



SNS COLLEGE OF TECHNOLOGY

(An Autonomous Institution)

Re-accredited by NAAC with A+ grade, Accredited by NBA(CSE, IT, ECE, EEE & Mechanical)
Approved by AICTE, New Delhi, Recognized by UGC, Affiliated to Anna University, Chennai



ENTERPRISE DATA



Course: **Mobile Application Development**

Unit : III – Building Blocks of Mobile Apps - II

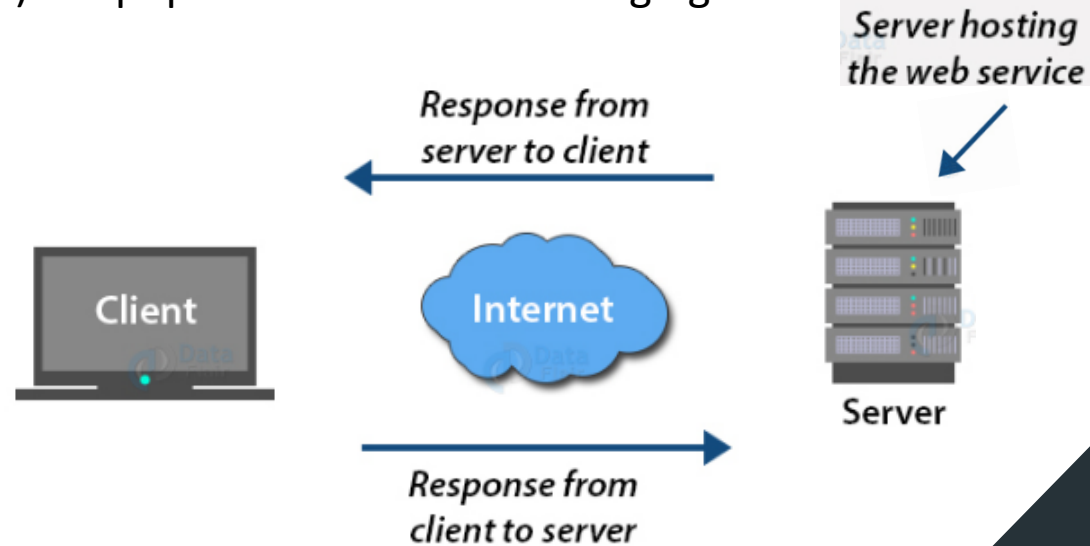
Class / Semester: II MCA / III Semester

Department of MCA



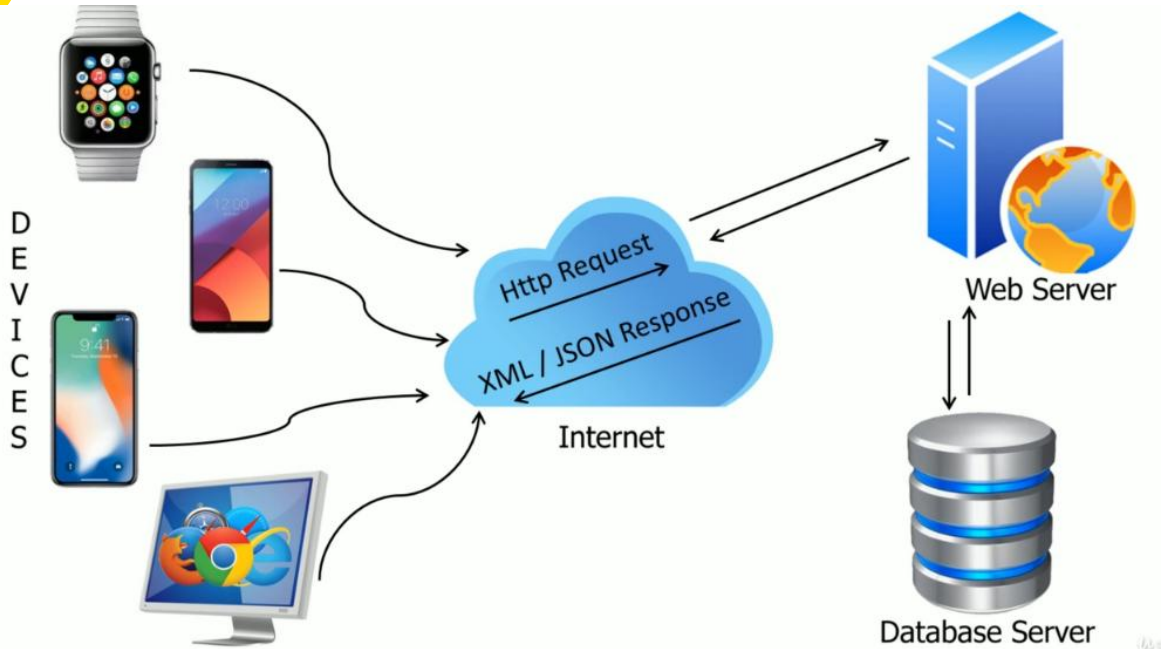
WEB SERVICE

- ❑ Web service - standard for exchanging information between applications on network
- ❑ RESTful web services are light weight, highly scalable and maintainable
- ❑ JSON3 (JavaScript Object Notation) is a popular format for exchanging small chunks of data in key–value pairs





ILLUSTRATION



Name	Age
Ramkumar	Saran

XML

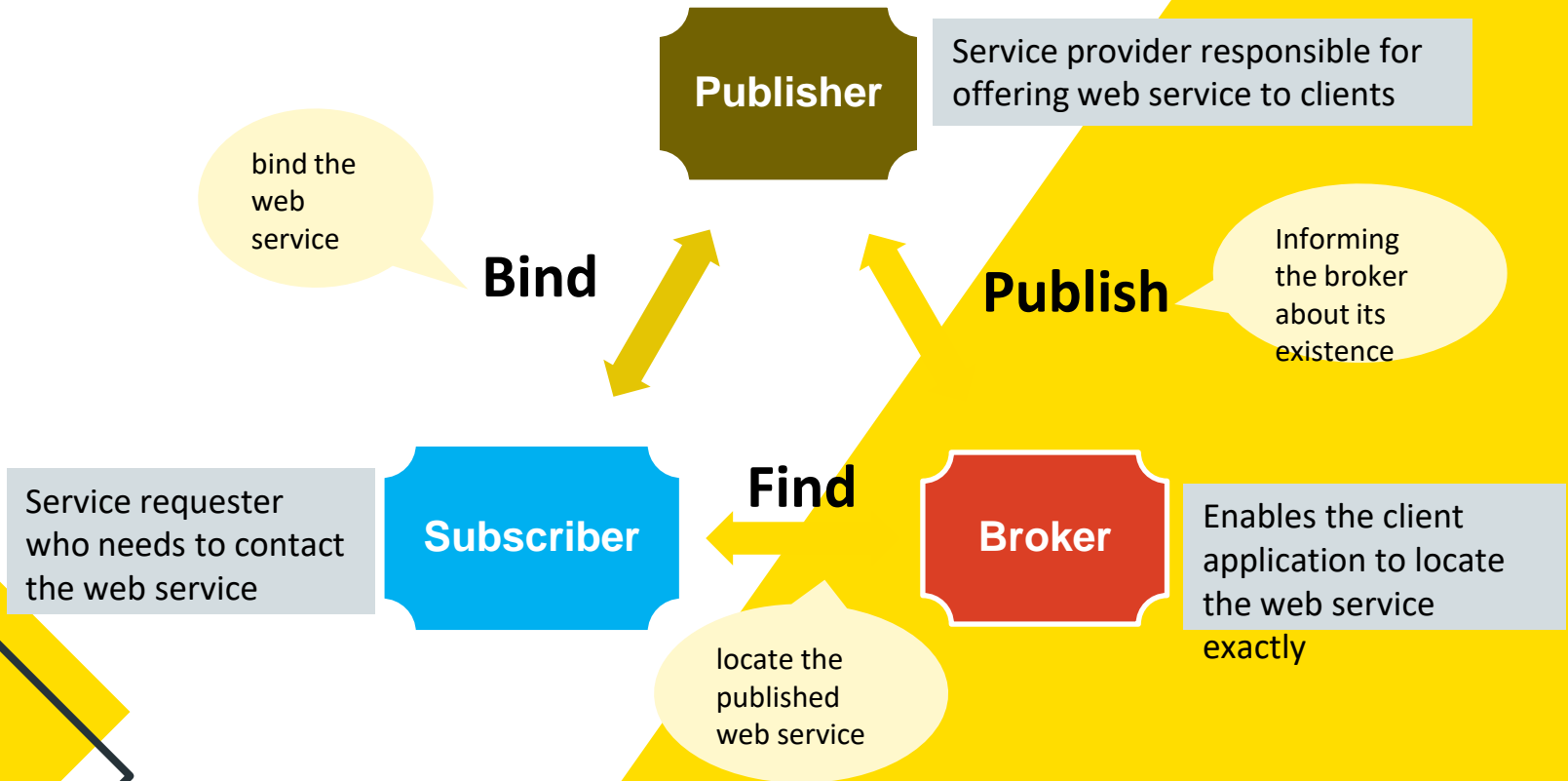
```
<cname>
<firstname>Ramkumar</firstname>
>
<age>23</age>
</cname>
```

JSON

```
Name
{
("firstname": "Ramkumar",
"age": 23)
}
```



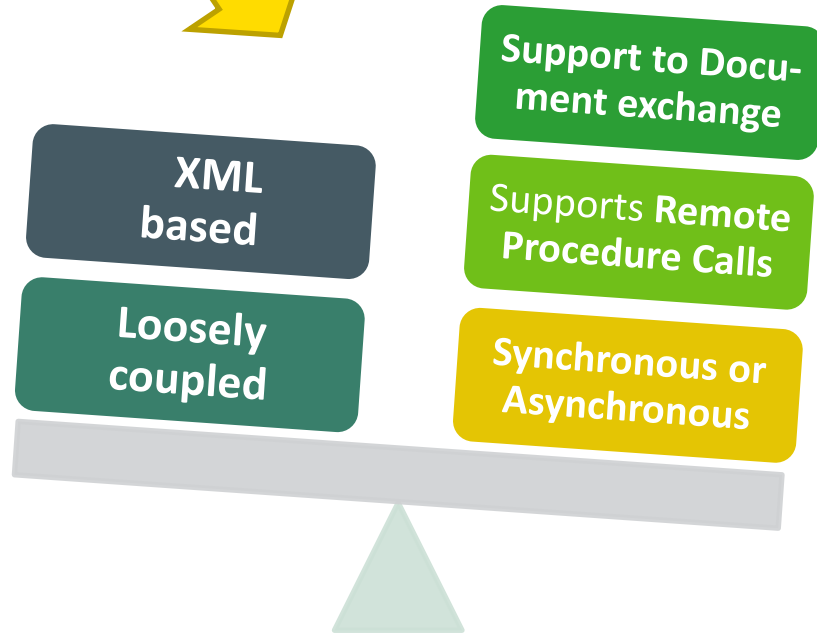
Android Web Service Components





Web Service Components

characteristics





Types of Web Services

XML based protocol for the exchange of data between devices over internet

XML- RPC

UDDI (Universal Descriptive, discovery, and integration) is an standard for detailing, publishing and discovering new web services

UDDI

SOAP (Simple object access protocol) is an web service protocol for the exchange of data /docs over HTTP/ SMTP

SOAP

REST (Representation al State Transfer) provides communication and connectivity between devices and the internet

REST



Advantages

Advantages

- interoperability among Applications
- Reusability
- faster communications
- use a quality industry-standard protocol for communication
- low-cost internet web services
- deployed over the standard internet technologies

Limitations

- Web services do not access from the browser
- don't leverage emerging Web developments
- HTTP protocol used by web services is not reliable and is insecure



Implementation

- ❑ To perform network operations in an application, we set permissions in manifest file

```
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

- ❑ To check the network connectivity using user-defined function , before we proceed

```
private boolean checkNetworkAccess()  
{  
    ConnectivityManager connectivityManager = (ConnectivityManager)  
        getSystemService(CONNECTIVITY_SERVICE);  
    NetworkInfo info = connectivityManager.getActiveNetworkInfo();  
    if (info != null && info.isConnected())  
    { return true;  
    } else {  
        Toast.makeText(MainActivity.this, "No network access, network resource not accessible",  
            Toast.LENGTH_SHORT).show();  
        return false; } }
```




Implementation

- ❑ Once the network connectivity is determined, the app needs to initiate an HTTP request to exchange data with RESTful Web service
- ❑ HttpURLConnection is to initiate HTTP request which facilitates CRUD operations using PUT, GET, POST, and DELETE, HTTP methods

```
HttpURLConnection connection = null;
try {
    URL url = new URL
("http://10.0.2.2:8080/ExpenseTracke/fetchExpServlet");
    connection = (HttpURLConnection) url.openConnection();
    connection.setReadTimeout(2000);
    connection.setConnectTimeout(4000);
    connection.setRequestMethod("GET");
    connection.connect();
    int responseCode = connection.getResponseCode();
    if (responseCode == 200) {
        InputStream inputStream = connection.getInputStream();
        BufferedReader bufferedReader = new BufferedReader(
new InputStreamReader(inputStream));
        StringBuilder builder = new StringBuilder();
        String line;
        while ((line = bufferedReader.readLine()) != null)
        {
            builder.append(line);
        } response = builder.toString(); }
}
```



Implementation

- ❑ Create a new project and go to activity_main.xml file, create List view
- ❑ Create another layout file list_row.xml and add the code

```
<?xml version="1.0" encoding="utf-8"?>
```

<RelativeLayout

```
xmlns:android="http://schemas.android.com/apk/res/android"  
android:layout_width="fill_parent"  
android:layout_height="wrap_content"  
android:orientation="horizontal"  
android:padding="5dip">
```

```
<!--TextView to display the name-->
```

<TextView

```
android:id="@+id/name"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:textSize="17dp"  
android:textStyle="bold" />
```

```
<!--TextView to display the designation-->
```

<TextView

```
android:id="@+id/designation"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:layout_below="@id/name"  
android:layout_marginTop="7dp"  
android:textColor="#343434"  
android:textSize="14dp" />
```

```
<!--TextView to display the location-->
```

<TextView

```
android:id="@+id/location"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:layout_alignBaseline="@+id/designation"  
android:layout_alignBottom="@+id/designation"  
android:layout_alignParentRight="true"  
android:textColor="#343434"  
android:textSize="14dp" />
```

```
</RelativeLayout>
```



REFERENCES

- ❑ Anubhav Pradhan, Anil V Deshpande, “Composing Mobile Apps using Android”, Wiley Edition, 2014
- ❑ https://www.tutorialspoint.com/android/android_application_components.htm
- ❑ <https://www.javatpoint.com/android-core-building-blocks>



**Thank
You**