Nombres Luis Adolfo Limachi Armijo

**1.** Describe los pasos que seguirías para desplegar una aplicación web con frontend y backend en un servidor de producción. ¿Qué consideraciones de seguridad tendrías en cuenta durante este proceso? (deberá responderse en un documento Word y adjuntarse a los archivos a compartir en el repositorio de GitHub explicado más adelante).

R.- Para desplegar el proyecto backend y front se realiza los siguientes paso

**Backend.-**

1.- Instalación del sistema operativo Linux bajo la distribución debían

2.- Configuración de la dinámica a estática

3.- Instalación de apache o ngnix

4.-Instalacion de php 8.2

5.- Instalación de componer

6.- Copiar la carpeta del proyecto a la dirección mediante fpt a la dirección /var/www

7.- Configurar la conexión al servidor de base de datos postgres

8.- configurar y re direccionar a la carpeta public del proyectó virtualhost

**Fronted**

1.- Instalación del sistema operativo Linux bajo la distribución debían

2.- Configuración de la dinámica a estática

3.- Instalación de apache o ngnix

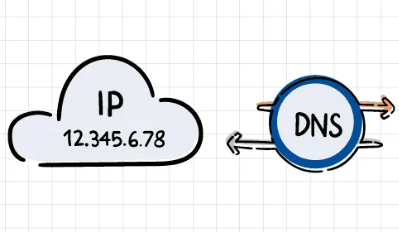
4.- Apuntar a los enpoint al backend con la direccion ip

5.- Configurar virtual host a la carpeta public para que salga el proyecto Front

6.- Poner un dominio utilizando un servidor de dns utilizando bind9 bajo distribución libre

7.- configuración el proxy inverso para que salga con el puerto 443 y certificado mediante https

dmz



**2.** Escriba una función en Kotlin que simule el acceso a la cámara del celular. La función podría tomar como entrada una “ruta de archivo” y devolver un “archivo de imagen”. Utilizando el entorno en línea del siguiente enlace: <https://play.kotlinlang.org/> (la respuesta deberá adjuntarse como código al mismo documento Word del 1er ejercicio)

import android.app.Activity

import android.content.Intent

import android.net.Uri

import android.os.Bundle

import android.os.Environment

import android.provider.MediaStore

import androidx.appcompat.app.AppCompatActivity

import androidx.core.content.FileProvider

import kotlinx.android.synthetic.main.activity\_main.\*

import java.io.File

import java.io.IOException

import java.text.SimpleDateFormat

import java.util.\*

class MainActivity : AppCompatActivity() {

private lateinit var currentPhotoPath: String

companion object {

const val REQUEST\_IMAGE\_CAPTURE = 1

}

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

captureButton.setOnClickListener {

dispatchTakePictureIntent()

}

}

private fun dispatchTakePictureIntent() {

Intent(MediaStore.ACTION\_IMAGE\_CAPTURE).also { takePictureIntent ->

takePictureIntent.resolveActivity(packageManager)?.also {

val photoFile: File? = try {

createImageFile()

} catch (ex: IOException) {

// Manejar la excepción según tus necesidades

null

}

photoFile?.also {

val photoURI: Uri = FileProvider.getUriForFile(

this,

"com.example.android.fileprovider",

it

)

takePictureIntent.putExtra(MediaStore.EXTRA\_OUTPUT, photoURI)

startActivityForResult(takePictureIntent, REQUEST\_IMAGE\_CAPTURE)

}

}

}

}

@Throws(IOException::class)

private fun createImageFile(): File {

val timeStamp: String = SimpleDateFormat("yyyyMMdd\_HHmmss").format(Date())

val storageDir: File = getExternalFilesDir(Environment.DIRECTORY\_PICTURES)!!

return File.createTempFile(

"JPEG\_${timeStamp}\_",

".jpg",

storageDir

).apply {

currentPhotoPath = absolutePath

}

}

override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {

super.onActivityResult(requestCode, resultCode, data)

if (requestCode == REQUEST\_IMAGE\_CAPTURE && resultCode == Activity.RESULT\_OK) {

}

}

}

override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {

super.onActivityResult(requestCode, resultCode, data)

if (requestCode == REQUEST\_IMAGE\_CAPTURE && resultCode == Activity.RESULT\_OK) {

val imageBitmap = data?.extras?.get("data") as Bitmap

imageView.setImageBitmap(imageBitmap)

}

}

}

**3.** Refactorice el siguiente código que muestra los números pares de una lista de números, para mejorar el rendimiento en su ejecución. Utilizando el entorno en línea del siguiente enlace: <https://play.kotlinlang.org/> (la respuesta deberá adjuntarse como código al mismo documento Word del 2do ejercicio)

fun main() {

val numbers = listOf(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)

val evenNumbers = numbers.filter { it % 2 == 0 }

println(evenNumbers)

}