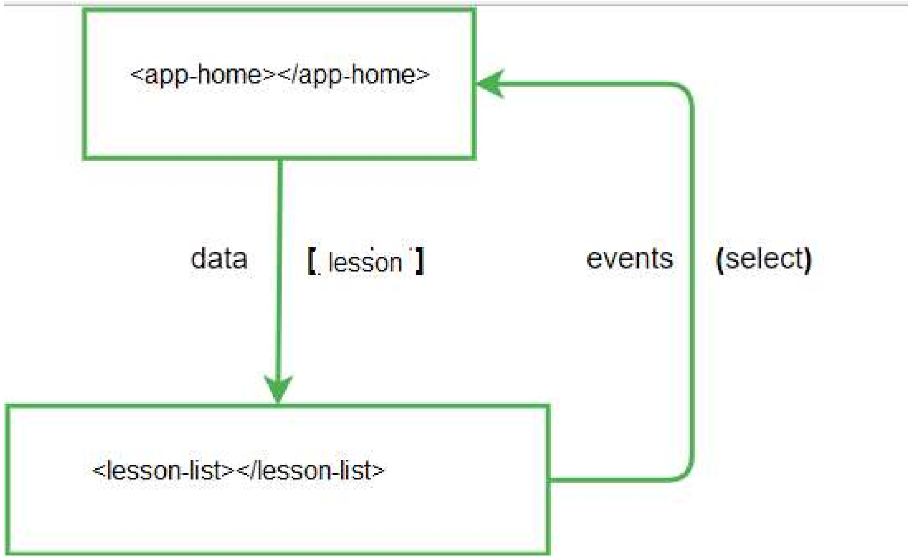
Smart Components vs Presentational Components

Would this presentation logic be useful elsewhere in the application?
Would it be useful to split things up further?
Are we creating unintended tight couplings in the application?

Angular Component Design

```
@Component({
  selector: 'app-home',
  template: `
    <h2>All Lessons</h2>
    <h4>Total Lessons: {{lessons?.length}}</h4>
    <div class="lessons-list-container v-h-center-block-parent">
        <table class="table lessons-list card card-strong">
            <tbody>
            <tr *ngFor="let lesson of lessons" (click)="selectLesson(lesson)">
                <td class="lesson-title"> {{lesson.description}} </td>
                <td class="duration">
                    <i class="md-icon duration-icon">access_time</i>
                    <span>{{lesson.duration}}</span>
                </td>
            </tr>
            </tbody>
        </table>
    </div>
`,
  styleUrls: ['./home.component.css']
})
export class HomeComponent implements OnInit {

    lessons: Lesson[];

  constructor(private lessonsService: LessonsService) {
  }

  ngOnInit() {
      this.lessonsService.findAllLessons()
          .do(console.log)
          .subscribe(
              lessons => this.allLessons = lessons
          );
  }

  selectLesson(lesson) {
    ...
  }

}
```

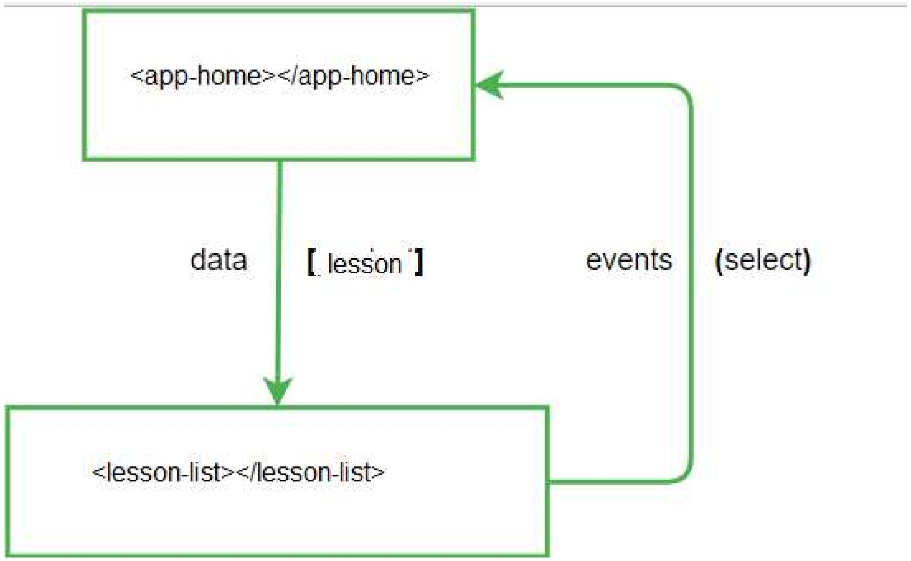
Angular Component Design



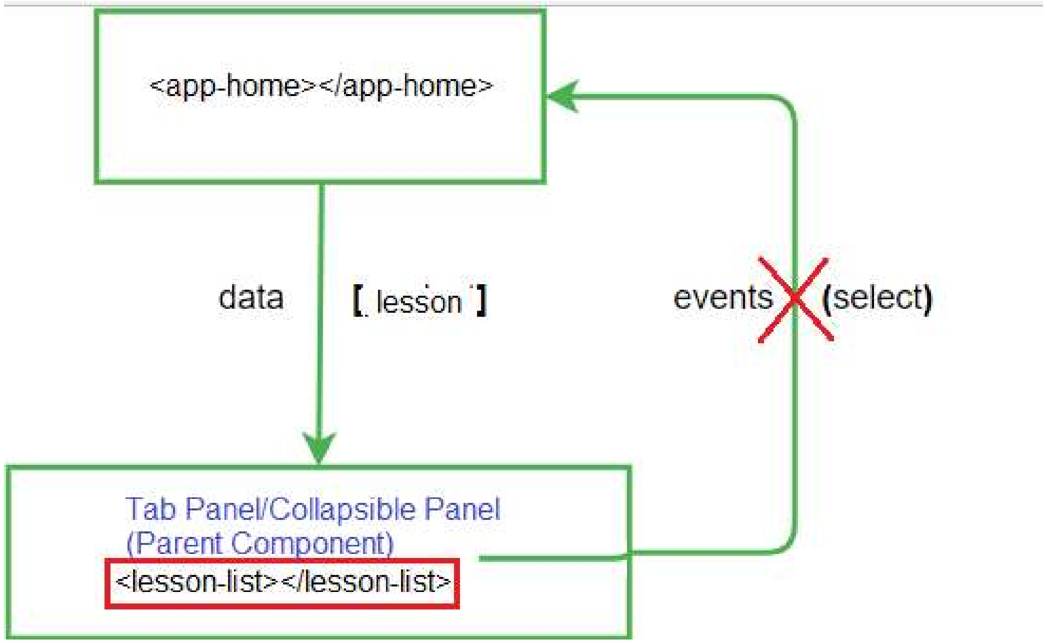
Angular Component Design

```
import {Component, OnInit, Input, EventEmitter, Output} from '@angular/core';
import {Lesson} from "../shared/model/lesson";

@Component({
  selector: 'lessons-list',
  template: `
      <table class="table lessons-list card card-strong">
          <tbody>
          <tr *ngFor="let lesson of lessons" (click)="selectLesson(lesson)">
              <td class="lesson-title"> {{lesson.description}} </td>
              <td class="duration">
                  <i class="md-icon duration-icon">access_time</i>
                  <span>{{lesson.duration}}</span>
              </td>
          </tr>
          </tbody>
      </table>
  `,
  styleUrls: ['./lessons-list.component.css']
})
export class LessonsListComponent {

  @Input()
  lessons: Lesson[];

  @Output('lesson')
  lessonEmitter = new EventEmitter<Lesson>();

    selectLesson(lesson:Lesson) {
        this.lessonEmitter.emit(lesson);
    }
}
```

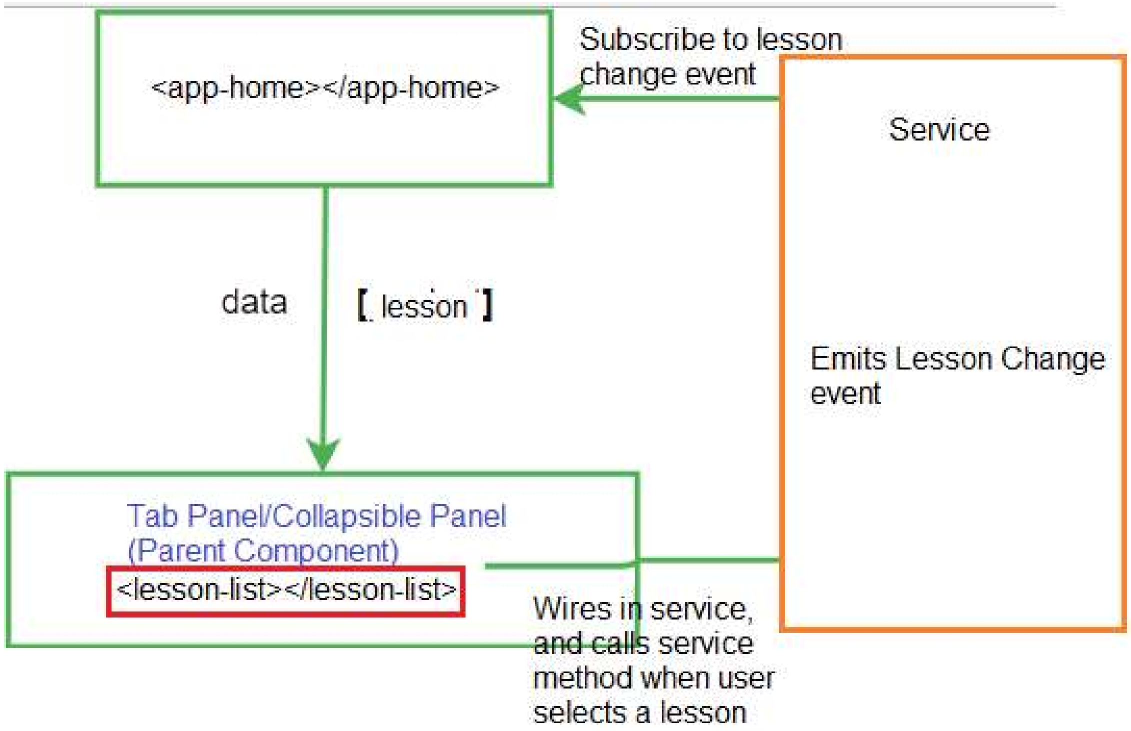
Angular Component Design



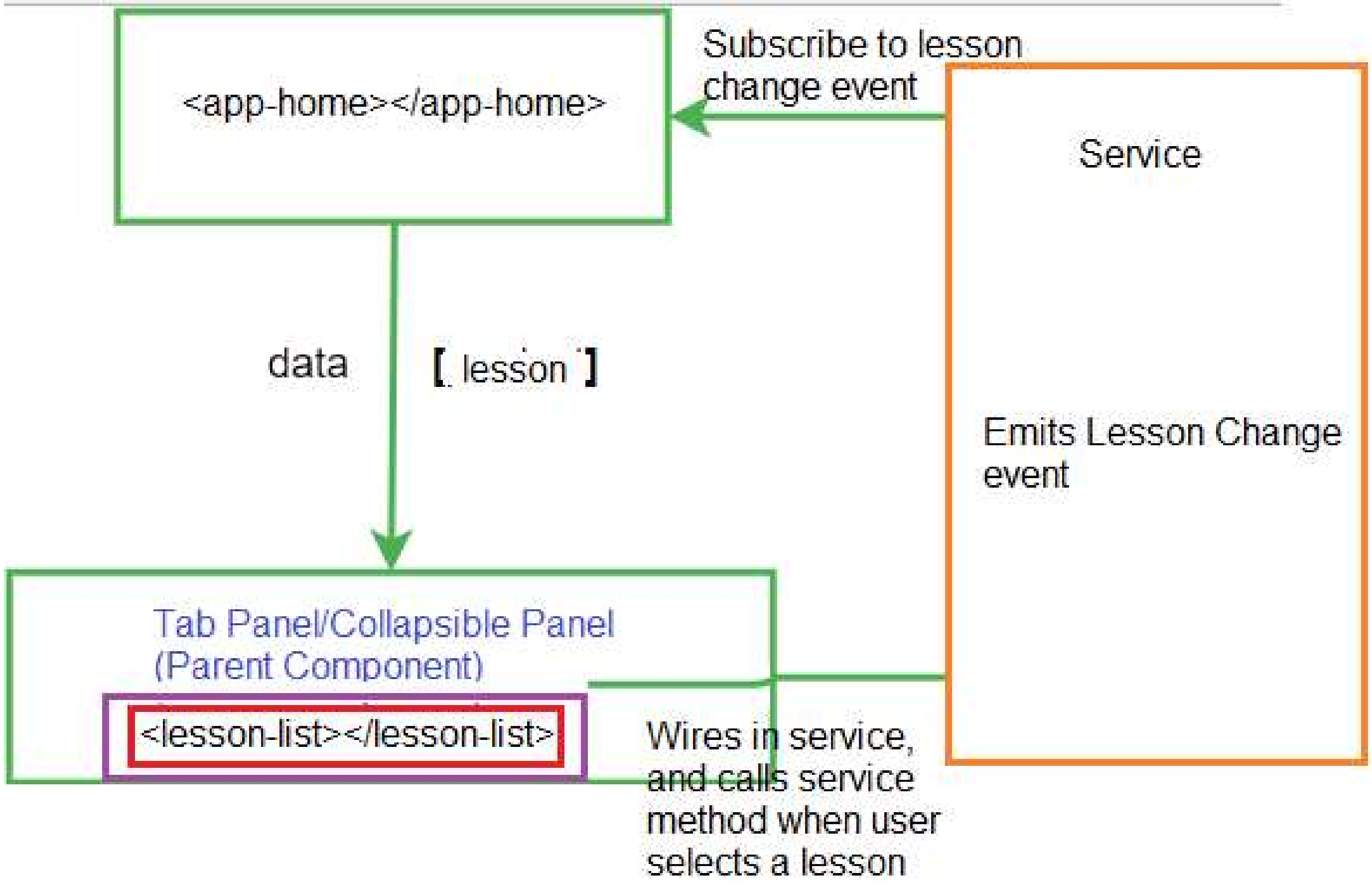
Angular Component Design



Angular Component Design


<app-home></app-home>
Subscribe to lesson change event
Service
data
[lesson]
Emits Lesson Change event
Tab Panel/Collapsible Panel (Parent Component)
<lesson-list></lesson-list>
Wires in service, and calls service method when user selects a lesson

Angular Component Design



Subscribe to lesson change event

<app-home></app-home>

Service

data

[lesson]

Emits Lesson Change event

Tab Panel/Collapsible Panel (Parent Component)

<lesson-list></lesson-list>

Wires in service, and calls service method when user selects a lesson

Contact Me

Simanta Sarma

Senior Mentor StackRoute

simanta.sarma@stackroute.in
ximanta.sarma@gmail.com

(M) 9972623673

https://simantas.wordpress.com/
https://github.com/ximanta