Task-2

We can use Google Apps Script as a "digital courier" to connect your Google Sheet with the outside world, automating the communication and absorbing the chaos of the clinic into this structured system.

Below is the detailed design of the automation logic, with a focus on protecting the doctor's time and ensuring billing/care discipline.

1. Step-by-Step Automation Logic

The automation has four strict steps in operation, so that not a single message is sent without prior context or approval.

Phase	Technical Action	Purpose
Read	Access the Care_Control sheet and pull all row data into the script's memory.	To see every patient interaction in one view.
Filter	Check each row for: Status = "Pending", Message Text is not empty, and Doctor Approval is "Not Required" or "Approved" .	To ensure only routine or doctor-vetted messages are sent.
Dispatch	Package the phone number and message text into a "JSON" packet and send it to the WhatsApp API.	To move communication off the doctor's personal phone.
Update	Change the cell in the Status column from Pending to Send.	To prevent the system from sending the same message twice.

2. Detailed Implementation (Apps Script)

Here is the logic written out, this can be directly copied into your Extensions > Apps Script editor.

JavaScript

```javascript
function processPendingMessages() {
  // 1. ACCESS THE SHEET
  const ss = SpreadsheetApp.getActiveSpreadsheet();
  const sheet = ss.getSheetByName("Care_Control"); // [cite: 114]
  const data = sheet.getDataRange().getValues(); // Read pending rows

  // 2. DEFINE THE API (Example using a WhatsApp API provider)
  const API_URL = "https://api.whatsapp-provider.com/v1/messages";
  const API_TOKEN = "YOUR_SECRET_TOKEN";

  // 3. LOOP THROUGH THE DATA (Skip the Header Row)
  for (let i = 1; i < data.length; i++) {
    let row = data[i];

    // Mapping our specific columns[cite: 115]:
    // Col A=0 (Patient), B=1 (Phone), E=4 (Message), F=5 (Approval), G=6 (Status)
    let phone = row[1];
    let messageText = row[4];
    let approvalStatus = row[5];
    let currentStatus = row[6];

    // 4. THE LOGIC GATE
    // Only send if Pending AND (No approval needed OR Doctor already Approved)
    if (currentStatus === "Pending" && messageText !== "") {
      if (approvalStatus === "Not Required" || currentStatus === "Approved") {

        // 5. SEND WHATSAPP MESSAGE VIA API
        let payload = {
          "to": phone,
          "body": messageText
        };

        let options = {
          "method": "post",
          "contentType": "application/json",
          "headers": { "Authorization": "Bearer " + API_TOKEN },
          "payload": JSON.stringify(payload),
          "muteHttpExceptions": true
        };

        try {
          UrlFetchApp.fetch(API_URL, options);

          // 6. UPDATE STATUS TO "SENT"
          // Column G is index 7 (i+1 refers to the sheet row)
          sheet.getRange(i + 1, 7).setValue("Sent");

        } catch (e) {
          sheet.getRange(i + 1, 7).setValue("Error");
          Logger.log("Failed for " + phone + ": " + e.toString());
        }
      }
    }
  }
}
```

3. Why This System Works for the Clinic

- **Absorbs Chaos:** The doctor does not have to toggle between HMS and WhatsApp. Every conversation is queued in the sheet.
- **10 Minute Review:** The doctor will only work on the rows where Approval=Required.
- Need for Human Judgment. Automated follow-ups (Type A, B, C) will be performed by the script itself, while custom queries (Type D, E) will have to wait for the doctor.

- **Status Transitions:** The transition from Pending ↣ Sent ↣ Closed allows the staff to make a Daily Closing Check in 5 minutes to ensure no critical matter has been overlooked.