

Model: Main

«enumeration» eTIPO	
A	
AB	
B	
O	

«enumeration» eSEXO	
FEMENINO	
MASCULINO	

«enumeration» eESTADO	
ESTABLE	
INESTABLE	
RECIBIO	

«enumeration» ePRIORIDAD	
uno	
dos	
tres	
cuatro	
cinco	

«enumeration» ePROVINCIA	
BuenosAires	
CABA	
Catamarca	
Chaco	
Cordoba	
Corrientes	
EntreRios	
Formosa	
Jujuy	
LaPampa	
LaRioja	
Mendoza	
Misiones	
Neuquen	
RioNegro	
Salta	
SanJuan	
SanLuis	
SantaCruz	
SantaFe	
SantiagoDelEstero	
TierraDelFuego	
Tucuman	

«enumeration» eCOLOR	
amarillo	
rojo	

cSANGRE	
-tipo: eTIPO	
-Rh: bool	
+cSANGRE(u_int, eTIPO, bool)	
-cSANGRE()	
+verificar_fecha_max(time_t): bool	
+compatibilidad(cFLUIDO*): bool	
+imprimir(): void	
+to_string(): string const	

cMEDULA	
-color: eCOLOR	
+cMEDULA(u_int, eCOLOR)	
-cMEDULA()	
+verificar_fecha_max(time_t): bool	
+compatibilidad(cFLUIDO*): bool	
+to_string(): string const	
+imprimir(): void	

cPLASMA	
-tipo: eTIPO	
+cPLASMA(u_int, eTIPO)	
-cPLASMA()	
+verificar_fecha_max(time_t): bool	
+compatibilidad(cFLUIDO*): bool	
+to_string(): string const	
+imprimir(): void	

cFLUIDO	
-#volumen: u_int	
+cFLUIDO(u_int)	
-cFLUIDO(): virtual	
+verificar_fecha_max(time_t): bool	
+compatibilidad(cFLUIDO*): bool	
+to_string(): string const	

cREGISTRO	
-fecha_extraccion: time_t	
+cREGISTRO(cFLUIDO, time_t, cCENTRO)	
+REGISTRO()	
-cREGISTRO()	
+imprimir(): void	
+to_string(): string const	

cDONANTE	
-peso: float	
+cDONANTE(cFLUIDO*, vector<REGISTRO, u_int, float, cHISTORIAL>*)	
-cDONANTE()	
+crear_registro(): void	
+imprimir(): void	
+to_string(): string const	

cHISTORIAL	
-enfermedades: bool	
-fecha_ult_tatuaje: time_t	
-tiene_tatoo: bool	
+cHISTORIAL(bool, time_t, bool)	
-cHISTORIAL()	
+imprimir(): void	
+to_string(): string const	

cBSA	
-lista_donantes: VECTOR<cDONANTE>	
-lista_receptores: VECTOR<cRECEPTOR>	
-lista_centros: VECTOR<cCENTRO>	
+cBSA(VECTOR<cDONANTE, VECTOR<cRECEPTOR, VECTOR<cCENTRO>) -cBSA()	
+agregar_paciente(cPACIENTE*): int	
+comprobar_requisitos(*cDONANTE): bool	
+buscar_posibles_receptores(*cDONANTE, VECTOR<cRECEPTOR*): void	
+elegir_receptor(VECTOR<cRECEPTOR*): *cPACIENTE	
+protocolo_transplante(*cDONANTE, *cRECEPTOR): void	
+iniciar_analisis(*cDONANTE*): cPACIENTE*	

cPACIENTE	
-#nombre: string	
-#fecha_nacimiento: time_t	
-#telefono: string	
-#sexo: eSEXO	
-#dni: string const	
-#edad: int	
+cPACIENTE(int, string, time_t, string, eSEXO, cFLUIDO*, cCENTRO*, string)	
-cPACIENTE(): virtual	
+to_string(): string const	
+operator(cPACIENTE*): bool	

cCENTRO	
-nombre: string	
-direccion: string	
-parido: string	
-provincia: ePROVINCIA	
-telefono: string	
+cCENTRO(string, string, string, ePROVINCIA, string)	
-cCENTRO()	
+realizar_transplante(): cTRANSPLANTE*	
+imprimir(): void	
+to_string(): string const	
+operator(cCENTRO*): bool	

cTRANSPLANTE	
-fecha: time_t	
-resultado: bool	
-num_transplante: int	
+cTRANSPLANTE(time_t, bool)	
-cTRANSPLANTE()	
+imprimir(): void	
+to_string(): string const	
+set_resultado(bool): void	
+get_resultado(): bool	

cRECEPTOR	
-fecha_ingreso: time_t	
-prioridad: ePRIORIDAD	
-estado: eESTADO	
+cRECEPTOR(int, string, time_t, string, eSEXO, cFLUIDO*, cCENTRO*, string, time_t, ePRIORIDAD, eESTADO)	
-cRECEPTOR()	
+imprimir(): void	
+to_string(): string const	
+operator(cRECEPTOR*): bool	