

Debate Forum

Design Inspection, Code Inspection, & Unit Testing

DESIGN INSPECTION DEFECTS

Product	Debate Forum • User Interface Module
Date	9/21/16
Author	Roy Ramstad
Moderator	Michael Schloss
Inspectors	Cody Tyson • Alex Rosenberg
Recorder	Naveen Ganessin

Defect #	Description	Severity	Solution
1	Expected User class to return new User with separate method besides constructor	1	User class now uses proper constructor an UI utilizes that
2	UI is missing the module for admins and mods that display comments submitted by users	2	

Product	Debate Forum • Object Management Module
Date	9/21/16
Author	Cody Tyson • Alex Rosenberg
Moderator	Michael Schloss
Inspectors	Naveen Ganessin
Recorder	Roy Ramstad

Defect #	Description	Severity	Solution
1	Expected to construct User with method besides constructor	1	Made proper constructor for UI to use
2	Stored passwords when unnecessary	3	Removed password accessor methods
3	Set passwords when unnecessary	3	Removed password mutator methods
4	Expected to handle invalid usernames, but JSON does	2	Assume all users exist (have an account for the provided username)
5	No way for JSON layer to create new user with all necessary attributes	1	Made additional constructor that accepts all applicable fields
6	Was initially only using method for creation of Debate and Post	1	Made proper constructor for UI to use

Product	Debate Forum • JSON Translation Module
Date	9/21/16
Author	Naveen Ganessin
Moderator	Michael Schloss
Inspectors	Cody Tyson • Roy Ramstad
Recorder	Alex Rosenberg

Defect #	Description	Severity	Solution
1	Returns User when Object management needs individual fields	1	Returns individual field values
2	Returns value	2	Returns null when no

Product	Debate Forum • Database Module
Date	9/21/16
Author	Michael Schloss
Moderator	Alex Rosenberg
Inspectors	Naveen Ganessin • Roy Ramstad
Recorder	Cody Tyson

Defect #	Description	Severity	Solution
1	Database returns an object when no data is actually returned	2	Return null if string "No Data" is found
2	SQL statement does not format properly	1	Rewrite 'formattedSQLStatement()' to return a properly formatted SQL Statement
3	Executing an SQL statement would cause the UI to lock up	2	Move database communication to a background thread

CODE INSPECTION DEFECTS

Product	Debate Forum • User Interface Module
Date	9/21/16
Author	Roy Ramstad
Moderator	Michael Schloss
Inspectors	Cody Tyson • Alex Rosenberg
Recorder	Naveen Ganessin

Defect #	Description	Severity	Solution
1	verifyUserLogin()' was being called when the method did not exist	1	Implement 'verifyUserLogin()'
2	UI modules on Create Account page are small	3	

Product	Debate Forum • Object Management Module
Date	9/21/16
Author	Cody Tyson • Alex Rosenberg
Moderator	Michael Schloss
Inspectors	Naveen Ganessin
Recorder	Roy Ramstad

Defect #	Description	Severity	Solution
1	'makeUser()', 'makeMod()', 'makeAdmin()' should update user type status only if JSON layer approves	2	Only update conditionally
2	ban(), 'unBan()' should update user banned status only if JSON layer approves	2	Only update conditionally
3	getUsername(), 'isBanned' check database every time for accessors	3	Use data stored in fields and only update from database when explicitly requested in mutators
4	toString()' use accessors to return values	3	Use fields to get data to reduce function call overhead
5	User(._) use JSON layer methods as if static without instantiation of the UserQuery class	1	Instantiate the UserQuery class for each User and initialize it to the default constructor in the User constructors
6	createPost(._)' attempted to store entire user object into each post	2	Edited to only store the username into the post object

Product	Debate Forum • JSON Translation Module
Date	9/21/16
Author	Naveen Ganessin
Moderator	Michael Schloss
Inspectors	Cody Tyson • Roy Ramstad
Recorder	Alex Rosenberg

Defect #	Description	Severity	Solution
1	UserQuery does not handle threading conditions for 'executeSQLStatement()'. All JSONObjects could be null when read	3	Needs to wait on 'returnedData(_:_)' to return. Hence functions need to wait on thread
2	'getUser()' returns entire User Object instead of the individual parameters	2	Create separate getters for the individual attribute of a User object
3	'verifyUserLogin()' always returns false	2	Change implementation to actually check for validation of the password and return appropriate boolean

Product	Debate Forum • Database Module
Date	9/21/16
Author	Michael Schloss
Moderator	Alex Rosenberg
Inspectors	Naveen Ganessin • Roy Ramstad
Recorder	Cody Tyson

Defect #	Description	Severity	Solution
1	When threading, DFDataDownloader and DFDataUploader would create copies of themselves	3	Removed copies
2	String comparisons were being done with '==' instead of '.equals()'	1	Changed to '.equals()' for proper comparison

USER TESTING DEFECTS

For unit testing, we used the automated testing frameworks Unit & java.awt.Robot

Product	Debate Forum • User Interface Module
Date	9/21/16
Author	Roy Ramstad
Module I/O	Input: JSON Translation Output: UI

Defect #	Description	Severity	Solution
1	Retrieved Location of UI component models are not accurate for robot to place mouse input	3	

Product	Debate Forum • Object Management Module
Date	9/21/16
Author	Cody Tyson • Alex Rosenberg
Module I/O	Input: JSON Translation Outputs: JSON Translation and UI

Defect #	Description	Severity	Solution
1	Serialized Objects not equal when they should be after saving and loading	2	Override 'equals(_:)' and with correct implementation for User object. Also, use 'toString()' comparison instead for additional reliability
2	Users sometimes not equal when not serialized with 'equals(_:)' comparison	2	Override 'equals(_:)' and with correct implementation for User object. Also, use 'toString()' comparison instead for additional reliability
3	Caller of User constructor sometimes received null in tests	1	The User constructor should only be called after checking that the username exists
4	Overridden method 'hashCode()' not generating hash	2	Convert String to hash in return statement
5	hashCode()' not generating unique hash for each field combination	2	Utilize all fields of User in return statement of 'hashCode()' when generating hash
6	Calling the simple User constructor for a User that does not exist results in null pointer exceptions	1	UserQuery functions must have null checks before utilizing objects from the database, specifically in 'getUserPriv(_:)' & 'getUserBanStatus(_:)'
7	App crashed upon use of UserQuery methods in User class due to uncaught exceptions	1	Add try/catch to handle exceptions in all methods utilizing UserQuery
8	'isOpen()' method in Debate was not initialized	1	Initialized 'isOpen()' variable in the constructor as well as the methods instead of just in the methods

Product	Debate Forum • JSON Translation Module
Date	9/21/16
Author	Naveen Ganessin
Module I/O	Inputs: Objects & UI Outputs: Database, Objects, & UI

Defect #	Description	Severity	Solution
1	Null Exceptions being thrown by getter methods when invalid username is supplied	2	Null checks should be performed and appropriate response should be generated
2	Even though, adding a new User fails, it still returns a User Object	2	User Object should be returned as null
3	Getting 'banStatus' always return false	1	Change the method of input from boolean to int from JSON and then convert it to boolean
4	Update Statements works but does not respect the where clause	3	The DFSQL must respect the WHERE statements as well under '.executeSQLStatement()' code
5	'verifyUserLogin()'	1	Properly check for existence of User and Password check
6	'verifyUserLogin()' does not explicitly say whether the password was wrong or username was wrong	1	Could throw InvalidUserException for an invalid User and return false for an incorrect password

Product	Debate Forum • Database Module
Date	9/21/16
Author	Michael Schloss
Module I/O	Input Output: JSON_Translation

Defect #	Description	Severity	Solution
1	Wrong formatted SQL statement returned No Data	2	Fixed typos in SQL Statement
2	No WHERE statement in UPDATE statement	1	Reformatted SQL 'formattedSQLStatement()' to insert WHERE statements when applicable
3	DFDataDownloader and DFDataUploader would lock main thread	2	Moved processing to a background thread