Exhibits: Africa, Aviary, North America.

Sub exhibits:

Africa: Watani grasslands (WG), red river hogs (RRH), forest edge (FE), African lion (AL), kitera Forest (KF), Lemur Island (LI), Forest Glade (FG), Baboons (Ba)

Aviary: Birds (Bi), land animals (LA)

United States: Rocky Coast (RC), Marsh (Ma), Cypress Swamp (CS), Stream side (SS), Bears (Be), Honey bee garden (HBG), Prairie (Pr), Red Wolf (RW), Desert (De).

The sub-exhibits will just be identifiers for the different animals. The Main exhibits will be containers for the animals, which can be sorted by the different sub-exhibits and then the zoo will be the container for everything.

Zoo

Africa Aviary US

WG RRH FE AL KF LI FG Ba Bi LA RC Ma CS SS Be HBG Pr RW De

From there we can implement the animals. I like the use of ArrayLists of animal objects but of course with that comes the lack of speed that hashing has. I don’t feel very comfortable with Hashing yet, so that could be something we work on instead.

Without stepping on the User interface designer, I think that there should be some layers of questions. So when you come into the interface, ask things like: are you searching, browsing or editing the list. From there go down into the layers: which exhibit would you like to search/browse/edit? I don't think we should assume people will know where they want to edit things past that, so there should be some way to figure out where it goes, aka compare it to similar animals in that exhibit and ask if it needs to go into that sub-exhibit. Again, not set in stone, just trying to help out and make our jobs less independent!

Update 4/9! I think instead of doing multiple array lists, it would be better to just make the sub-exhibit a category of the animal. That way it it can be sorted by something pertaining to that particular animal. The only thing I could see being a problem is if the animal needs to switch sub-exhibits, but that should be rare. So instead of coding a different class for each sub-exhibit, I am just going to code the three main exhibits and then the zoo, which will most likely be a class with a main method, starting the GUI.

Let me know what you think of that idea.