Group 12

DogeTix: Classes and Methods

A short description of the classes and methods involved in the DogeTix ticket system.

Collin Stubbs, Andrew Gulla, and Matthew Militante
3/5/2014

User: Holds all the information and available actions about and for a user.

getCredit: Gets the credit amount given in the creditAmount variable

changeCredit: takes two parameters, a bool (true for adding on to, false for taking away) and the amount to add/subtract from creditAmount

getUserName: returns username

getType: returns userType

getLoginState: returns loginState

changeLoginState: changes the state appropriately on login/logout

writeToDTF: writes to the daily transaction file using the dtfLog which is a text file that is updated appropriately to hold all user actions

giveRefund: given a user name and credit amount, that amount is taken out of the current users creditAmount and added to the others

buyTicket: given a number and an event name, the number is subtracted from the total number of tickets of the event

getCreatedEvents: returns the events that the user has created

handleEvent: returns an eventHandler to deal with any events created or to be created

handleUser: returns a userHandler for admins to deal with users

Events: Holds all the information about events

getTicNum: returns ticketNumber

getTicPrice: returns ticketPrice

getEventName: returns eventName

getEventCreator: returns eventCreator

changeTicNum: takes two parameters, a bool (true for adding on to, false for taking away) and the amount to add/subtract from ticketNumber

changeEventName: changes eventName to whatever the parameter is

UserHandler: used to handle users (accessed by admin only)

      addUser: adds a user to the database

      deleteUser: deletes a user from the database

      addUserCredit: adds credit to a user

EventHandler: used to handle events

      addEvent: adds a user created event to the database

      deleteEvent: deletes an event the user has created from the database

note: All methods that have something to do with the Daily Transaction File will update the log appropriately so when the user logs out the DTF can be written. Also note that these are all short descriptions and most functions will have conditions that need to be satisfied.

Ticket Service Class Diagram

User

-name : string
-type : string
-credit : long
-loginState : bool

+getCredit() : signed int
+changeCredit() : void
+getUserName() : string
+getType() : int
+getLoginState() : bool
+changeLoginState() : void
+writeToDTF() : void
+giveRefund() : void
+buyTicket() : void
+getCreatedEvents() : string
+handleEvent() : EventHandler
+handleUser() : UserHandler

EventHandler

+addEvent() : void
+deleteEvent() : void

Event

-eventName : string
-sellerName : string
-nTickets : int
-ticketPrice : long

+getTicNum() : int
+getTicPrice() : long
+getEventName() : string
+getEventCreator() : string
+changeTicNum() : void
+changeEventName() : void

UserHandler

+addUser() : void
+deleteUser() : void
+addUserCredit() : void

eventHandlerCreated   0..1

eventCreated   0..1

userHandlerCreated   0..1

userCreated   0..1 *