>>DIAGRAMA DE COLABORACIÓN<<

CLIENTE > Transferir dinero 5: solicitar CuentasTerceross 13 y 22: cuenta: 6: solicitar darCuentas():= CuentasTerceros cuentas (codigoCliente) 14: solicitar 7. buscarCuentas CuentasTerceross TercerosDe(codigoCliente) 12 y 21: cuentas: : Cliente : Buscador darCuentas():= 33: solicitarTransferencia \$ y 17: enviarResultado cuentas Cuentas() (#CuentaOrigen, 4: solicitar #CuentaReceptora, monto) 16. buscarCuentas CuentasTerceros() TercerosDe(codigoCliente) 34: transferirDinero 11 y 20: cuentas: 36: exito: 23: cuentas: (#cuentaOrigen, 26: IlenarListado(tipo) darCuentas():= fueProcesoExitoso():= darCuentas():= #cuentaReceptora, [true/false] cuentas cuentas[] monto) solicitarPagina 1: solicitarPagina Transferencia() Transferencia() 15: solicitar : Transformador : Home : PaginaTransferencia : GestorTransferencia 37: exito: CuentasTerceros :Tramitador fueProcesoExitoso():= (codigoCliente) [true/false] 9 y 18: enviarResultado 29: IlenarListado 32: solicitarTransferencia Emisoras() 35: transferirDinero (#CuentaOrigen, (#cuentaOrigen, #CuentaReceptora, monto) #cuentaReceptora, 10 y 19: cuentas: monto) darCuentas():= ← cuentas 24: mostrarPagina() : Cuenta 25: seleccionarTipoCuenta Receptora(tipo) Cliente 27 y 30: agregarListadoCuentas() 28: seleccionarTipoCuenta Emisora() 31: solicitarTransferencia (#CuentaOrigen, #CuentaReceptora, monto) 38: mostrarResultados() > Solicitar asociación 4: solicitarBusqueda Cuenta(numero) 6: buscarDatos 8: formatear Dueño(numero) (datos[]) 11: datos: darDatos():= 7: datos: listaDatosDueño solicitarPagina darDatos():= 1: solicitarPagina 5: solicitarBusqueda datosDueño Cuenta(numero) Asociacion() Asociacion() : Buscador : Formateador : Home : PaginaAsociacion : GestorAsociacion 9: datos: 10: datos: darDatos():= darDatos():= listaDatosDueño 18: resultado: listaDatosDueño fueProcesoExitoso():= [true/false] 15: solicitarEnviar 14: solicitarEnviar 17: resultado: Solicitud() fueProcesoExitoso():= 2: mostrarPagina() [true/false] 3: solicitarBusqueda : Registrador Cuenta(numero) 12: mostrarDatos() Cliente 13: solicitarEnviarSolicitud() 16: registrarSolicitud 19: mostarResultadoProceso() Asociacion() > Aceptar solicitud de asociación 2: solicitarCambio Estado(codigoSOlicitantes, 1: aceptarSolicitud() aceptarSolicitud() 3: cambiarEstado codigoSolicitado) A(codigoSolicitdo, : GestorAsociacion : Actualizador : PaginaAsociacion 5: exito: codigoSolicitante, tipoEstado) 6: actualizarPagina() fueProcesoExitoso():= fueProcesoExitoso():= [true/false] [true/false] > Rechazar solicitud de asociación 2: solicitarCambio Estado(codigoSOlicitantes, 1: rechazarSolicitud() rechazarSolicitud() 3: cambiarEstado codigoSolicitado) A(codigoSolicitdo, : PaginaAsociacion : GestorAsociacion : Actualizador 5: exito: 4: exito: 6: actualizarPagina() codigoSolicitante, tipoEstado) fueProcesoExitoso():= fueProcesoExitoso():= [true/false] [true/false] Cliente > Ver últimas 15 transacciones más grandes realizadas por cuenta en el último año : Cliente 6: darCodigo Cliente 5: solicitarCodigo Cliente 4: solicitarReporte 7: solicitarReporteUltimas 1: solicitarPagina solicitarPaginaReportes() Reportes() Ultimas15Grandes() 15Grandes(año) 8: buscar : Home : PaginaAsociacion : GestorReportes : Buscador 12. transaccion: Transacciones(año) 13: reporte: darDatos():= enviarReportes:= transacciones 9. resultado: reportes darDatos():= 2: mostrarPagina() 11. transaccion: resultados darDatos():= transacciones 3: solicitarReporteUltimas 15Grandes() : Conversor Cliente 14: mostrarReporte() 10: convertir(datos[]) > Ver movimientos de cuenta realizados en un intervalo de tiempo : Cliente 7: darCodigo Cliente 6: solicitarCodigo Cliente 1: solicitarPagina solicitarPaginaReportes() 5: solicitarMovimientos 8: solicitarMovimientos Reportes() Cuenta(tiempo) Cuenta(tiempo) 9: buscar : GestorReportes : Buscador : Home : PaginaAsociacion 13. transaccion: Transacciones(tiempo) 14: reporte: darDatos():= enviarReportes:= transacciones 10. resultado: reportes 2: mostrarPagina() darDatos():= 12. transaccion resultados darDatos():= 3: especificarIntervaloTiempo() transacciones : Conversor 4: solicitarMovimientosCuenta Cliente (tiempo) 15: mostrarReporte() 11: convertir(datos[]) > Ver transacciones de la cuenta con más dinero a partir de una fecha inicial : Cliente 7: darCodigo Cliente 6: solicitarCodigo Cliente 1: solicitarPagina 5: solicitarTansacciones 8: solicitarTansacciones solicitarPaginaReportes() Reportes() CuentaMayor(fechalnicio) CuentaMayor(fechalnicio) 9: buscar : PaginaAsociacion : Home : GestorReportes : Buscador 13. transaccion: Transacciones(fechalnicio) 14: reporte: darDatos():= enviarReportes:= transacciones 10: resultado: reportes 2: mostrarPagina() darDatos():= 12. transaccion resultados darDatos():= 3: especificarIntervaloTiempo() transacciones conversor Cliente 4: solicitarTansacciones CuentaMayor(fechalnicio) 15: mostrarReporte() 11: convertir(datos[]) > Ver historial de solicitudes recibidas junto con su estado 6: solicitarCodigoUsuario() : Cliente 7: darCodigoUsuario() \longrightarrow 1: solicitarPagina 4 .peditSolicitudes 5: solicitarSolicitudes solicitarPaginaReportes() Reportes() Recibidas() Recibidas() 8: buscarSolicitudes : Home : PaginaAsociacion : GestorReportes : Buscador 12. solicitud: Recibidas() darDatos():= solicitudes resultado: 13. solicitud: enviarResultado darDatos():= 2: mostrarPagina() Solicitudes():= solicitudes 11. solicitud: resultados enviarSolicitudes Enviadas():= solicitudes 3: pedirSolicitudesRecibidas : MostradorResultados : Formateador 15: mostrarListadoSolicitudes() 10: formatear(datos[]) 14: generarVista Solicitudes(solicitudes) > Ver historial de solicitudes de asociación enviadas 6: solicitarCodigoUsuario() : Cliente 7: darCodigoUsuario() 1: solicitarPagina 4 .peditSolicitudes 5: solicitarSolicitudes solicitarPaginaReportes() Reportes() Enviadas() Enviadas() 8: buscarSolicitudes : Home : PaginaAsociacion : GestorReportes : Buscador Enviadas() 12: solicitud: 13: reporte: enviarSolicitudes enviarReportes:= resultado: Enviadas():= reportes enviarResultado solicitudes 2: mostrarPagina() 11. solicitud: Solicitudes():= enviarSolicitudes resultados Enviadas():= solicitudes 3: peditSolicitudesEnviadas() Cliente 4: mostrarReporte() : Conversor

10: formatear(datos[])