<div align="center">**Design Studio 3**</div>

Task 1:

Diagram 1
- **Sequence Diagram**
  - **Objects/Actors:** Database, UI, Jose(Actor), Fluffy(Actor), App Server, Tracker Server
  - Sequence 1 - Add a new toy profile
    - Jose → UI → App Server → Database → App Server → UI → Jose

    Sequence 2 - Edit existing toys' info
    - Jose → UI → App Server → Database → App Server → UI → Jose
  - Sequence 3 - Tracking
    - Jose → UI → App Server → Database (Identify which toy to track) → App Server → Tracker Server → Fluffy → Tracker Server → App Server → Database (store the location) → App Server → UI → Jose
  - Sequence 4 - Buy a tracker
    - Jose → UI → App Server → Tracker Server → App Server → Database (store the order info) → App Server → UI → Jose
  - Sequence 5 - Deleting toys
    - Jose → UI → App Server → Database → App Server → UI → Jose
  - Sequence 6 - Register an account
    - Jose → UI → App Server → Database → App Server → UI → Jose

- **Summary**

  For this diagram, we first identified the major objects of the Toy Tracker: the application database, the Users Jose(Actor), Fluffy toys (Actor), the application server and the tracker server. Fluffy toys and Jose are the two actors that will interact with the system --- Fluffy toys' real-time location information needs to be stored with the system and Jose uses the system to track his Fluffy. The application database is used for storing the user account information, the Fluffy lists and their location. The application server enables us to connect the database and the user via a stable internet. The UI serves to give feedback to the users and represent all the needed requested information. Doing the sequence diagram helped us identify the architecture processes of the this application
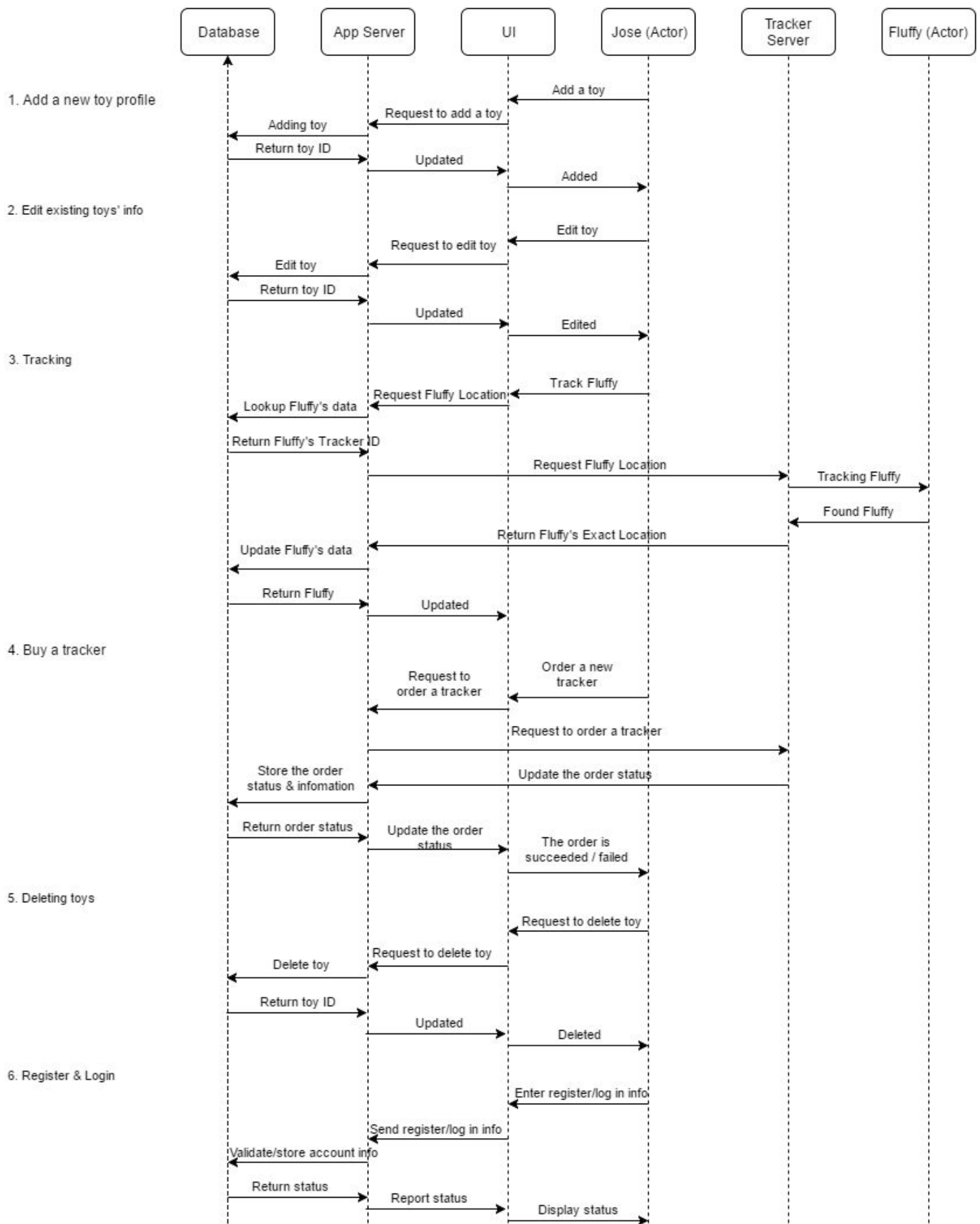
1. Add a new toy profile

Jose (Actor) → UI: Add a toy
UI → App Server: Request to add a toy
App Server → Database: Adding toy
Database → App Server: Return toy ID
App Server → UI: Updated
UI → Jose (Actor): Added

2. Edit existing toys' info

Jose (Actor) → UI: Edit toy
UI → App Server: Request to edit toy
App Server → Database: Edit toy
Database → App Server: Return toy ID
App Server → UI: Updated
UI → Jose (Actor): Edited

3. Tracking

Jose (Actor) → UI: Track Fluffy
UI → App Server: Request Fluffy Location
App Server → Database: Lookup Fluffy's data
Database → App Server: Return Fluffy's Tracker ID
App Server → Tracker Server: Request Fluffy Location
Tracker Server → Fluffy (Actor): Tracking Fluffy
Fluffy (Actor) → Tracker Server: Found Fluffy
Tracker Server → App Server: Return Fluffy's Exact Location
App Server → Database: Update Fluffy's data
Database → App Server: Return Fluffy
App Server → UI: Updated

4. Buy a tracker

Jose (Actor) → UI: Order a new tracker
UI → App Server: Request to order a tracker
App Server → Tracker Server: Request to order a tracker
Tracker Server → App Server: Update the order status
App Server → Database: Store the order status & infomation
Database → App Server: Return order status
App Server → UI: Update the order status
UI → Jose (Actor): The order is succeeded / failed

5. Deleting toys

Jose (Actor) → UI: Request to delete toy
UI → App Server: Request to delete toy
App Server → Database: Delete toy
Database → App Server: Return toy ID
App Server → UI: Updated
UI → Jose (Actor): Deleted

6. Register & Login

Jose (Actor) → UI: Enter register/log in info
UI → App Server: Send register/log in info
App Server → Database: Validate/store account info
Database → App Server: Return status
App Server → UI: Report status
UI → Jose (Actor): Display status

Diagram 1 - Sequence Diagram

Diagram 2
- **UML for app entities**
    - User
        - Email log in
            - Password
        - Name
        - Toy Collection
    - ToyCollection
        - Collection ID
        - Toys array
    - Toy
        - Name
        - Toy ID
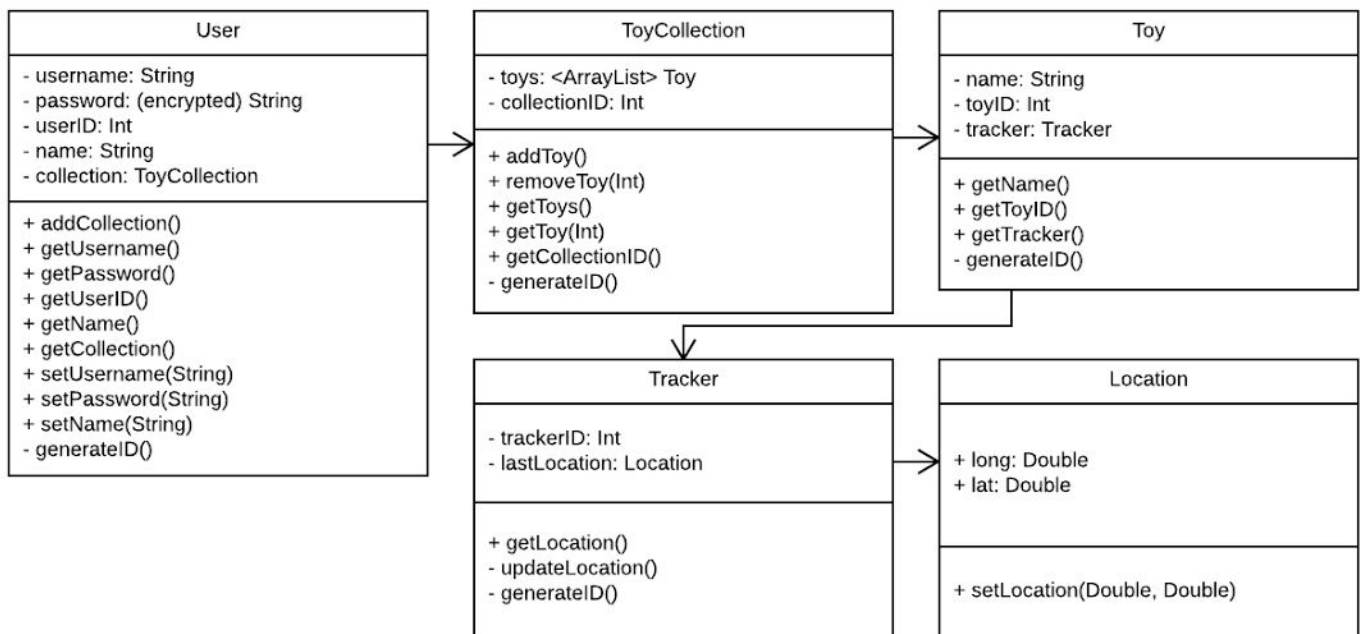   - Tracker
        - Photo
    - Tracker
        - Tracker ID
        - Last location

User
- username: String - password: (encrypted) String - userID: Int - name: String - collection: ToyCollection
+ addCollection() + getUsername() + getPassword() + getUserID() + getName() + getCollection() + setUsername(String) + setPassword(String) + setName(String) - generateID()

ToyCollection
- toys: <ArrayList> Toy - collectionID: Int
+ addToy() + removeToy(Int) + getToys() + getToy(Int) + getCollectionID() - generateID()

Toy
- name: String - toyID: Int - tracker: Tracker
+ getName() + getToyID() + getTracker() - generateID()

Tracker
- trackerID: Int - lastLocation: Location
+ getLocation() - updateLocation() - generateID()

Location
+ long: Double + lat: Double
+ setLocation(Double, Double)

Diagram 2 - UML

Task 2:

UI Mockups/ sitemap

See Diagram 3

1. Login/Make Account
   - Login → Homepage
   - Sign up → Create Account → homepage
2. Homepage
   - Toy Directory
     - Search for toy
   - Menu
     - Add Toys
       - Fill out item info
         - Name
         - Picture
         - Tracker ID
     - Your Toys
       - List of all Toys
         - Toy Info
           - Tracker ID
           - Name
           - Picture
           - Owner
           - Track
             - Find location of item
               - Open location in Google Maps
     - Your Account
       - Personal Information
         - Name
         - Email
         - Change password
     - Buy A Tracker
       - Individual
       - Packs
     - Settings
     - Help
       - Frequently Asked Questions
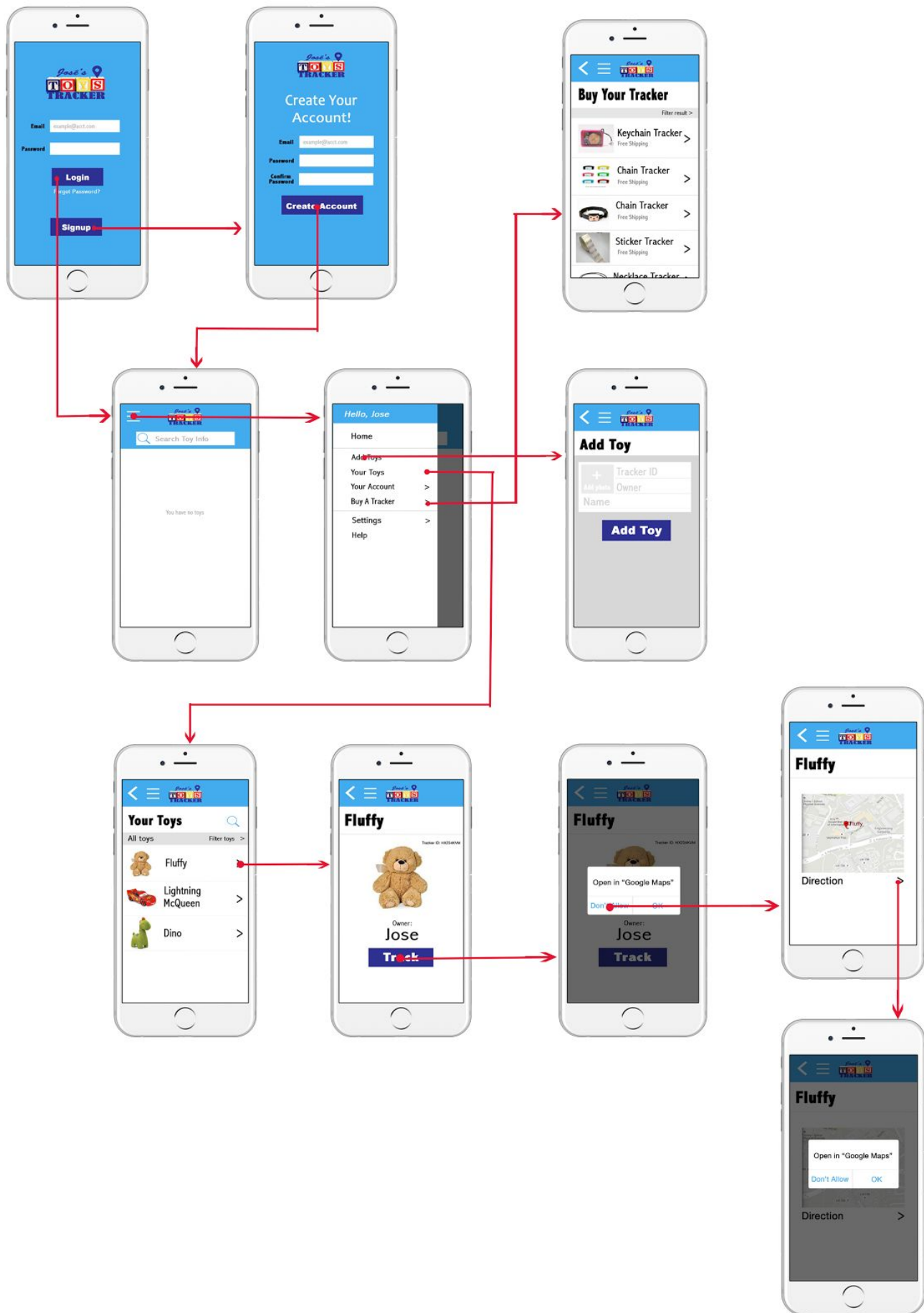
Diagram 3 - UI Mock-up/Site-Map

Design Methods

- **Personas**
    1. Max
        - 9-year old boy from New York and is in 4th grade at Johnson Elementary School
        - Max has 2 older brothers and a younger sister
        - His favorite color is red and loves toy trucks
        - Max has a favorite teddy bear, Mr. Brown, that he can't leave the house without
        - He will not go anywhere if he loses it and throws a tantrum
    2. Mary
        - 8-year old girl from Florida is in 3rd grade at Polk Elementary School
        - Mary is the youngest of 3 girls
        - Mary loves princesses and enjoys going to Disney World
        - Her favorite toy is an Elsa doll
        - She always misplaces it
        - Refuses to go to school without the doll
    3. Anna (Parent)
        - Anna has 5 children
        - She has a hard time keeping track of her children's toys
        - Her kids are ages 3,5,2,7 and 9
        - Anna is also a first grade teacher at Kennedy Elementary School
        - Her kids all have a favorite toy that they cannot sleep without
        - She has trouble keeping track of all 5 favorite toys
        - Anna likes to put her kids to bed early so that she can watch her favorite shows and drink the occasional glass of wine

    Report: The personas gave us perspective of what types of users to accommodate our design to. From doing the personas we were also able to modify some design elements to ensure that various types of users (children and parents) would be able to use the application successfully and efficiently. Writing the personas gave us more clarity on who the application is intended to gear towards as well as the purpose of it.

- **Scenarios**
    - Jose is an 8 year old boy who has a favorite teddy bear named Fluffy. When he was 7 years old he once lost fluffy and was miserable. He refused to eat or go to school and would throw tantrums. His mother bought him a new teddy bear and Jose vowed to never lose his new Fluffy ever again. He decided to use Jose's Toys Tracker to ensure that he always knows where Fluffy is at all times. He downloaded the application and added Fluffy to his toy directory. He asked his mom to help him order a pack of trackers through the application. Once he received the trackers he placed them on all his favorite toys, Fluffy, Lightning McQueen, and Dino. He tied the tracker onto his toys and added the toy's tracker ID to the toy's information on the application respectively. Five months later Jose went to his friend Rico's house for a sleepover and brought Fluffy with him. The next morning Jose's mother was in such a rush to pick him up and when he got home he realized he did not have Fluffy. He opened Jose's Toys Tracker and tracked where Fluffy was. His mother drove him back to Rico's house and they played the alarm notification which played a sound that led to Fluffy.

- Lisa is 5 years old and has a stuffed toy name Max. Lisa will never leave the house without Max and whenever it's time to head out the door, it takes at least 5 minutes to find Max in the house. Lisa's mother decides to find a solution for their troubles. She find the Toy Tracker on Amazon and decides to order one and give it a try. Upon arrival, the packaged quick-start guide has a QR code for the accompanying app. As she downloads the app on her phone, she opens the tracker and decides to attach the tracker with the chain around the collar of Max. Once she follows the onscreen instructions to add the tracker to her account, she is able to track the tracker's current location and set off the trackers locating alarm. Now whenever they need to head out and the toy is nowhere in sight, Lisa's mother would check the tracker location. If the location wasn't enough to find it, she would set off the homing alarm to locate the toy hidden in corners or under things so that they could head out the door in a timely manner.
- Jenny just turned 6 and has 10 stuffed animals with different names. Jenny always takes a toy with her to play with. However, there are 10 toys and sometimes she forgets where she leaves them. Jenny's dad bought her a tracker and downloaded the Toy Tracker application on his phone, hoping these would help her daughter locate her toys. They first set the tracker on Jenny's favourite toy, Piggy. One day, Jenny forgot where she put Piggy and started crying. Her dad used the Toy Tracker and turned on the alarm notification and they successfully found Piggy in the bathroom by following the sound. Jenny's dad was very satisfied with the Toy Tracker so he bought another 9 trackers for his daughter's other toys. From then on, whenever Jenny lost any toys her dad uses the Toy Tracker to locate the them.

Report: The scenarios allowed us to view specific use cases for the application. This helped us realize different features that users would need in different situations that we left out in our initial design. We realized that having a GPS location alone may not be enough to find a toy in smaller areas or if the toy was hidden from sight. With this in mind, we ended up adding a sound alarm to help them find their items more efficiently.

- **Mind Mapping**
    - See Diagram 4

Report: The map led us to thinking about the real life application of the physical product more and how it would work with the software.In order to keep the product low cost, only a passive RFID tracker would be used along with a homing alarm for the locating a toy. The compromise would be that there would only be a set range where the tracking would work and is only for smaller areas. Users would have to know at least the vicinity of where the toy would be. The upside was that only a small battery to power the alarm was needed and the tracker with the alarm makes it very easy to find a toy in a house or somewhere like a restaurant or park. The mind map allowed us to view every aspect of the application to ensure we had worked on every part and did not leave anything out. The map was a good way of viewing every detail down to the core.

Diagram 4 - Mind Map