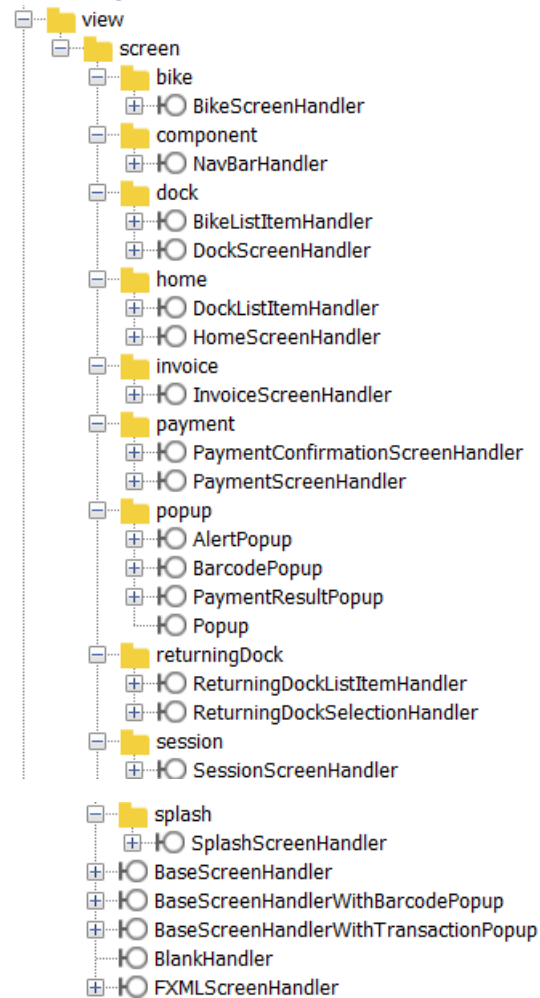
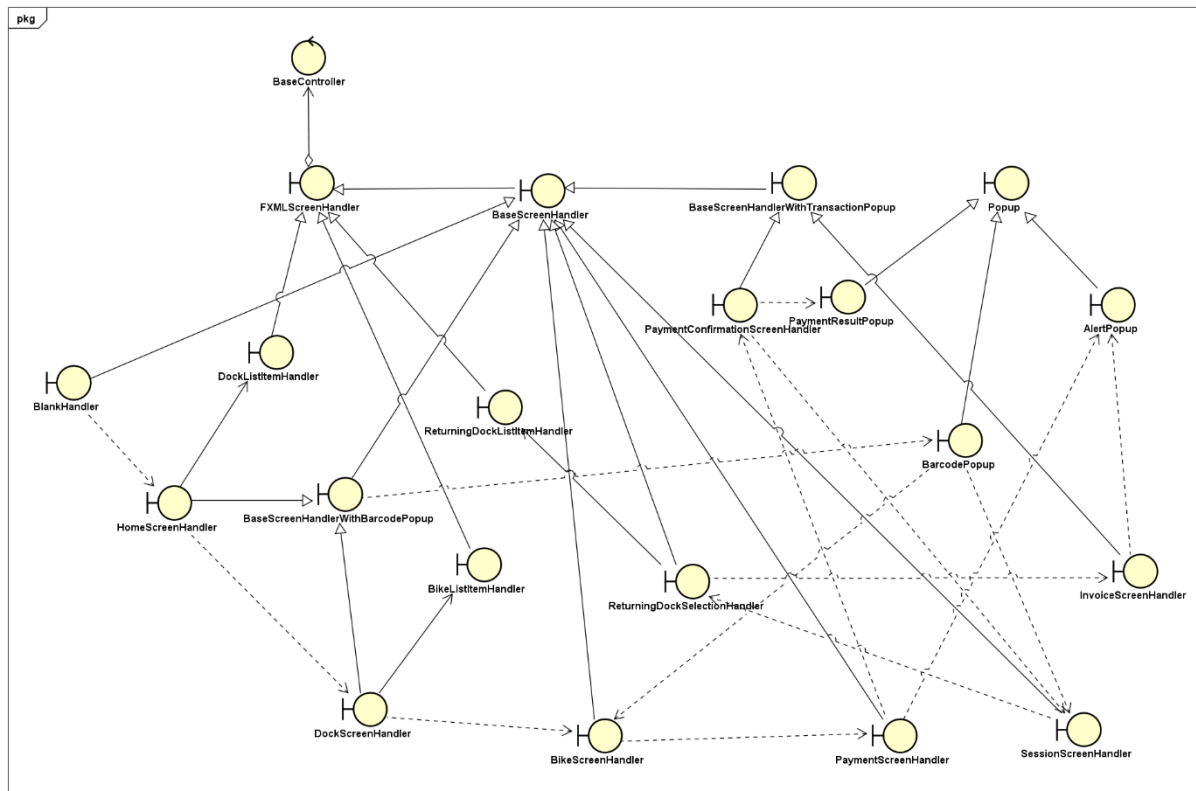


# Class Design - View

## 1. Create Initial Design Classes

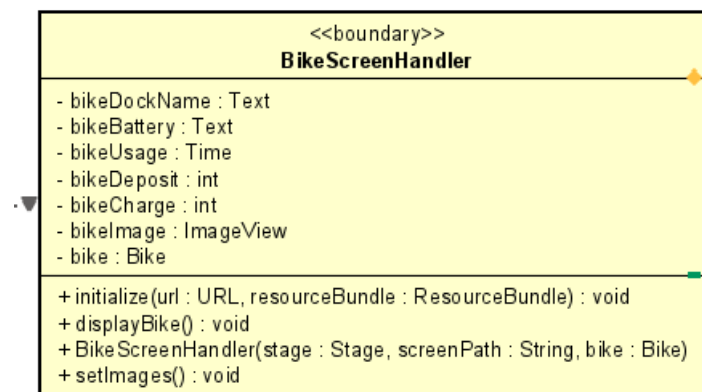


## 2. Define Relationships Between Classes



### 3. Class Design

### 3.1. Class “BikeScreenHandler”



## Attribute

#	Name	Data type	Default value	Description
1	bikeDockName	Text	NULL	Dock name fxml
2	bikeBattery	int	NULL	Bike battery fxml
3	bikeUsage	Time	NULL	Bike usage fxml
4	bikeDeposit	int	NULL	Bike deposit fxml
5	bikeCharge	int	NULL	Bike charge fxml
6	bikeImage	ImageView	NULL	Bike image
7	bike	Bike	NULL	bike

## Operation

#	Name	Return type	Description
1	initialize	void	initialize
2	displayBike	void	Display bike info screen
3	setImages	void	Set bike image

## Parameter:

- initialize
  - url - URL
  - resourceBundle - ResourceBundle
- BikeScreenHandle
  - stage - Stage

## Exception:

None

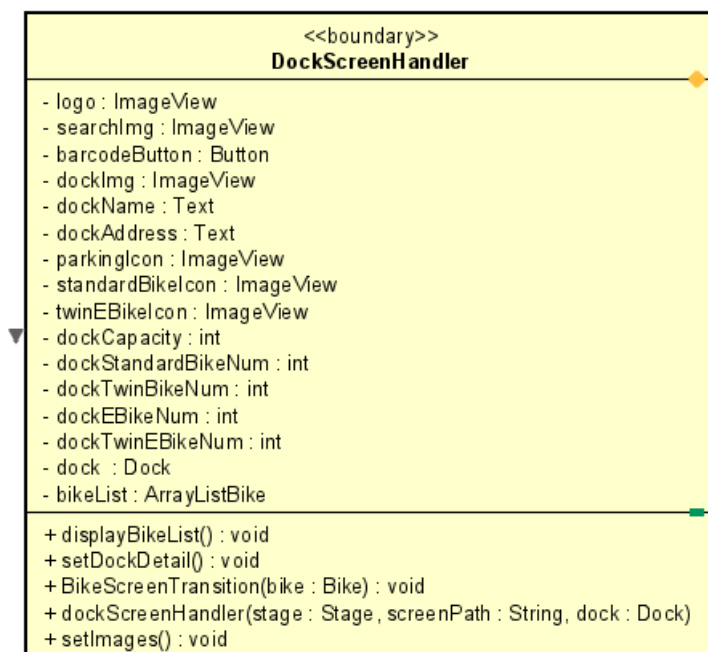
## Method

None

## State

None

## 3.2. Class “DockScreenHandler”



## Attribute

#	Name	Data type	Default value	Description
---	------	-----------	---------------	-------------

1	logo	ImageView	NULL	Logo image fxml
2	searchImg	imageView	NULL	Search image fxml
3	barcodeButton	Button	NULL	BarcodeButton fxml
4	dockName	Text	NULL	Dock name
5	dockAddress	Text	NULL	Dock address
6	dockImg	ImageView	NULL	Dock image fxml
7	parkingIcon	ImageView	NULL	Parking icon fxml
8	standardBikeIcon	ImageView	NULL	Standard bike icon fxml
9	twinEBikeIcon	ImageView	NULL	Twin bike icon fxml
10	dockStandardBikeNum	Int	NULL	Standard bike number
11	dockTwinBikeNum	Int	NULL	Twin bike number f
12	dockEBikeNum	Int	NULL	E-bike number
13	dockTwinEBikeNum	Int	NULL	Twin e-bikenumber
14	dock	Dock	NULL	Dock
15	bikeList	ArrayList<Bike>	NULL	Bike list in dock

## Operation

#	Name	Return type	Description
1	displayBikeList	void	Display bike list
2	setDockDetail	void	Set dock detail
3	BikeScreenTransition	void	Transition to bike screen
4	setImages	void	Set dock image

## Parameter:

- BikeScreenTransition
  - bike
- dockScreenHandle
  - stage – Stage
  - screenPath – String
  - dock - Dock

## Exception:

None

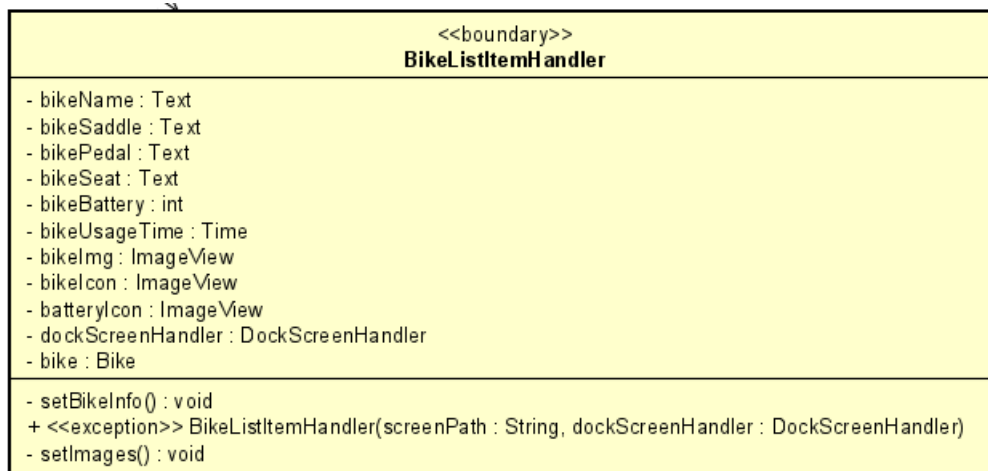
## Method

None

## State

None

### 3.3. Class “BikeListItemHandler”



#### Attribute

#	Name	Data type	Default value	Description
1	bikeImg	ImageView	NULL	Bike image fxml
2	bikeIcon	imageView	NULL	Bike image fxml
4	bikeName	Text	NULL	Bike name
5	bikeSaddle	Text	NULL	Bike saddle
6	bikePedal	Text	NULL	Bike pedal
7	bikeSeat	Text	NULL	Bike seat
8	dockScreenHandler	DockScreenHandler	NULL	object
9	bike	Bike	NULL	object
10	bikeBattery	Int	NULL	Bike Battery
11	BikeUsageTime	Time	NULL	Bike Usage Time

#### Operation

#	Name	Return type	Description
1	setBikeInfo	void	Set bike info
2	setImages	void	Set bike image in list

#### Parameter:

none

#### Exception:

IOException

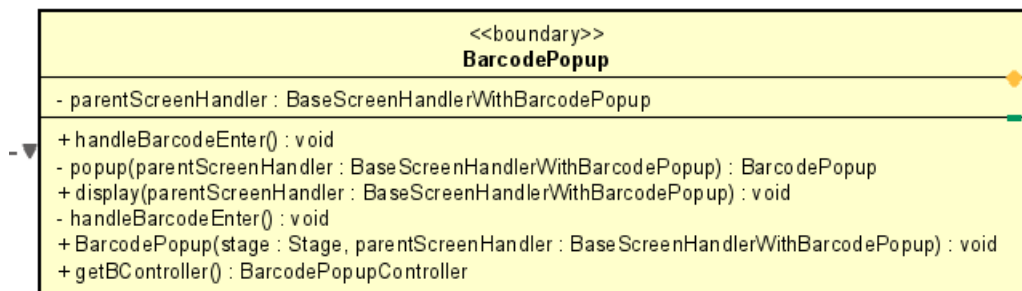
#### Method

None

#### State

None

### 3.4. Class “BarcodePopup”



#### Attribute

#	Name	Data type	Default value	Description
1	parentScreenHandler	BaseScreenHandlerWithBarcodePopup	NULL	Parent screen

#### Operation

#	Name	Return type	Description
1	handleBarcodeEnter	void	Handle action “enter barcode”
2	popup	BaseScreenHandlerWithBarcodePopup	BarcodePopup
3	display	void	Display popup
4	handleBarcodeEnter	void	Button enter handler
5	barcodePopup	void	Popup barcode
6	getBController	BarcodePopupController	Get barcode

#### Parameter:

- popup
  - parentScreenHandler
- display
  - parentScreenHandler
- barcodePopup
  - parentScreenHandler
  - stage

#### Exception:

None

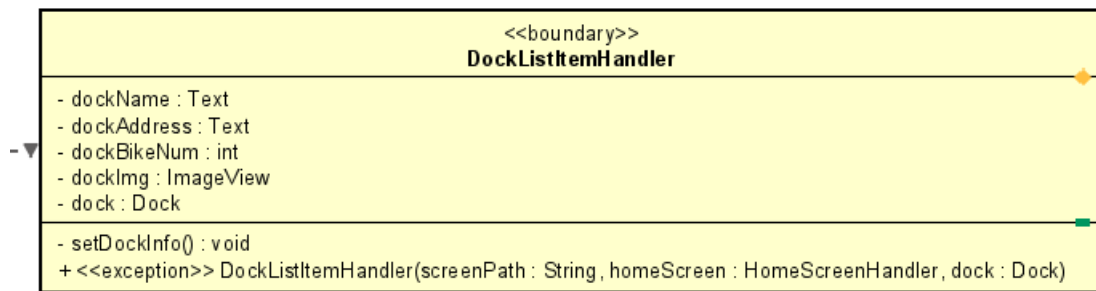
#### Method

None

#### State

None

### 3.5. Class “DockListItemHandler”



#### Attribute

#	Name	Data type	Default value	Description
1	dockName	Text	NULL	Dock name
2	dockAddress	Text	NULL	Dock address
3	dockBikeNum	int	NULL	Number of bikes in dock
4	dockImg	ImageView	NULL	Dock image
5	dock	Dock	NULL	dock

#### Operation

#	Name	Return type	Description
1	setDockInfo	void	Set dock info

#### Parameter:

None

#### Exception:

IOException

#### Method

None

#### State

None

### 3.6. Class “InvoiceScreenHandler”

<<boundary>> InvoiceScreenHandler	
- invoiceBikeImage : ImageView - invoiceCardNumber : Text - invoiceStartTime : DateTime - invoiceEndTime : DateTime - invoiceDeposit : int	
+ InvoiceScreenHandler(stage : Stage, screenPath : String, invoice : Invoice, controller : InvoiceScreenController) + handleCheckBox() : void + handleConfirm() : void - setTextFields() : void + continueAfterPopupClosed(paymentTransaction : PaymentTransaction) : void	

#### Attribute

#	Name	Data type	Default value	Description
2	invoiceBikeImage	ImageView	NULL	Bike image
3	invoiceCardNumber	Text	NULL	Invoice card number
4	invoiceStartTime	DateTime	NULL	Invoice start time
5	invoiceEndTime	DateTime	NULL	Invoice end time
6	invoiceDeposit	int	NULL	Invoice deposit

#### Operation

#	Name	Return type	Description
1	handleCheckBox	void	Handle check box
2	handleConfirmButton	void	Handle confirm button
3	setTextFields	void	Set text fields
4	continueAfterPopupClosed	void	Continue program after popup close

#### Parameter:

- continueAfterPopupClosed
  - paymentTransaction

#### Exception:

None

#### Method

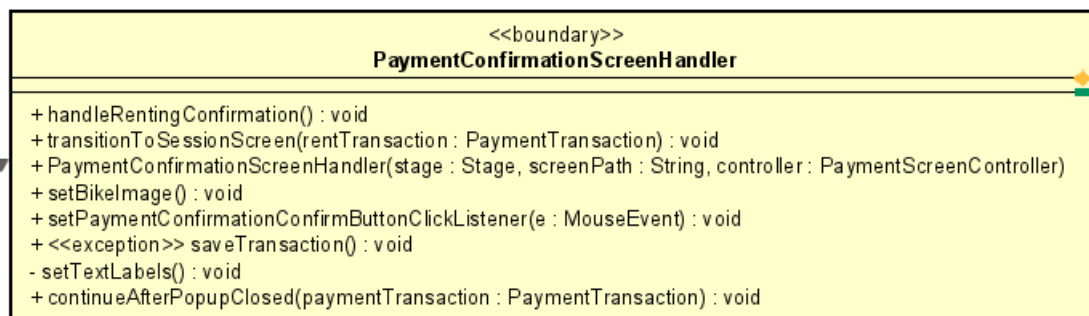
None

#### State

None



### 3.7. Class “PaymentConfirmationScreenHandler”



#### Attribute

#	Name	Data type	Default value	Description
---	------	-----------	---------------	-------------

#### Operation

#	Name	Return type	Description
1	handleRentingConfirmation	void	Handle confirmation action
2	transitionToSessionScreen	void	Transition to session screen
3	paymentConfirmationScreenHandler	void	Handle confirm payment
4	setBikeImage	void	Set bike image
5	setPaymentConfirmationConfirmButtonClickListener	void	Payment Confirmation Confirm Button Click Listener
6	saveTransaction	void	Save transaction
7	setTextLabels	void	Set text
8	continueAfterPopupClosed	void	Continue program after popup close

#### Parameter:

- transitionToSessionScreen
  - rentTransaction
- paymentConfirmationScreenHandler
  - stage
  - screenPath
  - controller
- setPaymentConfirmationConfirmButtonClickListener
  - e
- continueAfterPopupClosed
  - paymentTransaction

#### Exception:

IOException

**Method**

None

**State**

None

### 3.8. Class “PaymentScreenHandler”

<<boundary>> <b>PaymentScreenHandler</b>	
+ handleCardInfoSubmit() : void + gotoConfirmationScreen() : void + PaymentScreenHandler(stage : Stage, screenPath : String, paymentScreenController : PaymentScreenController) - setImages() : void	

**Attribute**

#	Name	Data type	Default value	Description
---	------	-----------	---------------	-------------

**Operation**

#	Name	Return type	Description
1	handleCardInfoSubmit	void	Handle action submit card
2	gotoConfirmationScreen	void	Transition to confirmation screen
3	setImages	vodi	Set payment image

**Parameter:**

None

**Exception:**

None

**Method**

None

**State**

None

### 3.9. Class “ReturningDockListSelectionHandler”

<<boundary>> ReturningDockSelectionHandler	
- dockImg : ImageView - dockAddress : Text - dockEmptySlots : int - returnBikeBtn : Button - dockInfo : Text - dock : Dock - session : Session - dockList : ArrayListDock	
+ displayDockList(dock : Dock) : void + onDockListItemClicked() : void + ReturningDockSelectionHandler(stage : Stage, screenPath : String, returningDockSelectionController : ReturningDockSelectionController, session : Session) + setImages() : void + onDockListItemClicked(dock : Dock) : void + displayDockList() : void + searchImgBtnListener(e : MouseEvent) : void + returnBikeBtnListener(e : MouseEvent) : void	

#### Attribute

#	Name	Data type	Default value	Description
1	returnBikeBtn	Button	NULL	Button to return bike
2	dockEmptySlots	int	NULL	Dock parking slot
3	dockImg	ImageView	NULL	Dock image fxml
4	dockInfo	Text	NULL	Dock information
5	dockAddress	Text	NULL	Dock address
6	dock	Dock	NULL	dock
7	session	Session	NULL	Session
8	dockList	ArrayListDock	NULL	Dock list

#### Operation

#	Name	Return type	Description
1	setDockInfo	void	Set dock info
2	onDockListItemClicked	void	Return dock clicked listener
3	setImages	void	Set return dock image
4	displayDockList	void	Display return dock list
5	searchImgBtnListener	void	Search image button listener
6	returnBikeBtnListener	void	Return bike button listener

#### Parameter:

None

#### Exception:

None

#### Method

None

#### State

None

### 3.10. Class “ReturningDockListItemHandler”

<<boundary>> ReturningDockListItemHandler	
- dockName : Text - dockParkingSlot : int - dockImg : ImageView - dock : Dock - returningDockSelectionHandler : ReturningDockSelectionHandler	
+ ReturningDockListItemHandler(screenPath : String, returningDockSelectionHandler : ReturningDockSelectionHandler) + setDockInfo() : void + setDock(dock : Dock) : void + setDockInfo() : void	

#### Attribute

#	Name	Data type	Default value	Description
1	dockName	Text	NULL	Dock name fxml
2	dockParkingSlot	int	NULL	Dock address fxml
3	dockImg	ImageView	NULL	Dock image fxml
4	dock	Dock	NULL	dock
5	returningDockSelectionHandler	ReturningDockSelectionHandler	NULL	Returning Dock Selection Handler

#### Operation

#	Name	Return type	Description
1	setDockInfo	void	Display dock information
2	setDock	void	Display dock

#### Parameter:

- setDock
  - dock

#### Exception:

None

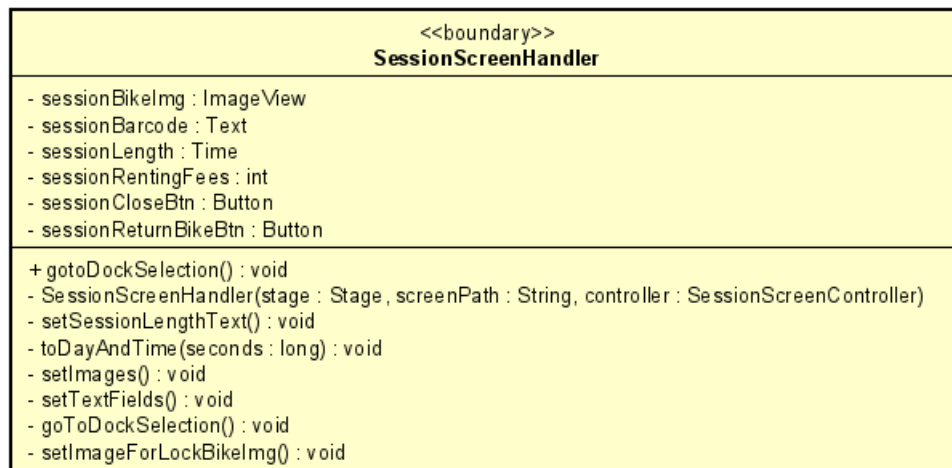
#### Method

None

## State

None

### 3.11. Class “SessionScreenHandler”



## Attribute

#	Name	Data type	Default value	Description
1	sessionBikeImg	ImageView	NULL	Bike image fxml
2	sessionBarcode	Text	NULL	Session barcode
3	sessionLength	Time	NULL	Session length
4	sessionRentingFees	Int	NULL	Session renting fees
5	sessionCloseBtn	Button	NULL	Button
6	sessionReturnBikeBtn	Button	NULL	Button

## Operation

#	Name	Return type	Description
1	gotoDockSelecion	void	Transition to dock selection
2	setSessionLengthText	void	Set session's length text
3	toDayAndTime	void	Day and time conversion
4	setImages	void	Set image for session
5	setTextFields	void	Set text fields for session
6	goToDockSelection	void	Back to the dock selection
7	setImageForLockBikeImg	void	Set image for bike

## Parameter:

- ToDayAndTime
  - seconds

#### Exception:

None

#### Method

None

#### State

None

### 3.12. Class “HomeScreenHandler”

<<boundary>> HomeScreenHandler	
- dockList : ArrayListDock - logo : ImageView - barcodeButton : Button - searchField : TextField	
+ displayDockList() : void + onDockListItemClicked(dock : Dock) : void + moveToSessionScreen(session : Session) : void + moveToBikeViewScreen(bike : Bike) : void + show() : void + <<exception>> HomeScreenHandler(stage : Stage, screenPath : String, homeScreenController : HomeScreenController)	

#### Attribute

#	Name	Data type	Default value	Description
1	vboxDockList	VBox	NULL	Vbox fxml
2	dockList	ArrayListDock	NULL	Dock list
3	logo	ImageView	NULL	Logo fxml
4	barcodeButton	Button	NULL	Barcode button fxml
5	searchField	TextField	NULL	Search field fxml

#### Operation

#	Name	Return type	Description
1	displayDockList	void	Display dock list
2	onDockListItemClicked	void	Dock list item clicked listener
3	moveToSessionScreen	void	Move to session screen
4	moveToBikeViewScreen	void	Move to bike view screen
5	show	void	Show screen

#### Parameter:

- onDockListItemClicked
  - dock
- moveToSessionScreen
  - session
- moveToBikeViewScreen
  - bike

**Exception:**

IOException

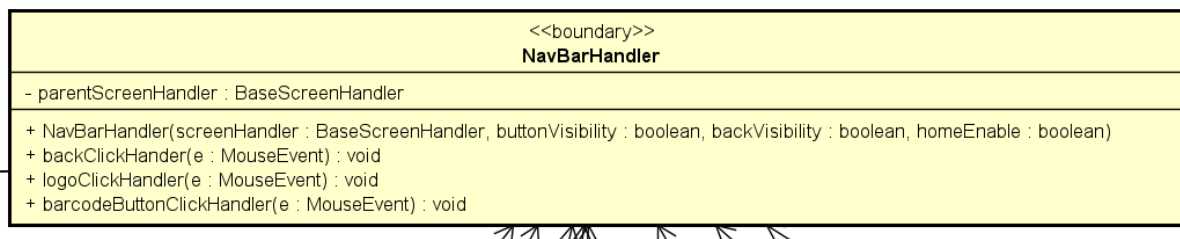
**Method**

None

**State**

None

### 3.13. Class “NavBarHandler”



**Attribute**

#	Name	Data type	Default value	Description
1	parentScreenHandler	BaseScreenHandler	NULL	handler

**Operation**

#	Name	Return type	Description
1	backClickHandler	void	go to previous screen
2	logoClickHandler	void	go to home screen
3	barcodeButtonClickHandler	void	bring up barcode popup

**Parameter:**

- backClickHandler
  - e – MouseEvent
- logoClickHandler
  - e - MouseEvent

**Exception:**

None

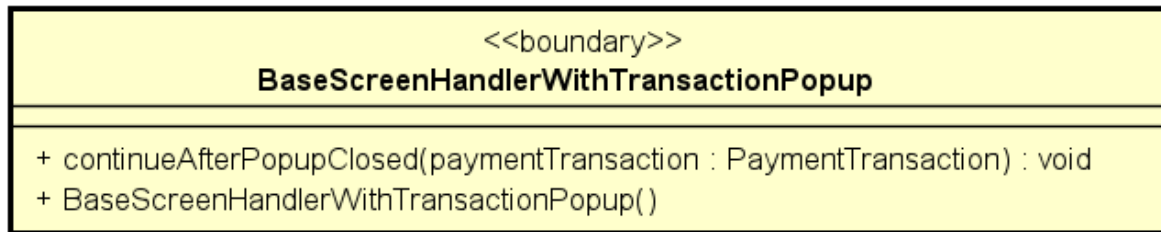
**Method**

None

## State

None

### 3.14. Class “BaseScreenHandlerWithTransactionPopup”



## Attribute

None

## Operation

#	Name	Return type	Description
1	continueAfterPopupClosed	void	show then closed popup of payment result

## Parameter:

- continueAfterPopupClosed
  - paymentTransaction – transaction to be shown result

## Exception:

None

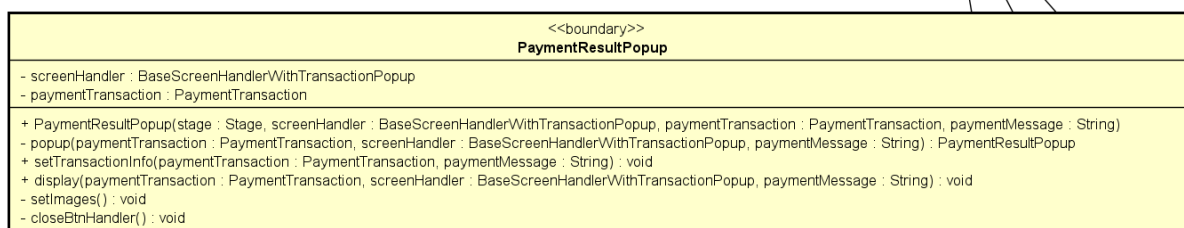
## Method

None

## State

None

### 3.15. Class “PaymentResultPopup”



## Attribute



#	Name	Data type	Default value	Description
1	screenHandler	BaseScreenHandlerWithTransactionPopup	new	handler
2	paymentTransaction	PaymentTransaction	NULL	transaction to be shown

### Operation

#	Name	Return type	Description
1	setTransactionInfo	void	Display result
2	display	void	Render result
3	setImage	void	Set image for screen
4	closeBtnHandler	void	Close the popup and move to next screen
5	popup	PaymentResultPopup	Create then return popup

### Parameter:

- popup
  - screenHandler
  - paymentTransaction
  - paymentMessage

### Exception:

None

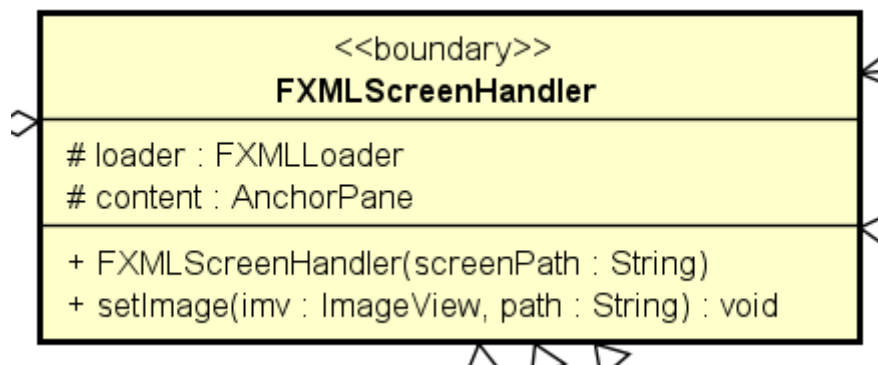
### Method

None

### State

None

### 3.16. Class “FXMLScreenHandler”



### Attribute

#	Name	Data type	Default value	Description
1	loader	FXMLLoader	new	
2	content	AnchorPane	NULL	main app anchor

### Operation

#	Name	Return type	Description
1	setImage	void	set image for screen

### Parameter:

- setImage
  - imv – image holder to be set
  - path – path to image

### Exception:

None

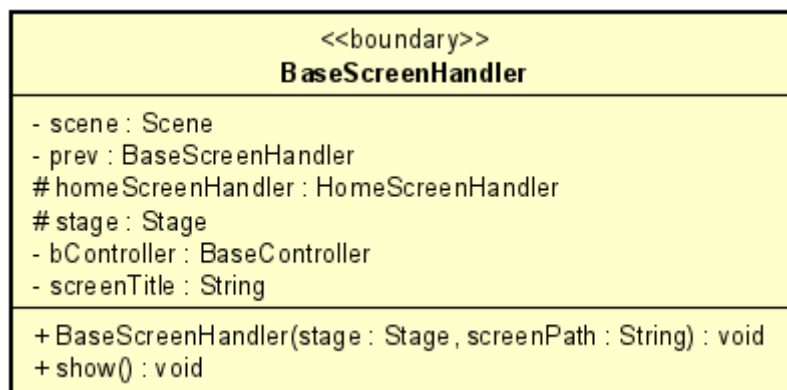
### Method

None

### State

None

## 3.17. Class “BaseScreenHandler”



### Attribute

#	Name	Data type	Default value	Description
1	scene	Scene	NULL	Scene to render
2	prev	BaseScreenHandler	NULL	Previous screen
3	homeScreenHandler	HomeScreenHandler	NULL	Home screen
4	stage	Stage	NULL	JavaFx stage
5	bController	BaseController	NULL	Controller for this screen

<b>6</b>	screenTitle	String	NULL	Title of this screen
----------	-------------	--------	------	----------------------

### Operation

#	Name	Return type	Description
1	show	void	Render the screen

### Parameter:

### Exception:

None

### Method

None

### State

None