# DIGITISED DRO (MYDRO)

Tapiwa Chikwanda

System Specifications Document

Table of Contents

## Summary

Michaelhouse currently employs an ineffective notice system between staff members and students. Abbreviated to DRO, the Daily Routine Order has sprouted a variety of complaints: some of which prompting, and others resulting from, an interview I had held with the School's Pastoral Administrator. Currently, DRO operation is as follows. Before a daily cut-off time of 08h30, staff members may send via email to the Administrator a notice for the day's DRO. After the cut-off, the Administrator spends 30 minutes at the shortest, manually sifting through these notices for problematic formatting. Once ready, a notice falls under one of six broader sections – To See Staff; [General-]Notices; Sport; Clubs and Societies; Sanatorium; [Detention] Restrictions) – which are visually separated on the final .pdf of the DRO. In time for mid-morning breaks, hard-copies of this file are printed and distributed for display at high-traffic areas on campus - such as the Boarding Houses, Media Centre, and Dining Halls - until removed the following weekday.

Unfortunately, strategic placement does not guarantee that a DRO message will reach its recipients, let alone the student body at large. This is the foremost access shortcoming of the DRO. Moreover, while the current DRO system is in some respect digital, its processing shortcoming is the required manual collation. I intend to address both shortcomings with my proposed application, automating collation by staff and providing electronic access to students.

Note: Greyed Text in this document indicates proposals beyond the scope of a Minimum Viable Product.

## Specifications of Program Function

Users will belong to one of three categories, Students, Staff, and Administrators. These will grant cumulative levels of privileges to the DRO System i.e. all Admins have Staff privileges; all Staff have Student privileges. Only the minimum authorisation required is shown.

The application's functions comprise 3 broader categories, falling primarily to the class of users in parentheses:

- **Management** of the DRO, its structure
  (**Admins**)
- **Publishing** content to be shown
  (**Staff**)
- **Viewing** notices as required
  (**Students**)

### AUTHENTICATION AND AUTHORISATION (STUDENTS)

Allow users to login using some form of unique identification and password, and gain their relevant access to the DRO system

- Show login fields (ID and Password) and allow user to type into both
- If both inputs are valid, fetch and apply user rights
  - Progress to respective landing pages

- Otherwise, inform user of error

### VIEW NOTICES (STUDENTS)

Allow user to view and filter notices

- A categorised list of the day's notices shows
- The list may be filtered by any of its fields

### SUBSCRIBE TO GROUPS (STUDENTS)

Allow user to specify which groups' notifications they receive

- Select/Deselect groups of interest to associate with profile

### NOTICE REQUEST (STAFF)

Add to, edit or, delete from the DRO – subject to Admins' approval

- Allow Staff Member to supply the notice's Title, Description, Dates, Times, DRO Group, Specific Students
- Notify user of declining

### EXPORT TO PDF (STAFF)

Optional formatting for a single day's DRO

- Allow selection of Date and Sections to display
- Output a .pdf file with the required information

### NOTICE APPROVAL (ADMINS)

Allow approval/declining of DRO publishing requests

- Show pending requests
- If approved, give option to edit formatting before publishing
- If declined, exclude the notice from the DRO

### CONFIGURE NOTICE GROUPS (ADMINS)

Allow Admins to perform CRUD on notice groups

- Show, Edit, Add, and Remove Notification Groups

## Specifications of UI (Subject to variation)

The application will consist of two responsive frontends; each interfacing, however, with a single backend which is further specified on page 6:

1. A Desktop Web-Based (Single Page Application)
   **(All Users)**
   Although the website is primarily for Admins and Staff, the School's phone restrictions warrant a method of desktop access available to the Students

2. A Mobile Progressive Web App (PWA)
   **(Students)**

Key:
[ ] – Condition ( ) – Property or Contents

### DESKTOP SINGLE PAGE APPLICATION (ADMINS AND STAFF)

- Login Page
  - Header (Michaelhouse Logo, Page Title, Menu: Manage, Publish, View [Based on profile type, if logged in], My Profile [if logged in], Logout [if logged in]), Help
  - Heading (Login)
  - Username Text Input
  - Password Text Input
  - 'Proceed' Button
- Logged In: Start Screen
  - Large horizontally adjacent and subtitled cards (Manage, Publish, View [Based on profile type])
- Available Functions
  - "Manage" hyperlinked list (Pending Notices, Notice Groups) [shown if Admin]
  - "Publish" hyperlinked list (New Notice, Export to PDF) [shown if Staff Member]
  - "View" hyperlinked list (All Notices, My Subscriptions, Feed)
    - Pending Notices
      - List of requests showing descriptions, user, toggle of approval with select all button in header row
      - Save button in the bottom right corner
    - Notice Groups
      - Four buttons (View All, Create, Edit, Delete)
      - Combo box of available groups ['Edit' or 'Delete']
      - Embedded table of all groups ['View All']
      - Text field and Save button ['Create']
    - New Notice
      - Description (standard text area)
      - Combo box (select group)
      - Manually tag intended students
      - Date Picker (Start Date)
    - Export to PDF

- Date Picker (Select Date)
- Check boxes (Groups to Display)
- Process Button (Export)
    - Opens .pdf in New Tab

- All Notices
    - Embedded table showing all notices on the System
        - Filtering Check Boxes or Text Fields for each column heading
    - Edit field in context menu, producing a text field at the cursor [Staff]
        - Request Button
        - Save Button [Admins]
    - Redirect to notice's own page on double click
        - Vertically positioned details to accommodate width
- My Subscriptions
    - List of current subscriptions, one underneath another
    - Edit Button
        - Check box appears next to current subscriptions
        - Similar list of other groups appears also with check boxes
        - Done button saves and switches back to viewing
- Feed
    - List of notices with personalised notifications, Teaser descriptions
    - Expand to reveal full content on click
        - Vertically positioned details to accommodate width

## MOBILE WEB APP – (STUDENTS)

- Login Page
    - Admin Number Text Input
    - 'View My Notices' button: greyed out until Admin Number valid
- Notices
    - Heading with the Student's Name
    - Collapsible side menu to configure how notices are displayed (All Notices; My Subscriptions; Notifications);
      Notification toggle ("Enable Notifications");
      logout button ("Log Out")
    - Vertical List of {Every notice available on the system ['All Notices'];
      Every notice in the Student's subscription group(s) ['My Subscriptions';
      Every notice in which the Student is tagged ['Notifications'];}
        - List is categorised by Section, then by Group
        - If any notice is tapped, it will direct to its own screen, where expanded details may be seen
        - In no notices are available, text informing the user of this will show instead
    - Return to Landing Screen ['Logout']

## Specifications of Help

### TOOLTIPS

Brief blurbs instructing a user how to use a component correctly will appear on hover (desktop) or on tap and hold (mobile). This will be subtle and non-intrusive.

### INSTRUCTIVE LABELLING

Where necessary, a field will have in its label an indication of the required format and/or range.

### ERROR MESSAGES AND FORMATTING

If an error occurs, a detailed message will inform a user how it arose and how to prevent such an error.

Any fields requiring data validation, such as Date and Time, and Admin Numbers will have visual emphasis whenever a user input is invalid. This will encompass any invalidity not already dealt with input restrictions and instructive labelling.

### EXTERNAL DOCUMENTATION

If required, a user may use the help button on each page to view detailed help information of its functions. This will open in a new tab as an html document or as a modal.

## Specifications of Data Storage

The web-based implementation of my application necessitates the use of a web-based storage system. With a slew of powerful scripting functions and interfacing capabilities between, for instance, websites and databases, a NodeJS Express API with a MySQL database will suit my application's needs.

### USERS

User ID
Rights Type
First name
Surname
Unique Identification (either e-mail or username as specified in the system design document)
Password

### NOTICES

Notice ID
Date
Time
Group
Title
Description
Approval Status

### GROUPS

Group ID
Section ID (based on the 6 currently used categories)
Name

## SECTIONS

Section ID
Name
Sort Order

## SUBSCRIPTIONS

User ID
Group ID

## MENTIONS

Notice ID
User ID

## Hardware and Software Requirements

Angular recommends the use of a Chromium based web browser. I have, therefore, tethered my application's system requirements to those of the most popular option of such a browser, Google Chrome.

### GENERAL

- Application Software
  - Browser which supports HTML 5; JS ECMAScript 2015; CSS3
    - e.g. Google Chrome v51+

- Development Tools and Languages
  - Front-end – HTML 5, TypeScript/JavaScript ECMAScript 2015, CSS 3
  - Backend and Databases – MySQL, NodeJS

### DESKTOP

- Hardware
  - Processor: 2 GHz or faster with SSE2 compatibility; Intel Pentium 4 or later
  - RAM: Minimum 128 MB, Recommended 4GB
  - Secondary Storage: Minimum 500 MB to install Chrome, Minimum 50 MB to store cookies and caches
- Operating System (32- or 64-bit)
  - Windows: 7+
  - Mac: OS X Yosemite 10.10+
  - Linux: 64-bit Ubuntu 14.04+, Debian 8+, openSUSE 13.3+, or Fedora Linux 24+

### MOBILE

- Hardware
  - (iOS: 64-bit) iPhone 5S, iPad Air, iPad Mini 2, iPod Touch 6, iPad Pro, or later models
  - (Android) Processor: 1.2 GHz or faster; Intel Atom Z2520 or later

    RAM: Minimum 512 MB, Recommended 2GB

    Secondary Storage: Between 850 MB and 1.2 GB, depending on the phone language version

    Video: 1280 x 800 pixels or higher
- Operating System (Chrome 50 compatible)
  - iOS: 11.3 or later
  - Android: 4.4 KitKat or later