MSC – CC – Assignment 02 - Room Scheduler

Brandon Rozario

Student No: 2955533

1. UI

- General UI Elements:
  -A theme is maintained throughout the app by using consistent colour scheme.
  - The navbar on each page is a black bar to separate it from the body.
  -The body background is either gradient blue or light orange, and usually changes to indicate change of page.
  - Accent colour is dark orange. It is used when hovering over navbar buttons, and is the same colour used for buttons.
  -The buttons change colour when they are hovered over, so they appear interactive to the user, letting them know that they can be clicked on.
  - Notes, Hints, or Error Messages are displayed at the bottom of the pages as required.
  - Error messages are written in Bold and red. They also have an animation that makes them appear to vibrate to get the user's attention.

- Addbooking.html
  - This page provides the GUI for adding a new booking to a new room
  - It displays the **current time in UTC** for standardization and simplification.
  - It contains a warning stating that no bookings can be made prior to this time.
  - The Labels and Text inputs are displayed using a borderless table so that they are vertically aligned.
  - The left column (input labels) are right aligned and the right column (input fields) are left aligned to minimize the distance between the two.
  - Placeholders indicate the format in which the date and time inputs are expected

- Addroom.html
  - The Labels and Text inputs are displayed using a borderless table so that they are vertically aligned.
  - The left column (input labels) are right aligned and the right column (input fields) are left aligned to minimize the distance between the two.

- Displayeach.html
  - This page is used to display the individual bookings in a room.
  - A table is used to display the data in an organized manner
  - The background of the table is set to light orange (same colour used as background colour on other pages) for aesthetic purposes and to make it clear where the table starts and ends.

- The last column containing a delete button is added to the table so that each button is clearly associated with each row.
- The 'Back' and 'Add Booking' buttons are center-aligned so that they stay in the same place irrespective of the change in width of the table, and so that they are clearly visible to the user.
- All bookings are displayed in <u>ascending order of start time</u>.

- Displayrooms.html
    - This is the first page a user sees when he logs in to the application. It displays the list of rooms in the application.
    - Once again, a table is used to display data in an organized manner.
    - The Room Name of each room is formatted to look like an HTML Link so that the user knows that it is clickable.
    - The cursor changes to a hand when the user hovers over it.
    - The current time is displayed in the top left-hand corner so that it is the first piece of information the user sees.
    - It is displayed for the convenience of the user in case the user's machine is set to any time other that UTC.
    - The "Expired bookings auto deleted" counter is placed in the top-right hand corner as an easy indicator of when the special feature is working.
    - A note with a refresh link is given at the bottom of the table in case the user reads through the table and finds that the changes are not yet reflected.
    - This page also displays error messages in bold red letters to capture the attention of the user. For example: when a user tries to delete a room that has existing bookings in it.
    - All rooms are displayed in <u>ascending order of unique room name.</u>

- Filterbydate.html
    - This page too displays the current time in the top-left hand corner for the user's easy reference.
    - As with the previous page, a table with a white background is used to organize and display data.

2. Methods and Modules
- Addbooking.py
    - This file handles requests to add new bookings to a particular room.
    - It receives the Id of the room to which the booking is to be added, and various information about the room such as Unique Booking Reference, Start date, and End date.
    - DateTime objects are used instead of separate date and time objects since a single object is easier to manage and compare.
    - Various validations are done on the start and end times of the bookings before accepting them namely:
        - ❖ Is the start-date of the booking after the current time and date

- ❖ Is the end-date after the start date
- ❖ Is the end-date same as the start date
- ❖ If a booking within that period already exists

- **Note: The exact start time of one booking cannot be the same as the end time of the next booking. For example: If one booking ends at 01-01-2018 05:00, the next booking cannot start at 05:00. It will have to start from 01-01-2018 05:01 onwards.**

- If any of the above conditions are not satisfied, then the appropriate error message is displayed.

- Addroom.py
  - This file is used to add new rooms to the datastore.
  - It accepts the name of the room and checks if a room with the same name exists in already. If it does, then an error is displayed on the webpage to the user.

  - **Note: All names (ROOM NAMES as well as BOOKING NAMES) are not case-sensitive. All names are converted to lower-case before they are stored to avoid confusion.**

- Displayeach.py
  - This file is used to display the bookings or a particular room.
  - It uses the unique room name to extract bookings of the room, and then passes the list of bookings to HTML via jinja
  - It also handles the deletion of the bookings.
  - It uses the unique booking reference of the booking to delete it from the ndb datastore.

- Displayrooms.py
  - This file displays the list of all rooms on the home page.
  - It simply queries the datastore and returns the names of each rooms.
  - It also handles the deletion of rooms.
  - If a room still contains bookings, it passes an error to the HTML page via Jinja.

- Filterbydate.py
  - It displays information of all rooms and bookings that are booked during a particular day.
  - Since booking start and end dates are stored as datetime objects in ndb, it first converts them into date() objects and then checks the booking period of each booking of each room.
  - It then sorts the bookings in ascending order of start date and displays the resulting list to the user.

- Login.py
  - It checks whether a user is already logged in.
  - If it is, it forwards the user to the /displayrooms page

- Else, the user is displayed the login page to the application.

- **Booking.py**
  - This is the ndb model used for storing the bookings
  - It contains a unique booking reference to identify each booking of a room.
  - It contains a start date and end date to mark the booking period of each booking.
  - The reference is a String property
  - The startdate and enddate are DateTime() properties so that the date and the time can be stored together making it easier for comparison operations.

- **Room.py**
  - This is the ndb model used to store individual rooms.
  - It contains a repeated structured property 'bookings'
  - Each booking is stored as an item in this list.
  - Each room will hence have its own individual list of bookings.
  - The unique name of each room is stored as the ID in the datastore, and hence is not included in this model.

Extra Feature: Auto Delete Elapsed Bookings.

- This feature deletes elapsed bookings from the datastore.
- It does this whenever the home page ('/displayrooms') is loaded

1. **UI**
   - The top right-hand corner of the Home Page ('/displayrooms') contains a counter of the number of bookings that were auto-deleted.
   - Whenever the homepage is loaded, the counter is refreshed.
   - The home page also displays the time at which the home page was loaded for easy reference.
   - Once the end time of a booking has elapsed, that booking is deleted from the system, and the counter is updated.

2. **Methods and Working**
   - When the home page is loaded, the timestamp for the end of every booking of every room is checked against the current time i.e. the time at which the page was loaded.
   - If any of the bookings has an end time lower than that of the current time, it is deleted from the datastore.
   - Once this is completed, the list of all rooms is sorted alphabetically and displayed in a table.
   - The counter of the number of deleted bookings is also displayed above the table.