

Proyecto final

Generado por Doxygen 1.12.0

1 UNO - Card Game	1
1.1 Instructions	1
1.2 Event Storming	1
1.3 Core Domain	1
1.3.1 Uno Card	2
1.3.2 Player	2
1.3.3 Game	2
1.3.4 Domain Events	2
2 Índice de espacios de nombres	3
2.1 Lista de paquetes	3
3 Índice jerárquico	5
3.1 Jerarquía de clases	5
4 Índice de clases	7
4.1 Lista de clases	7
5 Índice de archivos	9
5.1 Lista de archivos	9
6 Documentación de espacios de nombres	11
6.1 Paquete application	11
6.2 Paquete application.dto	11
6.3 Paquete domain	11
6.4 Paquete domain.card	12
6.5 Paquete domain.common	12
6.6 Paquete domain.game	12
6.7 Paquete domain.game.events	13
6.8 Paquete domain.player	13
6.9 Paquete domain.testhelper	13
6.10 Paquete ui	13
6.11 Paquete ui.common	13
6.12 Paquete ui.view	13
7 Documentación de clases	15
7.1 Referencia de la clase domain.card.AbstractCard	15
7.1.1 Documentación de constructores y destructores	15
7.1.1.1 AbstractCard()	15
7.1.2 Documentación de funciones miembro	16
7.1.2.1 equals()	16
7.1.2.2 getColor()	16
7.1.2.3 getType()	16
7.1.2.4 hashCode()	16

7.1.2.5 toString()	16
7.2 Referencia de la clase domain.card.ActionCard	16
7.2.1 Documentación de constructores y destructores	17
7.2.1.1 ActionCard()	17
7.2.2 Documentación de funciones miembro	17
7.2.2.1 equals()	17
7.2.2.2 hashCode()	17
7.2.2.3 toString()	18
7.3 Referencia de la clase ui.AppFrame	18
7.3.1 Documentación de constructores y destructores	18
7.3.1.1 AppFrame()	18
7.4 Referencia de la interface domain.card.Card	18
7.4.1 Documentación de funciones miembro	19
7.4.1.1 getColor()	19
7.4.1.2 getType()	19
7.5 Referencia de la enumeración domain.card.CardColor	19
7.5.1 Documentación de datos miembro	19
7.5.1.1 BLUE	19
7.5.1.2 GREEN	19
7.5.1.3 RED	19
7.5.1.4 YELLOW	20
7.6 Referencia de la clase domain.card.CardCounter	20
7.6.1 Documentación de constructores y destructores	20
7.6.1.1 CardCounter()	20
7.6.2 Documentación de funciones miembro	20
7.6.2.1 getDrawTwoColorCounts()	20
7.6.2.2 getNumberColorCounts()	20
7.6.2.3 getNumberCounts()	20
7.6.2.4 getReverseColorCounts()	21
7.6.2.5 getSkipColorCounts()	21
7.6.2.6 getWildCardCounts()	21
7.7 Referencia de la clase domain.card.CardCounterAssertionHelper	21
7.7.1 Documentación de funciones miembro	21
7.7.1.1 assertDrawTwoCards()	21
7.7.1.2 assertNumberCards()	21
7.7.1.3 assertReverseCards()	22
7.7.1.4 assertSkipCards()	22
7.7.1.5 assertWildCards()	22
7.8 Referencia de la clase domain.card.CardDeck	22
7.8.1 Descripción detallada	22
7.8.2 Documentación de constructores y destructores	22
7.8.2.1 CardDeck()	22

7.8.3 Documentación de funciones miembro	22
7.8.3.1 getImmutableCards()	22
7.9 Referencia de la clase domain.game.events.CardDrawn	23
7.9.1 Documentación de constructores y destructores	23
7.9.1.1 CardDrawn()	23
7.9.2 Documentación de funciones miembro	23
7.9.2.1 getPlayerId()	23
7.10 Referencia de la clase domain.game.events.CardPlayed	23
7.10.1 Documentación de constructores y destructores	24
7.10.1.1 CardPlayed()	24
7.10.2 Documentación de funciones miembro	24
7.10.2.1 getPlayedCard()	24
7.10.2.2 getPlayerId()	24
7.11 Referencia de la clase domain.game.CardRules	24
7.11.1 Documentación de funciones miembro	25
7.11.1.1 isValidActionCard()	25
7.11.1.2 isValidNumberCard()	25
7.11.1.3 isValidWildCard()	25
7.12 Referencia de la clase domain.testhelper.CardTestFactory	25
7.12.1 Documentación de funciones miembro	26
7.12.1.1 createActionCard()	26
7.12.1.2 createDrawTwoCard() [1/2]	26
7.12.1.3 createDrawTwoCard() [2/2]	26
7.12.1.4 createNumberCard() [1/2]	26
7.12.1.5 createNumberCard() [2/2]	26
7.12.1.6 createReverseCard() [1/2]	26
7.12.1.7 createReverseCard() [2/2]	26
7.12.1.8 createSkipCard() [1/2]	26
7.12.1.9 createSkipCard() [2/2]	27
7.12.1.10 createWildCard()	27
7.12.1.11 createWildColorCard() [1/2]	27
7.12.1.12 createWildColorCard() [2/2]	27
7.12.1.13 createWildDrawFourCard() [1/2]	27
7.12.1.14 createWildDrawFourCard() [2/2]	27
7.13 Referencia de la enumeración domain.card.CardType	27
7.13.1 Documentación de datos miembro	28
7.13.1.1 DRAW_TWO	28
7.13.1.2 NUMBER	28
7.13.1.3 REVERSE	28
7.13.1.4 SKIP	28
7.13.1.5 WILD_COLOR	28
7.13.1.6 WILD_DRAW_FOUR	28

7.14 Referencia de la clase domain.card.CardUtil	28
7.14.1 Documentación de funciones miembro	29
7.14.1.1 isWildCard()	29
7.14.1.2 validateActionType()	29
7.14.1.3 validateColor()	29
7.14.1.4 validateNumber()	29
7.15 Referencia de la clase ui.view.CardView	29
7.15.1 Documentación de constructores y destructores	30
7.15.1.1 CardView() [1/2]	30
7.15.1.2 CardView() [2/2]	30
7.15.2 Documentación de funciones miembro	30
7.15.2.1 getDimension()	30
7.15.2.2 paintComponent()	30
7.16 Referencia de la clase ui.view.ColorPicker	30
7.16.1 Documentación de funciones miembro	31
7.16.1.1 getInstance()	31
7.16.1.2 show()	31
7.17 Referencia de la clase domain.game.DealerService	31
7.17.1 Documentación de funciones miembro	31
7.17.1.1 dealInitialHandCards()	31
7.17.1.2 shuffle() [1/2]	31
7.17.1.3 shuffle() [2/2]	31
7.17.2 Documentación de datos miembro	32
7.17.2.1 TOTAL_INITIAL_HAND_CARDS	32
7.18 Referencia de la enumeración domain.player.Direction	32
7.18.1 Documentación de datos miembro	32
7.18.1.1 CLOCKWISE	32
7.18.1.2 COUNTER_CLOCK_WISE	32
7.19 Referencia de la clase domain.common.DomainEvent	32
7.19.1 Documentación de constructores y destructores	33
7.19.1.1 DomainEvent()	33
7.19.2 Documentación de funciones miembro	33
7.19.2.1 getOccurredDate()	33
7.20 Referencia de la clase domain.common.DomainEventPublisher	33
7.20.1 Documentación de funciones miembro	33
7.20.1.1 publish()	33
7.20.1.2 reset()	33
7.20.1.3 subscribe()	34
7.20.1.4 unsubscribe()	34
7.21 Referencia de la interface domain.common.DomainEventSubscriber	34
7.21.1 Documentación de funciones miembro	34
7.21.1.1 handleEvent()	34

7.22 Referencia de la clase domain.game.DrawPile	34
7.22.1 Documentación de constructores y destructores	35
7.22.1.1 DrawPile()	35
7.22.2 Documentación de funciones miembro	35
7.22.2.1 drawCard()	35
7.22.2.2 getSize()	35
7.23 Referencia de la clase domain.common.Entity	35
7.23.1 Documentación de constructores y destructores	35
7.23.1.1 Entity() [1/2]	35
7.23.1.2 Entity() [2/2]	36
7.23.2 Documentación de funciones miembro	36
7.23.2.1 getId()	36
7.24 Referencia de la clase domain.game.Game	36
7.24.1 Documentación de constructores y destructores	37
7.24.1.1 Game()	37
7.24.2 Documentación de funciones miembro	37
7.24.2.1 drawCard()	37
7.24.2.2 getCurrentPlayer()	37
7.24.2.3 getHandCards()	37
7.24.2.4 getPlayers()	37
7.24.2.5 getWinner()	37
7.24.2.6 isOver()	37
7.24.2.7 peekTopCard()	38
7.24.2.8 playCard() [1/2]	38
7.24.2.9 playCard() [2/2]	38
7.25 Referencia de la clase application.GameAppService	38
7.25.1 Documentación de constructores y destructores	39
7.25.1.1 GameAppService()	39
7.25.2 Documentación de funciones miembro	39
7.25.2.1 drawCard()	39
7.25.2.2 getCurrentPlayer()	39
7.25.2.3 getHandCards()	39
7.25.2.4 getPlayerInfos()	39
7.25.2.5 getWinner()	40
7.25.2.6 isGameOver()	40
7.25.2.7 peekTopCard()	40
7.25.2.8 playCard()	40
7.26 Referencia de la clase domain.game.GameBuilder	40
7.26.1 Documentación de funciones miembro	41
7.26.1.1 build()	41
7.26.1.2 withPlayer()	41
7.27 Referencia de la clase domain.game.events.GameOver	41

7.27.1 Documentación de constructores y destructores	41
7.27.1.1 GameOver()	41
7.27.2 Documentación de funciones miembro	42
7.27.2.1 getWinner()	42
7.28 Referencia de la clase ui.view.GameStatusView	42
7.28.1 Documentación de constructores y destructores	42
7.28.1.1 GameStatusView()	42
7.28.2 Documentación de funciones miembro	42
7.28.2.1 handleEvent()	42
7.28.2.2 paintComponent()	43
7.28.2.3 setError()	43
7.29 Referencia de la clase domain.player.HandCardList	43
7.29.1 Documentación de funciones miembro	43
7.29.1.1 addCard()	43
7.29.1.2 getCardStream()	43
7.29.1.3 hasCard()	43
7.29.1.4 removeCard()	44
7.29.1.5 size()	44
7.30 Referencia de la interface application.IGameAppService	44
7.30.1 Documentación de funciones miembro	44
7.30.1.1 drawCard()	44
7.30.1.2 getCurrentPlayer()	44
7.30.1.3 getHandCards()	45
7.30.1.4 getPlayerInfos()	45
7.30.1.5 getWinner()	45
7.30.1.6 isGameOver()	45
7.30.1.7 peekTopCard()	45
7.30.1.8 playCard()	45
7.31 Referencia de la clase domain.player.ImmutablePlayer	46
7.31.1 Documentación de constructores y destructores	46
7.31.1.1 ImmutablePlayer()	46
7.31.2 Documentación de funciones miembro	46
7.31.2.1 getHandCards()	46
7.31.2.2 getId()	46
7.31.2.3 getName()	46
7.31.2.4 getTotalCards()	47
7.32 Referencia de la clase domain.card.NumberCard	47
7.32.1 Documentación de constructores y destructores	47
7.32.1.1 NumberCard()	47
7.32.2 Documentación de funciones miembro	48
7.32.2.1 equals()	48
7.32.2.2 getValue()	48

7.32.2.3 hashCode()	48
7.32.2.4 toString()	48
7.33 Referencia de la clase domain.player.Player	48
7.33.1 Documentación de constructores y destructores	49
7.33.1.1 Player()	49
7.33.2 Documentación de funciones miembro	49
7.33.2.1 addToHandCards()	49
7.33.2.2 getHandCards()	49
7.33.2.3 getName()	50
7.33.2.4 hasHandCard()	50
7.33.2.5 removePlayedCard()	50
7.33.2.6 toImmutable()	50
7.34 Referencia de la clase application.dto.PlayerInfoDTO	50
7.34.1 Documentación de constructores y destructores	50
7.34.1.1 PlayerInfoDTO()	50
7.34.2 Documentación de funciones miembro	51
7.34.2.1 getId()	51
7.34.2.2 getName()	51
7.35 Referencia de la clase domain.player.PlayerRoundIterator	51
7.35.1 Documentación de constructores y destructores	51
7.35.1.1 PlayerRoundIterator()	51
7.35.2 Documentación de funciones miembro	51
7.35.2.1 getCurrentPlayer()	51
7.35.2.2 getPlayerById()	51
7.35.2.3 next()	52
7.35.2.4 reverseDirection()	52
7.35.2.5 stream()	52
7.36 Referencia de la clase domain.testhelper.PlayerTestFactory	52
7.36.1 Documentación de funciones miembro	52
7.36.1.1 createPlayers()	52
7.37 Referencia de la clase ui.view.PlayerView	52
7.37.1 Documentación de constructores y destructores	53
7.37.1.1 PlayerView()	53
7.37.2 Documentación de funciones miembro	53
7.37.2.1 handleEvent()	53
7.38 Referencia de la clase ui.common.StyleUtil	53
7.38.1 Documentación de funciones miembro	54
7.38.1.1 convertCardColor()	54
7.38.1.2 getValueToDisplay()	54
7.38.2 Documentación de datos miembro	54
7.38.2.1 blackColor	54
7.38.2.2 blueColor	54

7.38.2.3	DEFAULT_FONT	54
7.38.2.4	greenColor	54
7.38.2.5	redColor	54
7.38.2.6	yellowColor	54
7.39	Referencia de la clase ui.view.tableView	55
7.39.1	Documentación de constructores y destructores	55
7.39.1.1	tableView()	55
7.39.2	Documentación de funciones miembro	55
7.39.2.1	handleEvent()	55
7.40	Referencia de la clase domain.TestHandCardList	55
7.40.1	Documentación de funciones miembro	56
7.40.1.1	GivenDifferentNumberCard_ShouldNotExist()	56
7.40.1.2	GivenSameNumberCard_ShouldExist()	56
7.41	Referencia de la clase UnoApp	56
7.41.1	Documentación de funciones miembro	56
7.41.1.1	main()	56
7.42	Referencia de la clase domain.card.WildCard	56
7.42.1	Documentación de constructores y destructores	57
7.42.1.1	WildCard() [1/2]	57
7.42.1.2	WildCard() [2/2]	57
7.42.2	Documentación de funciones miembro	57
7.42.2.1	equals()	57
7.42.2.2	hashCode()	57
7.42.2.3	toString()	57
8	Documentación de archivos	59
8.1	Referencia del archivo project docs/README.md	59
8.2	Referencia del archivo src/main/java/application/dto/PlayerInfoDTO.java	59
8.3	Referencia del archivo src/main/java/application/GameAppService.java	59
8.4	Referencia del archivo src/main/java/application/IGameAppService.java	60
8.5	Referencia del archivo src/main/java/domain/card/AbstractCard.java	60
8.6	Referencia del archivo src/main/java/domain/card/ActionCard.java	60
8.7	Referencia del archivo src/main/java/domain/card/Card.java	60
8.8	Referencia del archivo src/main/java/domain/card/CardColor.java	61
8.9	Referencia del archivo src/main/java/domain/card/CardDeck.java	61
8.10	Referencia del archivo src/main/java/domain/card/CardType.java	61
8.11	Referencia del archivo src/main/java/domain/card/CardUtil.java	61
8.12	Referencia del archivo src/main/java/domain/card/NumberCard.java	62
8.13	Referencia del archivo src/main/java/domain/card/WildCard.java	62
8.14	Referencia del archivo src/main/java/domain/common/DomainEvent.java	62
8.15	Referencia del archivo src/main/java/domain/common/DomainEventPublisher.java	63
8.16	Referencia del archivo src/main/java/domain/common/DomainEventSubscriber.java	63

8.17 Referencia del archivo src/main/java/domain/common/Entity.java	63
8.18 Referencia del archivo src/main/java/domain/game/CardRules.java	63
8.19 Referencia del archivo src/main/java/domain/game/DealerService.java	64
8.20 Referencia del archivo src/main/java/domain/game/DrawPile.java	64
8.21 Referencia del archivo src/main/java/domain/game/events/CardDrawn.java	64
8.22 Referencia del archivo src/main/java/domain/game/events/CardPlayed.java	65
8.23 Referencia del archivo src/main/java/domain/game/events/GameOver.java	65
8.24 Referencia del archivo src/main/java/domain/game/Game.java	65
8.25 Referencia del archivo src/main/java/domain/game/GameBuilder.java	66
8.26 Referencia del archivo src/main/java/domain/player/Direction.java	66
8.27 Referencia del archivo src/main/java/domain/player/HandCardList.java	66
8.28 Referencia del archivo src/main/java/domain/player/ImmutablePlayer.java	67
8.29 Referencia del archivo src/main/java/domain/player/Player.java	67
8.30 Referencia del archivo src/main/java/domain/player/PlayerRoundIterator.java	67
8.31 Referencia del archivo src/main/java/ui/AppFrame.java	68
8.32 Referencia del archivo src/main/java/ui/common/StyleUtil.java	68
8.33 Referencia del archivo src/main/java/ui/view/CardView.java	68
8.34 Referencia del archivo src/main/java/ui/view/ColorPicker.java	68
8.35 Referencia del archivo src/main/java/ui/view/GameStatusView.java	69
8.36 Referencia del archivo src/main/java/ui/view/PlayerView.java	69
8.37 Referencia del archivo src/main/java/ui/view/TableView.java	69
8.38 Referencia del archivo src/main/java/UnoApp.java	70
8.39 Referencia del archivo src/test/java/domain/card/CardCounter.java	70
8.40 Referencia del archivo src/test/java/domain/card/CardCounterAssertionHelper.java	70
8.41 Referencia del archivo src/test/java/domain/card/TestCard.java	71
8.42 Referencia del archivo src/test/java/domain/card/TestCardDeck.java	71
8.43 Referencia del archivo src/test/java/domain/card/TestCardUtil.java	71
8.44 Referencia del archivo src/test/java/domain/common/TestDomainEventPublisher.java	71
8.45 Referencia del archivo src/test/java/domain/game/TestCardRules.java	72
8.46 Referencia del archivo src/test/java/domain/game/TestDealerService.java	72
8.47 Referencia del archivo src/test/java/domain/game/TestDrawPile.java	72
8.48 Referencia del archivo src/test/java/domain/game/TestGameBuilder.java	73
8.49 Referencia del archivo src/test/java/domain/game/TestGamePlay.java	73
8.50 Referencia del archivo src/test/java/domain/game/TestGameStart.java	73
8.51 Referencia del archivo src/test/java/domain/player/TestPlayersIterator.java	74
8.52 Referencia del archivo src/test/java/domain/player/TestHandCardList.java	74
8.53 Referencia del archivo src/test/java/domain/TestHandCardList.java	74
8.54 Referencia del archivo src/test/java/domain/testhelper/CardTestFactory.java	75
8.55 Referencia del archivo src/test/java/domain/testhelper/PlayerTestFactory.java	75

Capítulo 1

UNO - Card Game

v2.0: Rebuilt the core game logic in domain-driven-design.

Note: GUI is built merely to try out the core domain logic. It is not suitable for serious game. :)

1.1. Instructions

Run Application

Run the `main` method in the **UnoApp** (pág. 56) class or use the following command in the terminal:

```
./gradlew run
```

Run Unit Tests

```
./gradlew test
```

1.2. Event Storming

.

1.3. Core Domain

The core game logic is built according to `official uno rules`.

1.3.1. Uno Card

Cards are value objects, i.e. immutable. The following card types are available in Uno:

- Number Card
- Skip Action Card
- Reverse Action Card
- Draw Two Action Card
- Wild Color Card
- Wild Draw Four Card

Initially wild cards don't have a color. When drawn, a new value object is created with the chosen color.

1.3.2. Player

`Player` is an entity which contains a list of hand cards to play.

`PlayerRoundIterator` manages the players and switches turn as if the players are in a round table.

1.3.3. Game

`Game` is the aggregate which maintains the state of **players**, **draw pile** and **discard pile** as the cards are played.

1.3.4. Domain Events

`Game` aggregate produces domain events using `DomainEventPublisher`.

Subscribers can register for these events and handle them accordingly.

Capítulo 2

Índice de espacios de nombres

2.1. Lista de paquetes

Estos son los paquetes con breves descripciones (si están disponibles):

application	11
application.dto	11
domain	11
domain.card	12
domain.common	12
domain.game	12
domain.game.events	13
domain.player	13
domain.testhelper	13
ui	13
ui.common	13
ui.view	13

Capítulo 3

Índice jerárquico

3.1. Jerarquía de clases

Este listado de herencia está ordenado de forma general pero no está en orden alfabético estricto:

domain.card.CardColor	19
domain.card.CardCounter	20
domain.card.CardCounterAssertionHelper	21
domain.card.CardDeck	22
domain.game.CardRules	24
domain.testhelper.CardTestFactory	25
domain.card.CardType	27
domain.card.CardUtil	28
ui.view.ColorPicker	30
domain.game.DealerService	31
domain.player.Direction	32
domain.common.DomainEvent	32
domain.game.events.CardDrawn	23
domain.game.events.CardPlayed	23
domain.game.events.GameOver	41
domain.common.DomainEventPublisher	33
domain.common.DomainEventSubscriber	34
ui.view.GameStatusView	42
ui.view.PlayerView	52
ui.view.TableView	55
domain.game.DrawPile	34
domain.common.Entity	35
domain.game.Game	36
domain.player.Player	48
domain.game.GameBuilder	40
domain.player.HandCardList	43
JFrame	
ui.AppFrame	18
JPanel	
ui.view.CardView	29
ui.view.GameStatusView	42
ui.view.PlayerView	52
ui.view.TableView	55
domain.player.PlayerRoundIterator	51

domain.testhelper.PlayerTestFactory	52
ui.common.StyleUtil	53
domain.TestHandCardList	55
UnoApp	56
Serializable	
application.IGameAppService	44
application.GameAppService	38
application.dto.PlayerInfoDTO	50
domain.card.Card	18
domain.card.AbstractCard	15
domain.card.ActionCard	16
domain.card.NumberCard	47
domain.card.WildCard	56
domain.player.ImmutablePlayer	46

Capítulo 4

Índice de clases

4.1. Lista de clases

Lista de clases, estructuras, uniones e interfaces con breves descripciones:

domain.card.AbstractCard	15
domain.card.ActionCard	16
ui.AppFrame	18
domain.card.Card	18
domain.card.CardColor	19
domain.card.CardCounter	20
domain.card.CardCounterAssertionHelper	21
domain.card.CardDeck	22
domain.game.events.CardDrawn	23
domain.game.events.CardPlayed	23
domain.game.CardRules	24
domain.testhelper.CardTestFactory	25
domain.card.CardType	27
domain.card.CardUtil	28
ui.view.CardView	29
ui.view.ColorPicker	30
domain.game.DealerService	31
domain.player.Direction	32
domain.common.DomainEvent	32
domain.common.DomainEventPublisher	33
domain.common.DomainEventSubscriber	34
domain.game.DrawPile	34
domain.common.Entity	35
domain.game.Game	36
application.GameAppService	38
domain.game.GameBuilder	40
domain.game.events.GameOver	41
ui.view.GameStatusView	42
domain.player.HandCardList	43
application.IGameAppService	44
domain.player.ImmutablePlayer	46
domain.card.NumberCard	47
domain.player.Player	48
application.dto.PlayerInfoDTO	50
domain.player.PlayerRoundIterator	51

domain.testhelper.PlayerTestFactory	52
ui.view.PlayerView	52
ui.common.StyleUtil	53
ui.view.TableView	55
domain.TestHandCardList	55
UnoApp	56
domain.card.WildCard	56

Capítulo 5

Índice de archivos

5.1. Lista de archivos

Lista de todos los archivos con breves descripciones:

src/main/java/ UnoApp.java	70
src/main/java/application/ GameAppService.java	59
src/main/java/application/ IGameAppService.java	60
src/main/java/application/dto/ PlayerInfoDTO.java	59
src/main/java/domain/card/ AbstractCard.java	60
src/main/java/domain/card/ ActionCard.java	60
src/main/java/domain/card/ Card.java	60
src/main/java/domain/card/ CardColor.java	61
src/main/java/domain/card/ CardDeck.java	61
src/main/java/domain/card/ CardType.java	61
src/main/java/domain/card/ CardUtil.java	61
src/main/java/domain/card/ NumberCard.java	62
src/main/java/domain/card/ WildCard.java	62
src/main/java/domain/common/ DomainEvent.java	62
src/main/java/domain/common/ DomainEventPublisher.java	63
src/main/java/domain/common/ DomainEventSubscriber.java	63
src/main/java/domain/common/ Entity.java	63
src/main/java/domain/game/ CardRules.java	63
src/main/java/domain/game/ DealerService.java	64
src/main/java/domain/game/ DrawPile.java	64
src/main/java/domain/game/ Game.java	65
src/main/java/domain/game/ GameBuilder.java	66
src/main/java/domain/game/events/ CardDrawn.java	64
src/main/java/domain/game/events/ CardPlayed.java	65
src/main/java/domain/game/events/ GameOver.java	65
src/main/java/domain/player/ Direction.java	66
src/main/java/domain/player/ HandCardList.java	66
src/main/java/domain/player/ ImmutablePlayer.java	67
src/main/java/domain/player/ Player.java	67
src/main/java/domain/player/ PlayerRoundIterator.java	67
src/main/java/ui/ AppFrame.java	68
src/main/java/ui/common/ StyleUtil.java	68
src/main/java/ui/view/ CardView.java	68
src/main/java/ui/view/ ColorPicker.java	68
src/main/java/ui/view/ GameStatusView.java	69

src/main/java/ui/view/ PlayerView.java	69
src/main/java/ui/view/ TableView.java	69
src/test/java/domain/ TestHandCardList.java	74
src/test/java/domain/card/ CardCounter.java	70
src/test/java/domain/card/ CardCounterAssertionHelper.java	70
src/test/java/domain/card/ TestCard.java	71
src/test/java/domain/card/ TestCardDeck.java	71
src/test/java/domain/card/ TestCardUtil.java	71
src/test/java/domain/common/ TestDomainEventPublisher.java	71
src/test/java/domain/game/ TestCardRules.java	72
src/test/java/domain/game/ TestDealerService.java	72
src/test/java/domain/game/ TestDrawPile.java	72
src/test/java/domain/game/ TestGameBuilder.java	73
src/test/java/domain/game/ TestGamePlay.java	73
src/test/java/domain/game/ TestGameStart.java	73
src/test/java/domain/player/ TestHandCardList.java	74
src/test/java/domain/player/ TestPlayersIterator.java	74
src/test/java/domain/testhelper/ CardTestFactory.java	75
src/test/java/domain/testhelper/ PlayerTestFactory.java	75

Capítulo 6

Documentación de espacios de nombres

6.1. Paquete application

Paquetes

- package **dto**

Clases

- class **GameAppService**
- interface **IGameAppService**

6.2. Paquete application.dto

Clases

- class **PlayerInfoDTO**

6.3. Paquete domain

Paquetes

- package **card**
- package **common**
- package **game**
- package **player**
- package **testhelper**

Clases

- class **TestHandCardList**

6.4. Paquete domain.card

Clases

- class **AbstractCard**
- class **ActionCard**
- interface **Card**
- enum **CardColor**
- class **CardCounter**
- class **CardCounterAssertionHelper**
- class **CardDeck**
- enum **CardType**
- class **CardUtil**
- class **NumberCard**
- class **TestCard**
- class **TestCardDeck**
- class **TestCardUtil**
- class **WildCard**

6.5. Paquete domain.common

Clases

- class **DomainEvent**
- class **DomainEventPublisher**
- interface **DomainEventSubscriber**
- class **Entity**
- class **TestDomainEventPublisher**

6.6. Paquete domain.game

Paquetes

- package **events**

Clases

- class **CardRules**
- class **DealerService**
- class **DrawPile**
- class **Game**
- class **GameBuilder**
- class **TestCardRules**
- class **TestDealerService**
- class **TestDrawPile**
- class **TestGameBuilder**
- class **TestGamePlay**
- class **TestGameStart**

6.7. Paquete domain.game.events

Classes

- class **CardDrawn**
- class **CardPlayed**
- class **GameOver**

6.8. Paquete domain.player

Classes

- enum **Direction**
- class **HandCardList**
- class **ImmutablePlayer**
- class **Player**
- class **PlayerRoundIterator**
- class **TestHandCardList**
- class **TestPlayersIterator**

6.9. Paquete domain.testhelper

Classes

- class **CardTestFactory**
- class **PlayerTestFactory**

6.10. Paquete ui

Paquetes

- package **common**
- package **view**

Classes

- class **AppFrame**

6.11. Paquete ui.common

Classes

- class **StyleUtil**

6.12. Paquete ui.view

Classes

- class **CardView**
- class **ColorPicker**
- class **GameStatusView**
- class **PlayerView**
- class **TableView**

Capítulo 7

Documentación de clases

7.1. Referencia de la clase domain.card.AbstractCard

Diagrama de herencia de domain.card.AbstractCard

Diagrama de colaboración de domain.card.AbstractCard:

Métodos públicos

- **CardType** `getType ()`
- **CardColor** `getColor ()`
- abstract boolean **equals** (Object o)
- abstract int **hashCode** ()
- abstract String **toString** ()

Métodos públicos heredados de domain.card.Card

Métodos protegidos

- **AbstractCard** (**CardType** type, **CardColor** color)

7.1.1. Documentación de constructores y destructores

7.1.1.1. AbstractCard()

```
domain.card.AbstractCard.AbstractCard (
    CardType type,
    CardColor color) [protected]
```

7.1.2. Documentación de funciones miembro

7.1.2.1. equals()

```
abstract boolean domain.card.AbstractCard.equals (
    Object o) [abstract]
```

Reimplementado en **domain.card.ActionCard** (pág. 17), **domain.card.NumberCard** (pág. 48) y **domain.card.WildCard** (pág. 57).

7.1.2.2. getColor()

```
CardColor domain.card.AbstractCard.getColor ()
```

Implementa **domain.card.Card** (pág. 19).

Gráfico de llamadas a esta función:

7.1.2.3. getType()

```
CardType domain.card.AbstractCard.getType ()
```

Implementa **domain.card.Card** (pág. 19).

Gráfico de llamadas a esta función:

7.1.2.4. hashCode()

```
abstract int domain.card.AbstractCard.hashCode () [abstract]
```

Reimplementado en **domain.card.ActionCard** (pág. 17), **domain.card.NumberCard** (pág. 48) y **domain.card.WildCard** (pág. 57).

7.1.2.5. toString()

```
abstract String domain.card.AbstractCard.toString () [abstract]
```

Reimplementado en **domain.card.ActionCard** (pág. 18), **domain.card.NumberCard** (pág. 48) y **domain.card.WildCard** (pág. 57).

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/card/ **AbstractCard.java**

7.2. Referencia de la clase domain.card.ActionCard

Diagrama de herencia de domain.card.ActionCard

Diagrama de colaboración de domain.card.ActionCard:

Métodos públicos

- **ActionCard** (**CardType** type, **CardColor** color)
- boolean **equals** (Object o)
- int **hashCode** ()
- String **toString** ()

Métodos públicos heredados de domain.card.AbstractCard

- **CardType** **getType** ()
- **CardColor** **getColor** ()

Métodos públicos heredados de domain.card.Card

Otros miembros heredados

Métodos protegidos heredados de domain.card.AbstractCard

- **AbstractCard** (**CardType** type, **CardColor** color)

7.2.1. Documentación de constructores y destructores

7.2.1.1. ActionCard()

```
domain.card.ActionCard.ActionCard (  
    CardType type,  
    CardColor color)
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.2.2. Documentación de funciones miembro

7.2.2.1. equals()

```
boolean domain.card.ActionCard.equals (  
    Object o)
```

Reimplementado de **domain.card.AbstractCard** (pág. 16).

Gráfico de llamadas de esta función:

7.2.2.2. hashCode()

```
int domain.card.ActionCard.hashCode ()
```

Reimplementado de **domain.card.AbstractCard** (pág. 16).

Gráfico de llamadas de esta función:

7.2.2.3. toString()

```
String domain.card.ActionCard.toString ()
```

Reimplementado de **domain.card.AbstractCard** (pág. 16).

Gráfico de llamadas de esta función:

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/card/ **ActionCard.java**

7.3. Referencia de la clase ui.AppFrame

Diagrama de herencia de ui.AppFrame

Diagrama de colaboración de ui.AppFrame:

Métodos públicos

- **AppFrame** (**IGameAppService** appService)

7.3.1. Documentación de constructores y destructores

7.3.1.1. AppFrame()

```
ui.AppFrame.AppFrame (
    IGameAppService appService)
```

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/ui/ **AppFrame.java**

7.4. Referencia de la interface domain.card.Card

Diagrama de herencia de domain.card.Card

Diagrama de colaboración de domain.card.Card:

Métodos públicos

- **CardType** getType ()
- **CardColor** getColor ()

7.4.1. Documentación de funciones miembro

7.4.1.1. getColor()

CardColor domain.card.Card.getColor ()

Implementado en **domain.card.AbstractCard** (pág. 16).

Gráfico de llamadas a esta función:

7.4.1.2. getType()

CardType domain.card.Card.getType ()

Implementado en **domain.card.AbstractCard** (pág. 16).

Gráfico de llamadas a esta función:

La documentación de esta interface está generada del siguiente archivo:

- src/main/java/domain/card/ **Card.java**

7.5. Referencia de la enumeración domain.card.CardColor

Atributos públicos

- **RED**
- **GREEN**
- **BLUE**
- **YELLOW**

7.5.1. Documentación de datos miembro

7.5.1.1. BLUE

domain.card.CardColor.BLUE

7.5.1.2. GREEN

domain.card.CardColor.GREEN

7.5.1.3. RED

domain.card.CardColor.RED

7.5.1.4. YELLOW

```
domain.card.CardColor.YELLOW
```

La documentación de esta enumeración está generada del siguiente archivo:

- src/main/java/domain/card/ **CardColor.java**

7.6. Referencia de la clase domain.card.CardCounter

Métodos públicos

- **CardCounter** (List< **Card** > cards)
- int[] **getNumberCounts** ()
- int[] **getNumberColorCounts** ()
- int[] **getSkipColorCounts** ()
- int[] **getReverseColorCounts** ()
- int[] **getDrawTwoColorCounts** ()
- long **getWildCardCounts** (**CardType** type)

7.6.1. Documentación de constructores y destructores

7.6.1.1. CardCounter()

```
domain.card.CardCounter.CardCounter (
    List< Card > cards)
```

7.6.2. Documentación de funciones miembro

7.6.2.1. getDrawTwoColorCounts()

```
int[] domain.card.CardCounter.getDrawTwoColorCounts ()
```

Gráfico de llamadas de esta función:

7.6.2.2. getNumberColorCounts()

```
int[] domain.card.CardCounter.getNumberColorCounts ()
```

7.6.2.3. getNumberCounts()

```
int[] domain.card.CardCounter.getNumberCounts ()
```


7.6.2.4. getReverseColorCounts()

```
int[] domain.card.CardCounter.getReverseColorCounts ()
```

Gráfico de llamadas de esta función:

7.6.2.5. getSkipColorCounts()

```
int[] domain.card.CardCounter.getSkipColorCounts ()
```

Gráfico de llamadas de esta función:

7.6.2.6. getWildCardCounts()

```
long domain.card.CardCounter.getWildCardCounts (
    CardType type)
```

La documentación de esta clase está generada del siguiente archivo:

- src/test/java/domain/card/ **CardCounter.java**

7.7. Referencia de la clase domain.card.CardCounterAssertionHelper

Métodos públicos estáticos

- static void **assertNumberCards** (List< **Card** > cards)
- static void **assertSkipCards** (List< **Card** > cards)
- static void **assertReverseCards** (List< **Card** > cards)
- static void **assertDrawTwoCards** (List< **Card** > cards)
- static void **assertWildCards** (List< **Card** > cards)

7.7.1. Documentación de funciones miembro

7.7.1.1. assertDrawTwoCards()

```
static void domain.card.CardCounterAssertionHelper.assertDrawTwoCards (
    List< Card > cards) [static]
```

7.7.1.2. assertNumberCards()

```
static void domain.card.CardCounterAssertionHelper.assertNumberCards (
    List< Card > cards) [static]
```

7.7.1.3. `assertReverseCards()`

```
static void domain.card.CardCounterAssertionHelper.assertReverseCards (
    List< Card > cards) [static]
```

7.7.1.4. `assertSkipCards()`

```
static void domain.card.CardCounterAssertionHelper.assertSkipCards (
    List< Card > cards) [static]
```

7.7.1.5. `assertWildCards()`

```
static void domain.card.CardCounterAssertionHelper.assertWildCards (
    List< Card > cards) [static]
```

La documentación de esta clase está generada del siguiente archivo:

- `src/test/java/domain/card/ CardCounterAssertionHelper.java`

7.8. Referencia de la clase `domain.card.CardDeck`

Métodos públicos

- `CardDeck ()`
- `List< Card > getImmutableCards ()`

7.8.1. Descripción detallada

UNO **Card** (pág.18) Deck consists of 108 cards. See [https://en.wikipedia.org/wiki/Uno_\(card_game\)#/media/File:UNO_cards_deck.svg](https://en.wikipedia.org/wiki/Uno_(card_game)#/media/File:UNO_cards_deck.svg) for list of card types.

7.8.2. Documentación de constructores y destructores

7.8.2.1. `CardDeck()`

```
domain.card.CardDeck.CardDeck ()
```

7.8.3. Documentación de funciones miembro

7.8.3.1. `getImmutableCards()`

```
List< Card > domain.card.CardDeck.getImmutableCards ()
```

Gráfico de llamadas a esta función:

La documentación de esta clase está generada del siguiente archivo:

- `src/main/java/domain/card/ CardDeck.java`

7.9. Referencia de la clase domain.game.events.CardDrawn

Diagrama de herencia de domain.game.events.CardDrawn

Diagrama de colaboración de domain.game.events.CardDrawn:

Métodos públicos

- **CardDrawn** (UUID playerId)
- UUID **getPlayerId** ()

Métodos públicos heredados de domain.common.DomainEvent

- LocalDate **getOccurredDate** ()

Otros miembros heredados

Métodos protegidos heredados de domain.common.DomainEvent

- **DomainEvent** ()

7.9.1. Documentación de constructores y destructores

7.9.1.1. CardDrawn()

```
domain.game.events.CardDrawn.CardDrawn (  
    UUID playerId)
```

7.9.2. Documentación de funciones miembro

7.9.2.1. getPlayerId()

```
UUID domain.game.events.CardDrawn.getPlayerId ()
```

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/game/events/ **CardDrawn.java**

7.10. Referencia de la clase domain.game.events.CardPlayed

Diagrama de herencia de domain.game.events.CardPlayed

Diagrama de colaboración de domain.game.events.CardPlayed:

Métodos públicos

- **CardPlayed** (UUID playerId, **Card** playedCard)
- UUID **getPlayerId** ()
- **Card** **getPlayedCard** ()

Métodos públicos heredados de domain.common.DomainEvent

- LocalDate **getOccurredDate** ()

Otros miembros heredados**Métodos protegidos heredados de domain.common.DomainEvent**

- **DomainEvent** ()

7.10.1. Documentación de constructores y destructores**7.10.1.1. CardPlayed()**

```
domain.game.events.CardPlayed.CardPlayed (
    UUID playerId,
    Card playedCard)
```

7.10.2. Documentación de funciones miembro**7.10.2.1. getPlayedCard()**

```
Card domain.game.events.CardPlayed.getPlayedCard ()
```

7.10.2.2. getPlayerId()

```
UUID domain.game.events.CardPlayed.getPlayerId ()
```

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/game/events/ **CardPlayed.java**

7.11. Referencia de la clase domain.game.CardRules**Métodos públicos estáticos**

- static boolean **isValidNumberCard** (**Card** topCard, **NumberCard** playedCard)
- static boolean **isValidActionCard** (**Card** topCard, **ActionCard** playedCard)
- static boolean **isValidWildCard** (**WildCard** playedCard)

7.11.1. Documentación de funciones miembro

7.11.1.1. isValidActionCard()

```
static boolean domain.game.CardRules.isValidActionCard (
    Card topCard,
    ActionCard playedCard) [static]
```

Gráfico de llamadas de esta función:

7.11.1.2. isValidNumberCard()

```
static boolean domain.game.CardRules.isValidNumberCard (
    Card topCard,
    NumberCard playedCard) [static]
```

Gráfico de llamadas de esta función:

7.11.1.3. isValidWildCard()

```
static boolean domain.game.CardRules.isValidWildCard (
    WildCard playedCard) [static]
```

Gráfico de llamadas de esta función:

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/game/ CardRules.java

7.12. Referencia de la clase domain.testhelper.CardTestFactory

Métodos públicos estáticos

- static NumberCard createNumberCard ()
- static NumberCard createNumberCard (int value, CardColor color)
- static ActionCard createActionCard (CardType type, CardColor color)
- static ActionCard createSkipCard ()
- static ActionCard createSkipCard (CardColor color)
- static ActionCard createReverseCard ()
- static ActionCard createReverseCard (CardColor color)
- static ActionCard createDrawTwoCard ()
- static ActionCard createDrawTwoCard (CardColor color)
- static WildCard createWildCard (CardType type)
- static WildCard createWildColorCard ()
- static WildCard createWildColorCard (CardColor color)
- static WildCard createWildDrawFourCard ()
- static WildCard createWildDrawFourCard (CardColor color)

7.12.1. Documentación de funciones miembro

7.12.1.1. createActionCard()

```
static ActionCard domain.testhelper.CardTestFactory.createActionCard (  
    CardType type,  
    CardColor color) [static]
```

7.12.1.2. createDrawTwoCard() [1/2]

```
static ActionCard domain.testhelper.CardTestFactory.createDrawTwoCard () [static]
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.12.1.3. createDrawTwoCard() [2/2]

```
static ActionCard domain.testhelper.CardTestFactory.createDrawTwoCard (  
    CardColor color) [static]
```

7.12.1.4. createNumberCard() [1/2]

```
static NumberCard domain.testhelper.CardTestFactory.createNumberCard () [static]
```

Gráfico de llamadas a esta función:

7.12.1.5. createNumberCard() [2/2]

```
static NumberCard domain.testhelper.CardTestFactory.createNumberCard (  
    int value,  
    CardColor color) [static]
```

7.12.1.6. createReverseCard() [1/2]

```
static ActionCard domain.testhelper.CardTestFactory.createReverseCard () [static]
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.12.1.7. createReverseCard() [2/2]

```
static ActionCard domain.testhelper.CardTestFactory.createReverseCard (  
    CardColor color) [static]
```

7.12.1.8. createSkipCard() [1/2]

```
static ActionCard domain.testhelper.CardTestFactory.createSkipCard () [static]
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.12.1.9. createSkipCard() [2/2]

```
static ActionCard domain.testhelper.CardTestFactory.createSkipCard (  
    CardColor color) [static]
```

7.12.1.10. createWildCard()

```
static WildCard domain.testhelper.CardTestFactory.createWildCard (  
    CardType type) [static]
```

7.12.1.11. createWildColorCard() [1/2]

```
static WildCard domain.testhelper.CardTestFactory.createWildColorCard () [static]
```

7.12.1.12. createWildColorCard() [2/2]

```
static WildCard domain.testhelper.CardTestFactory.createWildColorCard (  
    CardColor color) [static]
```

7.12.1.13. createWildDrawFourCard() [1/2]

```
static WildCard domain.testhelper.CardTestFactory.createWildDrawFourCard () [static]
```

7.12.1.14. createWildDrawFourCard() [2/2]

```
static WildCard domain.testhelper.CardTestFactory.createWildDrawFourCard (  
    CardColor color) [static]
```

La documentación de esta clase está generada del siguiente archivo:

- src/test/java/domain/testhelper/ **CardTestFactory.java**

7.13. Referencia de la enumeración domain.card.CardType**Atributos públicos**

- **NUMBER**
- **SKIP**
- **REVERSE**
- **DRAW_TWO**
- **WILD_COLOR**
- **WILD_DRAW_FOUR**

7.13.1. Documentación de datos miembro

7.13.1.1. DRAW_TWO

`domain.card.CardType.DRAW_TWO`

7.13.1.2. NUMBER

`domain.card.CardType.NUMBER`

7.13.1.3. REVERSE

`domain.card.CardType.REVERSE`

7.13.1.4. SKIP

`domain.card.CardType.SKIP`

7.13.1.5. WILD_COLOR

`domain.card.CardType.WILD_COLOR`

7.13.1.6. WILD_DRAW_FOUR

`domain.card.CardType.WILD_DRAW_FOUR`

La documentación de esta enumeración está generada del siguiente archivo:

- `src/main/java/domain/card/ CardType.java`

7.14. Referencia de la clase `domain.card.CardUtil`

Métodos públicos estáticos

- static void **validateColor** (**CardColor** color)
- static void **validateNumber** (int number)
- static void **validateActionType** (**CardType** type)
- static boolean **isWildCard** (**Card** card)

7.14.1. Documentación de funciones miembro

7.14.1.1. isWildCard()

```
static boolean domain.card.CardUtil.isWildCard (  
    Card card) [static]
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.14.1.2. validateActionType()

```
static void domain.card.CardUtil.validateActionType (  
    CardType type) [static]
```

Gráfico de llamadas a esta función:

7.14.1.3. validateColor()

```
static void domain.card.CardUtil.validateColor (  
    CardColor color) [static]
```

Gráfico de llamadas a esta función:

7.14.1.4. validateNumber()

```
static void domain.card.CardUtil.validateNumber (  
    int number) [static]
```

Gráfico de llamadas a esta función:

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/card/ **CardUtil.java**

7.15. Referencia de la clase ui.view.CardView

Diagrama de herencia de ui.view.CardView

Diagrama de colaboración de ui.view.CardView:

Métodos públicos

- **CardView** (**Card** card)
- **CardView** (**Card** card, Consumer< **Card** > onCardClick)
- Dimension **getDimension** ()

Métodos protegidos

- void **paintComponent** (Graphics g)

7.15.1. Documentación de constructores y destructores

7.15.1.1. **CardView()** [1/2]

```
ui.view.CardView.CardView (  
    Card card)
```

7.15.1.2. **CardView()** [2/2]

```
ui.view.CardView.CardView (  
    Card card,  
    Consumer< Card > onCardClick)
```

Gráfico de llamadas de esta función:

7.15.2. Documentación de funciones miembro

7.15.2.1. **getDimension()**

```
Dimension ui.view.CardView.getDimension ()
```

7.15.2.2. **paintComponent()**

```
void ui.view.CardView.paintComponent (  
    Graphics g) [protected]
```

Gráfico de llamadas de esta función:

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/ui/view/ **CardView.java**

7.16. Referencia de la clase ui.view.ColorPicker

Métodos públicos

- **CardColor** **show** ()

Métodos públicos estáticos

- static **ColorPicker** **getInstance** ()

7.16.1. Documentación de funciones miembro

7.16.1.1. getInstance()

```
static ColorPicker ui.view.ColorPicker.getInstance () [static]
```

7.16.1.2. show()

```
CardColor ui.view.ColorPicker.show ()
```

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/ui/view/ **ColorPicker.java**

7.17. Referencia de la clase domain.game.DealerService

Métodos públicos estáticos

- static List< **Card** > **shuffle** (List< **Card** > cards)
- static **DrawPile** **shuffle** (**DrawPile** drawPile, **Card** lastPlayedCard)
- static **HandCardList**[] **dealInitialHandCards** (**DrawPile** drawPile, int totalPlayers)

Atributos públicos estáticos

- static final int **TOTAL_INITIAL_HAND_CARDS** = 7

7.17.1. Documentación de funciones miembro

7.17.1.1. dealInitialHandCards()

```
static HandCardList[] domain.game.DealerService.dealInitialHandCards (
    DrawPile drawPile,
    int totalPlayers) [static]
```

Gráfico de llamadas de esta función:

7.17.1.2. shuffle() [1/2]

```
static DrawPile domain.game.DealerService.shuffle (
    DrawPile drawPile,
    Card lastPlayedCard) [static]
```

Gráfico de llamadas de esta función:

7.17.1.3. shuffle() [2/2]

```
static List< Card > domain.game.DealerService.shuffle (
    List< Card > cards) [static]
```

Cards are shuffled using the modern version of Fisher-Yates shuffle. Refer https://en.wikipedia.org/wiki/Fisher-Yates_shuffle#The_modern_algorithm

Parámetros

<i>cards</i>	list will not be modified
--------------	---------------------------

Devuelve

a new shuffled card list

Gráfico de llamadas a esta función:

7.17.2. Documentación de datos miembro**7.17.2.1. TOTAL_INITIAL_HAND_CARDS**

```
final int domain.game.DealerService.TOTAL_INITIAL_HAND_CARDS = 7 [static]
```

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/game/ **DealerService.java**

7.18. Referencia de la enumeración domain.player.Direction**Atributos públicos**

- **CLOCKWISE**
- **COUNTER_CLOCK_WISE**

7.18.1. Documentación de datos miembro**7.18.1.1. CLOCKWISE**

```
domain.player.Direction.CLOCKWISE
```

7.18.1.2. COUNTER_CLOCK_WISE

```
domain.player.Direction.COUNTER_CLOCK_WISE
```

La documentación de esta enumeración está generada del siguiente archivo:

- src/main/java/domain/player/ **Direction.java**

7.19. Referencia de la clase domain.common.DomainEvent

Diagrama de herencia de domain.common.DomainEvent

Métodos públicos

- LocalDate **getOccurredDate** ()

Métodos protegidos

- **DomainEvent** ()

7.19.1. Documentación de constructores y destructores**7.19.1.1. DomainEvent()**

```
domain.common.DomainEvent.DomainEvent () [protected]
```

7.19.2. Documentación de funciones miembro**7.19.2.1. getOccurredDate()**

```
LocalDate domain.common.DomainEvent.getOccurredDate ()
```

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/common/ **DomainEvent.java**

7.20. Referencia de la clase domain.common.DomainEventPublisher**Métodos públicos estáticos**

- static void **subscribe** (**DomainEventSubscriber** subscriber)
- static void **unsubscribe** (**DomainEventSubscriber** subscriber)
- static void **publish** (final **DomainEvent** event)
- static void **reset** ()

7.20.1. Documentación de funciones miembro**7.20.1.1. publish()**

```
static void domain.common.DomainEventPublisher.publish (
    final DomainEvent event) [static]
```

Gráfico de llamadas a esta función:

7.20.1.2. reset()

```
static void domain.common.DomainEventPublisher.reset () [static]
```

7.20.1.3. subscribe()

```
static void domain.common.DomainEventPublisher.subscribe (  
    DomainEventSubscriber subscriber) [static]
```

Gráfico de llamadas a esta función:

7.20.1.4. unsubscribe()

```
static void domain.common.DomainEventPublisher.unsubscribe (  
    DomainEventSubscriber subscriber) [static]
```

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/common/ **DomainEventPublisher.java**

7.21. Referencia de la interface domain.common.DomainEventSubscriber

Diagrama de herencia de domain.common.DomainEventSubscriber

Métodos públicos

- void **handleEvent** (**DomainEvent** event)

7.21.1. Documentación de funciones miembro

7.21.1.1. handleEvent()

```
void domain.common.DomainEventSubscriber.handleEvent (  
    DomainEvent event)
```

Implementado en **ui.view.GameStatusView** (pág. 42), **ui.view.PlayerView** (pág. 53) y **ui.view.TableView** (pág. 55).

La documentación de esta interface está generada del siguiente archivo:

- src/main/java/domain/common/ **DomainEventSubscriber.java**

7.22. Referencia de la clase domain.game.DrawPile

Métodos públicos

- **DrawPile** (List< **Card** > shuffledCards)
- **Card** **drawCard** ()
- int **getSize** ()

7.22.1. Documentación de constructores y destructores

7.22.1.1. DrawPile()

```
domain.game.DrawPile.DrawPile (
    List< Card > shuffledCards)
```

7.22.2. Documentación de funciones miembro

7.22.2.1. drawCard()

```
Card domain.game.DrawPile.drawCard ()
```

Gráfico de llamadas a esta función:

7.22.2.2. getSize()

```
int domain.game.DrawPile.getSize ()
```

Gráfico de llamadas a esta función:

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/game/ **DrawPile.java**

7.23. Referencia de la clase domain.common.Entity

Diagrama de herencia de domain.common.Entity

Métodos públicos

- UUID **getId** ()

Métodos protegidos

- **Entity** ()
- **Entity** (UUID id)

7.23.1. Documentación de constructores y destructores

7.23.1.1. Entity() [1/2]

```
domain.common.Entity.Entity () [protected]
```

7.23.1.2. Entity() [2/2]

```
domain.common.Entity.Entity (
    UUID id) [protected]
```

7.23.2. Documentación de funciones miembro

7.23.2.1. getId()

```
UUID domain.common.Entity.getId ()
```

Gráfico de llamadas a esta función:

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/common/ **Entity.java**

7.24. Referencia de la clase domain.game.Game

Diagrama de herencia de domain.game.Game

Diagrama de colaboración de domain.game.Game:

Métodos públicos

- **Game** (**DrawPile** drawPile, **PlayerRoundIterator** players)
- Stream< **ImmutablePlayer** > **getPlayers** ()
- **ImmutablePlayer** **getCurrentPlayer** ()
- Stream< **Card** > **getHandCards** (UUID playerId)
- **Card** **peekTopCard** ()
- void **playCard** (UUID playerId, **Card** playedCard)
- void **playCard** (UUID playerId, **Card** playedCard, boolean hasSaidUno)
- void **drawCard** (UUID playerId)
- boolean **isOver** ()
- **ImmutablePlayer** **getWinner** ()

Métodos públicos heredados de domain.common.Entity

- UUID **getId** ()

Otros miembros heredados

Métodos protegidos heredados de domain.common.Entity

- **Entity** ()
- **Entity** (UUID id)

7.24.1. Documentación de constructores y destructores

7.24.1.1. Game()

```
domain.game.Game.Game (
    DrawPile drawPile,
    PlayerRoundIterator players)
```

7.24.2. Documentación de funciones miembro

7.24.2.1. drawCard()

```
void domain.game.Game.drawCard (
    UUID playerId)
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.24.2.2. getCurrentPlayer()

```
ImmutablePlayer domain.game.Game.getCurrentPlayer ()
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.24.2.3. getHandCards()

```
Stream< Card > domain.game.Game.getHandCards (
    UUID playerId)
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.24.2.4. getPlayers()

```
Stream< ImmutablePlayer > domain.game.Game.getPlayers ()
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.24.2.5. getWinner()

```
ImmutablePlayer domain.game.Game.getWinner ()
```

Gráfico de llamadas a esta función:

7.24.2.6. isOver()

```
boolean domain.game.Game.isOver ()
```

Gráfico de llamadas a esta función:

7.24.2.7. peekTopCard()

```
Card domain.game.Game.peekTopCard ()
```

Gráfico de llamadas a esta función:

7.24.2.8. playCard() [1/2]

```
void domain.game.Game.playCard (
    UUID playerId,
    Card playedCard)
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.24.2.9. playCard() [2/2]

```
void domain.game.Game.playCard (
    UUID playerId,
    Card playedCard,
    boolean hasSaidUno)
```

Gráfico de llamadas de esta función:

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/game/ **Game.java**

7.25. Referencia de la clase application.GameAppService

Diagrama de herencia de application.GameAppService

Diagrama de colaboración de application.GameAppService:

Métodos públicos

- **GameAppService** ()
- List< **PlayerInfoDTO** > **getPlayerInfos** ()
- **PlayerInfoDTO** **getCurrentPlayer** ()
- Stream< **Card** > **getHandCards** (UUID playerId)
- void **playCard** (UUID playerId, **Card** card, boolean hasSaidUno)
- void **drawCard** (UUID playerId)
- **Card** **peekTopCard** ()
- boolean **isGameOver** ()
- **ImmutablePlayer** **getWinner** ()

Métodos públicos heredados de `application.IGameAppService`

7.25.1. Documentación de constructores y destructores

7.25.1.1. `GameAppService()`

```
application.GameAppService.GameAppService ()
```

Gráfico de llamadas de esta función:

7.25.2. Documentación de funciones miembro

7.25.2.1. `drawCard()`

```
void application.GameAppService.drawCard (  
    UUID playerId)
```

Implementa `application.IGameAppService` (pág. 44).

Gráfico de llamadas de esta función:

7.25.2.2. `getCurrentPlayer()`

```
PlayerInfoDTO application.GameAppService.getCurrentPlayer ()
```

Implementa `application.IGameAppService` (pág. 44).

Gráfico de llamadas de esta función:

7.25.2.3. `getHandCards()`

```
Stream< Card > application.GameAppService.getHandCards (  
    UUID playerId)
```

Implementa `application.IGameAppService` (pág. 45).

Gráfico de llamadas de esta función:

7.25.2.4. `getPlayerInfos()`

```
List< PlayerInfoDTO > application.GameAppService.getPlayerInfos ()
```

Implementa `application.IGameAppService` (pág. 45).

Gráfico de llamadas de esta función:

7.25.2.5. `getWinner()`

```
ImmutablePlayer application.GameAppService.getWinner ()
```

Implementa **application.IGameAppService** (pág. 45).

Gráfico de llamadas de esta función:

7.25.2.6. `isGameOver()`

```
boolean application.GameAppService.isGameOver ()
```

Implementa **application.IGameAppService** (pág. 45).

Gráfico de llamadas de esta función:

7.25.2.7. `peekTopCard()`

```
Card application.GameAppService.peekTopCard ()
```

Implementa **application.IGameAppService** (pág. 45).

Gráfico de llamadas de esta función:

7.25.2.8. `playCard()`

```
void application.GameAppService.playCard (  
    UUID playerId,  
    Card card,  
    boolean hasSaidUno)
```

Implementa **application.IGameAppService** (pág. 45).

Gráfico de llamadas de esta función:

La documentación de esta clase está generada del siguiente archivo:

- `src/main/java/application/ GameAppService.java`

7.26. Referencia de la clase `domain.game.GameBuilder`

Métodos públicos

- **GameBuilder** `withPlayer` (String name)
- **Game** `build` ()

7.26.1. Documentación de funciones miembro

7.26.1.1. build()

```
Game domain.game.GameBuilder.build ()
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.26.1.2. withPlayer()

```
GameBuilder domain.game.GameBuilder.withPlayer (  
    String name)
```

Gráfico de llamadas a esta función:

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/game/ **GameBuilder.java**

7.27. Referencia de la clase domain.game.events.GameOver

Diagrama de herencia de domain.game.events.GameOver

Diagrama de colaboración de domain.game.events.GameOver:

Métodos públicos

- **GameOver** (**ImmutablePlayer** winner)
- **ImmutablePlayer** **getWinner** ()

Métodos públicos heredados de domain.common.DomainEvent

- **LocalDate** **getOccurredDate** ()

Otros miembros heredados

Métodos protegidos heredados de domain.common.DomainEvent

- **DomainEvent** ()

7.27.1. Documentación de constructores y destructores

7.27.1.1. GameOver()

```
domain.game.events.GameOver.GameOver (  
    ImmutablePlayer winner)
```

7.27.2. Documentación de funciones miembro

7.27.2.1. `getWinner()`

```
ImmutablePlayer domain.game.events.GameOver.getWinner ()
```

La documentación de esta clase está generada del siguiente archivo:

- `src/main/java/domain/game/events/ GameOver.java`

7.28. Referencia de la clase `ui.view.GameStatusView`

Diagrama de herencia de `ui.view.GameStatusView`

Diagrama de colaboración de `ui.view.GameStatusView`:

Métodos públicos

- `GameStatusView (IGameAppService appService)`
- `void setError (String errorMgs)`
- `void handleEvent (DomainEvent event)`

Métodos públicos heredados de `domain.common.DomainEventSubscriber`

Métodos protegidos

- `void paintComponent (Graphics g)`

7.28.1. Documentación de constructores y destructores

7.28.1.1. `GameStatusView()`

```
ui.view.GameStatusView.GameStatusView (
    IGameAppService appService)
```

Gráfico de llamadas de esta función:

7.28.2. Documentación de funciones miembro

7.28.2.1. `handleEvent()`

```
void ui.view.GameStatusView.handleEvent (
    DomainEvent event)
```

Implementa `domain.common.DomainEventSubscriber` (pág. 34).

7.28.2.2. paintComponent()

```
void ui.view.GameStatusView.paintComponent (
    Graphics g) [protected]
```

7.28.2.3. setError()

```
void ui.view.GameStatusView.setError (
    String errorMsg)
```

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/ui/view/ **GameStatusView.java**

7.29. Referencia de la clase domain.player.HandCardList

Métodos públicos

- void **addCard** (**Card** newCard)
- boolean **removeCard** (**Card** card)
- Stream< **Card** > **getCardStream** ()
- boolean **hasCard** (**Card** card)
- int **size** ()

7.29.1. Documentación de funciones miembro

7.29.1.1. addCard()

```
void domain.player.HandCardList.addCard (
    Card newCard)
```

Gráfico de llamadas a esta función:

7.29.1.2. getCardStream()

```
Stream< Card > domain.player.HandCardList.getCardStream ()
```

Gráfico de llamadas a esta función:

7.29.1.3. hasCard()

```
boolean domain.player.HandCardList.hasCard (
    Card card)
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.29.1.4. removeCard()

```
boolean domain.player.HandCardList.removeCard (
    Card card)
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.29.1.5. size()

```
int domain.player.HandCardList.size ()
```

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/player/ **HandCardList.java**

7.30. Referencia de la interface application.IGameAppService

Diagrama de herencia de application.IGameAppService

Diagrama de colaboración de application.IGameAppService:

Métodos públicos

- List< **PlayerInfoDTO** > **getPlayerInfos** ()
- **PlayerInfoDTO** **getCurrentPlayer** ()
- Stream< **Card** > **getHandCards** (UUID playerId)
- void **playCard** (UUID playerId, **Card** card, boolean hasSaidUno)
- void **drawCard** (UUID playerId)
- **Card** **peekTopCard** ()
- boolean **isGameOver** ()
- **ImmutablePlayer** **getWinner** ()

7.30.1. Documentación de funciones miembro

7.30.1.1. drawCard()

```
void application.IGameAppService.drawCard (
    UUID playerId)
```

Implementado en **application.GameAppService** (pág. 39).

7.30.1.2. getCurrentPlayer()

```
PlayerInfoDTO application.IGameAppService.getCurrentPlayer ()
```

Implementado en **application.GameAppService** (pág. 39).

7.30.1.3. `getHandCards()`

```
Stream< Card > application.IGameAppService.getHandCards (
    UUID playerId)
```

Implementado en `application.GameAppService` (pág. 39).

7.30.1.4. `getPlayerInfos()`

```
List< PlayerInfoDTO > application.IGameAppService.getPlayerInfos ()
```

Implementado en `application.GameAppService` (pág. 39).

7.30.1.5. `getWinner()`

```
ImmutablePlayer application.IGameAppService.getWinner ()
```

Implementado en `application.GameAppService` (pág. 40).

7.30.1.6. `isGameOver()`

```
boolean application.IGameAppService.isGameOver ()
```

Implementado en `application.GameAppService` (pág. 40).

7.30.1.7. `peekTopCard()`

```
Card application.IGameAppService.peekTopCard ()
```

Implementado en `application.GameAppService` (pág. 40).

7.30.1.8. `playCard()`

```
void application.IGameAppService.playCard (
    UUID playerId,
    Card card,
    boolean hasSaidUno)
```

Implementado en `application.GameAppService` (pág. 40).

La documentación de esta interface está generada del siguiente archivo:

- `src/main/java/application/IGameAppService.java`

7.31. Referencia de la clase domain.player.ImmutablePlayer

Diagrama de herencia de domain.player.ImmutablePlayer

Diagrama de colaboración de domain.player.ImmutablePlayer:

Métodos públicos

- **ImmutablePlayer** (**Player** player)
- UUID **getId** ()
- String **getName** ()
- Stream< **Card** > **getHandCards** ()
- int **getTotalCards** ()

7.31.1. Documentación de constructores y destructores

7.31.1.1. ImmutablePlayer()

```
domain.player.ImmutablePlayer.ImmutablePlayer (
    Player player)
```

7.31.2. Documentación de funciones miembro

7.31.2.1. getHandCards()

```
Stream< Card > domain.player.ImmutablePlayer.getHandCards ()
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.31.2.2. getId()

```
UUID domain.player.ImmutablePlayer.getId ()
```

Gráfico de llamadas de esta función:

7.31.2.3. getName()

```
String domain.player.ImmutablePlayer.getName ()
```

Gráfico de llamadas de esta función:

7.31.2.4. getTotalCards()

```
int domain.player.ImmutablePlayer.getTotalCards ()
```

Gráfico de llamadas de esta función:

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/player/ **ImmutablePlayer.java**

7.32. Referencia de la clase domain.card.NumberCard

Diagrama de herencia de domain.card.NumberCard

Diagrama de colaboración de domain.card.NumberCard:

Métodos públicos

- **NumberCard** (int value, **CardColor** color)
- int **getValue** ()
- boolean **equals** (Object o)
- int **hashCode** ()
- String **toString** ()

Métodos públicos heredados de domain.card.AbstractCard

- **CardType** **getType** ()
- **CardColor** **getColor** ()

Métodos públicos heredados de domain.card.Card

Otros miembros heredados

Métodos protegidos heredados de domain.card.AbstractCard

- **AbstractCard** (**CardType** type, **CardColor** color)

7.32.1. Documentación de constructores y destructores

7.32.1.1. NumberCard()

```
domain.card.NumberCard.NumberCard (
    int value,
    CardColor color)
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.32.2. Documentación de funciones miembro

7.32.2.1. equals()

```
boolean domain.card.NumberCard.equals (  
    Object o)
```

Reimplementado de **domain.card.AbstractCard** (pág. 16).

Gráfico de llamadas de esta función:

7.32.2.2. getValue()

```
int domain.card.NumberCard.getValue ()
```

Gráfico de llamadas a esta función:

7.32.2.3. hashCode()

```
int domain.card.NumberCard.hashCode ()
```

Reimplementado de **domain.card.AbstractCard** (pág. 16).

Gráfico de llamadas de esta función:

7.32.2.4. toString()

```
String domain.card.NumberCard.toString ()
```

Reimplementado de **domain.card.AbstractCard** (pág. 16).

Gráfico de llamadas de esta función:

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/card/ **NumberCard.java**

7.33. Referencia de la clase domain.player.Player

Diagrama de herencia de domain.player.Player

Diagrama de colaboración de domain.player.Player:

Métodos públicos

- **Player** (String name, **HandCardList** handCards)
- String **getName** ()
- Stream< **Card** > **getHandCards** ()
- void **addToHandCards** (**Card** card)
- boolean **removePlayedCard** (**Card** card)
- boolean **hasHandCard** (**Card** card)
- **ImmutablePlayer** **toImmutable** ()

Métodos públicos heredados de domain.common.Entity

- UUID **getId** ()

Otros miembros heredados

Métodos protegidos heredados de domain.common.Entity

- **Entity** ()
- **Entity** (UUID id)

7.33.1. Documentación de constructores y destructores

7.33.1.1. Player()

```
domain.player.Player.Player (
    String name,
    HandCardList handCards)
```

7.33.2. Documentación de funciones miembro

7.33.2.1. addToHandCards()

```
void domain.player.Player.addToHandCards (
    Card card)
```

Gráfico de llamadas de esta función:

7.33.2.2. getHandCards()

```
Stream< Card > domain.player.Player.getHandCards ()
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.33.2.3. getName()

```
String domain.player.Player.getName ()
```

Gráfico de llamadas a esta función:

7.33.2.4. hasHandCard()

```
boolean domain.player.Player.hasHandCard (
    Card card)
```

Gráfico de llamadas de esta función:

7.33.2.5. removePlayedCard()

```
boolean domain.player.Player.removePlayedCard (
    Card card)
```

Gráfico de llamadas de esta función:

7.33.2.6. toImmutable()

```
ImmutablePlayer domain.player.Player.toImmutable ()
```

Gráfico de llamadas a esta función:

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/player/ **Player.java**

7.34. Referencia de la clase application.dto.PlayerInfoDTO

Diagrama de herencia de application.dto.PlayerInfoDTO

Diagrama de colaboración de application.dto.PlayerInfoDTO:

Métodos públicos

- **PlayerInfoDTO** (UUID id, String name)
- UUID **getId** ()
- String **getName** ()

7.34.1. Documentación de constructores y destructores

7.34.1.1. PlayerInfoDTO()

```
application.dto.PlayerInfoDTO.PlayerInfoDTO (
    UUID id,
    String name)
```

7.34.2. Documentación de funciones miembro

7.34.2.1. getId()

```
UUID application.dto.PlayerInfoDTO.getId ()
```

7.34.2.2. getName()

```
String application.dto.PlayerInfoDTO.getName ()
```

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/application/dto/ **PlayerInfoDTO.java**

7.35. Referencia de la clase domain.player.PlayerRoundlterator

Métodos públicos

- **PlayerRoundlterator** (**Player**[] players)
- **Stream**< **Player** > **stream** ()
- **Player** **getCurrentPlayer** ()
- **Player** **getPlayerByld** (UUID playerId)
- void **reverseDirection** ()
- **Player** **next** ()

7.35.1. Documentación de constructores y destructores

7.35.1.1. PlayerRoundlterator()

```
domain.player.PlayerRoundIterator.PlayerRoundIterator (  
    Player[] players)
```

7.35.2. Documentación de funciones miembro

7.35.2.1. getCurrentPlayer()

```
Player domain.player.PlayerRoundIterator.getCurrentPlayer ()
```

Gráfico de llamadas a esta función:

7.35.2.2. getPlayerByld()

```
Player domain.player.PlayerRoundIterator.getPlayerById (  
    UUID playerId)
```

Gráfico de llamadas a esta función:

7.35.2.3. next()

```
Player domain.player.PlayerRoundIterator.next ()
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.35.2.4. reverseDirection()

```
void domain.player.PlayerRoundIterator.reverseDirection ()
```

7.35.2.5. stream()

```
Stream< Player > domain.player.PlayerRoundIterator.stream ()
```

Gráfico de llamadas a esta función:

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/domain/player/ **PlayerRoundIterator.java**

7.36. Referencia de la clase domain.testhelper.PlayerTestFactory

Métodos públicos estáticos

- static **Player**[] createPlayers (int total)

7.36.1. Documentación de funciones miembro

7.36.1.1. createPlayers()

```
static Player[] domain.testhelper.PlayerTestFactory.createPlayers (
    int total) [static]
```

La documentación de esta clase está generada del siguiente archivo:

- src/test/java/domain/testhelper/ **PlayerTestFactory.java**

7.37. Referencia de la clase ui.view.PlayerView

Diagrama de herencia de ui.view.PlayerView

Diagrama de colaboración de ui.view.PlayerView:

Métodos públicos

- **PlayerView** (**PlayerInfoDTO** player, **IGameAppService** appService)
- void **handleEvent** (**DomainEvent** event)

Métodos públicos heredados de domain.common.DomainEventSubscriber

7.37.1. Documentación de constructores y destructores

7.37.1.1. PlayerView()

```
ui.view.PlayerView.PlayerView (
    PlayerInfoDTO player,
    IGameAppService appService)
```

Gráfico de llamadas de esta función:

7.37.2. Documentación de funciones miembro

7.37.2.1. handleEvent()

```
void ui.view.PlayerView.handleEvent (
    DomainEvent event)
```

Implementa **domain.common.DomainEventSubscriber** (pág. 34).

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/ui/view/ **PlayerView.java**

7.38. Referencia de la clase ui.common.StyleUtil

Métodos públicos estáticos

- static Color **convertCardColor** (**CardColor** color)
- static String **getValueToDisplay** (**Card** card)

Atributos públicos estáticos

- static final Color **redColor** = new Color(192, 80, 77)
- static final Color **blueColor** = new Color(31, 73, 125)
- static final Color **greenColor** = new Color(0, 153, 0)
- static final Color **yellowColor** = new Color(255, 204, 0)
- static final Color **blackColor** = new Color(0, 0, 0)
- static final String **DEFAULT_FONT** = "Helvetica"

7.38.1. Documentación de funciones miembro

7.38.1.1. `convertCardColor()`

```
static Color ui.common.StyleUtil.convertCardColor (  
    CardColor color) [static]
```

Gráfico de llamadas a esta función:

7.38.1.2. `getValueToDisplay()`

```
static String ui.common.StyleUtil.getValueToDisplay (  
    Card card) [static]
```

Gráfico de llamadas de esta función: Gráfico de llamadas a esta función:

7.38.2. Documentación de datos miembro

7.38.2.1. `blackColor`

```
final Color ui.common.StyleUtil.blackColor = new Color(0, 0, 0) [static]
```

7.38.2.2. `blueColor`

```
final Color ui.common.StyleUtil.blueColor = new Color(31, 73, 125) [static]
```

7.38.2.3. `DEFAULT_FONT`

```
final String ui.common.StyleUtil.DEFAULT_FONT = "Helvetica" [static]
```

7.38.2.4. `greenColor`

```
final Color ui.common.StyleUtil.greenColor = new Color(0, 153, 0) [static]
```

7.38.2.5. `redColor`

```
final Color ui.common.StyleUtil.redColor = new Color(192, 80, 77) [static]
```

7.38.2.6. `yellowColor`

```
final Color ui.common.StyleUtil.yellowColor = new Color(255, 204, 0) [static]
```

La documentación de esta clase está generada del siguiente archivo:

- `src/main/java/ui/common/ StyleUtil.java`

7.39. Referencia de la clase ui.view.TableView

Diagrama de herencia de ui.view.TableView

Diagrama de colaboración de ui.view.TableView:

Métodos públicos

- **TableView** (**IGameAppService** appService)
- void **handleEvent** (**DomainEvent** event)

Métodos públicos heredados de domain.common.DomainEventSubscriber

7.39.1. Documentación de constructores y destructores

7.39.1.1. TableView()

```
ui.view.TableView.TableView (
    IGameAppService appService)
```

Gráfico de llamadas de esta función:

7.39.2. Documentación de funciones miembro

7.39.2.1. handleEvent()

```
void ui.view.TableView.handleEvent (
    DomainEvent event)
```

Implementa **domain.common.DomainEventSubscriber** (pág. 34).

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/ui/view/ **TableView.java**

7.40. Referencia de la clase domain.TestHandCardList

Métodos públicos

- void **GivenSameNumberCard_ShouldExist** ()
- void **GivenDifferentNumberCard_ShouldNotExist** ()

7.40.1. Documentación de funciones miembro

7.40.1.1. GivenDifferentNumberCard_ShouldNotExist()

```
void domain.TestHandCardList.GivenDifferentNumberCard_ShouldNotExist ()
```

Gráfico de llamadas de esta función:

7.40.1.2. GivenSameNumberCard_ShouldExist()

```
void domain.TestHandCardList.GivenSameNumberCard_ShouldExist ()
```

Gráfico de llamadas de esta función:

La documentación de esta clase está generada del siguiente archivo:

- src/test/java/domain/ **TestHandCardList.java**

7.41. Referencia de la clase UnoApp

Métodos públicos estáticos

- static void **main** (String[] args)

7.41.1. Documentación de funciones miembro

7.41.1.1. main()

```
static void UnoApp.main (
    String[] args) [static]
```

La documentación de esta clase está generada del siguiente archivo:

- src/main/java/ **UnoApp.java**

7.42. Referencia de la clase domain.card.WildCard

Diagrama de herencia de domain.card.WildCard

Diagrama de colaboración de domain.card.WildCard:

Métodos públicos

- **WildCard** (**CardType** type)
- **WildCard** (**CardType** type, **CardColor** color)
- boolean **equals** (Object o)
- int **hashCode** ()
- String **toString** ()

Métodos públicos heredados de domain.card.AbstractCard

- `CardType getType ()`
- `CardColor getColor ()`

Métodos públicos heredados de domain.card.Card

Otros miembros heredados

Métodos protegidos heredados de domain.card.AbstractCard

- `AbstractCard (CardType type, CardColor color)`

7.42.1. Documentación de constructores y destructores

7.42.1.1. WildCard() [1/2]

```
domain.card.WildCard.WildCard (
    CardType type)
```

Gráfico de llamadas a esta función:

7.42.1.2. WildCard() [2/2]

```
domain.card.WildCard.WildCard (
    CardType type,
    CardColor color)
```

Gráfico de llamadas de esta función:

7.42.2. Documentación de funciones miembro

7.42.2.1. equals()

```
boolean domain.card.WildCard.equals (
    Object o)
```

Reimplementado de **domain.card.AbstractCard** (pág. 16).

Gráfico de llamadas de esta función:

7.42.2.2. hashCode()

```
int domain.card.WildCard.hashCode ()
```

Reimplementado de **domain.card.AbstractCard** (pág. 16).

Gráfico de llamadas de esta función:

7.42.2.3. toString()

```
String domain.card.WildCard.toString ()
```

Reimplementado de **domain.card.AbstractCard** (pág. 16).

Gráfico de llamadas de esta función:

La documentación de esta clase está generada del siguiente archivo:

- `src/main/java/domain/card/ WildCard.java`

Capítulo 8

Documentación de archivos

8.1. Referencia del archivo project docs/README.md

8.2. Referencia del archivo

src/main/java/application/dto/PlayerInfoDTO.java

```
import java.io.Serializable;
```

Gráfico de dependencias incluidas en PlayerInfoDTO.java:

Clases

- class **application.dto.PlayerInfoDTO**

Paquetes

- package **application.dto**

8.3. Referencia del archivo

src/main/java/application/GameAppService.java

```
import application.dto.PlayerInfoDTO;
```

Gráfico de dependencias incluidas en GameAppService.java:

Clases

- class **application.GameAppService**

Paquetes

- package **application**

8.4. Referencia del archivo `src/main/java/application/IGameAppService.java`

```
import application.dto.PlayerInfoDTO;
```

Gráfico de dependencias incluidas en IGameAppService.java:

Clases

- interface `application.IGameAppService`

Paquetes

- package `application`

8.5. Referencia del archivo `src/main/java/domain/card/AbstractCard.java`

Clases

- class `domain.card.AbstractCard`

Paquetes

- package `domain.card`

8.6. Referencia del archivo `src/main/java/domain/card/ActionCard.java`

```
import java.util.Objects;
```

Gráfico de dependencias incluidas en ActionCard.java:

8.7. Referencia del archivo `src/main/java/domain/card/Card.java`

```
import java.io.Serializable;
```

Gráfico de dependencias incluidas en Card.java:

Clases

- interface `domain.card.Card`

Paquetes

- package `domain.card`

8.8. Referencia del archivo src/main/java/domain/card/CardColor.java

Clases

- enum **domain.card.CardColor**

Paquetes

- package **domain.card**

8.9. Referencia del archivo src/main/java/domain/card/CardDeck.java

```
import java.util.ArrayList;
```

Gráfico de dependencias incluidas en CardDeck.java:

Clases

- class **domain.card.CardDeck**

Paquetes

- package **domain.card**

8.10. Referencia del archivo src/main/java/domain/card/CardType.java

Clases

- enum **domain.card.CardType**

Paquetes

- package **domain.card**

8.11. Referencia del archivo src/main/java/domain/card/CardUtil.java

Clases

- class **domain.card.CardUtil**

Paquetes

- package **domain.card**

8.12. Referencia del archivo `src/main/java/domain/card/NumberCard.java`

```
import java.util.Objects;
```

Gráfico de dependencias incluidas en `NumberCard.java`:

Clases

- class `domain.card.NumberCard`

Paquetes

- package `domain.card`

8.13. Referencia del archivo `src/main/java/domain/card/WildCard.java`

```
import java.util.Objects;
```

Gráfico de dependencias incluidas en `WildCard.java`:

Clases

- class `domain.card.WildCard`

Paquetes

- package `domain.card`

8.14. Referencia del archivo `src/main/java/domain/common/DomainEvent.java`

```
import java.time.LocalDate;
```

Gráfico de dependencias incluidas en `DomainEvent.java`:

Clases

- class `domain.common.DomainEvent`

Paquetes

- package `domain.common`

8.15. Referencia del archivo src/main/java/domain/common/DomainEventPublisher.java

```
import java.util.ArrayList;
```

Gráfico de dependencias incluidas en DomainEventPublisher.java:

Clases

- class **domain.common.DomainEventPublisher**

Paquetes

- package **domain.common**

8.16. Referencia del archivo src/main/java/domain/common/DomainEventSubscriber.java

Clases

- interface **domain.common.DomainEventSubscriber**

Paquetes

- package **domain.common**

8.17. Referencia del archivo src/main/java/domain/common/Entity.java

```
import java.util.UUID;
```

Gráfico de dependencias incluidas en Entity.java:

Clases

- class **domain.common.Entity**

Paquetes

- package **domain.common**

8.18. Referencia del archivo src/main/java/domain/game/CardRules.java

Clases

- class **domain.game.CardRules**

Paquetes

- package **domain.game**

8.19. Referencia del archivo

src/main/java/domain/game/DealerService.java

```
import domain.card.Card;
```

Gráfico de dependencias incluidas en DealerService.java:

Clases

- class **domain.game.DealerService**

Paquetes

- package **domain.game**

8.20. Referencia del archivo src/main/java/domain/game/DrawPile.java

```
import domain.card.Card;
```

Gráfico de dependencias incluidas en DrawPile.java:

Clases

- class **domain.game.DrawPile**

Paquetes

- package **domain.game**

8.21. Referencia del archivo

src/main/java/domain/game/events/CardDrawn.java

```
import domain.common.DomainEvent;
```

Gráfico de dependencias incluidas en CardDrawn.java:

Clases

- class **domain.game.events.CardDrawn**

Paquetes

- package **domain.game.events**

8.22. Referencia del archivo src/main/java/domain/game/events/CardPlayed.java

```
import domain.card.Card;
```

Gráfico de dependencias incluidas en CardPlayed.java:

Clases

- class **domain.game.events.CardPlayed**

Paquetes

- package **domain.game.events**

8.23. Referencia del archivo src/main/java/domain/game/events/GameOver.java

```
import domain.common.DomainEvent;
```

Gráfico de dependencias incluidas en GameOver.java:

Clases

- class **domain.game.events.GameOver**

Paquetes

- package **domain.game.events**

8.24. Referencia del archivo src/main/java/domain/game/Game.java

```
import domain.common.DomainEventPublisher;
```

Gráfico de dependencias incluidas en Game.java:

Clases

- class **domain.game.Game**

Paquetes

- package **domain.game**

8.25. Referencia del archivo src/main/java/domain/game/GameBuilder.java

```
import domain.card.Card;
```

Gráfico de dependencias incluidas en GameBuilder.java:

Clases

- class **domain.game.GameBuilder**

Paquetes

- package **domain.game**

8.26. Referencia del archivo src/main/java/domain/player/Direction.java

Clases

- enum **domain.player.Direction**

Paquetes

- package **domain.player**

8.27. Referencia del archivo src/main/java/domain/player/HandCardList.java

```
import domain.card.Card;
```

Gráfico de dependencias incluidas en HandCardList.java:

Clases

- class **domain.player.HandCardList**

Paquetes

- package **domain.player**

8.28. Referencia del archivo src/main/java/domain/player/ImmutablePlayer.java

```
import domain.card.Card;
```

Gráfico de dependencias incluidas en ImmutablePlayer.java:

Clases

- class **domain.player.ImmutablePlayer**

Paquetes

- package **domain.player**

8.29. Referencia del archivo src/main/java/domain/player/Player.java

```
import domain.card.Card;
```

Gráfico de dependencias incluidas en Player.java:

Clases

- class **domain.player.Player**

Paquetes

- package **domain.player**

8.30. Referencia del archivo src/main/java/domain/player/PlayerRoundIterator.java

```
import java.util.Arrays;
```

Gráfico de dependencias incluidas en PlayerRoundIterator.java:

Clases

- class **domain.player.PlayerRoundIterator**

Paquetes

- package **domain.player**

8.31. Referencia del archivo src/main/java/ui/AppFrame.java

```
import application.IGameAppService;
```

Gráfico de dependencias incluidas en AppFrame.java:

Clases

- class **ui.AppFrame**

Paquetes

- package **ui**

8.32. Referencia del archivo src/main/java/ui/common/StyleUtil.java

```
import domain.card.Card;
```

Gráfico de dependencias incluidas en StyleUtil.java:

Clases

- class **ui.common.StyleUtil**

Paquetes

- package **ui.common**

8.33. Referencia del archivo src/main/java/ui/view/CardView.java

```
import domain.card.Card;
```

Gráfico de dependencias incluidas en CardView.java:

Clases

- class **ui.view.CardView**

Paquetes

- package **ui.view**

8.34. Referencia del archivo src/main/java/ui/view/ColorPicker.java

```
import domain.card.CardColor;
```

Gráfico de dependencias incluidas en ColorPicker.java:

Clases

- class **ui.view.ColorPicker**

Paquetes

- package **ui.view**

8.35. Referencia del archivo src/main/java/ui/view/GameStatusView.java

```
import application.IGameAppService;
```

Gráfico de dependencias incluidas en GameStatusView.java:

Clases

- class **ui.view.GameStatusView**

Paquetes

- package **ui.view**

8.36. Referencia del archivo src/main/java/ui/view/PlayerView.java

```
import application.IGameAppService;
```

Gráfico de dependencias incluidas en PlayerView.java:

Clases

- class **ui.view.PlayerView**

Paquetes

- package **ui.view**

8.37. Referencia del archivo src/main/java/ui/view/TableView.java

```
import application.IGameAppService;
```

Gráfico de dependencias incluidas en TableView.java:

Clases

- class **ui.view.TableView**

Paquetes

- package **ui.view**

8.38. Referencia del archivo src/main/java/UnoApp.java

```
import application.GameAppService;
```

Gráfico de dependencias incluidas en UnoApp.java:

Clases

- class **UnoApp**

8.39. Referencia del archivo src/test/java/domain/card/CardCounter.java

```
import java.util.List;
```

Gráfico de dependencias incluidas en CardCounter.java:

Clases

- class **domain.card.CardCounter**

Paquetes

- package **domain.card**

8.40. Referencia del archivo src/test/java/domain/card/CardCounterAssertionHelper.java

```
import java.util.Arrays;
```

Gráfico de dependencias incluidas en CardCounterAssertionHelper.java:

Clases

- class **domain.card.CardCounterAssertionHelper**

Paquetes

- package **domain.card**

8.41. Referencia del archivo src/test/java/domain/card/TestCard.java

```
import domain.testhelper.CardTestFactory;
```

Gráfico de dependencias incluidas en TestCard.java:

Clases

- class **domain.card.TestCard**

Paquetes

- package **domain.card**

8.42. Referencia del archivo src/test/java/domain/card/TestCardDeck.java

```
import org.junit.jupiter.api.Assertions;
```

Gráfico de dependencias incluidas en TestCardDeck.java:

Clases

- class **domain.card.TestCardDeck**

Paquetes

- package **domain.card**

8.43. Referencia del archivo src/test/java/domain/card/TestCardUtil.java

```
import domain.testhelper.CardTestFactory;
```

Gráfico de dependencias incluidas en TestCardUtil.java:

Clases

- class **domain.card.TestCardUtil**

Paquetes

- package **domain.card**

8.44. Referencia del archivo src/test/java/domain/common/TestDomainEventPublisher.java

```
import domain.card.Card;
```

Gráfico de dependencias incluidas en TestDomainEventPublisher.java:

Clases

- class **domain.common.TestDomainEventPublisher**
- class **domain.common.TestDomainEventPublisher.TestSubscriber**

Paquetes

- package **domain.common**

8.45. Referencia del archivo **src/test/java/domain/game/TestCardRules.java**

```
import domain.card.ActionCard;
```

Gráfico de dependencias incluidas en TestCardRules.java:

Clases

- class **domain.game.TestCardRules**

Paquetes

- package **domain.game**

8.46. Referencia del archivo **src/test/java/domain/game/TestDealerService.java**

```
import domain.player.HandCardList;
```

Gráfico de dependencias incluidas en TestDealerService.java:

Clases

- class **domain.game.TestDealerService**

Paquetes

- package **domain.game**

8.47. Referencia del archivo **src/test/java/domain/game/TestDrawPile.java**

```
import domain.testhelper.CardTestFactory;
```

Gráfico de dependencias incluidas en TestDrawPile.java:

Clases

- class `domain.game.TestDrawPile`

Paquetes

- package `domain.game`

8.48. Referencia del archivo `src/test/java/domain/game/TestGameBuilder.java`

```
import domain.player.ImmutablePlayer;
```

Gráfico de dependencias incluidas en `TestGameBuilder.java`:

Clases

- class `domain.game.TestGameBuilder`

Paquetes

- package `domain.game`

8.49. Referencia del archivo `src/test/java/domain/game/TestGamePlay.java`

```
import domain.card.Card;
```

Gráfico de dependencias incluidas en `TestGamePlay.java`:

Clases

- class `domain.game.TestGamePlay`

Paquetes

- package `domain.game`

8.50. Referencia del archivo `src/test/java/domain/game/TestGameStart.java`

```
import domain.card.Card;
```

Gráfico de dependencias incluidas en `TestGameStart.java`:

Clases

- class **domain.game.TestGameStart**

Paquetes

- package **domain.game**

8.51. Referencia del archivo **src/test/java/domain/player/TestPlayersliterator.java**

```
import domain.testhelper.PlayerTestFactory;
```

Gráfico de dependencias incluidas en TestPlayersliterator.java:

Clases

- class **domain.player.TestPlayersliterator**

Paquetes

- package **domain.player**

8.52. Referencia del archivo **src/test/java/domain/player/TestHandCardList.java**

```
import domain.card.CardColor;
```

Gráfico de dependencias incluidas en TestHandCardList.java:

Clases

- class **domain.player.TestHandCardList**

Paquetes

- package **domain.player**

8.53. Referencia del archivo **src/test/java/domain/TestHandCardList.java**

```
import domain.card.CardColor;
```

Gráfico de dependencias incluidas en TestHandCardList.java:

Clases

- class `domain.TestHandCardList`

Paquetes

- package `domain`

8.54. Referencia del archivo `src/test/java/domain/testhelper/CardTestFactory.java`

Clases

- class `domain.testhelper.CardTestFactory`

Paquetes

- package `domain.testhelper`

8.55. Referencia del archivo `src/test/java/domain/testhelper/PlayerTestFactory.java`

```
import domain.player.HandCardList;
```

Gráfico de dependencias incluidas en `PlayerTestFactory.java`:

Clases

- class `domain.testhelper.PlayerTestFactory`

Paquetes

- package `domain.testhelper`

Índice alfabético

- AbstractCard
 - domain.card.AbstractCard, 15
- ActionCard
 - domain.card.ActionCard, 17
- addCard
 - domain.player.HandCardList, 43
- addToHandCards
 - domain.player.Player, 49
- AppFrame
 - ui.AppFrame, 18
- application, 11
- application.dto, 11
- application.dto.PlayerInfoDTO, 50
 - getId, 51
 - getName, 51
 - PlayerInfoDTO, 50
- application.GameAppService, 38
 - drawCard, 39
 - GameAppService, 39
 - getCurrentPlayer, 39
 - getHandCards, 39
 - getPlayerInfos, 39
 - getWinner, 39
 - isGameOver, 40
 - peekTopCard, 40
 - playCard, 40
- application.IGameAppService, 44
 - drawCard, 44
 - getCurrentPlayer, 44
 - getHandCards, 44
 - getPlayerInfos, 45
 - getWinner, 45
 - isGameOver, 45
 - peekTopCard, 45
 - playCard, 45
- assertDrawTwoCards
 - domain.card.CardCounterAssertionHelper, 21
- assertNumberCards
 - domain.card.CardCounterAssertionHelper, 21
- assertReverseCards
 - domain.card.CardCounterAssertionHelper, 21
- assertSkipCards
 - domain.card.CardCounterAssertionHelper, 22
- assertWildCards
 - domain.card.CardCounterAssertionHelper, 22
- blackColor
 - ui.common.StyleUtil, 54
- BLUE
 - domain.card.CardColor, 19
- blueColor
 - ui.common.StyleUtil, 54
- build
 - domain.game.GameBuilder, 41
- CardCounter
 - domain.card.CardCounter, 20
- CardDeck
 - domain.card.CardDeck, 22
- CardDrawn
 - domain.game.events.CardDrawn, 23
- CardPlayed
 - domain.game.events.CardPlayed, 24
- CardView
 - ui.view.CardView, 30
- CLOCKWISE
 - domain.player.Direction, 32
- convertCardColor
 - ui.common.StyleUtil, 54
- COUNTER_CLOCK_WISE
 - domain.player.Direction, 32
- createActionCard
 - domain.testhelper.CardTestFactory, 26
- createDrawTwoCard
 - domain.testhelper.CardTestFactory, 26
- createNumberCard
 - domain.testhelper.CardTestFactory, 26
- createPlayers
 - domain.testhelper.PlayerTestFactory, 52
- createReverseCard
 - domain.testhelper.CardTestFactory, 26
- createSkipCard
 - domain.testhelper.CardTestFactory, 26
- createWildCard
 - domain.testhelper.CardTestFactory, 27
- createWildColorCard
 - domain.testhelper.CardTestFactory, 27
- createWildDrawFourCard
 - domain.testhelper.CardTestFactory, 27
- dealInitialHandCards
 - domain.game.DealerService, 31
- DEFAULT_FONT
 - ui.common.StyleUtil, 54
- domain, 11
- domain.card, 12
- domain.card.AbstractCard, 15
 - AbstractCard, 15
 - equals, 16

- getColor, 16
- getType, 16
- hashCode, 16
- toString, 16
- domain.card.ActionCard, 16
 - ActionCard, 17
 - equals, 17
 - hashCode, 17
 - toString, 17
- domain.card.Card, 18
 - getColor, 19
 - getType, 19
- domain.card.CardColor, 19
 - BLUE, 19
 - GREEN, 19
 - RED, 19
 - YELLOW, 19
- domain.card.CardCounter, 20
 - CardCounter, 20
 - getDrawTwoColorCounts, 20
 - getNumberColorCounts, 20
 - getNumberCounts, 20
 - getReverseColorCounts, 20
 - getSkipColorCounts, 21
 - getWildCardCounts, 21
- domain.card.CardCounterAssertionHelper, 21
 - assertDrawTwoCards, 21
 - assertNumberCards, 21
 - assertReverseCards, 21
 - assertSkipCards, 22
 - assertWildCards, 22
- domain.card.CardDeck, 22
 - CardDeck, 22
 - getImmutableCards, 22
- domain.card.CardType, 27
 - DRAW_TWO, 28
 - NUMBER, 28
 - REVERSE, 28
 - SKIP, 28
 - WILD_COLOR, 28
 - WILD_DRAW_FOUR, 28
- domain.card.CardUtil, 28
 - isWildCard, 29
 - validateActionType, 29
 - validateColor, 29
 - validateNumber, 29
- domain.card.NumberCard, 47
 - equals, 48
 - getValue, 48
 - hashCode, 48
 - NumberCard, 47
 - toString, 48
- domain.card.WildCard, 56
 - equals, 57
 - hashCode, 57
 - toString, 57
 - WildCard, 57
- domain.common, 12
 - domain.common.DomainEvent, 32
 - DomainEvent, 33
 - getOccurredDate, 33
 - domain.common.DomainEventPublisher, 33
 - publish, 33
 - reset, 33
 - subscribe, 33
 - unsubscribe, 34
 - domain.common.DomainEventSubscriber, 34
 - handleEvent, 34
 - domain.common.Entity, 35
 - Entity, 35
 - getId, 36
 - domain.game, 12
 - domain.game.CardRules, 24
 - isValidActionCard, 25
 - isValidNumberCard, 25
 - isValidWildCard, 25
 - domain.game.DealerService, 31
 - dealInitialHandCards, 31
 - shuffle, 31
 - TOTAL_INITIAL_HAND_CARDS, 32
 - domain.game.DrawPile, 34
 - drawCard, 35
 - DrawPile, 35
 - getSize, 35
 - domain.game.events, 13
 - domain.game.events.CardDrawn, 23
 - CardDrawn, 23
 - getPlayerId, 23
 - domain.game.events.CardPlayed, 23
 - CardPlayed, 24
 - getPlayedCard, 24
 - getPlayerId, 24
 - domain.game.events.GameOver, 41
 - GameOver, 41
 - getWinner, 42
 - domain.game.Game, 36
 - drawCard, 37
 - Game, 37
 - getCurrentPlayer, 37
 - getHandCards, 37
 - getPlayers, 37
 - getWinner, 37
 - isOver, 37
 - peekTopCard, 37
 - playCard, 38
 - domain.game.GameBuilder, 40
 - build, 41
 - withPlayer, 41
 - domain.player, 13
 - domain.player.Direction, 32
 - CLOCKWISE, 32
 - COUNTER_CLOCK_WISE, 32
 - domain.player.HandCardList, 43
 - addCard, 43
 - getCardStream, 43
 - hasCard, 43

- removeCard, 43
- size, 44
- domain.player.ImmutablePlayer, 46
 - getHandCards, 46
 - getId, 46
 - getName, 46
 - getTotalCards, 46
 - ImmutablePlayer, 46
- domain.player.Player, 48
 - addToHandCards, 49
 - getHandCards, 49
 - getName, 49
 - hasHandCard, 50
 - Player, 49
 - removePlayedCard, 50
 - toImmutable, 50
- domain.player.PlayerRoundIterator, 51
 - getCurrentPlayer, 51
 - getPlayerById, 51
 - next, 51
 - PlayerRoundIterator, 51
 - reverseDirection, 52
 - stream, 52
- domain.TestHandCardList, 55
 - GivenDifferentNumberCard_ShouldNotExist, 56
 - GivenSameNumberCard_ShouldExist, 56
- domain.testhelper, 13
- domain.testhelper.CardTestFactory, 25
 - createActionCard, 26
 - createDrawTwoCard, 26
 - createNumberCard, 26
 - createReverseCard, 26
 - createSkipCard, 26
 - createWildCard, 27
 - createWildColorCard, 27
 - createWildDrawFourCard, 27
- domain.testhelper.PlayerTestFactory, 52
 - createPlayers, 52
- DomainEvent
 - domain.common.DomainEvent, 33
- DRAW_TWO
 - domain.card.CardType, 28
- drawCard
 - application.GameAppService, 39
 - application.IGameAppService, 44
 - domain.game.DrawPile, 35
 - domain.game.Game, 37
- DrawPile
 - domain.game.DrawPile, 35
- Entity
 - domain.common.Entity, 35
- equals
 - domain.card.AbstractCard, 16
 - domain.card.ActionCard, 17
 - domain.card.NumberCard, 48
 - domain.card.WildCard, 57
- Game
 - domain.game.Game, 37
- GameAppService
 - application.GameAppService, 39
- GameOver
 - domain.game.events.GameOver, 41
- GameStatusView
 - ui.view.GameStatusView, 42
- getCardStream
 - domain.player.HandCardList, 43
- getColor
 - domain.card.AbstractCard, 16
 - domain.card.Card, 19
- getCurrentPlayer
 - application.GameAppService, 39
 - application.IGameAppService, 44
 - domain.game.Game, 37
 - domain.player.PlayerRoundIterator, 51
- getDimension
 - ui.view.CardView, 30
- getDrawTwoColorCounts
 - domain.card.CardCounter, 20
- getHandCards
 - application.GameAppService, 39
 - application.IGameAppService, 44
 - domain.game.Game, 37
 - domain.player.ImmutablePlayer, 46
 - domain.player.Player, 49
- getId
 - application.dto.PlayerInfoDTO, 51
 - domain.common.Entity, 36
 - domain.player.ImmutablePlayer, 46
- getImmutableCards
 - domain.card.CardDeck, 22
- getInstance
 - ui.view.ColorPicker, 31
- getName
 - application.dto.PlayerInfoDTO, 51
 - domain.player.ImmutablePlayer, 46
 - domain.player.Player, 49
- getNumberColorCounts
 - domain.card.CardCounter, 20
- getNumberCounts
 - domain.card.CardCounter, 20
- getOccurredDate
 - domain.common.DomainEvent, 33
- getPlayedCard
 - domain.game.events.CardPlayed, 24
- getPlayerById
 - domain.player.PlayerRoundIterator, 51
- getPlayerId
 - domain.game.events.CardDrawn, 23
 - domain.game.events.CardPlayed, 24
- getPlayerInfos
 - application.GameAppService, 39
 - application.IGameAppService, 45
- getPlayers
 - domain.game.Game, 37
- getReverseColorCounts

- domain.card.CardCounter, 20
- getSize
 - domain.game.DrawPile, 35
- getSkipColorCounts
 - domain.card.CardCounter, 21
- getTotalCards
 - domain.player.ImmutablePlayer, 46
- getType
 - domain.card.AbstractCard, 16
 - domain.card.Card, 19
- getValue
 - domain.card.NumberCard, 48
- getValueToDisplay
 - ui.common.StyleUtil, 54
- getWildCardCounts
 - domain.card.CardCounter, 21
- getWinner
 - application.GameAppService, 39
 - application.IGameAppService, 45
 - domain.game.events.GameOver, 42
 - domain.game.Game, 37
- GivenDifferentNumberCard_ShouldNotExist
 - domain.TestHandCardList, 56
- GivenSameNumberCard_ShouldExist
 - domain.TestHandCardList, 56
- GREEN
 - domain.card.CardColor, 19
- greenColor
 - ui.common.StyleUtil, 54
- handleEvent
 - domain.common.DomainEventSubscriber, 34
 - ui.view.GameStatusView, 42
 - ui.view.PlayerView, 53
 - ui.view.TableView, 55
- hasCard
 - domain.player.HandCardList, 43
- hasHandCard
 - domain.player.Player, 50
- hashCode
 - domain.card.AbstractCard, 16
 - domain.card.ActionCard, 17
 - domain.card.NumberCard, 48
 - domain.card.WildCard, 57
- ImmutablePlayer
 - domain.player.ImmutablePlayer, 46
- isGameOver
 - application.GameAppService, 40
 - application.IGameAppService, 45
- isOver
 - domain.game.Game, 37
- isValidActionCard
 - domain.game.CardRules, 25
- isValidNumberCard
 - domain.game.CardRules, 25
- isValidWildCard
 - domain.game.CardRules, 25
- isWildCard
 - domain.card.CardUtil, 29
- main
 - UnoApp, 56
- next
 - domain.player.PlayerRoundIterator, 51
- NUMBER
 - domain.card.CardType, 28
- NumberCard
 - domain.card.NumberCard, 47
- paintComponent
 - ui.view.CardView, 30
 - ui.view.GameStatusView, 42
- peekTopCard
 - application.GameAppService, 40
 - application.IGameAppService, 45
 - domain.game.Game, 37
- playCard
 - application.GameAppService, 40
 - application.IGameAppService, 45
 - domain.game.Game, 38
- Player
 - domain.player.Player, 49
- PlayerInfoDTO
 - application.dto.PlayerInfoDTO, 50
- PlayerRoundIterator
 - domain.player.PlayerRoundIterator, 51
- PlayerView
 - ui.view.PlayerView, 53
- project docs/README.md, 59
- publish
 - domain.common.DomainEventPublisher, 33
- RED
 - domain.card.CardColor, 19
- redColor
 - ui.common.StyleUtil, 54
- removeCard
 - domain.player.HandCardList, 43
- removePlayedCard
 - domain.player.Player, 50
- reset
 - domain.common.DomainEventPublisher, 33
- REVERSE
 - domain.card.CardType, 28
- reverseDirection
 - domain.player.PlayerRoundIterator, 52
- setError
 - ui.view.GameStatusView, 43
- show
 - ui.view.ColorPicker, 31
- shuffle
 - domain.game.DealerService, 31
- size
 - domain.player.HandCardList, 44
- SKIP

- domain.card.CardType, 28
- src/main/java/application/dto/PlayerInfoDTO.java, 59
- src/main/java/application/GameAppService.java, 59
- src/main/java/application/IGameAppService.java, 60
- src/main/java/domain/card/AbstractCard.java, 60
- src/main/java/domain/card/ActionCard.java, 60
- src/main/java/domain/card/Card.java, 60
- src/main/java/domain/card/CardColor.java, 61
- src/main/java/domain/card/CardDeck.java, 61
- src/main/java/domain/card/CardType.java, 61
- src/main/java/domain/card/CardUtil.java, 61
- src/main/java/domain/card/NumberCard.java, 62
- src/main/java/domain/card/WildCard.java, 62
- src/main/java/domain/common/DomainEvent.java, 62
- src/main/java/domain/common/DomainEventPublisher.java, 63
- src/main/java/domain/common/DomainEventSubscriber.java, 63
- src/main/java/domain/common/Entity.java, 63
- src/main/java/domain/game/CardRules.java, 63
- src/main/java/domain/game/DealerService.java, 64
- src/main/java/domain/game/DrawPile.java, 64
- src/main/java/domain/game/events/CardDrawn.java, 64
- src/main/java/domain/game/events/CardPlayed.java, 65
- src/main/java/domain/game/events/GameOver.java, 65
- src/main/java/domain/game/Game.java, 65
- src/main/java/domain/game/GameBuilder.java, 66
- src/main/java/domain/player/Direction.java, 66
- src/main/java/domain/player/HandCardList.java, 66
- src/main/java/domain/player/ImmutablePlayer.java, 67
- src/main/java/domain/player/Player.java, 67
- src/main/java/domain/player/PlayerRoundIterator.java, 67
- src/main/java/ui/AppFrame.java, 68
- src/main/java/ui/common/StyleUtil.java, 68
- src/main/java/ui/view/CardView.java, 68
- src/main/java/ui/view/ColorPicker.java, 68
- src/main/java/ui/view/GameStatusView.java, 69
- src/main/java/ui/view/PlayerView.java, 69
- src/main/java/ui/view/TableView.java, 69
- src/main/java/UnoApp.java, 70
- src/test/java/domain/card/CardCounter.java, 70
- src/test/java/domain/card/CardCounterAssertionHelper.java, 70
- src/test/java/domain/card/TestCard.java, 71
- src/test/java/domain/card/TestCardDeck.java, 71
- src/test/java/domain/card/TestCardUtil.java, 71
- src/test/java/domain/common/TestDomainEventPublisher.java, 71
- src/test/java/domain/game/TestCardRules.java, 72
- src/test/java/domain/game/TestDealerService.java, 72
- src/test/java/domain/game/TestDrawPile.java, 72
- src/test/java/domain/game/TestGameBuilder.java, 73
- src/test/java/domain/game/TestGamePlay.java, 73
- src/test/java/domain/game/TestGameStart.java, 73
- src/test/java/domain/player/TestHandCardList.java, 74
- src/test/java/domain/player/TestPlayersIterator.java, 74
- src/test/java/domain/TestHandCardList.java, 74
- src/test/java/domain/testhelper/CardTestFactory.java, 75
- src/test/java/domain/testhelper/PlayerTestFactory.java, 75
- stream
 - domain.player.PlayerRoundIterator, 52
- subscribe
 - domain.common.DomainEventPublisher, 33
- TableView
 - ui.view.TableView, 55
- toImmutable
 - domain.player.Player, 50
- toString
 - domain.card.AbstractCard, 16
 - domain.card.ActionCard, 17
 - domain.card.NumberCard, 48
 - domain.card.WildCard, 57
 - TOTAL_INITIAL_HAND_CARDS
 - domain.game.DealerService, 32
- ui, 13
- ui.AppFrame, 18
 - AppFrame, 18
- ui.common, 13
- ui.common.StyleUtil, 53
 - blackColor, 54
 - blueColor, 54
 - convertCardColor, 54
 - DEFAULT_FONT, 54
 - getValueToDisplay, 54
 - greenColor, 54
 - redColor, 54
 - yellowColor, 54
- ui.view, 13
- ui.view.CardView, 29
 - CardView, 30
 - getDimension, 30
 - paintComponent, 30
- ui.view.ColorPicker, 30
 - getInstance, 31
 - show, 31
- ui.view.GameStatusView, 42
 - GameStatusView, 42
 - handleEvent, 42
 - paintComponent, 42
 - setError, 43
- ui.view.PlayerView, 52
 - handleEvent, 53
 - PlayerView, 53
- ui.view.TableView, 55
 - handleEvent, 55
 - TableView, 55
- UNO - Card Game, 1
- UnoApp, 56
 - main, 56
- unsubscribe
 - domain.common.DomainEventPublisher, 34
- validateActionType

- domain.card.CardUtil, 29
- validateColor
 - domain.card.CardUtil, 29
- validateNumber
 - domain.card.CardUtil, 29
- WILD_COLOR
 - domain.card.CardType, 28
- WILD_DRAW_FOUR
 - domain.card.CardType, 28
- WildCard
 - domain.card.WildCard, 57
- withPlayer
 - domain.game.GameBuilder, 41
- YELLOW
 - domain.card.CardColor, 19
- yellowColor
 - ui.common.StyleUtil, 54