TeXtTop

Stage I - Proposal and Specifications

CSC 470
Dr. Depasquale
February 7 2013

Dan Daly
Derek Duchesne
Steven Kolln
Dan Seminara

Github: https://github.com/stee11/TeXt-Top
WikiPage: https://tcnj.instructure.com/groups/81511/wiki/front-page

Problem Statement:

Users who cannot access their computer for a long period of time are unable to do actions such as triggering programs and creating files. Although remote desktop software exists, such as in situations where no computers are nearby or when network access is impossible, being able to manage tasks on your computer solely by texting would be extremely beneficial. Through the use of cloud instances and storage, users could be able to freely use this software at any time without having to worry about the current status of their computer.

System Objective:

TeXtTop will provide users with the ability to remotely control their computer via texting from a phone using AWS' services. The system will utilize Amazon's EC2, S3, DynamoDB, SNS, and SQS systems to register and store user accounts, send alerts by text and email, and allow the user to create their own text commands by uploading python scripts to the system's cloud server.

Desired End Product:

The product will be composed of a website where users will register their account and phone number as well as a desktop application that the user can download from the website. The desktop application will contain a folder encompassing all of the user's commands, including default commands already packaged with the software and the user's own personal Python scripts. When the user sends a text message matching a command, the EC2 instance of TeXtTop that monitors a Google Voice account will verify that the command is valid and add it to a queue. From the queue, each command will be executed from the local application or, in the case of the computer being off, will wait in the queue until the computer is turned on.

Importance:

TeXtTop will give users more control over their daily tasks by enabling remote desktop use simply by texting. Having a smart phone is not required, and services in the cloud abstract the command processing and user verification processes that the user would have to deal with when routinely logging into a website. Instead, this service provides easy use and one-time installation.

Similar Systems:

Remote desktop programs such as Chrome Remote Desktop, TeamViewer, and LogMeIn all support accessing a desktop through another computer through use of a browser. In the case of TeamViewer, there is an emphasis especially on group sharing of desktops and files. TweetMyPC is probably the most similar software to what TeXtTop will be; that program reads tweets and performs specific actions based on what was tweeted.

Innovation:

No other program will let a user be able to control his/her computer via text messages. In addition to that, the ability to create custom commands will bring about many unforeseen applications. Though remotely controlling your computer is possible through the use of other programs, this system specifically uses cloud services in order to queue up commands even when the computer is off and manages storage virtually.

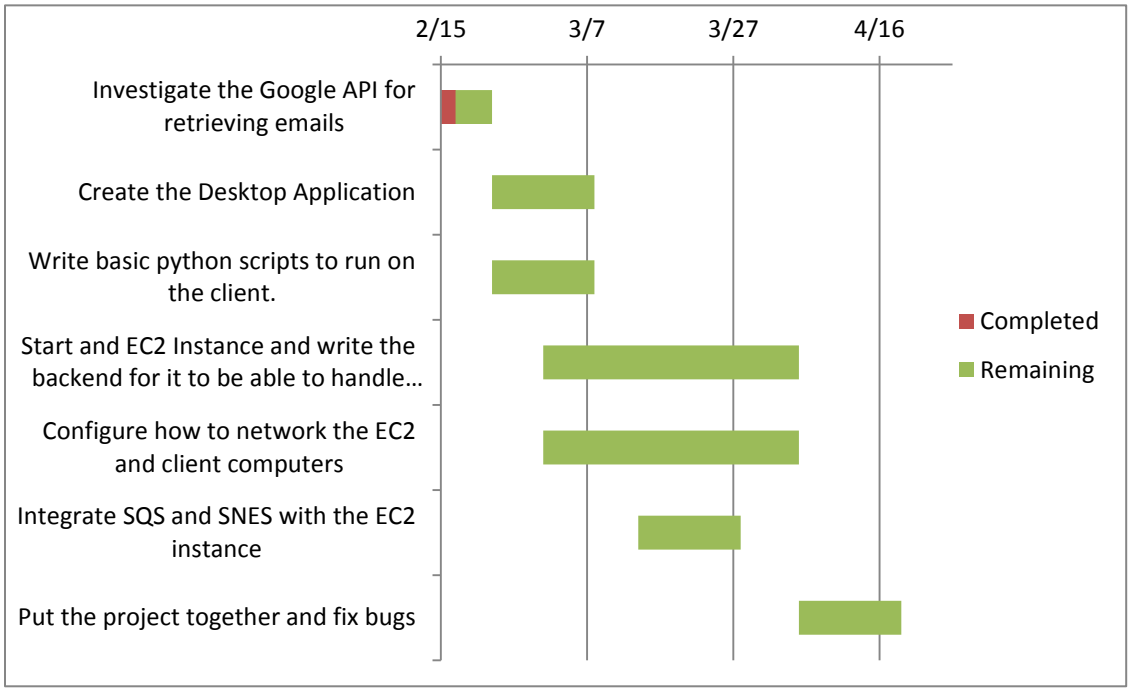
Technologies that need to be learned:

Python will be the main programming language used for development, which includes the use of the Google Voice and Gmail APIs for Python. We will need to learn how to make GUIs in Python for creating the remote host. We will be responsible for writing Python code that can interact with AWS' services (EC2, S3, DynamoDB, SNS, and SQS).

Other Applications:

TeXtTop could be used to keep memos and schedule events by setting up commands that send messages to your computer on startup or at specific times of the day. Custom commands allow users with Python programming knowledge to add near limitless capability for their own specific needs.

Gannt Chart:



**Our other dates and milestones can be found on Github @*
https://github.com/stee11/TeXt-Top/issues?milestone=12&state=open

System Boundary Diagram