

Nama: Ahmad Fatihin  
Role : Software Engineer Mobile

1. - Inisialisasi dan deklarasi variabel count = 0  
- Ubah string menjadi Array berisi kata-kata dengan memisahkan setiap kata dengan spasi  
- Iterasi setiap kata dalam array  
- Jika kata sama dengan "SIAPA", tambahkan 1 pada count  
- Kembalikan count sebagai hasil akhir

Contoh dengan kotlin:

```
fun countSiapa(sentence: String): Int {  
    var count = 0  
    val words = sentence.split(" ")  
    for (word in words) {  
        if (word == "SIAPA") {  
            count++  
        }  
    }  
    return count  
}
```

2. a. SELECT Nama  
FROM TabelKedua  
WHERE NIM IN (  
 SELECT NIM  
 FROM TabelPertama  
 WHERE Tanggal = '15 Juni 2020' OR Tanggal = '17 Juni 2020'  
)  
  
b. SELECT Nama, COUNT(NIM) AS JumlahKehadiran  
FROM TabelKedua  
WHERE NIM IN (  
 SELECT NIM  
 FROM TabelPertama  
 WHERE Tanggal = '15 Juni 2020' OR Tanggal = '17 Juni 2020'  
)  
GROUP BY Nama

3. a. Variabel hasil tidak diinisialisasi, Yang diinisialisai hanya variabel 'hasil'. Penulisan variabel total seharusnya hasil.  
  
b. public int perkalianSederhana (int j, int k)  
{ int hasil = 0;

```

        while(j > 0)
        { hasil += k;
          j--;
        }
        return hasil;
    }

```

4. algoritma sederhana untuk menghitung potensi pendapatan maksimum dari toko 'kawanLamaJewel':
  - Inisialisasikan variabel dan jumlah anting Ruby, Topaz, dan Permata
  - Tentukan harga jual masing-masing jenis anting
  - Hitung jumlah pendapatan dari penjualan anting Ruby
  - Hitung jumlah pendapatan dari penjualan anting Topaz
  - Hitung jumlah pendapatan dari penjualan anting Permata
  - Tambahkan jumlah pendapatan dari masing-masing jenis anting
  - Cetak jumlah pendapatan maksimum yang didapat dari toko KawanLamaJewel.

Contoh Kode:

```

fun main() {
    val hargaRuby = 1000000
    val hargaTopaz = 1250000
    val hargaPermata = 3000000

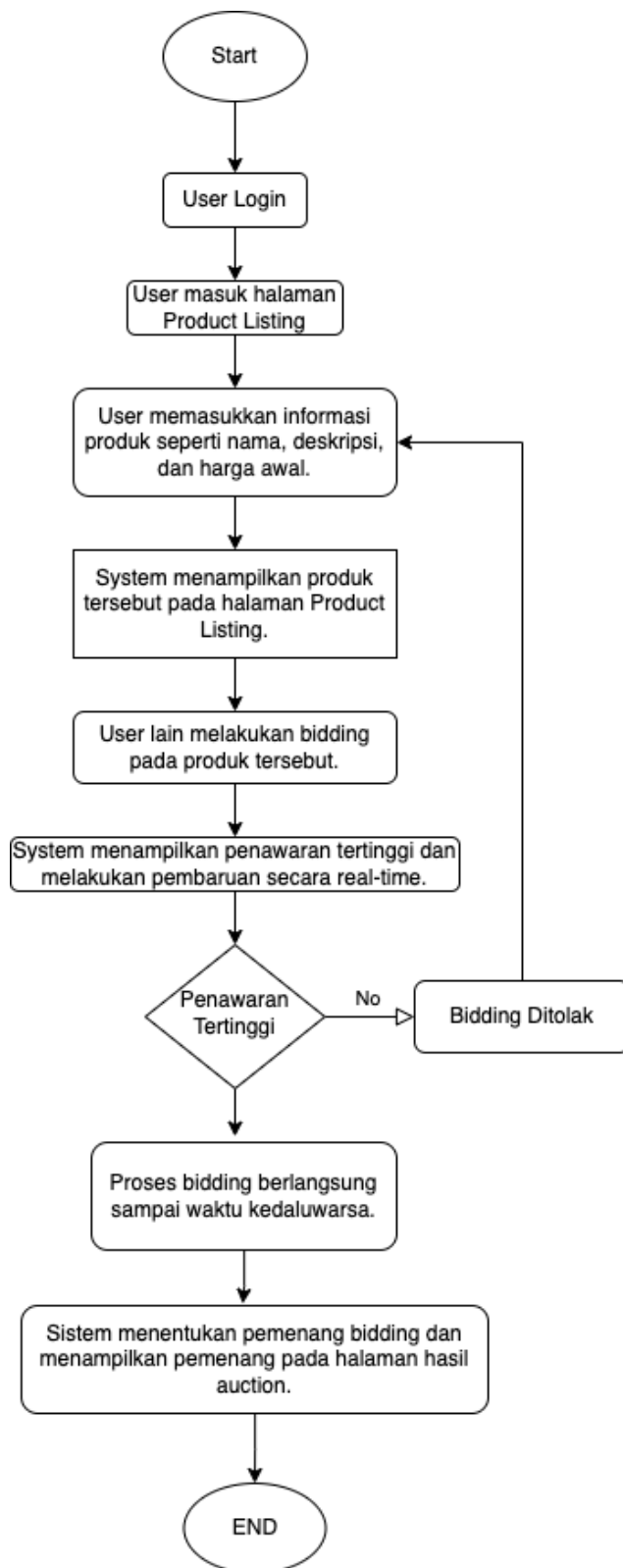
    val jumlahRuby = 5
    val jumlahTopaz = 3
    val jumlahPermata = 1

    val pendapatanRuby = hargaRuby * jumlahRuby
    val pendapatanTopaz = hargaTopaz * jumlahTopaz
    val pendapatanPermata = hargaPermata * jumlahPermata

    val pendapatanTotal = pendapatanRuby + pendapatanTopaz +
    pendapatanPermata

    println("Potensi pendapatan maksimum adalah sebesar Rp $pendapatanTotal")
}

```



5.

Teknologi yang digunakan:

- HTML, CSS, dan JavaScript untuk membangun tampilan website. Untuk memudahkan dapat menggunakan Library CSS, seperti Bootstrap atau Tailwind.

- Flutter untuk membangun Apps Android dan IOS. Dengan State Management Riverpod, BLOC, atau GetX
- PHP atau framework seperti Laravel, CodeIgniter atau Ruby on Rails untuk membangun CMS aplikasi.
- Database seperti MySQL atau MongoDB(NoSQL) untuk menyimpan data produk dan bidding.
- Web server seperti Apache atau Nginx untuk menjalankan aplikasi.
- Service Tambahan seperti Firebase, Firebase authentication untuk mudah login dengan sosmed, Firebase Push Notif untuk menampilkan local notification di Apps untuk menampilkan status bidding yang akan menggunakan Deep link untuk masuk ke page tertentu (Page Bidding, Produk atau Notifikasi) Firebase Analytics untuk mendapatkan Analytic tentang apps, juga Crashlytics untuk informasi User yang Crash. Firebase Remote Config untuk melakukan Perubahan remote terhadap aplikasi seperti merubah icon saat Natal, Ramadhan, Tahun Baru dll tanpa melakukan update/install baru pada aplikasi.
- Google Cloud atau AWS
- Github atau Gitlab untuk kolaborasi Developer
- PM Tracking dengan menggunakan JIRA atau Trello
- Fastlane, Github Action , Xcode Cloud untuk Continue Deployment.

6. Flow chart untuk system antrian SPBU dapat dibuat sebagai berikut:

- Start: User memasukkan jenis bahan bakar yang dibutuhkan (pertalite, pertamax, atau pertamax plus).
- Pemeriksaan Terminal: Sistem memeriksa terminal yang tersedia untuk jenis bahan bakar yang dipilih oleh user.
- Penentuan Antrian: Sistem menentukan antrian sesuai dengan jumlah terminal tersedia untuk jenis bahan bakar tersebut.
- Pemberian Nomor Antrian: Sistem memberikan nomor antrian kepada user dan menampilkan nomor antrian pada layar.
- Menunggu Giliran: User harus menunggu giliran sampai nomor antrian yang diterima sesuai dengan nomor yang sedang dipanggil.
- Pelayanan: Setelah giliran tiba, sistem memanggil user untuk melakukan pembelian bahan bakar.
- Transaksi: User melakukan transaksi pembelian bahan bakar.
- End: Proses antrian selesai setelah user melakukan transaksi pembelian bahan bakar.