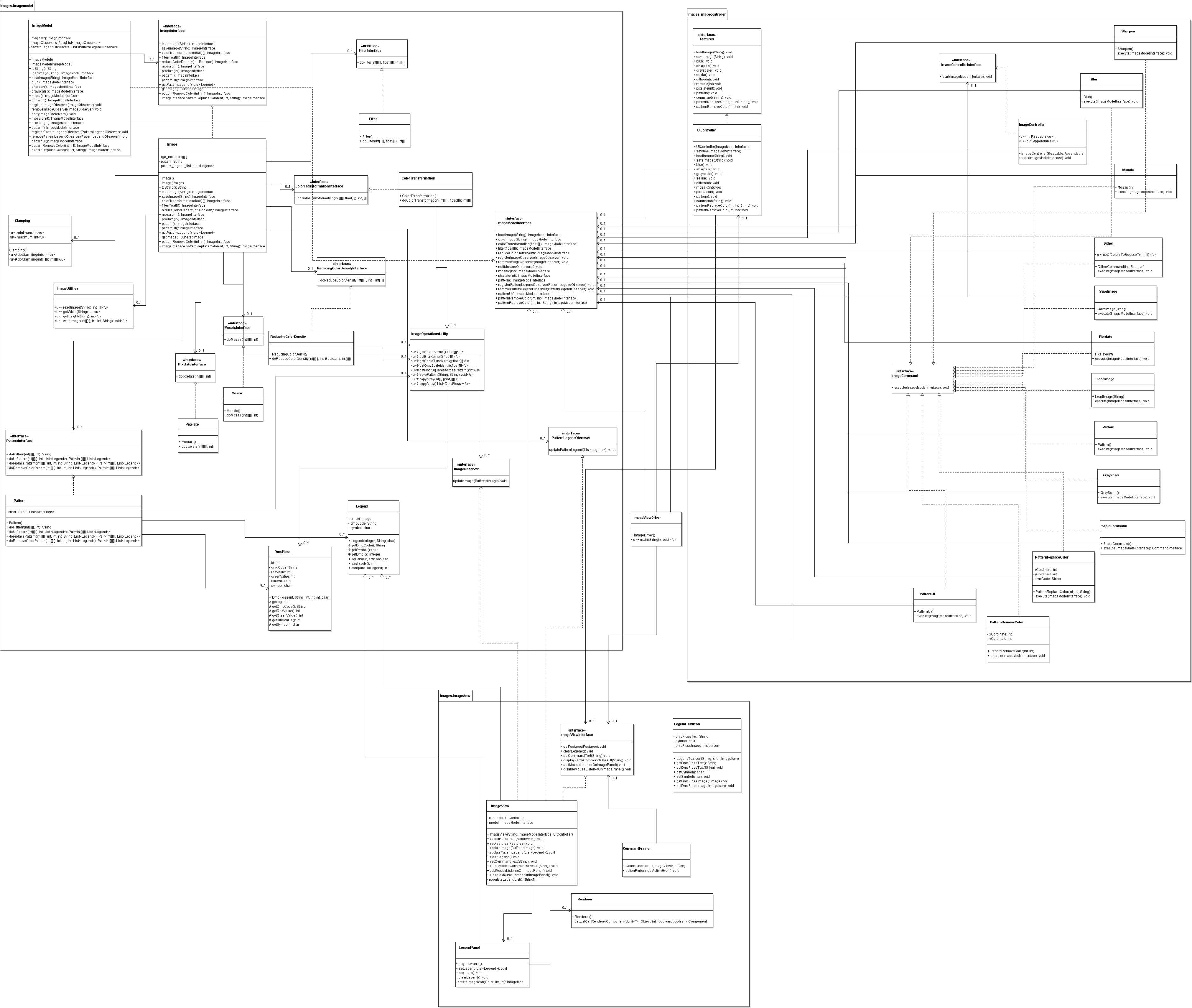
images.imagemodel
ImageModel
- imageObj: ImageInterface
- imageObservers: ArrayList<ImageObserver>
- patternLegendObservers: List<PatternLegendObserver>
+ ImageModel()
+ ImageModel(ImageModel)
+ toString(): String
+ loadImage(String): ImageModelInterface
+ saveImage(String): ImageModelInterface
+ blur(): ImageModelInterface
+ sharpen(): ImageModelInterface
+ grayscale(): ImageModelInterface
+ sepia(): ImageModelInterface
+ dither(int): ImageModelInterface
+ registerImageObserver(ImageObserver): void
+ removeImageObserver(ImageObserver): void
+ notifyImageObservers(): void
+ mosaic(int): ImageModelInterface
+ pixelate(int): ImageModelInterface
+ pattern(): ImageModelInterface
+ registerPatternLegendObserver(PatternLegendObserver): void
+ removePatternLegendObserver(PatternLegendObserver): void
+ patternUi(): ImageModelInterface
+ patternRemoveColor(int, int): ImageModelInterface
+ patternReplaceColor(int, int, String): ImageModelInterface
«interface»
ImageInterface
+ loadImage(String): ImageInterface
+ saveImage(String): ImageInterface
+ colorTransformation(float[][]): ImageInterface
+ filter(float[][]): ImageInterface
+ reduceColorDensity(int, Boolean): ImageInterface
+ mosaic(int): ImageInterface
+ pixelate(int): ImageInterface
+ pattern(): ImageInterface
+ patternUi(): ImageInterface
+ getPatternLegend(): List<Legend>
+ getImage(): BufferedImage
+ patternRemoveColor(int, int): ImageInterface
+ ImageInterface patternReplaceColor(int, int, String): ImageInterface
«interface»
FilterInterface
+ doFilter(int[][][], float[][]): int[][][]
Filter
+ Filter()
+ doFilter(int[][][], float[][]): int[][][]
Image
- rgb_buffer: int[][][]
- pattern: String
- pattern_legend_list: List<Legend>
+ Image()
+ Image(Image)
+ toString(): String
+ loadImage(String): ImageInterface
+ saveImage(String): ImageInterface
+ colorTransformation(float[][]): ImageInterface
+ filter(float[][]): ImageInterface
+ reduceColorDensity(int, Boolean): ImageInterface
+ mosaic(int): ImageInterface
+ pixelate(int): ImageInterface
+ pattern(): ImageInterface
+ patternUi(): ImageInterface
+ getPatternLegend(): List<Legend>
+ getImage(): BufferedImage
+ patternRemoveColor(int, int): ImageInterface
+ ImageInterface patternReplaceColor(int, int, String): ImageInterface
«interface»
ColorTransformationInterface
+ doColorTransformation(int[][][], float[][]): int[][][]
ColorTransformation
+ ColorTransformation()
+ doColorTransformation(int[][][], float[][]): int[][][]
Clamping
<u>- minimum: int</u>
<u>- maximum: int</u>
Clamping()
<u># doClamping(int): int</u>
<u># doClamping(int[][][]): int[][][]</u>
«interface»
ReducingColorDensityInterface
+ doReduceColorDensity(int[][][], int): int[][][]
ImageUtilities
<u>+ readImage(String): int[][][]</u>
<u>+ getWidth(String): int</u>
<u>+ getHeight(String): int</u>
<u>+ writeImage(int[][][], int, int, String): void</u>
«interface»
MosaicInterface
+ doMosaic(int[][][], int)
ReducingColorDensity
+ ReducingColorDensity
+ doReduceColorDensity(int[][][], int, Boolean): int[][][]
ImageOperationsUtility
<u># getSharpKernel():float[][]</u>
<u># getBlurKernel():float[][]</u>
<u># getSepiaToneMatrix():float[][]</u>
<u># getGrayScaleMatrix():float[][]</u>
<u># getNoofSquaresAcrossPattern():int</u>
<u># savePattern(String, String):void</u>
<u># copyArray(int[][][]):int[][][]</u>
<u># copyArray():List<DmcFloss></u>
«interface»
PixelateInterface
+ dopixelate(int[][][], int)
Mosaic
+ Mosaic()
+ doMosaic(int[][][], int)
Pixelate
+ Pixelate()
+ dopixelate(int[][][], int)
«interface»
ImageModelInterface
+ loadImage(String): ImageModelInterface
+ saveImage(String): ImageModelInterface
+ colorTransformation(float[][]): ImageModelInterface
+ filter(float[][]): ImageModelInterface
+ reduceColorDensity(int): ImageModelInterface
+ registerImageObserver(ImageObserver): void
+ removeImageObserver(ImageObserver): void
+ notifyImageObservers(): void
+ mosaic(int): ImageModelInterface
+ pixelate(int): ImageModelInterface
+ pattern(): ImageModelInterface
+ registerPatternLegendObserver(PatternLegendObserver): void
+ removePatternLegendObserver(PatternLegendObserver): void
+ patternUi(): ImageModelInterface
+ patternRemoveColor(int, int): ImageModelInterface
+ patternReplaceColor(int, int, String): ImageModelInterface
«interface»
PatternLegendObserver
updatePatternLegend(List<Legend>): void
«interface»
ImageObserver
updateImage(BufferedImage): void
«interface»
PatternInterface
+ doPattern(int[][][], int): String
+ doUIPattern(int[][][], int, List<Legend>): Pair<int[][][], List<Legend>>
+ doreplacePattern(int[][][], int, int, int, String, List<Legend>): Pair<int[][][], List<Legend>>
+ doRemoveColorPattern(int[][][], int, int, int, List<Legend>): Pair<int[][][], List<Legend>>
Pattern
- dmcDataSet: List<DmcFloss>
+ Pattern()
+ doPattern(int[][][], int): String
+ doUIPattern(int[][][], int, List<Legend>): Pair<int[][][], List<Legend>>
+ doreplacePattern(int[][][], int, int, int, String, List<Legend>): Pair<int[][][], List<Legend>>
+ doRemoveColorPattern(int[][][], int, int, int, List<Legend>): Pair<int[][][], List<Legend>>
Legend
- dmcId: Integer
- dmcCode: String
- symbol: char
+ Legend(Integer, String, char)
# getDmcCode(): String
# getSymbol():char
# getDmcId():Integer
+ equals(Object): boolean
+ hashcode(): int
+ compareTo(Legend): int
DmcFloss
- id: int
- dmcCode: String
- redValue: int
- greenValue: int
- blueValue:int
- symbol: char
+ DmcFloss(int, String, int, int, int, char)
# getId():int
# getDmcCode(): String
# getRedValue(): int
# getGreenValue(): int
# getBlueValue(): int
# getSymbol(): char
images.imagecontroller
«interface»
Features
+ loadImage(String): void
+ saveImage(String): void
+ blur(): void
+ sharpen(): void
+ grayscale(): void
+ sepia(): void
+ dither(int): void
+ mosaic(int): void
+ pixelate(int): void
+ pattern(): void
+ command(String): void
+ patternReplaceColor(int, int, String): void
+ patternRemoveColor(int, int): void
UIController
+ UIController(ImageModelInterface)
+ setView(ImageViewInterface)
+ loadImage(String): void
+ saveImage(String): void
+ blur(): void
+ sharpen(): void
+ grayscale(): void
+ sepia(): void
+ dither(int): void
+ mosaic(int): void
+ pixelate(int): void
+ pattern(): void
+ command(String): void
+ patternReplaceColor(int, int, String): void
+ patternRemoveColor(int, int): void
«interface»
ImageControllerInterface
+ start(ImageModelInterface): void
Sharpen
+ Sharpen()
+ execute(ImageModelInterface): void
Blur
+ Blur()
+ execute(ImageModelInterface): void
ImageController
<u>- in: Readable</u>
<u>- out: Appendable</u>
+ ImageController(Readable, Appendable)
+ start(ImageModelInterface): void
Mosaic
+ Mosaic(int)
+ execute(ImageModelInterface): void
Dither
<u>- noOfColorsToReduceTo: int[][]</u>
+ DitherCommand(int, Boolean)
+ execute(ImageModelInterface): void
SaveImage
+ SaveImage(String)
+ execute(ImageModelInterface): void
Pixelate
+ Pixelate(int)
+ execute(ImageModelInterface): void
«interface»
ImageCommand
+ execute(ImageModelInterface): void
LoadImage
+ LoadImage(String)
+ execute(ImageModelInterface): void
Pattern
+ Pattern()
+ execute(ImageModelInterface): void
GrayScale
+ GrayScale()
+ execute(ImageModelInterface): void
SepiaCommand
+ SepiaCommand()
+ execute(ImageModelInterface): CommandInterface
ImageViewDriver
+ ImageDriver()
<u>+ main(String[]): void </u>
PatternReplaceColor
- xCordinate: int
- yCordinate: int
- dmcCode: String
+ PatternReplaceColor(int, int, String)
+ execute(ImageModelInterface): void
PatternUI
+ PatternUI()
+ execute(ImageModelInterface): void
PatternRemoveColor
- xCordinate: int
- yCordinate: int
+ PatternRemoveColor(int, int)
+ execute(ImageModelInterface): void
images.imageview
«interface»
ImageViewInterface
+ setFeatures(Features): void
+ clearLegend(): void
+ setCommandText(String): void
+ displayBatchCommandsResult(String): void
+ addMouseListenerOnImagePanel():void
+ disableMouseListenerOnImagePanel(): void
LegendTextIcon
- dmcFlossText: String
- symbol: char
- dmcFlossImage: ImageIcon
+ LegendTextIcon(String, char, ImageIcon)
+ getDmcFlossText(): String
+ setDmcFlossText(String): void
+ getSymbol(): char
+ setSymbol(char): void
+ getDmcFlossImage():ImageIcon
+ setDmcFlossImage(ImageIcon): void
ImageView
- controller: UIController
- model: ImageModelInterface
+ ImageView(String, ImageModelInterface, UIController)
+ actionPerformed(ActionEvent): void
+ setFeatures(Features): void
+ updateImage(BufferedImage): void
+ updatePatternLegend(List<Legend>): void
+ clearLegend(): void
+ setCommandText(String): void
+ displayBatchCommandsResult(String): void
+ addMouseListenerOnImagePanel():void
+ disableMouseListenerOnImagePanel(): void
- populateLegendList(): String[]
CommandFrame
+ CommandFrame(ImageViewInterface)
+ actionPerformed(ActionEvent): void
Renderer
+ Renderer()
+ getListCellRendererComponent(JList<?>, Object, int , boolean, boolean): Component
LegendPanel
+ LegendPanel()
+ setLegend(List<Legend>): void
+ populate(): void
+ clearLegend(): void
- createImageIcon(Color, int, int): ImageIcon

Cross Stich Controller Testing Plan

S.No	Description	Test Case	Expected
1	Load the file by giving file name as "Test.png"and verify result by log.toString() and out.toString()	TestLoad()	Log.toString() = Filename: Test.png and UniqueCode: 12345 Out.toString() = load command is executed.\n
2	Save the image file by giving file name "Test.png" and verify result by log.toString() and out.toString()	TestSaveImage()	Log.toString() = Filename: Test.png and UniqueCode: 12346 Out.toString() = save command is executed.\n
3	Grayscale the image and verify result by log.toString() and out.toString()	TestGrayScale()	Log.toString() = UniqueCode: 12347 Out.toString() = grayscale command is executed.\n
4	Sepia the image and verify result by log.toString() and out.toString()	TestSepia()	Log.toString() = UniqueCode: 12348 Out.toString() = sepia command is executed.\n
5	Blur the image and verify result by log.toString() and out.toString()	TestBlur()	Log.toString() = UniqueCode: 12349 Out.toString() = blur command is executed.\n
6	Sharpen the image and verify result by log.toString() and out.toString()	TestSharpen()	Log.toString() = UniqueCode: 12350 Out.toString() = sharpen command is executed.\n
7	Dither the image by giving 2 values per channel and verify result by log.toString() and out.toString()	TestDither()	Log.toString() = noOfColorsToReduceTo: 2 and UniqueCode: 12351 Out.toString() = dither command is executed.\n
8	Mosaic the image by giving number of seeds as 6600 and verify result by log.toString() and out.toString()	TestMosaic()	Log.toString() = noOfSeeds: 6600 and UniqueCode: 12352 Out.toString() = mosaic command is executed.\n
9	Pixelate the image by	TestPixelate()	Log.toString() =

	giving number of squares across width as 30 and verify result by log.toString() and out.toString()		noOfSquaresAcross: 30 and UniqueCode: 12353 Out.toString() = pixelate command is executed.\n
10	Generate the cross-stitch pattern and verify result by log.toString() and out.toString()	TestPattern()	Log.toString() = UniqueCode: 12354 Out.toString() = pattern command is executed.\n
11	Save the cross-stitch pattern by giving file name as Test.txt and verify result by log.toString() and out.toString()	TestSavePattern()	Log.toString() = Filename: Test.txt and UniqueCode: 12355 Out.toString() = savePattern command is executed.\n
12	Give Invalid command as saveImage	TestInvalidCommand()	Should through IllegalArgumentException
13	Pass parameter in as null to controller object	TestNullIn()	Should through NullPointerException
14	Pass parameter out as null to controller object	TestNullOut()	Should through NullPointerException
15	Pass parameter model as null to start() of controller object	TestNullModel()	Should through NullPointerException
16	Generate the cross-stitch pattern for UI and verify result by log.toString() and out.toString()	TestUIPattern()	Log.toString() = UniqueCode: 12356 Out.toString() = uipattern command is executed.\n
17	Generate the cross-stitch pattern after replacing the color and verify result by log.toString() and out.toString()	TestPatternReplaceColor()	Log.toString() = UniqueCode: 12357 Out.toString() = uipattern command is executed.\n
18	Generate the cross-stitch pattern after removing the color and verify result by log.toString() and	TestPatternRemoveColor()	Log.toString() = UniqueCode: 12358 Out.toString() = uipattern command is executed.\n

	out.toString()		
19	Execute batch commands and verify result by log.toString() and out.toString()	TestBatchCommands()	Log.toString() = UniqueCode: 12359 Out.toString() = uipattern command is executed.\n