Desk Reservation

By Jeffrey Zhou

Project Description

Verteo's Tip Top Tupperware is used to store syringes. Our employees work from two offices. Desks in offices 1 and 2 are occupied by traveling salespeople who are rarely in the office. To be more efficient, we are instituting hotelling and employees will use Salesforce to reserve their desks when they need them.

Please Note: Our salespeople are very social and like to pick desks to sit near their friends.

Synopsis

For our demo today, we will have 2 offices(Office 1 and Office 2). For each office, there will be 2 floors and each floor will have 16 desks; making a total of 32(64 overall) desks per office. The Desk Reservation page will display the desks in a 4x4 structure per floor. The page will only be accessible to users with System Administrator or Sales profile. Each user will only be able to reserve one desk for the selected date. The user will be able to select more dates in the future, but one reservation per date. The user can only reserve a desk for him/herself.

Happy Path - The System Admin or Sales user will go into Salesforce and click on the Desk Reservation tab. This will direct the user to the Desk Reservation page. The user will be prompted to select a desired office on the dropdown, select a desired date on the datepicker, and click on the Submit button:

Once that is done, the user will then see all the available desks listed on the page:

Available desk(Green): As pictured above

Example unavailable desk(Red):
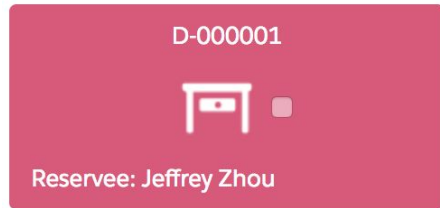
The user selects the desired desk to reserve by clicking on the checkbox associated to the desk and click on the Reserve button. This will create a reservation record for the selected desk for the user. The reservation record will have information of the desired date and associated desk. The page will be reloaded and the desk that the user selected is now unavailable with the user's name as the reservee. A table section will appear at the bottom of the page with rows of all current(present or future) reservation records. The new reservation will be listed as a row on this table as well:

Current Reservations

	NAME	DATE	DESK	OFFICE	FLOOR #	OWNER
☐	D-000001 - 11/14/2018	11/14/2018	D-000001	Office 1	1	Jeffrey Zhou
☐	D-000026 - 11/15/2018	11/15/2018	D-000026	Office 1	2	Jeffrey Zhou
☐	D-000034 - 11/16/2018	11/16/2018	D-000034	Office 2	1	Jeffrey Zhou
☐	D-000062 - 11/17/2018	11/17/2018	D-000062	Office 2	2	Jeffrey Zhou

Cancel Reservations

Only the owners can cancel their own Reservations or if the user is a System Administrator, the user can cancel all Reservations. The checkboxes will not be displayed if either cases are not satisfied. This is an example running as the user Larry Jovanovic in Salesforce:

Current Reservations

	NAME	DATE	DESK	OFFICE	FLOOR #	OWNER
☑	D-000001 - 11/14/2018	11/14/2018	D-000001	Office 1	1	Jeffrey Zhou
	D-000002 - 11/14/2018	11/14/2018	D-000002	Office 1	1	Desy Velcheva
	D-000003 - 11/14/2018	11/14/2018	D-000003	Office 1	1	Brian Aggen
☑	D-000004 - 11/14/2018	11/14/2018	D-000004	Office 1	1	Larry Jovanovic
	D-000026 - 11/15/2018	11/15/2018	D-000026	Office 1	2	Jeffrey Zhou
	D-000034 - 11/16/2018	11/16/2018	D-000034	Office 2	1	Jeffrey Zhou
	D-000062 - 11/17/2018	11/17/2018	D-000062	Office 2	2	Jeffrey Zhou
☑	D-000018 - 11/22/2018	11/22/2018	D-000018	Office 1	2	Larry Jovanovic
☑	D-000015 - 11/30/2018	11/30/2018	D-000015	Office 1	1	Larry Jovanovic

Cancel Reservations

<u>Sad Paths</u> - Please see the cases under Validations section.

Data Model
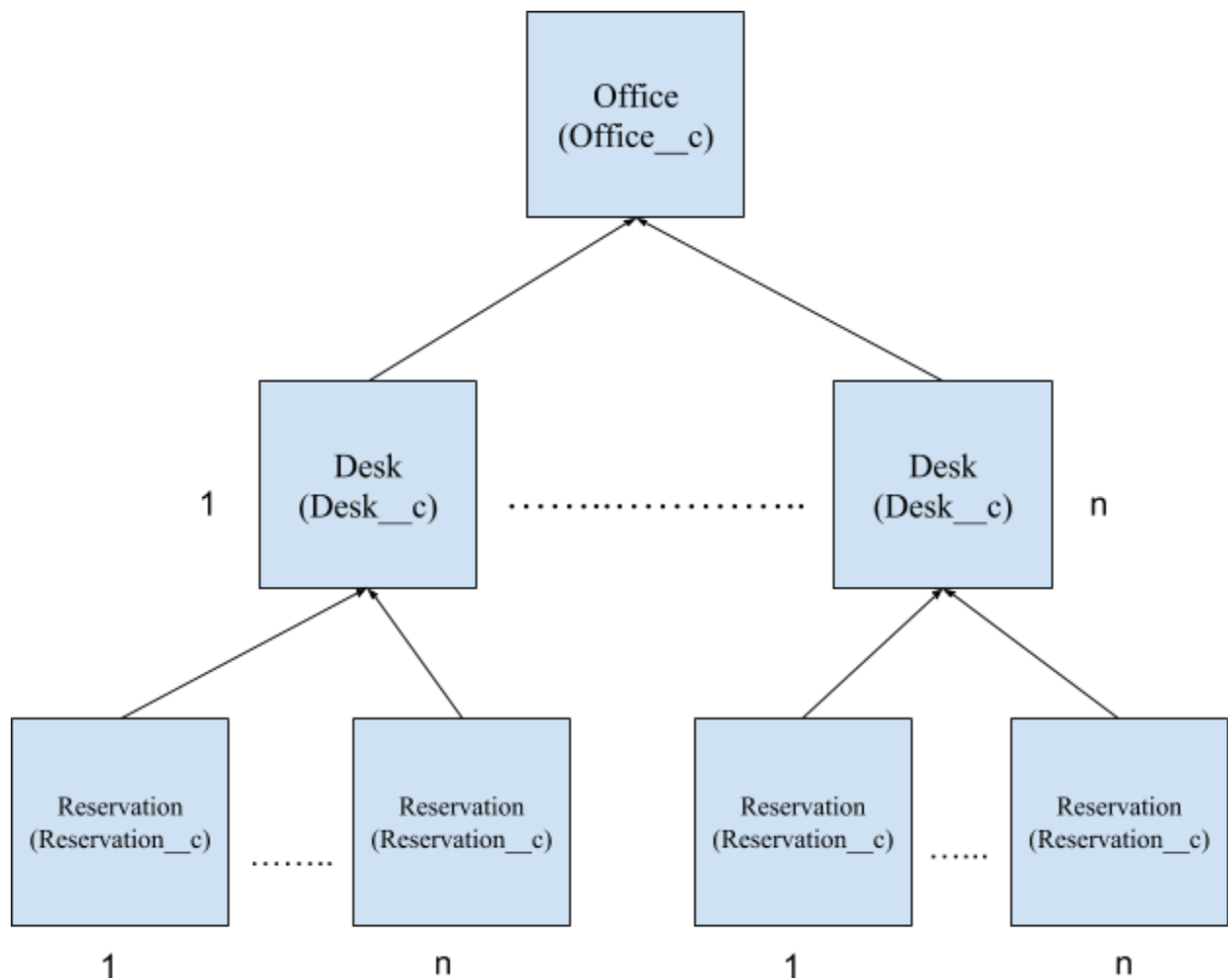
1. **Office**

--Object--
    <u>Label</u>: Office

API: Office__c

--*Fields*--
1. <u>Label</u>: Office Name
   <u>API</u>: Name
   <u>Data Type</u>: Text(80)
   <u>Required</u>: True
2. <u>Label</u>: Number of Floors
   <u>API</u>: Number_of_Floors__c
   <u>Data Type</u>: Number(18, 0)
   <u>Required</u>: True

2. *Desk*

--*Object*--
<u>Label</u>: Desk
<u>API</u>: Desk__c

--*Fields*--
1. <u>Label</u>: Desk Name
   <u>API</u>: Name
   <u>Data Type</u>: Auto Number(D-{000000})
   <u>Required</u>: True
2. <u>Label</u>: Office
   <u>API</u>: Office__c
   <u>Data Type</u>: Lookup(Office)
   <u>Required</u>: False
3. <u>Label</u>: Floor Number
   <u>API</u>: Floor_Number__c
   <u>Data Type</u>: Number(18, 0)
   <u>Required</u>: True

3. *Reservation*

--*Object*--
<u>Label</u>: Reservation
<u>API</u>: Reservation__c

--*Fields*--

1. Label: Desk Name
   API: Name
   Data Type: Text(80)
   Required: True
2. Label: Desk
   API: Desk__c
   Data Type: Lookup(Desk)
   Required: True
3. Label: Date
   API: Date__c
   Data Type: Date
   Required: True

Validations

Validation Rules on Reservation
Case: If the user tries to change the date on the reservation in Salesforce.
Rule Name - Cannot_Change_Date
Error Condition Formula - ISCHANGED(Date__c)
Error Message - You cannot change the date on a reservation. Please cancel this reservation and create a new reservation with your desired date.

Case: If the user tries to change the desk lookup on a reservation in Salesforce.
Rule Name - Cannot_Change_Desk_Lookup
Error Condition Formula - ISCHANGED(Desk__c)
Error Message - You cannot change desk look up on a reservation. Please delete this reservation and create a new reservation through the Desk Reservation page.

Case: If the user selects a date in the past for the reservation in Salesforce.
Rule Name - Cannot_Select_Date_In_The_Past
Error Condition Formula - Date__c < TODAY()
Error Message - You cannot select a date in the past. Please select a present or a future date.

Validation on the Desk Reservation page(in DeskReservationController)
Case: If the user selects nothing for date and clicks on the Submit button
Error Message - Please select a date.

Case: If the user selects a date in the past for the reservation.
Error Message - Please select a present or future date.

Case: If there no available desks for the selected office and the selected date.
Info Message - There are currently no desks available for this office and this date. Please try a different office or a different date.

Case: If the user currently have a reservation for the selected date.
Info Message - You currently have a reservation for this date. To change your reservation, please cancel your current reservation and make a new reservation.

Case: If no desk has been selected and the user clicks on the Reserve button.
Error Message - Please select a desk to reserve.

Case: If there are no reservations selected and the user clicks on the Cancel Reservations button.
Error Message - Please select at least one reservation to cancel.

Case: If there are no Office records in Salesforce
Error Message - No office records were found in Salesforce.

Validation in ReservationTrigger(logic in ReservationService)
Case: If the user tries to insert or update a batch of reservations where the reservations in the batch have the same date.
Error Message - Duplicate reservation records with the same date(' + {reservation date} + ') were found in insert/update list.

Case: If the user already have a reservation for a specific date and the user tries to insert another reservation with the same date or update a reservation to the same date.
Error Message - You already have a reservation on this date(' + {reservation date} + '). Please select another date.

Case: If there is already an existing reservation for the date on the reservation that the user is trying to create.
Error Message - There is already a reservation for this desk on this date(' + {reservation date} + '). Please select another date.

Workflow

Workflow Rule

Rule Name: Generate Reservation Name Workflow

Object: Reservation

Description: Workflow to generate Reservation Name on create or edit.

Evaluation Criteria: Evaluate the rule when a record is created, and every time it's edited

Rule Criteria - OR(ISNEW(),ISCHANGED(Name))

Workflow Field Update

Name: Generate Reservation Name Workflow

Unique Name - Generate_Reservation_Name

Object - Reservation

Field to Update - Reservation: Reservation Name

Field Data Type - Text

Formula Value - Desk__r.Name + " - " + TEXT(MONTH(Date__c)) + '/' + TEXT(DAY(Date__c)) + '/' + TEXT(YEAR(Date__c))

Things to Consider

What should happen to existing reservations with past dates?

Solution 1: Create a batch class to query for all existing reservations with past dates and delete them. We can schedule this batch class to run once a month, once a week, every day, etc.

Solution 2: Keep the existing reservations for record.

Source Code

https://github.com/Jzhou1991/Desk-Reservation