1. Confidentiality / integrity / availability
    a. Confidentiality
        i. A potential threat could be that an attack seeks to collect stored user data to collect information about who a user is and discover their location for further harm. To protect confidentiality, the user data should be stored securely and be encrypted. Furthermore, the least amount of information on the user should be stored while still making the app work.
    b. Integrity
        i. A potential threat to integrity could be that an attacker works to create and use false identities within the app to maliciously interact with users and send viruses to them. A way to mitigate this would be to have strict authentication policies for logging into the system, and actively scan for potential threats.
    c. Availability
        i. An availability security issue could be a user lockout, where an attacker maliciously takes advantage of a lockout system for incorrectly guessed passwords/etc. A solution to this problem could be to make a password reset/re-entry into the program easily accessible via email. Furthemore, a user should receive automated emails informing them of incorrectly guessed passwords.

2. Privacy Policy Sketch:
    a. We will be collecting biographic user data, in regards to what an individual user is like both physically and what they are interested in socially. A large component of the user data will be in relation to their musical taste, which is paired with one's overall account in order to better match and understand social relationships. Some example of the biographic data we will be storing is:
        i. Age
        ii. Gender
        iii. Geographic Location
        iv. Sexuality
        v. Etc.
    b. The user data we are collecting, both in regards to one's biography and spotify trends, are stored in FireBase. FireBase is a publicly trusted and

well known set of hosting services for various types of applications. FireBase is built on Google’s infrastructure, and is categorized as a NoSQL database program, which stores data in JSON-like formats. There is little security threat when using such a trusted and publicly accepted framework.