EECS4443 review

Jerry Wu

2023-12-11

Contents

Section 1

Quiz answers

Quiz 1: UI design, intro to activities, etc

1. A design is efficient, if:

   a) it accomplishes the user's task in a satisfying way.

   b) it accomplishes the user's task.

   c) **it accomplishes the user's task without creating additional problems.**

2. Using auto-complete may help you satisfy which UX (user experience) design principle?

   a) **Minimize User Input**

   b) KISS (keep it simple stupid!)

   c) Make Navigation Intuitive

3. Android Applications usually follow this architectural style:

   a) **MVVM (a variant of MVC)**

   b) Layered

   c) Plugin

4. A design is effective, if:

   a) it accomplishes the user's task without creating additional problems.

   b) it accomplishes the user's task in a satisfying way.

    c) **it accomplishes the user's task.**

5. A design is exciting, if:

    a) it accomplishes the user's task without creating additional problems.

    b) **.it accomplishes the user's task in a satisfying way.**

    c) it accomplishes the user's task.

6. When an interface puts an interactive element in a place where another element is most commonly found in other applications, what is this UX antipattern called?

    a) Misdirection

    b) **Bait and Switch**

    c) Roach Motel

7. In Android, an `Activity` corresponds to:

    a) a transition between screens.

    b) **a screen**

    c) a use case

8. If an `Activity` has been "Destroyed",

    a) the `onSaveInstanceState()` method is invoked automatically, so that we can always recover the Activity.

    b) it is impossible to be recovered and has to be created again from scratch.

    c) **our code needs to explicitly call the `onSaveInstanceState()` in order to recreate the Activity for certain circumstances.**

9. When an `Activity` is Paused,

    a) it is completely visible to the user, but inactive.

    b) it is hidden from the user and can be destroyed by the system.

    c) **it is not in the user's main focus and may be destroyed by the system.**

10. If an `Activity` is Started,

    a) it is not fully created, but not yet visible to the user.

    b) **it is visible, but not yet running.**

    c) it has been created, but it is hidden from the user.

Quiz 2: Views, layouts, etc

1. If I flip the orientation of my device, when I have a `GridView`,

   a) It will maintain the same number of rows and columns.

   b) It will maintain the same number of rows and columns, but it will stretch or screen the dimensions of the items to better fit the screen.

   c) **It will adapt the number of rows and columns to better fit the screen.**

2. A `Toast` is a popup dialog that

   a) Requires a user's action to disappear

   b) Provides important feedback, like a warning or an error to a user.

   c) **Automatically disappears after some time.**

3. A layout is used **only** to

   a) Define all the UI elements used in an application

   b) **Define all the UI elements of an activity and how they are organized.**

   c) Define the position of UI elements when the orientation of the screen changes.

   d) All of the above

4. A `ListView` is used to create and present lists in an `Activity`. When the `ListView` is created and presented,

   a) It loads all data and create the visualizations for all list items.

   b) **It loads all data in the Adapter but creates visualizations only for the visible items.**

   c) It loads only the data that will be visible, but creates placeholder items (UI elements) for all possible data items.

5. In this layout, the elements are organized "in rows and columns".

   a) Linear layout

   b) Relative Layout

   c) **Grid Layout**

6. In this layout, I can assign elements a certain "weights" which defines the space they will take in the layout relative to other elements in the layout.

   a) **Linear layout**

   b) Relative Layout

   c) Grid Layout

7. In this layout, the elements are organized using anchors, like other elements, the parent elements, or specific positions.

   a) Linear layout

   b) **Relative Layout**

   c) Grid Layout

8. In the Manifest, we can declare that our application requires permission to access

   a) **any kind of resource external to the application, which includes Internet sources or the storage of the device.**

   b) only the external storage of our device.

   c) only data sources on the Internet.

9. What is the class that is used to as a link between the layout and the data source?

   a) Adapter

   b) **AdapterView**

   c) Intent

10. When I am using a ListView, which of the following statements is true? (only one is true)

    a) I can have only single line textual items.

    b) I can have both single and multiple line items, but only textual items.

    c) **I can have items that combine multiple data types, text, images, icons, action buttons, and span many lines.**

Quiz 3: Software testing methods

1. In software testing, these modules accept test data from high-level modules and pass computed data.

   a) **Stubs**

   b) Test cases

   c) Drivers

2. In this type of software testing, the system is tested in parts usually following the order of development.

   a) Big Bang

   b) **Incremental**

   c) Alpha testing

3. Which MotionEvent action is called when a second finger touches the screen?

   a) `ACTION_DOWN`

   b) **ACTION_POINTER_DOWN**

   c) `ACTION_MOVE`

   d) `ACTION_POINTER_UP`

4. In this type of testing, users provide feedback on an incomplete version of the system.

   a) **Alpha testing**

   b) Beta testing

   c) Exploratory testing

5. Which of the following interactions is NOT a gesture?

   a) Swipe

   b) Flick

   c) **Click**

6. How many test cases do we typically need to test a software functionality?

   a) A lot!

   b) Exactly three (happy path, boundary path, exceptional path)

    c) **We need one test case for each invalid and boundary value and enough test cases to cover all valid equivalence classes.**

7. What is considered a "gesture" in the context of Android development?

    a) A "digital" handshake used for electronic verifications

    b) A hand motion, like a wave, capture by the camera of a mobile device.

    c) **Any tactile, i.e., using touch, interaction with the screen of the mobile device.**

8. A Unistroke is

    a) A single straight line

    b) **A continuous single line that represents a character or another symbol.**

    c) A line to represent the number 1.

9. In the Espresso Testing Framework, what is an Idling Resource?

    a) A device resource, like CPU, memory, and disk, that does not perform any task at the moment.

    b) An activity that is not visible at the moment.

    c) **An object that represents an asynchronous task running in the background.**

10. A "pinching" movement on the screen requires this class to be captured.

    a) `GestureDetector`

    b) Either `GestureDetector` or `ScaleGestureDetector`

    c) **ScaleGestureDetector**

Quiz 4: Fragments, SPE, etc.

1. What is the difference between profiling and monitoring?

    a) Profiling is static, while monitoring is dynamic.

    b) Profiling is for software, while monitoring is for hardware.

    c) **Profiling is measuring at development time, monitoring is measuring after deployment.**

    d) They are the same concept.

2. What class do I use to pass data between two Activities?

    a) `Bundle`

    b) `TouchListener`

    c) **Intent**

3. What can be detected by the gyroscope?

    a) The acceleration force

    b) **The rate of rotation.**

    c) The geomagnetic field strength

4. One of the goals of Software Performance Engineering is to

    a) Allocate more resources to the software to improve its performance.

    b) Improve the efficiency of the developers.

    c) **Reduce the maintenance costs necessary to resolve performance issues.**

5. How can I invoke a transition between two activities?

    a) **By creating an Intent object to start the new activity and pass data between the activities.**

    b) by calling `Bundle.startActivity()`

    c) by calling `Intent.startActivity()`

6. How can I create transitions between fragments?

    a) With Intents like in the Activities.

    b) By declaring the fragments as "containers" in the layout XML.

c) **By beginning a transaction through the Fragment manage and replacing the current Fragment with another one.**

7. When an operation is too short, which profiling method do we prefer?

   a) Sampling

   b) **Instrumentation**

   c) Both are equally acceptable.

   d) None of the above

8. Which of the following are acceptable profiling methods? (multiple answers, wrong answers will receive -25

   a) Basic profiling

   b) **Sampling**

   c) **Instrumentation**

   d) Hybrid

9. What is a fragment?

   a) **a "sub-activity"**

   b) a popup dialog

   c) a simple UI element like a menu

10. Why do we say that the performance being perceived by the users is a limitation of Software Performance Engineering?

   a) **Because during development we cannot test the performance in the same conditions, same environment and under the exact same scenarios as the users.**

   b) Because users cannot provide technical feedback about the performance of the software.

   c) Because users are not involved in the development of the software when performance requirements are implemented.

Section 2

Sample final exam solutions

Multiple choice questions

1. In the manifest of an Android application, among other things, we specify

   a) **the activities of the application.**

   b) the UI elements of the application.

   c) the string constants used in our code.

2. When an Activity is Paused

   a) it is hidden from the user and can be destroyed by the system.

   b) **it is not in the user's main focus and may be destroyed by the system.**

   c) it is completely visible to the user, but inactive.

3. When an interface puts an interactive element in a place where another element is most commonly found in other applications, what is this UX antipattern called?

   a) Misdirection

   b) **Bait and Switch**

   c) Roach Motel

4. In the XML file of a layout, we typically include

   a) The dependencies of our app

   b) The activities of our app

c) **The elements of an activity and their attributes**

5. Which of the following UI elements allow me to make a single choice in the presence of multiple options.

   a) Spinners and Action Buttons

   b) Radio buttons and Checkboxes

   c) **Spinners and Radio buttons**

6. In the Espresso Testing Framework, a ViewMatcher allows me to

   a) **Find a view and focus on it.**

   b) Perform an action on a view.

   c) Test the value of a property or data of a view.

7. In this layout, the elements are organized "one after the other".

   a) **Linear layout**

   b) Relative Layout

   c) Grid Layout

8. In this type of software testing, the system is tested in parts usually following the order of development

   a) Big Bang

   b) **Incremental**

   c) Alpha testing

9. What is considered a "gesture" in the context of Android development?

   a) A "digital" handshake used for electronic verifications.

   b) A hand motion, like a wave, capture by the camera of a mobile device.

   c) **Any tactile, i.e., using touch, interaction with the screen of the mobile device.**

10. This type of performance test allows us to test the behavior of the system for a given and expected workload.

    a) Capacity testing

    b) **Load testing**

    c) Endurance testing

Section 3

UI Design problems

Problem 1

You are given the UI of an activity. In the picture of the UI, there are 5 design or UX problems. Point them out, explain why they are a problem and propose a solution to fix each problem. (10 points) *Note: Some problems can be found multiple times. You need to identify 5 different problems.*
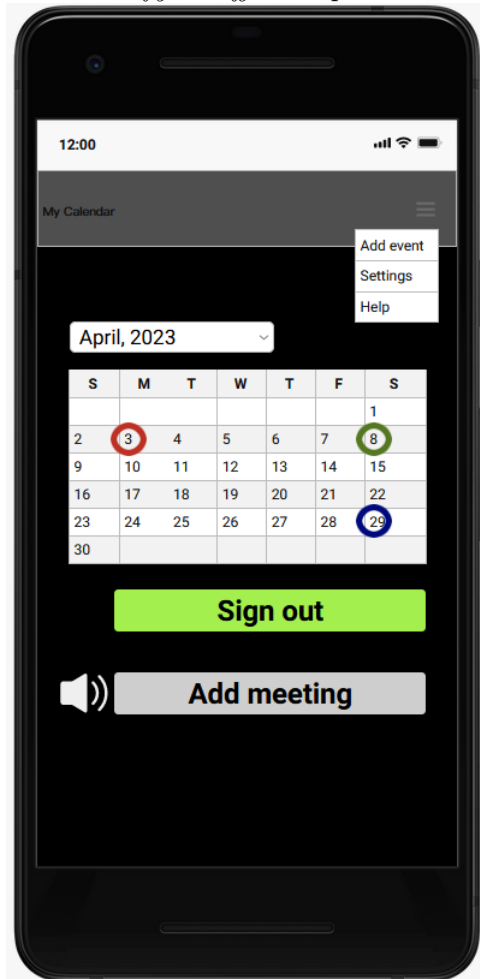
Solution

We will make a list of possible UI design issues for this activity:

- The font for the title text at the top of the screen "My Calendar" is too small, so it is hard for the user to see which activity we are in. Therefore we should increase the size of the title font.

- The "add event" option in the spinner context menu is an important action for this activity, so it should not be in the context menu in the menu bar. We should put the action as a button under the calendar to fix this.

- There is no indication as to what the colours of the circle on the calendar mean. We can combat this issue by adding a legend telling us what each colour means.

- The action bar's colour is way too dark and it is hard to tell that it is the action bar; it isn't distinguishable. We can change this by making it a brighter colour.

- The "sign out" button is in a prominent position on the screen (under the calendar) when it isn't at all relevant to what the activity is supposed to do (likely adding events to a calendar). Not only that, but it is a different colour from the rest of the buttons and UI elements on the screen currently. Therefore we should get rid of it from the bottom of the screen and put it in the action bar's context menu.

- There is a volume icon beside "add meeting" which makes no sense. Therefore we should change the icon to something better like a "+" symbol.

Problem 2

You are designing an assembly service for IKEA. The service allows users to input the pieces of furniture they want assembled and get a quote. The basic cost for any assembly is $15 per item. Other factors include:

a. Type of furniture: Beds, cabinets, drawers (+$10 per item); Desks, dining sets (+$5 per item); Bookcases, chairs (basic price)

b. Time of assembly: Morning (basic cost), Afternoon (+$20 for entire assembly)

c. Additional hourly rate: 0-3 hours (basic cost), any additional hour up to 6 hours (+$10 per additional hour beyond 3 and until 6)

Provide the equivalence classes and the test cases for this use case. (10 points) NOTE: *You do not have to accurately calculate the result of the price for every test case.*

Solution

We have 3 variables.
    There are

- 3 valid classes for type of furniture

- 2 valid classes for time of assembly

- 2 valid classes for hourly rate

There is at least one invalid class per variable (any other type of furniture, any other time of assembly and additional hours out of bounds).
There are also four marginal values for hourly rate

- 0

- 3

- 3.1

- 6

There must be at least 3 test cases for valid values, 4 test cases for marginal values and 3 test cases for invalid values. *Deduct -1 for every test case missing (max -3 points). Do not deduct two points if it is missing both the value and the test case.*

Section 4

Prototyping problem

You are designing "PetGPS", a mobile app to complement a wearable digital collar for pets. The collar emits a geolocation signal captured by the app. In the app, the user can register up to 5 collars to track different pets. The user can track the pets on a map. The user can interact with the map (zoom in and out, tap on the pet's location to see more information about the specific pet). If a pet is further than 10 meters away from where the phone with the app is, a proximity alarm goes off in the app. The app can also be connected with an autofeeder and let you know when it's time to order more food. This can be done from the pet's profile that contains information about the pet, including a picture of the pet, and gets data from the autofeeder. The user can place an order directly from the app, specifying the quantity and the type of the pet food and paying by credit card (which is already saved in the phone). Finally, the user should create a profile associated with the app and get a personalized page where they can see all their registered pets. (30 points total)

Read the description carefully and answer the following questions as best and as complete as you can:

a. What would you include in the manifest of this app? Mention activities, permissions, intent-filters and anything else you deem relevant. (10 points)

Solution

The following is a list of things included in the manifest for the app:

A. **Activities**
   1. UserProfileActivity
   2. PetProfileActivity

    3. `MapActivity`
    4. `OrderFoodActivity`
    5. `CreateProfileActivity`

B. **Intent filters for activities**

    1. `UserProfileActivity` → `PetProfileActivity`
    2. `PetProfileActivity` → `OrderFoodActivity`
    3. `CreateProfileActivity` → `UserProfileActivity`
    4. `PetProfileActivity` → `MapActivity`

C. **Permissions**

    1. Internet for ordering food online
    2. Google services for Google maps
    3. Local storage access for pet photos
    4. GPS for location services

b. What kind of listeners or other helper classes do your activities need to implement to allow for the intended interactions between the user and the app? (5 points)

Solution

We can make a list of classes that we would use to implement the application's features:

1. `OnTouchListener` for touch interaction with on screen lists
2. `ScaleGestureDetector` for zooming in and out of the map
3. `OnClickListener` for checking regular button clicks
4. `Adapter` for handling list items

c. Provide two UI mockups, one for the user's profile and another for the pet's profile. What kind of UI elements would you choose? What kind of Layouts would you use for the two activities? (10 points)

Solution

Let us list out all the UI elements we will use for both activities

    A. **User profile**
- List of pets
- Button for adding a new pet
- Menu for editing profile
- Sign out button, settings button, etc.

    B. **Pet profile**
- Activity with 2 fragments: one with photo(s) of pet and another with the pet's information under the photo. We also need buttons to open the map to locate the pet (it opens `MapActivity`) and to order food (it opens `OrderFoodActivity`).

d. Write test case to test the use case where the user places an order for pet food after they have received a signal from the autofeeder. (5 points)

Solution

We do this by listing setup, data, procedure, and expected result.

    A. **Setup:**
1. Food alarm has gone off
2. User opens `OrderFoodActivity` for one of their pets

    B. **Data:**
- Pet's name is Lucy
- Past order is "2023-02-15, 7kg dry food"
- Order was fulfilled by PetSmart (URL)

    C. **Procedure:**
1. User clicks on renew order
2. User clicks on confirm new order
3. User clicks pay button to pay for order
4. User confirms payment for stored payment method (credit card)

    D. **Expected result:**
1. Confirmation toast shows up to alert the user that payment was confirmed
2. Data about last order is updated in pet's profile
3. Order status is set to "pending"

Section 5

Constructing equivalence class tables

To construct this table, we need the following columns:

- Variable

- Valid class

- Valid value

- Boundary value

- Invalid class

- Invalid value

The minimum number of test cases is calculated with the following formula:

$$N_{cases} = max\{N_{vals,valid}\} + max\{N_{vals,boundary}\} + N_{vals,invalid}$$

Variable	Valid class	Valid value	Boundary value	Invalid class	Invalid value
Day	(1) mon, tue, wed, thu, fri	mon		(1) Any other value (not a day)	mox
	(2) sat, sun	sat			
Member	(1) M	M		(1) Anything other than M or NM	88
	(2) NM	NM			
Entry hour	(1) 6h-19h	7h55	6h, 19h	(1) <6h	4h40
	(2) 19h01-24h	20h44	19h01, 24h	(2) Any other string value (not an hour)	&@
Age	(1) 0.0-16.00	8.4	0.0, 16.00	(1) Any other string value (not an age)	TTR
	(2) 16.01-60.00	42.7	16.01, 60.00	(2) Age out of bounds (age < 0 or age > 120)	150.1
	(3) 60.01-120.00	65.0	60.01, 120.00		

In this chart:

- The valid class with most valid values would be age. So $Max\{N_{vals,valid}\} = 3$ since we take 8.4, 42.7, and 65

- We can use age's boundary values as well since we want the one with the most values. Therefore $Max\{N_{vals,boundary}\} = 6$

- Finally, $N_{vals,invalid} = 6$

All in all, 3+6+9=15, so we have that the minimum number of test cases for this set of specifications would be 15.

No.	Day	Membre	Entry hour	Age	Result
1	Lun	NM	7h55	8.4	$5.00
2	Sam	M	20h44	42.7	$8.00
3	Sam	M	20h44	65.0	$5.50
4	Sam	M	6h00	0.0	$3.50
5	Sam	M	19h00	16.00	$3.50
6	Sam	M	19h01	16.01	$8.00
7	Sam	M	19h01	60.00	$8.00
8	Sam	M	24h00	60.01	$5.50
9	Sam	M	24h00	120.0	$5.50
10	Mox	NM	7h55	8.4	Invalid day
11	Lun	88	7h55	8.4	Invalid member
12	Lun	NM	4h40	8.4	Hour out of bounds
13	Lun	NM	&@	8.4	Invalid hour
14	Lun	NM	7h55	TTR	Invalid age
15	Lun	NM	7h55	150.1	Age out of bounds

Here is the chart detailing each of the test case.

Section 6

Extra quiz questions