

Seminář 6: Navigace

Tvorba mobilních aplikací

Roman Vyjídáček

Obsah semináře

- Navigace



Navigace

Navigace

- Navigace určuje, jak se uživatel pohybuje mezi obrazovkami.
- Ve SwiftUI se používá komponenta NavigationStack (od iOS 16), dříve NavigationView.
- Navigace může být spouštěna uživatelem (např. pomocí NavigationLink) nebo programově (změnou cesty).
- Cílem navigace je zobrazení nové obrazovky nebo komponenty v reakci na akci uživatele.

Typy navigace

- Hierarchická (drill-down): Uživatel postupuje skrze vrstvy obsahu, například ze seznamu položek k detailu konkrétní položky.
- Laterální (tab-based): Přepínání mezi nezávislými sekcemi aplikace pomocí záložek.

Základní NavigationStack

- Komponenta zavedená v iOS 16, která nahrazuje NavigationView pro správu hierarchické navigace.
- NavigationStack obaluje obsah a spravuje navigační cestu.
- NavigationLink naviguje na jinou obrazovku s danou hodnotou.

Hlavní vlastnosti

- Lepší správa navigačního stavu.
- Podpora pro programatické řízení navigace.

```
NavigationStack {  
    NavigationLink("Tap Me") {  
        Text("Detail View")  
    }  
}
```

Problém s inline NavLink

- `NavLink(destination:)` okamžitě vytváří cílové View.
- Může to zpomalit aplikaci, pokud se destinace vytváří ve smyčce (např. `List`).
- Správnější přístup: použít `NavLink(value:)` + `navigationDestination()`.
- Cílové View se vytváří až v momentě, kdy se na něj naviguje.
- Zlepšuje to výkon a řeší problém se "zbytečnými" Views.

```
NavigationStack {  
    List(0..<5) { i in  
        NavLink("Číslo \$(i)", value: i)  
    }  
    .navigationDestination(for: Int.self)  
    { value in  
        Text("Zvolili jste \$(value)")  
    }  
}
```

Programatická navigace

- Pomocí @State nebo @Observable proměnné můžeme manipulovat s navigační cestou.
- Tím lze např. přeskočit do detailu bez uživatelské interakce.
- Vhodné pro reakce na logiku (např. po úspěšném uživatelském přihlášení).
- Proměnná path je pole hodnot, které reprezentují historii navigace.
- Každý append() v poli provede navigaci.

```
@State private var path = [Int]()

NavigationStack(path: $path) {
    VStack {
        Button("Zobraz 1") { path.append(1) }
        Button("Zobraz 2") { path.append(2) }
    }
    .navigationDestination(for: Int.self)
    { value in
        Text("Detail \$(value)")
    }
}
```

Navigace s různými datovými typy

- Pomocí `NavigationPath` můžeme uchovávat více různých typů v cestě.
- Např. `Int`, `String`, nebo vlastní modely (které implementují `Hashable`).
- Každý typ musí mít definovaný `navigationDestination`.
- Vhodné např. pro aplikace, kde typ navigace závisí na datech.

```
@State private var path = NavigationPath()

NavigationStack(path: $path) {
    List {
        Button("Zvol číslo") { path.append(42) }
        Button("Zvol text") { path.append("Swift") }
    }
    .navigationDestination(for: Int.self) { value in
        Text("Zvolené číslo: \(value)")
    }
    .navigationDestination(for: String.self) { value in
        Text("Zvolný text: \(value)")
    }
}
```

Modalní prezentace

Co je modalní prezentace?

- Zobrazení nové obrazovky na vrchu stávajícího obsahu, často pro získání uživatelského vstupu nebo zobrazení doplňujících informací.

Typy modalních prezentací

- Sheet: Obrazovka se vysune ze spodní části displeje a částečně překryje původní obsah.
- Full-Screen Cover: Nová obrazovka překryje celý displej.

Sheet

- Pokrývá část obrazovky.
- Uživatel může zavřít tažením dolů.
- Pro doplňkové úkoly nebo informace, které nevyžadují plnou pozornost uživatele.

```
struct ContentView: View {
    @State private var isSheetPresented = false

    var body: some View {
        Button("Zobrazit sheet") {
            isSheetPresented.toggle()
        }
        .sheet(isPresented: $isSheetPresented) {
            SheetView()
        }
    }
}

struct SheetView: View {
    @Environment(\.dismiss) var dismiss

    var body: some View {
        VStack {
            Text("Toto je sheet")
            Button("Zavřít") {
                dismiss()
            }
        }
        .padding()
    }
}
```

Full-Screen Cover

- Pokrývá celou obrazovku.
- Vyžaduje explicitní akci pro zavření (např. tlačítko “Zavřít”).
- Pro úkoly vyžadující plné soustředění, jako je přehrávání videa nebo vyplnění formuláře.

```
struct ContentView: View {
    @State private var isFullScreenPresented = false

    var body: some View {
        Button("Zobrazit na celé obrazovce") {
            isFullScreenPresented.toggle()
        }
        .fullScreenCover(isPresented:
$isFullScreenPresented) {
            FullScreenView()
        }
    }
}

struct FullScreenView: View {
    @Environment(\.dismiss) var dismiss

    var body: some View {
        VStack {
            Text("Toto je full-screen cover")
            Button("Zavřít") {
                dismiss()
            }
        }
        .padding()
    }
}
```

TabView (Tab Bar)

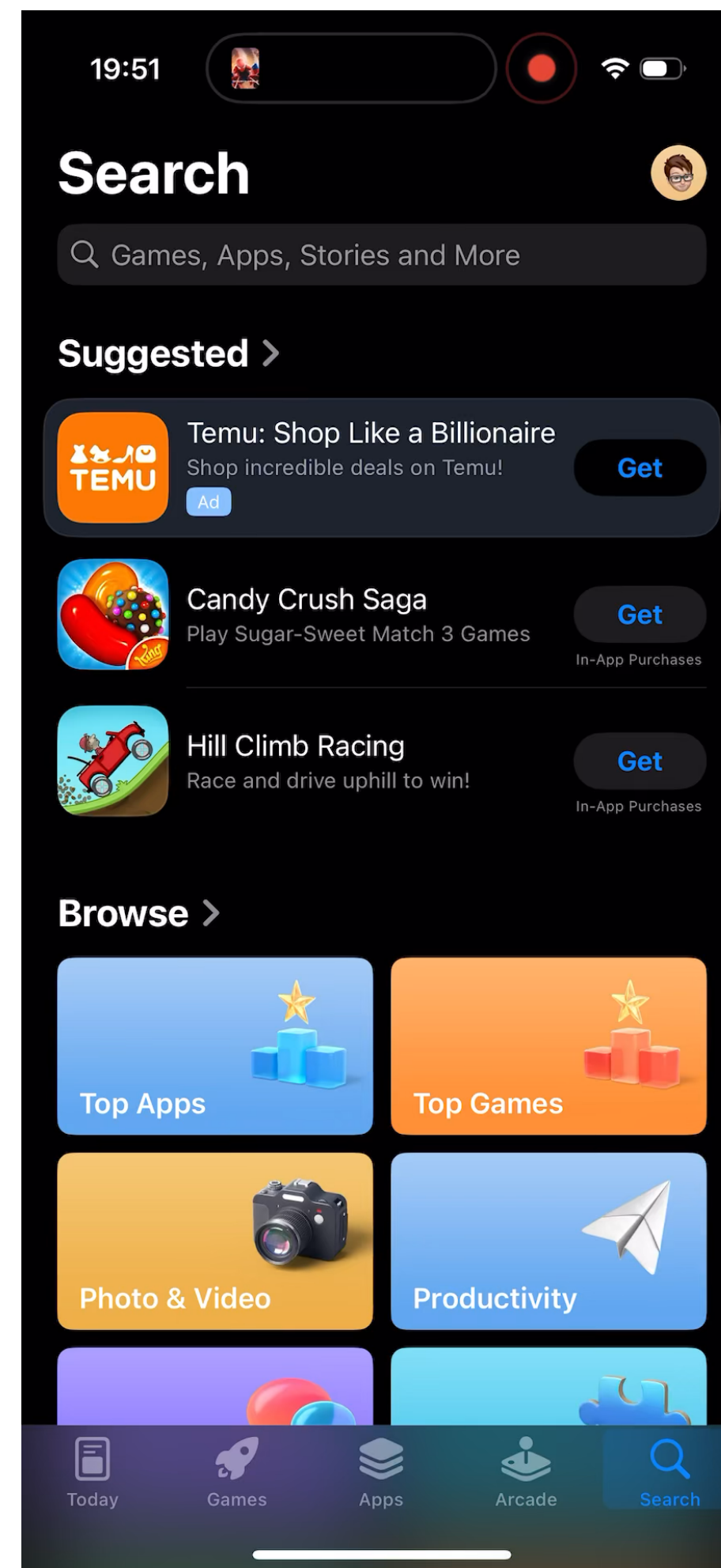
- TabView umožňuje rozdělit aplikaci do více sekcí pomocí spodní lišty.
- Každý pohled v TabView může mít svou vlastní navigaci (např. NavigationStack).
- TabView se hodí pro aplikace s více nezávislými obrazovkami: např. „Domů“, „Nastavení“, „Profil“.
- Každá karta má vlastní NavigationStack.
- Ikony a názvy sekcí jsou zobrazeny ve spodní liště (tab baru).

```
TabView {  
  NavigationStack {  
    Text("Domácí obrazovka")  
  }  
  .tabItem {  
    Label("Domů", systemImage: "house")  
  }  
  
  NavigationStack {  
    Text("Nastavení aplikace")  
  }  
  .tabItem {  
    Label("Nastavení", systemImage: "gear")  
  }  
}
```

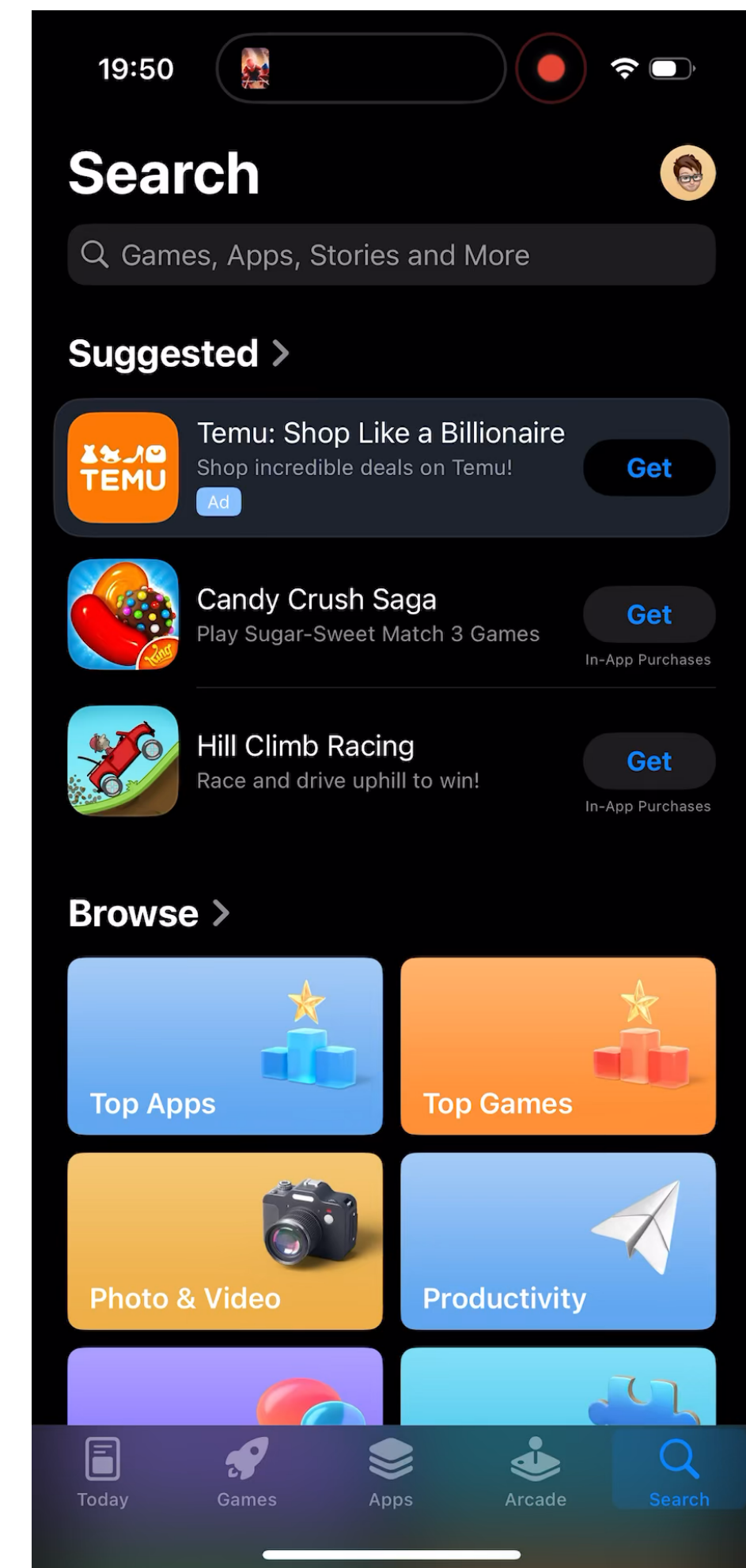
Ukázka navigace v iOS



Tab Bar



Push

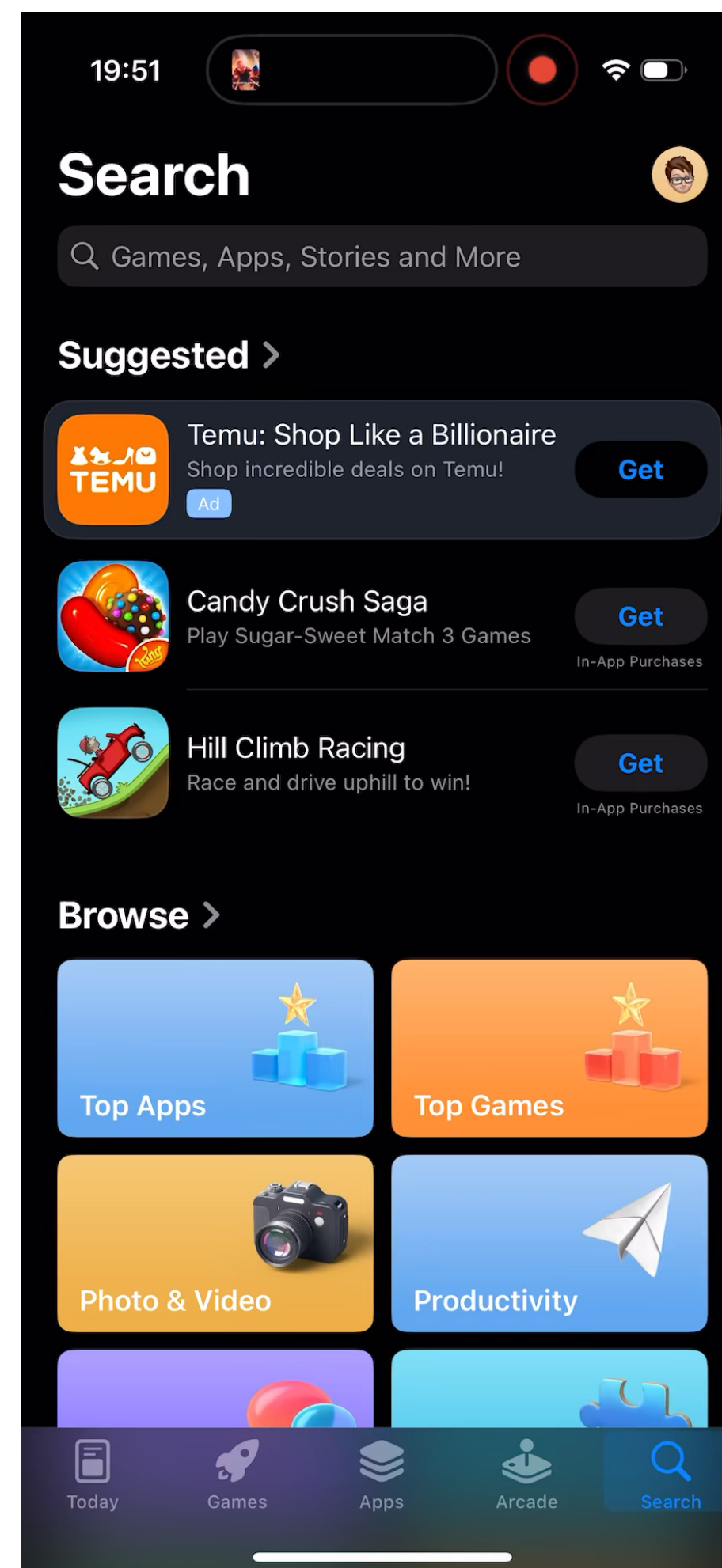


Sheet

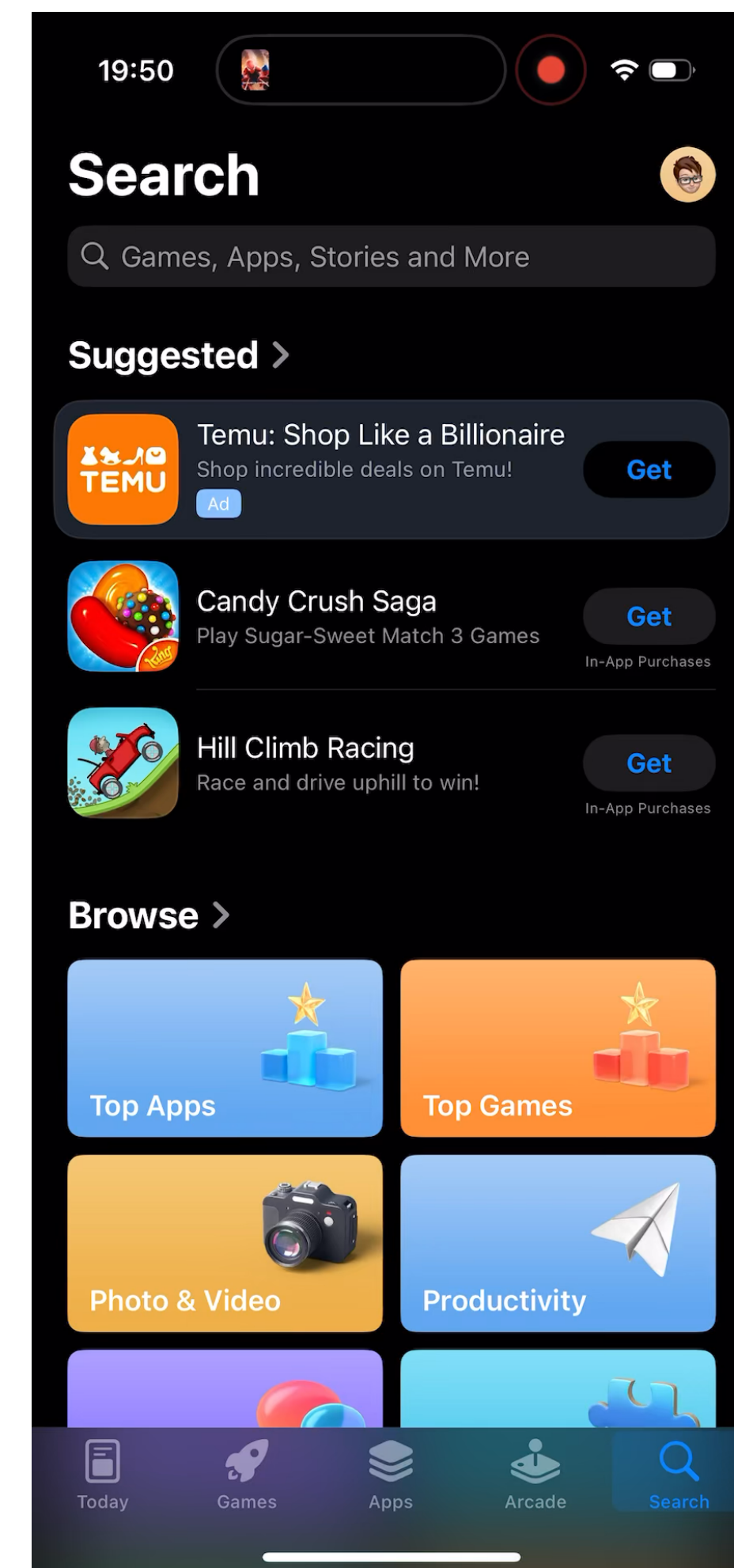
Ukázka navigace v iOS



Tab Bar



Push



Sheet

Doporučené přednášky

- <https://developer.apple.com/videos/play/wwdc2022/10054/>
- <https://developer.apple.com/videos/play/wwdc2022/10001/>

Alerty

- Alert je systémové vyskakovací okno, které informuje uživatele nebo žádá potvrzení.
- Slouží pro:
 - potvrzení akce (např. smazání),
 - upozornění na chybu,
 - zobrazování důležitých informací.
 - Zobrazují se modálně a blokují ostatní interakce.
- Základní typy tlačítek:
 - default – běžné potvrzení,
 - cancel – zavření bez akce,
 - destructive – varuje uživatele (červené tlačítko).

Alert

- Alert se zobrazí, pokud je `showAlert == true`.
- Obsahuje nadpis a dvě akce – potvrzení a zrušení.

```
@State private var showAlert = false

Button("Smazat položku") {
    showAlert = true
}
.alert("Opravdu chcete smazat?",
    isPresented: $showAlert) {
    Button("Ano", role: .destructive) {
        // akce mazání
    }
    Button("Zrušit", role: .cancel) { }
}
```

Pokročilejší alert s textem

- message: umožňuje přidat vlastní text pod nadpis.
- Hodí se např. pro chybová hlášení nebo varování.

```
.alert("Přihlášení se nezdařilo",  
      isPresented: $showError) {  
    Button("OK", role: .cancel) { }  
} message: {  
    Text("Zadané údaje nejsou správné.")  
}
```

ConfirmationDialog

- ConfirmationDialog je alternativou k Alert, vhodnou pro nabídku více voleb.
- Zobrazí se jako akční list (Action Sheet) – typicky ze spodní části obrazovky.
- Hodí se pro situace typu: „Vyberte způsob“, „Zvolte akci“, „Sdílet přes...“
- Hlavní výhody:
 - Lze definovat více tlačítek s různými rolemi.
 - Může obsahovat title, volitelný message a skupinu akcí.

ConfirmationDialog

- titleVisibility určuje, zda se má nadpis zobrazit.
- Každé tlačítko může mít roli (default, cancel, destructive).
- Vhodné pro iPhone i iPad – adaptivní zobrazení.

```
@State private var showDialog = false
@State private var selectedOption = ""

Button("Zobraz nabídku") {
    showDialog = true
}
.confirmationDialog("Vyber akci",
    isPresented: $showDialog,
    titleVisibility: .visible) {

    Button("Upravit") {
        selectedOption = "edit"
    }

    Button("Smazat", role: .destructive) {
        selectedOption = "delete"
    }

    Button("Zrušit", role: .cancel) { }
}
```

Kdy použít Alert vs. ConfirmationDialog?

Situace	Použij
Potvrzení jediné akce	Alert
Upozornění na chybu	Alert
Nabídka více možností	ConfirmationDialog
Výběr akce (např. „Sdílet přes...“)	ConfirmationDialog
Potvrzení destruktivní akce	Alert s role: .destructive

Úkoly

<https://vyjidacek.cz/tmai/seminar6.php>

