Milestone 5 – UNO Card Game

Kelly Lamb | Professor Brandon Bass | Grand Canyon University: CST-105 Computer Programming I

Problem / Question

Create a graphic representation of a game of UNO that can play the entire game – all 108 cards and determine winner – Use JavaFX

Hypothesis

- Create java classes to handle data model, logic, and visual representations of the UNO card game. Utilize advanced techniques of Java and JavaFX to produce the end result.
- Design processes incrementally – confirming results in stages

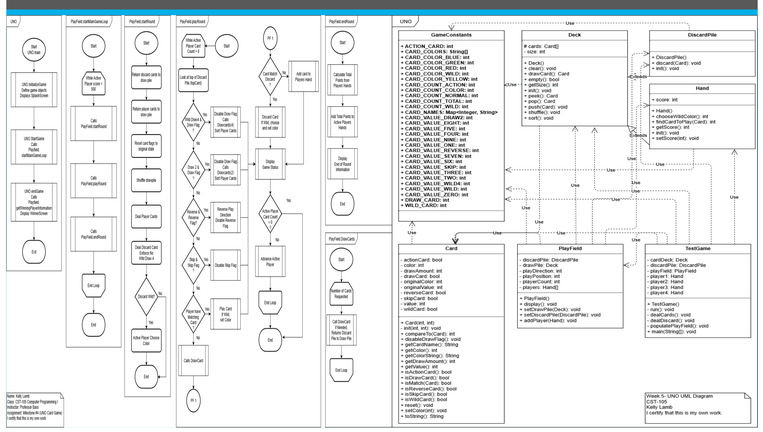
Project Overview

1. Develop game logic and flesh it out in a story board
2. Design Abstract Data Types (ADT) to model data and logic
3. Create UML documents and define dependencies / relationships
4. Create Flowcharts (2), initial logic; updated version with classes
5. Write Java code – confirm that it follows flowchart logic
6. Create videos to demonstrate effective progress in each step
7. Modify code to final stage to produce visually appealing game
8. Modify story board to more accurately represent final product
9. Produce a poster that highlights the process

Classes / Research

Initial Classes	Additional Classes	Visual Classes
Card	DiscardPile	AbstractView
Deck	DrawPile	CardView
Hand	UNO	DiscardPileView
GameConstants		DrawPileView
		HandView
		UNOImageBox

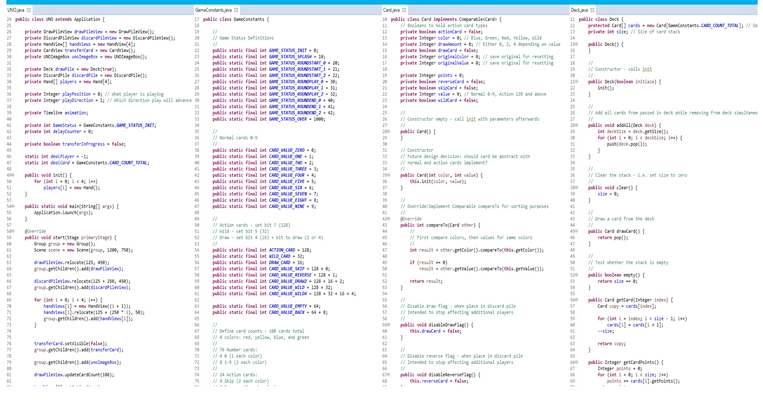
Flow Chart / UML



Procedure - Milestones

Step 1	Step 2	Step 3	Step 4
			
Design initial logic flowchart and story board	Design Data Model With ADT and UML	Incremental Design, Write, Test, Research JavaFX	Create Visual classes and merge with data model

Base Code Snippets



Results - Screenshots

- Game Initializes, plays, and determines a winner (Success)
- Screenshots: 1.Title; 2. Play; 3. UNO Called; 4. Winner Screen
- Number of classes: 14 (including anonymous classes)
- Number of lines of code: 1399

Conclusion

- UNO – Surprisingly fun and easy game to play but provides some tough logic requirements and decisions
- Flowchart, ADT, UML, and Story board are great tools to help in the design process
- JavaFX offers a robust graphics programming platform with animation but requires significant research and understanding
- Game results support hypothesis; game completes with a winner!

Works Cited

- docs.oracle.com/javafx/2/get_started/jfxpub-get_started.htm
- docs.oracle.com/javase/tutorial/uiswing/misc/splashscreen.html
- examples.javacodegeeks.com/desktop-java/javafx/javafx-transition-example/
- Introduction-to-Java-Programming-10th-ed.-Comprehensive-Version-Liang-2014-01-06.pdf
- tutorials.jenkov.com/javafx/index.html
- www.genuinecoder.com/javafx-splash-screen-loading-screen/
- www.javacodegeeks.com/javafx-tutorials
- www.javatpoint.com/javafx-tutorial
- www.tutorialspoint.com/javafx/index.htm