Assignment:

Server Side Rendering of Content to play on Virtual Display Screen (VDS).

Expected Outcome:

Design and implement back-end REST APIs for following

1. CRUD operations for VDS

   Create and save VDS along with unique URL generated

   List of VDS along with their details including unique URL

2. CRUD operations for Content

   Create content: Upload content (images/videos) to S3 and save the S3 URL along with details like duration, contentType, tags etc in DB.

   Create content mapping to VDS: Create content to VDS mappings and save in DB. This will be used later to fetch content for each VDS.

   List of content with their mappings to VDS

3. Respond to requests from VDS

   Design a common API endpoint to receive requests from each unique URL (e.g. {baseURL}/endpointname/{uniqueId})

   The API takes care of the following…

   i. Fetch VDS details from DB, based on the parameters in unique URL
   ii. Fetch content mapped to that VDS from the DB
   iii. Prepare and render a webpage with the content and scripts, using server side rendering (e.g. you can use template engines such as “pug”)
   iv. Send response to the VDS

Definitions and details for understanding:

What is a Virtual Display Screen (VDS) and what does it do?

A VDS is an internet connected screen with capability to play any content fetched from server using a unique URL. The unique URL acts as the identifier for the screen. All requests to the Server are made using the unique URL. The Server, based on the unique URL, prepares and sends content in the form of server side rendered web page. The VDS receives the web page and displays it. The web page has content details and scripts to play the content as per the details.

Note: The content needs to play automatically and responsively in full screen mode occupying the entire viewport.

What is the unique URL and how it gets created for each VDS?

On the admin dashboard, we provide details of each VDS and submit it to the server which creates a URL that uniquely identifies the VDS and saves the unique URL along with the VDS details for one-one mapping. The URL essentially points to an API end point on the server which has server side logic to respond to requests from that URL.

User input fields for creating VDS and unique URL:

```
{
name:
referenceId:
mediaType: VDS

screenWidth:
screenHeight:
screenSizeDiagonal:
aspectRatio: 16:9/16:10/4:3/2:1/1:1/custom/1:1/1:2/3:4/10:16/9:16
resolutionWidth:
resolutionHeight:
orientation: +90,-90,180,0

siteType: Stationary/Moving
location: lat,long
locationAddress:
```

premises:

premisesType:

placement:

keywords:

photos:

spotduration:

spotprice:

currency:

pricingUnit: CPM/impression

}

Send a POST request with above details to Server, which takes care of below steps…

Generate uniqueURL: {baseURL}/endpointname/{id}

Generate QR code for this URL

Save the details in DB and

Send response with those details

What is contained in the web page prepared by the Server and how is it prepared?

Each unique URL sends request to an API endpoint on the server. The server side implementation of the API takes care of all the aspects including

1. mapping the URL to VDS screen
2. fetching content (list of items to play) specific to that VDS screen and
3. preparing the web page with content and scripts

The server side rendered web page contains

1. The content schema which contains a list of items to display along with duration and logic for playing each item.
2. Player Script to play content according to the logic and item duration in the schema
3. Scripts to send playback stats to the server

The web page is prepared using Server Side rendering based on the parameters in the request coming from each VDS when the unique URL is opened/loaded. It is generated fresh on the server, every time the unique URL is opened/loaded on the VDS and sent in response to the request from the VDS.