1. Read in data from files
2. Input user names (2 players)
3. Play game
4. Display position and details after each turn/action
5. Output winner

Step 1: Read Data file

1.1 Open PropertyCost.csv for reading

1.2 Repeat 40 times

1.3 Read Row

1.4 End Repeat

1.5 Close PropertyCost.csv

1.6 Open PropertyRent.csv for reading

1.7 Repeat 40 times

1.8 Read Row

1.9 End Repeat

1.10 Close PropertyRent.csv

1.11 Open Cards.csv for reading

1.12 Repeat 16 times

1.13 Read Row

1.14 End Repeat

1.15 Close Cards.csv

Step 2: Input User names

2.1 Display message prompting for player 1’s name

2.2 name1 equals name inputted

2.3 While name1 contains anything other than letters

2.4 Error is displayed and user is prompted for another name to be assigned to name1

2.5 End while

2.6 Display message prompting for player 2’s name

2.7 name2 equals name inputted

2.8 While name2 contains anything other than letters

2.9 Error is displayed and user is prompted for another name

2.10 End while

Step 3 Play Game

ALL REPEATED UNTIL ONE PLAYER GOES BANKRUPT {

3.1 Display user interface

3.2 User clicks roll dice button

3.3 Two random numbers are generated between 1-6

3.4 Values are added together

3.5 activePlayer is moved forward by the value of this number

3.6 If activePlayer position is greater than 39

3.7 run passGo method

3.8 End If

3.9 If activePlayer position is on a property

3.10 If property is owned by other player

3.11 disable buy property button

3.12 End If

3.13 prompt user to buy property

3.14 Else If activePlayer position is on a community chest and chance tile

3.15 disable buy property button

3.16 run takeCard method

3.17 Else If activePlayer position is on electric company or waters works

3.18 enable buy property button

3.19 run scanUtilities method

3.20 Else If activePlayer position is on a jail tile

3.21 disable buy property button

3.22 If activePlayer position is on “Just Visiting”

3.23 display refreshed user interface

3.24 Else

3.25 display message to tell user they must go to jail

3.25 change activePlayer position to JAIL

3.25 run jailProcedure

3.28 End If

3.29 Else If activePlayer position is on a super tax or income tax

3.30 disable buy property button

3.31 If activePlayer position is on income tax

3.32 display message to tell user they have been charged £200 income tax

3.33 deduct 200 from activePlayer balance

3.34 add 200 to free parking total

3.35 run the checkBankrupt method

3.36 Else

3.37 display message to tell user they have been charged £200 income tax

3.38 deduct 200 from activePlayer balance

3.39 add 200 to free parking total

3.40 run the checkBankrupt method

3.41 End If

3.42 Else If activePlayer position is on free parking

3.43 disable buy property button

3.44 run freeParking method

3.45 Else If active user is on GO

3.46 disable buy property button

3.47 display user interface

3.48 End If

3.49 display user interface

3.50 User clicks end turn button

3.51 If activePlayer equals 0

3.52 set activePlayer to 1

3.53 Else

3.54 set activePlayer to 0

3.55 End If

3.56 Set turnC to true to active turn

}

Notes for Step 4

UI = user interface

\*˅

/\*\*All variables are used to construct the user interface

\* activePlayer is the 0 if it is player 1’s turn and 1 if it’s player 2’s turn

\* die1 is the value for the first dice and is used to select the image

\* die2 is the value for the second dice and is used to select the image

\* name1 is player 1’s name and is used when it is their turn

\* name2 is player 2’s name and is used when it is their turn

\* position1 is player 1’s position and is used each time the UI is refreshed

\* position2 is player 2’s position and is used each time the UI is refreshed

\* onProperty is used to communicate if the activePlayer is on a property and used to enable and disable the buy property button

\* turnC is used to communicate if the user can roll the dice, e.g. if they have just started their turn or have rolled a double

\* owned is used to communicate if the property is owned by the activePlayer so that the UI knows the display the buy houses/hotel button  
\*/

Step 4: Display position and details after each turn/action

4.1 position1 = player 1’s position

4.2 position2 = player 2’s position

4.3 balance1 = player 1’s balance

4.4 balance2 = player 2’s position

4.5 Create new JFrame object passing variables (activePlayer, die1, die2, name1, name2 , position1, position2, balance1, balance2, onProperty turnC, owned)\*

4.6 set frame to visible

Step 5: Output Winner (checkBankrupt method)

5.1 If activePlayer balance is below 0

5.2 Assign trophy image to variable

5.3 If activePlayer equals 0

5.4 Display that player 1 has lost the game

5.5 Display that player 2 has won with trophy image

5.6 Else

5.7 Display that player 2 has lost the game

5.5 Display that player 1 has won the game with trophy image

5.9 End If

5.10 End If