

CC3001 - Tarea 02

Alumno: Joaquín Cruz Cancino
Rut: 20200007K

Profesora: Nancy Hitschfeld

Auxiliares: Matías Ramírez
Ignacio Slater

Ayudantes: Tomás Calderón
Benjamín Del Pino
Sebastián Moreno
Esterban Ramírez
Matías Venegas

10 de Diciembre, 2020

Link del Repositorio

<https://github.com/CC3002-Methodologies/final-reality-Cubolink>

Diagramas



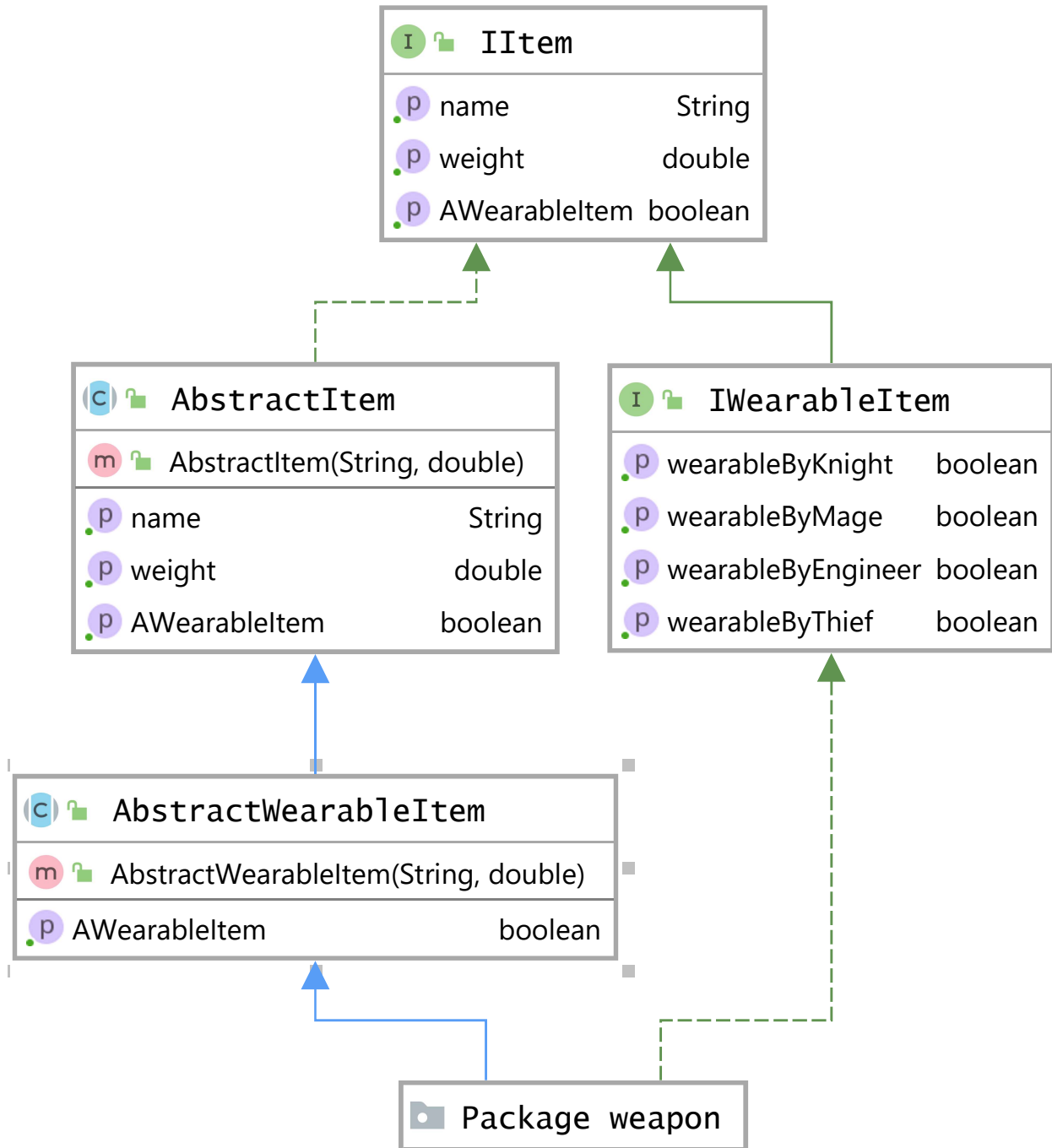
Package statuseffects



Package character



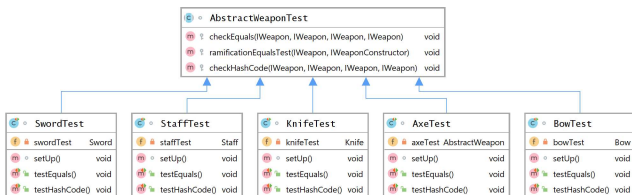
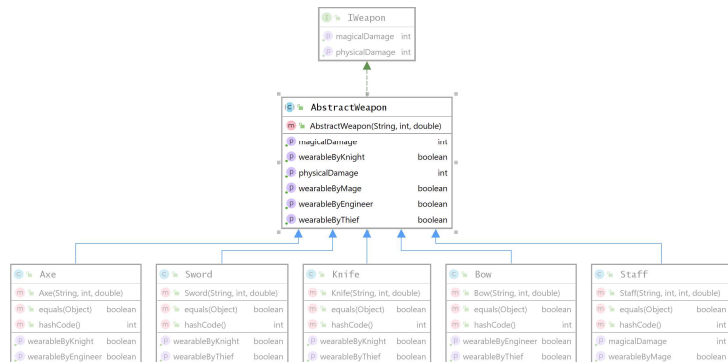
Package items



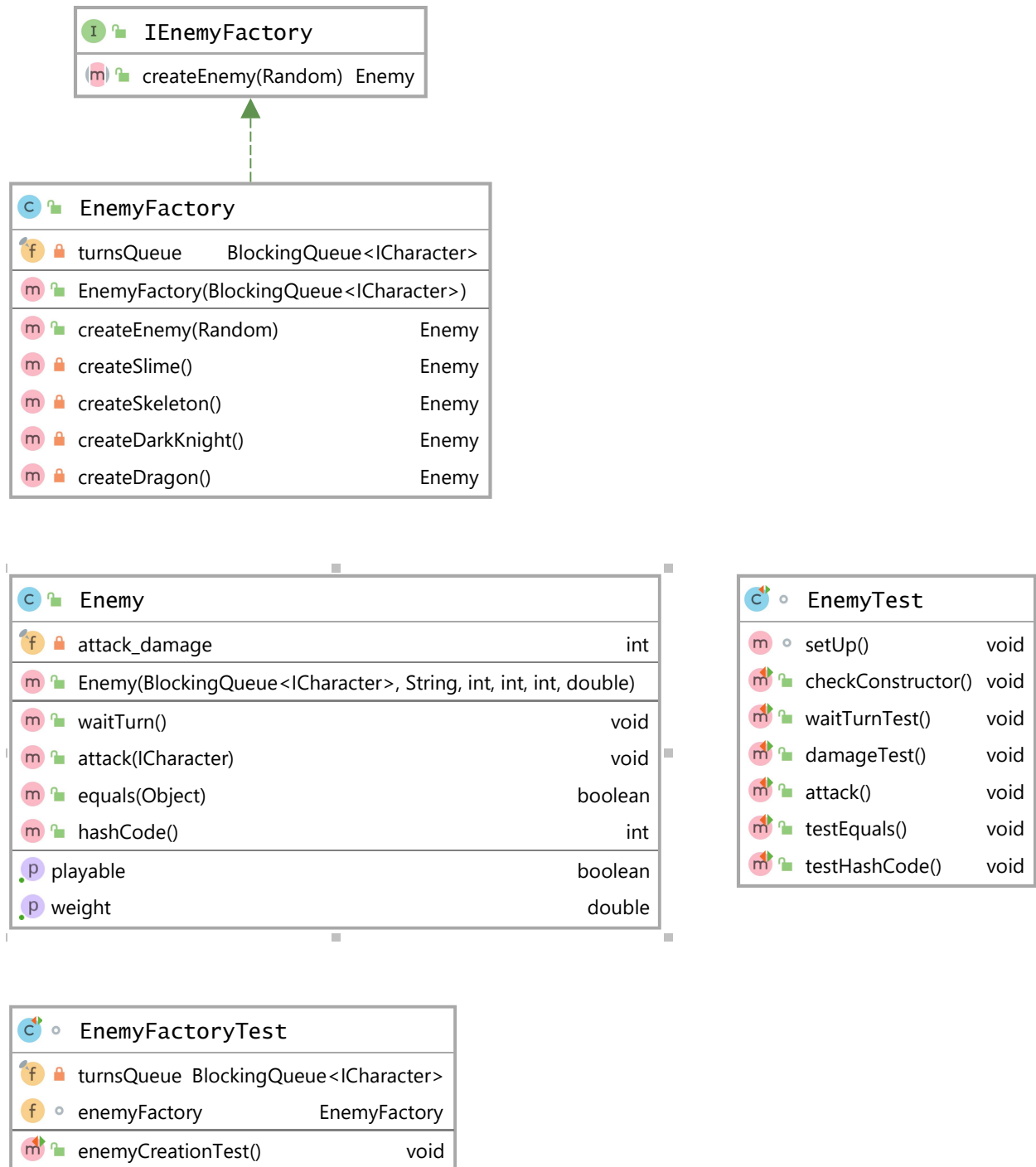
IWeaponFactory		
createBronzeAxe()	IWeapon	
createIronAxe()	IWeapon	
createSteelAxe()	IWeapon	
createSilverAxe()	IWeapon	
createBronzeBow()	IWeapon	
createIronBow()	IWeapon	
createSteelBow()	IWeapon	
createSilverBow()	IWeapon	
createBronzeKnife()	IWeapon	
createIronKnife()	IWeapon	
createSteelKnife()	IWeapon	
createSilverKnife()	IWeapon	
createBronzeSword()	IWeapon	
createIronSword()	IWeapon	
createSteelSword()	IWeapon	
createSilverSword()	IWeapon	
createNormalStaff()	IWeapon	

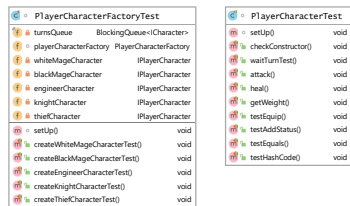
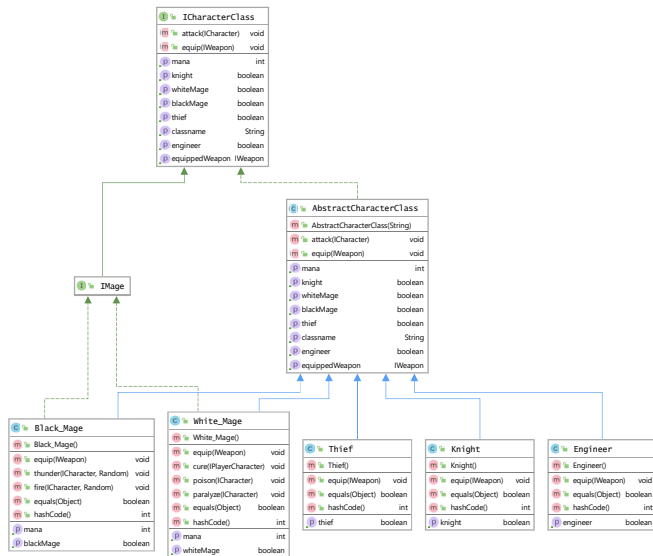
WeaponFactory		
BRONZE_NAME	String	
IRON_NAME	String	
STEEL_NAME	String	
SILVER_NAME	String	
BRONZE_BASE_DMG	int	
IRON_BASE_DMG	int	
STEEL_BASE_DMG	int	
SILVER_BASE_DMG	int	
BRONZE_BASE_WEIGHT	int	
IRON_BASE_WEIGHT	int	
STEEL_BASE_WEIGHT	int	
SILVER_BASE_WEIGHT	int	
AXE_NAME	String	
BOW_NAME	String	
KNIFE_NAME	String	
STAFF_NAME	String	
SWORD_NAME	String	
AXE_FACTOR	double	
BOW_FACTOR	double	
KNIFE_FACTOR	double	
STAFF_FACTOR	double	
SWORD_FACTOR	double	
calculateName(String, String)	String	
calculateDamage(int, double)	int	
calculateWeight(int, double)	int	
createAxe(String, int, int)	IWeapon	
createBronzeAxe()	IWeapon	
createIronAxe()	IWeapon	
createSteelAxe()	IWeapon	
createSilverAxe()	IWeapon	
createBow(String, int, int)	IWeapon	
createBronzeBow()	IWeapon	
createIronBow()	IWeapon	
createSteelBow()	IWeapon	
createSilverBow()	IWeapon	
createKnife(String, int, int)	IWeapon	
createBronzeKnife()	IWeapon	
createIronKnife()	IWeapon	
createSteelKnife()	IWeapon	
createSilverKnife()	IWeapon	
createSword(String, int, int)	IWeapon	
createBronzeSword()	IWeapon	
createIronSword()	IWeapon	
createSteelSword()	IWeapon	
createSilverSword()	IWeapon	
createNormalStaff()	IWeapon	
hashCode()	int	
equals(Object)	boolean	

WeaponFactoryTest		
weaponFactoryTest	IWeaponFactory	
bronzeAxe	IWeapon	
ironAxe	IWeapon	
steelAxe	IWeapon	
silverAxe	IWeapon	
bronzeBow	IWeapon	
ironBow	IWeapon	
steelBow	IWeapon	
silverBow	IWeapon	
bronzeSword	IWeapon	
ironSword	IWeapon	
steelSword	IWeapon	
silverSword	IWeapon	
bronzeKnife	IWeapon	
ironKnife	IWeapon	
steelKnife	IWeapon	
silverKnife	IWeapon	
normalStaff	IWeapon	
setUp()	void	
physicalDamageHierarchyTest(IWeapon[])	void	
putInOrder(IWeapon[], IWeapon, IWeapon, IWeapon, IWeapon)	void	
hierarchyTest()	void	
creationTest()	void	
testEqualsAndHashCode()	void	











IGameController	
start()	void
isCharacterWaiting()	void
end()	void
nextCharacterInQueue()	void
getCharacterInfo()	void
waitCharacter()	void
playerAttackCharacter(Character)	void
equipWeaponToCharacter(Weapon, IPlayerCharacter)	void
storeItem(Item)	void
takeItem(Item)	Item
dropItem(Item)	void
createEnemy()	void
createWhiteMagePlayer()	void
createBlackMagePlayer()	void
createEngineerPlayer()	void
createKnightPlayer()	void
createThiefPlayer()	void
createBronzeAxe()	void
createIronAxe()	void
createSteelAxe()	void
createSilverAxe()	void
createBronzeBow()	void
createIronBow()	void
createSteelBow()	void
createSilverBow()	void
createBronzeKnife()	void
createIronKnife()	void
createSteelKnife()	void
createSilverKnife()	void
createBronzeSword()	void
createIronSword()	void
createSteelSword()	void
createSilverSword()	void
createNormalStaff()	void
characterPlayerList	List<IPlayerCharacter>
itemSet	Set<Item>
theGameFinished	boolean
enemyList	List<Enemy>
currentCharacter	ICharacter
turnQueue	BlockingQueue<ICharacter>

GameController	
endGameHandler	EndGameHandler
fallenCharacterHandler	FallenCharacterHandler
characterReadyInQueueHandler	CharacterReadyInQueueHandler
random	Random
playerInventory	Inventory
turnQueue	BlockingQueue<ICharacter>
characters	List<ICharacter>
playerCharacterList	List<IPlayerCharacter>
enemiesList	List<Enemy>
weaponFactory	WeaponFactory
enemyFactory	IEnemyFactory
playerFactory	IPlayerCharacterFactory
MAX_PLAYER_CHARACTER_NUM	int
MAX_ENEMY_CHARACTER_NUM	int
current_number_of_player_characters	int
current_number_of_enemy_characters	int
GameController()	
start()	void
isCharacterWaiting()	void
end()	void
nextCharacterInQueue()	void
getCharacterInfo(IPlayerCharacter)	void
getCharacterInfo(IEnemy)	void
waitCharacter()	void
playerAttackCharacter(ICharacter)	void
equipWeaponToCharacter(IWeapon, IPlayerCharacter)	void
storeItem(Item)	void
takeItem(Item)	Item
dropItem(Item)	void
characterCreationSetUp(ICharacter)	void
enemyCreationSetUp(IEnemy)	void
playerCharacterCreationSetUp(IPlayerCharacter)	void
canCreateEnemyPlayer()	boolean
canCreateCharacterPlayer()	boolean
createEnemy()	void
createWhiteMagePlayer()	void
createBlackMagePlayer()	void
createEngineerPlayer()	void
createKnightPlayer()	void
createThiefPlayer()	void
createBronzeAxe()	void
createIronAxe()	void
createSteelAxe()	void
createSilverAxe()	void
createBronzeBow()	void
createIronBow()	void
createSteelBow()	void
createSilverBow()	void
createBronzeKnife()	void
createIronKnife()	void
createSteelKnife()	void
createSilverKnife()	void
createBronzeSword()	void
createIronSword()	void
createSteelSword()	void
createSilverSword()	void
createNormalStaff()	void
characterPlayerList	List<IPlayerCharacter>
itemSet	Set<Item>
theGameFinished	boolean
enemyList	List<Enemy>
currentCharacter	ICharacter
turnQueue	BlockingQueue<ICharacter>

GameControllerTest	
outputStream	ByteArrayOutputStream
originalOut	PrintStream
controller	GameController
weaponFactory	WeaponFactory
playerCharacterFactory	PlayerCharacterFactory
enemyFactory	EnemyFactory
muteOutput()	void
unmuteOutput()	void
setUp()	void
tearDown()	void
testStart()	void
testCheckEndGame()	void
testEnd()	void
testGetCharacterInfo()	void
testPlayerAttackCharacter()	void
testWaitCharacter()	void
equipWeaponToCharacterTest()	void
testStoreItem()	void
testGetItemGet()	void
testTakeItem()	void
testDropItem()	void
createEnemyWhenFullEnemyParty()	void
createEnemy()	void
checkEmptyParty()	void
createPlayerWhenFullParty()	void
createWhiteMagePlayer()	void
createBlackMagePlayer()	void
createEngineerPlayer()	void
createKnightPlayer()	void
createThiefPlayer()	void
createBronzeAxe()	void
createIronAxe()	void
createSteelAxe()	void
createSilverAxe()	void
createBronzeBow()	void
createIronBow()	void
createSteelBow()	void
createSilverBow()	void
createBronzeKnife()	void
createIronKnife()	void
createSteelKnife()	void
createSilverKnife()	void
createBronzeSword()	void
createIronSword()	void
createSteelSword()	void
createSilverSword()	void
createNormalStaff()	void









IInventory	
storeItem(Item)	void
takeItem(Item)	Item
dropItem(Item)	void
itemSet	Set<Item>
empty	boolean









Inventory	
inventory	Map<Item, Integer>
storeItem(Item)	void
takeItem(Item)	Item
dropItem(Item)	void
itemSet	Set<Item>
empty	boolean







InventoryTest	
inventoryTest	Inventory
itemTest1	Item
itemTest2	Item
itemTest3	Item
itemTest4	Item
setUp()	void
storeItem()	void
takeItem()	void
dropItem()	void
testEmptiness()	void

Package listeners	
-------------------	--

		FallenCharacterHandler	
		controller	GameController
		characterDefeatedEvent	PropertyChangeSupport
		FallenCharacterHandler(GameController, PropertyChangeListener)	
		propertyChange(PropertyChangeEvent)	void

		CharacterReadyInQueueHandler	
		controller	GameController
		CharacterReadyInQueueHandler(GameController)	
		propertyChange(PropertyChangeEvent)	void

		EndGameHandler	
		controller	GameController
		EndGameHandler(GameController)	
		propertyChange(PropertyChangeEvent)	void

		FallenCharacterHandlerTest	
		gameController	GameController
		eventListenerTest()	void