**PROJECT DEVELOPMENT PHASE**

|  |  |
| --- | --- |
| **Date** | **01 November 2023** |
| **Team ID** | **NM2023TMID08494** |
| **Project Name** | **Creating a social media Ad campaign in Facebook** |

**No of functional features included in the function**

Ad Copy: Text content to communicate your message, call-to-action, and details about your product or service.

Visual Media: Images, videos, or carousels that grab attention and showcase your offering.

Call-to-Action (CTA): Buttons or text encouraging users to take action (e.g., "Shop Now," "Learn More," "Sign Up," etc.).

Targeting Parameters: Audience segmentation based on demographics, interests, behaviors, etc., to reach the right users.

Ad Placement: Selection of where the ad appears (feed, stories, in-stream, etc.).

Budget & Schedule: Setting the spending limit and duration of the campaign.

Analytics & Tracking: Tools to monitor ad performance, track conversions, and measure the success of the campaign.

Each of these features plays a crucial role in creating an effective ad campaign on Facebook.

**Code-Layout ,Readabalitity and Reusability**

Folder Structure:

AdCampaign

assets

images

videos

src

index.html (or other frontend file)

styles

main.css (or other CSS files)

scripts

main.js (or other JavaScript files)

components (for reusable components)

header.js

footer.js

config

facebookAPI.js (for handling Facebook API calls)

utils

helperFunctions.js (common utility functions)

Code Structure:

HTML (index.html or equivalent)

html

<!DOCTYPE html>

<html>

<head>

<title>Ad Campaign</title>

<link rel="stylesheet" href="styles/main.css">

</head>

<body>

<header>

<---Include header component -->

</header>

<main>

<!-- Your ad content -->

</main>

<footer>

<!-- Include footer component -->

</footer>

<script src="scripts/main.js"></script>

</body>

</html>

JavaScript (main.js or equivalent)

javascript

// Import necessary modules

import { fetchData } from './utils/helperFunctions.js';

import { facebookAPI } from './config/facebookAPI.js';

import { Header, Footer } from './components/header.js';

// Your ad campaign logic

// Utilize reusable components and functions

CSS (main.css or equivalent)

C

**Utilization of Algorithms Dynamic Programming Optical memory Utilization**

**1**. **Project Focus**

Objective: Enhance understanding or promote awareness of algorithms, dynamic programming, and optical memory.

Target Audience: Students, tech enthusiasts, or professionals in related fields.

**2. Content Strategy**

Educational Posts: Share engaging content about algorithms, dynamic programming, and optical memory.

Visuals & Infographics: Create visually appealing content explaining these concepts.

Tutorials & Tips: Offer helpful tips or tutorials related to these topics.

**3. Engagement Techniques**

Question & Answer Sessions: Host live Q&A sessions discussing these topics.

Challenges or Puzzles: Post challenges related to algorithms, dynamic programming, etc.

User-Generated Content: Encourage users to share their experiences or insights.

**4. Optimization and Utilization**

Algorithm Insights: Share real-world applications and case studies of algorithm utilization.

Dynamic Programming Success Stories: Showcase successful projects or systems employing dynamic programming.

Optical Memory Advantages: Highlight benefits and advancements in optical memory usage.

5**. Campaign Execution**

Create a Content Calendar: Plan regular posts focusing on these topics.

Utilize Facebook Ads: Target audiences interested in technology, education, or programming.

Collaborate & Partner: Work with influencers or experts in these fields for more credibility and reach.

**6. Measuring Success**

Metrics to Track: Engagement rates, reach, shares, comments, and website traffic.

Adaptation: Analyze data to refine the strategy for better performance.

**7. Community Building**

Facebook Groups: Create a community where people can discuss these topics.

Networking: Engage with like-minded groups or communities to expand reach and impact.

Remember, this is a high-level plan. Execution will involve detailed content creation, scheduling, and constant evaluation of what works best. It's essential to adapt based on user interaction and platform changes**.**

**Debugging and Traceability exception Handling**

Error Tracking Tools: Implement error tracking tools like Sentry or Rollbar to monitor the application for any exceptions or errors that occur during the ad campaign.

Comprehensive Testing: Thoroughly test the ad campaign before launching it. Check for potential errors or issues that users might encounter.

Robust Exception Handling: Develop a structured exception handling process within your code to catch and manage different types of exceptions or errors that might arise during the campaign.

Error Logging: Set up a system to log errors comprehensively, including relevant information such as error messages, stack traces, and user actions that led to the issue.

Real-time Monitoring: Monitor the campaign in real-time once it's live. Set up alerts to notify you of any critical errors or issues that may need immediate attention.

**Quick Response Plan**: Have a plan in place to quickly respond and resolve any identified issues. This might involve pausing the campaign, rolling back changes, or deploying quick fixes.