

Euchre Project

Josh Techentin

Jarret Swales

Derrick Fleming

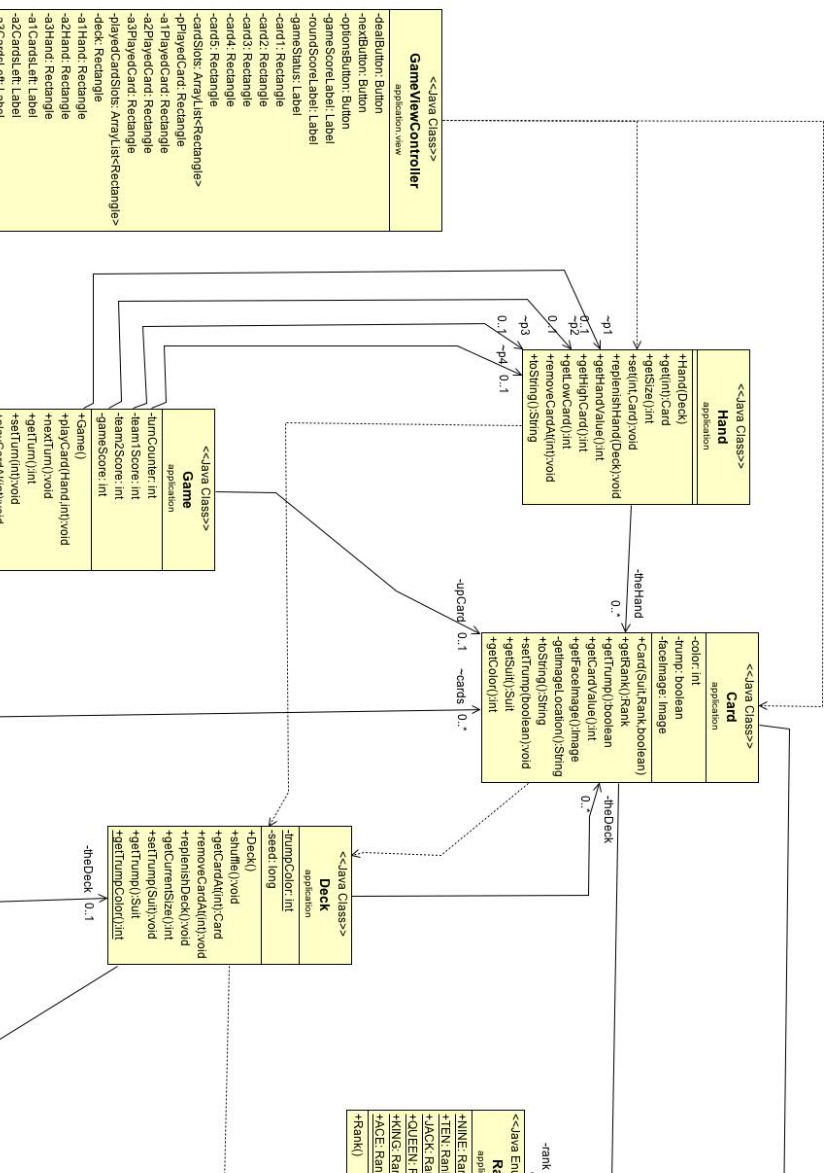
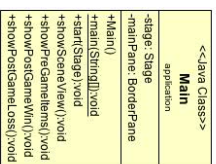
An application that allows people to play a digital game of Euchre

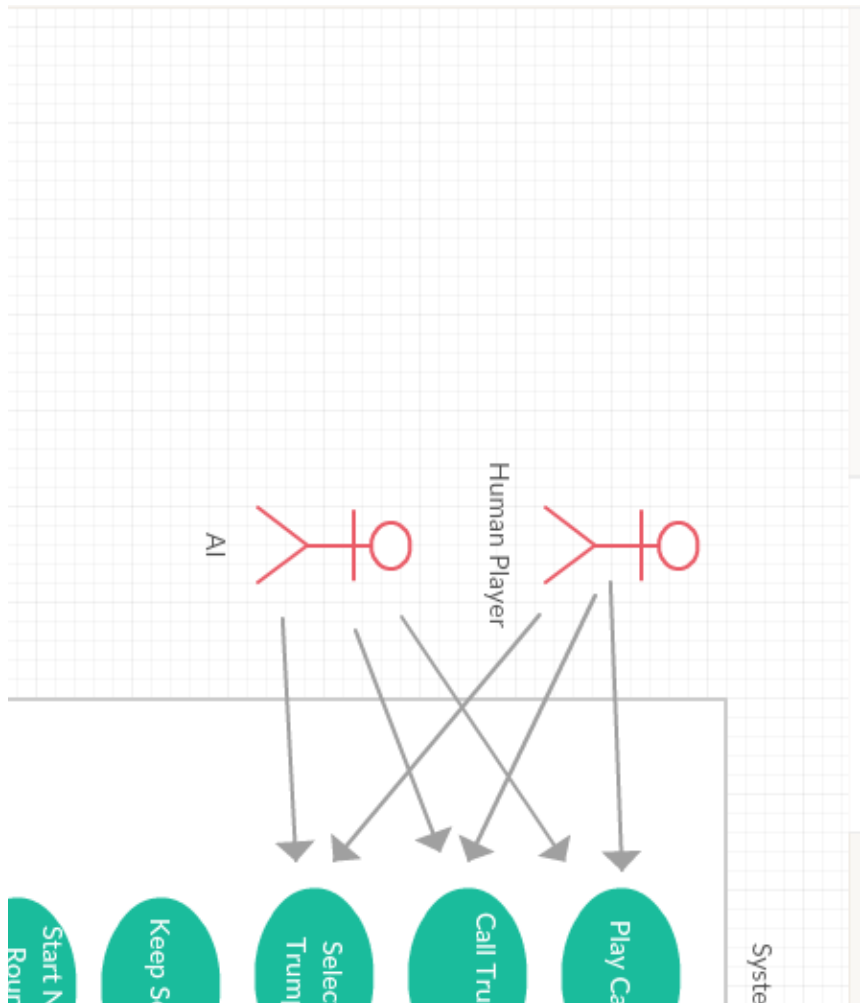
Feature List

Allows humans to play a single player game of Euchre

Face computer players as opponents with AI

Actual graphics, not text-based





Name	Play Card
ID	UC1
Brief Description	Allows the Human Player or AI to play a card
Actors (primary and supporting/secondary)	Human player and AI are primary actors Game interface is supporting
Triggers	For humans, when they press the card button For AI, when it is their turn
Preconditions	It is the human or AI's turn, trump is determined, and the player (if human) did not push a non-existent card button
Primary Flow	<ol style="list-style-type: none"> 1. Allow player to select card (if AI) 2. Add selected card to the center of the table 3. Remove the card from the player's hand 4. Goes to the next turn
Alternate Flows	1A. The human tries to play a card that is non-existent 1A.1 The use case ends without altering anything
Minimal Guarantees	The case ended with no changes to the program.
Success Guarantees	The card selected was added to the center, removed from the player's hand, and the program moved on to the next turn.

Name	Call Trump
ID	UC2
Brief Description	Allows players to call or pass on the given trump
Actors (primary and supporting/secondary)	Primary actors are Human Player and AI Secondary actor is Game interface
Triggers	Human pushes the pass button or the given trump card button or it is AI's turn
Preconditions	It is the given player's turn and the initial card has not been called or passed
Primary Flow	<ol style="list-style-type: none"> 1. The game notes whether the player calls trump or not. 2. The game makes the card trump 3. The game announces to everyone which trump has been called 4. Goes to the next turn
Alternate Flows	1A. The player passes on the trump card 1A.1 It goes to the next player's turn immediately
Minimal Guarantees	The use case ended without altering the program in any way
Success Guarantees	The player (human or AI) either called trump or passed it to the next player

Name	Choose Trump
ID	UC3
Brief Description	Allows players to choose which Suit they want to be trump
Actors (primary and supporting/secondary)	Human player and AI are primary actors Game interface is the secondary actor
Triggers	Human player pushes card button or pass button or it is AI's turn
Preconditions	Must be the player's turn, trump must not have been called yet, and the trump card must have been passed off of the table
Primary Flow	<ol style="list-style-type: none"> 1. Game interface registers the player's choice of trump 2. The new trump is announced 3. The game goes to the next turn
Alternate Flows	<p>1A Player decides to pass on trump</p> <p>1A.1 It goes to the next turn</p> <p>2A Last player passes on choosing trump</p> <p>2A.1 A random card is drawn and that is made trump</p> <p>2A.2 The trump is announced</p> <p>2A.3 The game goes to the next turn</p>
Minimal Guarantees	The use case ended without altering the program in any way
Success Guarantees	Player successfully made a card of their choice trump or passed it off to the next player. The next turn triggered

Name	Start New Round
ID	UC4
Brief Description	The game interface starts a new round in the game
Actors (primary and supporting/secondary)	The game interface is the primary actor There are no secondary actors
Triggers	New game has been started or a team is assigned point(s) from the previous round
Preconditions	The team who won the previous round has not won yet
Primary Flow	<ol style="list-style-type: none"> 1. Potential points each team can win are reset to 2 2. The deck of cards is reformed and shuffled 3. Each player gets a new hand of cards from the deck 4. A potential trump card is drawn and put into the center of the table
Alternate Flows	NA
Minimal Guarantees	The use case ends with no changes to the program
Success Guarantees	A new round started

Name	Keep Score
ID	UC5
Brief Description	The game interface keeps track and updates the players' scores
Actors (primary and supporting/secondary)	Game interface is the primary actor No supporting actors
Triggers	The round ends
Preconditions	NA
Primary Flow	<ol style="list-style-type: none"> 1. Determine which team won the round 2. Add the team's potential points to their score 3. Start a new round
Alternate Flows	1A The team who won the round now has 10 points or more 1A.1 The game ends 1A.2 The interface announces which team won 1A.3 The program returns to the main menu
Minimal Guarantees	The use case ended with no changes to the program
Success Guarantees	The team who won the round's score was updated

Git repository: <https://github.com/derrikf2/cs350Project>

Git log:

Commits on Oct 20, 2016

Merge branch 'master' of
git@github.com:derrikf2/cs350Project.git
derrikf2 committed 9 hours ago
javadocs, checkstyle near
completion

derrikf2 committed 23 hours ago

Commits on Oct 19, 2016

Merge remote-tracking branch
'origin/master'
swalesj committed 23 hours ago
Trump selection and scoring
complete, button disabling added.
swalesj committed a day ago
nothing

derrikf2 committed a day ago

javadocs/checkstyle complete.

derrikf2 committed a day ago

javadoc/checkstyle complete (less
magic number errors)

derrikf2 committed a day ago

javadoc/checkstyle complete (less
magic number errors)

derrikf2 committed a day ago

javadoc/checkstyle complete

derrikf2 committed a day ago

javadoc complete

derrikf2 committed a day ago

Hopefully fixes checkstyle

joshtechentin96 committed a day
ago

Update GameController.java

swalesj committed on **GitHub** 2
days ago

Merge remote-tracking branch
'origin/master'

swalesj committed 2 days ago

Game is 1-round functional!

swalesj committed 2 days ago

small fix in refresh to clear out image
before attempting to set fill.

derrikf2 committed 2 days ago

add items to GameView.fxml, import
to controller.

derrikf2 committed 2 days ago

added refresh(), null check for
images.

derrikf2 committed 2 days ago

Commits on Oct 18, 2016

Player one hand display
functionality; phasing out Player
class

swalesj committed 2 days ago

PlayCard now removes the card
from the player's hand.

joshtechentin96 committed 2 days
ago

Added methods for all of the button
pushes in the game class (excluding
joshtechentin96 committed 2 days
ago

update

derrikf2 committed 2 days ago

Commits on Oct 15, 2016

nothing significant

derrikf2 committed 5 days ago

clone and sync

derrikf2 committed 5 days ago

Fixed a ton of checkstyle errors.

joshtechentin96 committed 5 days
ago

package info

derrikf2 committed 5 days ago

CS errors (public to private)

accessors, created .java for
Suit/Rank

derrikf2 committed 5 days ago
uml

derrikf2 committed 6 days ago

Commits on Oct 14, 2016

Delete PreGameController.java

derrikf2 committed on **GitHub** 6
days ago

Delete PreGameItems.fxml

derrikf2 committed on **GitHub** 6
days ago

commented game loop, added
getters/setters for Game attrib's

derrikf2 committed 6 days ago

Commits on Oct 10, 2016
fixed path to Image cardFace
derrikf2 committed 10 days ago

Commits on Oct 9, 2016
Continuing GUI, added Image
facelImage field to Card.java
derrikf2 committed 11 days ago
move images, settle
PreGameItems.fxml discrepancy.
derrikf2 committed 11 days ago

Commits on Oct 8, 2016
Added getSize() and
removeCardAt() functions to the
Hand class.
joshtechentin96 committed 12 days
ago

Commits on Oct 5, 2016
Fixed the tests so they all pass.
joshtechentin96 committed 15 days
ago
Added test cases for the Card, Deck,
Hand, and Player classes. Added
joshtechentin96 committed 15 days
ago
Added test cases for Card, Hand,
Deck, and Player classes.
joshtechentin96 committed 15 days
ago

Commits on Oct 3, 2016
update
derrikf2 committed 17 days ago

Commits on Sep 29, 2016
nothing major.
derrikf2 committed 22 days ago
Furthering preparations of view sub-
package.
derrikf2 committed 22 days ago
Add Windows paths where
necessary. Untested, currently
commented.
derrikf2 committed 22 days ago
Continued progress on CheckStyle
errors
derrikf2 committed 22 days ago
Created sub-package 'view' to hold
all things GUI.
derrikf2 committed 22 days ago

Commits on Sep 24, 2016
Merge branch 'master' of
github.com:derrikf2/cs350Project

derrikf2 committed 26 days ago

Commits on Sep 23, 2016
Checkstyle comments started
derrikf2 committed 27 days ago

Commits on Sep 19, 2016
Added a nice skeleton to the AI,
Player, and Game classes
joshtechentin96 committed on Sep
19

Commits on Sep 12, 2016
Added card images
application.images.<suit-type>
derrikf2 committed on Sep 12
started assembling GUI via Scene
Builder
derrikf2 committed on Sep 12

Commits on Sep 11, 2016
JavaDocs comments, not complete
derrikf2 committed on Sep 11
JavaDocs comments, not complete
derrikf2 committed on Sep 11
Merge branch 'master' of
git@github.com:derrikf2/cs350Project.git
derrikf2 committed on Sep 11
test EGit functionality
derrikf2 committed on Sep 11
test EGit functionality
derrikf2 committed on Sep 11
Delete Player.java
derrikf2 committed on **GitHub** on
Sep 11
Merge branch 'master' of
github.com:derrikf2/cs350Project
derrikf2 committed on Sep 11
Forgot to add Josh's Player class
derrikf2 committed on Sep 11
Delete Main.java
derrikf2 committed on **GitHub** on
Sep 11
Delete Hand.java
derrikf2 committed on **GitHub** on
Sep 11
Delete Game.java
derrikf2 committed on **GitHub** on
Sep 11
Delete Deck.java
derrikf2 committed on **GitHub** on
Sep 11
Delete Round.java
derrikf2 committed on **GitHub** on
Sep 11

Delete Card.java derrikf2 committed on GitHub on Sep 11 Delete AI.java derrikf2 committed on GitHub on Sep 11 josh added: AI Round Game derrikf2 committed on Sep 11 Delete Player.java derrikf2 committed on GitHub on Sep 11 Delete AI.java derrikf2 committed on GitHub on Sep 11 josh added: AI Game Round classes derrikf2 committed on Sep 11 Delete EuchrePanel.java derrikf2 committed on GitHub on Sep 11 Delete EuchreGUI.java derrikf2 committed on GitHub on Sep 11 Add files via upload joshtechentin96 committed on GitHub on Sep 11 Commits on Sep 11, 2016 getting josh current src derrikf2 committed on Sep 11 Added EuchreGUI.java and EuchrePanel.java derrikf2 committed on Sep 11 Added EuchreGUI.java and EuchrePanel.java derrikf2 committed on Sep 11 Added EuchreGUI.java and EuchrePanel.java derrikf2 committed on Sep 11 comments added to Hand.java derrikf2 committed on Sep 11	Changed player, to allow AI to extend. derrikf2 committed on Sep 11 Player class created. derrikf2 committed on Sep 11 Merge branch 'master' of github.com:derrikf2/cs350Project derrikf2 committed on Sep 11 Player class added. derrikf2 committed on Sep 11 Commits on Sep 10, 2016 Delete of manually uploaded files swalesj committed on Sep 10 Merge branch 'master' of https://github.com/derrikf2/cs350Project swalesj committed on Sep 10 Update 2 swalesj committed on Sep 10 Commits on Sep 9, 2016 Upload of semi-functional Deck and Card structure swalesj committed on GitHub on Sep 9 Merge remote-tracking branch 'origin/master' swalesj committed on Sep 9 Names and email added swalesj committed on Sep 9 Commits on Sep 8, 2016 Create README.md derrikf2 committed on GitHub on Sep 8 Delete README.md derrikf2 committed on GitHub on Sep 8 Commits on Sep 3, 2016 first commit derrikf2 committed on Sep 3
---	--

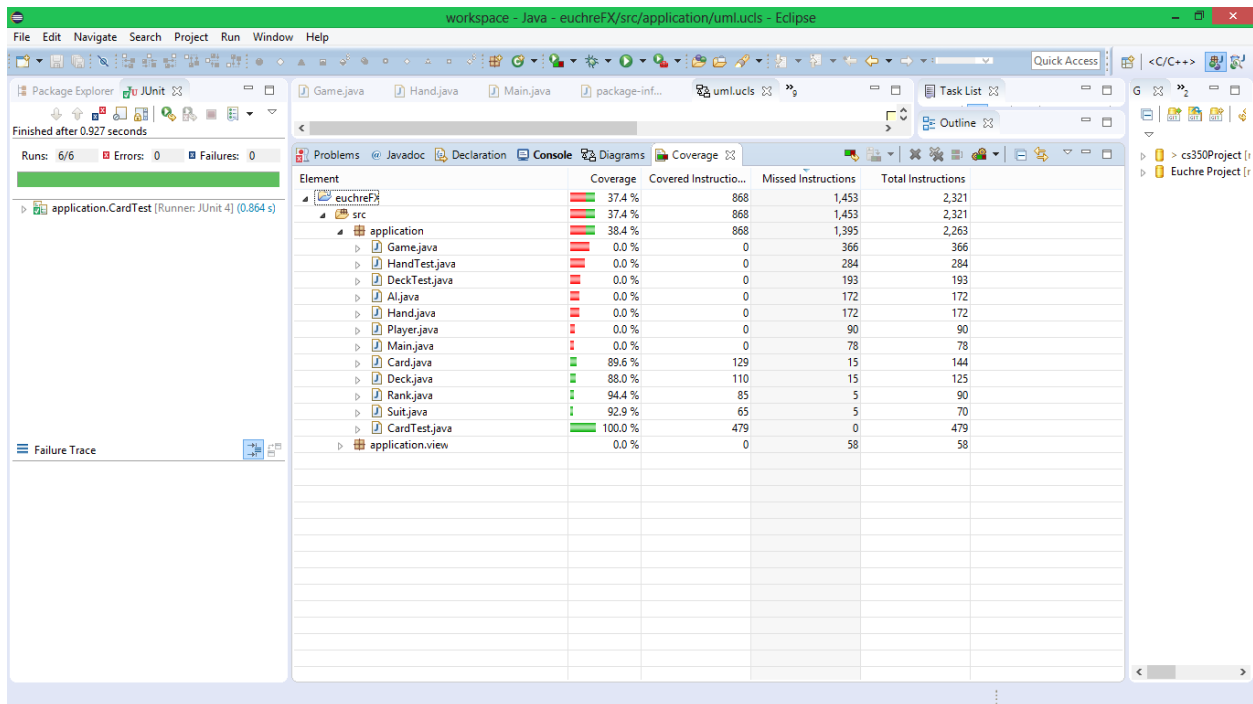
Checkstyle Error Report:

ProblemsJavadocDeclarationSearchConsoleGit StagingError LogCheckstyle violations xCoverage

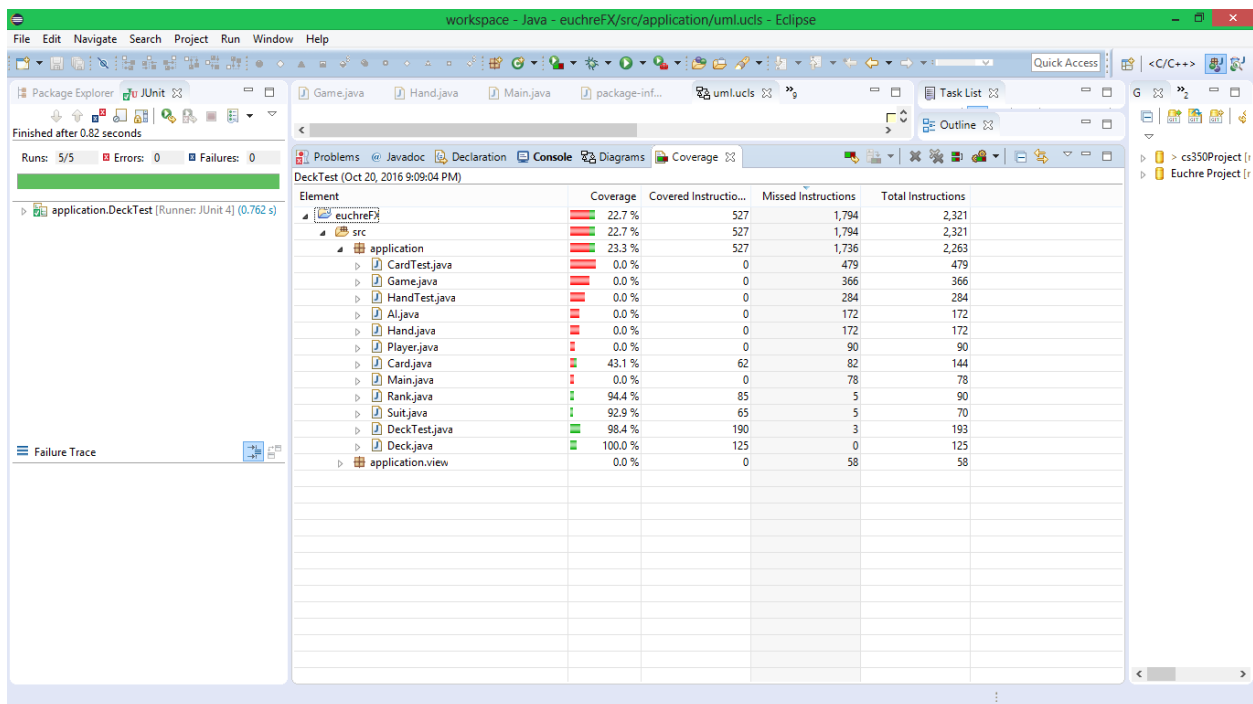
Overview of Checkstyle violations - 0 markers in 0 categories (Filter matched 0 of 0 items)

Checkstyle violation type	Marker count
---------------------------	--------------

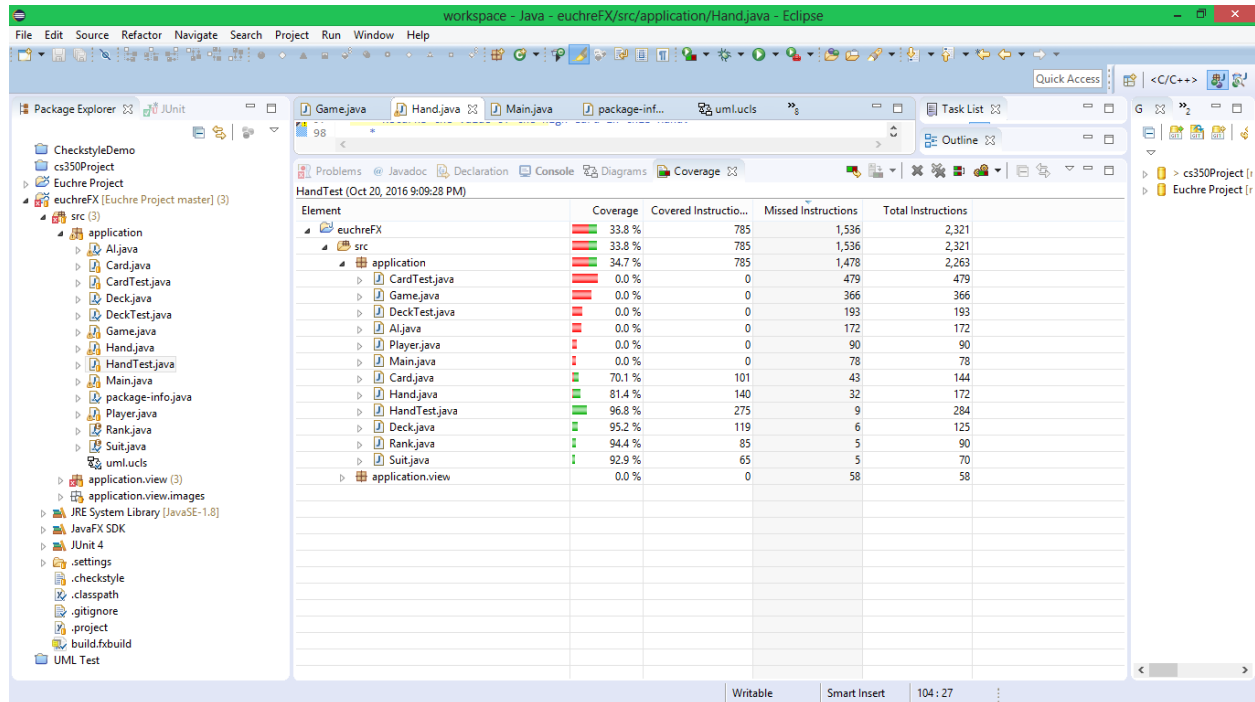
CardTest code coverage



DeckTest code coverage



HandTest code coverage



Team member responsibilities

Josh Techentin: JUnit testing/EclEmma coverage, creating Player and AI classes (which in the end got merged into the Game class), made main skeleton of Game class, made use case diagram and use case descriptions, helped with removing Checkstyle errors and FindBug bugs

Jarret Swales: I created the underlying objects and structure for the game, including Deck, Player (now obsolete), Card, and Hand. Once Derrik finished the GUI, I built the system for selecting trump and continuing gameplay accordingly.

Derrik Fleming: I was responsible for a large part of the JavaFX UI, and how it interfaces with the Game class. Additionally I contributed some extra methods to Game, Card, and Deck (as well as the former Player class) outside of the groundwork that both Jarret and Josh laid out; along with Javadocs, Checkstyle violations, UML, identification/fixing of bugs that arose along the way, handling unchecked scenarios, and removing/consolidating unnecessary code. Jarret and I had to collaborate on restructuring a lot of the original code mid-way through the development process in an attempt to achieve a loosely based MVC design for the application, and sensibly limiting the amount of relationships between the different classes used that were present in previously written code.

Self reflections:

Josh Techentin:

I had the most problems getting eclipse and EGit up and running. It took me a week or two before I could get them both working properly. Despite this slow start, I was able to contribute a lot to this project. Not only did I create the JUnits tests and use case diagram/descriptions, but I also created the Player, AI, and a large chunk of the Game class. I

also was the one who came up with the idea for Euchre being the project we worked on. I believe I pulled my own weight at the very least, and contributed quite a bit to the project.

Jarret Swales:

This was my first opportunity to build an application using JavaFX; and consequently I will probably never use Swing again! The case being that none of us have used FX before, I am happy with the result. Learning this new platform had a significant impact on our ability to plan the structure of the project, which resulted in some wasted effort but has proven to be worth the expense. I am content with what we have accomplished, but would like to see a fully refined and featured product for release two. Each team member has contributed significant value to the project.

Derrik Fleming:

Despite the fact that none of us had any experience with JavaFX prior to starting the project I am very satisfied in the way it turned out appearance-wise. It took a lot of time reading/researching design to determine the best way to approach the structure of the project (in regards to the GUI) with the features we wanted to include and the code that we had already written in mind. I'm just happy to have gotten the opportunity to be able to work with an updated GUI package because Swing just isn't very pretty. I'll admit we could have been a little more organized heading into the project and were a little over-zealous from the start. We ended up trimming quite a bit of fat in the last half of the development period; but in our defense this was partially due to us not really having a concrete idea/understanding of the project design as a whole from the beginning. We do have our work pretty well cut out for us for Release 2 though. And of course, what would any project be without issues; there were a few times despite being familiar with both EGit and GitHub that I had issues, primarily due to us working in the same

classes and overwriting each other, but I will say by the end of the development period we had the delegation of tasks straightened out, and communication improved drastically.