



Safety App

Spring 2024
Construction
Management
Team 1

iOS application developed by Swift



Team 1 Member Roles

Lee, Sangkyu –Research regarding market analysis

Lee, Jongkyu – Code configuration

Erdenekhuu, Enkhmaa Emma – Research and analysis regarding implications of application, presentation

Kim, Eunbi – Conceptualize the main idea of the application

Heo, Eunjin – Making and designing presentation material

Table of Contents

01

Introduction

Purpose and
Motivation for
Developing the
Safety App

02

App Service

Safety App
Functions

03

App Development

Process
Explanation of the
Swift Code

04

Sales Goal

Estimating Sales
based on Market
Analysis

01. Safety App Introduction

■ Preventing hazardous accidents caused by bad weather conditions and a lack of safety helmets.



Outdoor construction sites



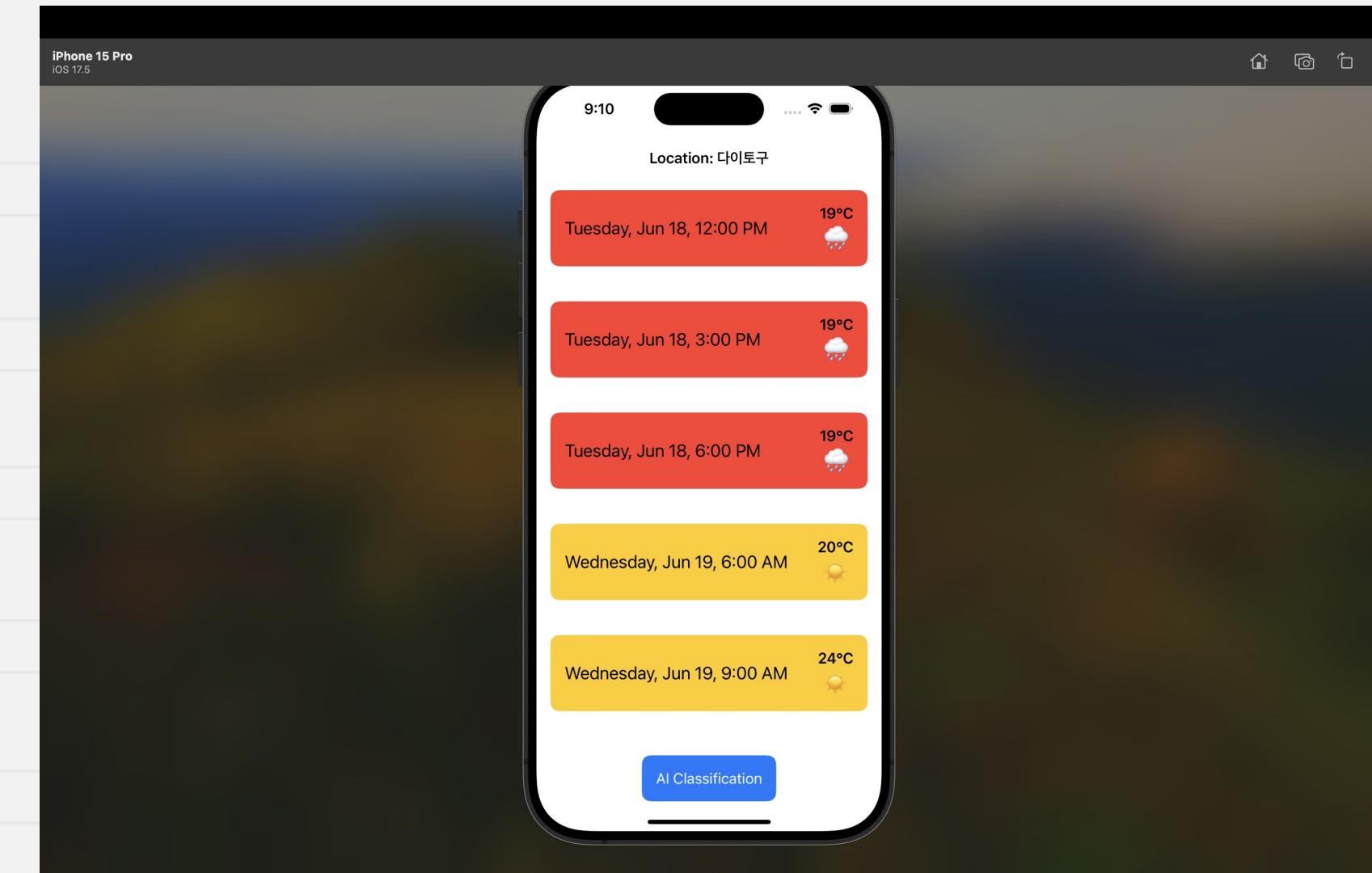
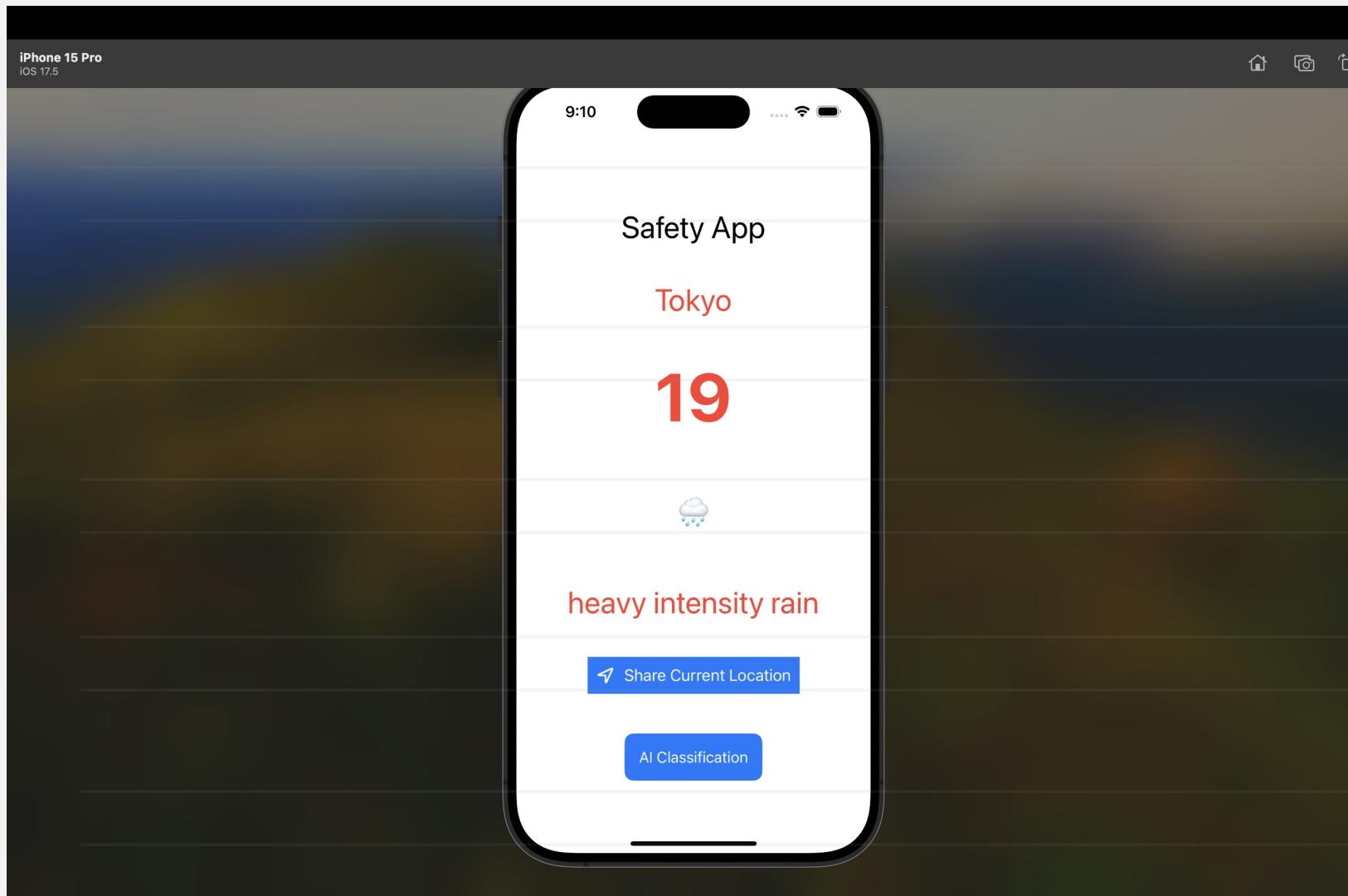
Bad weather conditions



A lack of safety helmets

02. Safety App

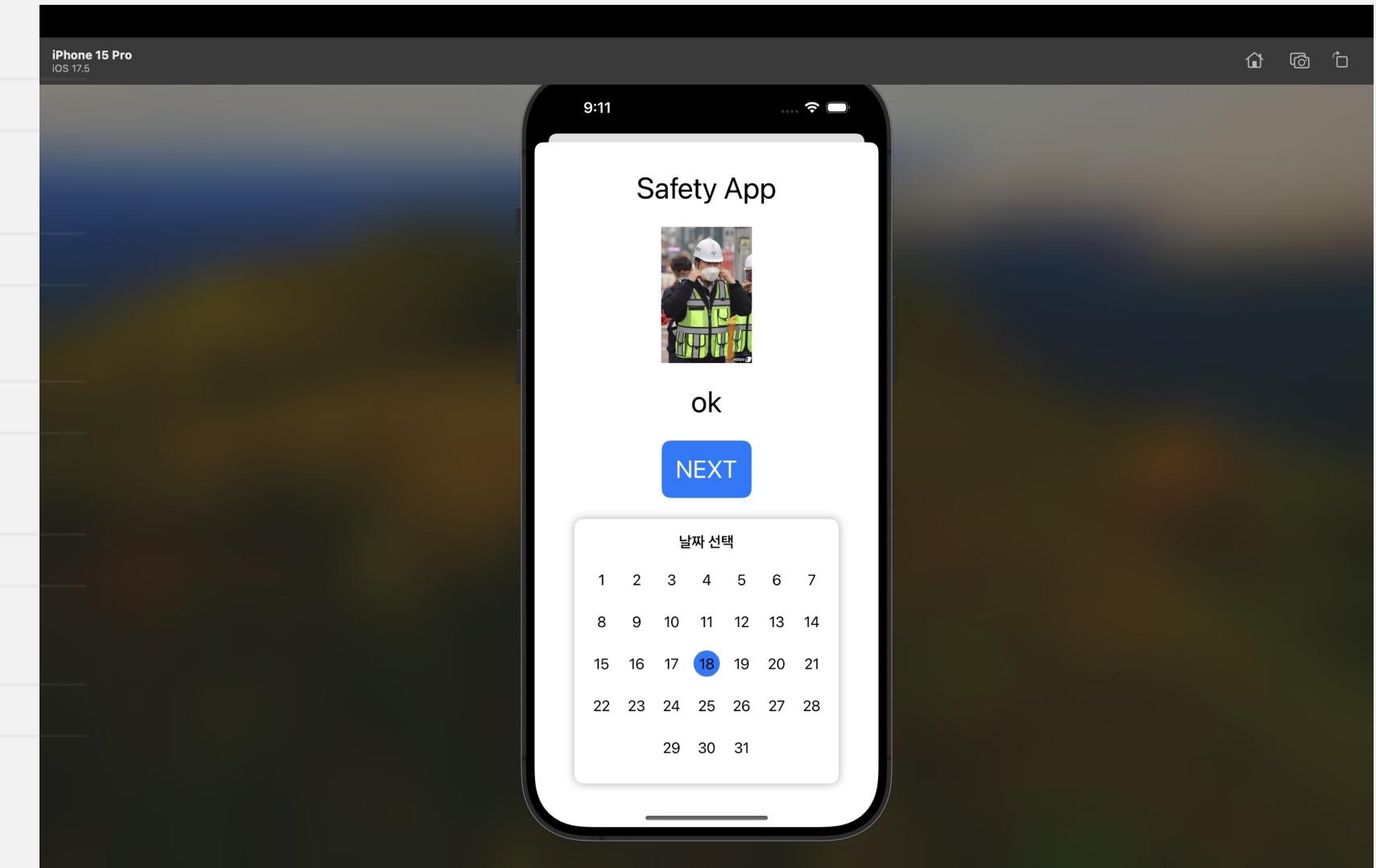
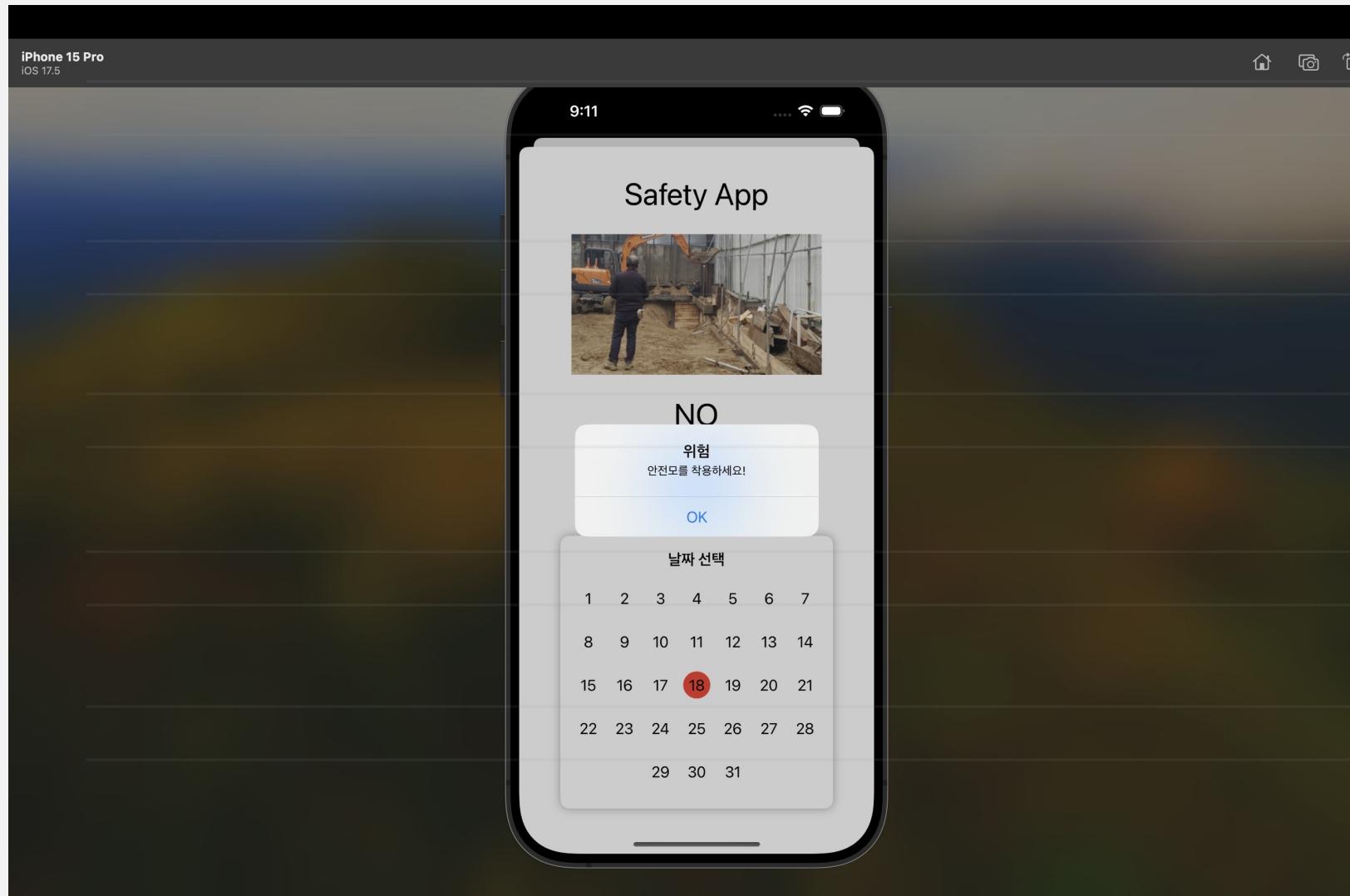
Providing Weather Analysis features for outdoor working environments



02. Safety App Service

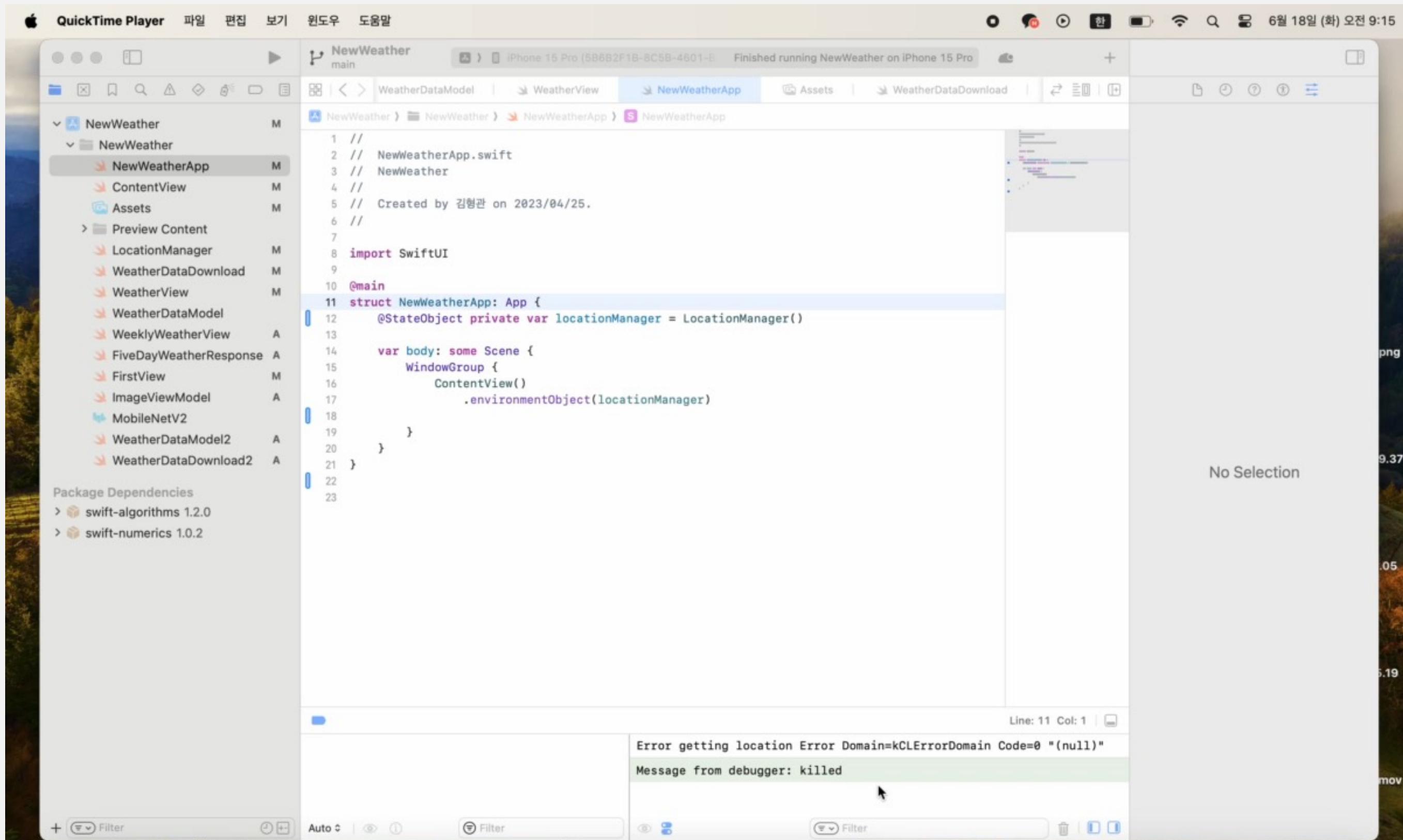
Checking whether safety helmets are being worn

+ Ringing and buzzing function until safety helmets are properly worn



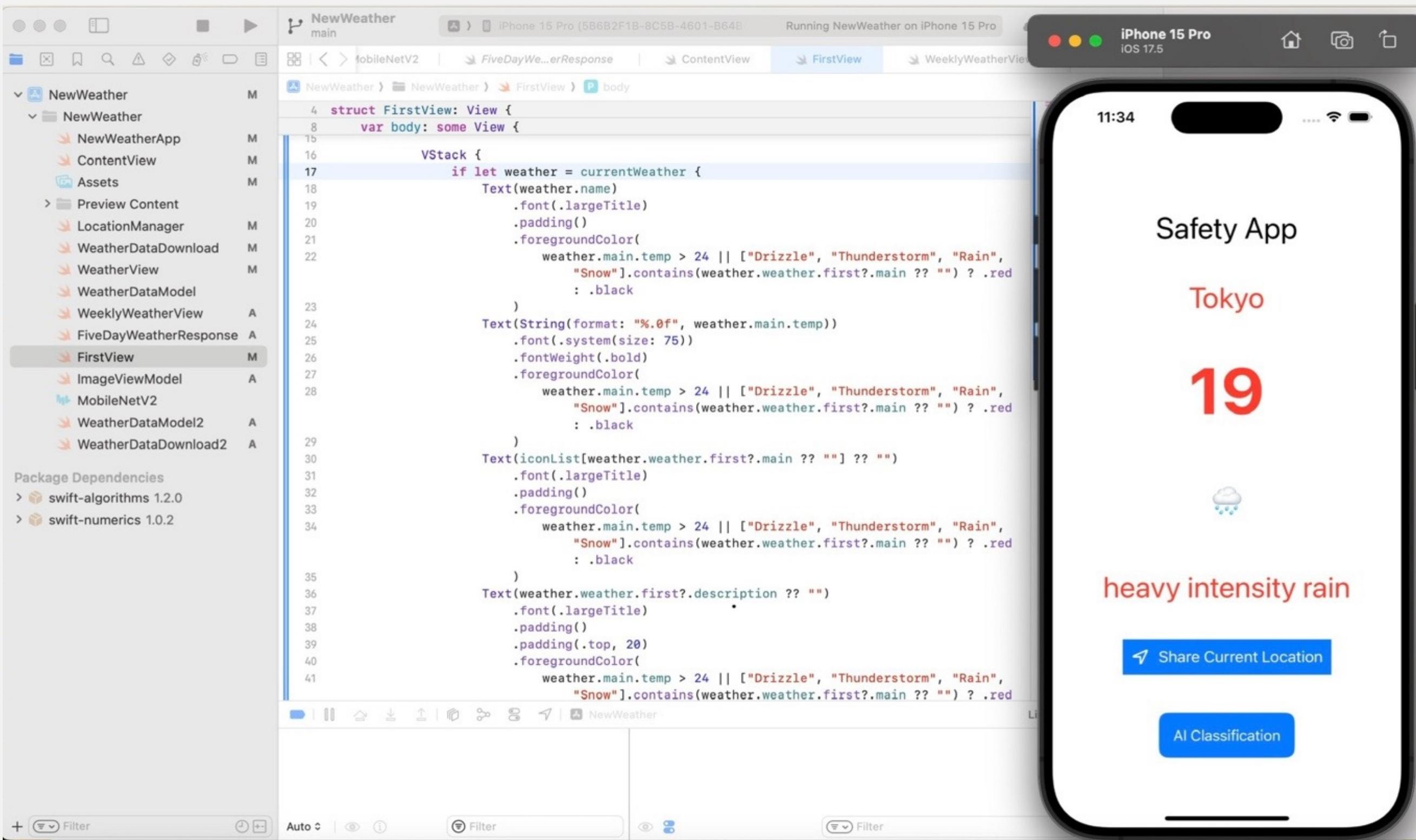
02. Safety App Service

App Video



03.App Developing Process

Explanation of the Swift Code



The image shows the Xcode IDE on the left and a running iPhone 15 Pro simulator on the right.

Xcode Project Structure:

- Project: NewWeather
- Module: NewWeather
- Files:
 - NewWeatherApp
 - ContentView
 - Assets
 - Preview Content
 - LocationManager
 - WeatherDataDownload
 - WeatherView
 - WeatherDataManager
 - WeeklyWeatherView
 - FiveDayWeatherResponse
 - FirstView
 - ImageViewModel
 - MobileNetV2
 - WeatherDataManager2
 - WeatherDataDownload2
- Package Dependencies:
 - swift-algorithms 1.2.0
 - swift-numerics 1.0.2

Swift Code (FirstView.swift):

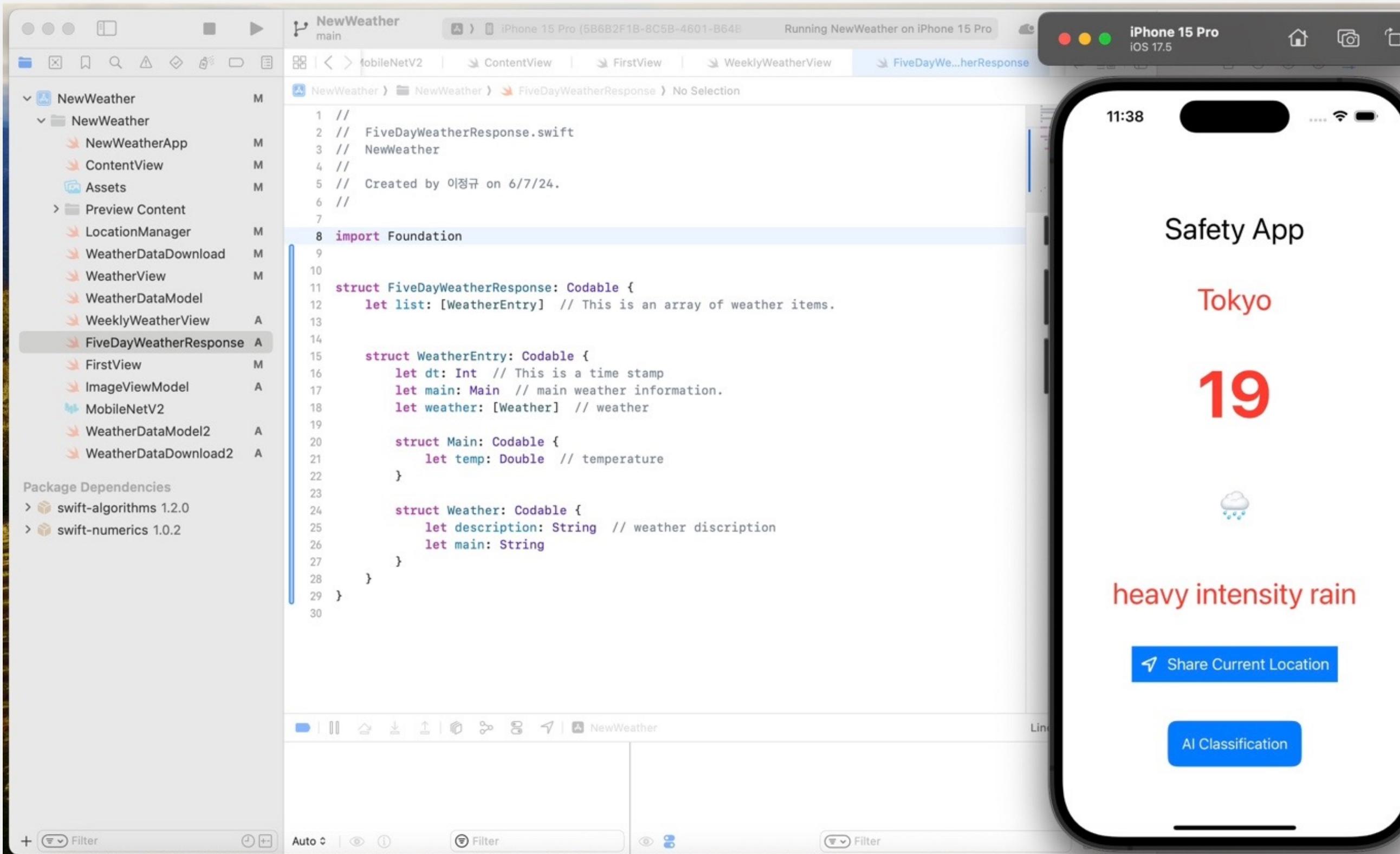
```
4 struct FirstView: View {
8     var body: some View {
15         VStack {
17             if let weather = currentWeather {
18                 Text(weather.name)
19                     .font(.largeTitle)
20                     .padding()
21                     .foregroundColor(
22                         weather.main.temp > 24 || ["Drizzle", "Thunderstorm", "Rain",
23                         "Snow"].contains(weather.weather.first?.main ?? "") ? .red
24                         : .black
25                 )
26                 Text(String(format: "%.0f", weather.main.temp))
27                     .font(.system(size: 75))
28                     .fontWeight(.bold)
29                     .foregroundColor(
30                         weather.main.temp > 24 || ["Drizzle", "Thunderstorm", "Rain",
31                         "Snow"].contains(weather.weather.first?.main ?? "") ? .red
32                         : .black
33                 )
34                 Text(iconList[weather.weather.first?.main ?? "" ?? "])
35                     .font(.largeTitle)
36                     .padding()
37                     .foregroundColor(
38                         weather.main.temp > 24 || ["Drizzle", "Thunderstorm", "Rain",
39                         "Snow"].contains(weather.weather.first?.main ?? "") ? .red
40                         : .black
41                 )
42                 Text(weather.weather.first?.description ?? "")
43                     .font(.largeTitle)
44                     .padding()
45                     .padding(.top, 20)
46                     .foregroundColor(
47                         weather.main.temp > 24 || ["Drizzle", "Thunderstorm", "Rain",
48                         "Snow"].contains(weather.weather.first?.main ?? "") ? .red
49                         : .black
50                 )
51             }
52         }
53     }
54 }
```

iPhone 15 Pro Simulator Preview:

- Display: Safety App
- Location: Tokyo
- Temperature: 19
- Icon: Rain
- Description: heavy intensity rain
- Buttons:
 - Share Current Location
 - AI Classification

03.App Developing Process

Explanation of the Swift Code

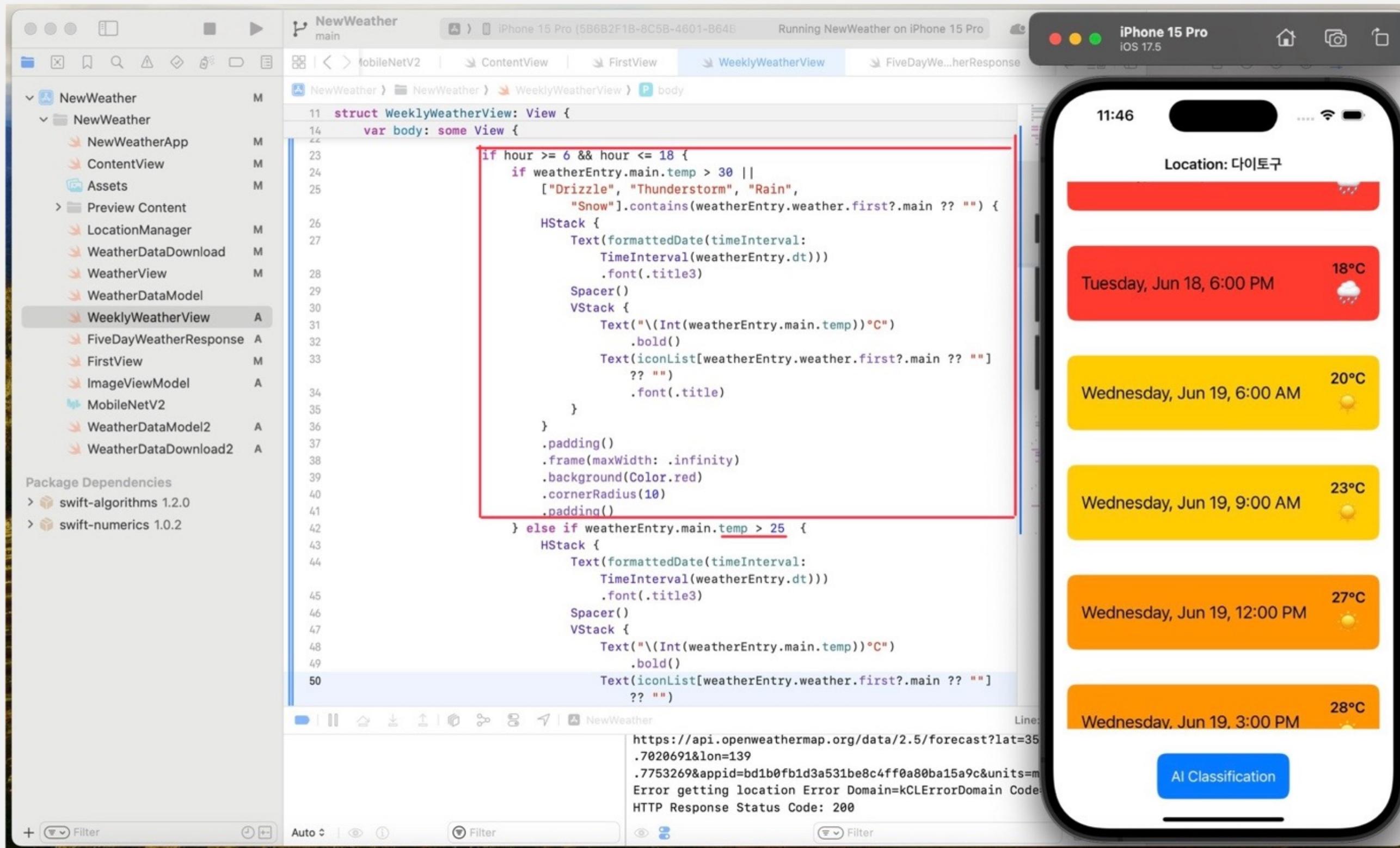


The image shows the Xcode interface with the 'NewWeather' project open. The left sidebar displays the project structure, including files like NewWeatherApp, ContentView, Assets, and several Weather-related files. The main editor window shows the 'FiveDayWeatherResponse.swift' file, which defines a Codable struct for weather data. The code includes nested structs for WeatherEntry, Main, and Weather, along with their properties. The right side of the image shows a screenshot of the 'Safety App' running on an iPhone 15 Pro. The app's interface includes a large red '19' representing temperature, the location 'Tokyo', and a weather description 'heavy intensity rain' with a rain icon. There are also buttons for 'Share Current Location' and 'AI Classification'.

```
1 //  
2 //  FiveDayWeatherResponse.swift  
3 //  NewWeather  
4 //  
5 //  Created by 이정규 on 6/7/24.  
6 //  
7  
8 import Foundation  
9  
10  
11 struct FiveDayWeatherResponse: Codable {  
12     let list: [WeatherEntry] // This is an array of weather items.  
13  
14     struct WeatherEntry: Codable {  
15         let dt: Int // This is a time stamp  
16         let main: Main // main weather information.  
17         let weather: [Weather] // weather  
18     }  
19  
20     struct Main: Codable {  
21         let temp: Double // temperature  
22     }  
23  
24     struct Weather: Codable {  
25         let description: String // weather description  
26         let main: String  
27     }  
28 }  
29  
30 }
```

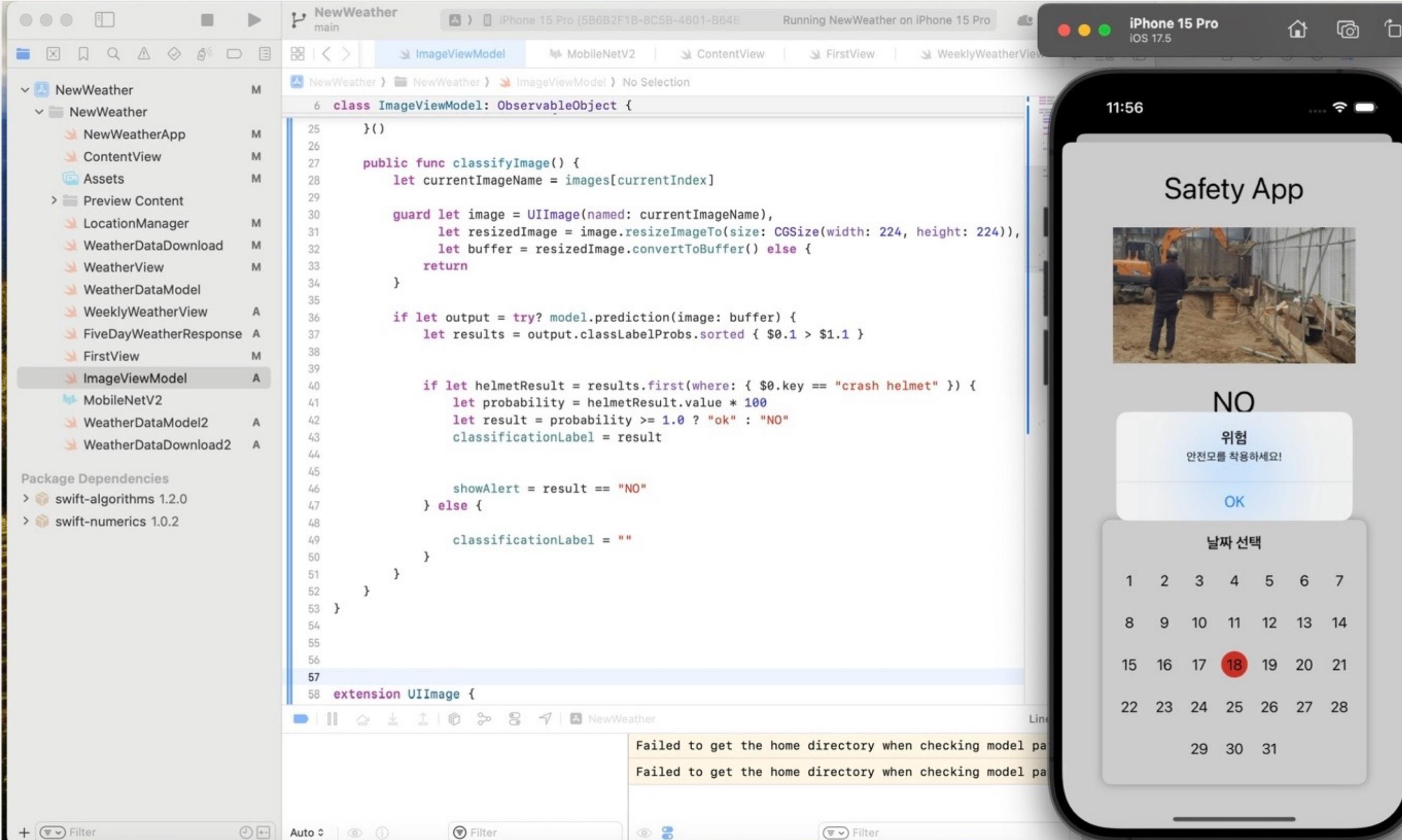
03.App Developing Process

Explanation of the Swift Code Used



03. App Development Process

Explanation of the Swift Code Used



The image shows a developer's workspace with Xcode open. On the left, the Project Navigator displays the 'NewWeather' project structure. The 'ImageViewModel' file is selected in the list, and its code is visible in the main editor window. The code implements a classification logic using a MobileNetV2 model to detect if a person is wearing a crash helmet. If the result is 'NO', an alert is shown. On the right, an iPhone 15 Pro screen shows the 'Safety App' interface. The app displays a camera feed of a construction site where a worker is not wearing a helmet. A large red 'NO' button is prominent, with a smaller 'OK' button below it. A Korean message at the bottom reads '위험 안전모를 착용하세요!' (Danger! Please wear a safety helmet!). Below the phone is a calendar showing the month of October, with the 18th circled in red.

```
6 class ImageViewModel: ObservableObject {
    ...
    public func classifyImage() {
        let currentImageName = images[currentIndex]
        guard let image = UIImage(named: currentImageName),
              let resizedImage = image.resizeImageTo(size: CGSize(width: 224, height: 224)),
              let buffer = resizedImage.convertToBuffer() else {
            return
        }
        if let output = try? model.prediction(image: buffer) {
            let results = output.classLabelProbs.sorted { $0.1 > $1.1 }
            if let helmetResult = results.first(where: { $0.key == "crash helmet" }) {
                let probability = helmetResult.value * 100
                let result = probability >= 1.0 ? "ok" : "NO"
                classificationLabel = result
                showAlert = result == "NO"
            } else {
                classificationLabel = ""
            }
        }
    }
}
extension UIImage {
    ...
}
```

Failed to get the home directory when checking model pa
Failed to get the home directory when checking model pa

03. App Development Process

Explanation of the Swift Code

The image shows a developer's workspace with Xcode open. On the left is the file browser, showing a project structure for 'NewWeather' with files like 'ContentView.swift', 'ImageViewModel.swift', and 'MobileNetV2.swift'. In the center is the code editor with the following Swift code:

```
58 struct ContentView2: View {
73     var body: some View {
100         VStack {
101             Text("날짜 선택")
102                 .font(.headline)
103                 .padding(.bottom, 10)
104
105             ForEach(dateGroups, id: \.self) { group in
106                 HStack(spacing: 10) {
107                     ForEach(group, id: \.self) { day in
108                         Text("\(day)")
109                             .frame(width: 30, height: 30)
110                             .background(day == Calendar.current.component(.day, from: today) ? (imageViewModel.classificationLabel == "NO" ? Color.red : Color.blue) : Color.clear)
111                             .foregroundColor(.primary)
112                             .cornerRadius(15)
113
114                     }
115                 }
116             }
117         }
118         .padding(.bottom, 10)
119     }
120
121     }
122
123     }
124
125     }
126
127     }
128
129     }
130
131     }
132
133     }
134     .alert(isPresented: $imageViewModel.showAlert) {
135         Alert(title: Text("위험"), message: Text("안전모를 착용하세요!"), dismissButton:
136             .default(Text("OK")))
137     }
138 }
```

The code defines a `ContentView2` view that displays a calendar for selecting a date. It uses `VStack` and `HStack` to layout the text and calendar. The calendar items are colored based on their classification ('NO' or 'OK'). An alert is shown if the user tries to select a date without a helmet.

On the right, an iPhone 15 Pro simulator is running the app. The screen shows the title 'Safety App' and a calendar for the month of October. A specific date, October 18th, is highlighted with a red circle. A small overlay window titled 'NO' contains the text '위험 안전모를 착용하세요!' (Danger Wear a safety helmet!). At the bottom, there are 'OK' and '날짜 선택' (Date Selection) buttons.

04. Sales Goal

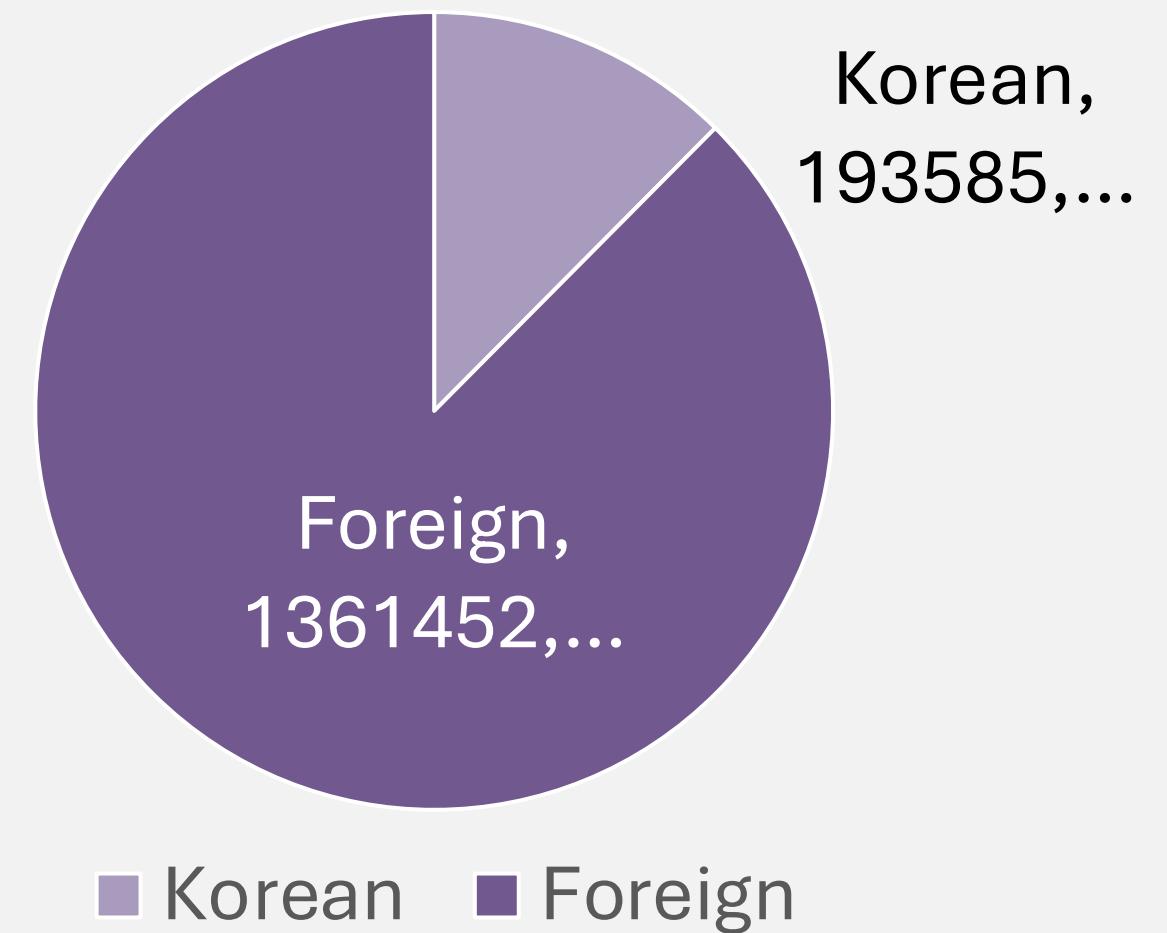
Market Analysis

Total number of construction workers : 1.5 million

Goal of users : 75,000 users (5% of the target customers)



Total Number of Construction
Workers in Korea



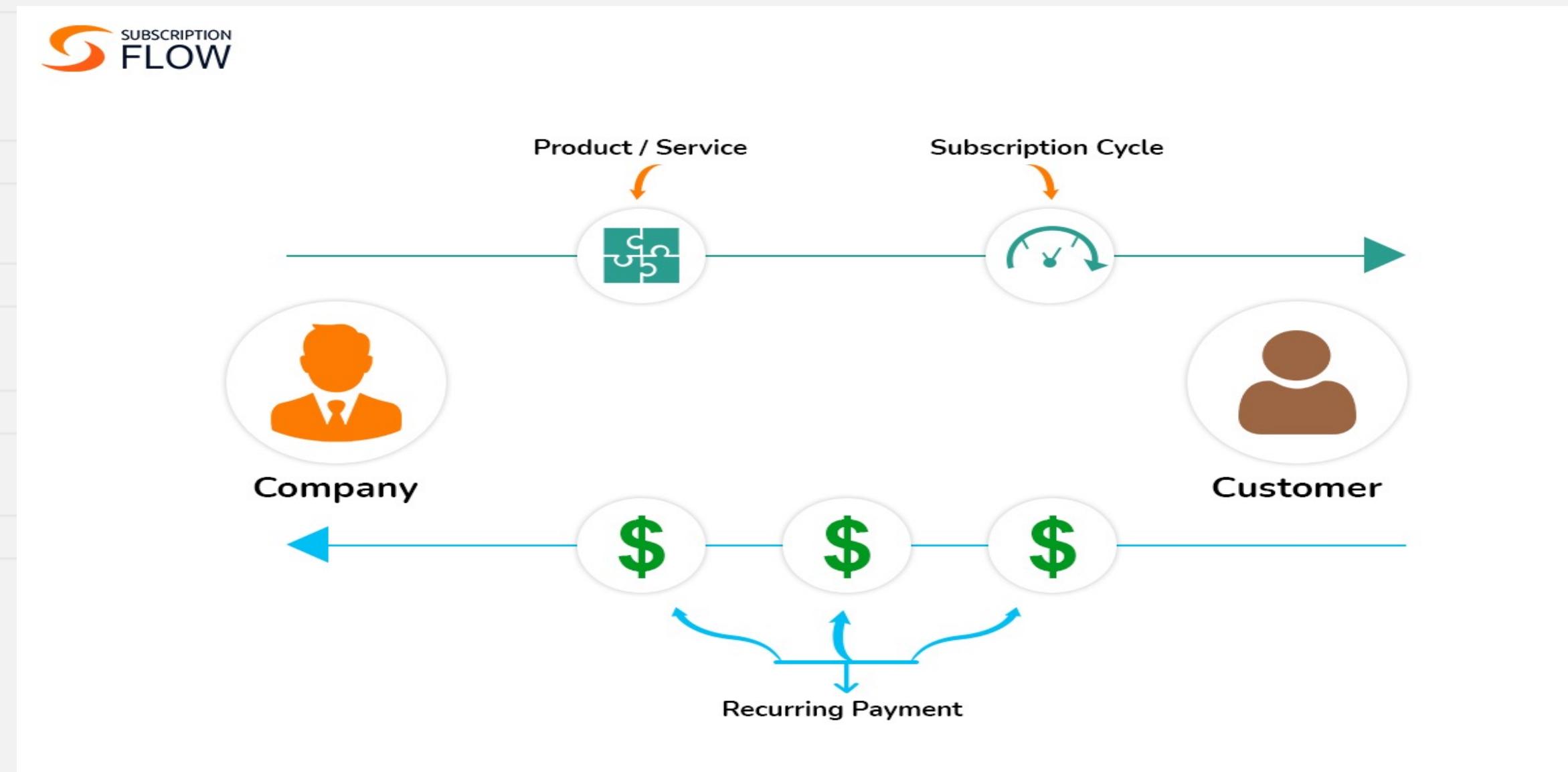
Source: KBS

Total Number of Construction Workers =
1,555,037

04. Sales Goal

Plan A) B2C Subscription Business Model

- 75,000 users (5% of the target customers)
- Annual subscription price : \$300 → Annual Recurring Revenue(ARR): \$22.5 million



04. Sales Goal

Plan B) B2G Exit



Exit through B2G deal

Valuation method : Comparable transactions method

Using the multiple of consumer services(8)
→ valuation = \$180 million

**Thank you for your
attention.**

Team 1