

Defect Log - Team 4

Matthew Lenartowicz, Tyler Grikas, Aaron Dierking, Evan Tragesser, Guntas Grewal, Jacob Nordland

Design Inspection Defect Log

Product	StudyBuddy Design Inspection
Date	2/12/2015
Author	StudyBuddy Team
Moderator	Tyler Grikas
Inspectors	Aaron Dierking, Evan Tragesser, Guntas Grewal, Jacob Nordland
Recorder	Matthew Lenartowicz

Defect #	Description	Severity	How Corrected
1	Dependency issues when integrating ParseLoginUI with app skeleton	1	Added user permission lines to AndroidManifest.xml file to allow application to access the internet and be in a network access state.
2	Integration of Purdue.io API introduced numerous defects, due to issues with the API itself.	1	We decided to modify the Purdue.io API to return a complete list of courses at Purdue. We then use this to populate the Course list database on Parse.
3	Sinch API is not supported for x86 architectures.	2	Remove x86 devices from our target device list.
4	Integration of Parse push notifications caused fatal errors.	1	Added user permission lines to AndroidManifest.xml file to allow application to access the certain device features, including: wake_lock, receive_boot_completed, vibrate, get_accounts, and receive.
5	When creating a request on Parse, we have a relation that points to all members of the group. This is redundant and overcomplicates things.	3	We removed the members relation in Parse.
6	We are saving Parse objects locally when we only should be saving/pulling them from the database.	2	We removed the code saving Parse objects locally and instead save/pull from the database.

Code Inspection Defect Log

Product	StudyBuddy Code Inspection
Date	2/12/2015
Author	StudyBuddy Team
Moderator	Tyler Grikas

Inspectors	Aaron Dierking, Evan Tragesser, Guntas Grewal, Jacob Nordland
Recorder	Matthew Lenartowicz

Defect #	Description	Severity	How Corrected
1	In ParseLoginActivity.java where we set the string of classes, we add a blank class to our local copy of class list. This could result in errors when accessing database information.	1	We removed this line of code.
2	In LoginHandler.java we don't check if linking the user with the installation succeeded. If it fails, it would prevent push notifications from working.	1	Added a callback method that determines if this succeeded, displaying an error if it does not.
3	DrawerActivity.java drawerItems string array should be in strings.xml since they are UI constants.	3	Removed this string array from the java file and added it to strings.xml.
4	In DrawerActivity.java we have redundant methods that can be removed: onCreateOptionsMenu, onSectionsAttached, and placeHolderFragment.	3	Removed these methods.
5	In DrawerActivity.java line 179 should use a string constant in strings.xml	3	Removed the string from the java file and added it to strings.xml.
6	In ClassListFragment.java, we do not alert the user if an error occurs while trying to retrieve their classes.	1	Added a callback method which checks if classes were pulled correctly. If not, it displays an error.
7	In RequestListActivity.java, we incorrectly name the view as classList, when it should be called requestListView.	3	We changed the view to be called requestListView.
8	In RequestListActivity.java, we display a very long toast if there are no requests for the selected course. Instead, we want to have a static message box appear, that when clicked, brings the user to the new request screen.	2	We removed the toast and made the sole list item a message box stating there are no active requests for the course. If you click the list item, it brings you to the new request screen.
9	The Parse query in RequestListActivity.java can return null, and it doesn't check if this is the case. This could result in a null pointer exception.	1	Added a statement to check if the query is null, and if it is don't proceed. It also notifies the user of an error.
10	In NewRequestActivity.java the toast messages indicating that the input fields are not correct should be string constants in strings.xml	3	Removed the strings from the java file and added them to strings.xml.
11	In NewRequestActivity.java the error toasts should display a generic error so we don't confuse the user.	2	Replaced the error messages to be more generic and easier to understand.
12	In NewRequestActivity.java we save the request id which is redundant and not necessary.	3	Removed the code that does this.
13	In RequestInfoActivity.java we have an async task that doesn't have purpose.	3	Remove the Async task and combine the onReturn and Async task together

14	When creating a new request, the application does not check if the course object is properly saved on the server.	1	Moved the code to save the course object inside the callback method which saves the user object, thus requiring the course object to be saved properly for the app to display a message indicating success.
15	RequestInfoActivity displays the number of helpers and leechers, when it should display the number of helpers and the total number of group members.	2	Remove the leechers statistic, adjust it to be helpers and total people in the group. The leecher statistic will be repurposed for this total
16	ClassAddActivity.java error messages should be more generic, so we don't confuse the user.	2	Replaced the error messages to be more generic and easier to understand.
17	When adding a new course to a user's course list, ClassAddActivity.java saves the course locally regardless of whether or not it saved successfully on the server.	1	Added the code to save the course locally inside of a callback method, which gets called after the course is saved on the server. If it is not successful, it does not save it locally.

Unit Testing Defect Log

Product	StudyBuddy Unit Test
Date	2/9/2015
Author	StudyBuddy Team

Defect #	Description	Severity	How Corrected
1	After closing the app via the Android multi-tasking menu, the app should not remain open. But, the app remains open and causes Android OS to display an error message every time the phone is unlocked. It reads: "Unfortunately, StudyBuddy has stopped."	3	The Parse framework was being initialized from an activity instead of from the Application class. Initialization was moved to Application.onCreate() as required.
2	Tapping the phone's "back" button from the Course List screen should close the app. But, it takes it back to the non-functioning placeholder login screen instead.	1	Added a call to "finish()" from the login activity and removed the line to setContentView to the old login view.
3	Tapping the "submit" button in the New Request menu should take you back to the Request List page. But, it stays on the New Request menu.	2	Added a call to "finish()", which closes the New Request menu.
4	Adding a new class to the user's class list should add it once, but instead it adds it twice.	1	Class was being added to the adapter and the dataset that backed it. Removed the call to add the class to the adapter.
5	If you are logged in, quit the application, and open it back up again, you should be automatically logged in. However, you have to manually log in every time you open the app.	3	Added a shared preferences boolean to keep track if a user has logged in.
6	When starting the app, automatic login caused it to crash with a NullPointerException, when Parse user information was requested.	1	Followed Parse documentation on how to handle automatic login instead of reading from ParseLoginUI's configuration

7	Logging out and then pressing the Back button lets you go back to your profile page when it should quit the app instead.	2	Add a finish() call after pressing the logout button, so the user can't do this.
8	Tapping the "submit" button in the New Request menu should create a new request, if all forms are filled correctly. Instead, an error ("Unfortunately, StudyBuddy has stopped.") occurs, and the menu closes.	1	String variable holding the currently selected class was named "class_selected" and "selected_class" in different sections, causing a null pointer exception. Changed them all to "selected_class".
9	You should not be able to successfully submit a new request with any of the fields empty. However, the app allows you to do this.	1	The app now checks that each field isn't empty before creating a new request. If it is, it displays a toast and stays on the New Request menu.
10	You should not be able to successfully submit a new request with a duration of 0 ms. However, the app allows you to do this.	1	The app now checks that the duration is greater than 0. If it is not, it displays a toast and stays on the New Request menu.
11	If you tap on a course and there are currently no active requests for said course, the app should display a message stating this. Instead, the list simply contains no items.	3	The app now displays a message (toast) if there are no active requests for a course.
12	If you attempt to add a class to your course list and for whatever reason the server interaction fails, a relevant error message should appear and the app should stay on the Edit Courses screen. Instead, a generic error message ("Unfortunately, StudyBuddy has stopped.") occurs, and the app goes back to the main Course List screen.	2	The error prompt was referencing a null object, causing a null pointer exception. This reference was removed and replaced by a string constant.
13	When you add a new course to your course list, you should not be able to add the same class twice. However, the app allows you to do this, displaying two of the same class.	2	When adding a class, a check will be made before adding the class. Check the selection against the array adapter for the list, and if the selection already exists then send a toast to the user notifying them to not pick duplicate classes.
14	After you create a new request, the request list should automatically update. Instead, it only updates after you leave the page and then come back.	2	After a new request is successfully made, we pull the latest request list from the database and display it in the UI.
15	If you are on the course list page and you do not have any courses registered to your profile, the app should display a message saying this is the case. Instead, the page is blank and no message appears.	3	We now display a message on the course page, which you can click. It brings you directly to the Add Courses screen.
16	If you are on the messages page, and select the recipient text box the system does not pop up a keyboard.	1	In the .java file, added a statement to force show the keyboard.
17	If you tap on the Messages tab, the page title does not change to Messages. Instead, it stays as whatever title it was previously.	3	Added a string resource for the title and changed the textView when selecting messages.
18	When adding a course, there should only be one entry for each class. Instead, for classes like CS 490, there are many entries to choose from.	2	Added full descriptions to the classes so multiple courses with the same id will be distinguishable.

19	When adding a class, clicking on the course name should immediately add the course to your course list. Instead, you are required to hit the "+" button after selecting a course.	3	Removed the "+" button and made it so that clicking a course immediately adds it.
20	In RequestInfoActivity and NewRequestActivity, the toolbar on the top scrolls with the view (moving offscreen), when it should always stick to the top of the screen.	2	Surrounded the activity layout in a LinearLayout and placed the toolbar outside of the scrollview.