

DATA MODELS

1. Resource:

- `name = ndb.StringProperty(indexed=True)`
The name of the resource
- `availability = ndb.StructuredProperty(Availability, repeated=True)`
An object of the Availability data model indicating the start and end time of the resource availability
- `tags = ndb.StringProperty(repeated=True)`
List of tags associated with the resource
- `owner = ndb.StringProperty()`
Name of the owner of the resource
- `id = ndb.StringProperty(indexed=True, required=True)`
Unique ID for the resource
- `lastReservedTime = ndb.DateTimeProperty(auto_now_add=False)`
Last reserved time of the resource. For a new resource this is null
- `startString = ndb.StringProperty()`
Start time in string format
- `endString = ndb.StringProperty()`
End time in string format
- `dateString = ndb.StringProperty()`
Resource availability date in string format
- `smallImg = ndb.BlobProperty()`
Small version (icon) of the image uploaded for an image resource
- `imageId = ndb.StringProperty()`
Id of the image associated with the image resource
- `imageDescription = ndb.StringProperty()`
Description of the image resource
- `count = ndb.IntegerProperty()`
Number of times a resource has been reserved
- `image = ndb.BooleanProperty()`
Boolean property to indicate if the resource is an image resource
- `capacity = ndb.IntegerProperty()`
Overlapping reservations capacity of the resource

2. Images:

- `imageId = ndb.StringProperty(indexed=True, required=True)`
Unique ID of an image
- `fullImage = ndb.BlobProperty()`
Full image to be displayed on the image page
- `description = ndb.StringProperty()`
Description of the image

3. Availability(ndb.Model): I could remove this class as it does not serve much purpose but it's too late now! ☹

- `startTime = ndb.DateTimeProperty(auto_now_add=False)`
The start time of an available time slot.
 - `endTime = ndb.DateTimeProperty(auto_now_add=False)`
The end time of an available time slot.
4. `class Reservations(ndb.Model):`
- `owner = ndb.StringProperty(indexed=False)`
The owner of the reservation
 - `resourceId = ndb.StringProperty(indexed=True, required=True)`
The ID of the resource that is reserved.
 - `resourceName = ndb.StringProperty(indexed=False)`
The name of the reserved resource
 - `startTime = ndb.DateTimeProperty(auto_now_add=False)`
The starttime of the reservation
 - `duration = ndb.IntegerProperty()`
The duration of the reservation in minutes
 - `strignStart = ndb.StringProperty()`
The starttime of the reservation in string format
 - `uid = ndb.StringProperty(indexed=True, required=True)`
The unique id of the reservation
 - `date = ndb.StringProperty()`
The date of reservation in string

ASSUMPTIONS

1. Resources can have duplicate names.
2. If a resource is edited, associated reservations are deleted. This can be handled better in the future.

EXTRA CREDITS

1. The resource page shows how many times a resource has been reserved. This includes all undeleted reservations(past, current and future).
2. Resources can be searched for by name.
3. Images can be uploaded as resources. The description of an image resource is not shown on the landing page as it would be inconsistent with non-image resources.
4. When a reservation is made a confirmation email is sent to the user (reservor).
5. When a reservation starts an email confirmation is sent to the user. This was implemented using a cron job.
6. Every resource has a capacity that indicates the number of allowable overlapping reservations. Once the capacity is reached no more overlapping reservations can be made for an user.