

Handling Recipient SMS Replies with Twilio and .NET

Twilio's robust cloud communications platform allows developers to integrate messaging, voice, video, and email services into their applications with ease. Among its many features, Twilio enables applications to receive and process replies to SMS messages sent from your Twilio number.

When a recipient responds to an SMS, Twilio sends the reply to a webhook endpoint that you configure. This guide will walk you through setting up such an endpoint in .NET to capture and handle SMS responses.

Step 1: Set Up a Webhook Endpoint

Create an API endpoint that Twilio can post recipient SMS replies to. This endpoint will process the incoming message, log its details, and send an optional automated response back to the user.

Example Implementation

```
using Microsoft.AspNetCore.Mvc;
using Twilio.TwiML;
using Twilio.AspNet.Core;

[Route("api/sms")]
[ApiController]
public class SMSController : TwilioController
{
    [HttpPost("reply")]
    public IActionResult CaptureSMSReply([FromForm] string Body, [FromForm] string From,
    [FromForm] string To)
    {
        // Log or process the recipient's response
        Console.WriteLine($"Received SMS from {From} (reply to {To}): {Body}");

        // respond back with a simple message
        var response = new MessagingResponse();
        response.Message("Thanks for your reply! We've received your message.");

        return TwiML(response); // return the response. You can use this for logging.
    }
}
```

Explanation of Key Parameters

From: Represents the phone number of the recipient who replied.

To: Represents the Twilio phone number that sent the initial message.

Body: Contains the text of the recipient's response message.

Step 2: Configure the Webhook in Twilio

To enable Twilio to send SMS responses to your application, you must configure the webhook URL in your Twilio Console.

Steps to Set Up the Webhook:

1. Log into Twilio Console: Navigate to the Twilio Console.
2. Manage Numbers: Go to Phone Numbers > Manage Numbers.
3. Select Your Number: Click the phone number you used to send the initial SMS message.
4. Set Webhook URL: Under the Messaging section, find the A Message Comes In field. Enter the URL of your API endpoint (e.g., <https://yourdomain.com/api/sms/reply>).
5. Save: Click Save to apply the changes.

Workflow

1. Twilio Receives Reply: When a recipient replies to your SMS, Twilio sends the response details (including the sender's number and the message content) to the configured webhook endpoint.
2. Endpoint Processes Message: The webhook captures the data, allowing you to log or further process it.
3. Send Automated Response (Optional): You can use Twilio's `MessagingResponse` to send an automatic acknowledgment or additional information back to the recipient.

Example Use Cases

Feedback Collection: Automatically log responses to gather customer feedback.

Interactive Services: Build conversational experiences where replies trigger specific workflows.

Support Systems: Capture responses to follow up or route messages to the appropriate support team.

Conclusion

Setting up a webhook to handle SMS replies with Twilio and .NET is a straightforward way to engage with users and integrate two-way messaging into your application. By combining Twilio's messaging capabilities with your application logic, you can create interactive and responsive communication systems without needing to manage telecom infrastructure.