

Experiment no. 1

Survey on Online Shopping Behaviour Using Google Forms

Objective:

To create and conduct an online survey using Google Forms.

Procedure:

1. Open Google Forms at <https://forms.google.com>.
2. Click on **Blank Form** to start a new survey.
3. Add a **title** and **description** for the form.
4. Create questions using various types: multiple choice, checkboxes, short answer, paragraph, and linear scale.
5. Customize the form theme (colors, fonts).
6. Adjust settings to:
 - Allow multiple responses
 - Show confirmation message after submission
7. Preview and test the form before sharing.
8. Share the form link with respondents through email, WhatsApp, or social media.
9. Collect and analyze responses through Google Forms' built-in summary tools or export to Google Sheets.

Sample Questions Used in the Form:

Q. No.	Question	Type
1	How often do you shop online?	Multiple Choice
2	Which platforms do you use most for online shopping?	Checkboxes
3	What type of products do you usually buy online?	Checkboxes
4	What factor influences your purchase the most?	Multiple Choice
5	How satisfied are you with your recent online purchase?	Linear Scale
6	Have you ever faced issues with online orders (e.g., late delivery)?	Multiple Choice
7	How do you usually pay for your online purchases?	Multiple Choice
8	Do you read product reviews before buying?	Multiple Choice
9	What is your average monthly spend on online shopping?	Short Answer
10	Any suggestions to improve your online shopping experience?	Paragraph

Result:

The survey was successfully created using Google Forms and shared with users. The responses gave insights into shopping habits, preferred platforms, and satisfaction levels of online shoppers. Google Forms is an effective tool for quick, real-time data collection.

Experiment – 2

Automating Email Notification using Zapier when Google Form is Submitted

Objective:

To automate the process of sending an email notification every time a new response is submitted to a Google Form using Zapier.

Procedure:

Step 1: Create a Google Form

- Go to Google Forms
- Add required fields (e.g., Name, Email, Message).
- Click “Responses” → Link it to a new Google Sheet.

Step 2: Setup Zapier Account

- Visit <https://zapier.com> and sign up/login.
- Click on “+ Create Zap”.

Step 3: Set the Trigger (Google Sheets)

- App: Google Sheets
- Event: New Spreadsheet Row
- Connect your Google Account and select the Sheet linked to your Google Form.

Step 4: Set the Action (Gmail)

- App: Gmail (or Email by Zapier)
- Event: Send Email
- Set up the email fields:
 - o To: Your email address
 - o Subject: New Form Submission
 - o Body: Include form fields (Name, Email, Message)

Step 5: Test & Turn On

- Zapier will test the workflow using existing data.
- Click “Turn on Zap”

Result:

The Zap successfully sends an email whenever the Google Form receives a new response.

EXPERIMENT - 3

Web Scraping using ParseHub (No-Code Tool)

Aim:

To extract specific data (e.g., course information) from a website using the no-code scraping tool ParseHub.

Procedure:

Steps: 1. Sign Up for ParseHub o Visit <https://www.parsehub.com> o Create a free account by signing up with your email or Google account

2. Install ParseHub o Download and install the ParseHub desktop app o Available for Windows, macOS, and Linux

3. Start a New Project

o Open the ParseHub app and click on “New Project”

o Enter the URL of the website you want to scrape (e.g.,
https://parsehub.com/sandbox/ph_showtimes)

4. Select Data to Scrape o After the page loads, click on elements you want to extract (e.g., movie name)

o ParseHub will highlight similar elements and allow group selection

5. Run the Project

o Click the “Run” button

o Choose either “Run locally” or “Run on server”

6. Export Data

o After scraping completes, click on “Get Data”

o Export the result as CSV, JSON, or access it via API Sample

Output (CSV Format):

movies_name

The Shawshank Redemption

Schindler's List

The Godfather

Experiment - 4

Working with the Designer Interface of Webflow

Aim:

To understand and explore the Webflow Designer interface and learn the basics of creating and styling elements visually without writing code.

Procedure:

- 1. Login and Start a New Project**
 - Go to <https://webflow.com>
 - Sign in or create a new account
 - Click “**New Site**” and choose a blank or template-based project
- 2. Explore the Interface**
 - Identify all major parts of the Designer UI
 - Hover over each panel to see tooltips
- 3. Add Elements**
 - Click on the **Add Panel (+)** on the left
 - Drag a **Section**, then a **Container**, and finally a **Heading** into the canvas
- 4. Style Elements**
 - Click the element you added
 - Use the **Style Panel (right sidebar)** to apply styling:
 - Change font size
 - Set background color
 - Add padding and margins
- 5. Use the Navigator**
 - Click on the **Navigator** to view the hierarchy of elements
 - Rename elements for better structure (e.g., Header, Hero Section)
- 6. Preview the Design**
 - Click on the **Preview Eye Icon** to view how the design looks in real-time
- 7. Save and Publish**
 - Click on **Publish** (top-right corner)
 - View the site using the provided Webflow subdomain link

Observations:

Element Added	Style Applied	Comments

Heading Font: 32px Applied via Style Panel

Section BG Color: #f2f2f2 Added margin-top

Button Hover effect Added simple interaction

Result:

Successfully explored and used the Webflow Designer interface to build and style a basic web layout visually without writing code.

Experiment - 5

Create a Responsive Webpage on Introduction to No-Code Applications Using Webflow

Objective: Design a responsive webpage that introduces **No-Code Applications**, covering their **definition, need, and scope**. The webpage should be clear, structured, and visually adaptive across devices.

Step-by-Step Instructions:

1. Access Webflow

- Visit <https://webflow.com>
- Login or create a new account
- Click **New Site** > Choose **Blank Template** > Name your project (NoCodeIntro)

2. Build Page Content

a. Header Section

- Add a **Section**
- Inside it, place a **Heading (H1)**: "*Introduction to No-Code Applications*"
- Style it prominently (font size, bold)

b. Definition Section

- Add a new **Section**
- Insert a **Heading (H2)**: "*What are No-Code Applications?*"
- Add a **Paragraph** explaining:

No-code applications are software solutions developed without writing traditional code, using visual interfaces and drag-and-drop tools instead.

c. Need for No-Code Applications

- Add another **Section**
- Heading (H2): "*Why Do We Need No-Code Applications?*"
- Add a **Bulleted List** describing:
 - Faster development time
 - Enables non-developers to build apps
 - Cost-effective solution for startups and businesses
 - Simplifies prototyping and iteration

d. Scope of No-Code Applications

- Add a new **Section**
- Heading (H2): "*Scope of No-Code Applications*"

- Use a **Paragraph** or **Numbered List** to explain:
 - Wide applicability in building websites, mobile apps, workflows
 - Growing demand in businesses to accelerate digital transformation
 - Integration with various APIs and services expanding functionality
 - Empowering citizen developers and innovators

3. Style Your Webpage

- Use consistent font styles and colors
- Add padding and margin for readability
- Optionally add background colors or images to separate sections visually
- Use Webflow's Style panel to make your page visually appealing

4. Make the Page Responsive

- Use Webflow's device preview tool to switch between Desktop, Tablet, Mobile Landscape, and Mobile Portrait views
- Adjust text size, spacing, and element layout for smaller screens
- Use Flexbox or Grid to rearrange content for narrow widths

Result: A responsive webpage is created.

Experiment - 6

Using Bubble build features like sign up forms, expense trackers, inboxes, shopping carts

Objective: To gain hands-on experience using the no-code platform Bubble.io to design and build the following features:

- Sign-up & Login Form
- Expense Tracker
- Messaging Inbox
- Shopping Cart

Part 1: User Sign-up & Login

Objective: Create a user registration and login system.

Tasks:

1. Create a new page or section called SignUp_Login.
2. Add:
 - Input fields: Email, Password
 - Buttons: "Sign Up", "Log In"
3. Use Bubble's built-in "Sign the user up" and "Log the user in" workflows.
4. Show a confirmation message or redirect to a "Dashboard" page after login.

Part 2: Expense Tracker

Objective: Build a basic expense tracking tool for logged-in users.

Tasks:

1. Create a Data Type: Expense with fields:
 - Amount (Number)
 - Category (Text)
 - Date (Date)
 - User (User)
2. Create a form to add a new expense.
3. Create a repeating group to list all expenses for the current user.
4. Add a total expense calculator using dynamic expressions.

Part 3: Inbox (Messaging System)

Objective: Build a basic private messaging feature.

Tasks:

1. Create a Data Type: Message with fields:

- Sender (User)
- Recipient (User)
- Message Text (Text)
- Timestamp (Date)

2. Create:

- Dropdown or search to select recipient
- Input field to type message
- Button to send message
- Repeating group to display messages between users

Part 4: Shopping Cart

Objective: Implement a simple shopping cart

system. Tasks:

1. Create Data Types:

- Product: Name (Text), Price (Number), Image (Image)
- CartItem: Product (Product), Quantity (Number), User (User)

2. Display products in a repeating group with an “Add to Cart” button.

3. Show the cart contents on a separate page.

4. Allow users to update quantity or remove items.

5. Calculate and display the total price.

Result:

Features like sign up forms, expense trackers, inboxes, shopping carts are built using Bubble.io.

Experiment - 7

Mobile Application Development

Objective: To build a simple mindfulness app using Glide where students can enter and track their participation activities along with their personal and academic details.

Procedure:

Step 1: Create Google Sheet

1. Open Google Sheets and create a new sheet named Mindfulness App Data.
2. Create a header row with the following columns:
 - o Name
 - o Course Name
 - o Semester
 - o Activity Participation
3. Leave the rows below empty (these will be filled through the app).

Step 2: Create a Glide App

1. Go to Glide Apps and sign in with your Google account.
2. Click on New Project and select Google Sheets as your data source.
3. Choose the Google Sheet you just created (Mindfulness App Data).
4. Glide will generate a basic app with your sheet's data.

Step 3: Customize the App Interface

1. In the Glide editor, navigate to the Layout section.
2. Change the layout to Form to enable new entries.
3. Add form components corresponding to the columns:
 - o Text entry for Name
 - o Text entry for Course Name
 - o Choice or dropdown for Semester (e.g., 1, 2, 3, 4, 5, 6)
 - o Text or choice entry for Activity Participation
4. Customize labels and hints to guide users.

Step 4: Set Privacy and Permissions

1. Under the Settings, enable Sign-in required if you want to restrict app access.
2. Otherwise, allow open access so anyone with the link can fill the form.

Step 5: Test the App

1. Use the Preview to open the app on your phone or desktop browser.
2. Enter sample data into the form and submit.
3. Verify that the data appears correctly in the Google Sheet.

Step 6: Publish and Share

1. Once satisfied, click Publish to generate a shareable app link or QR code.
2. Share the link with peers for testing.

Result:

Simple mindfulness app was developed using Glide where students can enter and track their participation activities along with their personal and academic details.

Experiment – 8

Build a Task Tracker App Using Glide

Objective: To design and build a simple **Task Tracker App** using **Glide**, enabling users to add, edit, and manage tasks efficiently through a Google Sheet-powered no-code interface.

Procedure:

Step 1: Set Up Google Sheet

1. Open Google Sheets and create a new spreadsheet.
2. Rename the sheet to "**Tasks**".
3. Create the following column headers:
 - o Task
 - o Description
 - o Due Date
 - o Status (e.g., Pending, In Progress, Completed)
 - o Priority (Optional)
4. Add a few sample tasks to populate your data.

Step 2: Create the App on Glide

1. Go to <https://www.glideapps.com>.
2. Click on "**Sign in**" and log in with your Google account.
3. Click "**New App**" → Choose "**Glide Apps**".
4. Select **Google Sheets** as your data source and choose the sheet created above.

Step 3: Configure Your Task Tracker App

1. Glide will automatically generate screens from your sheet.
2. Customize the **Tasks** tab:
 - o Display fields: Task, Description, Due Date, Status.
 - o Use a **Card** or **List** layout for a clean interface.
3. Enable **Add** functionality so users can add new tasks via a form.
4. Add **Edit** and **Delete** options for task items.

Step 4: Add Filtering & Sorting

1. Enable **filters** so users can view tasks by:
 - o Status (e.g., show only "Pending" tasks).
 - o Due Date (e.g., upcoming tasks).

2. Allow **sorting** by:

- Priority
- Due Date (ascending or descending)

Step 5: Test and Deploy

1. Preview your app on both mobile and desktop views.
2. Share the app using the link generated by Glide.
3. Try adding, editing, and deleting tasks.
4. Observe real-time syncing with your Google Sheet.

Result:

A simple Task Tracker App was developed using Glide.

Experiment – 9

Build an App using Thunkable

Objective:

To design and develop a mobile application using Thunkable for selling children's clothes targeted at ages 0–5 years with prices ranging from ₹500 to ₹2000.

Procedure:

Step 1: Setup Project

1. Go to Thunkable.com and log in.
2. Click on “**Create New Project**”.
3. Name your project: KidsClothingStore

Step 2: Design Home Screen

- Use **Label** to display: Welcome to Kids Clothing Store!
- Add a **Button** labeled Browse Clothes
- Navigate this button to the **Product Listing Screen**
- Step 3: Product Listing Screen
 - Add 4–5 **product cards** using **Rows/Columns** and **Images**, such as:
 - Baby T-shirt – ₹600
 - Cotton Romper – ₹800
 - Wool Sweater – ₹1500
 - Party Dress – ₹1800
 - Each product has:
 - Image
 - Name
 - Price
 - A "Buy Now" or "Add to Cart" button

Step 4: Cart/Order Screen (Optional)

- Show selected item summary
- Add a **Label**: Thank you for your order!
- Simulate order confirmation

Result:

A mobile application was developed using Thunkable for selling children's clothes targeted at ages 0–5 years with prices ranging from ₹500 to ₹2000.

Experiment – 10

Detect and Classify Face Masks using Google Teachable Machine

Objective:

To detect and classify images of people wearing face masks and those without masks using the Google Teachable Machine platform.

Procedure:

Step 1: Access Google Teachable Machine

1. Open a web browser and go to Google Teachable Machine.
2. Click on Get Started.

Step 2: Choose Project Type

1. Select Image Project to create an image classification model.

Step 3: Create Classes

1. Rename the default classes to Mask and No Mask.
2. Each class will represent images of people with and without face masks. Step 4: Upload or Capture Images
 1. For each class, upload images or capture images using your webcam.
 2. Upload at least 30-50 images per class for better model accuracy.
 - For the Mask class: Upload images of people wearing face masks.
 - For the No Mask class: Upload images of people without face masks. Step 5: Train the Model
 1. Once images are uploaded, click on the Train Model button.
 2. Wait for the training process to complete. This may take a few minutes depending on the number of images and system speed.

Step 6: Test the Model

1. After training, use the webcam or upload new test images to check if the model correctly classifies images as "Mask" or "No Mask."
2. Observe the confidence percentages displayed for each prediction. Step 7: Export or Use the Model
 1. If desired, export the model for integration into other applications.
 2. Google Teachable Machine allows you to download the model or get a URL for deployment.

Observations:

Input Image Description	Predicted Class	Confidence Score (%)
Person with mask	Mask	95
Person without mask	No Mask	92
Person with mask	Mask	89
Person without mask	No Mask	93

Result:

The model successfully detected and classified face masks with high accuracy, demonstrating the effectiveness of Google Teachable Machine in creating image classification models without coding.

Experiment – 11

Build an Image Classification Model Using Lobe.ai

Objective:

To create and train an image classification model using Lobe.ai without writing any code, and to test its ability to classify images into different categories.

Procedure:

Step 1: Download and Open Lobe.ai

1. Download the Lobe desktop app from lobe.ai and install it, or use the online version if available.
2. Launch the Lobe app.

Step 2: Create a New Project

1. Click on “Create New Project”.
2. Select Image Classification as the project type.

Step 3: Add Classes

1. Define the classes/categories you want the model to recognize.
 - o Example: "Cat", "Dog", "Bird".
2. You can add as many classes as needed depending on your dataset.

Step 4: Import Images

1. For each class, import a collection of images.
 - o You can drag and drop images or import from folders.

2. Make sure to upload a balanced number of images for each class (at least 30-50 images per class is recommended).
3. Ensure the images are clear and representative of each category.

Step 5: Train the Model

1. Once images are imported and labeled, Lobe.ai will automatically start training the model.
2. Wait for the training to complete — the status bar will show progress.

Step 6: Evaluate the Model

1. After training, test the model using new images by uploading them to see the classification results.
2. Observe the confidence scores and verify if the model correctly classifies the images.

Step 7: Export or Use the Model

1. You can export the trained model for integration with other software or applications.
2. Lobe supports exporting in various formats like TensorFlow, ONNX, or as a web API.

Observations:

Test Image Description	Predicted Class	Confidence Score (%)
Image of a cat	Cat	94
Image of a dog	Dog	91
Image of a bird	Bird	88
Image of a cat	Cat	96

Result:

The image classification model was successfully created and trained using Lobe.ai, demonstrating accurate classification of images across different categories.

Experiment – 12

Build a Conversational Chatbot using LandBot

Objective:

To design and deploy a conversational chatbot using LandBot, a no-code chatbot builder, which can interact with users and respond to their queries intelligently.

Procedure:

Step 1: Access LandBot Platform

1. Open your web browser and go to <https://landbot.io/>.
2. Sign up or log in to your LandBot account.

Step 2: Create a New Bot

1. Click on “Create New Bot”.
2. Choose “Start from Scratch” or select a template suitable for your project.
3. Name your chatbot (e.g., “Customer Support Bot”).

Step 3: Design Conversation Flow

1. Use the drag-and-drop interface to add conversation blocks.
2. Start with a Welcome Message to greet users.
3. Add Question Blocks to ask for user inputs such as name, email, or queries.
4. Use Condition Blocks to branch the conversation based on user responses.
5. Add Answer Blocks with pre-defined responses.
6. Incorporate Quick Replies or buttons to simplify user interaction.
7. Optionally, integrate APIs or external tools for advanced responses.

Step 4: Test the Chatbot

1. Use the Preview feature to simulate conversations.
2. Enter sample inputs and check if the chatbot responds correctly and follows the designed flow.
3. Modify the flow if necessary to improve user experience.

Step 5: Publish the Chatbot

1. Once satisfied, click on Publish.
2. Get the chatbot’s shareable link or embed code.
3. Deploy the chatbot on your website, social media, or messaging platforms.

Observations:

User Input	Bot Response	Comments
User enters name	Bot greets user by name	Personalized interaction
User asks about product price	Bot provides product price details	Relevant info given
User requests support	Bot provides support options or contact info	Proper routing to support
User inputs invalid data	Bot prompts for correct input	Handles errors gracefully

Result:

A fully functional conversational chatbot was built using LandBot that can interact with users, gather information, provide responses, and guide users through predefined flows without coding.

Experiment – 13

Create a Workflow in AirTable

Objective:

To create and automate a workflow using AirTable for managing data efficiently through tables, views, and automations.

Procedure:

Step 1: Access AirTable

1. Open your web browser and go to <https://airtable.com/>.
2. Sign up or log in to your AirTable account.

Step 2: Create a New Base

1. Click on “**Add a base**”.
2. Select “**Start from scratch**”.
3. Name the base (e.g., “Project Management”).

Step 3: Design Tables and Fields

1. Rename the default table (e.g., “Tasks”).
2. Add fields/columns such as:
 - Task Name (Single line text)
 - Assigned To (Single select or collaborator)
 - Due Date (Date)
 - Status (Single select: To Do, In Progress, Done)
 - Priority (Single select: High, Medium, Low)
3. Populate the table with sample data for tasks.

Step 4: Create Views

1. Create different views to organize data better:
 - Grid view for all tasks
 - Kanban view grouped by Status
 - Calendar view to see tasks by Due Date

Step 5: Set Up Automation Workflow

1. Click on the “**Automations**” tab.
2. Click “**Create an automation**”.
3. Choose a **Trigger** (e.g., When a record matches conditions).
 - Example: When Status changes to “Done”.
4. Set an **Action** (e.g., Send an email, Update a record, Send Slack message).
 - Example: Send an email notification to the project manager.

5. Test the automation by changing the status of a task to “Done” and observe the triggered action.
6. Enable the automation after successful testing.

Step 6: Collaborate and Share

1. Invite team members to the base.
2. Share views or the entire base as needed.

Observations:

Task Name	Assigned To	Due Date	Status	Automation Triggered	Action Result
Design Logo	Alice	2025-10-05	Done	Yes	Email sent to project manager
Develop Website	Bob	2025-10-10	In Progress	No	No action triggered
Write Content	Charlie	2025-10-07	Done	Yes	Email sent to project manager

Result:

A functional workflow was created in AirTable, automating notifications when task statuses are updated, thus improving project management efficiency.

Experiment – 14

Build Online Store Using Shopify

Objective:

To create and launch a fully functional online store using Shopify, enabling the sale of products through a professional e-commerce platform without coding.

Procedure:

Step 1: Create a Shopify Account

1. Open your browser and visit <https://www.shopify.com/>.
2. Click on Start free trial.
3. Enter your email address, password, and store name.
4. Fill in the required information about yourself and your business.

Step 2: Set Up Your Store

1. Access the Shopify dashboard.
2. Navigate to Products and click Add product.
3. Enter product details:
 - o Product name
 - o Description
 - o Upload product images
 - o Set price
 - o Set inventory quantity
4. Repeat for all products you want to sell.

Step 3: Customize Store Design

1. Go to Online Store > Themes.
2. Choose a free theme or browse paid themes.
3. Click Customize to edit the look and feel of your store.
4. Modify elements such as colors, fonts, header, footer, and homepage layout.

Step 4: Set Up Payments

1. Go to Settings > Payments.
2. Choose a payment gateway (Shopify Payments, PayPal, Stripe, etc.).
3. Follow instructions to configure payment methods.

Step 5: Configure Shipping Settings

1. Go to Settings > Shipping and delivery.
2. Set shipping zones, rates, and methods according to your delivery preferences.

Step 6: Test Your Store

1. Preview your online store.
2. Place a test order using the payment gateway's test mode.
3. Verify the entire purchase process, including payment, confirmation, and notifications.

Step 7: Launch Your Store

1. Choose a Shopify plan to publish your store (after the free trial).
2. Connect your domain or use Shopify's default URL.
3. Promote your store via social media or marketing campaigns.

Result:

An operational online store was successfully built using Shopify, complete with product listings, payment options, and shipping configurations.

Experiment – 15

Develop a Website Using Wix

Objective:

To design and develop a basic website using Wix website builder platform, applying features such as page creation, content insertion, and publishing.

Procedure:

Step 1: Access Wix Website

1. Open your preferred web browser.
2. Navigate to <https://www.wix.com>.
3. If you do not have an account, sign up for a free account using your email address or social media login.
4. Log in to your Wix account. Step 2: Choose a Template

Step 3: Customize Your Website

1. Edit Text
2. Add Pages
3. Add Elements
4. Change Background
5. Customize Header & Footer Step 4: Preview and Test Your Website Step 5: Publish Your Website

Result:

A fully functional website was successfully created using the Wix platform.