This ReadMe file is dedicated to setting up and running the software; the hardware, with its relatively much easier setup, is assumed to already have been done. By following the steps below, the user should be able to run the software as we did.

1. Download the Arduino IDE and run our file "ArduinoCodeForTransmission"
2. The IDE will ask you to create a new sketch folder upon running it for the first time; say yes, and the program will be then able to send the 434 MHz transmissions
3. Next, create an account at Twilio.com to make use of the text messaging service. You will receive an account SID, an authentication token, as well as a Twilio phone number. You will need these three items for later.
4. Install the rtl-sdr library onto the raspberry pi using the following command in the terminal: sudo pip install pyrtlsdr
5. Install the Twilio api using the following command in the terminal: sudo pip install Twilio
6. Unzip the berrybell.zip file into a convenient location.
7. Open the berrybell.py folder which you unzipped in a text editor
8. In the account_sid field, change the given value to the one you received when you made your Twilio account
9. In the auth_token field, change the given value to the token you received when you made your Twilio account
10. In the client.messages.create field, where it says "from = ", change the phone number listed to the phone number that was given to you when you made your Twilio account
    a. NOTE: proper international formatting is required for the "from = " value as well as the "to = " value. For example, a US phone number should be formatted as "+1XXXXXXXXXX".
11. In the client.messages.create field where it says "to = ", change the phone number listed to your phone number. Twilio is compatible with non-US phone numbers, so if you wanted to enter a Chinese mobile phone number, for example, the proper formatting would be "+861XXXXXXXXXX"
12. Save the changes you made to the Python file
13. Open another terminal window and change your directory to the folder where you unzipped the files, e.g. cd /Users/raspberry_pi_user/Documents/berrybell
14. Using the following command, run the Berrybell file to start polling for the transmitted signal: python PythonCodeForReceiving.py
15. The program should now receive the signal via the SDR when the button for the transmitter is pressed.
    a. NOTE: If you want to generate plots of the signal strength, run the command "python PythonCodeForPlotting" before Step 12