TrimbleCloud Requirements

Functional Objectives:
- Create a project boundary using a series of longitude and latitude points to define a valid polygon to encapsulate a project site
- Define an infinite number of zones consisting of a set of longitude and latitude points that must exist within the boundary to form a valid polygon
- Observe data within defined boundaries and be able to understand/navigate through that data
- Generate notification events as resources move between zones/boundaries, or important resource attributes change
- Define arbitrary resources to represent construction related content within the context of the project boundary and subsequently within the context of a zone
- Receive information about a change in zones with regard to a resource
- Add/remove/modify a resource

Non-Functional Objectives:
- Initial coordinate system hardcoded to noSQL server on AWS ec2
- Django framework for server
- User sends https GET request to server to receive JSON packets with current system data initially and for any subsequent changes (iterative)
- Creation of zones/resources are uploaded to unique user space in server with https POST requests
- Spatial Geometry Support will be interfaced through Geojson

Technology Stack:
- Django
- noSQL Database - DynamoDB
- AWS Services
    - Amazon Location Services
    - AWS SNS
    - AWS Cognito
    - AWS EventBridge
    - AWS Amplify

User Management:
- One total user class with full permissions to access any valid requests
- User authentication - not necessary/ out of scope for now

Use Case Name	Define new resource
Summary	Create a data representation of a resource in a designated space.
Basic Flow	- User chooses zone to assign resource to - Enter description/ field data for resource
Alt. Flow	- Confirm boundary lines - User is prompted to partition boundary into zones - Resources are allocated to zones as needed
Extension Pts.	---
Preconditions	- Defined boundary data set - Defined zone within boundary
Business Rules	---

Use Case Name	Remove existing resource
Summary	Remove resource from zone/boundary
Basic Flow	- User chooses a resource - Remove option is selected - Confirm removal - Resource data is cleared and dependencies are resolved
Alt. Flow	---
Extension Pts.	---
Preconditions	- At least one existing resource
Business Rules	---

Use Case Name	Modify existing resource
Summary	Change resource's data

Basic Flow	- User chooses a resource - Modify option is selected - Specify modification - Update resource
Alt. Flow	---
Extension Pts.	---
Preconditions	- At least one existing resource
Business Rules	---

Use Case Name	Moving resources
Summary	Generating events as a Resource moves between zones
Basic Flow	- Define relevant resource - Select the resource - Move option is selected - Choose new zone - Change event saved into db - Notification is generated (email) - Resource is updated
Alt. Flow	---
Extension Pts.	---
Preconditions	- At least one existing resource
Business Rules	---

Use Case Name	Define new zone
Summary	Create a new zone within the designated boundary

Basic Flow	- Provide set of coordinates to bound zone - Check validity of coordinates - Create zone within system
Alt. Flow	---
Extension Pts.	---
Preconditions	- Defined Data Set and boundary
Business Rules	---

Use Case Name	Remove existing zone
Summary	Remove existing zone from the designated boundary
Basic Flow	- Define zone for removal - Select zone for removal - Remove the zone
Alt. Flow	- If resources are still defined in the selected zone they must be moved prior to zone delete
Extension Pts.	---
Preconditions	- Defined boundary and at least 1 existing zone - No active resources within zone
Business Rules	---

Use Case Name	Modify existing zone
Summary	Change the zone perimeter

Basic Flow	- Add/remove nodes - Check for node validity - Adjust node locations
Alt. Flow	- If node is out of zone, prompt another option - Must be at least 3(?) nodes
Extension Pts.	---
Preconditions	- Defined boundary and at least 1 existing zone
Business Rules	---