PROJECT DESIGN & DOCUMENTATION

The application follows the MVC architectural pattern for implementing user interfaces. This in turn allows efficient code reuse and parallel development.

MODELS

Resource: The Resource Model has the following attributes:

1. id: The unique id of the resource. This is also an indexed and required field
2. resourceName: The name of the resource. This is also an indexed field
3. ownerID: The id of the user that created/owns the resource. This is also an indexed field
4. availableStartTime: The time from which the resource is available
5. availableEndTime: The time up to which the resource is available
6. tags: Comma separated string values with which the resource is tagged
7. numberOfTimesReserved: The total number of times the resource has been reserved up till now, including completed and deleted reservations. The default value is 0.
8. lastReservationTime: The last time the resource was reserved by a user
9. capacity: The maximum number of times a reservation can be made at the same time for the resource. The default value is 1.
10. avatar: An image with which the resource is associated
11. description: A description of the resource

Reservation: The Reservation Model has the following attributes:

1. reservationID: The unique id of the reservation. This is also an indexed and required field
2. resourceID: The id of the resource for which the reservation has been made. This is also an indexed and required field
3. date: The date for which the reservation has been made. This is also an indexed field
4. ownerID: The id of the user that made the reservation. This is also an indexed field
5. startTime: The time at which the reservation starts. This is also an indexed field
6. endTime: The time at which the reservation ends
7. resourceName: The name of the resource for which the reservation has been made
8. duration: The total duration of the reservation
9. ownerEmail: The email ID of the user that made the reservation

CONTROLLERS

LandingPage: This is the default handler of the application. It is used to generate the index.html template which displays a list of all resources currently saved in the GAE data store, the current user's reservations and resources created/owned by him/her in a tabular form. It also allows the facility to add a resource to the data store. Moreover, it allows searching of resources by name (case-sensitively) and by availability, i.e., by entering the desired date, time and duration of reservation. Once the end time for a reservation has passed, it is no longer displayed.

UserPage: This handler is used when the current user wishes to see reservations made by and resources owned by another user. The handler retrieves this information utilizing the requested user's ID.

CreateResource: This handler is used when the current user wishes to add a new resource to the GAE data store. The handler generates the CreateResource.html template. After the current user has submitted the request to add a new resource, the handler adds that resource to the data store, after stripping all user inputted data of leading and trailing white spaces. Since the fields: tags, avatar and description are optional, it is possible that the newly created resource may not have any data for these fields. Moreover, if the user has added an avatar image file, it is resized to 32x32 before being saved in the data store.

ResourcePage: This handler is used when the current user wishes to see the details of a resource. The handler generates the resourcePage.html template which displays all details of a resource – its name, availability start time, availability end time, tags, capacity, avatar, description and upcoming reservations of that resource. It also allows adding a new reservation for the resource and the option to edit the resource details (if the current user is the owner of the resource).

EditResource: This handler is used when the current user wishes to edit an already existing resource (that he/she owns). The handler generates the editResource.html template. Previously entered information about the resource can be seen on this page. After the current user has submitted the request to edit the resource, the handler finds the correct resource to be edited, using the unique id field, and adds the newly inputted values for the resource fields. The avatar is changed only if the current user has added a new image. In case he/she hasn't, the previous avatar remains unchanged, if any.

CreateReservation: This handler is used when the current user wishes to add a new reservation for an already existing resource. The handler generates the createReservation.html template. After the current user has submitted the request to add a new reservation, the handler validates the user inputted values:

- It first checks that the requested date and start time of the new reservation has not elapsed
- It then checks if the resource for which the reservation is being made has reached its capacity
- Finally, it checks if the requested date, time and duration of the new reservation overlaps with any of the current user's existing reservations.

If the requested date, time and duration passes all of the above validations, it is considered as a valid reservation and is saved in the data store. The resource's lastReservationTime and numberOfTimesReserved fields are also updated. Also, an email confirmation is sent out to the current user notifying him/her of the reservation.

ViewReservations: This handler is used when the current user wishes to view details of all upcoming reservations of a resource. The handler generates the viewReservations.html template which displays all current/upcoming reservations of the resource, i.e., all reservations of the resource whose end time has not yet passed.

DeleteReservation: This handler is used when the current user wishes to delete an upcoming reservation of their own. The handler finds the requested reservation with the unique reservationID field and deletes it from the data store.

TagPage: This handler is used when the current user wishes to see resources tagged by a given tag. The handler finds all resources in the data store that have been tagged with this tag (case-sensitively) and generates the tagPage.html template to display details of these resources, if any.

RSSPage: This handler is used when the current user wishes to generate RSS for all reservations of a resource. The handler finds the resource with the unique id field and all upcoming and past/undeleted reservations of that resource. It then generates the RSSPage.html template displaying all reservations of the resource, along with the details, in an XML format.

SearchName: This handler is used when the current user wishes to search for resources by the resource name. The handler finds all resources having that name (a case-sensitive exact match is made), and generates the searchResults.html template to display the list of resources, if any.

SendMailViaCron: A cron job has been added to the cron which runs every minute. This handler is used by the cron job to send an email to the owner of any upcoming reservation when the reservation's start time has arrived. Please note that the reservation confirmation and reminder emails are send out with the sender email: "anishabhatla281@gmail.com" as that was the email ID with which I created this project on Google Cloud.

GetImage: This handler is used when the current user wishes to see the avatar associated with a resource. This handler finds the resource with the unique id field and returns its avatar field.

SearchByAvailability: This handler is used when the current user wishes to search for resources by availability. The handler generates the searchByAvailability.html template. After the current user has submitted the request to search by availability, the handler finds validates the user inputted value by checking if the requested date and start time of the reservation has elapsed. If not, the handler finds all resources that are available and have not reached their capacity for the requested date, time and duration.

VIEWS

Please note that all UI testing has been done on Google Chrome Version 58.0.3029.110 (64-bit).

index.html

This page has 5 sections:

1. Your Reservations: All upcoming reservations of the current user and their details can be seen here. The reservations are sorted by reservation date and start time. Only those reservations can be viewed whose end time has not yet passed. There is also an option to delete the reservation. After deletion, the reservation is no longer seen in any view/page.
2. All Resources: This displays all resources that currently exist in the GAE data store. They are sorted in the reverse order, based on last made reservation (by any user).
3. Your Resources: This displays all resources in the GAE data store that are created/owned by the current user. These are sorted in the reverse order, based on last made reservation (by the current user).
4. Search Options: There are two options offered here: Search by Name and Search by Availability:

a. Search by Name: This can be used when the current user wishes to search for resources given a name. He/she may enter a resource name and then press 'Enter' on the keyboard. A list of resources with their name being a case-sensitive exact match is returned.

b. Search by Availability: This is a URL that links to searchByAvailability.html.

5. Add Resource: This is a URL that links to createResource.html.

Also, every resource name is a URL that links to the resource page for that resource. Every user email displayed is a URL that links to the UserPage. Every RSS superscript displayed above the resource name is a URL that links to an RSSPage that generates information about the reservations made by that resource in an XML format. Every tag of the resource is also a URL that links to the tagPage.html. This index.html is also called the "Home" page.

Please Note:

After the end time of a reservation has passed, the reservation can no longer be viewed on any page. However, the reservation still exists in the GAE data store and is not deleted.

Validations:

The search box of Search by Name is such that it does not submit if it is empty or filled with empty spaces only.

createResource.html

This page can be used by the current user to add a new resource to the GAE data store. Following are the mandatory fields required to be filled in by the user:

- Resource Name
- Available Start Time
- Available End Time
- Capacity

Optional fields are tags, image and description. Upon successful creation of the resource, a message will be displayed "ResourceName has been created successfully".

Please Note:

- Any leading/trailing white spaces in any of the user inputted values are removed before saving.
- The resource can be made available for a maximum time of 23 hours and 59 minutes (i.e., less than a day).
- Tags are comma separated values. Thus, comma is not allowed in the tag field. This is a limitation currently.
- Any leading/trailing spaces in any of the tag values are also removed before saving. Thus, "NYU, OST" is same as "NYU,OST".
- Two or more tags having the same value are considered redundant and will result in only one of the tags being saved. For instance, "NYU, OST, OST, OST,OST" will be saved as "NYU,OST"
- Capacity field allows integer values greater than or equal to 1 only. This field cannot be left blank.

- Only a single image file can be uploaded in the Image field. The image is resized to 32x32 before saving.
- The description field can have maximum of 50 characters.

Validations:

- Resource Name cannot be left empty or filled with empty spaces only.
- Available Start Time and Available End Time should be inputted in the 24-hour format (hours:minutes).
- Available End Time field should be greater than the Available Start Time field.

resourcePage.html

This page displays details about the resource – name, available start time, available end time, tags, capacity, number of times booked in the past (including deleted/completed reservations), image and description. This is the only page which displays the image and description fields of the resource. This page also has 3 sections:

1. View Upcoming Reservations: URL that links to viewReservations.html which displays all upcoming reservations of the resource, i.e., those whose end time has not yet passed.
2. Add Reservation: URL that links to createReservation.html, which helps in adding a reservation for that resource.
3. Edit Resource: URL that links to editResource.html, which helps in editing the resource details. This link is made available only to those users that are the owners/creators of the resource.

editResource.html

This page can be used by the current user to edit details of the resource. The entered information about the resource is displayed as default values. The same rules apply here as in the case of createResource.html. The only limitation is that if there is an image already uploaded for the resource, there is no way to remove it completely (without replacing it). After the editing has been done successfully, the index.html page is shown with the message "ResourceName has been edited".

Please Note:

If the resource is edited and its available start and end times changed, the upcoming reservations that already exist (with the old availability hours) persist and are left unchanged.

createReservation.html

This page can be used by the current user to add a new reservation for a resource. The resource may/may not be owned by the current user. Following are the mandatory fields required to be filled in by the user:

- Date
- Start Time
- Duration

On successful creation of a new reservation, the resourcePage.html is displayed with the message "The reservation has been made."

Please Note:

- The date is in the format mm/dd/yyyy. Currently, the reservation cannot be spanned over multiple days.

Validations:

- The Start Time and duration should be inputted in the 24-hour format (hours:minutes).
- The duration has a minimum value of 00:01, i.e., one minute.
- The Start Time and Duration is considered valid only if the entire length of the duration of the reservation falls within the available hours of that resource, i.e., the start and end time of the reservation should fall within the resource's available hours.
- The requested Date and Start Time should not be before the current time. Please note that since the seconds field is not taken into consideration in the comparison, the earliest possible booking for the current date is a minute after the current time. For instance, if the current time is 12:14 PM, a booking of 12:15 PM and beyond may be possible for today (provided the resource is available at that time).
- If the resource has already reached capacity for the requested time, the reservation will not be made.
- If the current user already has an overlapping reservation for the requested time, the reservation will not be made.

viewReservations.html

This page displays all upcoming reservations of a resource, along with their details – user, resource name, reservation date, reservation start time and reservation duration. This is sorted in the order of reservation date and start time. The user is a URL that links to the userPage. The resource name is a URL that links to the resourcePage of that resource. RSS is a URL that links to the RSSPage of the resource.

Please Note:

Throughout the application, I have not taken into consideration the seconds field when doing a comparison with the current time. Hence, it is not possible to view a booking that ends at the current time. For instance, Rit is not possible at 12:17 PM to view a reservation that ended at 12:17 PM.

Also, please note that the application functions as though its current time is EST.

User Page

This page displays the reservations made by and resources owned by the user (this may or may not be the current user). This page does not allow deleting of reservations. Deletion is possible only on the index.html page.

tagsPage.html

This page displays all resources that have been tagged by the given tag, along with their details. Each resource name links to the resource page of that resource. Each tag links to the tagPage for that tag. Each RSS links to the RSSPage of that resource.

RSSPage.html

This page displays the RSS, i.e., it generates the reservation details of all upcoming and past/undeleted reservations made by the given resource, sorted by reservation date and start time, in an XML format. Each reservation is an item in the XML. The description in each item describes details of the reservation – reservation date, start time and duration. It also indicates the owner of the reservation.

searchByAvailability.html

This page is used to search for resources given a date, start time and duration. Following are the mandatory fields required to be filled in by the user:

- Date
- Start Time
- Duration

Please Note:

- The date is in the format mm/dd/yyyy.

Validations:

- The Start Time and duration should be inputted in the 24-hour format (hours:minutes).
- The duration has a minimum value of 00:01, i.e., one minute.
- The requested Date and Start Time should not be before the current time. Please note that since the seconds field has not been taken into consideration in this application, for the current date, the earliest possible valid requested start time is one minute after the current time.
- If the resource has already reached capacity for the requested time, the resource is considered as unavailable.

searchResults.html

This page displays results of searches made by the current user – either search by name or by availability.

In case of search by name, all resources that have a resource name having an exact case-sensitive match to the given searched name are displayed. Please note that multiple resources can have the same name as it is not a unique field.

In case of search by availability, all resources that are available on the requested date during the requested hours, and have not reached capacity for that time are displayed in the searchResults.html.

Experiment:

I have turned off the send confirmation mail on booking of reservation feature here.

Extra Credit Done:

1. When showing a resource, shows how many times it has been reserved in the past.
2. Supports the ability to search for resources by name.

3. Supports the ability to search for resources by availability (user enters a time and duration, all resources available during those times are displayed).
4. Allows images to be uploaded for resources, as well as a description field to describe the resource. These can be viewed in the resource page of the resource (by clicking on the resource name URL on the landing page/index.html).
5. When reservations are made, an email confirmation is sent to the owner of the reservation.
6. When a reservation starts, an email confirmation is sent to the owner of the reservation. (Please note that when the quota limit for sending emails was reached, exceptions were being thrown. Thus, I added exception handling.)
7. Support a requirement where a user can only have one reservation at a time. The system does not allow a reservation to be made if it overlaps with another one made by the same user.
8. Support the ability to have more than one reservation at a time for a resource. This is done by supporting the notion of a *capacity* for a resource, which is how many reservations can be made at the same time for the resource. Thus, no more than *capacity* reservations exist at the same time for a resource.