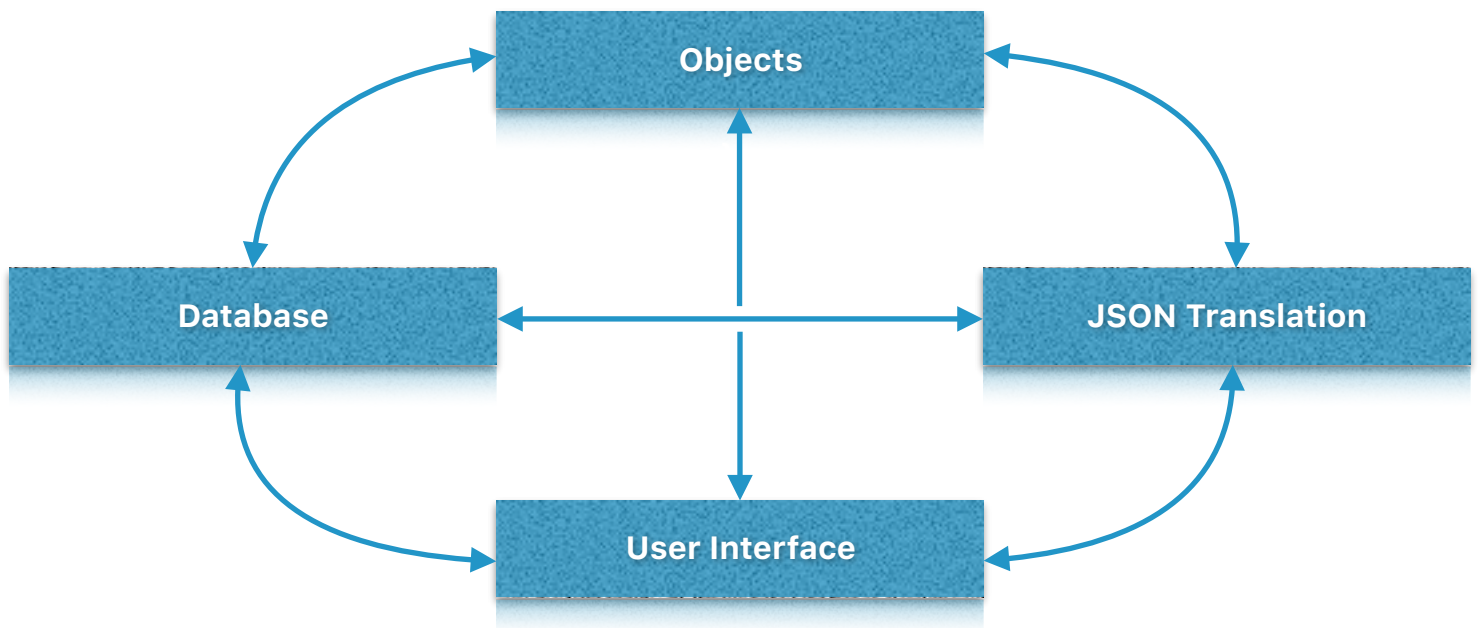


# Debate Forum

## Incremental & Regression Testing

### CLASSIFICATION OF COMPONENTS



Module	Input	Output	Dependent Components
Database	JSON Translation	JSON Translation	UI • JSON Translation
JSON Translation	Database	Objects • UI	Objects • UI • Database
Objects	JSON Translation	UI	UI • JSON Translation
UI	User • JSON Translation	User • JSON Translation	Database • Objects • JSON Translation

We used the **Bottom-up** testing form.

<b>Product</b>	Debate Forum • Object Management Module
<b>Author</b>	Cody Tyson • Alex Rosenberg

INCREMENTAL			
Defect #	Description	Severity	Solution
1	The standard User constructor depends on UserQuery (see JSON Translation Module) returning non-null from its constructor, which it may not, causing a null pointer exception	1	Null check inside standard User constructor before using UserQuery function results
2	The multi-field User constructor depends on UserQuery (see JSON Translation Module) returning non-null from its constructor, which it may not, causing a null pointer exception when accessors and mutators are used	1	Null check inside multi-field User constructor before return
Regression			
1	Standard User constructor may return non-null User that does not have valid field data due to failed UserQuery	2	Check that the username field of the returned User object is non-null
2	Multi-field User constructor may return non-null User that does not have valid field data due to failed UserQuery, causing accessors and mutators to return null	2	Check that the return value of accessors and mutators are valid prior to using their returned values
3	Cached objects may not be properly saved into local storage if saved in an unexpected order by UI	3	UI should use abstracted `saveSession()` method instead of low level
4	Cached Objects may not be properly loaded back into memory if read out of order by UI	3	UI should use abstracted `loadUser()` and `loadDebate()` methods

<b>Product</b>	Debate Forum • JSON Translation Module
<b>Author</b>	Naveen Ganessin

INCREMENTAL			
Defect #	Description	Severity	Solution
1	Translation should wait for response from the database before sending info to UI	1	UI should have a function that translation could call to let it know that data has been received and ready to access
Regression			
1	The UI would always get null Users and false booleans for `verifyLogin()`	1	UI should have a function that translation could call to let it know that data has been received and ready to access

<b>Product</b>	Debate Forum • Database Module
<b>Author</b>	Michael Schloss

INCREMENTAL			
Defect #	Description	Severity	Solution
1	DFDataDownloader and DFDataUploader do not check if SQLStatement parameter is null	1	Provide a check with an exception if SQLStatement is null to prevent thread fall-through
Regression			
1	DFDataDownloader and DFDataUploader are being executed on a background thread. JSON attempts to return immediately without wait for download/upload to complete	1	JSON Translation must wait for the delegate method to be called before returning data to the UI

<b>Product</b>	Debate Forum • User Interface Module
<b>Author</b>	Roy Ramstad

INCREMENTAL			
Defect #	Description	Severity	Solution
1	For fields such as username and password, there is not a maximum character limit. This raises a DFSQLError.	3	Implement a length limit for certain fields
2	For fields such as username and password, when the length of the fields exceeds the width of the screen, the fields shrink to minimum size	3	The method <code>setResizable(_:)</code> is available for Components which will restrict the components from being resized
Regression			
1	A whitespace character is allowed as valid input for the username field. This will cause an issue with the database module using the username as the primary key	2	Use <code>JFormattedTextField</code> as the UI's input field, and specify the expression which are allowed