Deadline - Friday 2/8

1. CSV

export properly functioning with all data about each piece of equipment, space or zone

2. Supply Chain

Should be complete with the ability to add links for products

INSERT

```
PREFIX : <http://www.semanticweb.org/chris/ontologies/
2018/10/untitled-ontology-7#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
INSERT DATA{
    :___existingEquip :hasItem :___newProductInventory
Link .
    :___newProductInventoryLink a :ProductLink ;
        :hasInventoryLink ""^^xsd:string ;
        :hasVendor ""^^xsd:string ;
        rdfs:label "" .
}
```

REQUEST

```
PREFIX : <http://www.semanticweb.org/chris/ontologies/
2018/10/untitled-ontology-7#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?link ?label ?vendor
WHERE {
    :___existingEquip :hasItem ?linkItem .
    ?linkItem a :ProductLink ;
        :hasInventoryLink ?link ;
        :hasVendor ?vendor ;
        rdfs:label ?label .
}
```

3. Related Section –

######*Components related to Equip*

```
PREFIX : <http://www.semanticweb.org/chris/ontologies/
2018/10/untitled-ontology-7#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
CONSTRUCT {
    ?node :hasRelated ?relative
}

WHERE{
```

```
    { ?node a :Equip . ?node :deliversService+ ?
relative . }
    UNION { ?node :hasComponent+ ?relative . }
    UNION { ?node :hasMONCON+ ?relative . }
    ?relative a :Equip
}
```

######*Equip related to rooms*

######*Equip related to zones*

4. Sense –

Refrigerator graph

Temperatures

Power

Elevator graphs:

Location

Velocity

Acceleration

Jerk

5. Work Order/Purchase Order

Should be able to create and save

WORK ORDER INSERT DATA

```
PREFIX : <http://www.semanticweb.org/chris/ontologies/
2018/10/untitled-ontology-7#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
INSERT DATA{
    :___workorderID a :WorkOrder ;
        a owl:NamedIndividual ;
        :hasDueDate ""^^xsd:dateTimeStamp ;
        :hasStatus ""^^xsd:string ;
        :hasTitle ""^^xsd:string ;
        :hasPriority ""^^xsd:string ;
        :isAssigned :[Contact ID] ;
     :hasEntryTimeStamp ""^^xsd:dateTimeStamp ;
     :hasComponent :[po], :[po], :[po]… ;
     :hasCreationDate ""^^xsd:dateTimeStamp .
}
```

######WORK ORDER REQUEST DATA

```
PREFIX : <http://www.semanticweb.org/chris/ontologies/
2018/10/untitled-ontology-7#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?due ?status ?title ?priority ?assignee ?aname ?
lastedit ?created
WHERE{
    :___workorderID :hasDueDate ?due ;
        :hasStatus ?status ;
        :hasTitle ?title ;
        :hasPriority ?priority ;
        :isAssigned ?assignee ;
     :hasEntryTimeStamp ?lastedit ;
     :hasCreationDate ?created .
    ?assignee :hasName ?aname .
}
```

######PURCHASE ORDER INSERT DATA

```
PREFIX : <http://www.semanticweb.org/chris/ontologies/
2018/10/untitled-ontology-7#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
INSERT DATA {
    :___purchaseorderID a :PurchaseOrder ;
        a :NamedIndividual ;
        :hasStatus ""^^xsd:string ;
        :hasEntryTimeStamp ""^^xsd:dateTimeStamp ;
        :hasCompany ""^^xsd:string ;
        :hasComponent :[item], :[item], :[item]... ;
        :hasAddress ""^^xsd:string ;
     :hasTotal ""^^xsd:double .

}
```

######PURCHASE ORDER REQUEST DATA

```
PREFIX : <http://www.semanticweb.org/chris/ontologies/
2018/10/untitled-ontology-7#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?status ?submit ?company ?shipping ?wo ?total
(GROUP_CONCAT(DISTINCT ?item;separator=", ") AS ?items)
WHERE{
```

```
    :___purchaseorderID :hasStatus ?status ;
        :hasEntryTimeStamp ?submit ;
        :hasCompany ?company ;
        :hasComponent ?item ;
        :hasAddress ?shipping ;
        ^:hasComponent ?wo ;
     :hasTotal ?created .

} GROUP BY ?status ?submit ?company ?shipping ?wo ?total
```

6. Task

Should be able to create a task and send a message to a phone
Should also be able to demonstrate that a message is received from a phone

I sent this to nitin a while back. This query. Is for committing tasks to DB.

```
PREFIX : <http://www.semanticweb.org/chris/ontologies/
2018/10/untitled-ontology-7#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-
ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

DELETE {
    ?s ?p ?o
}

INSERT {
    :_____existingTaskID____
a :____pick_a_subclass_of_task___ ;
        a owl:NamedIndividual ;
        :hasDueDate [] ;
        :hasDuration [] ;
        :hasEquipSubject [] ;
        :hasTaskDescription [] ;
        :hasStartDate [] ;
        :hasWorkType [] ;
        :hasTaskNotes [] ;
        :assignedTo [] ;
        :hasRelatedLink [] ;
        :hasEntryTimeStamp [] .
}
```

```
WHERE {
    ?s ?p ?o
    filter (?s = :_____existingTaskID____ || ?o
= :_____existingTaskID____)
}
```

CASE 1: `We modify a task and leave the reference chain intact` We can use the query I just provided, without any further modification.

CASE 2: `We delete a task in the chain` We have to get the next task of the task that we wish to delete, and replace the triple in the task before the one that we are deleting with that value.

CASE 3: `We are deleting the first task in the task chain` We have to replace the task reference in the ticket with the reference to the second task in the chain, so it becomes the first.

CASE 4: `We are adding a task in between two other tasks or task and ticket` The `:hasNextTask` reference of the ticket we insert will be the task that follows it in the new order. The reference in the Ticket or Task in the reference chain before the the new task will have the reference to the new task that we are adding.

7. Thermostat

Demonstrate that there is a working thermostat with basic control
We will provide the proper interface with more instruction on Monday

``````````````````````````````````````**MORE QUERIES**``````````````````````````````````````

THIS QUERY IS FOR GETTING THE ZONE CLASSIFICATION MENU INFORMATION:

```
PREFIX : <http://www.semanticweb.org/chris/ontologies/
2018/10/untitled-ontology-7#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>

SELECT ?z ?zone ?y ?yname(count(?hop) as ?distance)
WHERE{
```
```

```
     ?z rdfs:subClassOf* ?hop .
     ?hop rdfs:subClassOf+ :Zone .

FILTER(?z != owl:Thing)
FILTER(?z != owl:Nothing && ?z != :Zone)

    ?z rdfs:label ?zone .
    ?z rdfs:subClassOf ?y .
    ?y rdfs:label ?yname

FILTER(?y != owl:Thing)
FILTER(?y != owl:Nothing && ?y != :Zone && ?y != ?z)

}
GROUP BY ?zone ?z ?y ?yname
HAVING (?distance = 3)
```

CORRECTED POINT DROP DOWN:

```
PREFIX : <http://www.semanticweb.org/chris/ontologies/
2018/10/untitled-ontology-7#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>



SELECT ?point ?label WHERE{
         ?point rdfs:subClassOf :Point .
FILTER (?point != :Point && ?point != owl:Nothing && ?point
!= owl:Thing)
```

```
                ?point rdfs:label ?label .
}
```

MORE SOON
Chris