

NOTE: The folder `../CardGameGuiUltimate` contains a visually enhanced and optimised version of this project, but is not being submitted due to the excessive code size. (It implements a debouncing function, tweened card animations, bitmap caching, draggable cards on a JLayeredPane, and is a most beautiful user experience. You are encouraged to view the results for the sheer joy of doing so. *Or just watch the video.*

<https://youtu.be/Ruv-dXsDH5U>

The folder `../CardGameGui` contains the version of this project intended for marking.

To test other implementations of GameEngineImpl, simply place the implementation into `../CardGame/src/model/` and compile the project in `../CardGameGui`.

To implement this GUI extension to your own A1 style project, you can add all file and folders in the `../CardGameGui/res/*` and `../CardGameGui/src/` folders (besides the one in `src/client`), and copy the functionality of `client/SimpleTestClient.java`.

Layout of source code:

path	description
src	
├ client	
└ SimpleTestClientGUI.java	main(String[])
├ controller	
└ ApplicationController.java	Main Controller for Application JFrame
└ listeners	Listeners for for ApplicationController
└ └ AddPlayerButtonListener.java	
└ └ ClearBetButtonListener.java	
└ └ DealActionListener.java	
└ └ DealAllActionListener.java	
└ └ MenuItemListener.java	
└ └ MenuItemPlayerListener.java	
└ └ NewPlayerDialogCancelButtonListener.java	
└ └ NewPlayerDialogOkayButtonListener.java	
└ └ PlayerSelectionChangeListener.java	
└ └ RemovePlayerButtonListener.java	
└ └ SetBetButtonListener.java	
└ └ SummaryTableListSelectionListener.java	
├ model	
└ ForwardingGameEngine.java	Reusable forwarding class for GameEngine interface [Bloch17]
└ GameEngineImplEx.java	Extended GameEngine Implementation, extends ForwardingGameEngine
├ util	
└ Debug.java	Plugin replacements for System.out.println and format
└ MyMath.java	Replacement for JavaFX's <code>clamp</code> function
└ StringHelpers.java	Enhanced Integer.parseInt with default value return on exception
└ swing	

path	description
└─ containers	
└─ BackgroundImagePanel.java	JPanel with tiled background image (hue adjustable)
└─ ImagePanel.java	Abstract superclass for above
└─ FontHelper.java	Function to load TrueType fonts from resources
└─ JComponent.java	jQuery style (function chaining) JComponent builder
└─ layoutmanagers	
└─ AspectLayout.java	A layout to maximise component size to match a ratio
└─ CanvasLayout.java	Minimal layout for self-drawing components
└─ Timers.java	Common implementation of Thread.sleep with exception handling
└─ view	
└─ AppView.java	Main View (Application JFrame)
└─ GameEngineCallbackGUI.java	GameEngineCallback compliant implementation for GUI notification
└─ GameEngineCallbackGuiEx.java	Extended version of the above with extra (required) methods
└─ interfaces	
└─ GameEngineCallbackEx.java	Interface for extended methods of GameEngineCallback
└─ NewPlayerDialog.java	“Add Player” dialog with input fields
└─ PlayingCardPanel.java	Self-drawing Playing Card
└─ SummaryTable.java	Subclassed JTable with custom headers & cells
└─ viewmodel	
└─ PlayerList.java	List of PlayerWrappers & an Observable for added/removed entries
└─ PlayerWrapper.java	A wrapped (using composition) enhanced yet backwards compatible and interface compliant Player class. Players are aware of their own index (in lists and tables) and adjust their index via subscription to PlayerList's Observables