Greg Tropino

Challenge 2 video response

Section 0:

.1) The purpose behind object oriented design is to be able to organize your program in a clearer, more precise manner to avoid bugs, avoid dead ends, avoiding issues that could have been easily thought about and navigated around if planned more thoroughly.

.2) It is advantageous because once everything is planned out and designed it will be overall much quicker to do the programming since you will have a better understanding as to what is expected, and what needs to be done.

.3) The difference between the waterfall approach and agile is that there is much more flexibility with the agile approach since things are typically not static when it comes to software development, as things change or need to keep developing afterwards. Waterfall design is much more firm in the steps, where you must complete each part before you move on to the next, where as you are combining all the steps in the agile development to be more receptive to change at each step. Iterative is an incremental cycle, each including analysis, design and programming, where it is in weeks, not months. We use it in a manner that is not perfect, as it shouldn’t be, but to be flexible, and to try to have oversight as to what we are doing at each step.

Section 1:

1.1) There are numerous types of programming languages, some of them include Logic programming like Prolog, Functional programming language like Haskell, procedural programming and Object oriented programming like Java, objective c among many others

1.2) You will typically see several different properties in a computer object, such as the object having an identity, where it can be independent from other objects, attributes, where it describes anywhere from the size or contents and behavior where it will be determined as to how it will act.

1.3) Classes are very similar to blueprints in the same that you use both of them as frameworks to create objects as described by the class or the blueprint. Classes will determine as to what the attributes, behaviors and name of the object is, but the class is not the object itself, however the class can create as many objects as it wants.

1.4) Abstraction is the ability to conceptualize a simple idea and focus on the idea of it, not the irrelevant details.

1.5) It prevents modification of data unless its done through the appropriate method, it helps ensure tracking and ease of future development of the program in case certain structures are modified since encapsulation helps maintain fewer paths for methods to modify data

1.6) Inheritance is when a class takes the same abilities another class already has. It is useful when you are creating multiple classes that all have some fundamental similarities between each other.

1.7) The idea behind polymorphism is to get the program to take the correct action/behavior when it could take several different forms or paths . When classes inherit from another polymorphism helps us know how to do baseline interactions between numerous objects when we are unsure as to which class it was created from.

Section 2:

2.1) Gathering the requirements, describing the app, identify the main objects, describe the interactions, and creating a class diagram. With a clearer understanding as to what the app is supposed to do, what it needs to do, and in what order it needs to do it you can have a better overview as to how your program should be looking as you go along.

2.2) At bare minimum you should have some type of documentation going over the essential requirements that your app must perform or include.

2.3) UML stands for unified modeling language and is a diagraming technique for a object oriented design system. You should visualize it due to it becoming easier to identify what should be objects, classes and how they will or should interact with each other.

Section 3:

3.1)

Title: Creating event on iPhone

Actor: User

Scenario:

1. Locate the calendar app in your phone and tap it to open
2. Find and press the “+“ sign in the upper right hand corner of the application
3. Enter the appropriate information regarding even (title, location etc.)
4. Select the calendar type (Default, home, work etc.)
5. Be sure to establish the start and end time (especially the date) of the event
6. Set time zone that location will be held in (in case out of state)
7. Set any contacts you want to have invited to share the event
8. Upon completion of all the necessary info hit “Done” in the upper right hand corner.

3.2) YouTube, since the primary actor is the person who created and uploaded the content.

3.3)

Title: Creating event on iPhone

Actor: User

Scenario:

1. Add event
   1. Extension, if event is overlapping with another event review which event has priority
2. Enter appropriate information regarding title, location etc.
3. Be sure to establish the start and end time (especially the date) of the event
4. Upon completion of all the necessary info hit “Done” in the upper right hand corner.

3.4) They all show which actor(s) interact with what use cases.

3.5) As a user I want to have public busses listed so I can figure out how to avoid using my car as much. As a user I would want train schedules incorporated into travel times to let me know what time I have to be at the station. As a user I’d like the GPS self location to be more accurate so that I can travel easier in the city. As a user I’d like directions to be transferred to my sunglasses so that I don’t have to look down at my phone. As a user I want the locations of places I’m looking for to be more precise so I don’t get potentially lost.

Section 4:

4.2) Event, Contacts, Calendar types (Home, work, default), Time Zone, Date Range

4.3)

Identify the relationships among the classes you found above, create a conceptual model where you diagram these relationships and then upload a picture of your model below.

4.4)

Identify the responsibilities of the classes you found above. List them here.

Section 5:

5.1) Twitter app diagram

Class name: Feed

Attributes: showlist, category/trend following, follow name

Operations: getrecenttweet, getuseroftweet, gettimeoftweet

Class name: Tweet

Attributes: user name, content (140 characters), image attached, datestamp

Operations: Setfeed(Boolean), WhichCategory

I think it is easier to come up with attributes as opposed to behaviors since it is easier to think as to what an item must contain as opposed to what an item must do.

5.2) It becomes a lot easier to separate behaviors and operations from attributes.

5.3) A constructor is a method for object creation, and destructor is the method to delete or release an object to free up resources or close any connection that object may have to something else (like a database or files etc.)

5.4) Twitters tweets only being 140 characters, a car class always having 4 tires, a human always having 2 legs.

Section 6:

6.1) Inheritance is taking a shared foundation that can be spread across to one or more classes. It is very useful because if you come across numerous classes that share a lot of the same essentials you can save a lot of time by creating a parent class for just those shared traits and then being able to “transfer” those into the sub classes.

6.2) The phone app inherits the contacts list, almost all apps share a foundational inheritance from the object class that comes with xcode, the calculator app inherits from the math class.