Embedded Systems
University of Hiroshima V. Poupet

# An Introduction to iOS - Homework Assignment

## Object Programming in *Swift*

**1** Define a class `Vehicle` with the following properties :
— `name` (of type `String`)
— `maxSpeed` (of type `Int`)
— `canFly` (of type `Bool`)
Add an initializer `init(name: String, maxSpeed: Int, canFly: Bool)` for that class.

**2** Add a method `isFasterThan(other: Vehicle)` to the class `Vehicle` that returns a boolean indicating if the considered object is faster (larger `maxSpeed`) than the other object given as argument.

**3** Define two classes `Car` and `Airplane` as subclasses of `Vehicle`. The class `Car` has an extra property `numberOfWheels` (of type `Int`) and the class `Airplane` has an extra property `numberOfEngines` (of type `Int`).
Write an appropriate initializer for each class.
**Hint :** cars cannot fly and planes can fly, so it's not necessary to have an argument of *init* for `canFly`. The initializer should define the new property and then call the initializer for the parent class.

**4** Assuming that the previous classes have been properly defined, declare the three following variables :
— `boeing`, an airplane named ”Boeing 747” with 4 engines and a max speed of 920 km/h ;
— `roadster`, a car named ”BMW M Roadster” whose max speed is 249 km/h ;
— `tyrell`, a car named ”Tyrell P34” with 6 wheels and a max speed of 300 km/h.

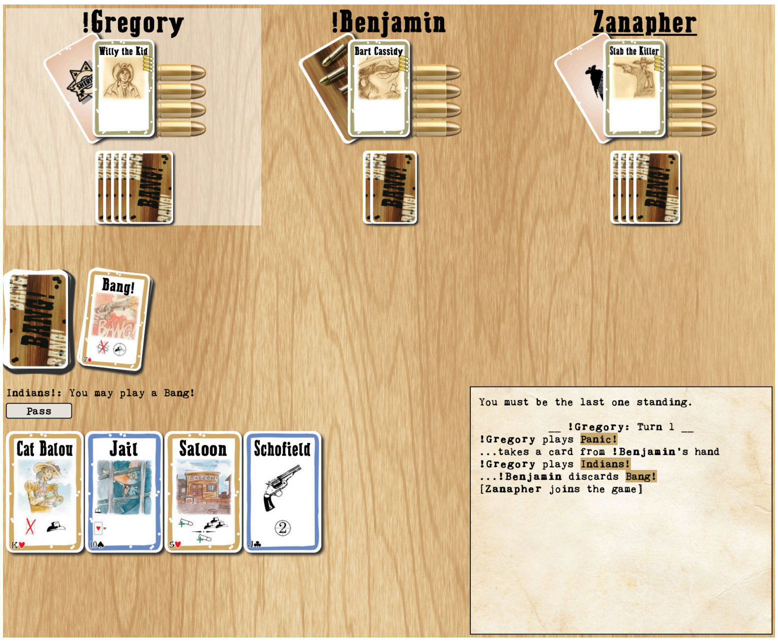
## 1 User Interface

Figure 1 represents the main view of an iOS app for a multi-player card game.

**5** Describe the view hierarchy that you would use to represent such a view, starting from the main view containing all the views on the screen, down to elementary elements such as `UIImage`, `UILabel`, `UIButton`, etc.

**6** How many view controllers should be used to manage such a view hierarchy ?

**7** According to the view hierarchy that you described in the previous question, describe how the application would determine the view that receives a touch event that starts on the “ Slab the Killer ” card (Zanapher's character card).
Describe which view is tested first for a hit, and which subsequent views are tested in order, and determine which view will eventually be considered as being hit by the touch. **Hint :** Follow the hit test protocol.



Indians!: You may play a Bang!

```
You must be the last one standing.

          __ !Gregory: Turn 1 __
!Gregory plays Panic!
...takes a card from !Benjamin's hand
!Gregory plays Indians!
...!Benjamin discards Bang!
[Zanapher joins the game]
```

FIGURE 1 – A screenshot of a card game app.

**8** Assuming that the view hit by the touch from the previous question doesn't handle touches by itself, describe (according to the view hierarchy and view controllers that you described previously) which responders get a chance to handle the touch (and in which order).
**Hint :** Follow the responder chain.

## 2 Events

We want to write an iOS app to draw line segments on the screen.

Whenever the user touches the screen, a segment is drawn from the point where the touch started to the point where the touch was released (so the user should drag his finger from the starting point to the ending point).

**9** Which are the 4 defined methods to track and react to user touches on a view ?

**10** Is it important to declare the `firstResponder` for this application ? (justify your answer)

**11** Describe what the methods `touchesBegan:withEvent:` and `touchesEnded:withEvent:` of the view controller should do.

Describe in particular which variables need to be set and read by each function and when the actual line drawing is performed.

**Note :** You can either write commentated *Swift* code or describe what each method does in plain English.

**12** Describe what the `touchesCancelled:withEvent:` should do. According to the answers of the previous questions, which variables should it delete ?