

Personal Calendar

Team 6: Iteration 1

William Hilliard, Caesar Llaga, Cyril Varughese, Peggy Soh

Requirements

Requirements

1. The calendar shall allow the user to switch between monthly, weekly and daily views.
 - 1.1. The calendar shall allow the user to view past events.
 - 1.1.1. The current date shall be highlighted.
 - 1.2. The monthly view shall have a change month option.
 - 1.3. The weekly view shall have a change week option.
 - 1.4. The daily view shall have a change day option.
 - 1.5. The views shall allow user to quickly go to today.
2. The calendar shall track and store events.
 - 2.1. The calendar shall have a summary view.
 - 2.1.1. The summary view shall have an option to show the next ten events.
 - 2.1.2. The summary view shall have an option to show all previous events.

Requirements Continue

- 2.2.** The calendar shall allow the user to enter an event.
 - 2.2.1.** The calendar shall not allow an event to be added if it conflicts with a preexisting event.
 - 2.2.2.** The calendar shall allow an event to be an all-day event.
 - 2.2.3.** The calendar shall allow an event to have a start and end time.
 - 2.2.4.** The calendar shall allow an event to have an optional description.
 - 2.2.5.** The calendar shall allow an event to have an optional location.
 - 2.2.6.** The calendar shall allow the user to schedule repeating events.
 - 2.2.7.** The calendar shall allow the user to enter emails of participants for an event.
- 2.3.** The calendar shall allow the user to remove an event.
- 2.4.** The calendar shall allow the user to edit an event.
- 2.5.** The calendar shall allow the user to set reminder alerts.
 - 2.5.1.** If specified by the user, the calendar shall send an email to event participants.
 - 2.5.2.** If specified by the user, the calendar shall send an email to the user.

Requirements Continue

- 3. The calendar shall allow participants.
 - 3.1. The calendar shall allow users to add contacts to be used as participants.
 - 3.2. The calendar shall allow users to delete participants.
 - 3.3. The calendar shall allow users edit participants' information.
- 4. The calendar shall allow users to customize their calendars with different settings.
 - 4.1. The settings shall allow users to change and personalize the calendar.

Domain Modeling

Brainstorming

1. ● Noun/ Noun Phrases
2. ● X of Y
3. ● Transitive Verbs
4. ● Adjectives
5. ● Numerics/ Enumerations
6. ● Possession Expressions
7. ● Constituents
8. ● Containment
9. ● X is a Y

Brainstorming Continue

Personal Calendar Requirements

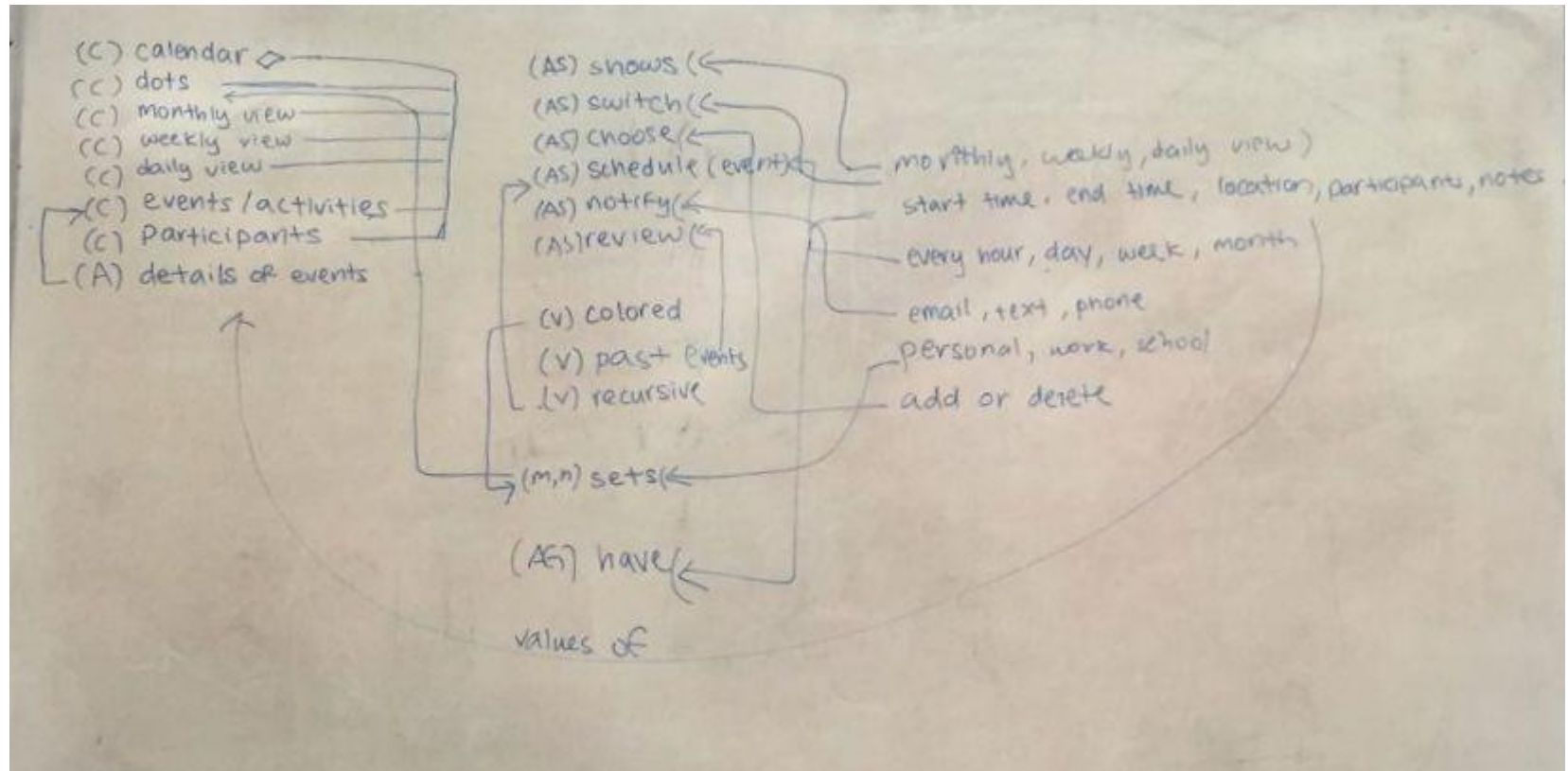
The calendar has a monthly, weekly and daily view. The monthly view shows colored dots that represent different sets such as personal events, school events and work events. The weekly view shows general details of events such as time and event. The daily view shows details of events for the day such as location, time, notes, etc. The user will be able to switch between the three different views. The user can choose to add and delete events. The events will have information such as start time, end time, location, participants and other notes. If the user chooses, he can add participants to an event and choose to have the calendar notify the participants by emailing them a reminder.

Brainstorming Continue

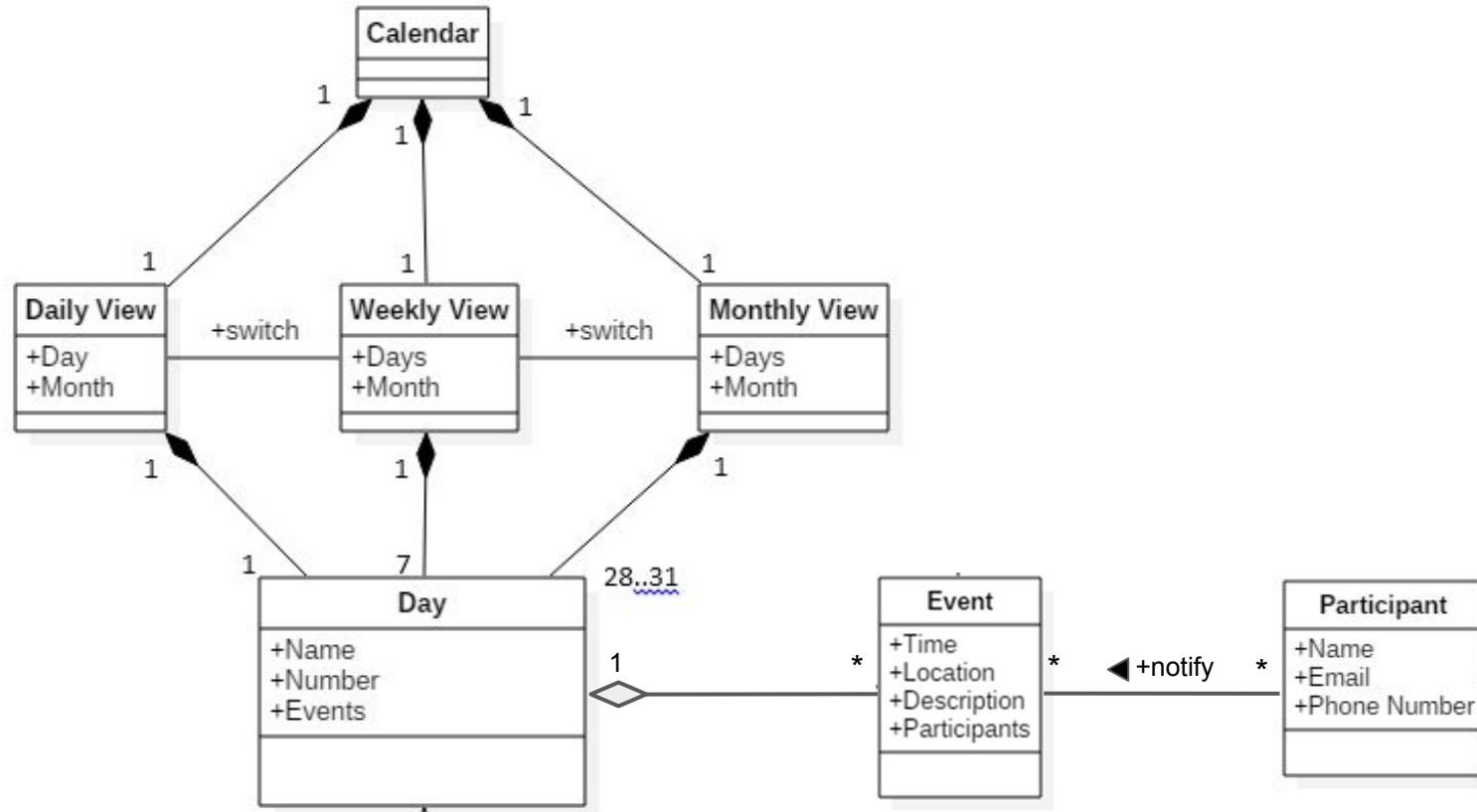
Exercise 4.4

This calendar management software^① allows the user to schedule personal activities^④ such as meetings and tasks to be performed. An activity can take place on a future date during a certain period of time. An activity can take place for several consecutive days. Each activity has a brief mnemonic description. An activity can be a recursive activity^④, which takes place repeatedly every hour, every day, every week, or every month. A user can schedule an activity^③ using a month-by-month calendar to select the date or dates, and then zooms in to select the begin time and end time on a date. The calendar system shall notify the user^③ by email, text message, or phone call the day before and on the activity day. The user can review^③ past activities^④ and modify the schedule including updating and deleting activities.

Classification



Domain Model Class Diagram



Use Case Modeling

Deriving Use Cases

1. The calendar shall allow the user to switch between monthly, weekly and daily views.
 - 1.1. The calendar shall allow the user to view past events.
 - 1.1.1. The current date shall be highlighted.
 - 1.2. The monthly view shall have a change month option.
 - 1.3. The weekly view shall have a change week option.
 - 1.4. The daily view shall have a change day option.
 - 1.5. The views shall allow user to quickly go to today.
2. The calendar shall have color sets.
 - 2.1. The user shall be able to create a new labeled set to contain specific events.
 - 2.2. The user shall be able to delete a labeled set.
 - 2.3. The user shall be able to choose a color for a particular set.
 - 2.4. The user shall be able to share the sets with participants.
 - 2.5. The user shall be able to remove participants from the set.
3. The calendar shall track and store events.
 - 3.1. The calendar shall have a summary view.
 - 3.1.1. The summary view shall have an option to show the next ten events.
 - 3.1.2. The summary view shall have an option to show all previous events.

Deriving Use Cases Continued

Verb-Noun Phrase	Business process?	Begin with actor?	End with actor?	Accomplish business task for actor?	Use case?	Actor	Subsystem
switch views	N	Y	Y	N	N	N/A	N/A
view past events	N	Y	Y	N	N	N/A	N/A
date highlighted	N	N	N	N	N	N/A	N/A
change month	N	Y	Y	N	N	N/A	N/A
change week	N	Y	Y	N	N	N/A	N/A
change day	N	Y	Y	N	N	N/A	N/A
go to today	N	Y	Y	N	N	N/A	N/A
share events	Y	Y	Y	Y	Y	User	Event Tracker
remove participants	N	N	N	N	N	N/A	N/A
track and store events	N	N	N	N	N	N/A	N/A
have a summary view	N	N	N	N	N	N/A	N/A
show the next ten events	N	Y	Y	N	N	N/A	N/A
show all previous events	N	Y	Y	N	N	N/A	N/A
enter an event	Y	Y	Y	Y	Y	User	Event Editor
be an all-day	N	N	N	N	N	N/A	N/A

Abstract Use Cases

- UC1.** Share event (Actor: User, System: Personal Calendar)
- UC2.** Enter an event (Actor: User, System: Personal Calendar)
- UC3.** Edit an event (Actor: User, System: Personal Calendar)
- UC4.** Remove an event (Actor: User, System: Personal Calendar)
- UC5.** Add contacts (Actor: User, System: Personal Calendar)
- UC6.** Delete contacts (Actor: User, System: Personal Calendar)
- UC7.** Edit contacts (Actor: User, System: Personal Calendar)
- UC8.** Customize calendar (Actor: User, System: Personal Calendar)

High-Level Use Cases

UC1. Share Event

TUCBW the user clicks on “Share” for an existing event.

TUCEW the user sees the calendar email the event to the participants.

UC2. Enter Event

TUCBW the user clicks on “New Event” button in the event tracker.

TUCEW the user sees the newly added event on the calendar.

UC3. Edit Event

TUCBW the user clicks “Edit” in the event detail window.

TUCEW the user sees the event details changed.

UC4. Remove Event

TUCBW the user clicks “Delete Event”.

TUCEW the user sees the event is removed from the calendar.

High-Level Use Cases Continued

UC5. Add Contact

TUCBW the user clicks “Add Contact”.

TUCEW the user sees the contact added to the list.

UC6. Delete Contact

TUCBW the user clicks “Delete Contact”.

TUCEW the user sees the contact is removed from the list.

UC7. Edit contacts

TUCBW the user clicks “Edit Contact”.

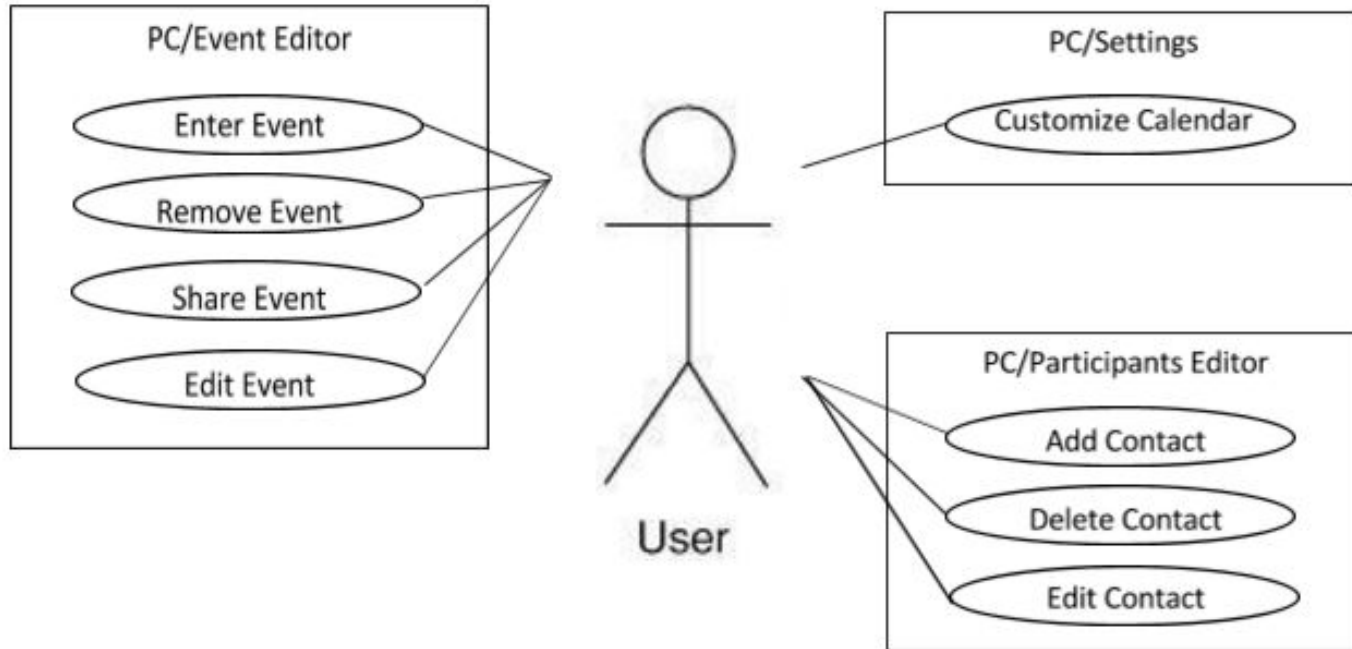
TUCEW the user sees the contact information has been updated.

UC8. Customize Calendar

TUCBW the user clicks “Settings”.

TUCEW the user sees the settings menu.

Use Case Diagram



Expanded Use Case (Add Event)

Precondition: None	
Actor: User	System: Personal Calendar
	0. System displays a '+' button on the upper right hand corner of the view window.
1. TUCBW user clicks on the '+' button.	2. System displays a create event form for the event name, start time, end time, description, location, and participants.
3. User enters event name, start time, end time, description, location, and participants and clicks the 'Confirm' button.	4. System saves the entered fields into the storage and displays a message confirming that event has been saved, or else the system displays a schedule full message.
5. TUCEW user sees the confirmation message.	
Post condition: None	

Expanded Use Case (Share Event)

Precondition: User must be in event detail view and there must be participant(s) in the event.	
Actor: User	System: Personal Calendar
	0. System displays a share button on the event details pop up window.
1. TUCBW user clicks on the share button.	2. System displays a message asking for confirmation to share event with all participants.
3. User clicks the 'Confirm' button on the message.	4. System sends email with event details (event name, start time, end time, location and description) to each participant and displays a message confirming that the email(s) has been sent.
5. TUCEW user sees the confirmation message.	
Post condition: None	

Requirements - Use Cases

Requirement	Priority	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8
R01	1								X
R02	2		X	X	X				
R03	3	X				X	X	X	
R04	4								X
UC Priority		5	1	2	3	4	7	6	8

Object Interaction Modeling

Scenarios (Add Event)

UC1: Add Event

1. User clicks '+' button to add new event.
2. Event creator provides user with an event form to fill out.
3. *User enters event name, date, start-time, end-time, location, participants and clicks the confirm button.
4. Event creator saves event.
 - 4.1. Event creator searches for available time slot in calendar storage.
 - 4.2. If storage has availability during given time frame
 - 4.2.1. Event creator saves newly entered data in calendar storage.
 - 4.2.2. Event form will display a confirmation message.
 - 4.3. Else
 - 4.3.1. Event form will display a "Schedule full" message.

Scenarios (Share Event)

UC2: Share Event

1. *User selects an event and its participants and clicks the “share” button.
2. Event controller sends email to participants
 - 2.1. Event controller fetches event name, date, start-time, end-time, location and participants.
 - 2.2. Event controller gathers emails of participants.
 - 2.3. Event controller sends event name, date, start-time, end-time and location to all participants.
 - 2.4. If email is send
 - 2.4.1. View window displays a confirmation message.
 - 2.5. Else
 - 2.5.1. View window displays an error message.

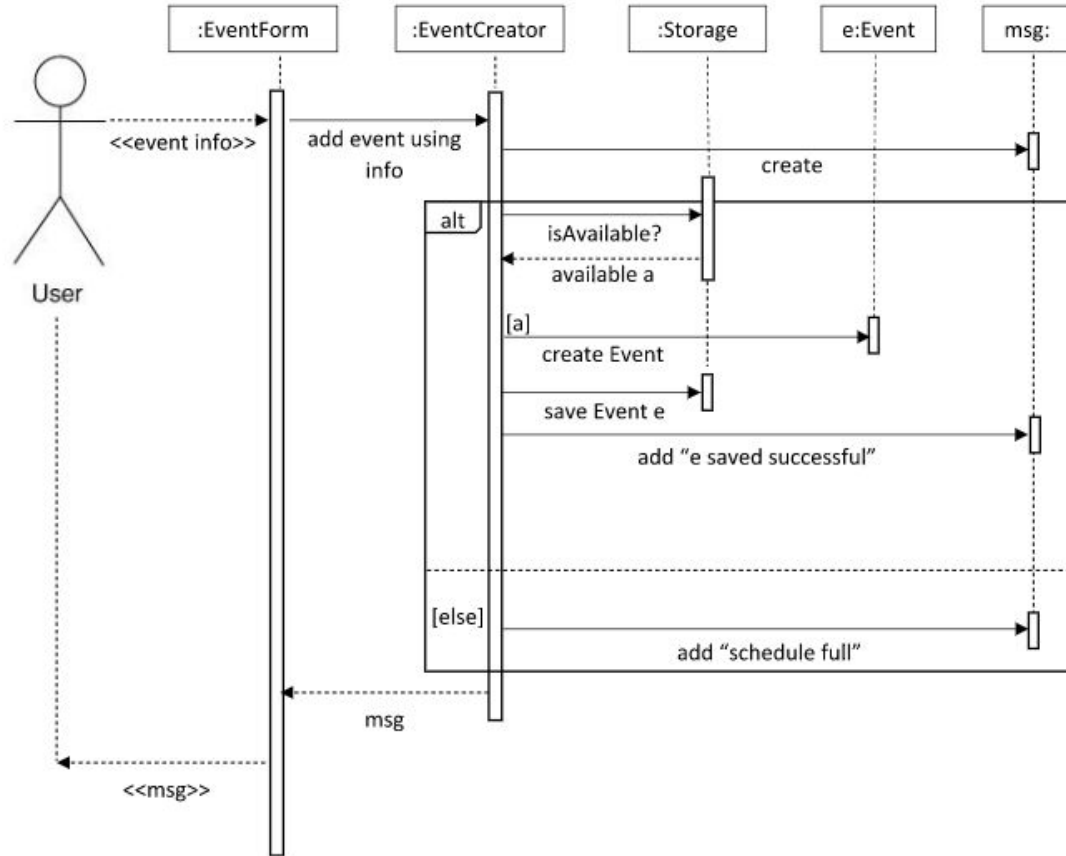
Scenario Table (Add Event)

#	Subject	Subject Action	Other Data/Objects	Object Acted Upon
3.	User	enters	event name, date, start-time, end-time, location, participants	event form
4.	Event creator	saves		event
4.1.	Event creator	searches	time slot	calendar storage
4.2.	If storage has availability during given time frame			
4.2.1.	Event creator	saves	event name, date, start-time, end-time, location, participants	calendar storage
4.2.2.	Event form	displays	“save successful”	message
4.3.	else			
4.3.1.	Event form	displays	“schedule full”	message

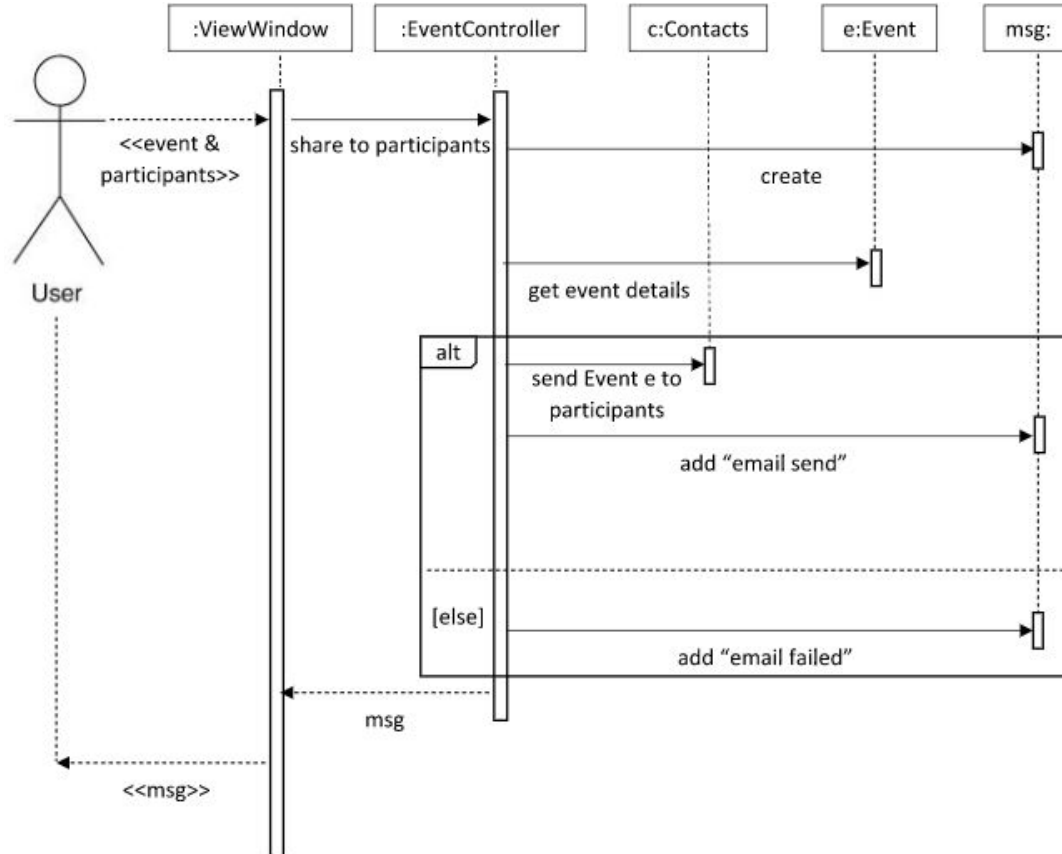
Scenario Table (Share Event)

#	Subject	Subject Action	Other Data/Objects	Object Acted Upon
1.	User	selects	share button	event, participants
2.	Event controller	sends	participants	emails
2.1.	Event controller	fetches		event name, date, start-time, end-time, location, participants
2.2.	Event controller	gathers	participants	emails
2.3.	Event controller	sends	participants	event name, date, start-time, end-time, location
2.4.	If email is send			
2.4.1.	View Window	displays	"email send"	message
2.5.	else			
2.5.1.	View Window	displays	"email failed"	message

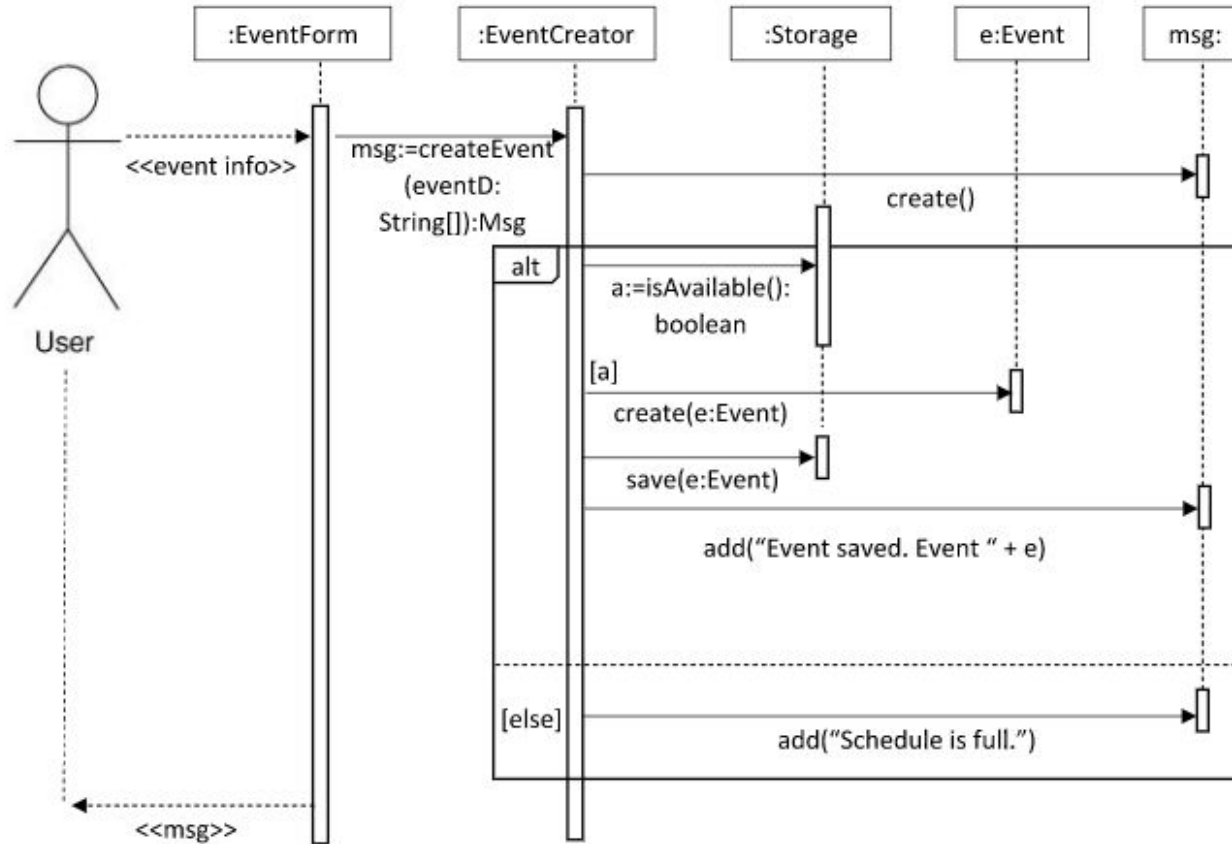
Informal Sequence Diagram (Add Event)



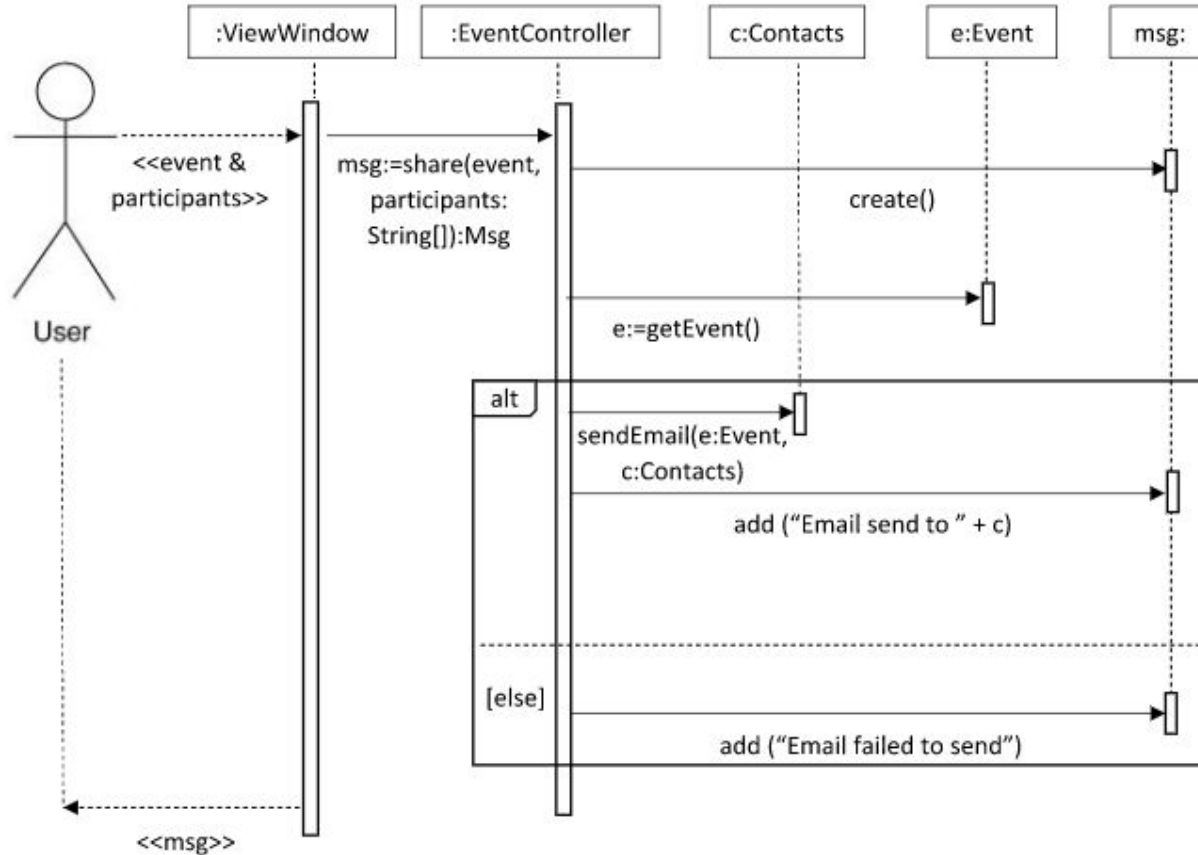
Informal Sequence Diagram (Share Event)



Formal Sequence Diagram (Add Event)



Formal Sequence Diagram (Share Event)



Design Class Diagram

Design Class Diagram

