

# BoilerX

## Code Inspection, Design Inspection, Unit Testing Log

Team Members: Sean Chew, Zuyuan Fan, Qi Meng, Ling Zhang

### Design Inspection

Product	BoilerX		
Date	Mar. 1, 2018		
Author	Development team		
Moderator	Sean Chew		
Inspector	Zuyuan Fan, Ling Zhang		
Recorder	Qi Meng		
Defect #	Description	Severity	How corrected
1	Top navigation bar does not have a profile button that allows for users to easily access their account information.	3	Replace the existing "logout" button with a profile button with user's profile picture. The logout option will be moved into the dropdown box after clicking the profile button.
2	There is no protection mechanism set in place for buyers to confirm a purchase. This might cause problems such as an accidental purchase of a good due to a misclick.	3	Implement a 2-step verification process that will prompt the buyer to confirm the purchase of the good.
3	After a buyer buys an item, the item is not immediately taken down from the site. This might cause problems such as multiple buyers purchasing the same item. However, this isn't a definite drawback as it gives the seller secondary options incase the primary deal falls through.	2	Implement a queue data structure in the database, recording the order of users who purchased the good. This will allow the seller to go through the queue and sell the good to the first available customer who meets the seller's requirements.

## Code Inspection

Code inspection revealed the following defects:

Product	BoilerX		
Date	Feb. 28, 2018		
Author	Development team		
Moderator	Sean Chew		
Inspector	Zuyuan Fan, Ling Zhang		
Recorder	Qi Meng		
1	get_item_by_user will get full item details of the user, many of which is not used in user profile. This also create additional stress to the client.	3	Use a 'projectionExpression' to get only attributes used in the item list displaying format. Write an additional get_item_by_id method to get full detail of a single item when the user request to view/edit one single item.
2	A typo in update_item	1	Delete an extra 't' in 'ExpressiontAttributeNames'. The Correct one should be 'ExpressionAttributeNames'.
3	In post_item, the attribute name 'sellerID' was written as 'sellerid', giving rise to inconsistency in pre-decided item schema and may cause future queries to fail	2	Change item parameter 'sellerid' in update_item.js to 'sellerID', and set the later version standard across all API files.
4	Missing update attribute in update_item.js. The attribute 'searchName' should be modified with the change in 'Name', but attribute 'searchName' is not present in the update item parameters.	2	Add in 'searchName' in update expression attributes.
5	'ConditionExpression' for delete item was written as 'ConditionalExpression'	1	Correct the typo.

6	Update profile function http method specification is wrong.	1	Change http method of this function from 'get' to 'put'. In serverless.yml configurations.
7	Update profile 'UpdateExpression' attribute username typo	1	Correct the typo 'usesname' to 'username'.

## Unit Testing Report

The system relies on the following modules:

- Query Module
- Item Display Interface
- User Profile Interface
- User Profile Module
- Interface Components
- Item Management Module
- Item Management Interface
- Authentication Module

### Item Display Interface

The item interface is the frontend user interface that handles displaying all fetched items on the search result page as well as handles displaying detailed item page. User interact with the interface by input in search bar and click select item filters. It is tested manually.

Product	BoilerX		
Module	Item Display Interface		
Author	Development team		
Defect #	Description	Severity	How corrected
1	Photo is not placed properly on the item page.	2	Not yet corrected. Correction will be to adjust the position so that the photo could be fully shown.
2	Send_email_button is hard-coded to send emails to a developer instead of the seller.	3	In SearchResultDetail.js, fetch seller email information from backend and inject it into send_email method, which will be triggered by send_email_button..

3	Resizing front page causes popular items to overlap with wave design at the bottom of the page.	1	Not yet corrected. Correction will be made by adding a margin at the end of the popular items so that they won't be covered by the wave design.
4	Popular item is non-clickable. User won't be able to be directed to the item after clicking on the item.	2	Not yet corrected. Correction will be made by adding call_back function when the item is clicked.

### User Profile Interface

The user profile interface is the frontend user interface that handles displaying the user's profile information, items the user bought and items the users sold. Users interact with the interface by clicking on their profile picture located at the top right of the page. It is tested using Mocha and Chai automation testing framework as well as manual testing.

Product	BoilerX		
Module	User Profile Interface		
Author	Development team		
Defect #	Description	Severity	How corrected
1	Profile picture does not automatically center the original picture to fit into the display constraints.	2	Use react frameworks found on Github that supports profile picture upload function with multiple functionalities.

### User Profile Module

The user profile management module is the backend module that handles requests to get and update detailed user profile. Web application client will request the user profile to be display. The module will look up the database and feed the client with full details of the required user. Database operations are tested using serverless to invoke local API methods with mock requests and automated using Linux shell scripts. Formatted mock requests are randomly generated with Node.js scripts. Generation of these random mock requests are automated in the test scripts.

Product	BoilerX
Module	User Profile Module
Author	Development team

Defect #	Description	Severity	How corrected
1	Get_profile failed with valid input, error message suggests that the key does not match the table schema	1	We initially created our user database using email as the primary key. As we move forward in our project, we change the user schema design and having userId as primary key. We create a new user table, with userId as primary key.
2	Update_profile will create a new user if the user does not exist.	2	Add a ConditionExpression in dynamoDB update parameters, specifying that the item being written should has userId that equals the userId in the request.

### Interface Components

Interface components include buttons, input manager, and other components in web interface. These interface components are tested using Mocha, Chai, and jsdom React testing framework with automated test runs.

Product	BoilerX		
Module	Input UI Widget		
Author	Development team		
Defect #	Description	Severity	How corrected
1	Resizing window causes profile button to be on a different line from search_bar.	1	Not yet corrected. Correction will be done by minimize the profile button to only showing profile picture so that it could be displayed on the same line as the search bar.

### Item Management Module

The item management module is the backend module that handles requests to update details of an item. Web application client will request the item to be display. The module will look up the database and feed the client with full details of the desired item. Database operations are tested using serverless to invoke local API methods with mock requests and automated using Linux shell scripts. Formatted mock requests are randomly generated with Node.js scripts. Generation of these random mock requests are automated in the test scripts.

Product	BoilerX		
Module	Item Management Module		
Author	Development team		
Defect #	Description	Severity	How corrected
1	Update request return error since attribute name 'name' of item is a reserved word in DynamoDB.	1	Change 'name' in updateExpression to '#name', and add in an field ExpressionAttributeNames to set '#name' indicating the 'name' field in Item table.
2	Update request return error suggesting there is a missing required parameter 'item'.	1	Change DynamoDB operation from 'put' to 'update', according to syntax in aws-sdk.
3	Cannot update item price. Error saying 'One or more parameter values were invalid: Type mismatch for Index Key price Expected: N Actual: M IndexName: price-index'	1	Setting price as sortkey for global secondary index indicating price should not change. Delete global secondary index price-Index for Item table.
4	Each updated attribute has value as a mapping from the attribute type to the updated attribute value.	2	Remove wrong type syntax when specifying attribute values to be updated. DynamoDB can automatically save these values in the identified type.
5	Delete item return error indicating the request key is not compatible with the required schema.	1	Wrong letter case for key in delete expression. Change key 'itemld' to 'itemID'.
5	Update_item will check if the request attributes has matching type with the item schema attributes.	2	Explicitly convert type of request attribute values. Any type-incompatible value will trigger an error and update will not carry out.

### Item Management Interface

The item interface is the frontend user interface that handles displaying all fetched items on the search result page as well as handles displaying detailed item page. User interact with the interface by input in search bar and click select item filters. It is tested manually.

Product	BoilerX		
Module	Item Management Interface		
Author	Development team		
Defect #	Description	Severity	How corrected
1	Unable to upload a picture from local file when the seller list an item.	2	Not yet corrected. Correction will be made by implementing picture picker widget in the frontend and call backend function to upload the picture to S3.