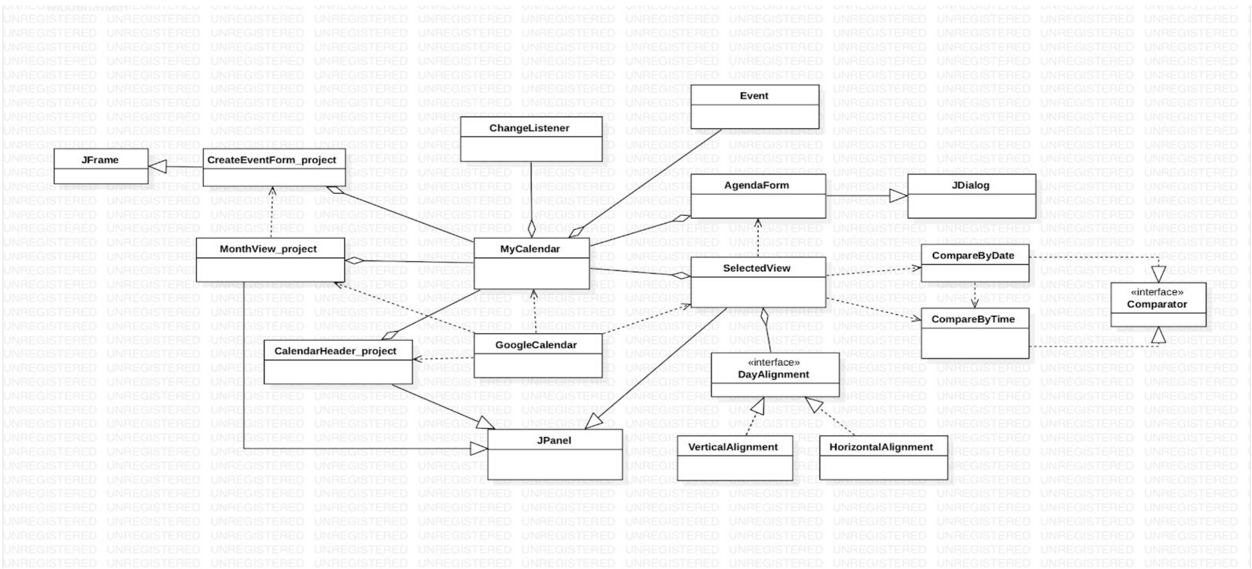
Team - RRJ

Rudra Gandhi
Joleena Marshall
Ritika Singhal
CS 151 - Calendar Project

UML Class Diagram



Use Cases

Use Case: Create Event

Step	User’s Actions	System’s Response
1	User clicks on “Create” button	
2		System displays Create Event form
3	User enters event title, starting time (HH:MM), and ending time (HH:MM)	
4	User clicks on “SAVE” button	
5		System records user’s event by updating the model

Use Case: Load From File

Step	User’s Actions	System’s Response
1	User clicks on “FROM FILE” button	
2		System open the specified file and reads the contents.
3		System parses each line into the required fields for an event and create events from it.
4		The events created are then stored and displayed in the calendar.

Use Case: Print Selected View (day/week/month)

Step	User’s Actions	System’s Response
1	User clicks on “DAY” button	
2		System changes layout according to day and displays all events on the current day

Variation #1
1.1. Start at Step 1
1.2. User clicks on “WEEK” button
1.3 System changes layout according to weeks and displays all events on the days of the current week
Variation #2
2.1. Start at Step 1
2.2. User clicks on “MONTH” button
2.3 System changes layout according to month and displays all events on the days of the current month

Use Case: Month Navigation Buttons (“<”, “>”)

Step	User’s Actions	System’s Response
1	User clicks on “<” button	
2		System navigates current calendar back by month in the month view by changing the date to that of a month before.

Variation #1
1.1. Start at Step 1
1.2. User clicks on “>” button
1.3 System navigates current calendar ahead by month in the month view by changing the date to that of a month after.

Use Case: Selected View Navigation Buttons

Step	User’s Actions	System’s Response
1	User clicks on “<” button	
2		System move the current selected view and month view back depending on the selected view by changing the display date

Variation #1

1.1. Start at Step 1

1.2. User clicks on “>” button

1.3 System move the current selected view and month view ahead depending on the selected view by changing the display date.

Use Case: Today

Step	User’s Actions	System’s Response
1	User clicks on “TODAY” button	
2		System moves the current view to the current day, week, or month depending on the currently selected view and today’s date

Use Case: Check Agenda

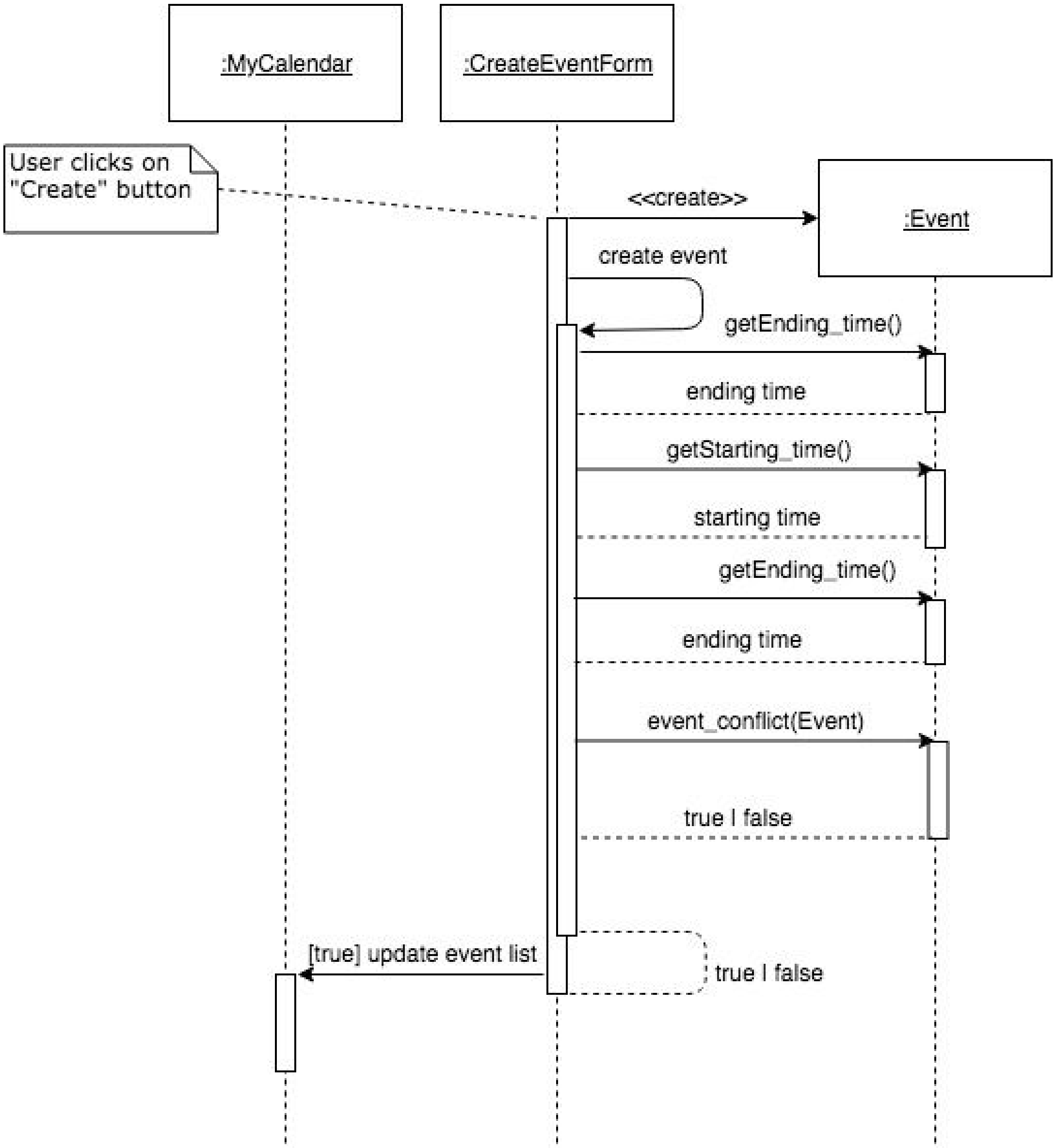
Step	User’s Actions	System’s Response
1	User clicks on “AGENDA” button	
2		System displays Agenda form
3	User enters starting date (MM/DD/YYYY) and ending date (MM/DD/YYYY)	
4	User clicks on “SAVE” button	
5		System displays all events during time period of inputted dates

Use Case: Click on Date

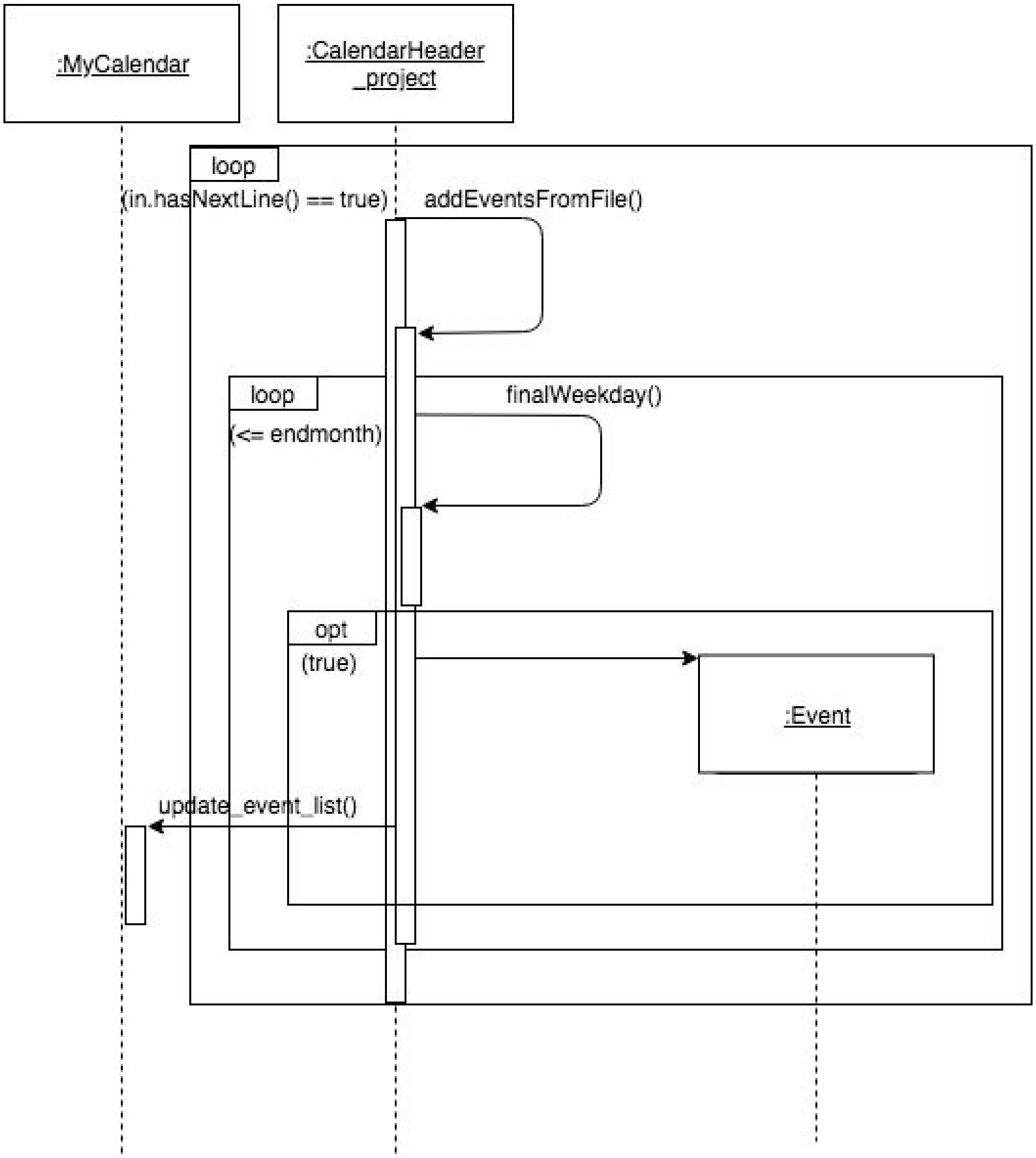
Step	User’s Actions	System’s Response
1	User clicks on date in calendar	
2		System changes current view and month view according to chosen date and sets chosen date as the current date.

Sequence Diagrams

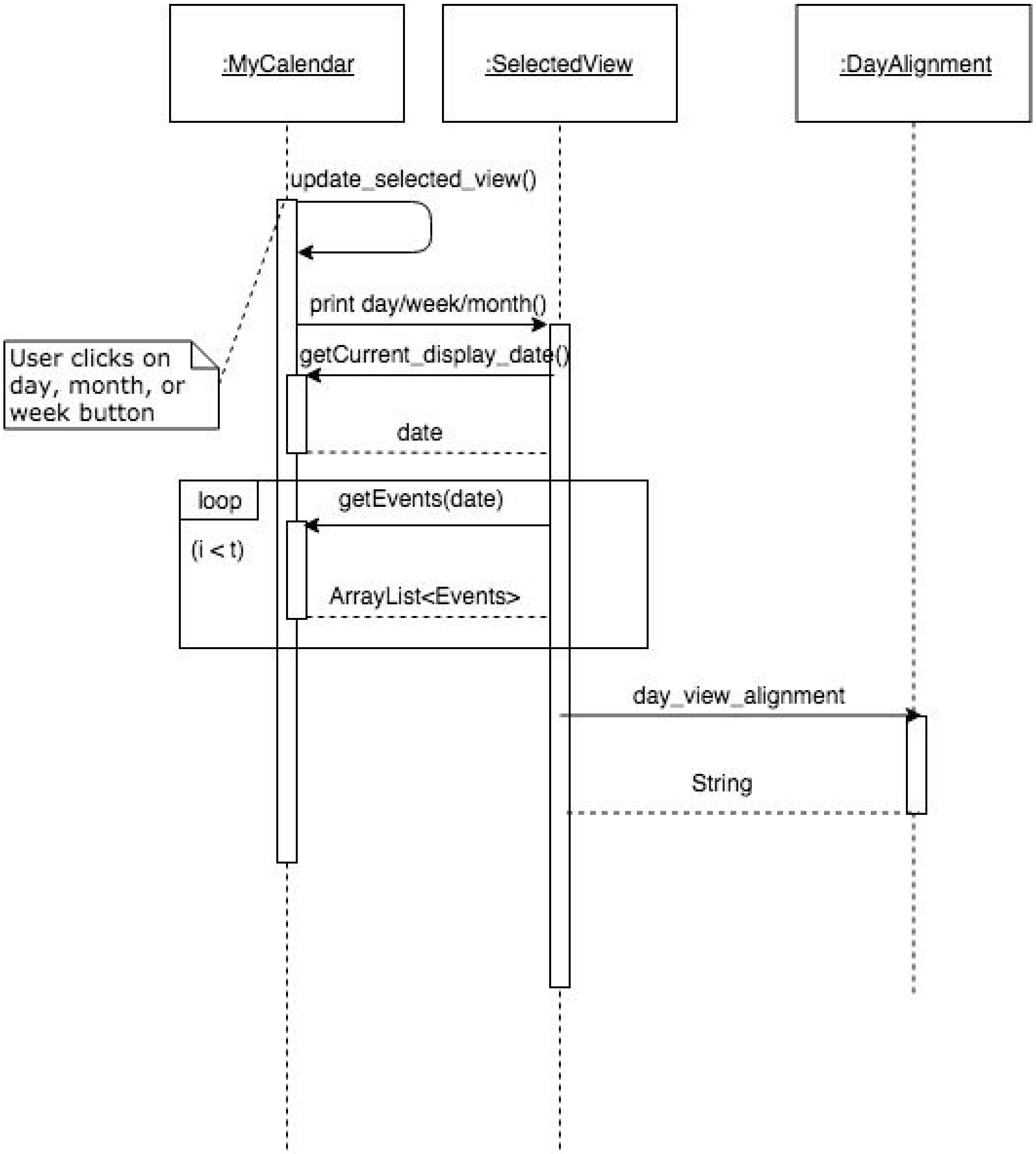
Create Event Use Case



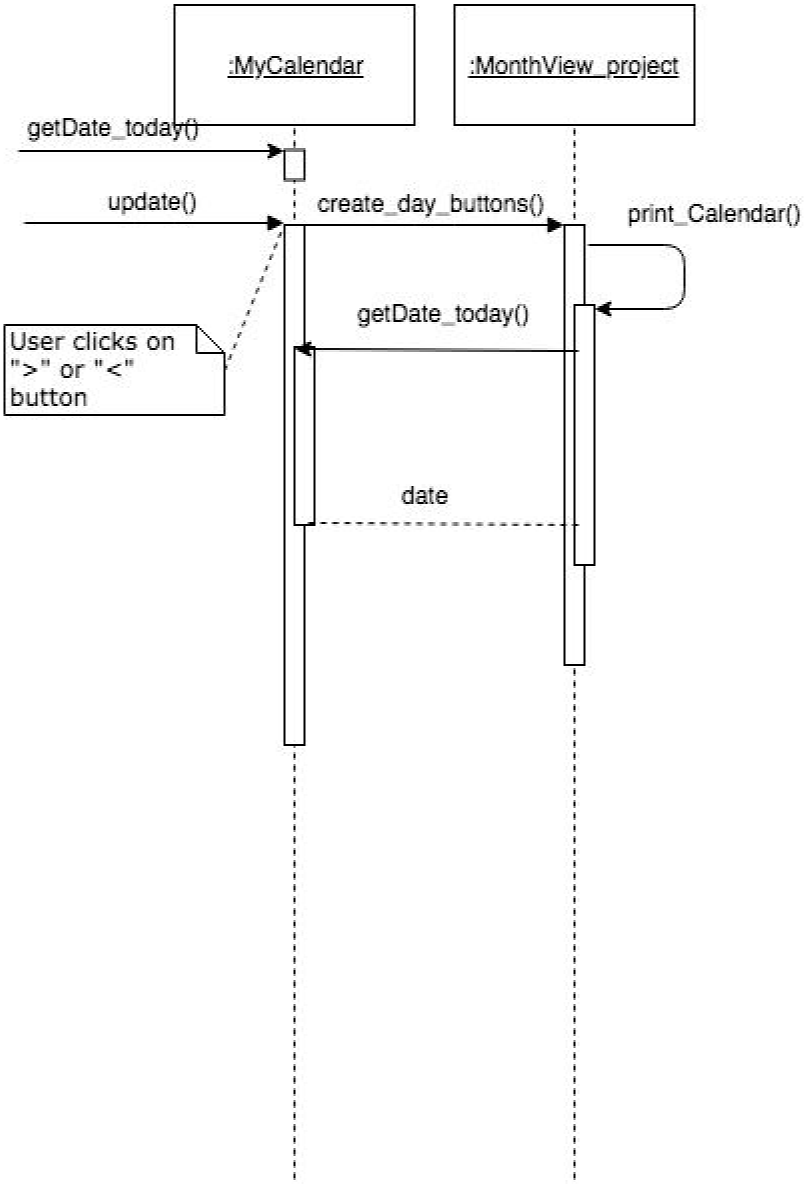
Load From File Use Case



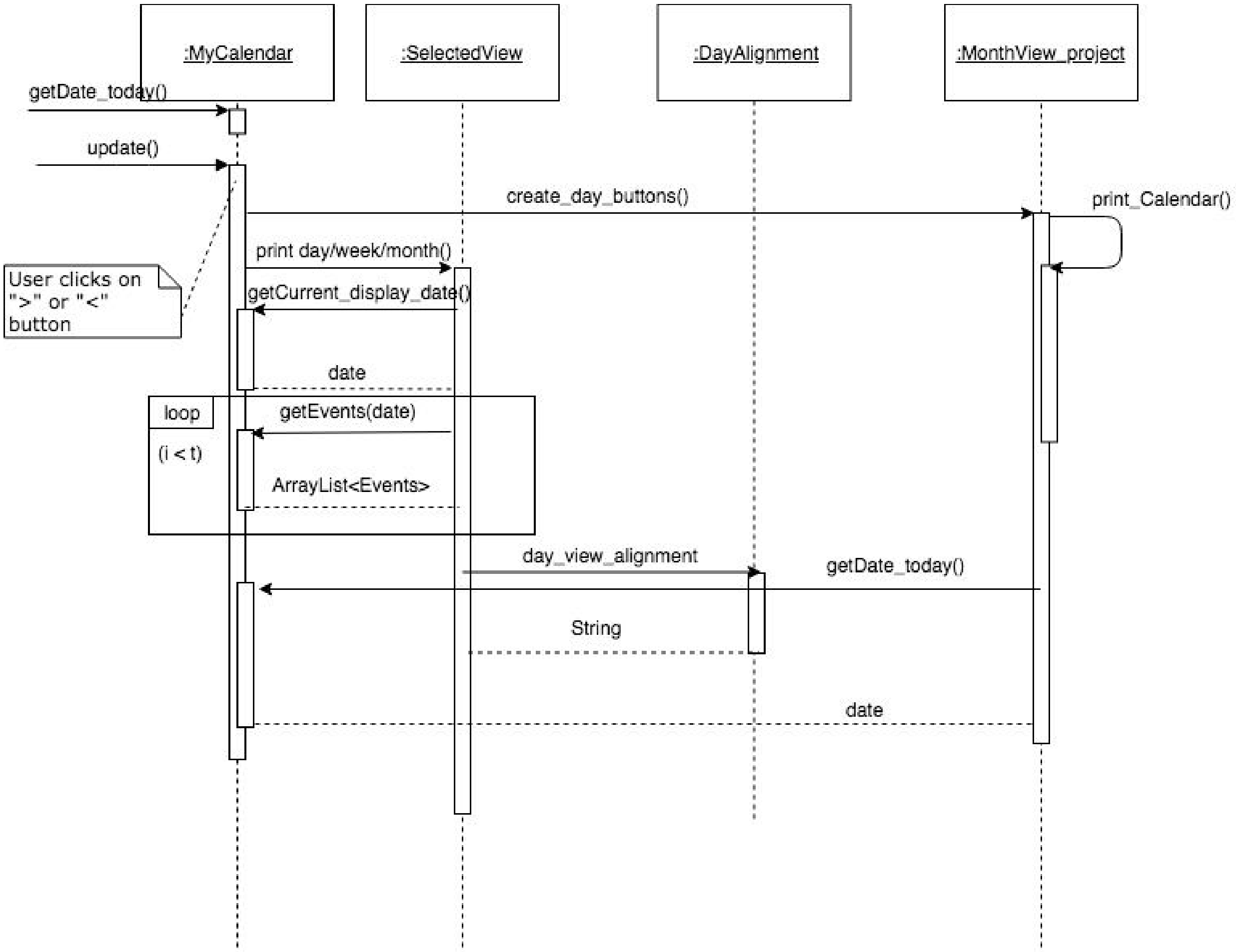
Selected View



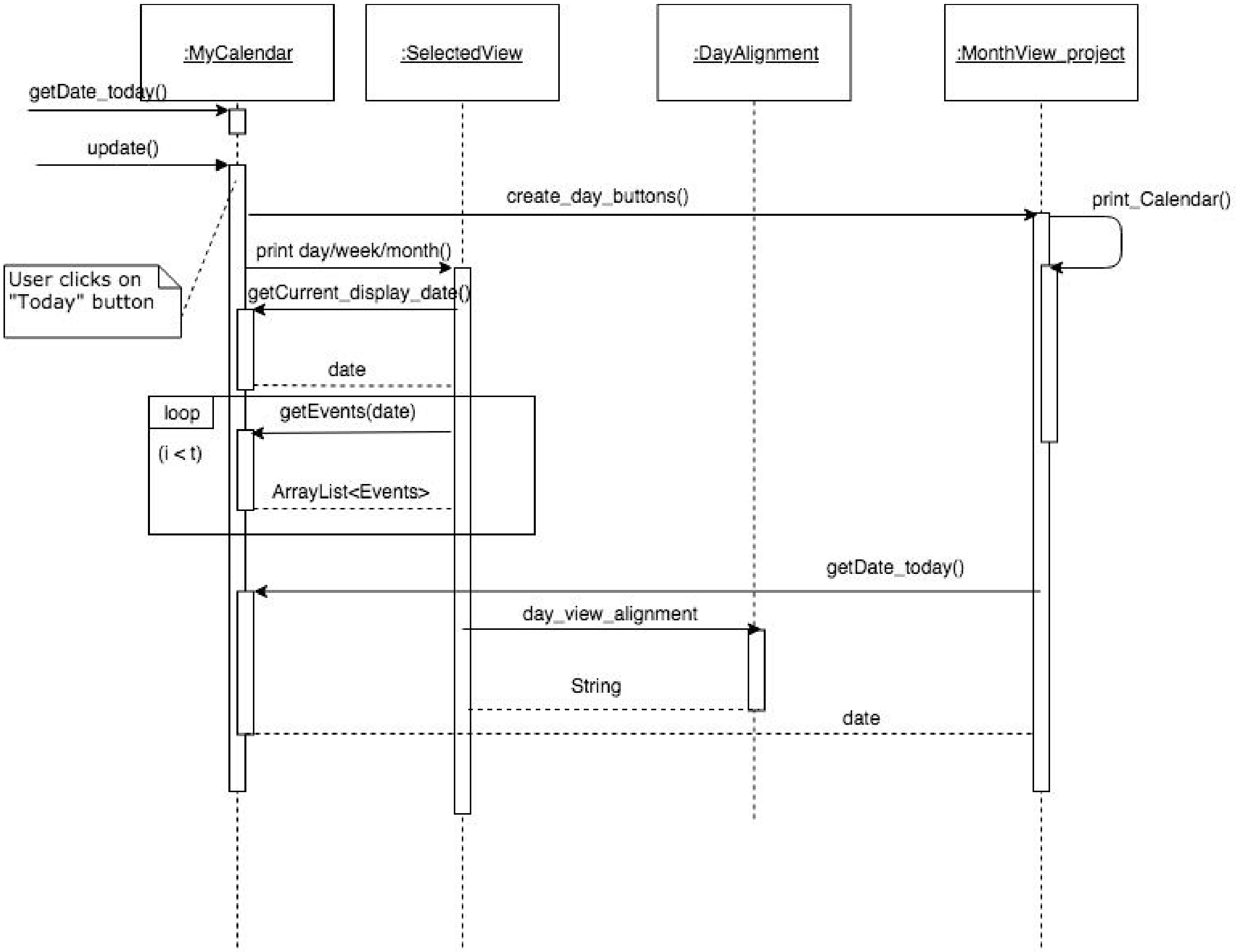
Month Navigation Buttons



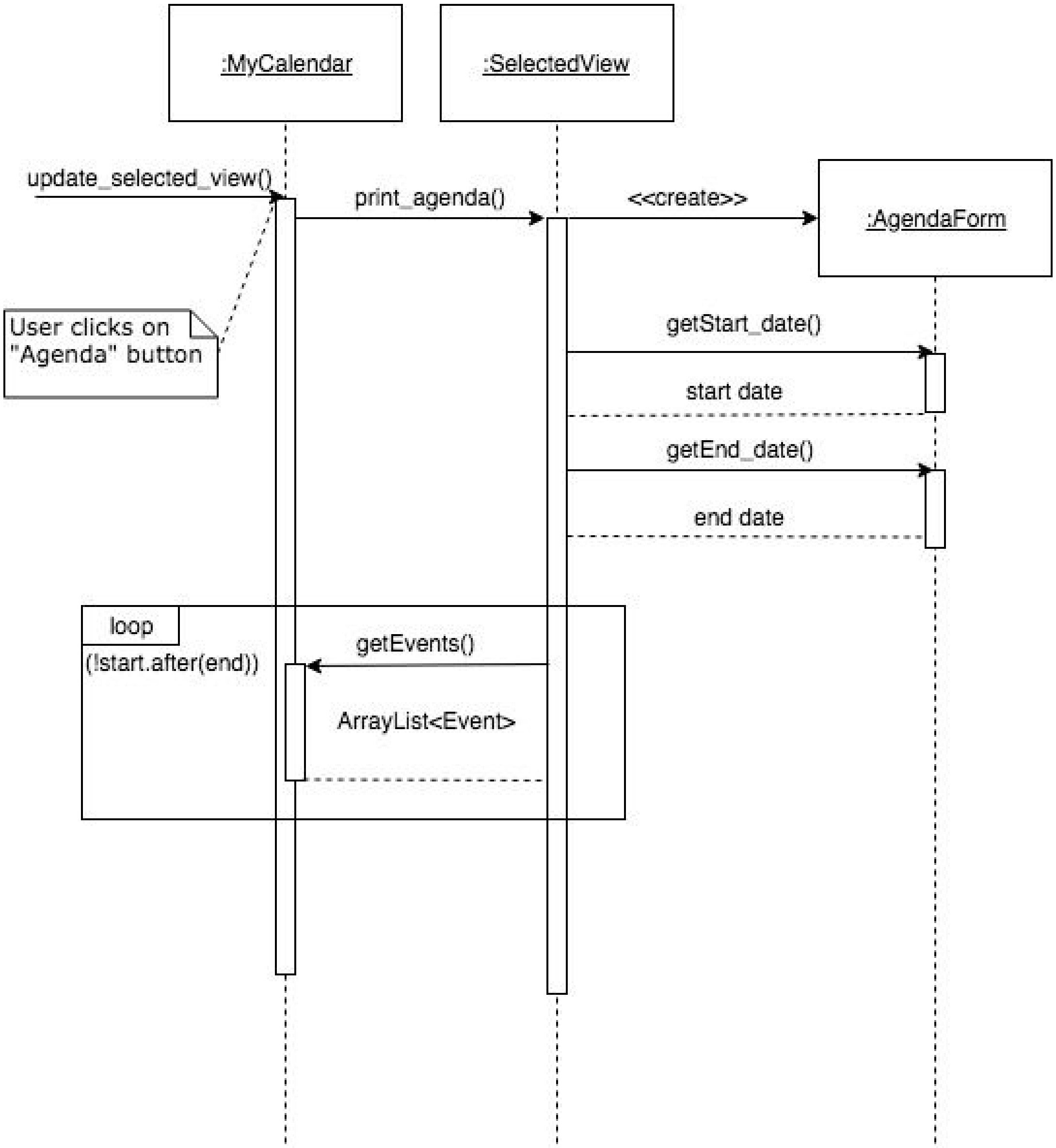
Selected View Navigation Buttons



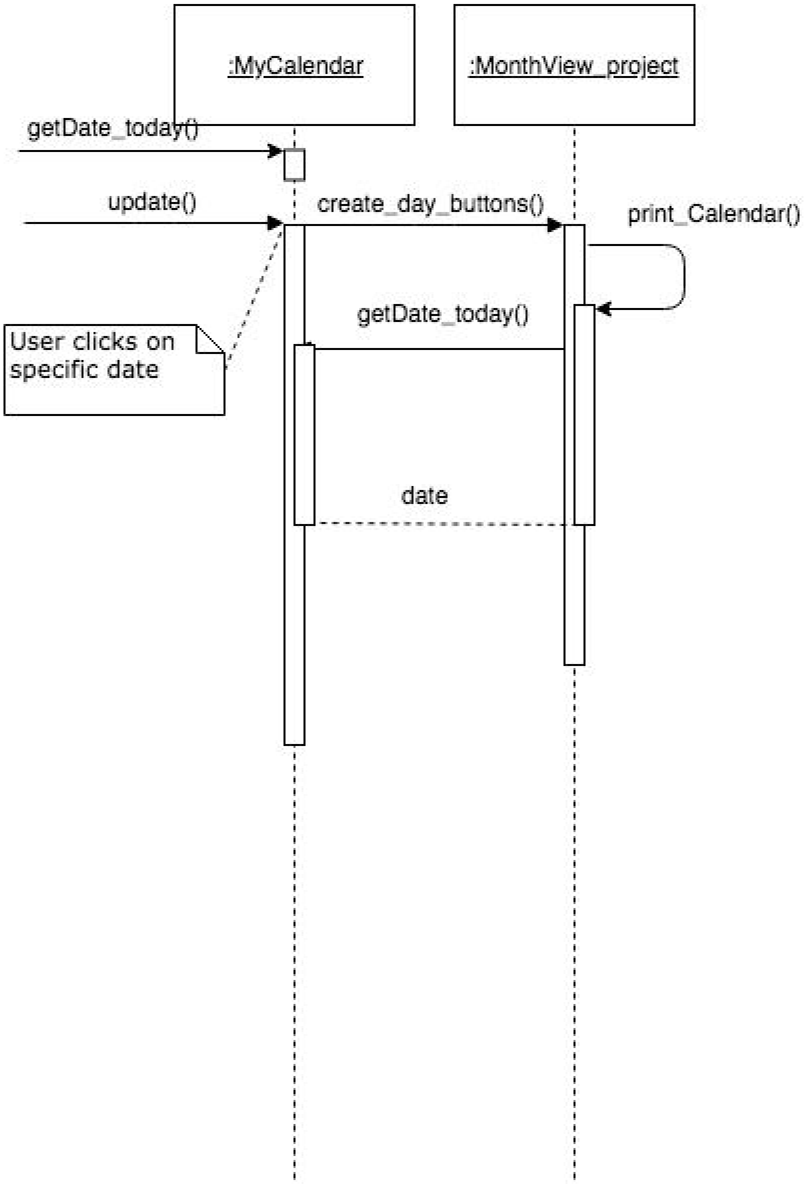
Today Use Case



Check Agenda Use Case



Click on Date Use Case



Summary

1. Which materials/key concepts from this course did you apply on the project?

   From this course, some of the key concepts applied in this project include creating class and sequence diagrams as well as defining the relationships between classes. Creating use cases made the implementation easier for us by allowing us to visualize how the user and program interact for us to provide a user-friendly interface. We also applied the MVC model. Here we were able to classify the model, view, and controller by using their respective roles to create an architecture for our program. We also applied the comparable interface, strategy pattern, adding listeners, and accessor and mutator methods. We were also able to use containers such as JFrame, JLabel, and text area in order to display data through the GUI.

2. Which topics did you have to learn through self-study in order to complete the project?

   A topic we had to learn through self study was the API of Gregorian Calendar. We also had to teach ourselves more about implementing built-in layout managers such as BoxLayout for our Create Event form, and GridLayout and JDialog for our Agenda Form. Essentially, we had to familiarize ourselves with a lot of the Swing and AWT java toolkits to optimize our GUI and user-program interaction.