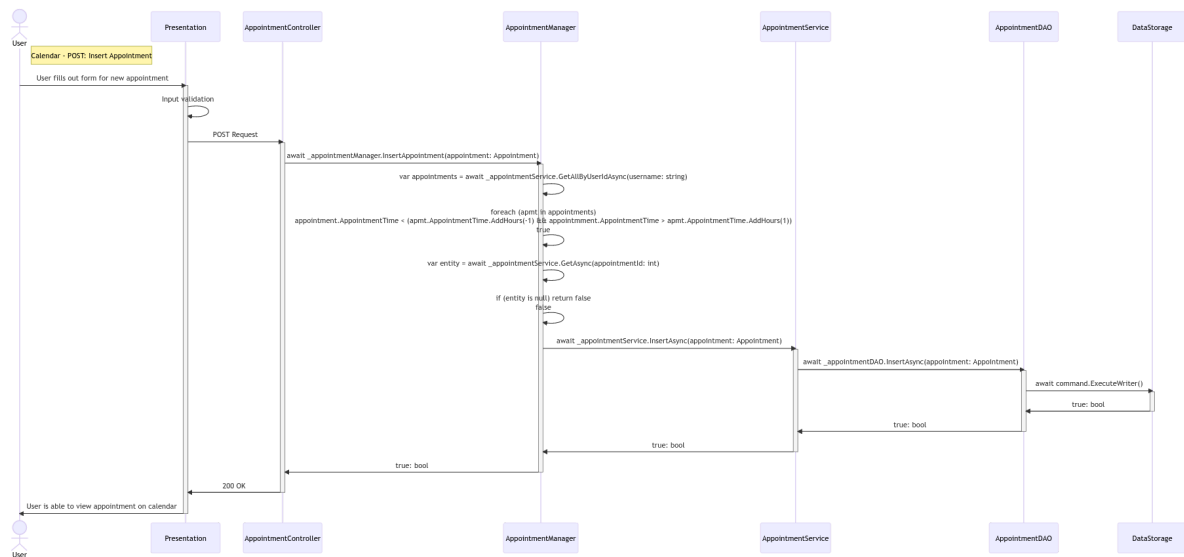
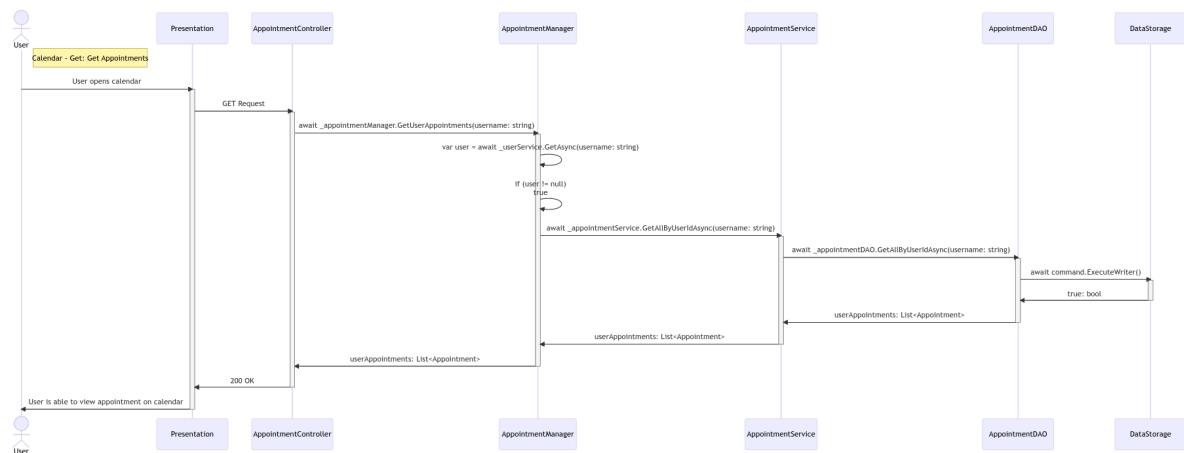


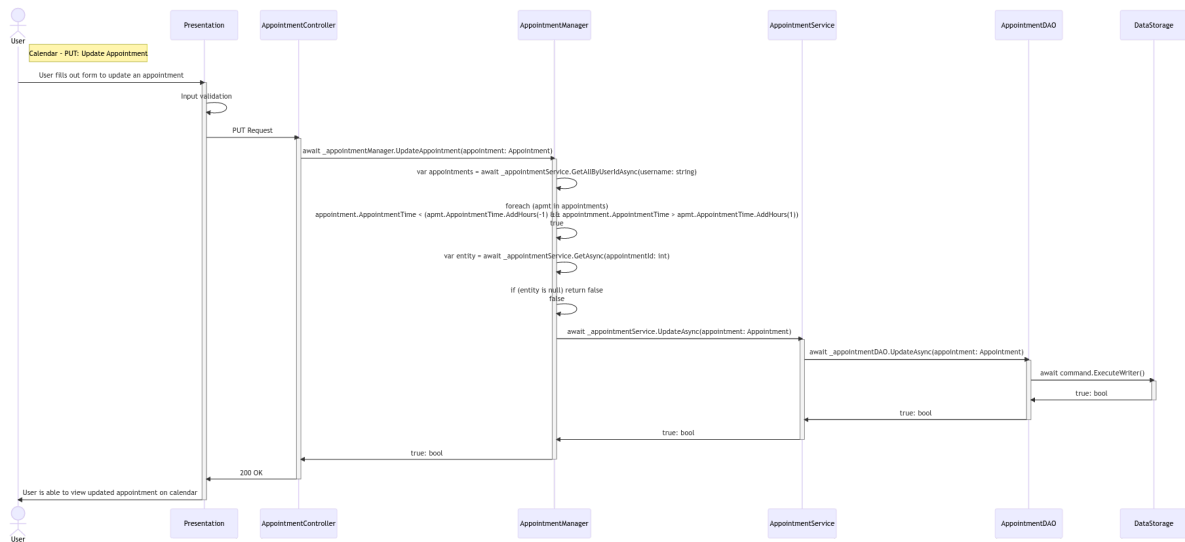
## Insert Appointment to Calendar



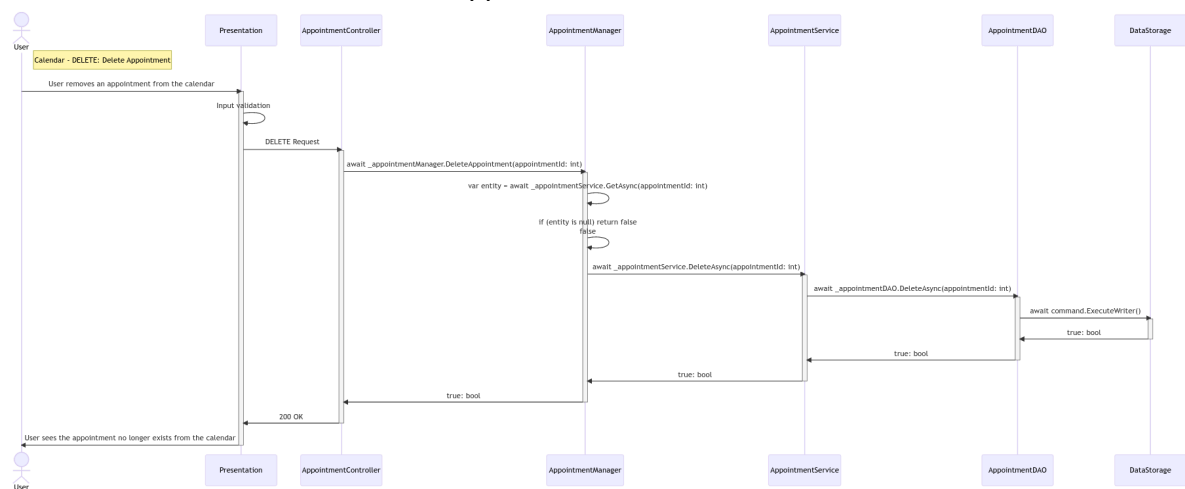
## Get All Appointments from a user and display on Calendar



## Update Appointment on Calendar



## Delete Appointment from Calendar



## Library

### sequenceDiagram

```

actor U as User
Note right of U: Calendar - POST: Insert Appointment
participant P as Presentation
participant AC as AppointmentController
participant AM as AppointmentManager
participant AS as AppointmentService
participant AD as AppointmentDAO
participant DS as DataStorage
U->>P: User fills out form for new appointment
P->>P: Input validation
  
```

```

P->>>+AC: POST Request
AC->>>+AM: await _appointmentManager.InsertAppointment(appointment:
Appointment)
AM->>>AM: var appointments = await
_appointmentService.GetAllByUserIdAsync(username: string)
AM->>>AM: foreach (apmt in
appointments)<br/>appointment.AppointmentTime <
(apmt.AppointmentTime.AddHours(-1) && appointment.AppointmentTime >
apmt.AppointmentTime.AddHours(1))<br/>true
AM->>>AM: var entity = await
_appointmentService.GetAsync(appointmentId: int)
AM->>>AM: if (entity is null) return false<br>>false
AM->>>+AS: await _appointmentService.InsertAsync(appointment:
Appointment)
AS->>>+AD: await _appointmentDAO.InsertAsync(appointment: Appointment)
AD->>>+DS: await command.ExecuteWriter()
DS->>>-AD: true: bool
AD->>>-AS: true: bool
AS->>>-AM: true: bool
AM->>>-AC: true: bool
AC->>>-P: 200 OK
P->>>-U: User is able to view appointment on calendar

```

Post

```

sequenceDiagram
    actor U as User
    Note right of U: Calendar - Get: Get Appointments
    participant P as Presentation
    participant AC as AppointmentController
    participant AM as AppointmentManager
    participant AS as AppointmentService
    participant AD as AppointmentDAO
    participant DS as DataStorage
    U->>>+P: User opens calendar
    P->>>+AC: GET Request
    AC->>>+AM: await _appointmentManager.GetUserAppointments(username:
string)
AM->>>AM: var user = await _userService.GetAsync(username: string)
AM->>>AM: if (user != null)<br/>true

```

```

    AM->>+AS: await _appointmentService.GetAllByIdAsync(username:
string)
    AS->>+AD: await _appointmentDAO.GetAllByIdAsync(username: string)
    AD->>+DS: await command.ExecuteWriter()
    DS->>-AD: true: bool
    AD->>-AS: userAppointments: List<Appointment>
    AS->>-AM: userAppointments: List<Appointment>
    AM->>-AC: userAppointments: List<Appointment>
    AC->>-P: 200 OK
    P->>-U: User is able to view appointment on calendar

```

Get

#### sequenceDiagram

```

actor U as User
Note right of U: Calendar - PUT: Update Appointment
participant P as Presentation
participant AC as AppointmentController
participant AM as AppointmentManager
participant AS as AppointmentService
participant AD as AppointmentDAO
participant DS as DataStorage
U->>+P: User fills out form to update an appointment
P->>P: Input validation
P->>+AC: PUT Request
AC->>+AM: await _appointmentManager.UpdateAppointment(appointment:
Appointment)
    AM->>AM: var appointments = await
    _appointmentService.GetAllByIdAsync(username: string)
    AM->>AM: foreach (apmt in
appointments)<br/>appointment.AppointmentTime <
(apmt.AppointmentTime.AddHours(-1) && appointment.AppointmentTime >
apmt.AppointmentTime.AddHours(1))<br/>true
    AM->>AM: var entity = await
    _appointmentService.GetAsync(appointmentId: int)
    AM->>AM: if (entity is null) return false<br/>>false
    AM->>+AS: await _appointmentService.UpdateAsync(appointment:
Appointment)
    AS->>+AD: await _appointmentDAO.UpdateAsync(appointment: Appointment)
    AD->>+DS: await command.ExecuteWriter()

```

```

DS->>-AD: true: bool
AD->>-AS: true: bool
AS->>-AM: true: bool
AM->>-AC: true: bool
AC->>-P: 200 OK
P->>-U: User is able to view updated appointment on calendar

```

Put

#### sequenceDiagram

```

actor U as User
Note right of U: Calendar - DELETE: Delete Appointment
participant P as Presentation
participant AC as AppointmentController
participant AM as AppointmentManager
participant AS as AppointmentService
participant AD as AppointmentDAO
participant DS as DataStorage
U->>+P: User removes an appointment from the calendar
P->>P: Input validation
P->>+AC: DELETE Request
AC->>+AM: await _appointmentManager.DeleteAppointment(appointmentId:
int)
AM->>AM: var entity = await
_appointmentService.GetAsync(appointmentId: int)
AM->>AM: if (entity is null) return false<br>>false
AM->>+AS: await _appointmentService.DeleteAsync(appointmentId: int)
AS->>+AD: await _appointmentDAO.DeleteAsync(appointmentId: int)
AD->>+DS: await command.ExecuteWriter()
DS->>-AD: true: bool
AD->>-AS: true: bool
AS->>-AM: true: bool
AM->>-AC: true: bool
AC->>-P: 200 OK
P->>-U: User sees the appointment no longer exists from the calendar

```

Delete