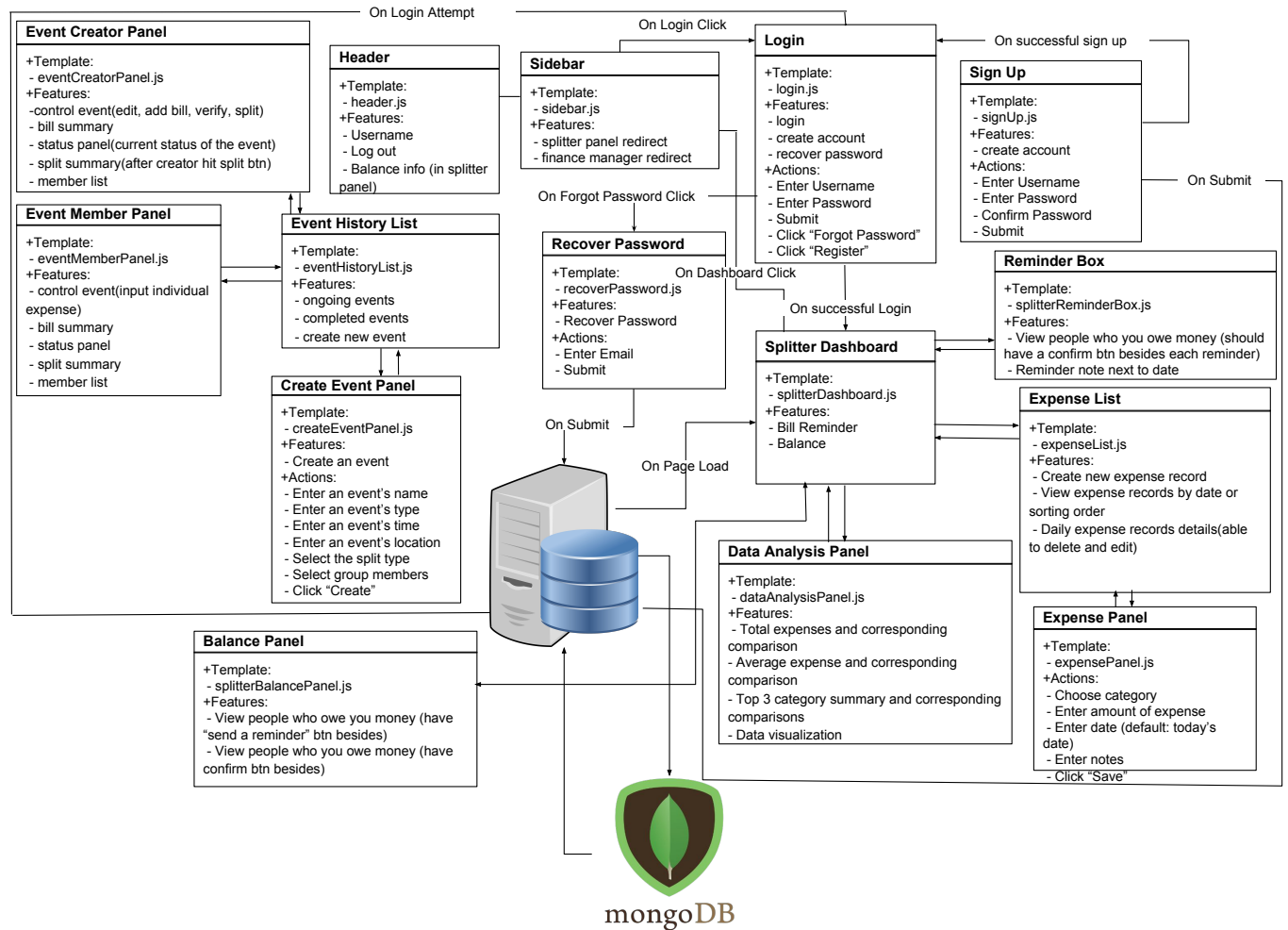


# WeCheck Defect Log for Sprint 1

## Team 3

### Design Diagram



**Design Inspection Defects(severity: 1 == most critical)**

Product	Wecheck Design Inspection		
Date	09/22/2017		
Author	Team 3		
Moderator	Xiaohua Shi		
Inspectors	Moon sun Hwang, Mengxue Luo, Yiyang Zhou		
Recorder	Dongji Cui		
Defect#	Description	Severity	How Corrected
1	No password hashing	1	Send password in plain text to server side to run hashing algorithm.
2	Event data structure has an attribute that contain a list of object (members), rather than a single object. Violate 1NF(first normal form).	3	Create a separate person_event database?. (Combine with eventID as primary key)
3	Only event creator has access to input the total amount and change status of the event. Members should have no access to do so.	3	Add an authentication field (user status) into person_event database. Perform different action based on the authentication status.
4	Only event creator can invite or remove a member to or from an event. Member should have no access to do so.	3	Add an authentication field (owner or member) into person_event database.
5	The username length is too long	2	Set the minimum and maximum between 4 to 20
6	The server allows duplication of user names.	3	Set the username field in User data structure as unique.
7	Sign in module does not validate the username and password	1	Add match function to backend data.

8	Static port number	3	Change code to automatically assign port when port = 3000 is not available
---	--------------------	---	--

### Code Inspection Defects(severity: 1 == most critical)

Product	Wecheck Design Inspection		
Date	09/22/2017		
Author	Team 3		
Moderator	Xiaohua Shi		
Inspectors	Moon sun Hwang, Mengxue Luo, Yiyang Zhou		
Recorder	Dongji Cui		
Defect#	Description	Severity	How Corrected
1	Wrong reference path for bundle.js file in index.html	1	Change its relative path “../public/bundle.js” to “bundle.js” since the compiled frontend files are put in same output directory called public(now called dist)
2	Outdated mongoose.connect() calling convention raises warning message.	3	Adapted to latest calling convention and get rid of warning message
3	Email address validation fails, anything can be accepted	2	Added email checking regular expression at the back and front end
4	Changepassword() function allows password that does not comply with password criteria.	3	Added checks for the length of characters , numbers and special characters.

5	split() function responses, even users have not inputted their amount.	2	Added splitchecking function to check if all users' inputs received.
6	Layout of panels are misplaced	3	Changed the layout to match the corresponding design
7	Delete member function does not work when click on it	3	Added clickListener to delete button
8	Pressing Creation button doesn't allow user to create an event	2	Added a keyboard listener to make sure create button working
9	Not differentiate creator and member	2	Added different flag for different group
10	User cannot login even with correct email address and passwords	2	Using compareTo() function instead of "=="
11	When User's name is too long, it get converted to another panel of the main page	3	Added username check function to constrain the maximum characters to 15.
12	After sending the reminder message, the text in textfield still remain.	2	Added a function to empty text area every time when the send button is hit.

## Unit Testing

### Create Account Module Unit Testing Defect Log(severity: 1 == most critical)

Product	WeCheck
Date	9/22/2017
Author	Team 3
Moderator	Xiaohua Shi
Inspectors	Moon sun Hwang, Mengxue Luo, Yiyang Zhou
Recorder	Dongji Cui

### Module Description

- Overall description of module: manipulate user sign up by creating new users in database, assigning unique user ids, and email address and password validation.

- Tested function:

- create\_user()

Input: email address, password, initial user status

Output: stored user in database with unique user id

- Untested function(code not yet implemented):

- email\_validate()

Input: email address

Output: Boolean indicates if email is valid

- Password\_validate()

Input: password

Output: Boolean indicates if password is valid

Defect#	Description	Severity	How Corrected
1	When creating user, unable to store user status	1	Fixed: Add additional field in the user schema.
2	When storing user password into database, the password is not hashed.	1	Not yet fixed. Planning to add hashing algorithm

## Database Modules Unit Testing Defect Log(severity: 1 == most critical)

### Connection Module

- Overall module description: process routes, POST request and GET request from the frontend
- Tested function:
  - mongoose.connect()  
Input: database url  
Output: message states whether it successfully connected to database

### API Module

- Overall module description: provide interface for retrieving data from database
- Tested function:
  - get\_all\_user()  
Input: none  
Output: json object with all stored user in database
- Untested function:(code not yet implemented)
  - set\_user\_username()  
Input: username, userid  
Output: updated username of a user in database
  - set\_user\_password()  
Input: user password, userid  
Output: updated user password in database
  - set\_user\_status()  
Input: user status, userid  
Output: updated user status for a specific user

# defect	Description	Severity	How corrected
1	When POST to database, node gives warning message “DeprecationWarning: Mongoose: mpromise (mongoose's default promise library) is deprecated, plug in your own promise library instead”	3	Not yet fixed