Saint Leo University COM-430-OL1

Prof. Mitch Raton

[Team Dev Op Project Submission 4](https://saintleo.brightspace.com/d2l/le/content/170245/viewContent/8311470/View) – (Version 1 to Test)

15 FEB 2025

Version: 1.0 Test

Authors: Nicholas P Orlick

Alisha Richardson

Theodore Chase

Levi Leuk

Nicholas Taylor

Dorde Racic

Contents

[Overview 1](#_Toc190511093)

[**Front-End RTM** 1](#_Toc190511094)

[**Resolution Plan & Troop to Task** 2](#_Toc190511095)

[**Backend RTM** 3](#_Toc190511096)

[**Resolution Strategy & Troop to Task – Back end** 4](#_Toc190511097)

# **Overview**

For this week's outline, Team 2 will discuss the testing portion of the development and provide a Requirements Traceability Matrix or RTM, which is a tool that Group 2 uses to track any defects so the team can ensure that the software is reliable. This document captures front-end development issues in the *TLAN Interactive Training* application, highlighting areas that require immediate attention. The team has categorized the problems based on severity and will provide solutions; this outline will show the front and back RTMs and help the team debug the TLAN Interactive Training application. This outline will also include a formalized breakdown on the front and back end and consolidate the troop to task this week.

# **Front-End RTM**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Component** | **Issue Description** | **Severity** | **Resolution** |
| Training Adapter | Symbol reference issue in ‘item\_training\_module’ | High | Fix XML reference |
| Training Adapter | Incorrect method call for TrainingAdapter | High | Update reference to correct class name |
| TrainingModule | Unresolved symbol in ‘TrainingProgress’ | High | Address linking issues |
| TrainingModule | Method ‘getProgressPercentage()’ not recognized | High | Ensure proper method reference |
| TrainingProgressFragment | Duplicate definition of ‘TrainingProgress’ | High | Remove redundant code |
| General UI Component | Incompatible type in SwitchMaterial usage | High | Adjust type to match expected ‘android.widget.Switch’ |
| Repository Call | The mismatched parameter in ViewModelProvider get() | High | Ensure type compatibility |
| User Profile | Cannot resolve symbol ‘profileWelcomeMessage’ | High | Define missing variable |
| TrainingProgress Model | Incorrect constructor call with invalid parameters | High | Update constructor parameters |
| Training Module Adapter | Class ‘TrainingModuleAdapter’ is never used | Low | Remove or refactor the unused class |
| Training Adapter | Problems related to adapter reference | Low | Ensure correct package import |
| Training Module Adapter | Constructor method was never used | Low | Assess the necessity of the method |
| Training Module Adapter | Unused method ‘updateModuleList()’ | Low | Remove or optimize implementation |
| UI Update Handling | Inefficient reliance on notifyDataSetChanged() | Low | Use specific change events |

Both tables will show the top 20 most critical issues. The team is prioritizing those that prevent core functionality. Most of these errors went on and on, but again, they revolved mostly around missing backend components, incorrect method calls, and reference issues.

# **Resolution Plan & Troop to Task**

**1. Critical Issues (High)**

* The biggest priority is resolving symbol-related errors and incorrect method calls to prevent runtime failures.
* The development team must verify file references, function names, and method calls to ensure module consistency.

**Alisha Richardson (Lead Developer)**

* Oversees and coordinates debugging efforts, confirming alignment with project objectives.
* Leads resolution of high-severity symbol reference and method call issues.
* Conducts integration testing and validation to confirm fixes are correctly implemented.
* Oversees final validation and ensures system-wide integrity post-fixes.

**2. Low-Severity Issues**

* Non-critical issues like unused code and incorrect naming conventions should be addressed to maintain clean and maintainable code.
* These should be tracked but can be resolved in later iterations if they don’t impact immediate functionality.

**Nick Taylor (Assistant Developer)**

* Handles lower-severity issues related to unused methods, class references, and outdated implementations.
* Assists with backend integration concerns related to TrainingProgress and repository calls.
* Collaborates on code cleanup and performance optimizations.

**3. Validation and Testing**

Once all fixes are made, the team will conduct testing to ensure all changes work correctly.

Developers must verify that updates function as intended and not introduce new errors.

**Theodore Chace (Assistant UI/UX)**

Focuses on UI-related issues such as SwitchMaterial type compatibility and unresolved UI symbols.

* UI components adhere to best practices for maintainability and accessibility.
* UI testing and refinement after backend fixes are completed.

# **Backend RTM**

| **Test Component** | **Issue Description** | **Severity** | **Resolution** |
| --- | --- | --- | --- |
| Training Adapter | Symbol reference issue in item\_training\_module | High | Fix XML reference |
| Training Adapter | Incorrect method call for TrainingAdapter | High | Update reference to correct class name |
| TrainingModule | Unresolved symbol in TrainingProgress | High | Address linking issues |
| TrainingModule | Method getProgressPercentage() not recognized | High | Ensure proper method reference |
| TrainingProgressFragment | Duplicate definition of TrainingProgress | High | Remove redundant code |
| General UI Component | Incompatible type in SwitchMaterial usage | High | Adjust type to the android.widget.Switch |
| Repository Call | Mismatched parameter in ViewModelProvider.get() | High | Ensure type compatibility |
| User Profile | Cannot resolve symbol profileWelcomeMessage | High | Define missing variable |
| TrainingProgress Model | Incorrect constructor call with invalid parameters | High | Update constructor parameters |
| Training Module Adapter | Class TrainingModuleAdapter is never used | Low | Remove or refactor unused class |
| Training Adapter | Problems related to adapter reference | Low | Correct package import |
| Training Module Adapter | The constructor method wasn't used | Low | Assess the necessity of method |
| Training Module Adapter | Unused method updateModuleList() | Low | Remove or optimize implementation |
| UI Update Handling | Inefficient reliance on notifyDataSetChanged() | Low | Use specific change events |

# **Resolution Strategy & Troop to Task – Back end**

**1. Critical Issues (High)**

* Symbol-related errors and incorrect method calls cause runtime failures.
* Backend data retrieval issues due to local database inconsistencies or Google Drive file syncing delays.

**Nicholas Orlick (Project Manager)**

* Oversees all debugging and ensures alignment with project goals.
* Coordinates issue tracking and developer task assignments.
* Manages communication between development and testing teams.
* Verifies that all changes follow the project's timeline and requirements.

**Dorde Racic (Assistant Developer)**

* Fixes symbol-related issues to prevent crashes in both frontend and backend.
* Method calls to reference the correct local database structure (ensuring SQL queries execute properly).
* Manages file syncing issues between the local database and Google Drive.
* Assists in backend troubleshooting, local JSON/DB files are correctly referenced.

**2. Low-Severity Issues**

* Non-critical issues like unused code, inconsistent naming conventions, and minor UI bugs that do not impact functionality.

**Dorde Racic (Assistant Developer)**

* Handles non-critical backend fixes, such as removing unused code and renaming functions for clarity.
* Cleans up old references to deprecated database schemas.
* Optimizes database queries to reduce load times on the local machine.

**Levi Leuck (Lead Tester)**

* Identifies UI/UX inconsistencies and reports minor graphical issues.
* Tests local database queries for expected results before fixes are finalized.
* Validates that UI components reference the correct backend data.

**3. Validation and Testing**

**Nicholas Orlick & Levi Leuck (Lead Tester)**

* Performs manual and automated testing on the local database and Google Drive data sync.
* Confirms data consistency between local storage and Google Drive.
* Validates that UI changes correctly reflect data from the local backend.
* Stress-tests local queries to check for performance bottlenecks.