

Smart Giving

The process for the end giver goes something like this: they text the word "Give" and the amount they want to give, i.e. "Give \$100.00", to the SmartGiving phone number. If this is the first time they have used the system, they receive a link to a form that they can use to register their phone. That's when we ask for their payment information, which is then linked to their phone number. If they text to the SmartGiving number again, their card is immediately charged.

The system that we built has three components: 1) The text message gateway, which is the part that sends and receives text messages to our system; 2) The payment gateway, which we use to actually store the credit card information and charge the card; and 3) our web server, which is the brains behind the system.

MESSAGE GATEWAY

We use Twilio (<http://www.twilio.com>) for our text message gateway. Twilio's service provides phone numbers that you can use to send and receive text messages. While you can initiate text messages using Twilio, we only use their system to "receive" and then "reply" to text messages. The setup is relatively simple. Once you have a phone number registered with Twilio, you can specify a URL that Twilio will send a request to every time the number receives a text message. Among other items, the information this URL receives contains the phone number of the sender, as well as the body of the text. What that URL responds with will determine what Twilio does; our URL sends back a response to return a specific text to the user.

PAYMENT GATEWAY

For our payment gateway, we use Authorize.net, one of the leading online payment gateway services. Authorize.net is not a merchant account, rather you link your merchant account to them in order to access the ability to charge credit cards via your own electronic system. One of the services they provide is what they call the Customer Information Manager, or CIM, which you can use to set up customer profiles that contain contact and billing information. For each user account we set up

a customer profile, and send the credit card and other billing information into this profile. Authorize.net in turn gives us an ID for the customer profile and an ID for the billing information. The billing information is no longer accessible to us directly, and we do not retain any user's credit card information ourselves, so we don't have to worry about PCI compliance regarding encrypting and storing this data. Instead, using the customer profile ID and the billing profile ID together we are able to place a charge on the card as needed. This is the same feature set that we use to process our automatic recurring giving.

WEBSITE

Our website ties these two systems together. When a request comes in to our site from Twilio, our system grabs the phone number and parses the body of the message. If we're unable to parse the message or if the message is one of the standard texts we must reply to (i.e. "help" or "options"), we return an appropriate response. Otherwise, we check to see if the phone number is currently in the database, and if it is, we charge the card the requested amount and send a confirmation text and e-mail. If it is not in the database, we store the phone number in another table along with a hash code that uniquely identifies this phone number. We then send back a text that contains a URL that contains the hash code. When they open the link, our system uses the hash code to look up in the corresponding phone number, and when they submit the form we send the info to Authorize.net and store the two ID's along with their phone number into our database. If we receive a "Stop" command, or if the charge ever fails, we remove the profile information from Authorize.net and delete the link in our database, which will require the user to register again with their updated information the next time they attempt to give via the system.