RealOffice System Modeling Document

CS14B023, Rahul Kejriwal CS14B007, Suhas CS14B045, Malireddi Sunil Kumar CS09B043, Shanker Lal Sharma CS09B031, Chandrakanth

Table Of Contents

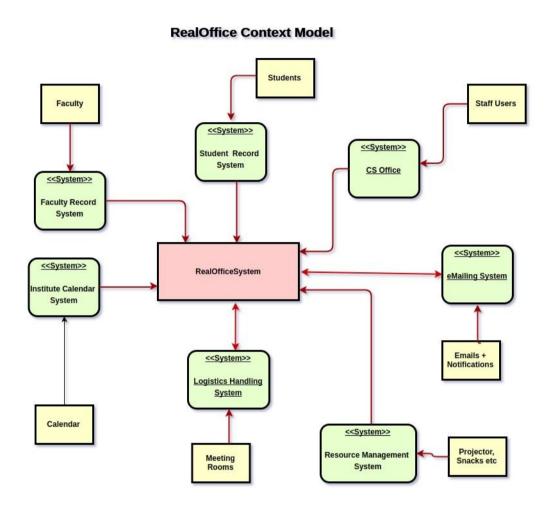
Table Of Contents	1
Introduction	2
Context Models	3
Context Of the RealOffice System	3
Structural Models	4
Class Association Diagram	4
Class Diagrams	4
Behavioral Models	5
Activity Diagrams (data-driven modeling)	5
Interaction Models	10
Sequence Diagrams	10

Introduction

RealOffice is an application software intended for use in the CSE department office to simplify routine procedural tasks. In essence, it shall allow office staff to schedule different kinds of meetings, track meeting requirements, manage room allotments, file meeting reports and also cancel arrangements for cancelled meetings. It shall also remind the staff regarding upcoming events that need attention and facilitate routine backups as replacement for hard paperwork. It shall also integrate data from the CSE department calendar about meetings scheduled from other platforms.

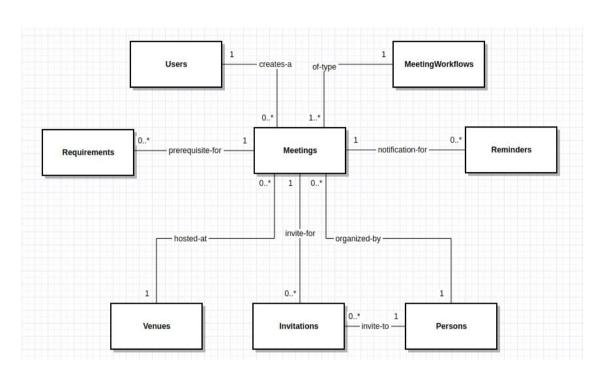
Context Models

1. Context Of the RealOffice System



Structural Models

1. Class Association Diagram



2. Class Diagrams

Users

userid : String type : String password : Hash

login() logout()

Meetings

name: String stime: DateTime etime: DateTime status: STATUS createdBy: UserId hostedAt: VenueId organizedBy: PersonId ofType: MeetingWorkflowId

reschedule() addRequirements() addReminders()

MeetingWorkflows

meetingType : String actionList : String[]

modifyActions()

Requirements

item : String qty : int cost : int

orderDetails : String isApproved : Boolean prerequisiteFor : MeetingId

approveRequirement() addOrderDetails()

Reminders

purpose : String sendDateTime : DateTime recipient : PersonId/UserId recipientType : String notificationFor : MeetingId

dispatchRequirement()

Invitations

invitationMeeting : MeetingId invitedPerson: PersonId isAttending : Boolean

changeAttendingStatus()

Persons

name : String email : String

Venues

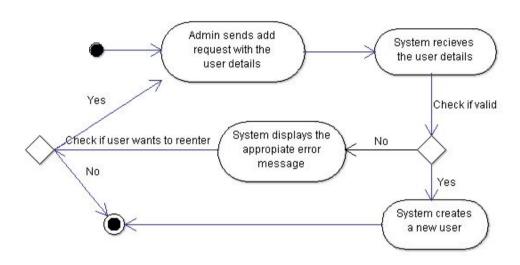
infrastructure : String[] room : String capacity : int

modifyInfrastructure() changeCapacity()

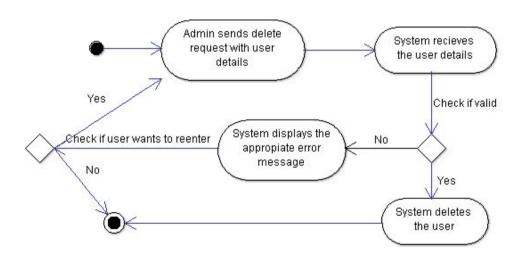
Behavioral Models

Activity Diagrams (data-driven modeling)

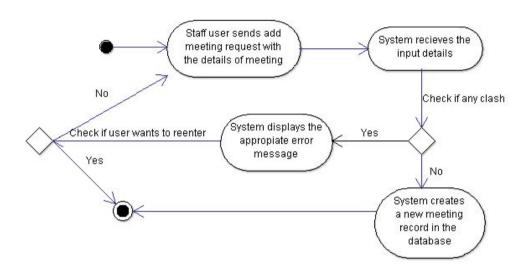
1. Admin Scenario 1: Add User



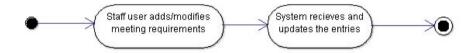
2. Admin Scenario 2: Delete User



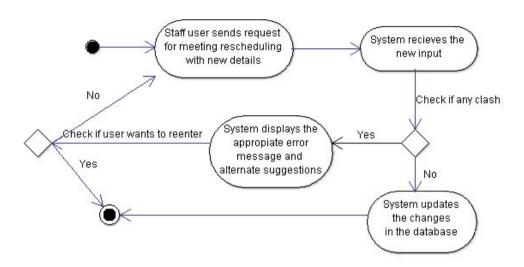
3. Staff User Scenario 1: Add Meeting



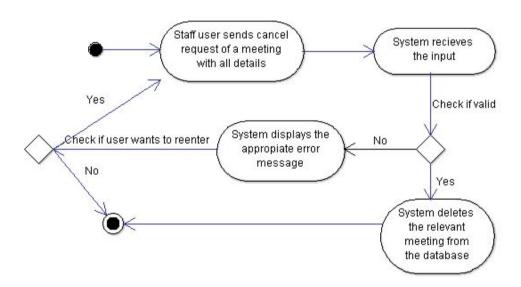
4. Staff User Scenario 2: Add Meeting Requirements



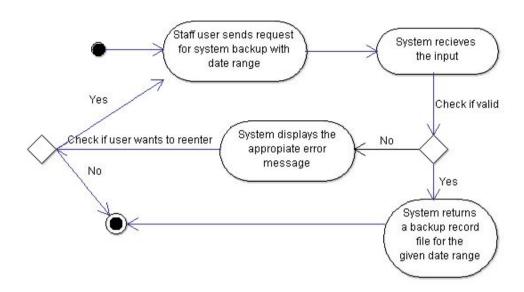
5. Staff User Scenario 3: Reschedule Meeting



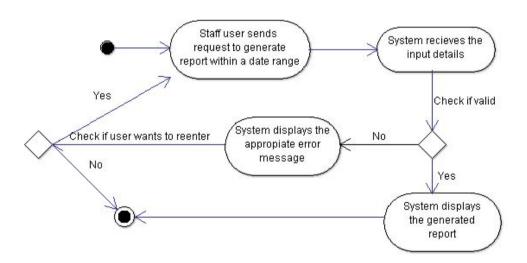
6. Staff User Scenario 4: Cancel Meeting



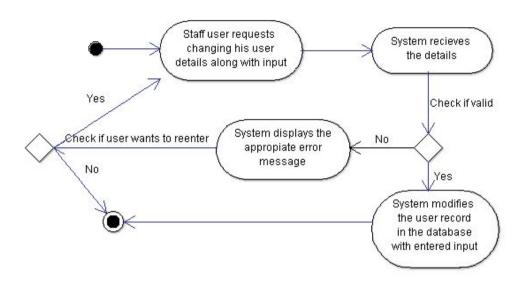
7. Staff User Scenario 5 : Generate Backup



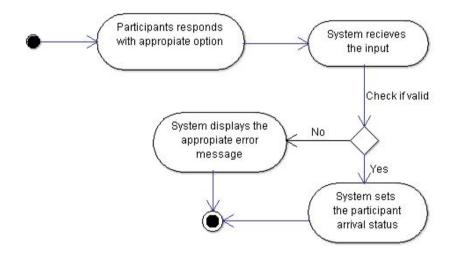
8. Staff User Scenario 6 : Generate Report



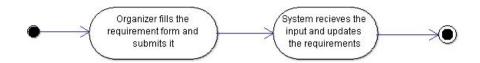
9. Staff User Scenario 7: Edit own user details



10. Participant Scenario 1: RSVP Remainders



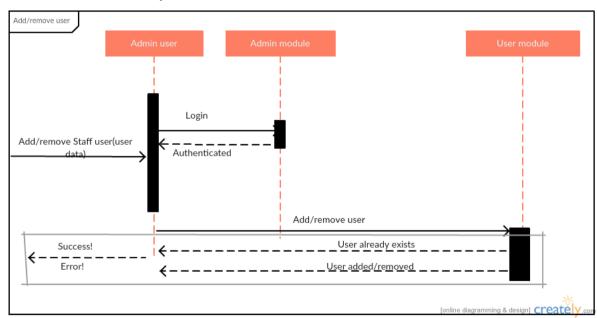
11. Organizer Scenario 1: Request Requirements



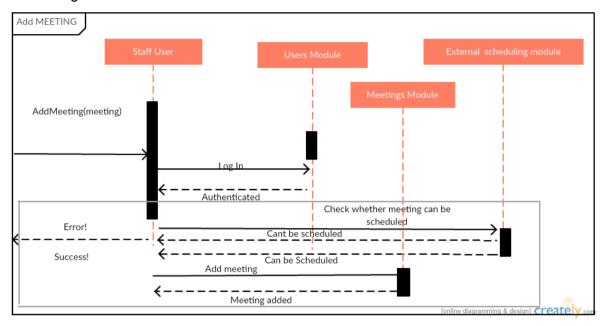
Interaction Models

Sequence Diagrams

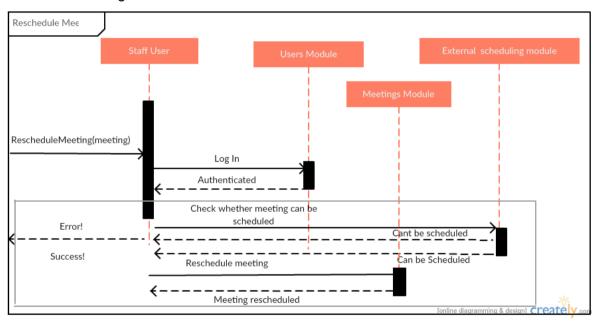
1) Add/remove staff users by admin users



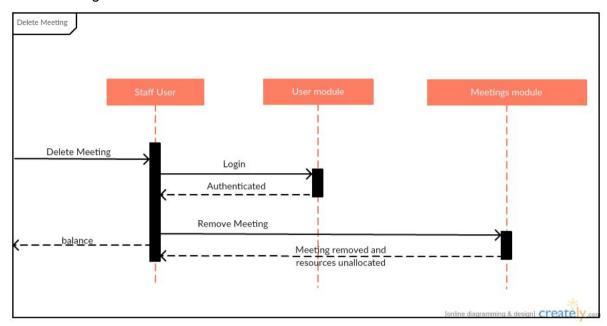
2) Add Meeting



3) Reschedule meeting



4) Delete meeting



5) Get meeting reports between 2 dates given

