

# MarkR Project: Phase 1

## Functional requirements

For this first phase, I will focus on beginning the process of implementing the first key function. That is:

1. Recording exam results in a database

This functionality will be broken down into a number of features.

## Use case

A typical use case for this feature would be a teacher who is looking to store marks for an exam. While they could set up an Excel spreadsheet, they appreciate the convenience of being able to quickly set up and modify exams. They also appreciate that the system will automatically store their exam results as they mark.

## Features

### Feature 1: The framework

The MarkR application will be built around a framework that will allow efficient addition of future features. This framework will be built from vanilla PHP and be based around the Model-View-Controller (MVC) approach. This essentially means that the application will run through controllers, called from a URL request. These controllers will call functions responsible for dealing with data (models) and presenting views to the client. This process will also involve the use of an SQL database. The user won't see this feature, however a considerable amount of time will go into implementing this feature properly.

### Feature 2: The login system

The application needs a login system. Users should be able to register a username and password. The username should be unique and the password should be hashed in the database. At this stage, users will be able to log out, but will not be able to change their password or delete their account. These features will be added in a future phase, as well as further security features.

### Feature 3: The layout

The application will have a modern layout, that is easy to navigate. Bootstrap will be used to provide CSS classes to components, and jQuery will be used to support client-side processing as well as load a common header and footer for each page.

### Feature 4: Creating a new exam

The user will be able to create a new exam by clicking a button on the home page. They can choose a subject, year level, class code and specify the number of questions. Then, they can see a summary of all the exams they have created. From this page, they can delete exams,

or they can add students by clicking “Edit students”. Here they will see a class list and be able to add students or delete then.

#### **Feature 5: Entering results**

From the exam list, the user can click “Mark”. This will take them to a page with a spreadsheet design. For each student, there is an input for each question in the exam. Here, the teacher can enter their marks. For this phase, the storage of marks within the database will not be implemented. This will be the first feature added in the next phase.