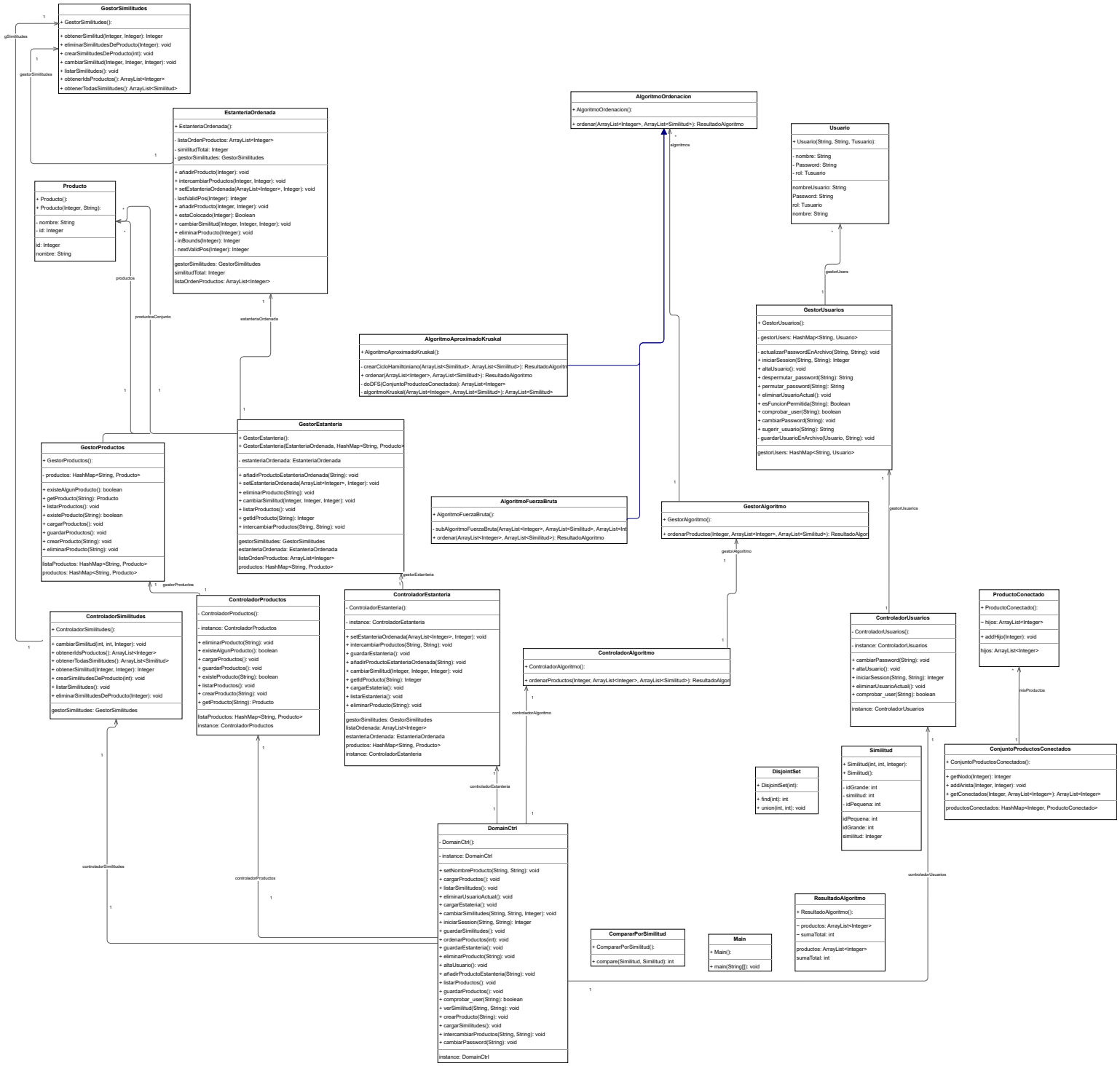
GestorSimilitudes
+ GestorSimilitudes():
+ obtenerSimilitud(Integer, Integer): Integer
+ eliminarSimilitudesDeProducto(Integer): void
+ crearSimilitudesDeProducto(int): void
+ cambiarSimilitud(Integer, Integer, Integer): void
+ listarSimilitudes(): void
+ obtenerIdsProductos(): ArrayList<Integer>
+ obtenerTodasSimilitudes(): ArrayList<Similitud>
gSimilitudes
gestorSimilitudes
EstanteriaOrdenada
+ EstanteriaOrdenada():
- listaOrdenProductos: ArrayList<Integer>
- similitudTotal: Integer
- gestorSimilitudes: GestorSimilitudes
+ añadirProducto(Integer): void
+ intercambiarProductos(Integer, Integer): void
+ setEstanteriaOrdenada(ArrayList<Integer>, Integer): void
- lastValidPos(Integer): Integer
+ añadirProducto(Integer, Integer): void
+ estaColocado(Integer): Boolean
+ cambiarSimilitud(Integer, Integer, Integer): void
+ eliminarProducto(Integer): void
- inBounds(Integer): Integer
- nextValidPos(Integer): Integer
gestorSimilitudes: GestorSimilitudes
similitudTotal: Integer
listaOrdenProductos: ArrayList<Integer>
AlgoritmoOrdenacion
+ AlgoritmoOrdenacion():
+ ordenar(ArrayList<Integer>, ArrayList<Similitud>): ResultadoAlgoritmo
algoritmos
Usuario
+ Usuario(String, String, Tusuario):
- nombre: String
- Password: String
- rol: Tusuario
nombreUsuario: String
Password: String
rol: Tusuario
nombre: String
gestorUsers
Producto
+ Producto():
+ Producto(Integer, String):
- nombre: String
- id: Integer
id: Integer
nombre: String
productos
productosConjunto
estanteriaOrdenada
GestorUsuarios
+ GestorUsuarios():
- gestorUsers: HashMap<String, Usuario>
- actualizarPasswordEnArchivo(String, String): void
+ iniciarSession(String, String): Integer
+ altaUsuario(): void
+ despermutar_password(String): String
+ permutar_password(String): String
+ eliminarUsuarioActual(): void
+ esFuncionPermitida(String): Boolean
+ comprobar_user(String): boolean
+ cambiarPassword(String): void
+ sugerir_usuario(String): String
- guardarUsuarioEnArchivo(Usuario, String): void
gestorUsers: HashMap<String, Usuario>
AlgoritmoAproximadoKruskal
+ AlgoritmoAproximadoKruskal():
- crearCicloHamiltoniano(ArrayList<Similitud>, ArrayList<Similitud>): ResultadoAlgoritm
+ ordenar(ArrayList<Integer>, ArrayList<Similitud>): ResultadoAlgoritmo
- doDFS(ConjuntoProductosConectados): ArrayList<Integer>
- algoritmoKruskal(ArrayList<Integer>, ArrayList<Similitud>): ArrayList<Similitud>
GestorEstanteria
+ GestorEstanteria():
+ GestorEstanteria(EstanteriaOrdenada, HashMap<String, Producto>
- estanteriaOrdenada: EstanteriaOrdenada
+ añadirProductoEstanteriaOrdenada(String): void
+ setEstanteriaOrdenada(ArrayList<Integer>, Integer): void
+ eliminarProducto(String): void
+ cambiarSimilitud(Integer, Integer, Integer): void
+ listarProductos(): void
+ getIdProducto(String): Integer
+ intercambiarProductos(String, String): void
gestorSimilitudes: GestorSimilitudes
estanteriaOrdenada: EstanteriaOrdenada
listaOrdenProductos: ArrayList<Integer>
productos: HashMap<String, Producto>
GestorProductos
+ GestorProductos():
- productos: HashMap<String, Producto>
+ existeAlgunProducto(): boolean
+ getProducto(String): Producto
+ listarProductos(): void
+ existeProducto(String): boolean
+ cargarProductos(): void
+ guardarProductos(): void
+ crearProducto(String): void
+ eliminarProducto(String): void
listaProductos: HashMap<String, Producto>
productos: HashMap<String, Producto>
AlgoritmoFuerzaBruta
+ AlgoritmoFuerzaBruta():
- subAlgoritmoFuerzaBruta(ArrayList<Integer>, ArrayList<Similitud>, ArrayList<Int
+ ordenar(ArrayList<Integer>, ArrayList<Similitud>): ResultadoAlgoritmo
GestorAlgoritmo
+ GestorAlgoritmo():
+ ordenarProductos(Integer, ArrayList<Integer>, ArrayList<Similitud>): ResultadoAlgor
gestorAlgoritmo
gestorUsuarios
gestorProductos
gestorEstanteria
ProductoConectado
+ ProductoConectado():
~ hijos: ArrayList<Integer>
+ addHijo(Integer): void
hijos: ArrayList<Integer>
ControladorSimilitudes
+ ControladorSimilitudes():
+ cambiarSimilitud(int, int, Integer): void
+ obtenerIdsProductos(): ArrayList<Integer>
+ obtenerTodasSimilitudes(): ArrayList<Similitud>
+ obtenerSimilitud(Integer, Integer): Integer
+ crearSimilitudesDeProducto(int): void
+ listarSimilitudes(): void
+ eliminarSimilitudesDeProducto(Integer): void
gestorSimilitudes: GestorSimilitudes
ControladorProductos
- ControladorProductos():
- instance: ControladorProductos
+ eliminarProducto(String): void
+ existeAlgunProducto(): boolean
+ cargarProductos(): void
+ guardarProductos(): void
+ existeProducto(String): boolean
+ listarProductos(): void
+ crearProducto(String): void
+ getProducto(String): Producto
listaProductos: HashMap<String, Producto>
instance: ControladorProductos
ControladorEstanteria
- ControladorEstanteria():
- instance: ControladorEstanteria
+ setEstanteriaOrdenada(ArrayList<Integer>, Integer): void
+ intercambiarProductos(String, String): void
+ guardarEstanteria(): void
+ añadirProductoEstanteriaOrdenada(String): void
+ cambiarSimilitud(Integer, Integer, Integer): void
+ getIdProducto(String): Integer
+ cargarEstateria(): void
+ listarEstanteria(): void
+ eliminarProducto(String): void
gestorSimilitudes: GestorSimilitudes
listaOrdenada: ArrayList<Integer>
estanteriaOrdenada: EstanteriaOrdenada
productos: HashMap<String, Producto>
instance: ControladorEstanteria
ControladorAlgoritmo
+ ControladorAlgoritmo():
+ ordenarProductos(Integer, ArrayList<Integer>, ArrayList<Similitud>): ResultadoAlgor
controladorAlgoritmo
ControladorUsuarios
- ControladorUsuarios():
- instance: ControladorUsuarios
+ cambiarPassword(String): void
+ altaUsuario(): void
+ iniciarSession(String, String): Integer
+ eliminarUsuarioActual(): void
+ comprobar_user(String): boolean
instance: ControladorUsuarios
misProductos
ConjuntoProductosConectados
+ ConjuntoProductosConectados():
+ getNodo(Integer): Integer
+ addArista(Integer, Integer): void
+ getConectados(Integer, ArrayList<Integer>): ArrayList<Integer>
productosConectados: HashMap<Integer, ProductoConectado>
Similitud
+ Similitud(int, int, Integer):
+ Similitud():
- idGrande: int
- similitud: int
- idPequena: int
idPequena: int
idGrande: int
similitud: Integer
DisjointSet
+ DisjointSet(int):
+ find(int): int
+ union(int, int): void
controladorEstanteria
DomainCtrl
- DomainCtrl():
- instance: DomainCtrl
+ setNombreProducto(String, String): void
+ cargarProductos(): void
+ listarSimilitudes(): void
+ eliminarUsuarioActual(): void
+ cargarEstateria(): void
+ cambiarSimilitudes(String, String, Integer): void
+ iniciarSession(String, String): Integer
+ guardarSimilitudes(): void
+ ordenarProductos(int): void
+ guardarEstanteria(): void
+ eliminarProducto(String): void
+ altaUsuario(): void
+ añadirProductoEstanteria(String): void
+ listarProductos(): void
+ guardarProductos(): void
+ comprobar_user(String): boolean
+ verSimilitud(String, String): void
+ crearProducto(String): void
+ cargarSimilitudes(): void
+ intercambiarProductos(String, String): void
+ cambiarPassword(String): void
instance: DomainCtrl
controladorSimilitudes
controladorProductos
controladorUsuarios
ResultadoAlgoritmo
+ ResultadoAlgoritmo():
~ productos: ArrayList<Integer>
~ sumaTotal: int
productos: ArrayList<Integer>
sumaTotal: int
CompararPorSimilitud
+ CompararPorSimilitud():
+ compare(Similitud, Similitud): int
Main
+ Main():
+ main(String[]): void