**Uruguay Database- MySQL Tables**

Samples

A sample is an abstract entity. A sample is “collected” in Uruguay inside a test tube. The sample stores information that remains constant, no matter how many times you aliquot the contents of the original test tube collected in Uruguay into more test tubes.

This constraint is reflected in the columns of the table. For example, after a sample is collected in Uruguay, if I pour that sample’s contents into another test tube, the child that it was collected from will not change. Hence, Child\_id is a property of the Samples table. The same can be said for every other column in the table.

Only the field researches in Uruguay should be allowed to insert rows into the Samples table.

Columns:

Id- unique to every sample (even if from the same child); automatically generated

Child\_id

Date\_collected

Type- one of blood/serum/plasma/urine/hair/soil/water/blood spot

(The following are chemical properties of the sample, and are subject to change)

Density

Hb

Pb

Tubes

A test tube represents the physical test tubes in which the sample is stored. After a sample is collected in a test tube in Uruguay, the contents of that test tube can be aliquoted into other test tubes.

There one main things that can change when you aliquot into another test tube: the treatment applied to the contents inside it. Thus, it is a column of the Test Tube table.

When a sample is collected in Uruguay, in addition to adding a new row into the Samples table, we also add a row into the Tubes table to represent the original test tube that the sample is stored in. Any subsequent aliquots of that test tube should be reflected by adding new rows into the Tubes table.

Columns:

Id- unique to every test tube (even if from the same sample); automatically generated

Sample\_id- references the id column of the Samples table that the contents of this test tube are associated with

Treatment- one of BHT/EDTA/Foil wrapping/Heparin/Heparin+MPA

Freezers

A freezer is place that a test tube can be stored. They can be coolers, lab freezers, etc. A lab/university can have multiple freezers. The freezer table should never change, unless we are updating information about its storage temperature.

TODO: will there be more freezers in the future? Can a freezer change temperature?

Columns:

Id- unique to every freezer (even if in the same lab); automatically generated

Lab\_name

Storage\_temp (C)

Shipment\_Batch

It is oftentimes the case (I think) that many test tubes are sent together from one lab to another. In that case, they would be bunched together into a “shipment batch,” and a lot of information regarding the shipment itself would pertain to all the test tubes inside the shipment batch.

TODO: Upon collecting the sample in Uruguay, we should have an initial shipment from Uruguay lab to Uruguay lab, just so that we know where it is if its been collected but not shipped to UB yet.

Columns:

Id- unique to every shipment batch

Date\_shipped

Shipping\_company

Source\_freezer- Id of the freezer the shipment batch was sent from

Destination\_freezer- Id of the freezer the shipment batch was sent to

Receipt- Boolean, which is updated to true when the shipment batch is received by the destination lab; on inserting a row into the table, this value is false

Shipment\_Tube

Any time any test tube is shipped, a row should be written inside this table. The purpose of this table is to track the path of any given test tube, so that we have detailed records as to where every test tube has been at any time.

Columns:

Id- unique to every tube shipment. In other words, any time a test tube is shipped, that row will get a new Id

Shipment\_id- references the id of the Batch\_Shipments table that the tube was shipped in

Tube\_id- references the id of the Tubes table