Introduction to SPA and Angular

Ground Rules

For a successful class, please:

• Arrive on time.

• Turn off cell phones.

• Assist your colleagues; show respect to all individuals regardless of their skill and knowledge level.

• Do not use class time to surf the net, check e-mail, or use instant messaging.

• Adhere to attendance policy as directed by your local training coordinator.

• Make the most of face-to-face training; ask questions and share your own insights and expertise.

• Leave the room only during break times or in the event of an emergency.

• Manage expectations for your other responsibilities.

Introduction to Angular

Module Objectives

At the end of this module, you should be able to understand:

- Single Page Applications (SPA)

- Traditional Applications Vs SPA

- AngularJS

- Angular

- Key Features

Topic List

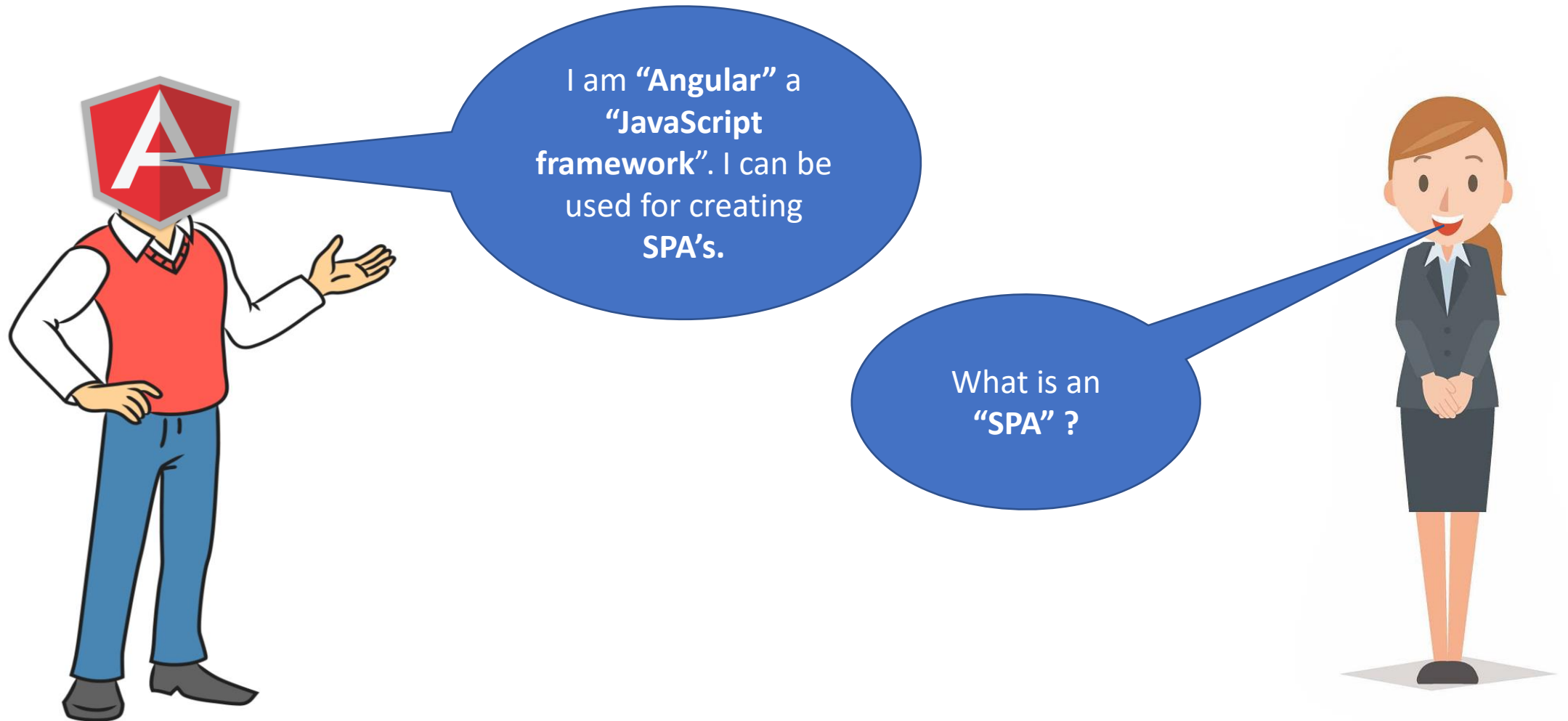| #No	Module Topics
1	Single Page Application and Traditional Application
2	Angular JS and Angular
3	Key features of Angular

Single Page Application

Single Page Application & Traditional Application
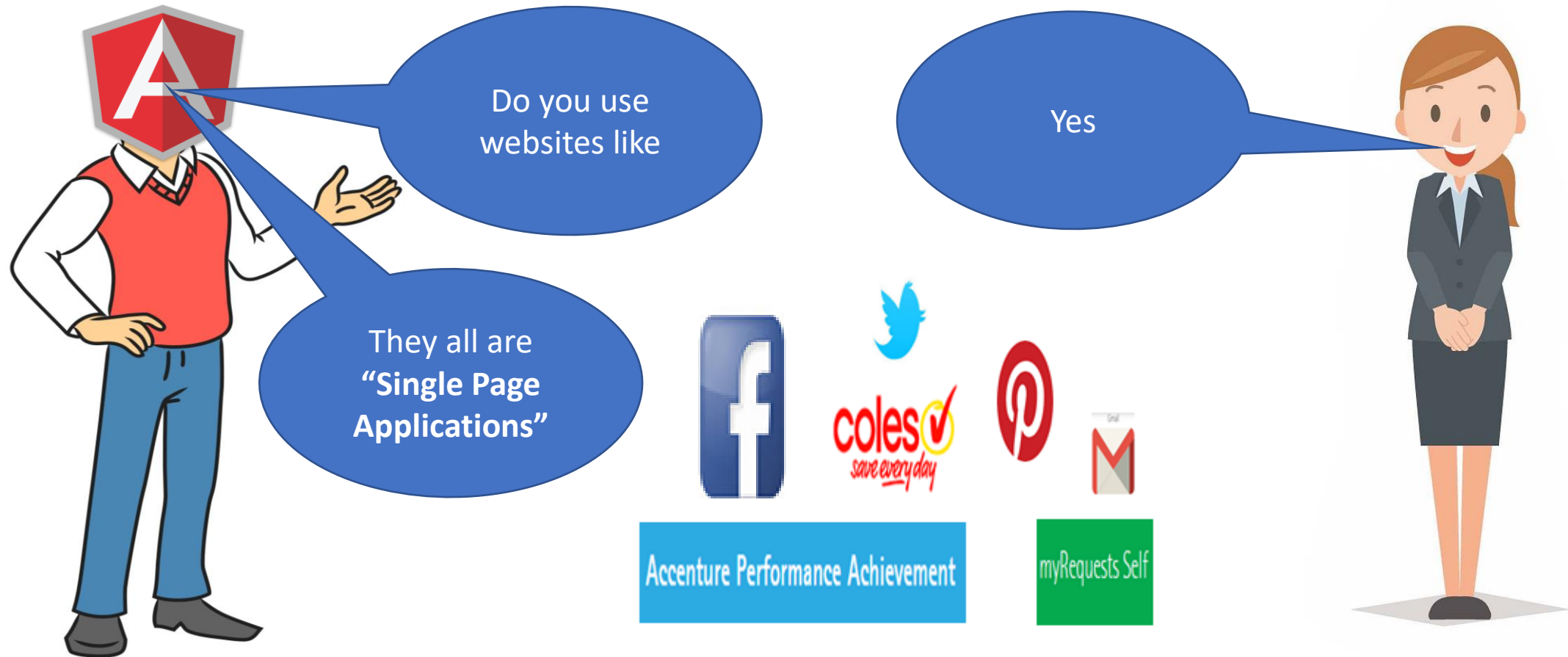
Introduction

We want to enhance our website using **"SPA"**
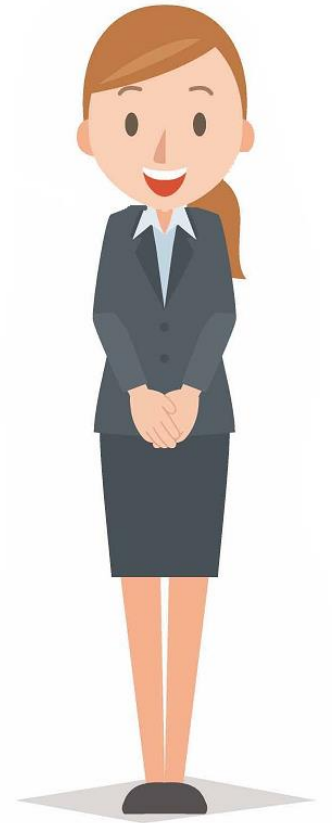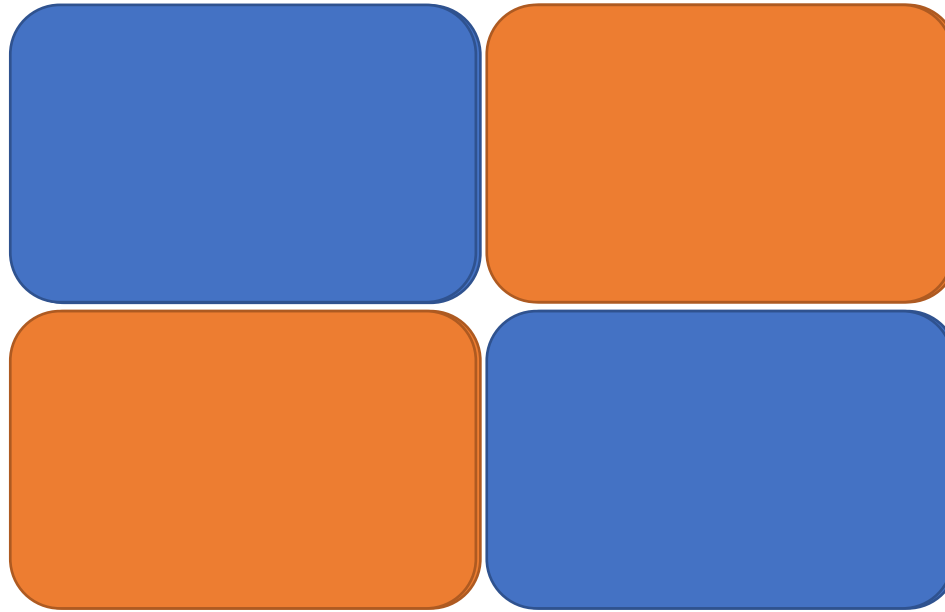
And for this we are exploring "Angular"

Single Page Application & Traditional Application

Introduction

Single Page Application & Traditional Application

Introduction
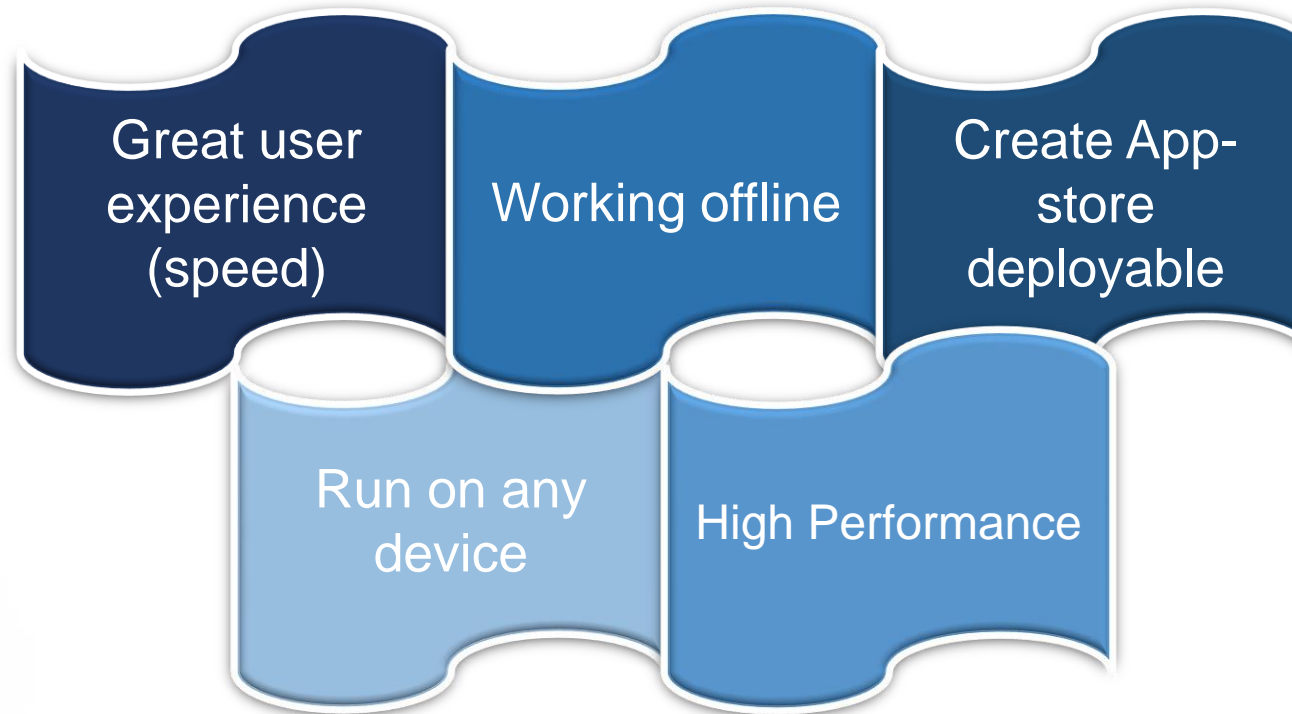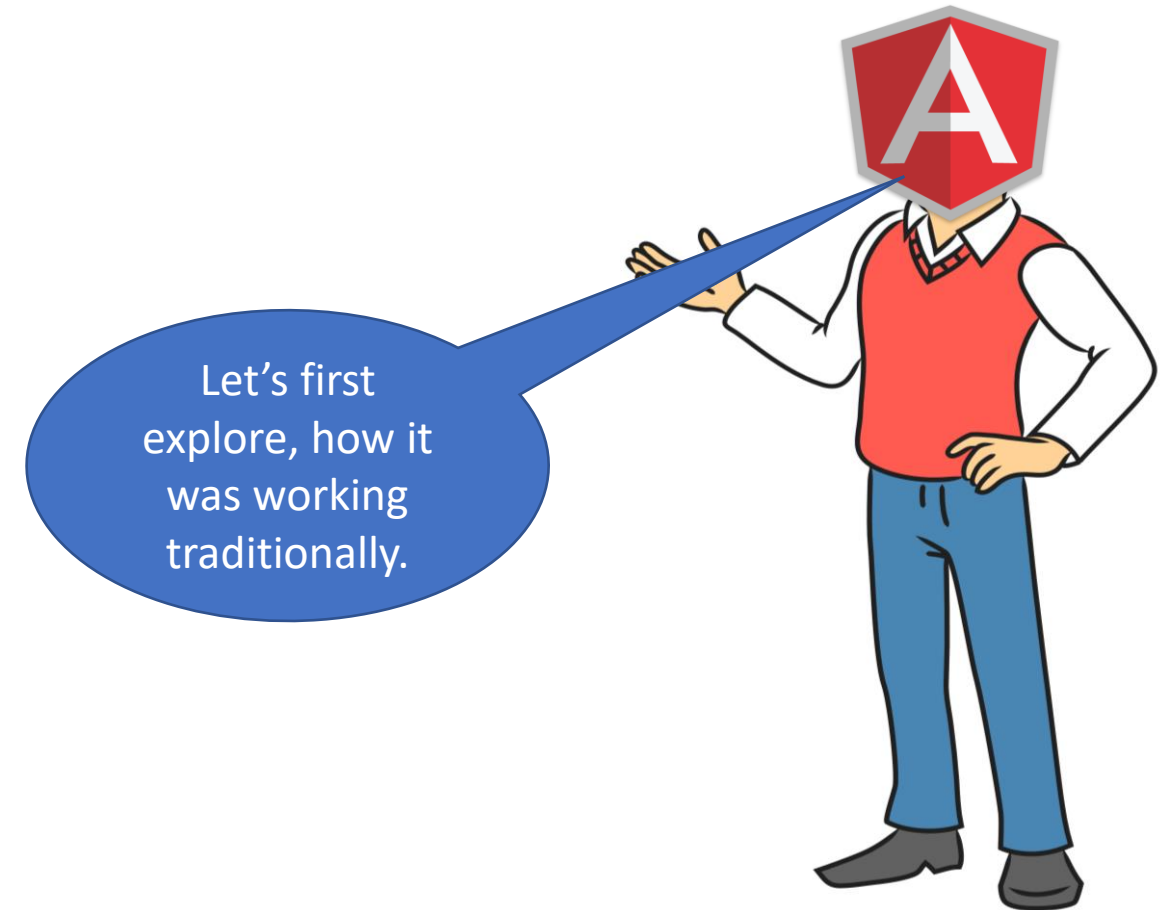
Single Page Application & Traditional Application

Introduction

Single Page Application & Traditional Application

Introduction

With SPA we can get benefits like

Great user experience (speed)

Working offline

Create App-store deployable

Run on any device

High Performance

Single Page Application & Traditional Application

Introduction

How does the **SPA** life Cycle function?
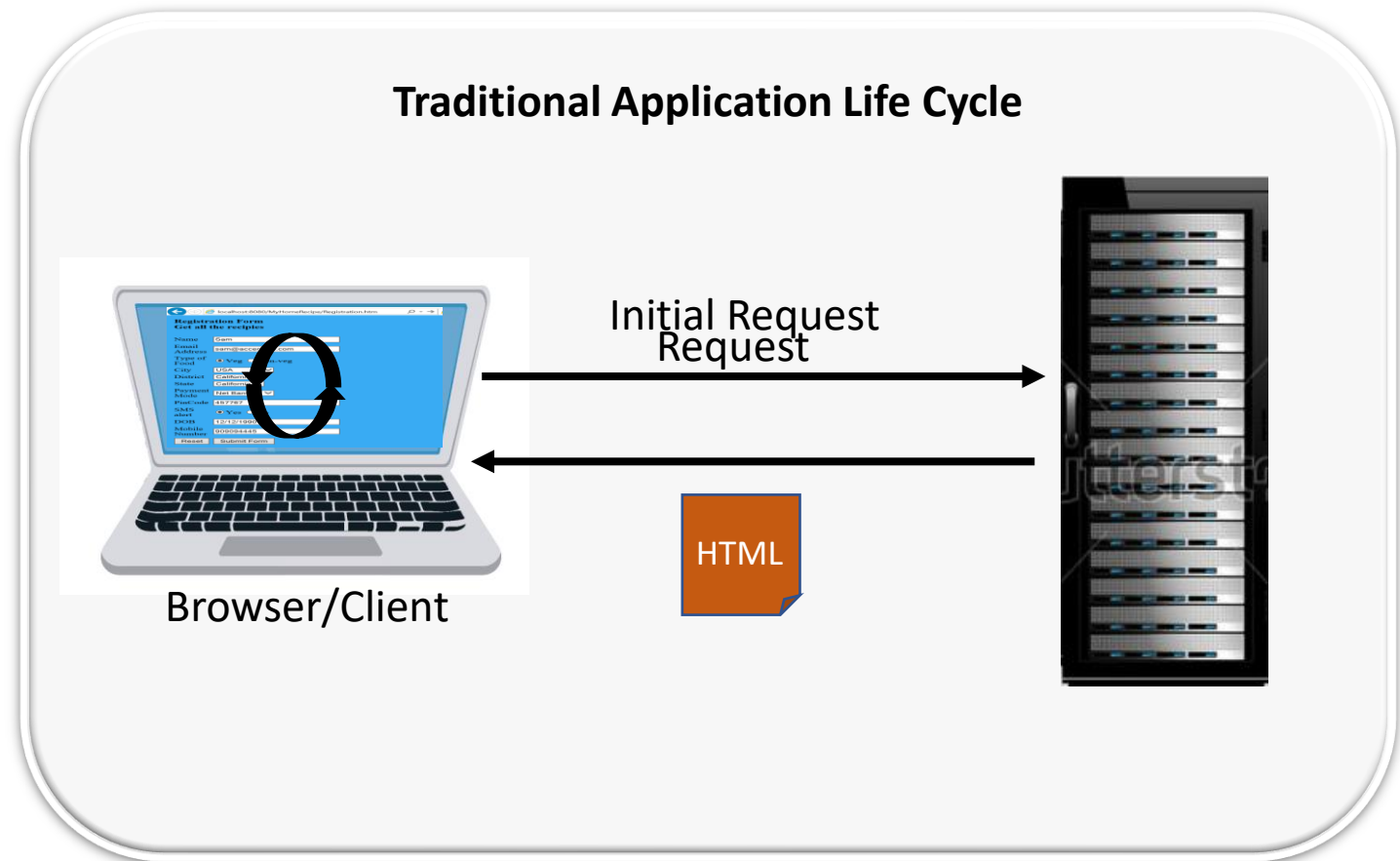
Let's first explore, how it was working traditionally.

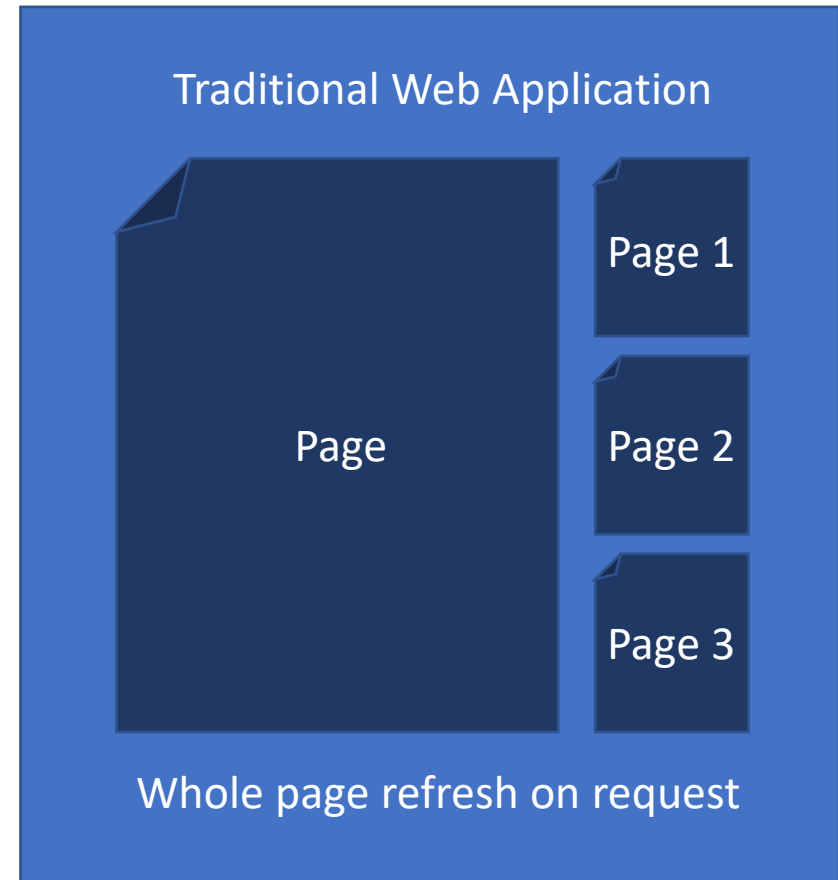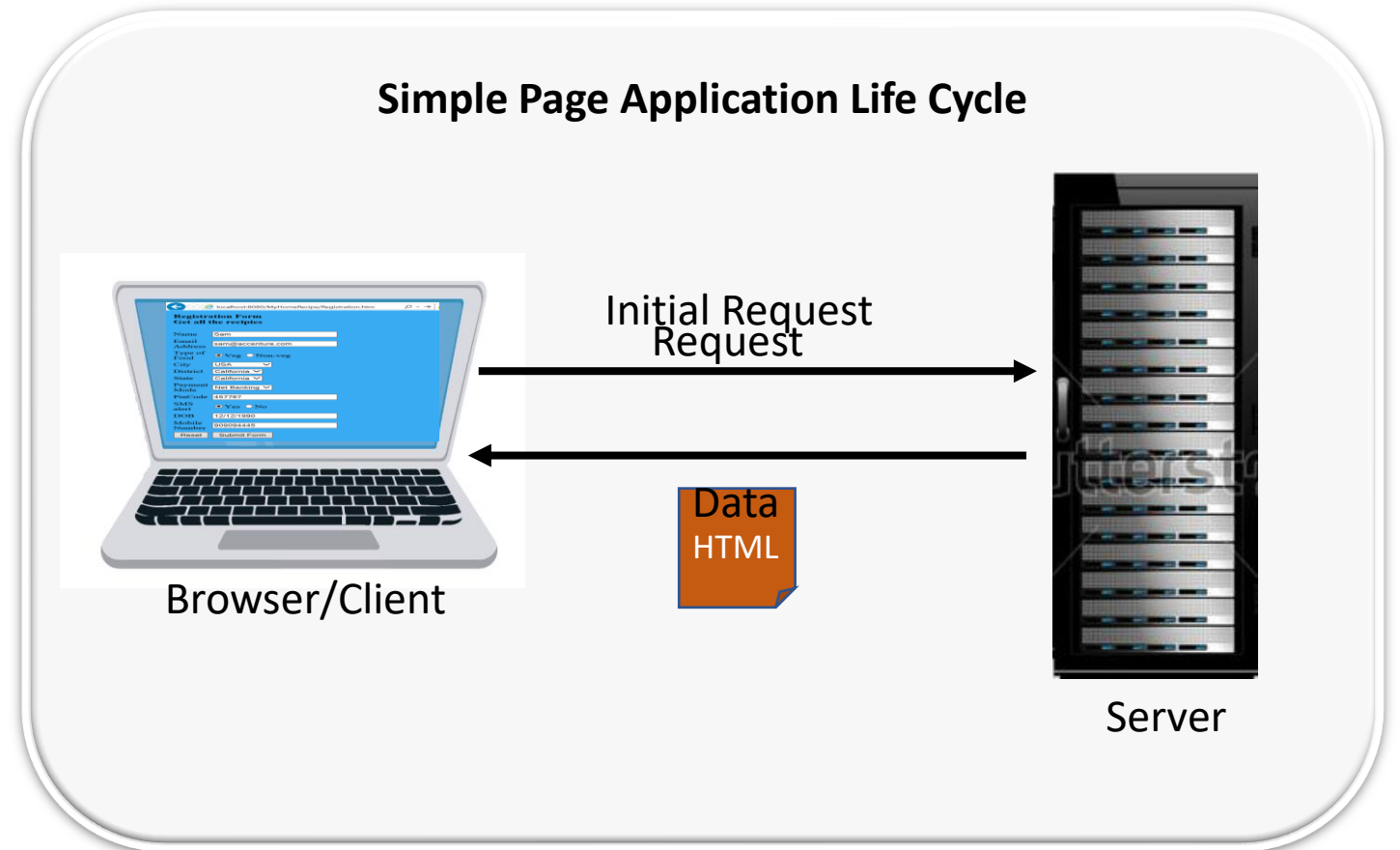# Single Page Application & Traditional Application

Traditional Application

In Traditional Applications, every time there is a request received from the client, the server sends a HTML document as response.

Reloads Page after every request

Traditional Application Life Cycle

Browser/Client

Initial Request
Request

HTML

Single Page Application & Traditional Application

Traditional Application

In Traditional Applications, every time there is a request received from the client, the server sends a HTML document as response.

Traditional Web Application

Page

Page 1

Page 2

Page 3

Whole page refresh on request

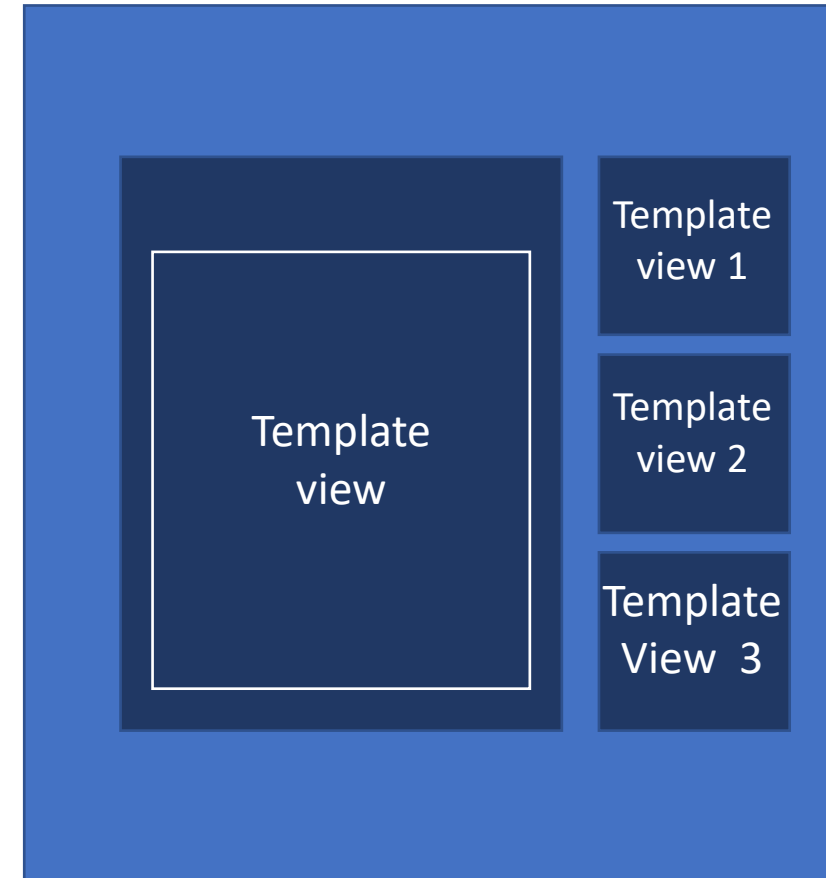# Single Page Application & Traditional Application

Single Page Application

In SPAs, HTML response is sent only the first time. During subsequent responses only the data is updated in the existing view(HTML).

Only Data will get loaded every time

Simple Page Application Life Cycle

Initial Request
Request

Data
HTML

Browser/Client

Server

Single Page Application & Traditional Application

Single Page Application

Template view

Template view 1

Template view 2

Template View 3

Single Page Application & Traditional Application
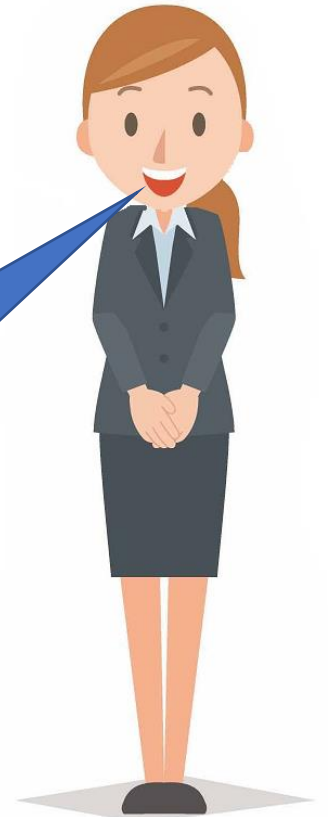
Single Page Application

Angular is one of the framework that can be used for creating Single Page Applications.

Let's explore

It's really interesting. Can you share some more information about history and journey of angular.

Angular JS & Angular

Angular JS and Angular

Angular JS

Referred as "**Angular.js**" or "**AngularJS 1.X**"

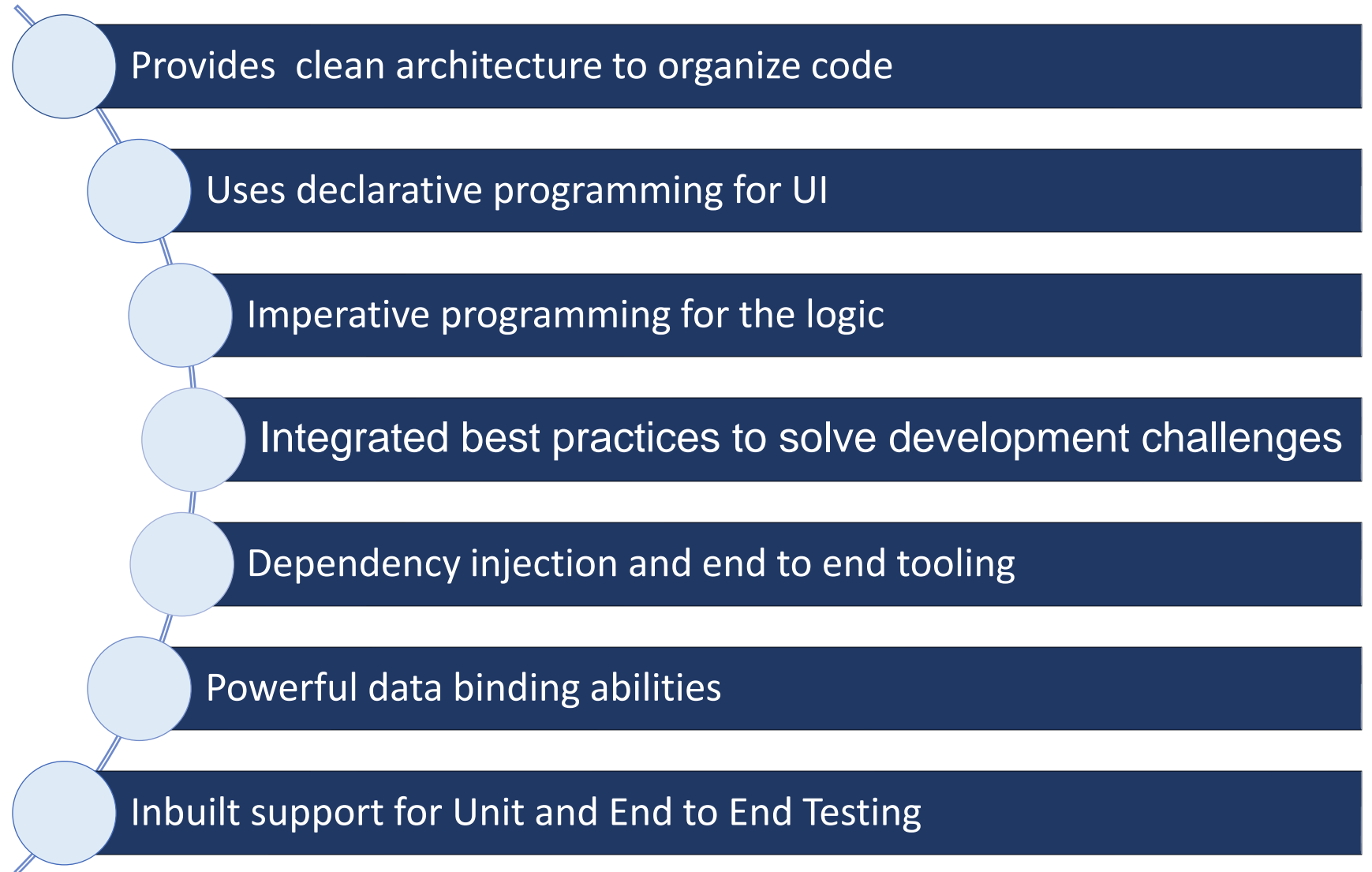JavaScript-based open-source framework

Help us to build CRUD style business applications

Supports modern desktop and mobile browsers

It uses a subset of jQuery called jQLite

Angular JS and Angular

Angular

- Provides clean architecture to organize code
- Uses declarative programming for UI
- Imperative programming for the logic
- Integrated best practices to solve development challenges
- Dependency injection and end to end tooling
- Powerful data binding abilities
- Inbuilt support for Unit and End to End Testing

Angular JS and Angular

Angular Language Choice

- No Transpilation required

ES 5

- Newer features like lexical scoping, classes, etc,.
- Requires Transpilation

ES 2015

Dart

- Non JavaScript based language

TypeScript

- Strong typing
- Has to be transpiled

Angular JS and Angular

Angular Key Features

- Improved Dependency Injection (DI)
- Annotation
- Dynamic Loading
- Templating
- Directives
- Routing
- Smaller and Faster
- Animation Support
- TypeScript 2.1 and 2.2 Compatibility
- View Engine

Module Summary

Now, you should be able to understand:

- Single Page Applications (SPA)

- Traditional Applications Versus Single Page Application flow

- Difference between AngularJS Angular

- Key Features of Angular

Thank You