VORMUND

Defect Log

Team 6

Seth Cook
Margarita Lee Li
Samuel Marshall
Adam Rice
Mitchell Stendal
John Vall

Design Inspection Log

Defect #	Description	Severity	How Corrected
1	Calling the password encryption method in Encryption.java from DBHelpers.java when the password was not a multiple of 16 bytes resulted in an error.	1	If password longer than 16 bytes, then it is truncated to 16 bytes for encryption, if shorter then padded to 16 bytes.
2	If DBHelpers.java made a call to a Database.java function with a null entry for the user “key”/password, the function called in Database.java would throw an error	1	Modified DBHelpers.java to ensure that the key was passed properly into Database.java
3	Database.java would inadvertently close result sets before passing it back to DBHelpers.java, which caused DBHelpers to not be able to access the data	1	Modified Database.java to not close statements that contained child result sets so that DBHelpers.java could successfully access the result set.
4	Records classes (e.g., notes, banks, SSNs, etc.) made direct calls to readFromBLOB and insertBLOB when they should be handled	3	Made higher-level calls to DBHelpers instead

	entirely in DBHelpers.		of attempting to read from binary objects directly
5	There were no separate functions checking for whether a username simply existed in the system and where a username & password were valid. There, the same dialog window asking whether to create a new account or not would always appear, eve if an existing username and wrong password were given.	2	Created separate function called checkUsername() to see if to ask user to create new account first, THEN if true check password
6	Website2 window did not serialize security question/answer pairs to 2-dimensional array object in records classes properly.	2	Website2 has been scrapped due to irreparable issues.

Code Inspection Defect Log

Product: GUI
Date: 9/12/2013
Author: Margarita Lee Li
Moderator: Adam Rice
Inspectors: Seth Cook, Samuel Marshall
Secretary: Mitchell Stendal

Defect #	Description	Severity	How Corrected

1	Website module does not contain secure questions and answer component.	1	Website2 module was created to add secure questions.
2	All the modules with textfield are not checking for blank entries.	1	Added a check for when users leave a blank entry for all modules.
3	Clicking on the secondary combo-box item when already clicking the primary combo-box in UserAccount module should show the information o selected item.	3	Modified the show button and created an event function for showing the information of the selected item in the secondary combo-box. Show button changed to update button.
4	LoginWindow should be closed when ‘Login’ button opens up.	3	LoginWindow is now being disposed after a new Window is open.
5	There should always be only one window open at the time.	3	All modules now are disposed when they are done. There will only be one window open at a time.
6	LoginWindow is not set as MainClass in executable	1	Point the MainClass to LoginWindow class

Product: Encryption
Date: 9/12/2013
Author: Adam Rice

Moderator: Samuel Marshall
Inspectors: John Vall, Margarita Lee Li
Secretary: Seth Cook

Defect #	Description	Severity	How Corrected
1	The encryptBlob() method within the Encryption class has two arguments, the user's password key and the data array to be encrypted. The variables were accidentally switched in the method. (Every place where the key was intended to be used, the data variable was used instead.)	1	The two variables were re-named to make the distinction between their use clear, and their use within the method was corrected.

Product: Database Schema/Class
Date: 9/12/2013
Author: Seth Cook
Moderator: Margarita Lee Li
Inspectors: Adam Rice, John Vall
Secretary: Samuel Marshall

Defect #	Description	Severity	How Corrected
1	Function Statement.getBinaryStream does not exist with the SQLite version of JDBC.	1	Replace with Statement.getBytes
2	Function Statement.setBinaryStream does not exist with the SQLite version of JDBC.	1	Replace with Statement.setBytes

3	Calling a ResultSet.get function before ResultSet.next raises and SQLException	2	Call ResultSet.next before ResultSet.get
4	SQL can be injected into update/get/insertBlob functions	3	Check input strings for SQL and throw an error if they contain SQL syntax

Product: Create/Update API Functions
Date: 9/12/2013
Author: Mitchell Stendal
Moderator: John Vall
Inspectors: Margarita Lee Li, Seth Cook
Secretary: Samuel Marshall

Defect #	Description	Severity	How Corrected
1	The newUser function of DBHelpers was not hashing the password provided by the user, and would have been a security threat	2	A call to the encryptHashString was made to secure the password String

Product: Delete API Functions
Date: 9/12/2013
Author: Samuel Marshall

Moderator: Mitchell Stendal
Inspectors: Adam Rice, Margarita Lee Li
Secretary: Seth Cook

Defect #	Description	Severity	How Corrected
1	The delete function can have null input which would delete every part of the database	2	Just put a conditional checking to make sure input isn't null.

Unit Testing Defect Log

Product: GUI
Date: 9/12/2013
Author: Margarita Lee Li

Defect #	Description	Severity	How Corrected
1	The checking for blanks on textfields was not working properly.	1	The variables used were not the correct ones. Switched them to correct ones.
2	Switch statements in mainCB and subCB functions were not	1	Changed the switch

	compiling properly.		statements to if-else with a compareTo.
3	Update button was only showing the newBank module.	2	Needed to add a check for main combo-box and sub combo-box to open the correct windows.

Product: Encryption
Date: 9/12/2013
Author: Adam Rice

Defect #	Description	Severity	How Corrected
1	The implementation used within the Encryption class method decryptBlob() to convert byte arrays to strings (Arrays.toString(byte[])) did not correctly convert the byte array to a character string. This issue led to an IllegalArgumentException, which halted the program.	1	After investigating alternative ways of converting byte arrays to strings, the following method was used: String temp = new String(byte[]). This solved the issue.

Product: Database Schema/Class
Date: 9/12/2013
Author: Seth Cook

Defect #	Description	Severity	How Corrected

1	The Blob object is not supported by the SQLite version of JDBC	1	Read Blobs as a byte array
2	The database file does not exist within the java resources tree	1	Create an empty file where the database file belongs
3	The SQL Schema file does not exist within the java resources tree	1	Copy the Schema file to the proper location in the resource tree
4	ResultSet.isClosed is an unimplemented abstract function	2	Remove calls to the function and test if open using try catch statements
5	Statement.isClosed is an unimplemented abstract function	2	Remove calls to the function and test if open using try catch statements
6	Resource Database file becomes corrupted in executable transfer	3	If code is uncorrupted, users will lose data. Write a function to empty the database file

Product: Create/Update API Functions
Date: 9/12/2013
Author: Mitchell Stendal

Defect #	Description	Severity	How Corrected
1	ResultSet in newUser is TYPE_FORWARD_ONLY therefore cannot call .first() upon the resultSet without causing an error	1	After researching the problem, .next() is used instead to check if resultSet it empty
2	Encrypted password and userName raise database constraint error	1	The returned encrypted password was too large and has been modified
3	Function NewWeb was improperly storing the security question/answer pairs into the CSV style string to be put into the database	1	Modified QA string creation for loop to properly place delimiting semicolons

Product: Read API Functions
Date: 9/12/2013
Author: John Vall

Defect #	Description	Severity	How Corrected
1	The read functions do not cross-check requests with the user ID of that instance, potentially leading to a security breach. Results are returned to matching regards without regard to user ID.	1	Added WHERE user_id='user_id' in

			query
2	All read code [gets()] attempts to use same ResultSet object (containing result data) to test for number of results. After moving the ResultSet pointer forward, it cannot be moved back again to read results.	1	Changed code so it counted the results as the iterator ran through the set
3	Notes, SSN, and Websites records attempting to parse ResultSet data all using Bank's serialization function.	2	Ensured each record was calling its own serialization functions
4	Get methods' queries do not contain a selection for data_id, making it impossible to retrieve it from database and use it in instance of records objects (returned as null)	1	Added data_id to the SELECT clause in SQL statement
5	Security QA's were removed as a feature for website records, but getWebsite and WebInfo.serializeCSVDump attempted to parse them into an array nevertheless	3	Filled array with temporary

			bogus data to circumvent null checking