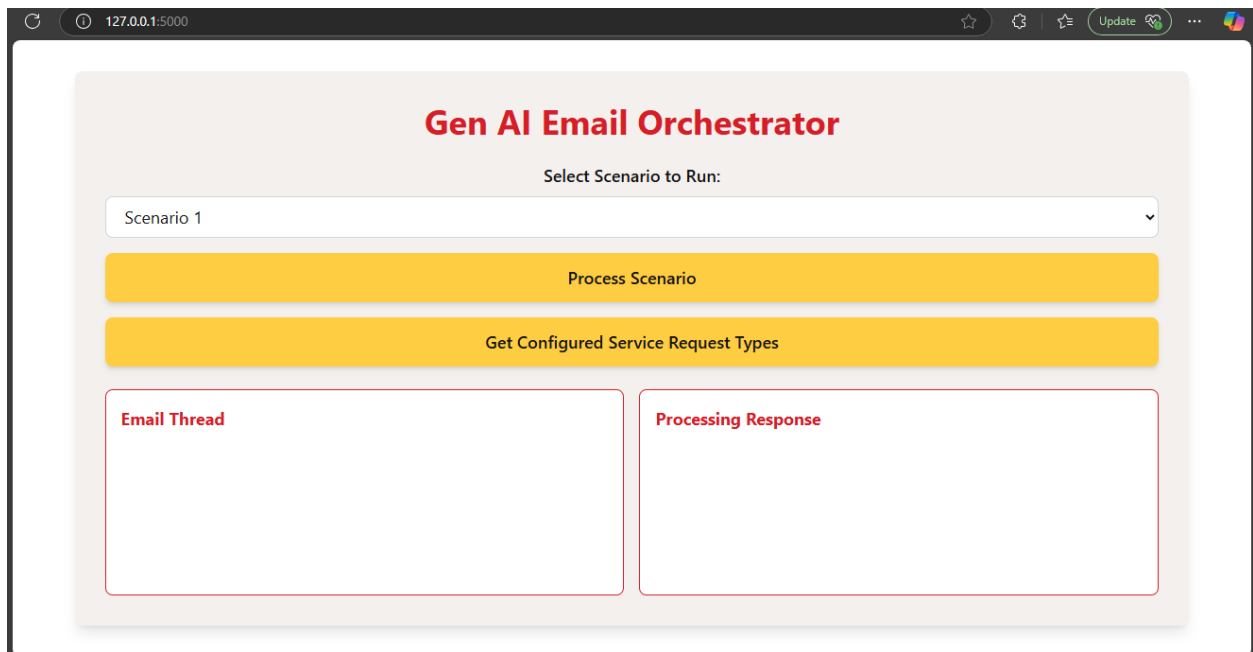


SCENARIOS CONSIDERED IN THE PROVIDED SOLUTION

Scenario 1: This scenario describes the basic flow of our solution. If an email is sent in the form of a JSON it interprets the Service Request, Sub-service request and the fields required.

Once the application is launched it looks like the picture shown below:



Select “Scenario 1” and choose “Process Scenario”

Request:

```
"Thread1": [
  Email(
    sender="customer@example.com",
    subject="Commitment Change Request",
    date="2025-03-22",
    body=(
      "Hello,I need to make a payment for an ongoing fee related to my loan."
      " The payment details, including the due date and fee period, are attached for your referen
      "Please process the payment accordingly and confirm once completed.Let me know if you need
      " information.Best regards,[Customer Name]"),
    email_id="103",
    attachments=["fee_payment_details.pdf"],
    thread_id="1008",
    sender_role="customer",
  ),
],
```

Response:

```
    },
    "reasonForNotProcessing": null,
    "responses": [
      {
        "confidence_score": 0.9022089242935181,
        "fields": {
          "Due Date": "2022-02-15",
          "Fee Period": "Monthly",
          "Loan Account ID": "123456789",
          "Payment Amount": "1000.00",
          "Payment Date": "2022-01-15",
          "Payment Method": "Credit Card",
          "Request Date": "2022-01-01"
        },
        "request_type": "Fee Payment",
        "sub_request_type": "Ongoing Fee"
      }
    ],
    "status": "processed"
  }
}
```

Scenario 2: Multiple requests in a single email

Request:

```
Email (
  sender="customer@example.com",
  subject="Request for Adjustment and AU Transfer",
  date="2025-03-22",
  body=(
    "Dear Support Team,\n\n"
    "I would like to request two actions related to my loan account:\n\n"

    "1. **Adjustment Request**\n"
    "  I need an adjustment of $2,500 due to an incorrect charge applied on March 15, 2025. "
    "The effective date for this adjustment should be March 20, 2025. The adjustment reason is an "
    "overcharge due to a system error.\n\n"

    "2. **AU Transfer Request**\n"
    "  Please transfer $5,000 from Asset Unit 'AU-12345' to 'AU-67890'. This transfer should take "
    "effect immediately.\n\n"

    "I have attached the supporting documents for your reference. Let me know if you require any addition"
    "Best regards,\n"
    "[Customer Name]"
  ),
  email_id="102",
  attachments=["adjustment_request.pdf", "au_transfer_details.pdf"],
  thread_id="1010",
  sender_role="customer",
)
```

Response: Shows 2 service requests created

```
"reasonForNotProcessing": null,
"responses": [
  {
    "confidence_score": 0.9528924822807312,
    "fields": {
      "Loan Account ID": "1",
      "Request Date": "March 15, 2025",
      "Source AU": "AU-12345",
      "Target AU": "AU-67890",
      "Transfer Amount": "5,000"
    },
    "request_type": "AU Transfer",
    "sub_request_type": null
  },
  {
    "confidence_score": 0.9397034645080566,
    "fields": {
      "Adjustment Amount": "2500",
      "Adjustment Reason": "Overcharge due to a system e
      "Effective Date": "March 20, 2025",
      "Loan Account ID": "1",
      "Request Date": "March 15, 2025"
    },
    "request_type": "Adjustment",
    "sub_request_type": null
  }
]
```

Scenario 3:

Email body contains all the fields required. Intent is taken from body and the fields are extracted also extracted from body without any attachments.

Request:

```
# Loan Information Request - enquiry
Email(
  sender="customer@example.com",
  subject="Loan Information Request",
  date="2025-03-23",
  body=(
    "Dear Support Team, "
    "I would like to request details regarding my loan. "
    "1. **Loan Type:** Business Loan "
    "2. **Planned Date:** March 25, 2025 "
    "Could you please provide the necessary details regarding interest rates and repayment terms? "
    "Best regards, "
    "[Customer Name]"
  ),
  email_id="104",
  attachments=[],
  thread_id="1009",
  sender_role="customer",
).
```

Response: Enquiry Service request type

```
"reasonForNotProcessing": null,
"responses": [
  {
    "confidence_score": 0.9869543313980103,
    "fields": {
      "Enquiry Reason": "Loan Information Request",
      "Loan Account ID": "123456789",
      "Loan Type": "Business Loan",
      "Planned Date": "2025-03-25",
      "Request Date": "2022-01-01"
    },
    "request_type": "Enquiry",
    "sub_request_type": "Loan Information"
  }
],
"status": "processed"
}
```

Scenario 4: Multiple emails in a single thread depicting a conversation between customer and support regarding the requests

Request:

```
"Thread4": [
  Email(
    date="2025-03-24",
    body=(
      "Dear Support Team, "
      "I would like to request an outbound money transfer as per the following details: "
      "1. **Transfer Date:** March 26, 2025 "
      "2. **Recipient Details:** ABC Financial Services "
      "3. **Transfer Amount:** $10,000 "
      "4. **Payment Method:** Wire Transfer "
      "Please confirm once the transaction is processed. "
      "Best regards, "
      "[Customer Name]"
    ),
    email_id="105",
    attachments=[],
    thread_id="1011",
    sender_role="customer",
  ),
  Email(
    sender="support@example.com",
    subject="RE: Outbound Money Transfer Request",
    date="2025-03-24",
    body=(
      "Dear customer, "
      "The requested transaction has been processed. Please verify from your end and reach out to u
      "happy to assist."
    ),
    email_id="106",
  )
]
```

Response: Service type being created only for the customer email and the support email has been skipped depicting that it's a support email. This can be configurable to keep the wellsFargo mailbox mail Id.

1→ shows the response for the customer email

2→ shows the response for the Support email in the thread

1.

```
"reasonForNotProcessing": null,
"responses": [
  {
    "confidence_score": 0.9136866331100464,
    "fields": {
      "Currency Type": "USD",
      "Exchange Rate": null,
      "Loan Account ID": "123456789",
      "Payment Method": "Wire Transfer",
      "Recipient Details": "ABC Financial Services",
      "Request Date": "March 26, 2025",
      "Transfer Amount": "10,000",
      "Transfer Date": "March 26, 2025"
    },
    "request_type": "Money Movement - Outbound",
    "sub_request_type": "Foreign Currency"
  }
]
```

2.

```
{
  "email": {
    "attachments": [],
    "body": "Dear customer, The requested transaction has",
    "date": "2025-03-24",
    "email_id": "106",
    "sender": "support@example.com",
    "sender_role": "support",
    "subject": "RE: Outbound Money Transfer Request",
    "thread_id": "1011"
  },
  "reasonForNotProcessing": "Support email. No processing",
  "responses": [],
  "status": "skipped"
}
]
```

Scenario 5: Here we considered the scenario where the customer first sends an email for the service request, support responds to it . Later the customer sends another follow up email where another service request should not be created.

Request:

```
Email(  
  sender="customer@example.com",  
  subject="Adjustment Request",  
  date="2025-03-25",  
  body=(  
    "Dear Support Team, "  
    "I am submitting a request for the following actions on my account: "  
    "I need an adjustment of $1,200 for an incorrect charge applied on March 10, 2025. "  
    "The effective date for this adjustment should be March 15, 2025. "  
    "I have attached supporting documents for your reference. Please let me know if further detail  
    "Best regards, "  
    "[Customer Name]"  
  ),  
  email_id="107",  
  attachments=["adjustment_proof.pdf"],  
  thread_id="1016",  
  sender_role="customer",  
)  
,  
Email(  
  sender="support@example.com",  
  subject="RE: Adjustment Request",  
  date="2025-03-25",  
  body=(  
    "Dear Customer"  
    "We are in the process of reviewing your request. Please await our response."  
  ),  
  email_id="109",  
  attachments=[],  

```

```
  thread_id="1016",  
  sender_role="support",  
)  
,  
Email(  
  sender="customer@example.com",  
  subject="Follow-up:Adjustment Request",  
  date="2025-03-27",  
  body=(  
    "this is a follow up mail for the following action on my account: "  
    "I need an adjustment of $1,200 for an incorrect charge applied on March 10, 2025. "  
    "The effective date March 15, 2025 for adjustment "  
    "I have attached supporting documents for your reference. let me know on the further details "  
    "Best regards, "  
    "[Customer Name]"  
  ),  
  email_id="110",  
  attachments=[],  
  thread_id="1016",  
  sender_role="customer",  
)  
,  
],
```

Response:

1. Shows the response for the initial email from the customer

```
"reasonForNotProcessing": null,
"responses": [
  {
    "confidence_score": 0.9807098507881165,
    "fields": {
      "Adjustment Amount": 1200,
      "Adjustment Reason": "Incorrect charge",
      "Effective Date": "2025-03-15",
      "Loan Account ID": "123456789",
      "Request Date": "2025-03-10"
    },
    "request_type": "Adjustment",
    "sub_request_type": null
  }
],
"status": "processed"
```

2. Shows that service request creation has been skipped because it was a support email

```
},
"reasonForNotProcessing": "Support email. No processing",
"responses": [],
"status": "skipped"
```

3. Shows that it detected a duplicate request and it does not create a service request again.

```
},
"reasonForNotProcessing": "Duplicate customer request.",
"responses": [],
"status": "skipped"
}
```

ADDITIONAL FEATURE ADDED:

We also considered showing all the available Service Request Types, Sub request types and the fields required for each of them

This can be achieved by clicking on the “Get Configured Service Request Types option”

The screenshot displays the 'Gen AI Email Orchestrator' interface. At the top, there is a dropdown menu labeled 'Select Scenario to Run:' with 'Scenario 1' selected. Below this are two yellow buttons: 'Process Scenario' and 'Get Configured Service Request Types'. A blue arrow points to the 'Get Configured Service Request Types' button. Below the buttons are two empty boxes labeled 'Email Thread' and 'Processing Response'. At the bottom, a large box titled 'Configured Service Request Types' displays a JSON response. The JSON shows a list of request types, with 'AU Transfer' having a description and dynamic fields, including 'Source AU' which is 'Required'.

```
{
  "request_types": {
    "AU Transfer": {
      "description": "Request to transfer an asset unit.",
      "dynamic_fields": {
        "Source AU": "Required",
```

These fields can later be made editable, and the model can automatically detect the request types later on.