RECURSOS HUMANOS agregar()
buscarUno()
eliminar() apellidos
cedula
cargo
departamento
cuentaBancaria
banco
asistencias:
ListaRegistrosAsistencias
sueldo: Sueldo
estado
sexo

obtenerCargo()
obtenerBanco()
obtenerCuenta() mostrar() Sueldo Dashboard Empleado calcularValor() nombres
apellidos
cedula
cargo
departamento
cuentaBancaria
banco
asistencias:
ListaRegistrosAsistencias
sueldo: Sueldo
estado
sexo calcularLiquidacion() =
ultimoSueldo/4 * numAños
numAños= fechaFin-fechalnicio nombres
apellidos
cedula
cargo
departamento
cuentaBancaria
banco
asistencias:
ListaRegistrosAsistencias
sueldo: Sueldo
estado
sexo valorApagar numAños: int ultimoSueldo: Sueldo calcularValoraPagar() registrarAsistencia(): boolean solicitarPermiso(): boolean horalnicio: Hora horaSalida:Hora fecha: Fecha añadirRegistroAsistencia() editarRegistroAsistencia() borrarRegistroAsistencia() mostrarRegistro(); obtenerCargo()
obtenerBanco()
obtenerCuenta()
obtenerCalificacion() ListaContratos

lista<Contratos>

anadirContrato()
eliminarContrato() setHoralnicio()
setHoraFin()
setFecha()
registrar() ControladorRegistros fechalnicio: Fecha;
fechaFin: Fecha;
tipo;
obtenerTipo()
obtenerFechaFin() anadirAspirante()
eliminarAspirante()
selectAll() mapa: MapaDisponibilidad validarRegistro() notificarRegistro() ControladorPruebaAdmision

puntajeBase
actitudesBase
aptitudesBase Calendar
calendario;
hora;
minutos;
segundos;

getMinutos()
getSegundos()
getHora()
setSegundos()
setMinutos()
setHora() Aspirante puntaje actitudes aptitudes nombre
apellidos
cedula
telefono
cargoAspirante
pruebaAdmision:
PruebaAdmision

getNombre()
getApellidos()
getCedula()
getTelefono()
getCargoAspirante()
getpruebaAdmision()
solicitarRegistroPrueba() Contratacion fechaCorrecta()
compararFechaActual()
esValidaFecha()
esBisiesto()
sumarDiasAFecha()
toString() validarPruebaAdmision()
verificarAptitudes()
verificarActitudes()
obtenerAspirantesAptos() numero fecha: Fecha estado obtenerPuntaje() obternerActitudes() obtenerAptitudes() registrarEmpleado()
validarRequisitos(
registrarContrato() La contratacion se hace validando que el aspirante cumpla con requerimientos de la Despido empresa, luego de esto se pasa a la cotratacion donde se resgitran datos y termina en un contrato liquidacion: Liquidacion fecha: Fecha RRHH ListaVehículos

vehículo: Lista<Vehiculo> vehiculos: ListaVehiculos conductores: ListaConductores contratacion: Contratacion pendientes: RolPagos fechaActual: fecha solicitarLiquidacion() solicitarPago() L---actualizarEstado()
asignar()

Texto Lista<Mantenimiento>

añadirMantenimiento() agregarVehiculo() eliminarVehiculo() contratarPersonal()
generarPagosEmpleados()
solicitarPago()
despedirEmpleado()
iterarEmpleados()
iteracionRoles()
obtenerRoles() inicializar() actualizarMapa() trazarRuta() fechaProximoMantenimiento: Fecha Autorizacion se relaciona con solicitud de pago para autorizar pago y para que si autoriza. A su vez Autorizacion se relaciona con pago puesto que envia el monto y la cuenta bancaria a la cual se debe realizar el pago. Mantenimiento

vehiculo: Vehiculo
gasto

realizarMantenimiento() GPS

posicion:Posicion

actualizarPosicion() solicitarAutorización()
solicitarPago()
setEstado()
actualizarSolicitud() tipo
kilometraje
fichaTecnica: FichaTecnica

actualizarEstado()
actualizarFicha()
notificarPosicion() PermisoNacimientoHijo
proporcional
numDias CalamidadDoméstica

proporcional
numDias SituaciónMédica

proporcional
numDias Todo trabajador tendrá derecho a gozar anualmente de un período ininterrumpido de 15 días de vacaciones, incluidos los días no laborables. Quienes hubieren prestado servicios más de cinco años al mismo empleador, gozarán adicionalmente de un día de vacaciones por cada uno de los años excedentes Posición

Iatitud:String
Iongitud:String
direccion:String
fcid: String

actualizarPosicion() Ruta

distancia
tiempoAproximado
origen:Posicion
destino:Posicion

definirRuta()
obtenerInformacin() calcularAfectaciónDeSueldo()
calcularNumeroDeDias() calcularAfectaciónDeSueldo() calcularNumeroDeDias() calcularAfectaciónDeSueldo() calcularNumeroDeDias() calcularAfectaciónDeSueldo()
calcularNumeroDeDias() fechaSolicitud: Fecha
enviarSolicitud() Habilitado En espera Inhabilitado ListaEntregas

entregas: Lista<Entrega>

añadirEntrega() enEntrega(); enMantenimiento(); enAccidente(); enPicoPlaca(); enEntrega(); enMantenimiento(); enAccidente(); enPicoPlaca(); enEntrega(); enMantenimiento(); enAccidente(); enPicoPlaca(); estado destino: Posicion

actualizarEstado() confirmarEntrega() verificarVehiculo() espera la autorización para realizar un cambio de estado. Presupuesto

libroDiario:LibroDiario

calcularPresupuesto() Pago

fechadePago:Fecha
nroCuenta:String
monto:double
presupuesto:Presupuesto

registrarPago(params)
realizarPago(params) Libro Diario

listaPagos<Pagos>
listaIngresos<Ingresos>
pagoTotal:double
ingresoTotal:double Ingreso

fechalngreso:Fecha
idFactura:String
total:double

registarIngreso(params) Una Compra de uno o varios productos de un proveedor, o un servicio de un proveedor. Tiene el inventario para poderlo actualizar. Y genera una lista de Bienes a comprar. Una Lista de compra tiene varios productos de varios proveedores o un servicio de un proveedor Una pestaña principal que nos muestra el analizador, otra la solicitud actual y otra pestaña para ver el registro de compra. calcularPagoTotal()
calcularIngresoTotal() inventario: Inventario
listaBienesAComprar:ListaCantidadDeBienes
autorizacion:SolicitudDePago
estado
montoTotal

comprar(listaBienes)
solicitarAutorizacion(proveedor, monto)
calculoMontoTotal()
registrarCompra() ListaProveedores

lista<Proveedor>

añadirProveedor(proveedor)
quitarProveedor(proveedor) Analizador

libroDiario:LibroDiario
fechaDesde:Fecha
fechaHasta:Fecha CompraDeProducto

comprar(listaBienes) calcularIngresoMaximo()
calcularIngresoMinimo()
calcularGastoMaximo()
calcularGastoMinimo()
calcularIngresoMensual()
calcularGastoMensual() ruc
razón social
dirección
n_cuenta

registrarProveedor()
actualizarProveedor() Fecha

dia;
mes;
año;

devolverMes()
devolverDia()
devolverAño() Analiza y crea una lista de compra si el Stock está en una cantidad definida. Dashboard

analizador:Analizador

mostrarIngresos()
mostrarGastos()
mostrarAnalisis() Le enviamos los productos que se ofrecen. Inventario listaBienes: ListaCantidadDeBienes cantidadDeBien:CantidadDeBien

aumentarStock(cantidad,bien) disminuirStock(cantidad, bien) añadirbien(cantidad,bien) quitarbien(cantidad,bien) Cliente

Nombre
Correo
FEcha de Nacimiento
Celular
Direccion
Cedula
Contraseña Producto

Nombre
Precio
proveedor:Proveedor

registrar()
actuaillzar()

Servicio

Nombre
Precio
proveedor:Proveedor

registrar()
actualizar() Este inventario tiene todos los productos que ofrece Trappi. Y los serviios que este modulo controla. El controlador Inventario se encarga de manejar la comunicación entre nuestro inventario y el módulo de E-commerce Lleva cuenta de cada producto <<interfaz>> Formulario -categoria Producto productos:Inventario obtenerStock()
borrarProductoStock()
añadirProductoStock()) registrarse()
editarMisDatos()
eliminarse() entrarAlSistema()
salirSistema()
añadirTarjeta()
editarTarjetaCredito()
editarTarjetaDebito() Obtiene los productos que se encuentran en inventario, y tienen una cantidad mayor a 0 para mostrar al cliente. EdiciónCliente RegistroCliente

Nombre
Correo
FEcha de Nacimiento
Celular
Direccion
Cedula
Contraseña

obtenerDatos() Nombre
Correo
FEcha de Nacimiento
Celular
Direccion
Cedula
Contraseña CarritodeCompra obtenerDatos() Lista<Articulo> factura:Factura -Lista>Articulo> productos:Inventario añadirProducto()
vaciarCarrito()
borrarProducto()
cantidadCompraProducto()
confirmarContenido() Obtiene los productos que se encuentran en inventario, y tienen una cantidad mayor a 0 para mostrar al cliente. n°Factura
Direccion
cedulaCliente
nombreCliente
pago:Pago
#No pagado a pagado, si se valida el
pago, se cambia de estado
Conexion: Conexion calcularTotal()
calcularSubtotal()
calcularImpuestos()
mostrar()
realizarConexion() n°Factura
Direccion
nombreCliente
valor total
detalleDEProductos Pago
valorAPagar realizarPago()
validarFondos()
mostrar() enviarAFinanzas() enviarAProveedores() enviarALogistica() //Si el pago no pasa, no se registrarà la factura validarPago()#Fondos necesarios Se encarga de enviar la información requerida por los modulos de finanzas(id, valor de factura), proveedores(producto comprados para reducción en inventario) y por ultimo a logistica(Productos comprados para que sepa cual es el pedido)

El encargado de realizar las ventas en nuestro proyecto.

EdiciónTarjeta

numero
CVV

obtenerDatos()

numero
CVV
caducidad

realizarPago()
validarFondos()
mostrar()

caducidad

RegistroTarjeta

cnumero
CVV
caducidad

obtenerDatos()