**Bug Reports App**

**Description**Application version of the Bug Reports App allowing a user to work with a database of bug reports. Users will have the ability to view, add, and edit bug reports and store them on the database.

**Use Cases to be solved:**  
 Browse list of bug reports  
 Select and view a bug report  
 Add a bug report  
 Edit a bug report  
 Move a bug report to the archive  
 Report number of bug reports in each status  
 Report number of bug reports in each area  
 Report number of bug reports in each feature  
 Report number of bug reports in each version  
 Report number of bug reports being worked on by each employee

**Minimum Viable Product**

**Main Window**

Bug list

Add bug button

Edit bug button

Export bug list button

Overview counts tracking how many bugs are in each status

View Bug

Displays the selected bug report in a window.

Previous and next button to go though the reports.

**Create bug**

Fields to enter bug report

Submit button

Cancel button (prompt about discarding data)

**Edit bug**

Get selected bug report

Fields to edit bug report

Submit button

Cancel button (prompt about discarding data)

**Storage**

Storage to database

**Overall Design**

**Organization:**

Data driven app that will utilize a three-tier model allowing for easier expansion in the future. The tiers are as follows:

BugReports (Connected to BugReportLogic and BugReportObject)

BugReportLogic (Connnected to DataAccess and BugReportObject)

DataAccess (Connected to BugReportObject)

The project will also need a custom data type representing a bug report object.

BugReportObject

The BugReports layer interacts with the user, getting and displaying information to the user and passing it to the logic layer.

The BugReportLogic references the DataAccess layer to gather any data from storage. It will also need to use and filter data objects for validity before passing.

All the layers need to reference the BugReportObject to use the data type created.

**Classes by Layer:**

**BugReportObjects**

BugReportObject

int Bug\_Ticket\_Id

string Bug\_Status\_Id

string Bug\_Description

int Bug\_Assigned\_To

string Bug\_Date

double Version\_Number

sring Bug\_Area\_Name

string Bug\_Last\_Worked\_Date

int Bug\_Last\_Worked\_Employee

int Submit\_Person\_Id

**BugReports**

BugListForm

Control for displaying a list of bug reports

Control for selecting the current employee

Button to view bug report

Button to add bug report

Button to edit bug report

Button to refresh bug report stats

BugReportView

Display bug report items

Return button

Buttons for previous/next

BugAddEditForm

Display bug report items

Button to create/update

Button to delete if update window

Button to cancel

**BugReportLogic**

BugReportManager

List<BugReport> BugList

BugReportManager() //Default constructor to populate list.

List<BugReport> EditBugReport(BugReport bugRep, int bug\_ticekt\_id)

List<BugReport> AddBugReport(BugReport bugRep)

List<BugReport> DeleteBugReport(int deleteIndex)

Int[] UpdateStatusList()

**DataAccess**

BugReportAccessor

List<BugReport> RetrieveBugList()

Bool UpdateBugReport(BugReport, List<BugReport> BugList)

Bool AppendBugReport(BugReport bugRep)

Bool ArchiveBugReport(BugReport report)

**Coding Notes:**