Shane Gardner
Olga Rocheeva
Ihar Simanovich
CSS490 - Cloud Computing
Program 4 Report

Location of the URL of the site

http://www.squadtext.com

List of the 6+ services utilized

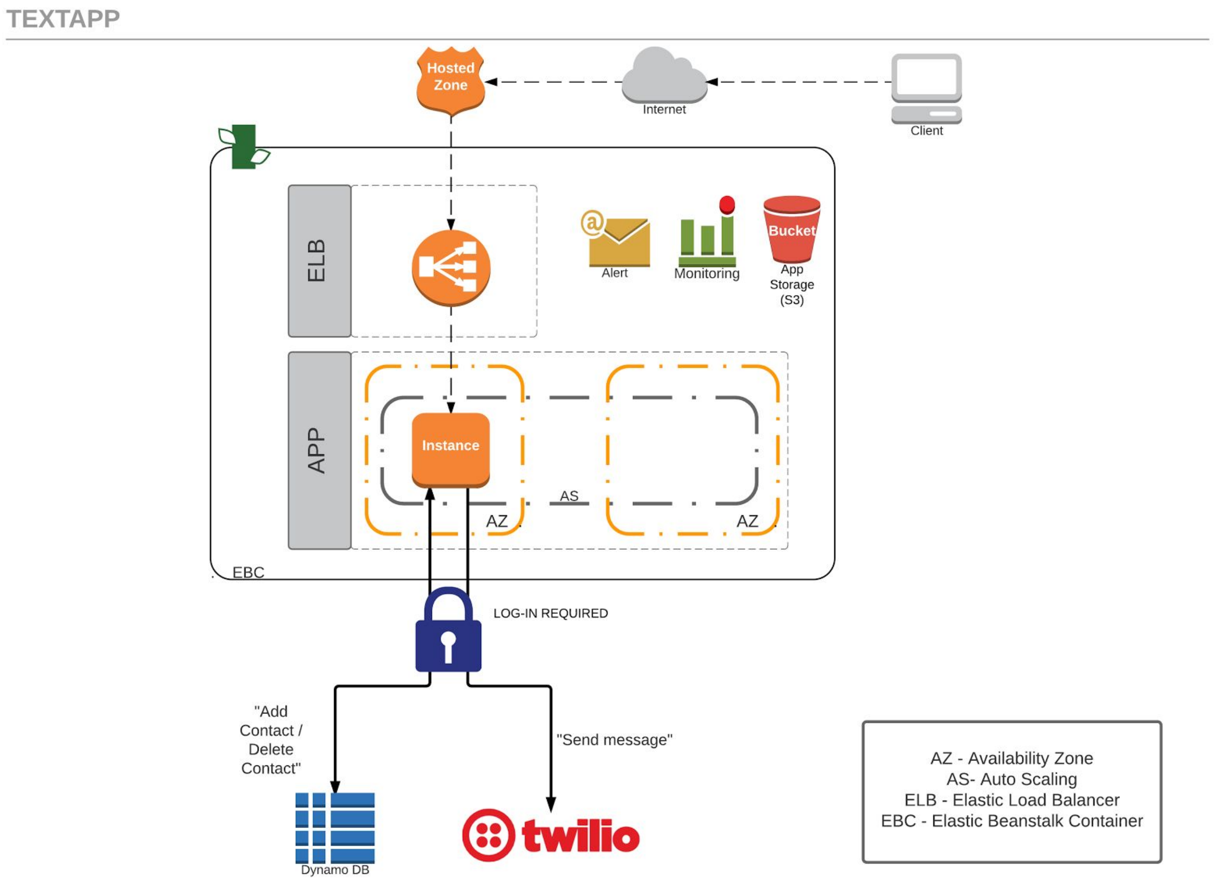
1. DynamoDB
2. Elastic Beanstalk
3. Twilio API
4. SQLAlchemy SQL DB
5. Github (development)
6. IAM (user management)
7. Route 53
8. GoDaddy
9. S3

Dependencies

- **AWS Elastic Beanstalk**: AWS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.
- **Flask:** A lightweight Python web framework based on Werkzeug and Jinja 2.
- **Boto3:** Boto is the Amazon Web Services (AWS) SDK for Python, which allows Python developers to write software that makes use of Amazon services.
- **AWS DynamoDB**: Amazon DynamoDB is a fast and flexible NoSQL database service for all applications that need consistent, single-digit millisecond latency at any scale.
- **SQLAlchemy**: NoSQL database framework for python to store user information and query for logging in.
- **Twilio**: SMS service that can be used to register a unique phone number and send messages to any other phone number. API is interfaced to many languages, including Python.
- **Github**: File-sharing repository for collaborative and iterative software projects
- **S3:** Hosts the web-application files to be used by Elastic Beanstalk
- **GoDaddy:** For a unique and relevant website name
- **Route 53:** Used to link the Elastic Beanstalk URL with the GoDaddy URL

Shane Gardner
Olga Rocheeva
Ihar Simanovich
CSS490 - Cloud Computing
Program 4 Report

Clear design diagrams of final project



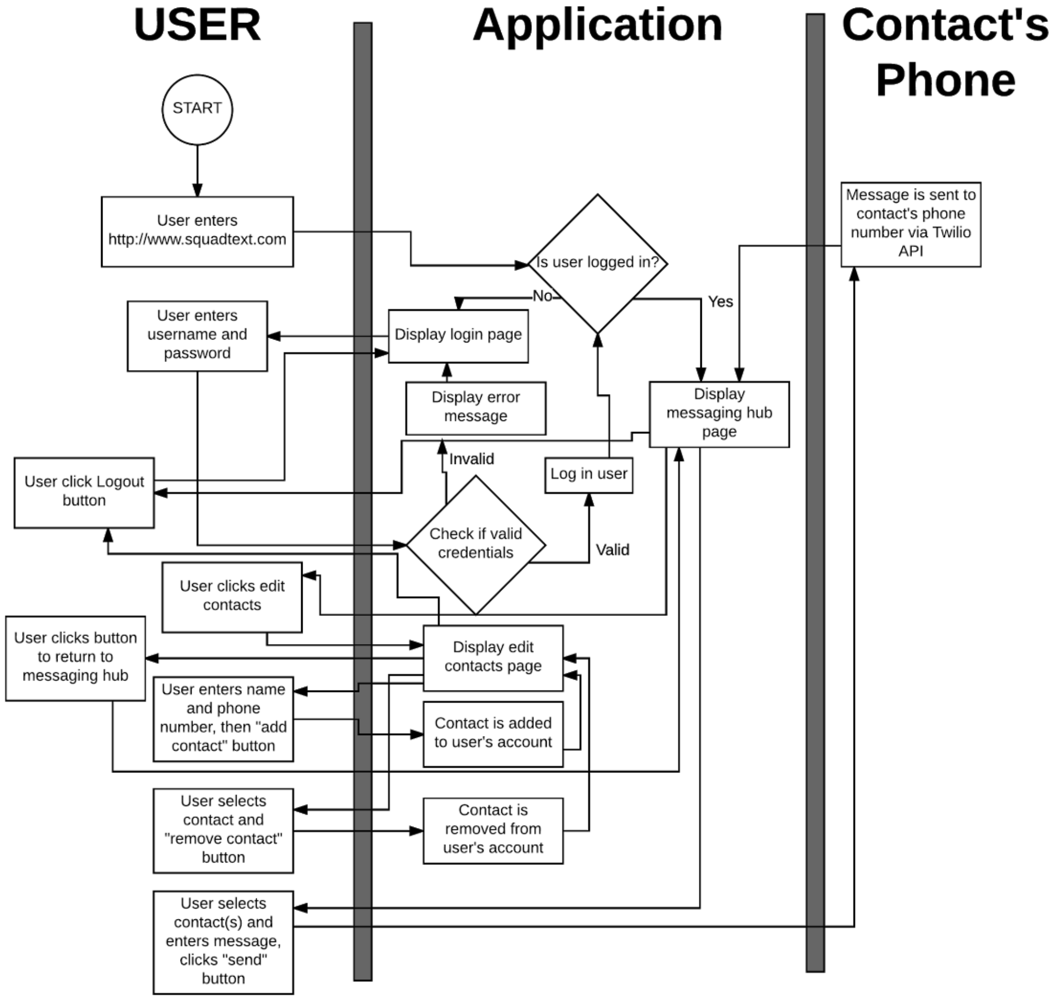

USER
Application
Contact's Phone
START
User enters http://www.squadtext.com
Is user logged in?
No
Yes
Message is sent to contact's phone number via Twilio API
User enters username and password
Display login page
Display error message
Display messaging hub page
Invalid
Log in user
User click Logout button
Check if valid credentials
Valid
User clicks edit contacts
User clicks button to return to messaging hub
Display edit contacts page
User enters name and phone number, then "add contact" button
Contact is added to user's account
User selects contact and "remove contact" button
Contact is removed from user's account
User selects contact(s) and enters message, clicks "send" button

Figure 1.2: Activity Diagram

Explanation as to why you chose Azure or Amazon

Our whole group had used AWS for Program 3. Flask-Python was very efficient for accomplishing what we needed before and would be easily expanded upon for Program 4. Meshing the HTML and Python functionality was very easy because of Flask-Python.

Clear statement of the SLA for your service and a discussion as to why the SLA is achievable

Shane Gardner
Olga Rocheeva
Ihar Simanovich
CSS490 - Cloud Computing
Program 4 Report

Our service requires that each of the services below are available and running smoothly to function. Given the risk that each of the services provided in their SLAs, our service can expect to be available **99.9%** of the time. In the event that the service is unavailable, notifications will be sent to all users with permission to monitor the service.

Twilio's SLA
"Twilio will use commercially reasonable efforts to make the Twilio API available 99.95% of the time. In the event Twilio does not meet the goal of 99.95% API availability in a given calendar month ("Monthly Uptime Percentage"), you will be eligible to receive a Service Credit as described below."

Route 53 SLA
"AWS will use commercially reasonable efforts to make Amazon Route 53 100% Available"

DynamoDB SLA
"The service runs across Amazon's proven, high-availability data centers. The service replicates data across three facilities in an AWS Region to provide fault tolerance in the event of a server failure or Availability Zone outage."

EC2/Elastic Beanstalk SLA
"AWS will use commercially reasonable efforts to make Amazon EC2 and Amazon EBS each available with a Monthly Uptime Percentage (defined below) of at least 99.95%, in each case during any monthly billing cycle (the "Service Commitment"). In the event Amazon EC2 or Amazon EBS does not meet the Service Commitment, you will be eligible to receive a Service Credit as described below."