Methods, Attributes and Funcitonalities of All Files

The Bowling Management System codebase has a collection of a total of 29 files. Each file has a collection of classes and funcitons that help simulate the entire game.

Here is a list of all the files and their corresponding characteristics:

SNo.	File Name	Methods	Attribute	Major Functionalities	Interlinked Classes
1.	AddPartyView	• void actionPerformed() • void valueChanged() • Vector getParty() • Vector getNames() • void updateNewPatron()	• Vector party • Vector bowlerdb • ControlDeskView controlDesk • String selectedNick • String selectedMember	• Adding a new patron to party • Removing a patron from a party • Creating a new patron • Finished party selection • Returning the latest state of the party	• NewPatronVie
2.	AddPartyView	• ControlDesk getControlDesk()	• ControlDesk controldesk	• Return Current state of ControlDesk	• ControlDesk
3.	Bowler	• String getNickName() • String getFullName ( ) • String getNick ( ) • String getEmail ( )	• String fullName • String nickName • String email	• Getter functions • Validation of the bolwer	NIL
4.	BowlerFile	• static Bowler getBowlerInfo(String nickName) • static void putBowlerInfo(String nickName,String fullName,String email) • static Vector getBowlers()	• static String BOWLER_DAT	• Adding a new bowler • Getting details of one bowler • Getting details of all bowlers	NIL
5.	ControlDesk	• void run() • Bowler registerPatron(String nickName) • void assignLane() • void addPartyQueue(Vector partyNicks) • Vector getPartyQueue() • int getNumLanes() • void publish(ControlDeskEvent event) • HashSet getLanes()	• HashSet lanes • Queue partyQueue • int numLanes • Vector subscribers	• Setter and Getter functions • Broadcast an event to subscribing objects. • Creating a new patron • Finished party selection • Returning party names to be displayed in the GUI representation of the wait queue. • Main loop for ControlDesk's thread • Registering a Patron • Assigning a lane	• Lane

SNo.	File Name	Methods	Attribute	Major Functionalities	Interlinked Classes
6.	ControlDeskEvent	• Vector getPartyQueue()	• Vector partyQueue	• Returns a vector of the names of the parties in the waiting queue	
7.	ControlDeskObserver	• void receiveControlDeskEvent	NIL	• Interface for classes that observe control desk events.	• --------
8.	ControlDeskView	• void actionPerformed(ActionEvent e) • void updateAddParty(AddPartyView addPartyView) • void receiveControlDeskEvent(ControlDeskEvent ce)	• int maxMembers • ControlDesk controlDesk	• Display the GUI for the control desk • Handler for actionEvents • Receive a new party from andPartyView • Receive a broadcast from a ControlDesk	• ControlDesk • AddPartyView
9.	drive	• static void main()	• int numLanes • int maxPatronsPerParty	• Driver class for the entire game • Creates and alley with numLanes number of lanes • Activates the control desk object • Render the GUI for the control desk via ControlDeskView	• ControlDesk • Alley • ControlDeskVie
10.	EndGamePrompt	• EndGamePrompt(String partyName) • void actionPerformed(ActionEvent e) • int getResult() • void distroy()	• int result • String selectedNick • String selectedMember	• Displaying the end promt • Destroying the currently active game object.	• --------
11.	EndGameReport	• EndGameReport(String partyName, Party party) • void actionPerformed(ActionEvent e) • Vecotr getResult() • void distroy() • static void main(String args[]) • void valueChanged(ListSelectionEvent e)	• int result • String selectedMember	• Displaying the end game repor • Destroying the currently active game object.	• --------

SNo.	File Name	Methods	Attribute	Major Functionalities	Interlinked Classes
12.	Lane	• void run() • void receivePinsetterEvent(PinsetterEvent pe) • void receivePinsetterEvent(PinsetterEvent pe) • void resetScores() • void assignParty(Party theParty) • void markScore(Bowler Cur, int frame, int ball, int score) • LaneEvent lanePublish( ) • void publish(LaneEvent event) • Setter and Getter functions	• Party party • Pinsetter setter • HashMap scores • Vector subscribers • boolean gameIsHalted • boolean partyAssigned • private boolean gameFinished; • Iterator bowlerIterator • int ball • int bowlIndex • int frameNumber • boolean tenthFreameStrike • intp[curScores • int[][] cumulScores • boolean canThrowAgain • int [][] finalScroes • int gameNumber • Bowler currentThrowe	• Simulates the bowling alley lanes in the game • Ensures cylic rounds of each bowlers turn • assigns a party to the lane • Keeps track and calculates bowlers score	• Bowler • Party • Pinsetter
13.	LaneEvent	• Setters and getter funcitons only	• Party p • int frame • int ball • Bowler bowler • boolean mechProb	• Setter and getter functions for all lane functionalities	• Party • Bowler
14.	LaneEventInterface	• An interface class	• ------	• Interfaces the multiple classes	• Party • Bowler
15.	LaneObserver	• An interface class	• ------	• Interfaces the multiple classes	• ------
16.	LaneServer	• An interface class	• ------	• Interfaces the multiple classes	• -------
17.	LaneStatusView	• LaneStatusView(Lane lane, int laneNum) • JPanel showLane() • void receiveLaneEvent(LaneEvent le) • void receivePinsetterEvent(PinsetterEvent pe)	• PinSetterView psv • LaneView lv • Lane lane • int laneNum • boolean laneShowing • booleean psShowing	• Rendering the GUI for the status of the lanes	• PinSetterView • • LaneView • Lane
18.	LaneView	• void show() • void high() • Jframe makeFrame • void receiveLaneEvent(LaneEvent le)	• int cur • int roll • boolean initDone • Iterator bowlIt • Lane lane	• Render the view GUI for the alley lanes	• Lane

SNo.	File Name	Methods	Attribute	Major Functionalities	Interlinked Classes
19.	NewPatronView	• void actionPerformed() • void valueChanged() • Vector getParty() • Vector getNames() • void updateNewPatron()	• int maxSize • boolean done • Strinf selectedNick • AddPartyView addParty • String selectedMember	• Setter and Getter functions	• AddPartyView
20.	Party	• Vector getMembers()	• Vecotr myBowlers	• Accessor for members belonging to a party	• ------
21.	Pinsetter	• void ballThrown() • void reset() • void resetPins() • void subscribe(PinsetterObserver subscriber)	• Vector subscribers • Random rnd • boolean[] pins • boolean foul • int throwNumber	• Updates the state of the pins across all subscribers • Simulates a ball being thrown and probabilistically creates a result for the ballThrown() function- either as a foul or some number of pins	• PinsetterObser
22.	PinsetterEvent	• boolean pinsKnockedDown() • int pinsDownOnThisThrow() • int totalPinsDown() • boolean isFoulCommited() • int gerThrowNumber	• boolean[] pinsStillStanding • boolean foulCommited • int throwNumber • int pinsDownThisThrow	• Includes functionalities that mimic the dropping of pins and probabilistaically (or randomly) determines this	• ------
23.	PinsetterObserver	• An interface class	• ------	• Interfaces the multiple classes	• -------
24.	PinSetterView	• void receivePinsetterEvent()	• Vector pinVect	• Constructs a Pin Setter GUI displaying which roll it is • Receives the current state of the PinSetter and the method changes how the GUI looks accordingly	• ------
25.	PrintableText	• int print(Graphics g, PageFormat pageFormat, int pageIndex)	• String text • int POINTS_PER_INCH	• Displays the graphical text on the UI including colour	• ------
26.	Queue	• void add(Object o) • boolean hasMoreElements() • Vector asVector() • Object next()	• Vector v	• Creates a new Queue	• -----

SNo.	File Name	Methods	Attribute	Major Functionalities	Interlinked Classes
27.	Score	• constructor, getter and setter functions	• String nick • String date • String score	• Sets the scores for the players in the game	• ------
28.	ScoreHistoryFile	• Vector getScores(string nick) • void addScore()	• String SCOREHISTORY_DAT	• Writes the scores of the playes into a .DAT file after a game finishes. Makes use of I/O options, reading/writing to a buffer etc	• ------
29.	ScoreReport	• void sendEmail() • void sendPrintout() • void sendln()	• String content	• Generates the ScoreReport and sends it via email/printout to the user.	• Bowler