Quarter 1 Code Review

Katherine Palevich

Background

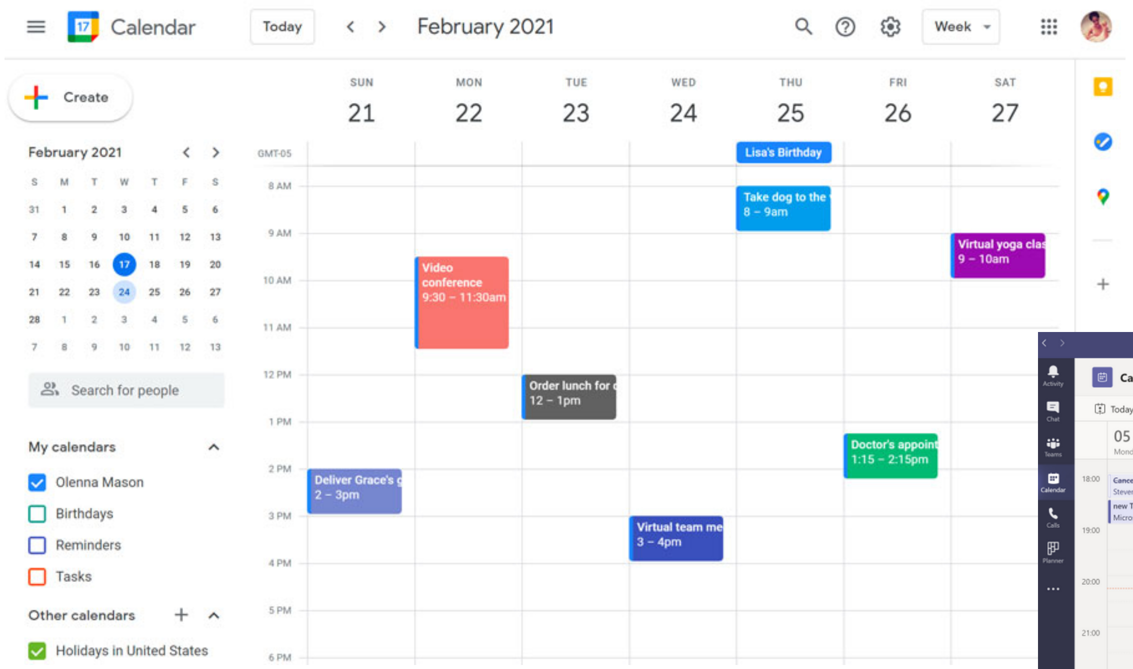
What is Aureate?

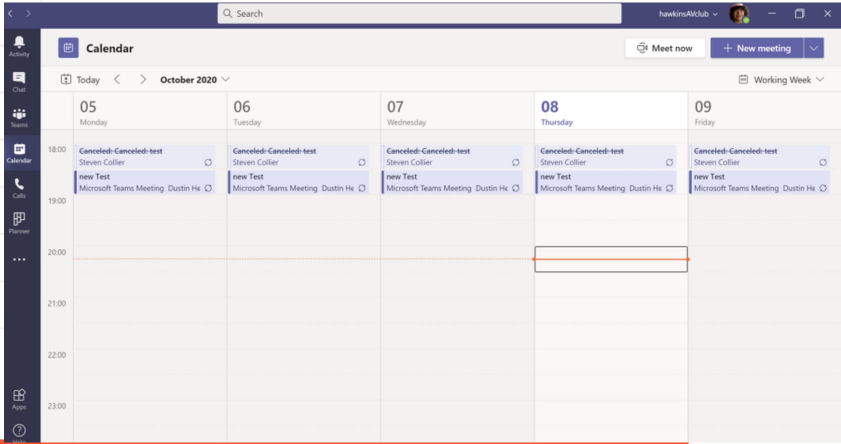
- An application that combines the use of a journal, planner, and dream tracker into one.

- Reduces the amount of paper weight and waste.

- Push notifications to remind you of important events and tasks.

Inspirations





Virtual Inspirations

Objectives for the First 2 Months

General Layout

Code the rough tabs and layout of the app

Journal Tab

- Allow the user to add a journal entry and edit it
- Be able to type into an entry along with using an Apple Pencil to doodle onto the page.

Calendar Tab

- Allow the user to add, edit, and view a calendar event
- Display events in a weekly and daily view
    - Events are spaced out in relation to when they start/end

What I Actually Got To

General Layout

Code the rough tabs and layout of the app

Journal Tab

- Allow the user to add a journal entry and edit it
- Be able to type into an entry along with using an Apple Pencil to doodle onto the page.

Calendar Tab

- Allow the user to add, edit, and view a calendar event
- Display events in a weekly and daily view
    - Events are spaced out in relation to when they start/end

What Went Well/How I Expected

Journal Tab

Only took me a week to implement.

Accessing User’s Calendar Events

Displaying an event using the given classes was fairly simple and straight forward.

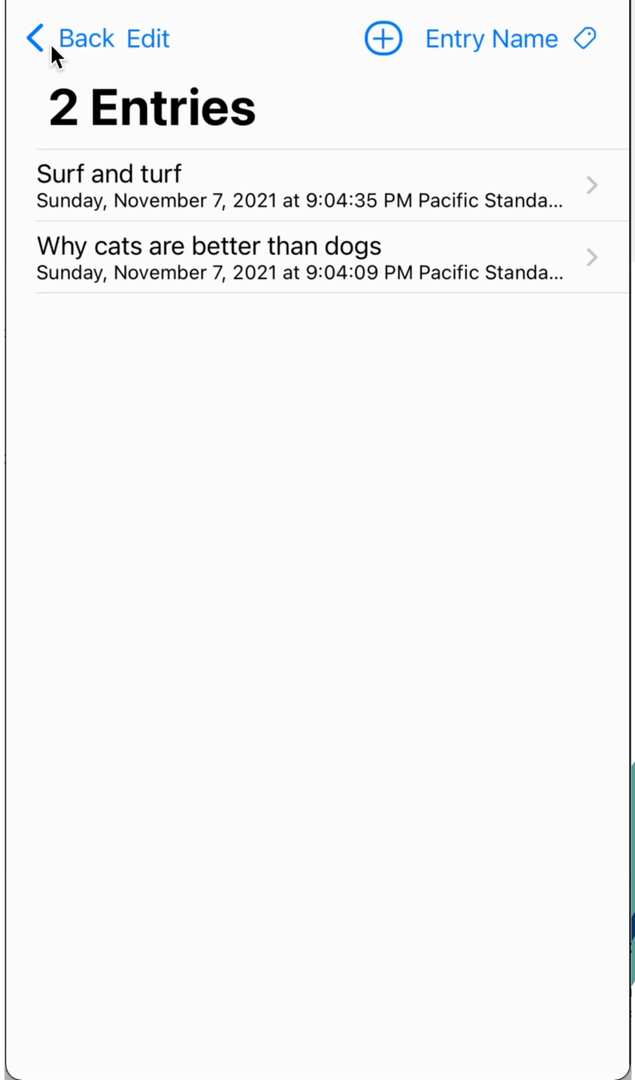
What Went Wrong/Worse Than Expected?

Drawing Feature in Journal Tab

- Hard to find examples online.
- No obvious SwiftUI in Apple Documentation

Planner Page

- Editing an event using Apple's EKEventViewController took way longer to implement (delegates)
- Getting the details right for a week was more complicated than initially thought



2 Entries

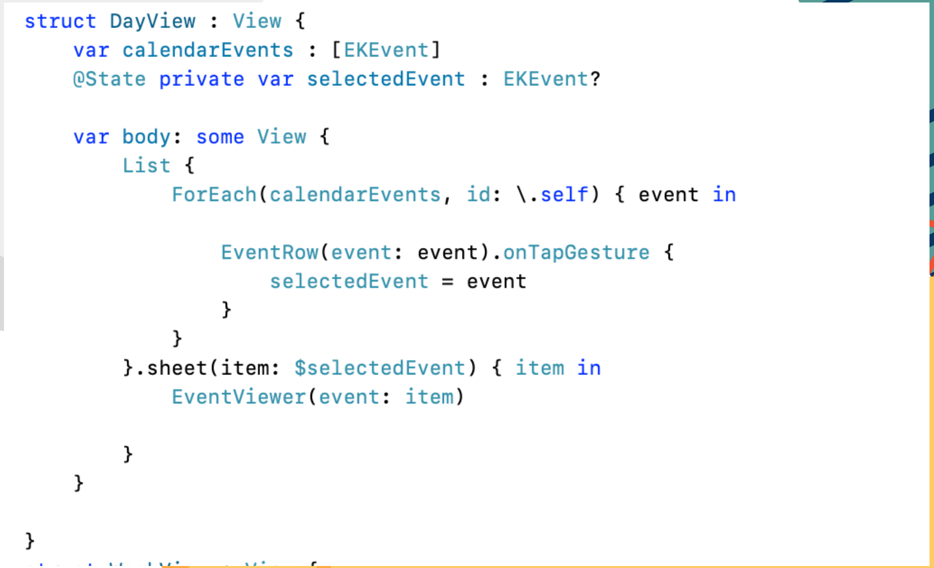
Surf and turf
Sunday, November 7, 2021 at 9:04:35 PM Pacific Standa...

Why cats are better than dogs
Sunday, November 7, 2021 at 9:04:09 PM Pacific Standa...

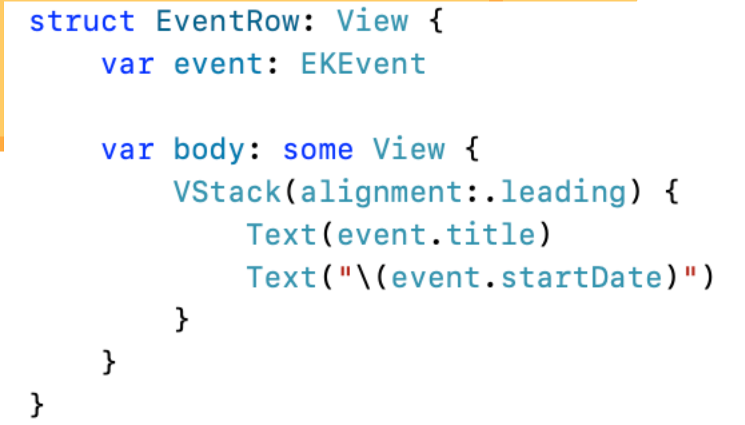
Events Class

```swift
private func populateDayEvents(date: Date){
    // Get the appropriate calendar.
    let calendar = Calendar.current

    let beginningOfDay = calendar.startOfDay(for: date)
    var dateComponents = DateComponents()
    dateComponents.minute = 0
    dateComponents.second = 0
    dateComponents.hour = 0

    let endOfDay = calendar.nextDate(after: date, matching: dateComponents, matchingPolicy: .nextTime)!

    let predicate = Events.eventStore.predicateForEvents(withStart: beginningOfDay, end: endOfDay, calendars:
        Events.eventStore.calendars(for: EKEntityType.event))
    dayEvents = Events.eventStore.events(matching: predicate)

}

private func populateWeekEvents(date: Date){
    // Get the appropriate calendar.
    let calendar = Calendar.current

    let beginningOfWeek = (calendar.nextWeekend(startingAfter: date, direction: .backward)?.end)?.addingTimeInterval(-60 * 60 * 24)
    var endOfWeek = (calendar.nextWeekend(startingAfter: date, direction: .forward)?.end)!.addingTimeInterval(-60 * 60 * 24)
    if(calendar.isDateInWeekend(date) && calendar.isDateInWeekend(date.addingTimeInterval(60 * 60 * 24))){
        endOfWeek = date
    }


    let predicate = Events.eventStore.predicateForEvents(withStart: beginningOfWeek!, end: endOfWeek, calendars:
        Events.eventStore.calendars(for: EKEntityType.event))
    weekEvents = Events.eventStore.events(matching: predicate)

}
```

Displaying Day/Week's Events



```swift
struct DayView : View {
    var calendarEvents : [EKEvent]
    @State private var selectedEvent : EKEvent?

    var body: some View {
        List {
            ForEach(calendarEvents, id: \.self) { event in

                EventRow(event: event).onTapGesture {
                    selectedEvent = event
                }
            }
        }.sheet(item: $selectedEvent) { item in
            EventViewer(event: item)

        }
    }

}
```



```swift
struct EventRow: View {
    var event: EKEvent

    var body: some View {
        VStack(alignment:.leading) {
            Text(event.title)
            Text("\(event.startDate)")
        }
    }
}
```

Thank You