>>DIAGRAMA DE SECUENCIAS<< **CAJERO** > Realizar depósito : PaginaDepositos : MostradorResultados : GestorTransacciones : GuardiaSeguridad : Registrador : Cajero : Home : Buscador : Tramitador Cajero solicitarDatos Trabajador() solicitarPagina solicitarPagina Deposito() Deposito() estaENturno(turno) valorDeVerdad== true] devolverValorVerdad() mostrarPagina() solicitarDatosCuenta solicitarDatosCuenta (numero) (numero) solicitarDatosCuenta (numero) buscarDatos Cuenta(numero) enviarDatos(datos[]) mostrarDatos(datos[]) enviarDatos(datos[]) solicitarDeposito solicitarDeposito solicitarDeposito (monto, numeroCuenta) (monto, numeroCuenta) (monto, numeroCuenta) solicitarDeposito (monto, numeroCuenta) depositar (monto, numeroCuenta) tramiteExitoso solicitarRegistro Transaccion (monto, #cuenta, codigoCajero, montoAntiguo) registrarCredito (monto, #cuenta, registroExitoso() codigoCajero, [registroExitoso ==true] montoAntiguo) solicitaEstadoCuenta: solicitarEstadoCuenta(codigo) darEstadoCuentaActual() mostrarPagina(estadoCuenta) dar Estado Cuenta Actual ()> Identificar dueño de cuenta para el retiro de dinero : GuardiaSeguridad : Cajero : Formateador : PaginaRetiros : GestorTransacciones : Home : Buscador Cajero solicitarDatos solicitarPaginaRetiros() solicitarPaginaRetiros() Trabajador() estaENturno(turno) devolverValor Verdad() [valorDeVerdad==true] mostrarPaginaRetiros() solicitarBusqueda solicitarBusqueda buscarDueño (DPI) (DPI) (DPI) solicitarBusqueda(numeroDPI) [existenResultados] enviarResultado(datos[]) transformar(datos[]) enviarDatos(datos[]) enviarDatos(datos[]) mostrarDatosDueño() > Realizar retiro : MostradorResultados : GestorTransacciones : PaginaRetiros : Transaccion : Buscador : Registrador <u>: Cajero</u> solicitarReirto solicitarReirto solicitarReirto (numeroCuenta) (numeroCuenta, ) (numeroCuenta, ) solicitarRetiro (numeroCuenta, monto) (numeroCuenta, monto) tramiteExitoso : [tramiteExitoso == true] solicitarRegistro Transaccion (monto, #cuenta, codigoCajero, montoAntiguo) registrarCredito [procesoExitoso == true] (monto, #cuenta, tramiteExitoso solicitarEstadoCuenta(codigo) fueProceso codigoCajero, RealizadoConExito() montoAntiguo) darEstadoCuentaActual() mostrarEstado darEstadoCuentaActual() deCuenta() > Ver listado de transacciones realizadas en el turno en transcurso mostrando el total existente en caja hasta el momento : GestorReportes : Conversor : Home : PaginaReportes : <u>Cajero</u> : Buscador Cajero solicitarPaginaReportes() solicitarPaginaReportes() mostrarPagina() solicitarVisaulizacionReporte() solicitarVisaulizacionReporte() solicitarCodigo() darCodigo() buscarTransaciones solicitarTransacciones Realizadas Realizadas(codigoTrabajador) (codigoTrabajador) enviarResultado() convertir(resultado) mostrarTransacciones() darReporteTransacciones() darTransacciones() <> > Ver listado de transacciones y balance final por transacciones realizadas en días pasados en un intervalo de tiempo definido : PaginaReportes : GestorReportes : Conversor : Buscador : Home : <u>Cajero</u> solicitarPaginaReportes() solicitarPaginaReportes() mostrarPagina() especificarIntervaloTiempo() solicitarVisaulizacionReporte solicitarVisaulizacionReporte() (fechalnicio, fechaFin) solicitarCodigo() darCodigo() buscarTransaciones Realizadas (codigo Trabajador,solicitarTransaccionesRealizadas fechalnicio fechaFin) (codigoTrabajador, fechalnicio, darResultado() darTransacciones()<> convertir(resultado) hallarBalanceFinal solicitarBalance(transacciones >) (transacciones⇔) mostrarReporteFusion() dar Reporte TransaccionesdarTotalPorTransacciones() Acumuladas() NOTA: Tienes que ver como mezclar dos tipos de formatos en Jasper... X