

# Google AppSheet Automation and Bots

## Version 1.1.1

**BMV SYSTEM INTEGRATION PRIVATE LIMITED**

Idea... Implementation... Innovation...

**:: CORPORATE HEAD OFFICE::**

A503, The First,

Behind The ITC Narmada Hotel & Keshavbaug Party Plot,

Off 132 ft Road, Vastrapur, Ahmedabad.

Gujarat- 380015

Phone: +91 (79) 40 30 53 02

Website: [www.systemintegration.in](http://www.systemintegration.in)

Mail: [info@systemintegration.in](mailto:info@systemintegration.in)



# Table of Content

1. [Introduction](#)
  - 1.1 Overview
  - 1.2 Purpose of Automation and Bots
2. [Key Components of Automation in AppSheet](#)
  - 2.1 Boat
  - 2.2 Event
  - 2.3 Process
  - 2.4 Task
3. [Creating a Bot](#)
  - 3.1 Create Event
  - 3.2 Create Task
  - 3.3 Configure Bot
  - 3.4 Test and Monitor
4. [Advanced features of the Task category](#)

## Introduction

### 1.1 Overview

AppSheet Automation and Bots offer powerful capabilities for automating workflows and streamlining business processes within the AppSheet platform. These features enable users to define automated actions, triggers, and conditions that respond to events and data changes, as well as create intelligent bots that can perform tasks autonomously based on predefined rules and logic.

Key aspects of AppSheet Automation and Bots include:

- **Event-driven Automation:** Users can define automation rules that trigger actions based on events such as data changes, form submissions, or scheduled intervals. This event-driven approach allows for proactive and responsive automation that adapts to dynamic data conditions.
- **Bot-based Workflow Automation:** Bots in AppSheet are intelligent agents capable of executing predefined tasks and workflows autonomously. Bots can be configured to monitor data, perform actions, send notifications, and even interact with external systems using APIs, enhancing efficiency and productivity.
- **Integration with AppSheet Apps:** Automation and bots seamlessly integrate with existing AppSheet apps, allowing users to enhance app functionality and automate repetitive tasks without the need for complex coding or external integrations. This tight integration simplifies the development and deployment of automated solutions within the familiar AppSheet environment.

### 1.2 Purpose of Automation and Bots

The primary purposes of AppSheet Automation and Bots are to:

- **Improve Efficiency:** By automating repetitive and manual tasks, automation and bots reduce the time and effort required to perform routine operations, allowing users to focus on higher-value activities.
- **Enhance Productivity:** Automation and bots streamline workflows, minimize delays, and ensure timely execution of tasks, thereby increasing overall productivity and throughput.

- **Increase Accuracy:** Automated processes and bots follow predefined rules and logic consistently, reducing the likelihood of errors and improving data accuracy and integrity.
- **Enable Scalability:** With automation and bots, users can scale their operations efficiently by automating processes that would otherwise be resource-intensive or time-consuming to perform manually.
- **Facilitate Decision-making:** By providing real-time insights, notifications, and alerts, automation and bots empower users to make informed decisions quickly, based on up-to-date information and data trends.

## Key Components of Automation in AppSheet

### 1. Bot

- **Description:** A bot defines the automation you want AppSheet to run. It acts as the central component that triggers actions and processes based on predefined conditions and events.
- **Functionality :**
  - Interprets user inputs or system events to initiate automation.
  - Executes predefined processes or tasks in response to triggers.
  - Enhances user interaction through natural language commands or event-based triggers.
- **Use Cases:**
  - Responding to user queries or commands.
  - Initiating workflows based on specific events or conditions.
  - Automating tasks such as data updates, notifications, or approvals.

### 2. Event

- **Description:** An event identifies the data change, schedule, or Chat app interaction that triggers the bot. Events act as triggers that initiate automation processes within the app.
- **Functionality :**
  - Monitors data changes in app tables.

- Executes scheduled tasks or actions at specified times or intervals.
- Listens for user interactions in Chat apps to trigger automation.
- **Use Cases:**
  - Triggering automation when a new record is added or updated.
  - Scheduling tasks for daily, weekly, or monthly execution.
  - Initiating automation based on user inputs in Chat apps.

### 3. Process

- **Description:** A process describes the set of steps performed when the bot is triggered. It defines the sequence of actions, tasks, conditions, and logic that form the automation workflow.
- **Functionality :**
  - Executes a series of steps in response to an event or user command.
  - Branches on conditions to determine the flow of automation.
  - Waits for specific conditions or time periods before proceeding.
  -
- **Use Cases:**
  - Creating multi-step workflows for complex processes.
  - Branching automation based on data conditions or user inputs.
  - Implementing time-based or conditional logic in automation processes.

### 4. Task

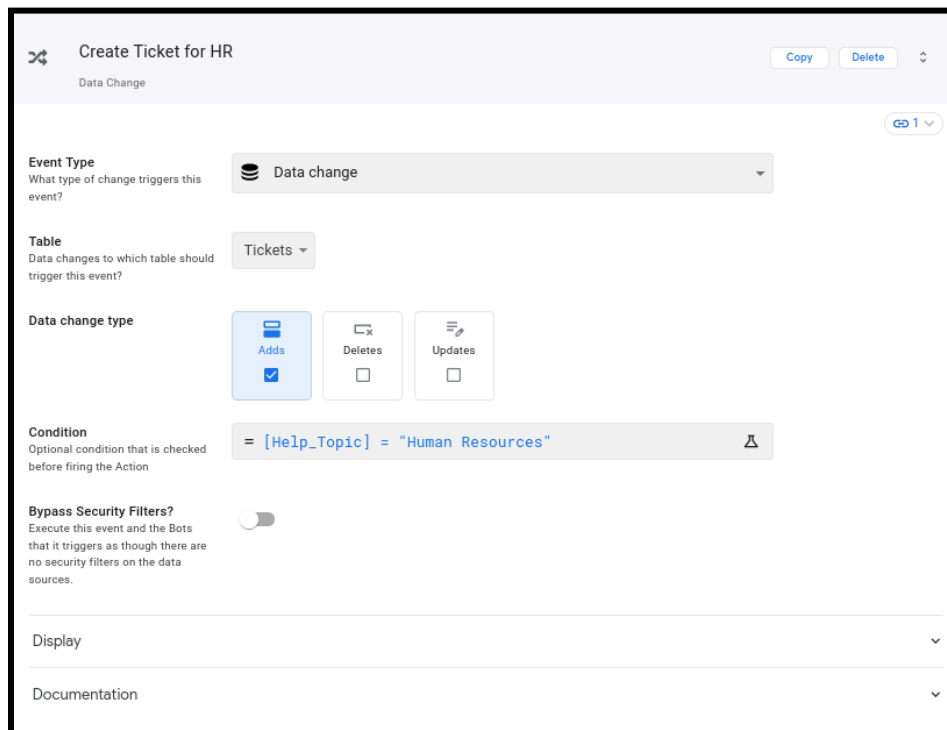
- **Description:** A task defines a task-based step in a process, such as sending an email, updating data, or triggering a notification. Tasks are the individual actions performed as part of the automation workflow.
- **Functionality :**
  - Executes specific actions or operations within the app.
  - Sends notifications, emails, or alerts to users or stakeholders.
  - Updates creates, or deletes data based on predefined rules.
- **Use Cases:**
  - Sending email notifications for new leads or customer inquiries.
  - Updating inventory levels based on sales transactions.
  - Triggering alerts for overdue tasks or pending approvals.

## Creating a Bot

### 3.1 Create Event

#### Steps to Create an Event:

1. **Navigate to Events Section:** In the AppSheet editor, go to the "Automation" tab and select "Events."
2. **Add New Event:** Click on the "Add Event" button to create a new event.
3. **Define Trigger:** Choose the trigger for the event, such as "Data Change," "Form Saved," or "Schedule."
4. **Configure Event Details:** Specify additional details for the event, including the table, column, and conditions that will trigger the event.
5. **Save Event:** Once configured, save the event to proceed.



Create Ticket for HR

Data Change

Copy Delete

Event Type  
What type of change triggers this event?

Data change

Table  
Data changes to which table should trigger this event?

Tickets

Data change type

Adds Deletes Updates

Condition  
Optional condition that is checked before firing the Action

= [Help\_Topic] = "Human Resources"

Bypass Security Filters?  
Execute this event and the Bots that it triggers as though there are no security filters on the data sources.

Display

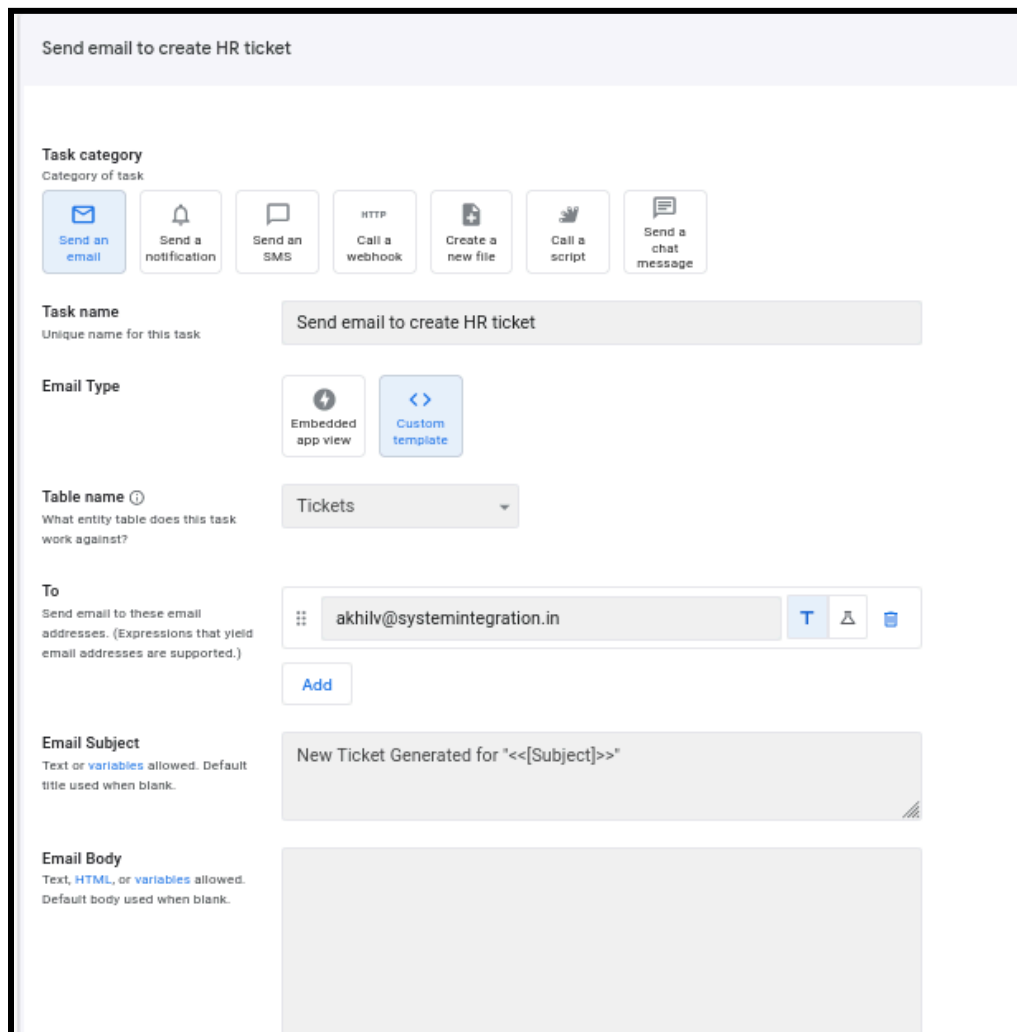
Documentation

6.

### 3.2 Create Task

#### Steps to Create a Task:

1. **Navigate to Tasks Section:** In the AppSheet editor, go to the "Automation" tab and select "Tasks."
2. **Add New Task:** Click on the "Add Task" button to create a new task.
3. **Define Task Actions:** Specify the actions that the task should perform when triggered. This can include data manipulation, sending notifications, or invoking workflows.
4. **Configure Task Details:** Provide details for the task, such as a name, description, and priority level.
5. **Save Task:** Once configured, save the task to proceed.



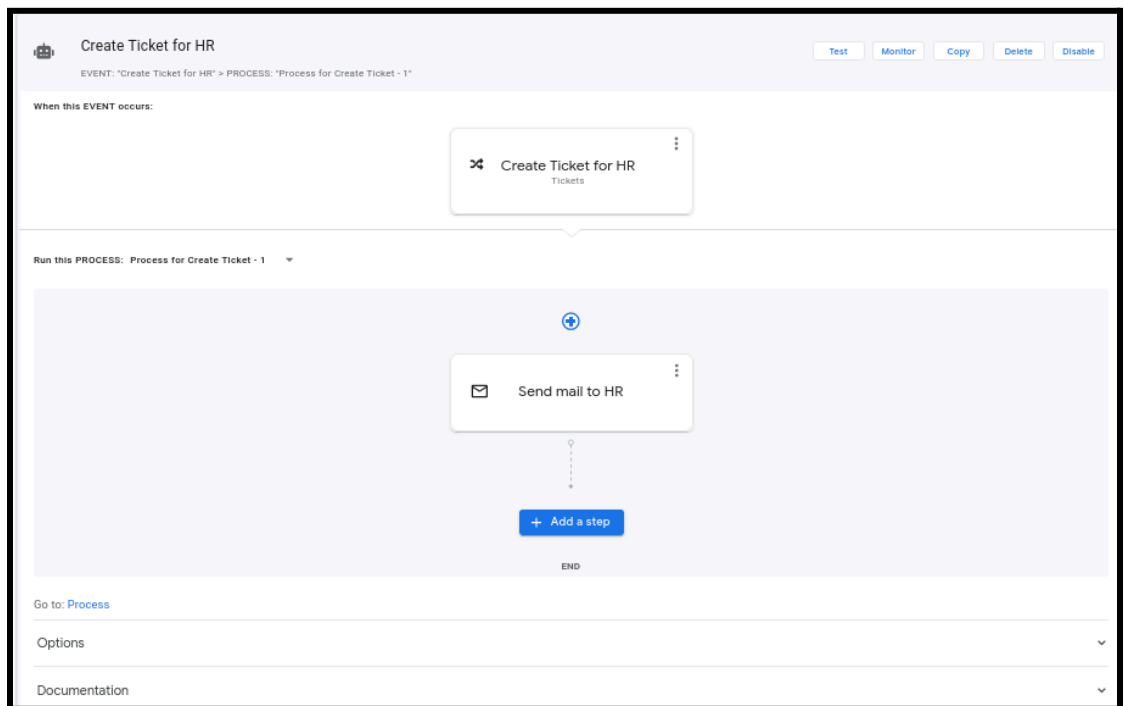
The screenshot shows the configuration interface for a task titled "Send email to create HR ticket". The interface is divided into several sections:

- Task category:** A row of icons for different task types: "Send an email" (selected), "Send a notification", "Send an SMS", "HTTP Call a webhook", "Create a new file", "Call a script", and "Send a chat message".
- Task name:** A text field containing "Send email to create HR ticket".
- Email Type:** Two buttons: "Embedded app view" and "Custom template" (selected).
- Table name:** A dropdown menu showing "Tickets".
- To:** A text field containing "akhilv@systemintegration.in" with an "Add" button below it.
- Email Subject:** A text field containing "New Ticket Generated for "<[Subject]>>".
- Email Body:** A large text area for the email body content.

### 3.3 Configure Bot

#### Steps to Configure Bot:

1. **Navigate to Bots Section:** In the AppSheet editor, go to the "Automation" tab and select "Bots."
2. **Add New Bot:** Click on the "Add Bot" button to create a new bot.
3. **Define Bot Triggers:** Specify the triggers that will activate the bot. This can include events created earlier, such as data changes or form submissions.
4. **Assign Tasks to Bot:** Select the tasks that the bot should execute when triggered. Choose from the tasks created earlier or create new tasks specifically for the bot.
5. **Configure Bot Settings:** Provide details for the bot, such as a name, description, and schedule for execution.
6. **Save Bot:** Once configured, save the bot to activate it.



7.

### 3.4 Test and Monitor

#### Test and Monitor the Bot:

1. **Test Bot:** After creating and configuring the bot, test its functionality to ensure that it performs as expected.
2. **Monitor Execution:** Monitor the bot's execution and performance over time to identify any issues or areas for improvement.

## Advanced Features of Task Category



## 1. Send an Email

- **Description:** This action allows the bot to send an email to one or more recipients.
- **Usage:** Useful for notifying users or stakeholders about important updates, reminders, or notifications.
- **Configuration:** Specify the email recipients, subject, body content, and any attachments as needed.

## 2. Send a SMS

- **Description:** This action enables the bot to send a text message (SMS) to a specified phone number.
- **Usage:** Ideal for sending quick alerts, reminders, or notifications to users who may not have access to email.
- **Configuration:** Specify the recipient's phone number and the message content to be sent via SMS.

## 3. Send a Notification

- **Description:** This action triggers a push notification to be sent to users who have the app installed on their mobile devices.
- **Usage:** Useful for instantly notifying users about important events, updates, or tasks.
- **Configuration:** Define the notification message and any additional parameters, such as the target audience or delivery schedule.

## 4. Send a Chat Message

- **Description:** This action allows the bot to send a message to a specified chat platform or service, such as Slack or Microsoft Teams.
- **Usage:** Enables seamless communication and collaboration by providing real-time updates and alerts in chat channels.
- **Configuration:** Specify the chat platform, channel, and message content to be sent.

## 5. Call a Script

- **Description:** This action executes a custom script or function defined in a scripting language, such as Google Apps Script or JavaScript.

- **Usage:** Offers flexibility to perform complex operations or integrations with external systems that are not supported by built-in actions.
- **Configuration:** Provide the script code or reference the script file to be executed, along with any input parameters required.

## 6. Create a New File

- **Description:** This action generates a new file in a specified format, such as a document, spreadsheet, or PDF.
- **Usage:** Useful for generating reports, documents, or other artifacts based on data within the app.
- **Configuration:** Specify the file format, template, and content to be included in the new file.

## 7. Call a Webhook

- **Description:** This action sends an HTTP request to a specified endpoint or URL, typically as part of an API integration.
- **Usage:** Facilitates integration with external systems, services, or APIs to exchange data or trigger actions.
- **Configuration:** Provide the target URL, request method, headers, and payload data as needed for the webhook call.