



Serverless Application with Step Functions, API Gateway, Lambda, and S3

The background features a large, irregular watercolor shape in shades of blue and green. The top portion is a darker blue, while the bottom portion is a lighter green. Scattered around this central shape are numerous small, dark blue dots. A thin, dark blue line curves across the right side of the image, starting from the top and ending near the bottom right. In the bottom right corner, there are three overlapping, hand-drawn loops in a dark blue color.

01

The Project Itself

Project (Purpose & About)

1. The overall idea was to create a fully functional serverless application using some of the most important services
2. Gain hands-on experience with architecting AWS solutions

Goals

01

Sol. Architecture

Get hands on experience with architecting solutions with AWS



Expanding on previous knowledge and applying the new concepts and creating better solutions

02

Serverless App Knowledge

Increase knowledge of serverless apps and understand the architecture while implementing one.



02

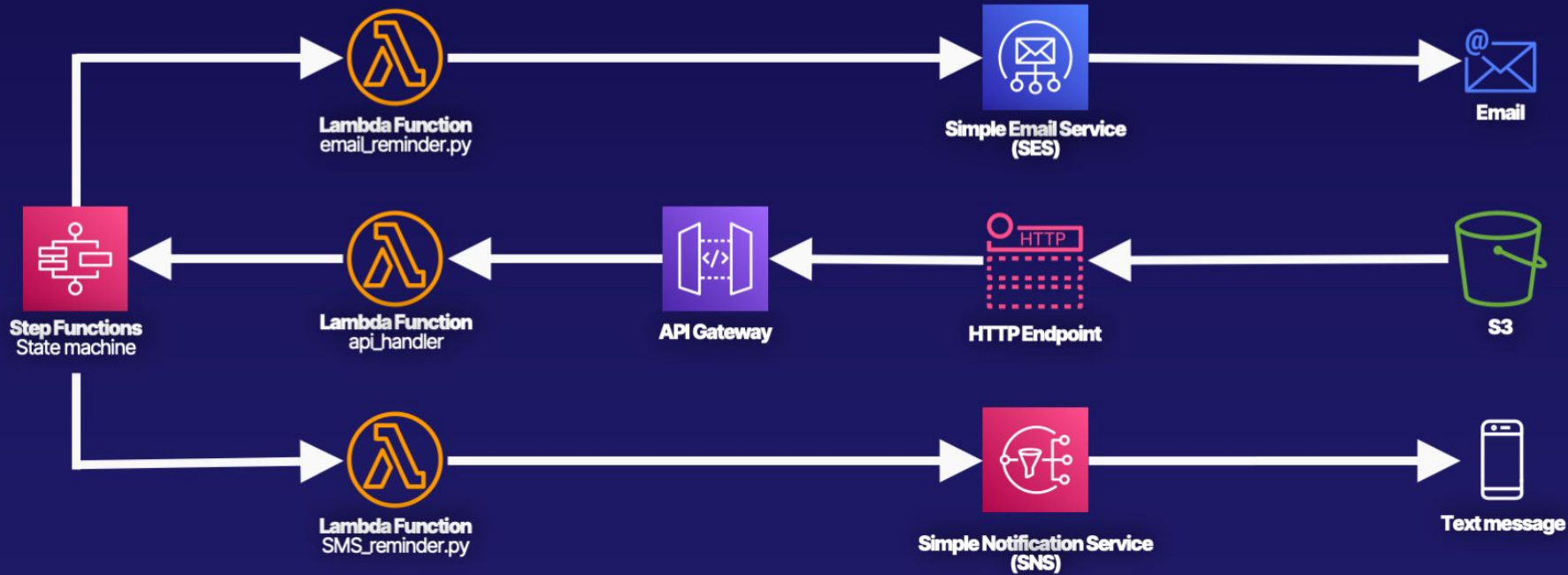
The Procedure



Technical/Architectural Requirements

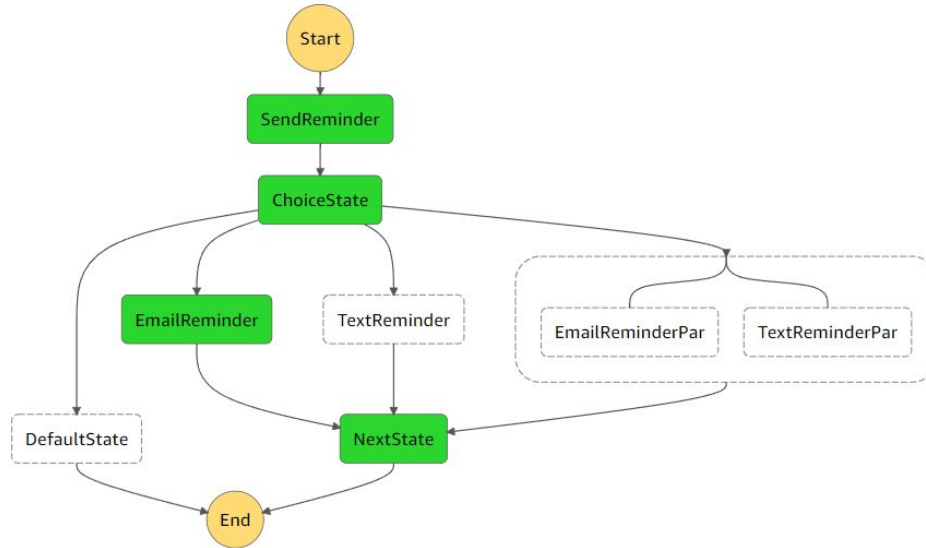
1. The app is used by people who have animals and need reminders for feeding time, cleaning, walks and etc.
 - a. The static website will be hosted on S3 bucket
 - b. The javascript will communicate with the API gateway
 - c. The API gateway interacts with api handler lambda function
 - i. The functions interacts with the Step Machine
 1. The Step machine interacts with the two Lambda functions, SMS and Email that send the said reminders

Topology Requirements (Pictorial)




Topology cont. (Step Functions)

- Below is the logic used by the Step Functions



Running Example

1. Here we put a “message” on the website for the reminder, in this case we say “hello.” We also specify how many reminders need to be sent
 - a. We then choose between an email or a mobile reminder (both can be picked)



FembotIT Pet Reminders
Remember all the things

Looks ok. But check the result below!
Required sections:

1

hello

Email address is required for email reminders, phone number is required for phone reminders:

example@gmail.com

+15556667788

Reminder Type:

both

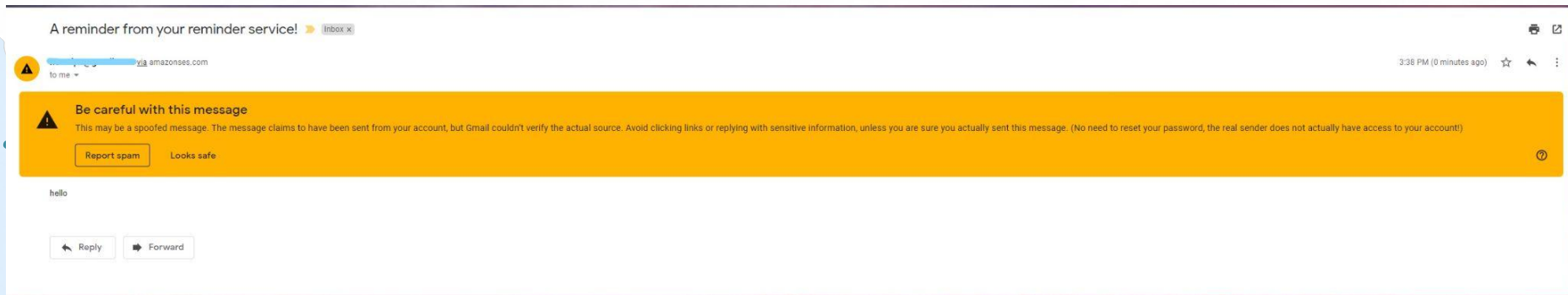
email

sms

("Status": "Success")

Results

Below is the result, after selecting the email reminder, we get an email with the exact message that we need.



Goals Achieved

01

Solutions Architecture

Successfully architected a serverless reminder app solution.

Increased understanding of architecture concepts

Valuable practical experience working with AWS Services

02

Serverless App Knowledge

Increased understanding of serverless applications and hands-on experience implementing them