

Design Inspection, Code Inspection and Unit Testing
Team 19

Sprint 1 User Stories

- 1) As a user, I would like to sign up for an account on Bulletin
- 2) As a user, I would like to be able to sign in to Bulletin
- 3) As a user, I would like to be able to create a flyer
- 4) As a user, I would like to be able to view others flyers
- 5) As an admin, I would like to be able to remove any flyer
- 6) As a user, I would like to remove my own flyer
- 7) As a user, I would like to reset my password

Design Inspection

(defects that occur while combining the modules or during the interaction between the modules)

An element of the database was missing the image url

Defect#	Description	Severity	How Corrected
1	Backend is implemented using Node.js. In order to run the server we need to install all the modules that are being used. Its very important to install same version of modules on all the machines.	1	Whenever developer adds a module he saves it in the package.json file, so other developers can install the correct version of the module.
2	If a user forgets their password, they have no way of recovering/resetting their account.	3	Implement a system that emails the user a new password to allow them to recover their account
3	User is allowed to access home route without signing in	3	Create route check in vue.js
4	The Next and Prev buttons along with create flyer button on the the page change styles upon use. They inherit the style defined in the login.vue page.	1	Implement separate styling for the home page.

Code Inspection Defects

(defects arising from inefficient or wrong coding techniques)

Defect#	Description	Severity	How Corrected
1	Api results were misdocumented, causing issues with frontend requests.	3	Updated the backend to match documentation.
2	Calling flagflyer will crash due to "Cannot read property 'owner' of null."	2	Remove functionality for testing if the flagging user is the owner, to make way for more important functionalities until solved.
3	Create flyers button doesn't work correctly because of an error "localStorageGet(auth, null) is not defined". It was preventing to get token from the local storage in order to simulate post request.	3	Changed the syntax. Vue.js understands "localStorage.getItem('auth', null)"
4	Calling flagflyer will crash due to "Can't set headers after they are sent."	2	Got rid of an extra return line that was unreachable.
5	When calling getflyerinfo, searching by _id would only return null	2	In order to successfully search by _id, we had to implement an ObjectId object.
6	When calling getflyers, would return an error for "not able to parse circular JSON structure"	3	In the mongoDB "find" call, use .toArray to keep it from trying to create a JSON string from a JSON string.
7	When calling flagflyer, it will always return "success" even if it could not successfully find the flyer.	1	Include the error condition in the flag() function after the mongoDB call.

Unit Testing Report

(we expect you to clearly define modules of your product, to identify the input and output to each module, and to list down the defects found during the unit testing of each module)

The system relies on the following modules:

- Home module
- Account manager module
- Flyer creation module
- Flyer Find Info Module
- Flyer Flag Module

Home Module

The Home module handles the presentation of flyers and their data to the user. It is used as a launch point for most of the applications functionality. It communicates with the database and displays the information to the user.

Product	Bulletin		
Module	Home		
Defect#	Description	Severity	How Corrected
1	A user can force the website to look for flyers beyond the end of the array. There is no upper limit, which could possibly end in an integer overflow.	2	Limit the corresponding local variable to match the length of the array of flyers.
2	The background effects do not extend to the limits of the screen. A thin white bar shows up on the left and right edges.	1	We need to add body { margin: 0 padding: 0 } In the styling of the main html page.
3	Flyer creation function was allowing user to create flyer with an empty title, description, image and date.	1	Added checking, so in case user tries to submit an empty text field, he receives an alert which

			prevents him from submitting the flyer.
	Flyer creation function was accepting date formats without verifying the date is valid, start date comes before the end date and dates are in the future.	1	Date validation added correctly.

Account Manager Module

The account manager module is a combination of the sign in and sign up functions. It will involve inputting emails and passwords, testing them, resetting them, and changing them.

Product	Bulletin		
Module	Account Manager Module		
Defect#	Description	Severity	How Corrected
1	Invalid emails are allowed through signup.	1	At signup, the backend checks the inputs for invalid email configuration.
2	When resetting password from the pre-login pages, updating the password causes user document and database to change all items of user info to null.	2	Replaced specific database function with another function to avoid replacing all user document elements.
3	When signed in, user is able to change password of another user	3	At login, temporary token (jwt) will be given to user and it will be necessary to access user-specific functionalities.
4	User is able to sign up for an account without specifying a password	3	At sign up, the backend checks to verify the password exists and is at least 8 characters.

Flyer Find Info Module

The Flyer Find Info module is a part of the backend that is responsible for bringing the user any information connected to a certain flyer. It will communicate with the database, and provide them with the proper information for the frontend to display.

Product	Bulletin		
Module	Flyer Find Info		
Defect#	Description	Severity	How Corrected
1	When searching for a flyer by its _id, the module would always return [null]	3	Something about MongoDB does not like searching by _id, so it was fixed by searching for "title" instead of _id.
2	When displaying a flyers information, it would not display everything (only id_, title, description, startdate, image_url).	1	Correct the formatting to include the rest of the fields.

Flyer Flag Module

This module is in charge of handling when a flyer is flagged by one of the users. If it has been flagged too many times (currently set to 5), it should remove the flyer from the database. If the owner of a post flags it, then the flyer should be removed no matter the number of times it has been flagged.

Product	Bulletin		
Module	Flyer Flag		
Defect#	Description	Severity	How Corrected
1	When searching for a flyer by its _id, the module would always return [null]	3	In order to successfully search by _id, we had to implement an ObjectId object.

2	When flagging a flyer, it would not increment by the right amount. Instead of incrementing by 1, it would concatenate '1' to the value. For example, a flyer with 1 flag would show '11' flags next.	2	Use parseInt() function to get the integer value instead of using it as a string that would concatenate.
---	--	---	--

Automated Testing

In order to effectively test our backend, we created some automated test cases using the Mocha framework. Mocha is a JS framework that we use to test our backend calls with the console command "npm test". Because we know what each function should return given certain parameters, Mocha makes it very easy to ensure that our api calls return exactly what they need to return in order to effectively communicate with the frontend avoiding any misalignment.