

# Crossroads: session 2

Group presentation, more iOS dev, big data, and more

# Corrections

- Exercise regularly and eat healthy
- ...
- ...
- ...
- Profit!

# Group a

- Interview a chemistry professor and summarize his area of expertise
- Describe inheritance, in terms of OOP

# Group b

- Interview a physics professor and summarize his area of expertise
- Describe method overriding

# Group c

- Interview a computer science professor and summarize his area of expertise
- Describe and compare: class variable & instance variable

# Group d

- Interview a biology professor and summarize his area of expertise
- Describe the concept of “interface” in OOP

# Group e

- Interview a computer science professor and summarize his area of expertise
- What is a design pattern, in terms of OOP

# What's the Point?

- Learn to listen and talk to someone you don't know
- Learn to summarize and present in a clear manner
- Object Oriented Programming is an important concept used in everyday work



# Graduate School in the U.S.

- What to study, why go? (Be honest!)
- Survey government scholarships first (Read the small print)
- GRE, GMAT, TOEFL
- Do everything yourself (But seek help & advice)
- You need 3 letters and an essay
- Degree granting programs can be... diverse

# Graduate School in the U.S.

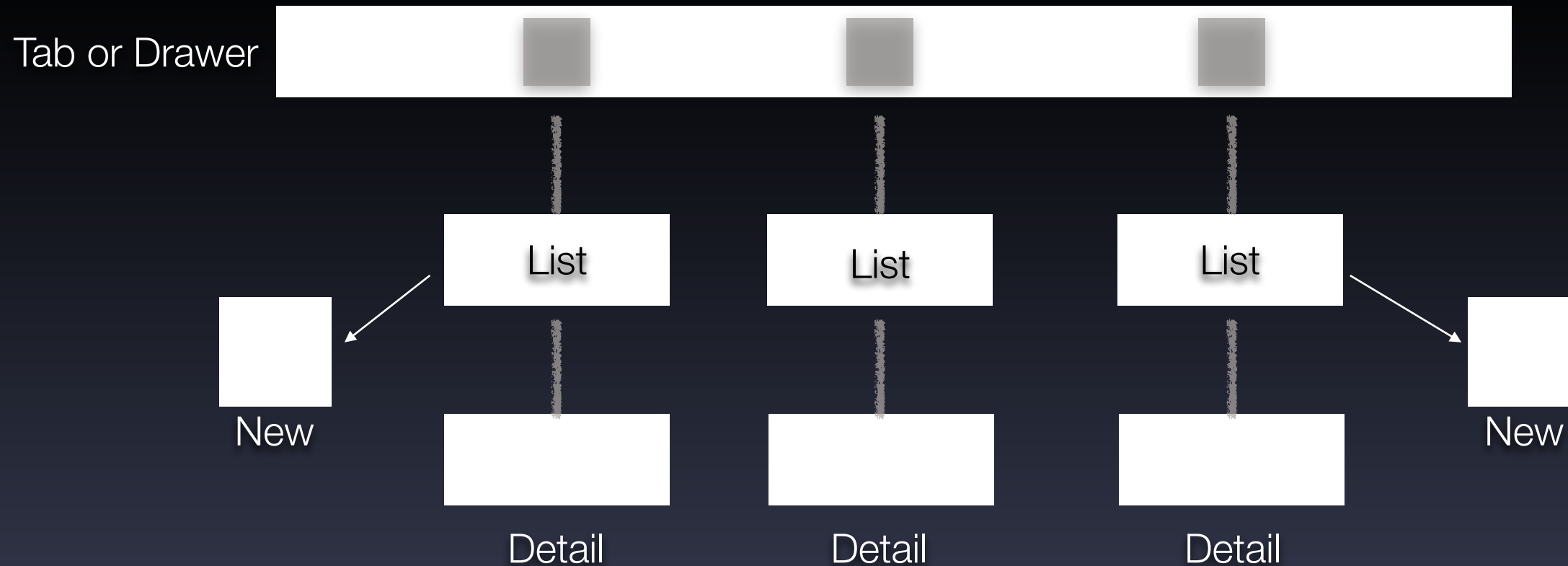
- It's really not that hard
- A lot of what you read online is bulls\*\*t
- Know the difference: Networking & Just getting drunk
- Optional Practical Training (OPT) and H1B

# Recap for iOS Development

- View Controllers are the basic building blocks
- Each class is implemented by .h and .m files
- “Ctrl + drag” to connect Storyboard object to .h file
- The “delegation” pattern for objects
- We use a cloud-based backend (MBAAS)

# New Concepts

- Use “segues” to transition between views
- Learn about localizing your app
- Learn how to implement a “function” into actual “code”, using basic chat as an example



- Visualizing a typical app flow
- UX flow can be designed independent of core logic, but in reality...

# Chat Example: Data

- What we start with: “User” objects
- Create the data classes we want: “Conversation” and “Message”
- Think about specific fields: “Conversation” might hold last message and time for convenience, “Message” should have a broadcast type...etc

# Chat Example: Interface

- We need the three most basic screens:
- A list of conversations
- A view to start a new chat
- The list of messages in a conversation



# Chat Example: Business Logic

- Next we have to consider the actual functions in each view:
- Conversation list is simple, list and present each conversation, with tapping into messages or new conversation.
- Messages need to present messages under a specific conversation, separating them into 3 types (self, other, broadcast), also need to input and send new messages.
- New conversation presents a list of Users depending on the logic of our project, with multiple selection, and finally cancel or create.
- Independent of view, we also need an update mechanism.



# Chat Example: Getting to Work

- Setup our data scheme in Parse
- Use Storyboard to layout the views
- Think about the methods we will need to implement

# Next Time

- C & Android (Java) development
- Group discussion & presentation
- What would you like to learn?

# Questions, Comments & Corrections

Chi-sheng Jan (詹啟聖)

- [jan0054@gmail.com](mailto:jan0054@gmail.com)
- You can write in Chinese or English