

## ViewController RefreshImageView StockView override var preferredStatusBarStyle func configure() func postNotification() let cardDeck: CardDeck func push(\_ card: Card, at index: Int?) var backgroundView: BackgroundView func removeAll() var stockView: StockView var wasteView: WasteView CardImageView var foundationContatinerView: FoundationContainerView WasteView var tableauContainerView: TableauContainerView @objc public func tapAction(~) var stcokViewModel: StockViewModel override func layoutSubViews() @objc public func var wasteViewModel: WasteViewModel private func removeAllSubView() dobuleTapActionWaste(~) var foundationViewModel: FoundationViewModel private func addAllSubView() @objc public func var tableauViewModel: TableauViewModel private func addGestureCardView(with dobuleTapActionTableau(~) func configureView() view: CardImageView) @objc public func panActionWaste(~) func configureDataSource() @objc public func panAction(~) func configureCard() func configureObservers() // Motion << Protocol>> override func motionBegan( motion: UIEvent.EventSubtype, with event: FoundationContainerView DeliverableViewable UIEvent?) func reconfigure() private func configure() func drawSubView() // Notification private func cardMold(x xValue: CGFloat, y func convert(at index: Int?, to view: vValue: CGFloat) -> UIView UIView) -> CGPoint? @objc private func redrawStock() @obic private func redrawWaste() override func lavoutSubviews() func topSubView(at index: Int?) -> @objc private func redrawTableau() private func removeAllSubView() UIView? @objc private func redrawFoundation() private func addAllSubView() @objc private func moveCardToWaste() subscript(index: Int) -> UIView @objc private func restoreCard() @objc private func completeGame() << Protocol>> // Common: Drag & DoubleTap DestinationViewable private func configureDelivery(\_ notification: Notification) -> Delivery func convert(at index: Int?, to view: private func popDeliveryCard(with delivery: Delivery) -> Card? **TableauContainerView** UIView) -> CGPoint? // DoubleTap @objc private func doubleTapCard(\_ notification: Notification) private func configure() private func aceEvent(with delivery: Delivery) private func cardMold(x xValue: CGFloat, v private func kingEvent(with delivery: Delivery) yValue: CGFloat) -> UIView private func normalEvent(with delivery: Delivery) override func layoutSubviews() private func findFoundation(with delivery: Delivery, card; Card) -> Bool << Protocol>> private func removeAllSubView() private func findTableau(with delivery: Delivery, card: Card) -> Bool CardReceiving private func addAllSubVie() private func moveCard(from delivery: Delivery, to destination: Destination) func selectedSubViews(at index: Int, sub var dataSource: SingleDataSource? { get } subIndex: Int) -> [CardImageView] @objc private func drag( notification: Notification) subscript(index: Int) -> UIView private func configureDragInfo(with notification: Notification) -> DragInfo? private func addGestureCardView(with private func configureSelected(with delivery: Delivery) -> ([UIView], Card)? view: CardImageView, index: Int, private func configureCenters(from selectedCardViews: [UIView], with subIndex: Int) subIndex: Int?) -> [CGPoint] private func calculateTargetIndex(from point: CGPoint) -> Int << Protocol>> private func movingCard(with recognizer: UIPanGestureRecognizer, view CardContainerReceiving selectedCardViews: [UIView]) var dataSource: MultipleDataSource? { get } private func separateCategory(with info: CoordinatesInfo) -> Bool private func moveCardTarget(from coordinatesInfo: CoordinatesInfo, with BackgroundView delivery: Delivery) private func configure() private func returnOriginalPosition(info originalCenters: [CGPoint], with selectedCardViews: [UIView])