Reci-Please

[Date]

Group members:

Zachari Nicholls – z5349535@ad.unsw.edu.au

Michael Anderson – z5349512@ad.unsw.edu.au

# Introduction

Brief explanation of the project and its functional requirements (you may refer to information from Project 1 if necessary)

# Live Application

Link to the live application (front-end and back-end integrated and fully functional)

# User Manual

User manual on how to use the APIs and any other important details, such as user login credentials to test the application

# API Application Details

URL address or IP address to the API application

# Fully Functional Application

URL address or IP address to the fully functional application

# GitHub Repository

Throughout this assignment both members of the group have obtained a drastic improvement in their ability to utilise GitHub. Prior to this assignment we have both not been proficient in its ability to assist in the maintenance of a project. However; we have begun to utilise it to it’s strengths and now are happy with how GitHub is used.

You can access our repository through the following link:

https://github.com/zac-ari/Reci-please.git

# Readme file information:

## Team members (including their email addresses)

## Project description

## Running the project instructions

# Cloning and Running the Application Locally

## Instructions on how the course convenors can clone and run the application locally

# Functional and Security Testing

## Explanation of the process of functional and security testing

## Testing APIs using Postman

Refer annex A for the testing documentation.

## Screenshots as evidence of successful testing

**Annex A – Testing**

We conducted our testing in a phase approach utilising Postman – These phases where create, read, update, and delete.

Phase one

To begin with I ensured that all data was removed from the database, as can be seen through Laragon in figure 1. Additionally 2 shows a Postman search whilst the tables are empty. Once this was established we then conducted several creates with basic and self-explanatory data for the inputs. \*Note\* at this point we are not doing edge testing, merely functionality and error handling. Figure 3 shows all three databases conducting a successful create through postman

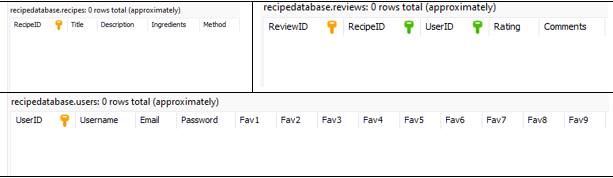


Figure 1:Empty tables