



Service now - training

Sethuramalingam.N



Day 1:

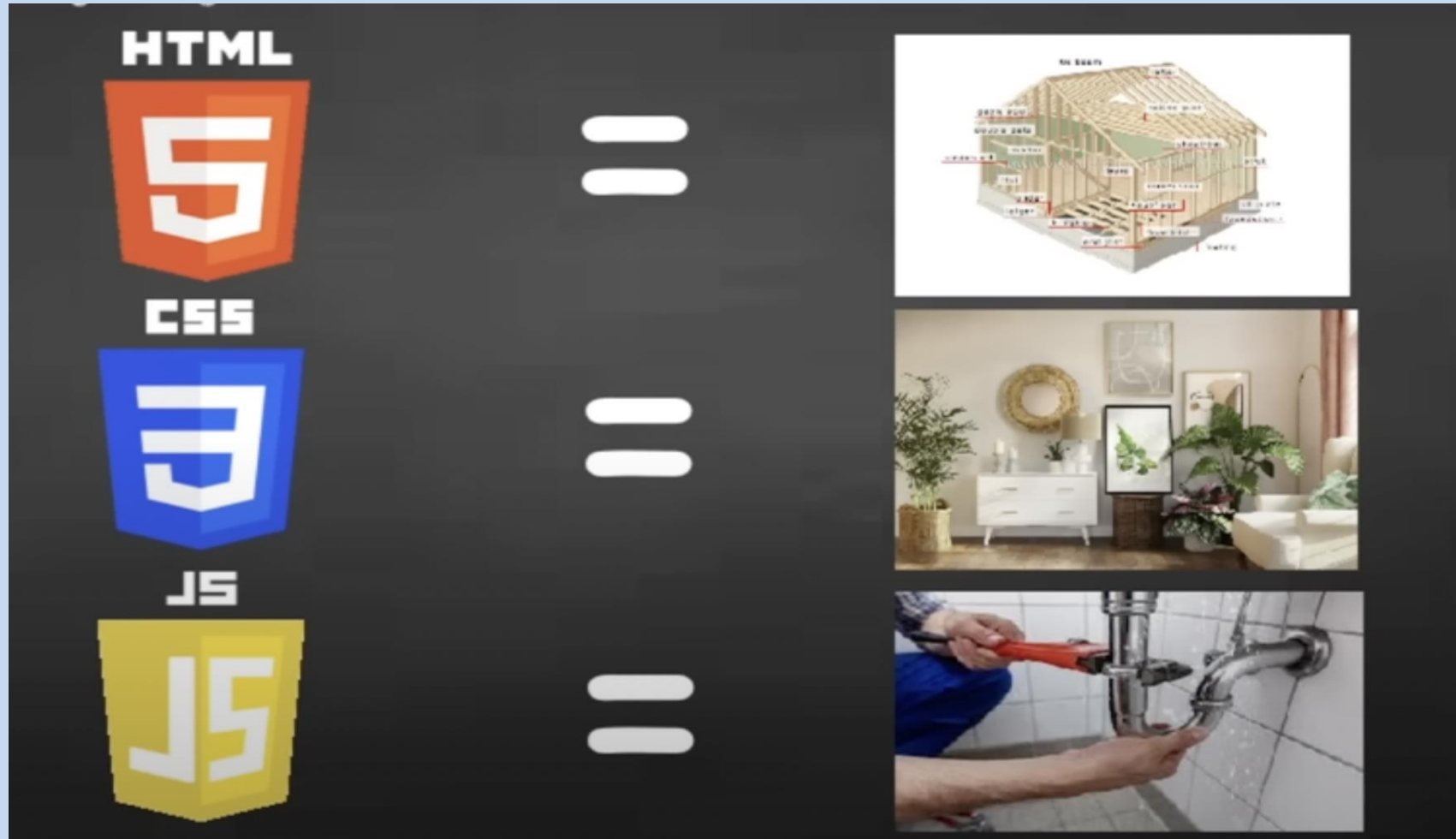
- Introduction to HTML.
- HTML forms.
- Introduction to CSS.
- CSS selectors.

Why HTML?



- HyperText Markup Language.
- Basic building block of the web, Allows you to add/modify the website content.
- Useful for Web developers, software developers, Marketing and sales professionals, Business owners, freelancer.

Building website is like building a house



What will you need?

1. Web browser:



2. Text Editor:





- How is html declared?
- Anchor links
- Image
- Audio
- Video
- List
- Table
- Span and div
- Meta
- I frame
- forms



- Different Ways of adding Css.
- Font properties
- Border
- Background
- Margins, paddings



Day 2:

- Html text formatting
- Background
- Margins, paddings
- Float
- Div positioning
- Pseudo
- shadow
- Icons
- Transform
- Animation
- Word wrap and z index



Day 2:

- Should never use !important.
- Should have a root class and write properly with child classes inside
- Should not directly point on element tags.
- Refer css selectors.



Day 3:

- Overview of Java Script
- Overview of JSON
- Event handlers
- Template literals
- Arrow functions
- Variable Scoping
- Strings & Arrays
- Creating Classes & functions
- DOM selection
- DOM manipulation
- Call back functions
- Stop events



- Static and dynamic webpages, java script is the scripting language needed for interaction with user.
- Light weight – untyped no strong typing, asynchronous, no extra software needed.
- Java script interpreter – doesn't have to be compiled to machine code
- A high-level programming language.
- Can be used on a server, client and in more complicated environments that are not web based like PDF docs, site specific browsers and desktop widgets.
- Multi paradigm-object oriented, imperative(with statements to change state of app), functional language.



What is ECMA Script?

- Java script uses ECMA script specifications – set of rules standardized by ECMA international.
- ECMA releases versions every year - last version update is ECMA 2022.
- As a java script developer, you need to be aware of latest version of ECMA and the features released.
- Implementation of ECMA script also includes JScript(Microsoft), ActionScript, dart-lang, C#, VBScript etc.



- Java Script Object Notation.
- Data representation format like XML, Yaml.
- Commonly used for API's, config's, games, text editor like vs code etc.
- Light weight, small size, Easy to read/write.
- Superset of Java script, easy to use with java script.
- Almost every language has some library to parse JSON strings into objects or classes – makes it working with Json is easy in a language.



JSON Types

- **Strings** “Hello World” “Kyle” “I”
- **Numbers** 10 1.5 -30 1.2e10
- **Booleans** true false
- **null** null
- **Arrays** [1, 2, 3] [”Hello”, “World”]
- **Objects** { “key”: “value” } { “age”: 30 }



Day 4:

- AJAX overview with example
- Introduction to Angular
- Angular architecture
- Angular MVC
- Set the dev environment
- Data and event binding(property, class, attribute, style, string interpolation, 1-way, 2-way bindings)

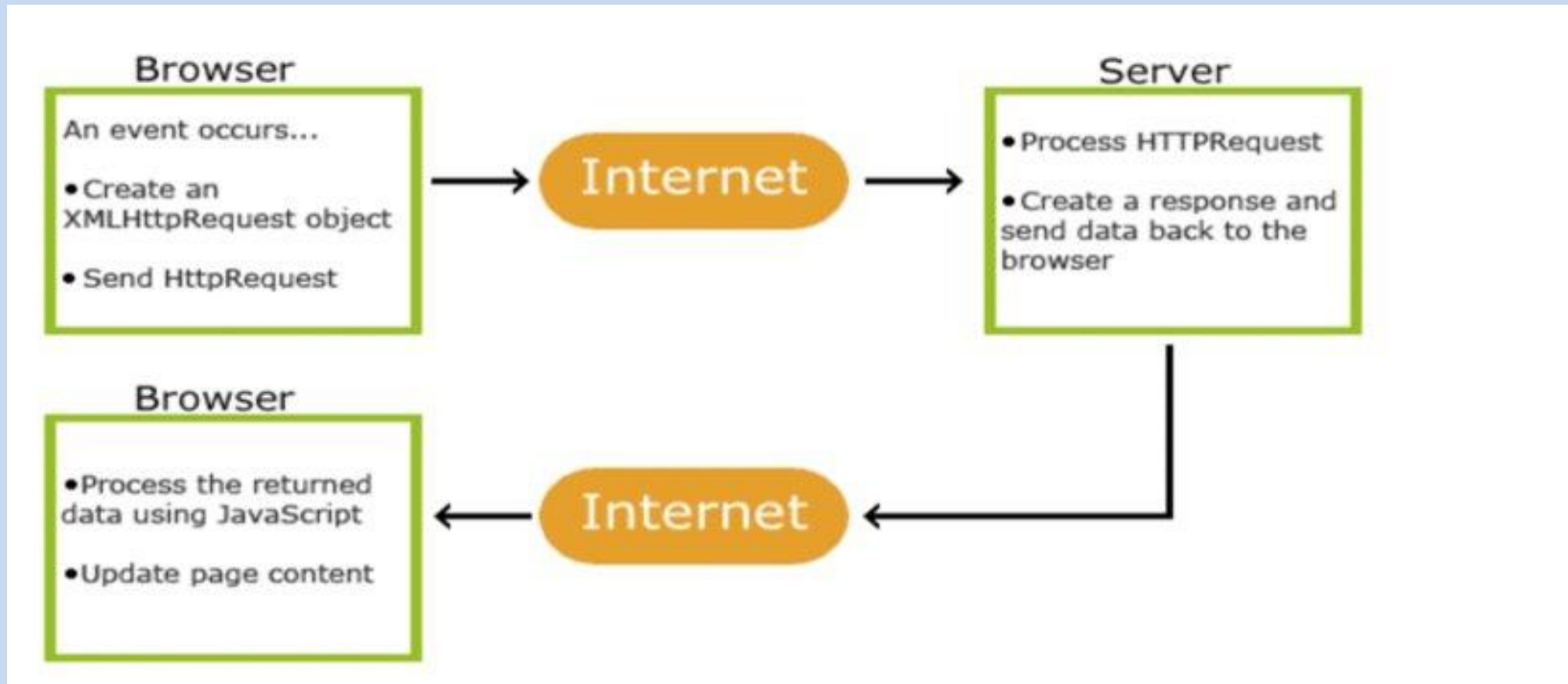


- Asynchronous JavaScript and XML.
- A technique that allows web pages to communicate with a server asynchronously without reloading the entire page.
- Makes web pages more interactive and responsive.
- Ajax uses a combination of a browser built in XMLHttpRequest object(to request data from a web server), Java script, HTML DOM(to display or use the data)
- Ajax can just use XML/plain text/JSON text for transporting, as name suggests it is not just XML.



- Allows web pages to be updated asynchronously by exchanging data with the web server.
- It is also possible to update parts of the web without reloading the entire page.
- In some way Ajax is used in Gmail, google maps, you tube, face book etc

AJAX flow



- An Api that provides client functionality for transferring data between client and server.
- Easy way to retrieve data from a url without having to do a full page refresh.
- It is not limited to XML, you can also use JSON, file, FTP.
- Example for creating XMLHttpRequest:

```
var xhttp;  
if(window.XMLHttpRequest) {  
    xhttp = new XMLHttpRequest();  
}
```



XMLHttpRequest Properties

- onreadystatechange
- readyState
- response
- responseText
- responseType
- responseURL
- responseXML
- status
- statusText
- timeout
- ontimeout
- upload
- withCredentials

XMLHttpRequest Methods

- abort()
- getAllResponseHeaders()
- getResponseHeader()
- open()
- overrideMimeType()
- send()
- setRequestHeader()



Sending a Request

```
xhttp.open("GET", "somefile.txt", true);
```

```
xhttp.send();
```



GET / POST

- **GET** is faster but should only be used to request data from a server
- **POST** is usually used to send a large amount of data to the server or send user input



A good example for GET would be a search form and POST would be a user registration form

Getting A Response

```
xmlDoc = xhttp.responseXML;  
txt = "";  
x = xmlDoc.getElementsByTagName("ARTIST");  
for (i = 0; i < x.length; i++) {  
    txt += x[i].childNodes[0].nodeValue + "<br>";  
}  
document.getElementById("demo").innerHTML = txt;
```




Events

onReadyState – Stores a function to be called everytime the readyState changes

readyState - Holds the status of the XMLHttpRequest object (0 - 4)

- 0: request not initialized
- 1: server connection established
- 2: request received
- 3: processing request
- 4: request finished and response is ready

Status – Request status

200: OK

404: Page Not Found



- A framework for building client applications in HTML, CSS and java script/typescript.
- Why angular? Why not just java script? Because when application grows java script becomes very difficult to maintain (Really hard concepts like Revealing Module pattern, Prototype pattern needs be implemented).
- Vanilla java script is hard to test also.

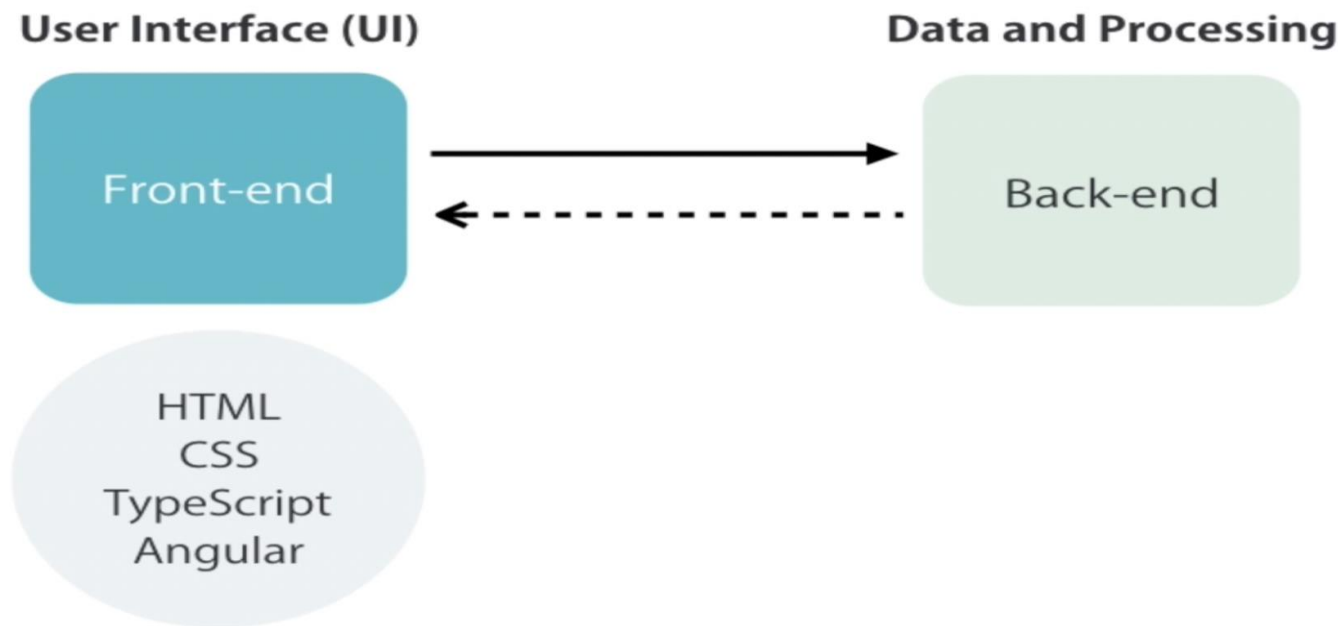


Benefits of Using Angular

- Gives our applications a clean structure
- Includes a lot of re-usable code
- Makes our applications more testable



Architecture



HTTP Services / APIs

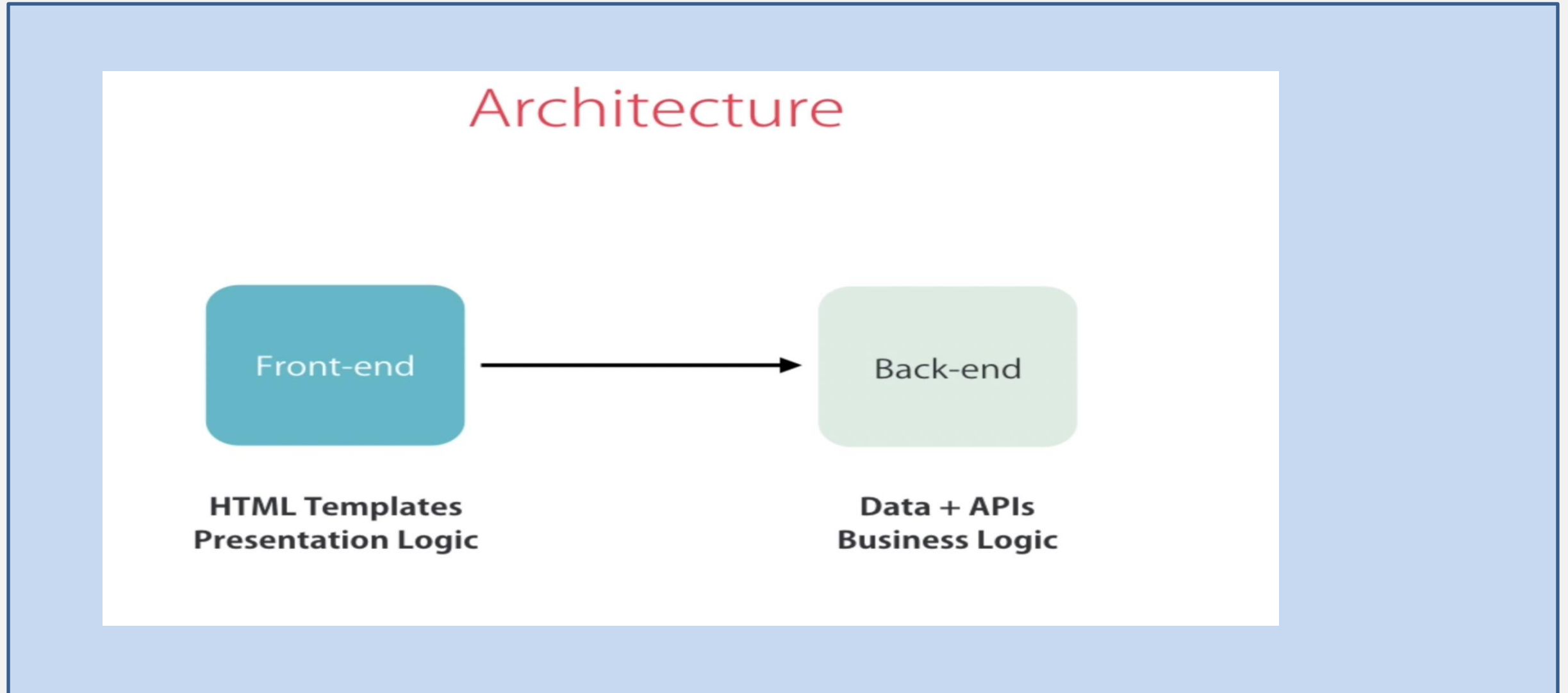
Endpoints that are accessible via the HTTP protocol.

HTTP Services





Angular architecture(MVVM)





- Model View Controller is a design pattern for developing web applications.
- Model: Responsible for storing and managing the applications data.
- View: Responsible for displaying the data to the user.
- Controller: Responsible to mediating between Model and view.



- Separation of concern: The model, view, controller are all independent of each other which makes code easier to understand and maintain.
- Reusability: The code can be reused in other applications which can save time and reduce cost.
- Testability: It can be easily tested in isolation which can help ensure application is working.



Day 5:

- Input, Output properties(p->c and c->p)
- Life cycle hooks
- Modules
- Directives
- Filters/pipes
- Service(custom, http)
- Routing(basic and lazy loading modules)



Questions ?

Get in touch

USA

Prodapt North America
Tualatin: 7565 SW Mohawk St.,
Phone: +1 503 636 3737

Dallas: 1333, Corporate Dr., Suite 101, Irving
Phone: +1 972 201 9009

New York: 1 Bridge Street, Irvington
Phone: +1 646 403 8161

CANADA

Prodapt Canada Inc.
Vancouver: 777, Hornby Street,
Suite 600, BC V6Z 1S4
Phone: +1 503 210 0107

UK

Prodapt (UK) Limited
Reading: Davidson House,
The Forbury, RG1 3EU
Phone: +44 (0) 11 8900 1068

EUROPE

Prodapt Solutions Europe
Rijswijk: De Bruyn Kopsstraat 14
Phone: +91 8939843230

Prodapt Consulting BV
Rijswijk: De Bruyn Kopsstraat 14
Phone: +31 (0) 70 4140722

Prodapt Germany GmbH
Münich: Brienner Straße, 80333
Phone: +31 (0) 70 4140722

Prodapt Switzerland GmbH
Zürich: In Gründung, 8008

SOUTH AFRICA

Prodapt SA (Pty) Ltd.
Johannesburg: No. 3,
3rd Avenue, Rivonia
Phone: +27 (0) 11 259 4000

INDIA

Prodapt Solutions Pvt. Ltd.
Chennai: Prince Infocity II, OMR
Phone: +91 44 4903 3000

"Chennai One" SEZ, Thoraipakkam
Phone: +91 44 4230 2300

IIT Madras Research Park II,
3rd floor, Kanagam Road, Taramani

Bangalore: "CareerNet Campus"
2nd floor, No. 53, Devarabisana Halli,
Phone: +91 80 4655 7008

THANK YOU!