

## Introduction

This project involves creating an AppSheet application integrated with Google Sheets to automate WhatsApp messaging workflows for the team. The goal is to streamline the process of sending WhatsApp messages, track message statuses, and enable dynamic customization of message templates.

---

## Objective

The primary objective of this project is to create an efficient, user-friendly mobile and desktop app that allows team members to:

1. Dynamically create personalized WhatsApp messages.
  2. Generate and send WhatsApp message links.
  3. Track message statuses (e.g., Sent, Pending).
  4. Analyze the data for reporting and improvement.
- 

## What's Expected from the Team Member

1. Create an AppSheet app linked to Google Sheets for seamless integration.
  2. Set up views for data entry, tracking, and analytics.
  3. Add dynamic message customization using templates.
  4. Implement action buttons for triggering WhatsApp messages via mobile or desktop.
  5. Ensure the system is user-friendly and functional for both manual and automated workflows.
- 

## Step-by-Step Instructions

### Step 1: Prepare the Google Sheet

1. Create a new Google Sheet with the following columns:
  - **Column A:** Name – Recipient's name.
  - **Column B:** Phone Number – Recipient's phone number (with country code).
  - **Column C:** Template – Message template (e.g., "Hi [Name], how are you?").
  - **Column D:** Link – This will store the WhatsApp link.

- **Column E: Status** – Track whether the message is "Pending," "Sent," or "Failed."
- 2. Add sample data to test the functionality:
- Example:

Name	Phone Number	Template	Link	Status
John Smith	+911234567890	Hi [Name], how are you?		Pending

---

## Step 2: Configure Google Apps Script

1. Open the Google Sheet.
2. Go to **Extensions > Apps Script**.
3. Replace the default code with the following script:

```
function generateWhatsAppLinks() {
  const sheet = SpreadsheetApp.getActiveSpreadsheet().getActiveSheet();
  const data = sheet.getDataRange().getValues();
  const headerRow = data[0];
  const nameCol = headerRow.indexOf('Name') + 1;
  const phoneCol = headerRow.indexOf('Phone Number') + 1;
  const templateCol = headerRow.indexOf('Template') + 1;
  const linkCol = headerRow.indexOf('Link') + 1;
  const statusCol = headerRow.indexOf('Status') + 1;

  for (let i = 1; i < data.length; i++) {
    const name = data[i][nameCol - 1];
    const phoneNumber = data[i][phoneCol - 1];
    const template = data[i][templateCol - 1];

    if (name && phoneNumber && template) {
      const personalizedMessage = template.replace('[Name]', name);
      const encodedMessage = encodeURIComponent(personalizedMessage);
      const link = `https://web.whatsapp.com/send?phone=${phoneNumber}&text=${encodedMessage}`;

      sheet.getRange(i + 1, linkCol).setValue(link).setFontColor('blue').setFontLine('underline');

      // Set status to Pending if it's empty
      const statusCell = sheet.getRange(i + 1, statusCol);
      if (!statusCell.getValue()) {
        statusCell.setValue('Pending').setFontColor('red');
      }
    }
  }
}
```

```
}  
}  
}
```

Save the script and deploy it as a **Web App**:

- Go to **Deploy > New Deployment > Web App**.
- Set:
  - **Description:** WhatsApp Link Generator
  - **Execute as:** Me
  - **Who has access:** Anyone
- Click **Deploy** and copy the Web App URL.

### Step 3: Create the AppSheet App

1. **Sign in to AppSheet:**
  - Go to [AppSheet](#) and log in with your Google account.
2. **Connect to Google Sheets:**
  - Click **"Start with your data"** and select the prepared Google Sheet.
3. **Customize the App:**
  - Add a form view for entering **Name**, **Phone Number**, and **Template**.
  - Add a table view to display all rows from the sheet.
4. **Add a Virtual Column for Dynamic WhatsApp Links:**
  - Go to **Data > Columns > Virtual Columns**.
  - Add a virtual column:

**Formula:**

appsheet

CopyEdit

```
CONCATENATE(  
  "https://web.whatsapp.com/send?phone=",  
  [Phone Number],  
  "&text=",  
  ENCODEURL(SUBSTITUTE([Template], "[Name]", [Name]))  
)
```

■

5. **Add an Action Button for Sending Messages:**
  - Go to **Behavior > Actions**.
  - Create a new action:
    - **Do this:** External: go to a website.
    - **URL:** Use the virtual column with the WhatsApp link.
  - Attach the action to your table view.

---

#### Step 4: Enable Analytics in AppSheet

1. Create a new dashboard view in AppSheet for analytics.
2. Use formulas in Google Sheets to calculate:

Number of messages sent:

excel

CopyEdit

```
=COUNTIF(E2:E, "Sent")
```

○

Number of pending messages:

excel

CopyEdit

```
=COUNTIF(E2:E, "Pending")
```

○

---

#### Expected Deliverables

1. A fully functional AppSheet app integrated with Google Sheets.
2. Features:
  - Data entry and customization of templates.
  - Action button to open WhatsApp with pre-filled messages.
  - A status tracker for messages (e.g., "Pending," "Sent").
  - A dashboard view for analytics.
3. Documentation of the process and any encountered issues.