Hii I want to make an automation for an insurance company. That 1). Will have an AI-Agent who will check the received email's and answer them. If he can not answer them directly he will then separate them and send them to the responsible employee in the right department. (We will use Sendgrid for testing purposes). 2. I want to have an AI-Agent who will be responsible for extracting Information of the documents (with OCR technology) he will receive the document per mail and then look trough them and extract all the Informations he will be needing to see what department this document belongs to and send it to the responsible employee. It will have 3 different cases for the document. a). a accident report document. b). a damage report document. c). documents who could be pictures or other documents who are not in the list if thats the case the AI\_Agent will extract the information and send the description with the document attached to the employee who will look trough it and decide where it goes. (It would be great if the AI-Agent would ask the employee where this kind of document has gone so he can memorize it and learn.) 3. I want an AI-Agent who will be responsible for the appointments who the employee will have with the clients. That would mean that this Agent is responsible that every employee will get an email in the morning with every appointment that the employee will have this day. The structure will be very important because i want that the employee sees directly in his report: a). which time the appointment is. b). what kind of appointment it is. c). the clientID with the informations about the client who are needed for this appointment. d). the documents or informations that will be needed for this appointment. e). a short pitch for the employee so that the employee knows whats the most important. 4. We will be needing also a AI-Agent who will make daily and weekly reports. This Agent will summarize the daily and weekly actions who were made from all the Agents and he will make reports on a daily basis and reports on a weekly basis. I want the daily reports to be summarized in an email and the weekly report summarized in a table for better structure and readability. ----- I want from you also that you will be asking me 5 questions for more Informations so that we will not forget or miss an important step. So that we can make a perfect prompt.

**ChatGPT said:**



This automation for an insurance company sounds like a comprehensive system involving several AI agents to streamline email handling, document processing, appointment management, and reporting. To create an optimal solution using Sendgrid, OCR, and AI technology, I will also need additional details to ensure the prompt covers all aspects. Here are five important questions to fine-tune the process:

1. **Email classification criteria**: How should the AI-Agent determine which emails can be answered automatically versus those needing escalation to a specific department? Are there any specific rules, keywords, or pre-set criteria the AI should use for classification and response?
2. **OCR document handling and AI learning**: For the OCR processing agent, what specific information needs to be extracted from accident and damage reports? Also, could you clarify the method for capturing new document types and the learning process for future handling—should the AI store this feedback in a database?
3. **Appointment structure details**: What type of scheduling system do you use to manage employee-client appointments? Are there external calendars (like Google Calendar or an internal tool) that the AI should integrate with to fetch appointments, or will this data be provided via email or another system?
4. **Employee feedback mechanism**: When an AI-Agent routes a document or asks for input from employees on document types or task categorization, should this feedback be provided via email, an internal platform, or another interface?
5. **Reporting customization**: For the daily and weekly reports, what specific metrics or data points should the AI-Agent focus on? Do you want any special formatting (e.g., KPIs, response times, or department-based summaries) to be included in the weekly table?

Once I have answers to these questions, I can generate an accurate, detailed prompt that ensures all aspects of your automation flow smoothly.

**You said:**

SchadensformularV.pdf

PDF

1. Email classification criteria: The AI-Agent will be using a knowledge base where he has all the informations about the company. (from simple informations about the company to the price questions about what the company "sells".) 2. OCR document handling and AI learning: I will give you a document so that you will see and understand to what i refer. I have made an explanation so you see what are the different field: " Das Formular trägt den Titel "Schadenmeldung UVG" und enthält am oberen Rand vier Checkboxen für die Kategorien "Unfall", "Zahnschaden", "Berufskrankheit" und "Rückfall". Es ist in 14 Hauptabschnitte unterteilt, die durchnummeriert sind: Arbeitgeber: Felder für Name und Adresse mit Postleitzahl, Tel.-Nr., Kunden-Nr., Betriebsteil Feld für den üblichen Arbeitsplatz des/der Verletzten (Betriebszweig) Verletzte/r: Felder für Name und Vorname, Geburtsdatum, AHV-Nummer Felder für Strasse, Tel.-Nr., Staatsangehörigkeit Felder für PLZ, Wohnort, Zivilstand Anstellung: Felder für Datum der Anstellung, Ausgeübter Beruf Checkboxen für Stellung: Höheres Kader, Mittleres Kader, Angestellte/r / Arbeiter/in, Lehrling, Praktikant/in Checkboxen für Verhältnis: unbefristeter Arbeitsvertrag, befristeter Arbeitsvertrag, Arbeitsverhältnis gekündigt Felder für Arbeitszeit, Beschäftigungsgrad, betriebsübliche Vollarbeitszeit Checkboxen für Arbeitseinsatz: unregelmässig, Kurzarbeit Schadendatum: Felder für Tag, Monat, Jahr, Zeit (Stunden, Minuten) Unfallort: Feld für Ort und Stelle Sachverhalt: Großes Textfeld für Unfallbeschreibung oder Verdacht auf Berufskrankheit Feld für beteiligte Person(en) Checkboxen für Polizeirapport: Ja, Nein, Unbekannt Berufsunfall: Feld für beteiligte Gegenstände Nichtberufsunfall: Felder für letzten Arbeitstag vor dem Unfall und Grund der Absenz Verletzung: Checkboxen für links, rechts, unbestimmt Felder für betroffenes Körperteil und Art der Schädigung Arbeitsunfähigkeit: Checkboxen und Felder für Details zur Arbeitsunfähigkeit Arztadressen: Felder für erstbehandelnden und nachbehandelnden Arzt bzw. Spital/Klinik Lohn: Detaillierte Felder für verschiedene Lohnkomponenten Sonderfälle: Checkboxen und Felder für spezielle Situationen Andere Sozialversicherungsleistungen: Textfeld für Angaben zu anderen Versicherungsansprüchen Am Ende des Formulars gibt es Felder für Ort, Datum, Stempel und Unterschrift." As for the learning from new document types or images the Agent should store it in a database for further references so at the end of the week he will make a ail to an employee where he will ask questions about the new and unknown informations so that he can learn them for the future. 3. Appointment structure details: We will be a). using airtable where all the appointments made trough our chatbot will be stored. b). some appointments will be also received via email but the Agent who is responsible for the email classification will inform the Appointment Agent and store the new appointment also to airtable for now. 4. Employee feedback mechanism: The feedback should be made vie email and then the received data or informations should then be stored in the Agent's database or knowledge base for further references. 5. Reporting customization: The Agent should summaries what every Agent has done for the day and for the week separatly. What task every Agent has done so far and where the Agent needed some more clarifications if he couldn't do the task or asked for more informations from an employee via email.



Based on the information you've provided, here's how we can approach the automation and structure the AI agents for the insurance company. Each agent will be designed with specific responsibilities, and we'll use SendGrid, OCR, and Airtable to implement key functionalities. I'll generate a detailed and optimized prompt for each of your requested features. Before I do, I'll summarize everything and ensure we are aligned.

**Breakdown of Automation:**

1. **Email Classification Agent**:
   * **Purpose**: To automatically classify emails using a knowledge base.
   * **Functionality**: The agent reads incoming emails, checks the knowledge base (for product info, company policies, etc.), and responds if it has the necessary information. If not, it forwards the email to the appropriate department, based on pre-defined keywords or employee tags.
   * **Testing**: SendGrid for email management.
2. **Document OCR Agent**:
   * **Purpose**: Extracts information from different document types and routes them accordingly.
   * **Functionality**:
     + **Accident reports**: Extract fields like employer info, injury details, etc.
     + **Damage reports**: Similar process for damage-related fields.
     + **Other documents (pictures or unidentified docs)**: The agent identifies the type, requests feedback via email if unsure, and stores the learning process in a database.
   * **Learning**: AI will learn from employee feedback on new/unrecognized documents. At the end of the week, it sends an email summarizing the unknown types and requests manual input to update its knowledge base.
3. **Appointment Manager Agent**:
   * **Purpose**: Handles daily scheduling by sending an email with all appointments for each employee.
   * **Functionality**:
     + Extracts data from **Airtable** (where chatbot appointments are stored).
     + Incorporates email appointment details (via the email classification agent).
     + Sends a structured email each morning, including the time, client details, appointment type, necessary documents, and a short pitch for the employee.
4. **Daily and Weekly Reporting Agent**:
   * **Purpose**: Summarizes actions of all agents and generates daily/weekly reports.
   * **Functionality**:
     + **Daily Reports**: Summarizes in an email what each agent did and where clarifications were needed.
     + **Weekly Reports**: A more structured summary presented in table format, breaking down tasks by each agent and highlighting any unresolved tasks or feedback requested.

**5 Key Questions Resolved:**

1. **Email classification**: Uses a company-specific knowledge base to answer or escalate queries.
2. **OCR document handling**: The AI processes accident and damage reports (based on the form you provided​) and learns from feedback on unidentified documents.
3. **Appointments**: Manages data from **Airtable** and email, sending structured daily reports.
4. **Feedback mechanism**: Feedback from employees is handled via email and stored in the agent's knowledge base for future learning.
5. **Reports**: Includes task summaries per agent, with focus on task completion and issues requiring further employee input.

**Suggested Prompts:**

**1. Email Classification Agent (SendGrid + Knowledge Base):**

aider

Copy code

@agent

Agent name: Email\_Classifier\_Agent

Task: Handle incoming emails via SendGrid.

- Use the company knowledge base for answering common questions (e.g., product info, pricing, policies).

- If an email cannot be answered, route it to the appropriate department based on keywords (e.g., "claims," "support").

- Store responses and employee feedback for future reference in a database.

Integration:

- Platform: SendGrid

- Database: [Your database solution]

Testing: Use SendGrid API for receiving emails.

**2. Document OCR Agent (Document Extraction + Learning):**

aider

Copy code

@agent

Agent name: Document\_OCR\_Agent

Task: Extract information from documents using OCR technology and route them.

- For "accident reports," extract fields like employer, injured person details, accident description, and submit to the relevant department.

- For "damage reports," extract similar fields and send it to claims.

- For unrecognized documents (e.g., images), request feedback via email and store the feedback for future learning.

Learning: Store feedback from employees in a database for future use.

Integration:

- OCR tool: [Your OCR solution]

- Database: [Your database solution]

**3. Appointment Manager Agent (Airtable + Email):**

aider

Copy code

@agent

Agent name: Appointment\_Manager\_Agent

Task: Manage and send daily appointment summaries.

- Extract appointment data from Airtable (from chatbot).

- Add appointments received via email.

- Send a structured daily email to each employee with:

1. Appointment time.

2. Type of appointment.

3. ClientID and related client information.

4. Necessary documents.

5. A short pitch to brief the employee on the key points.

Integration:

- Airtable API

- SendGrid API for sending email summaries.

**4. Daily and Weekly Reporting Agent:**

aider

Copy code

@agent

Agent name: Report\_Generator\_Agent

Task: Generate daily and weekly reports for employee actions.

- Daily reports: Summarize tasks completed by each agent, unresolved tasks, and requests for feedback.

- Weekly reports: Create a table format summarizing all agent actions, with additional emphasis on any feedback requests.

Integration:

- Database: [Your database solution]

- SendGrid API for emailing daily summaries and weekly tables.

**Next Steps:**

Once you approve or modify these prompts, I can assist with implementation details (e.g., specific APIs, error handling). Would you like to modify any aspect of this plan before we proceed?

**You said:**

could we make more detailed prompts? so we can ensure that aider will have all the informations that he needs to make the perfect structure for the Agents.

Absolutely! I can expand the prompts with more detailed instructions for Aider to ensure the agents are structured perfectly. I'll break down the key functionalities, integrations, and learning processes clearly, providing enough depth for each agent. Let’s go step by step for each agent and include all necessary details, dependencies, and structured logic that Aider will need to create an optimal solution.

**1. Email Classification Agent (SendGrid + Knowledge Base)**

This agent will be responsible for parsing emails, referencing a company knowledge base to answer common queries, and forwarding unresolved cases to the right department.

aider

Copy code

@agent

Agent name: Email\_Classification\_Agent

Task: Manage incoming emails and automatically classify or respond using a company knowledge base.

- \*\*Receive Emails\*\*:

- Connect to \*\*SendGrid API\*\* to monitor and receive incoming emails.

- Parse email content, including subject lines, body text, and attachments (if any).

- \*\*Knowledge Base Integration\*\*:

- Load company knowledge base: [Link or Database for Knowledge Base].

- Use natural language processing (NLP) to understand customer queries.

- Search the knowledge base for relevant answers to common queries, including:

- \*\*Company information\*\*: services, contact details.

- \*\*Product information\*\*: pricing, policies, coverage types.

- \*\*FAQs\*\*: most frequently asked customer questions.

- If a match is found, generate a response email with the corresponding information.

- \*\*Email Classification\*\*:

- If the email cannot be resolved via the knowledge base, use a keyword-based classification model to route the email.

- Keywords/Tags (e.g., "claims," "billing," "support," "policy update," "sales inquiry"):

- \*\*Claims Department\*\*: Keywords like “claim,” “damage,” “accident.”

- \*\*Billing\*\*: Keywords like “payment,” “invoice,” “billing.”

- \*\*Support\*\*: General support queries not related to billing or claims.

- \*\*Sales/Marketing\*\*: Keywords like “new policy,” “product inquiry,” “quote.”

- Route unresolved emails directly to the corresponding department by attaching tags (department\_name) and forwarding the original email.

- \*\*Forward to Employee\*\*:

- Identify the right department email address using employee contact details from a \*\*staff directory database\*\*.

- Forward the email to the appropriate employee/department, with the original email attached.

- \*\*Logging & Feedback\*\*:

- Log all email interactions in the agent’s database, storing classified emails, answers given, and responses forwarded.

- If employees respond with corrections or clarifications, the agent updates the knowledge base for future reference.

- \*\*Tools/Integrations\*\*:

- \*\*SendGrid API\*\*: For receiving and sending emails.

- \*\*Knowledge Base Storage\*\*: [Knowledge Base URL or Database Path].

- \*\*Staff Directory Database\*\*: For routing emails to the right departments.

- \*\*NLP Library\*\*: To analyze email text and match it to the knowledge base.

Testing:

- Simulate incoming emails with different subjects to test if the agent classifies and responds appropriately.

- Monitor the log database for classification accuracy and ensure correct routing.

Error Handling:

- Log unclassified or ambiguous emails in a separate error report to request manual review by an employee.

- Send an error notification if knowledge base access fails or a response cannot be generated.

**2. Document OCR Agent (OCR Document Extraction + Learning)**

This agent will receive documents via email, extract specific fields using OCR, classify them by type, and route them to the appropriate department. It will also learn from employee feedback to improve classification.

aider

Copy code

@agent

Agent name: Document\_OCR\_Agent

Task: Handle incoming documents (accident, damage reports, and other files) using OCR, extract relevant fields, and route them to the appropriate department.

- \*\*Receive Documents\*\*:

- Monitor incoming emails using \*\*SendGrid API\*\*, focusing on attachments (PDF, images).

- Parse email content and attachments.

- \*\*OCR Document Processing\*\*:

- Use \*\*OCR technology\*\* to extract text from attached documents.

- Recognize and extract key fields from known document types (based on the form you shared):

1. \*\*Accident Report\*\* (e.g., "Schadenmeldung UVG" form):

- Extract employer info: Name, Address, Postcode, Phone Number, Customer Number.

- Injured Person details: Name, Birthdate, AHV-Number, Address, Nationality.

- Injury specifics: Date, time, location, injury description, body part affected.

- Police report status: Yes/No/Unknown.

- Employment details: Job title, employment type (permanent/temp), working hours.

2. \*\*Damage Report\*\*:

- Similar to accident reports, but focus on object-related fields, like property damage or vehicle involvement.

3. \*\*Other Document Types\*\*:

- Identify unknown document types (such as pictures or reports not matching predefined forms).

- Extract key text elements (e.g., title, major headings, or description).

- \*\*Document Classification\*\*:

- Based on extracted fields, classify the document as:

1. \*\*Accident Report\*\*.

2. \*\*Damage Report\*\*.

3. \*\*Unrecognized Document Type\*\*.

- \*\*Learning from Employee Feedback\*\*:

- For unrecognized documents, email the employee with the extracted information and a request for classification.

- Store the employee’s feedback in the knowledge base for future automatic classification.

- Send a \*\*weekly summary\*\* of unclassified documents to an employee for review, asking for feedback on how to handle them in the future.

- \*\*Routing Documents\*\*:

- Route documents based on classification:

1. \*\*Accident Reports\*\* → Claims Department.

2. \*\*Damage Reports\*\* → Risk/Claims Department.

3. \*\*Other Documents\*\* → Admin or designated employee for manual review.

- \*\*Tools/Integrations\*\*:

- \*\*OCR Tool\*\*: [Your OCR solution, such as Tesseract or an API like Google Cloud Vision].

- \*\*SendGrid API\*\*: For receiving documents via email.

- \*\*Database\*\*: Store feedback and learnings from unrecognized document types.

- \*\*Feedback Mechanism\*\*: Email employees when uncertain classifications arise.

Testing:

- Test the OCR’s ability to correctly extract fields from the accident report PDF.

- Test routing logic by simulating various document types.

Error Handling:

- If OCR fails to extract required fields, flag the document for manual review.

- Log errors and send notifications if classification fails or feedback is needed.

**3. Appointment Manager Agent (Airtable + Email)**

This agent will manage employee-client appointments by fetching data from Airtable and email, and sending daily structured appointment reports.

aider

Copy code

@agent

Agent name: Appointment\_Manager\_Agent

Task: Manage daily employee-client appointments and send structured email reports.

- \*\*Fetch Appointment Data\*\*:

- Connect to \*\*Airtable API\*\* to pull appointment data, including:

- Appointment time.

- ClientID and relevant client information.

- Appointment type (consultation, follow-up, etc.).

- Required documents for the meeting.

- Monitor email for new appointment requests via the \*\*Email Classification Agent\*\*, and ensure these are also stored in Airtable.

- \*\*Generate Appointment Reports\*\*:

- Each morning, generate a structured email for each employee with the following format:

1. \*\*Time of Appointment\*\*: When the appointment is scheduled.

2. \*\*Appointment Type\*\*: Consultation, review, follow-up, etc.

3. \*\*ClientID\*\*: Include relevant client details (contact info, case history).

4. \*\*Documents Required\*\*: List of documents needed for the appointment (pulled from Airtable).

5. \*\*Short Brief/Pitch\*\*: Summarize the key points the employee should focus on during the appointment (pulled from a description field in Airtable).

- \*\*Email Delivery\*\*:

- Use \*\*SendGrid API\*\* to send structured emails to each employee by 7:00 AM every morning.

- Ensure all appointments for the day are included.

- \*\*Tools/Integrations\*\*:

- \*\*Airtable API\*\*: To fetch appointment data and update the database.

- \*\*SendGrid API\*\*: To send structured daily appointment emails.

Testing:

- Simulate appointment entries in Airtable and ensure proper formatting and delivery of emails.

- Confirm that email classification of new appointments also updates Airtable correctly.

Error Handling:

- If Airtable API fails, log the issue and send a fallback email notifying employees of the failure.

- If any data is missing (e.g., client info), flag the appointment for manual review in the email.

**4. Daily and Weekly Reporting Agent (Summarize Agent Actions)**

This agent will generate daily and weekly reports on all agent activities, focusing on completed tasks and tasks requiring clarification.

aider

Copy code

@agent

Agent name: Report\_Generator\_Agent

Task: Generate daily and weekly reports summarizing agent actions.

- \*\*Daily Report\*\*:

- At the end of each day, generate a summary email for the management team.

- Include a breakdown of what each agent has completed:

1. \*\*Email Classification Agent\*\*: Number of emails classified, number of forwarded cases, response success rate.

2. \*\*Document OCR Agent\*\*: Number of documents processed, classifications made, feedback requested.

3. \*\*Appointment Manager Agent\*\*: Number of appointments processed, any errors or missing information.

- Highlight any unresolved tasks or cases where agents requested employee input.

- \*\*Weekly Report\*\*:

- Generate a more structured weekly report, summarizing the same information in \*\*table format\*\*.

- Metrics to include:

- Task completion rates.

- Number of escalations to employees.

- Time taken to classify/respond/route tasks.

- Any outstanding issues or unclassified documents.

- \*\*Tools/Integrations\*\*:

- \*\*Database\*\*: For logging daily tasks completed by agents.

- \*\*SendGrid API\*\*: To email the daily and weekly reports.

Testing:

- Simulate a week of tasks and ensure reports are generated accurately with all the required metrics.

- Verify table formatting for the weekly report.

Error Handling:

- If an agent fails to log a task, send an error notification to the management team.