

# Reservation Master Documentation

Zhenwei Gu

## Project URL:

<http://reservation-master.appspot.com/>

## Website Introduction:

### *Landing Page:*

The “Landing Page” contains 5 parts.

The 1<sup>st</sup> part is “Reservation for the logged in user”, in which user’s reservations are listed. For each reservation, the resource name (with a link to the resource page), the resource owner (with a link to the user page), the start time & the duration in minutes are listed. Under each reservation, there’s a button to cancel the above reservation.

The 2<sup>nd</sup> part is “Resources Owned by the logged in user”, in which user’s created resources are listed. For each resource, the resource name (with a link to the resource page), the available time for the resource & resource’s tags (each with a link to the tag page) are listed.

The first two parts can be seen only when the user is logged in, otherwise, the notice “Please Sign In to Make Reservations” is shown.

The 3<sup>rd</sup> part is “All Resources”, in which all the resources in the system are listed. For each resource, the resource name (with a link to the resource page), the resource owner (with a link to the user page), the available time for the resource & resource’s tags (each with a link to the tag page) are listed.

The 4<sup>th</sup> part is “All Tags”, in which all existing tags in the system are listed, each with a link to the tag page.

The 5<sup>th</sup> part contains 3 function buttons, used to create a new resource (link to create resource page), go back to homepage (the landing page) & sign in/out (link to login page).

### ***Create Resource Page:***

The “Create Resource Page” contains 2 parts.

The 1<sup>st</sup> part contains a text field to accept resource name, two datetime-local fields to accept start time and end time for resource’s availability, and a text field to accept the tags of the resource (strings splitted by “,”). Under them is a button to create the resource.

The 2<sup>nd</sup> part contains two function buttons to go back to land page & sign in/out separately.

### ***Resource Page:***

If the logged in user is the owner of the resource, the 1<sup>st</sup> part shows the owner (with a link to the owner’s user page) and the current tags (each with a link to the corresponding tag page). The 2<sup>nd</sup> part contains 4 fields to edit the resource’s information (filled by the current information of the resource by default), and a button to edit the resource.

If the logged in user is no the owner of the resource, the 1<sup>st</sup> part shows the owner (with a link to the owner’s user page), the start and end time of resource’s availability, and resources’ tags (each with a link to the corresponding tag page). The 2<sup>nd</sup> part is for the logged in user to reserve the resource, with a datetime-local field to accept reservation start time, and a text field to accept reservation duration in minutes. Below is a button to make reservations.

The 3<sup>rd</sup> part is the current reservations for the resource, shows resource name (with a link to resource page), resource owner (with a link to owner’s user page), resource reservoir (with a link to the reservoir’s user page), reservation start time & duration.

The 4<sup>th</sup> part contains 3 function buttons, each to generate RSS file for this resource (with a link to RSS page), to go back to land page & sign in/out.

### ***User Page:***

The “User Page” contains 3 parts.

The 1<sup>st</sup> part is this user’s current reservations, for each reservation shows the reserved resource name (link to resource page), owner of the resource (link to owner’s user link),

start time of reservation & duration in minutes.

The 2<sup>nd</sup> part is this user's owned resources, for each resource shows the resource name (link to resource page), availability time & resource tags (each with link to tag page).

The 3<sup>rd</sup> part contains two function buttons, each to go back to land page & sign in/out.

### ***Tag Page:***

The "Tag Page" contains 2 parts.

The 1<sup>st</sup> part lists all the resources with this tag, for each resource shows resource name (link to resource page), resource owner (link to owner's user page), resource availability time & all the tags of the resource (each with link to tag pag).

The 2<sup>nd</sup> part contains 2 function buttons each to go back to land page & sign in/out.

### ***RSS Page:***

"RSS Page" is reached when you press the "Generate RSS for Resource" button in the resource page. And it contains 2 parts.

The 1<sup>st</sup> part is the xml file generated for this resource. The xml file contains following information: resource name, resource owner, reservation user, reservation start time & reservation duration in minutes.

The 2<sup>nd</sup> part contains 2 function buttons each to go back to land page and log in/out.

## **Code Introduction:**

### ***app.yaml:***

Configuration file describing the handler scripts used for different URLs.

### ***backtohome.py:***

Every time the user wants to jump back to the land page, or the updated status needed to be shown, the redirect function in backtohome.py is called to query for all the resources & reservations, and to set necessary contexts.

***mainpage.py:***

Defines MainPageHandler, query for all the reservations and resources to be processed and shown in the land page.

***mainpage.html:***

Defines format of the main page, show reservations filtered by the logged in user for “reservations for user” part, show resources filtered by owner for “resource owned by user” part, show all the resources for “all resources” part, show all the tags for “all tags” part.

***create\_resources.py:***

The input information from create resource page is processed. Date and time are parsed into datetime object. Necessary contexts are stored for showing resource creation results. If a new resource is created successfully, it will be stored into the resources\_datastore. If not, the corresponding error information will be shown on the resource page.

***create\_resource.html:***

Defines format of the create resource page.

***login.py:***

Decides whether user is logged in or not, showing corresponding content on the sign in/out button.

***resource.py:***

Defines resourcePageHandler function, which tries to create a new reservation and add to the reservation datastore on the clicked resource if the current user is not the owner of the resource, and to edit this resource and modify the resource datastore if the current user is the owner of the resource.

***resource.html:***

Defines format of resource page, show resource reservation interface if the current user is not the owner of the resource, show resource editing interface if the current user is the owner of the resource.

***user.py:***

Defines UserPageHandler function. The current user, all reservations and resources are stored as contexts for user page display.

***user.html:***

Defines format of the user page. Resources and reservations are filtered by current user to display the reservations and resources belong to the current user.

***tag.py:***

Defines TagPageHandler function. Find the resources that has the clicked tag.

***tag.html:***

Defines format of the tag page. Show the resources that contains the clicked tag.

***rss.py:***

Defines XMLPageHandler function. Select the reservations that are made on the current resource.

***rss.html:***

Defines format of the rss page. Show the formatted xml file for reservations on the current resource.

***reservations\_datastore.py:***

Defines the informations stored for each reservation, including reservation time, reservationGUID, reservationOwner (email, name & ID), reservationStartTime, reservationStartTime, reservationDuration, resourceGUID, resourceOwner &

resourceName.

***resources\_datastore.py:***

Defines the informations stored for each resource, including resourceGUID, resourceOwner, resourceEmail, resourceName, resourceStartTime (raw input & processed output), resourceEndTime (raw input & processed output) and resourceTags.

***.pyc files:***

byte codes for the .py files.