

## ***Campus Navigation System for Anna University.***

### **Software:**

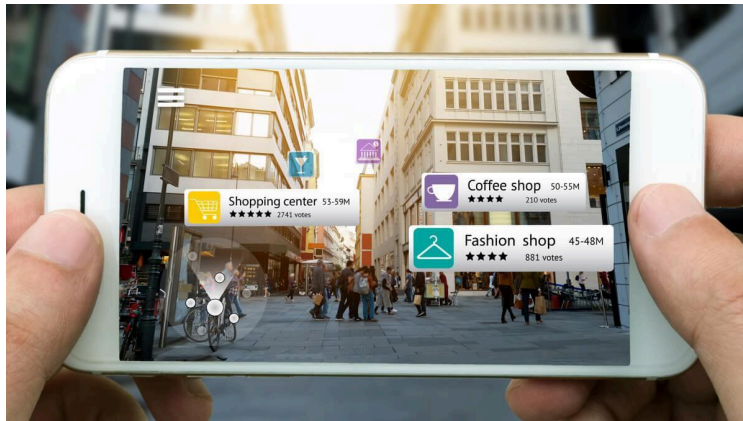
#### 1. What is the difference between AR and VR?

The difference lies on the level of immersion in a simulated environment(mimics real world activities).

**Augment:** make something greater by adding to it.

Augmented Reality adds digital elements to the user's view. User can interact in both real and virtual world. AR is mostly accessed in smartphones. In simpler terms we can say that an interactive experience that enhances the real world with computer generated information.

Eg: Look at the street view before we visit the place via Google maps.



### **Virtual Reality:**

Virtual Reality completely immerses the user in a simulated environment. Users are restricted from interacting with virtual environments. VR requires computer hardware and headsets.



We could Meta the parent company of Facebook.Inc as a virtual reality company.

#### 2. What are the pillars of Object-Oriented Programming, and how would you apply them in AR app development?

The pillars of OOPS are **Abstraction,Encapsulation,Inheritance and Polymorphism.**

**Encapsulation:** Bundling of attributes and functions into a single unit or class.

In terms of Encapsulation, components such as 3D models,camera controls and UI(User's Interface) Elements are encapsulated within a class.

Let's take a virtual object which consists of its size, position, texture and many more are defined within the class and are being encapsulated. In addition to encapsulation also helps in hiding the internal representation of the object.

**Inheritance:**One class inherits the properties of another class.

Let's take an AR app which has different objects like buildings,vehicles and many more. These subclasses can inherit from the Base class. These subclasses inherit the general behaviour like scaling,movements etc. It is told that it allows easy updates across similar objects.

## ***Campus Navigation System for Anna University.***

**Polymorphism:** Object of different class to be treated as an object of a superclass. It also enables method overriding which means it allows to replace a method in a subclass which is already defined in superclass. The implementation might vary. In AR app development there is a method called `interact()`, which is implemented in one way for building and another way to a character.

**Abstraction:** It hides the complex details and shows only the essential ones. In AR there might be many low level details like camera calibration, sensor data etc. which are abstracted away and only the essential ones are being considered like tracking objects and interacting with AR environments. Abstract class and interface are used in trackable objects which helps to track in the AR world.

3. What is the difference between structured and unstructured data, Under what category 3D model data comes in?

### **Structured data:**

It is defined and searchable. It is said to be predefined data models which is easy to store and analyse. Examples of structure data are phone numbers, dates, serial number etc...

### **Unstructured data:**

It is difficult to categorise and search. Unstructured data are non numeric and text-heavy. Examples of unstructured data are photos, social media post and so on.

3D model data is considered to be unstructured data because it does not fit into traditional databases.

4. How would you reduce the APK size of an AR app?

### **APK-Android Package Kit file.**

It has all the information needed to install and run an app on an Android device.

Reduce the APK size of an AR app using:

- Only necessary libraries and dependencies
- Optimise app resources
- Use vector drawable
- Use proGuard to reduce app size.
- Use APK splitting and Android App bundles.
- Optimise code and layout.
- Minimise resource duplication
- Use android size analyzer plugin

By reducing the APK file it helps to improve the user experience and storage space on devices.

5. What is database sharding, and when would you consider using it in an AR application?

It is a technique that can be useful in AR applications. When a database needs to handle large amounts of data or clients.

Sharding works by splitting up the large data into smaller ones called shards. These shards are responsible for their respective range and type of data.

6. Find the error (C++):

```
#include <iostream>
using namespace std;
class ARObj {
public:
    ARObj(int size) {
        this->size = size;
    }
}
```

```
void display() {
cout << "AR Object Size: " << size << endl;}
private:
int size;
};
int main() {
ARObject obj;
obj.display();
return 0;
}
```

Corrected one:

```
#include <iostream>
using namespace std;
class ARObject {
public:
ARObject(int size) {
this->size = size;
}
void display() {
cout << "AR Object Size: " << size << endl;}
private:
int size;
};
int main() {
ARObject obj(10);
obj.display();
return 0;
}
```

When you create an object without any argument it calls the default constructor. However in this program the ARObject has a parameter int size so in the main function we gotta declare an argument.

#### 7. How does garbage collection work in Java or C#?

In C programming to free up the memory we gotta use free() but in Java and C# it automatically manages the memory by removing the unwanted objects. By doing this it frees the memory up.

8. Explain why uncommenting obj.~ARObject(); would cause a segmentation fault, and provide a solution to avoid it.

```
#include <iostream>
using namespace std;
class ARObject {
public:
ARObject() {
size = new int;
*size = 100;
}
~ARObject() {
delete size;
}
void displaySize() {
cout << "AR Object Size: " << *size << endl;
}
private:
int* size;
```

## ***Campus Navigation System for Anna University.***

```
};  
int main() {  
    ARObjekt obj;  
    obj.displaySize();  
    // Uncommenting this will cause a segmentation fault, why?  
    // obj.~ARObjekt();  
    obj.displaySize();  
    return 0;  
}
```

The destructor will cause a double deletion of the dynamic allocated size.

In the main function the destructor is called automatically. Once the destructor is called the memory pointed to the size gets deleted. Once the object goes out of the main function the destructor will be called again and causes double deletion. To avoid this we should not call destructor explicitly.