Date: 25th March 2011, Friday

Meeting Facilitator: Justin

Minute Taker: Surbhi

Time Keeper: Jessica

Discussed presence of duplicate methods in player and storage class

Alternatives:

* Member variable in player is storage since player is taking methods from storage
* Player object that has its own hands, chips- all these things are part of storage hence the player object is a storage object
* Not all card games will have hands, hence we separated the classes
* Instantiate chip object and put into player
* To be changed in the player class is to change the fields of to storage class
* Keep hand member separate? Hand as a list inside the player instead of a separate class
* Player extends storage- storage object inside variables
* Methods to remove from player- constructor that instantiates the storage object
* Player has a constructor that takes the user name and password, which in turn calls load player from storage which has all the variables required by a player (realName, password etc)
* Player calls load player , modify profile- from storage and table methods

Discussed storage class

* How to clean up the storage class code?

Alternatives discussed:

* Define win and loss streak to be continuous as in if the player quits a game then the total games won that far would be the win streak and games lost would be the loss streak
* How do the statistics get stored? savePlayer()
* Trying to make separate classes for winStreak() and lossStreak() which will have set and get methods
* Storage will store the biggest winStreak and biggest lossStreak from the corresponding methods
* Each player then needs an object of each of these classes..But if these classes are tied to storage and player is tied to storage as well- how does this work?
* Dropped the idea of winStreak() and lossStreak()
* << “if this works then lets not mess with it..”- Jessica>>

Storing statistics for last two days- is this desirable or optional?!!!!!!!

Ignoring money stored in the last two days

Discussed player and Table class relation:

Alternatives

* Are we going through the table object to get the player and its hand? Table runs the game, has calls for each player- call deal method in black jack and pass the player as parameter
* Table class sets u p the players, getPlayers() returns the arraylist of players and in blackjack you call the arraylist and then play the games
* Blackjack should get the list of players from table but not store in blackjack, as it will take forever to update the game
* Blackjack should just have logic- hit deal double down
* What part of the program is going through each player- table/something separate?
* Blackjack class and game engine..relation? should it be implemented in the game interface/package?
* Table go through all players
* All functionality of game in player/game?
* Looked at use cases for various options. Consult TA if possible