[[1]](#footnote-2)

*Overview*— This project was to create a to do list Android application that would allow users to add, delete, read, and update items in a to-do list while also scheduling notifications for those to-dos.

Assignment 2 – To-Do List

Luke Brandon

, 2) explain your methodology, 3) explain the outcomes. Shoot for between 200-300 words.

# INTRODUCTION

This project is to design and develop and Android application that is a functioning To-Do List. This app should allow the user to add, delete, read, and update the todos that are on their to-do list. The application consists of 2 activities, one for viewing the entire to-do list and another for viewing and/or editing the contents of a to-do. The application should store all of the information for all of the to-dos in a SQLite database on the phone.

# Application Design

This section is used to describe your development methodology. It would be apt to include a figure here to describe the View of your system. If you settled on a particular design ideology (MVC, MVP, MVVM) explain that choice. Identify the important variables in your model. Explain where they are manipulated by the controller/presenter. Identify important functions and their parameters.

## Flow

When creating a new to-do it sets the date to the current day

# Results

This section should describe how your application performs. For class projects, you should describe how it compares to the written test objectives. **Show an image or two of your screen**. This should mainly focus on does the application work as intended, and as described in the introduction and the application design sections.

# Future Work

The code architecture for this application is less than optimal. From the start I should have decided on a design methodology like MVP and developed the app on top of that. I did my best to abstract out the different aspects like views and the model, but currently a lot of the log is done in the activities when it should be done in presenters that present the information that needs to be displayed to the views. With the additional complexity that not having a rigid design structure added, in the future I will decide on and develop under the MVP methodology from the start to avoid the additional complexity and difficulty of figuring out problems. If this application were to be developed further, the MVP methodology would make the code scale significantly more simply, and efficiently.

For class projects, if you run into bugs in your application that you haven’t fixed, include them here with a test case. If they fail one of the hidden tests, but you explain what you think is happening and that you identified the bug, I may take fewer points off.

# Discussion

Optional section where you may include any relevant details that you didn’t think fit in other sections. The following section is the References. I have left the example formats so that you can cite any work that you borrowed for the project, whether book, online, journal, etc…

1. [↑](#footnote-ref-2)