2022

Josiah Rowden

A-level Project

2/8/2022

Cadet Link

A picture containing text, cup

Description automatically generatedLogo

Description automatically generated with low confidence

Contents

[The Problem: 2](#_Toc104364791)

[Stakeholders 2](#_Toc104364792)

[*Who use the current system and would use the new system and how.* 2](#_Toc104364793)

[Why this Problem is suited to be solvable by computational methods 2](#_Toc104364794)

[Computational methods I will use: 3](#_Toc104364795)

[Problem Decomposition 3](#_Toc104364796)

[Divide and conquer 3](#_Toc104364797)

[The Analysis 4](#_Toc104364798)

[Similar Products 4](#_Toc104364799)

[Meetings 4](#_Toc104364800)

[Surveys 4](#_Toc104364801)

[Requirements 4](#_Toc104364802)

[Success criteria 5](#_Toc104364803)

# The Problem:

Through my experience in a cadet force I have learnt many things, one of these things is communication is verry important, however as highlighted by experience, communication is not always easy for a Cadet force, especially one that is split over two schools such as mine. Getting new kit, communicating about lessons and activities for each week, and receiving information about trips are, despite the efforts of our CFAV’s (Cadet Force Adult Volunteers), less efficient than what would be desirable – especially trying to get new Uniform. These issues are normally not because of Our CFAV’s but mainly due to the amount of paperwork required in these tasks, paperwork that I believe could be streamed lined by computational methods.

In the current system Orders are emailed, Whatsapped or sent through teams. Uniform is kept track of on a manually updated Excel spreadsheet. Communication is generally done through email, and physical meetings although the Senior NCO’s do have an informal group chat on WhatsApp. In this system there is no central space for communication, which limits coordination and collaboration and overall cohesion of the Force.

Our Current system for acquiring kit involves emailing a CFAV, normally Lt Howson our SSI (Senior School Instructor) and quarter master either collectively or individually. This email can get Lost in inboxes and sometimes the CFAV’s are simply to busy to deal with a request. This results in long delays for new kit or in some cases new kit is never issued; this is obviously not ideal.

## Stakeholders

## Who use the current system and would use the new system and how.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Role | Interaction | Availability |
| Cpt Aggrey | User: Contingent commander | Would use to issues orders and post information about trips | Once a week in person, email |
| Lt Howson | User: Quartermaster/SSI | Would use the app to sort out uniform issues and know what Equipment is needed for each session | Once a week in person, email |
| A Cadet | User | Would Use the App to Make Requests and Find orders for next session | For TJWA cadets any school time,  Benenden, once a week, possible email/WhatsApp |

## Why this Problem is suited to be solvable by computational methods

This problem is suited to suited to be solvable by computational methods for a variety of reasons. My planned solution is to use a web-based app that would be accessible to download on any device or be accessed through the internet. Each cadet would have their own log in Information about trips, orders for this week, what kit to bring (and whenever its short sleeves or not), scores for section competitions and a way for uniform issues to be logged would all be displayed in an easy-to-use personalised central display. *The Exiting System is outlined in the first section of The Problem*

The benefits over the existing system would be, a central space for communication, essay use from anywhere (with internet connection), a full computerised database which would allow easy data retrieval, comparison, and entry, removing a lot of human error and allowing for easier management of stores and hopefully lead to cadets getting new kit more sooner.

## Computational methods I will use:

### Problem Decomposition

This Project is verry modular as each feature are mostly independent of each other; this will allow me to tackle each problem individually, hopefully leading to a more complete project even if all planned features are not implemented. As such this also leads to the ability to easily add new features even after launch. As such I have split the purposed features on terms of most to least important:

1. **Database**, **Login and Signup** and **Dashboard** systems
2. **Uniform request** system
3. **Trip manager**, **Section and** **Troop manger** and **Order**s systems
4. The ability to **talk to an NCO**, and any other systems could be to implement

### Divide and conquer

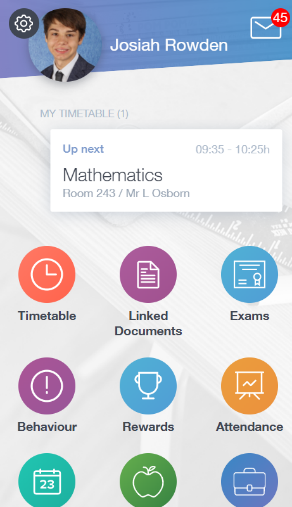
I have split each section into different html/php pages and works on each page till I get it at a degree of functionality that I consider it workable or completely finish it I use the below diagram to help me chose what page to tackle next:

Diagram

Description automatically generated

# The Analysis

## Similar Products

* **Edulink One:** Edulink is a Software that My school uses to set Homework; Promote trips; generally, Communicate. It has a central dashboard which acts as hub for all the features of the App, something that I am planning to replicate in my design.



* **MOD Portal:** *description to be filled in when I have the info to do so*
* **Winchester:**

## Meetings

I have Met with Lt Howson (*Stakeholder*) discussing What he would want In Such a software that I would want to develop. He was Optimistic about the prospect of a web-based app saying “It’s all going to be in one place. It means things can be updated and shared easier and hopefully more people can understand how it works.” And “It's less work for me, which means I can then do other work.” He also gave Me Some other Ideas for updates in that could come later – “If it was to do something to do with bar codes and things, that would be quite useful, I could scan, kit out and then gives me a full running total of each item with each bit of equipment. That'd be quite handy.”

## Surveys

*I have made a Surveys though MS forms and will be Sending this To Clients Soon.*

## Requirements

|  |  |  |
| --- | --- | --- |
| Requirement | Description | Rational |
| Kit Request System. | System Should be able to Manage a Kit Request from a Cadet, Notify and Present the Request to the QM which should them be able to use the system to find the Kit and Issue to the | This part of the System would replace the Current System which has many flaws and is a need if this system is to be Successful |
| Kit Management System | Allows the QM or other CFAV’s see the Stock Numbers and Add or Adjust where needed  And search for specific Stock | Replacement Of existing System: Will be simpler and easer to use than the Existei |
| System User Interface | A User interface that is intuitive and similar to what users | If a Computer System is overly complicated it slows down the function of the users and makes them frustrated which will ultimately lead to the System not getting used. |
| System needs to work on a multitude of devices from Phones to PC’s. | Working meads there are limited issues with elements overlapping – all key data is visible and accessible to be able to use | The system needs to be Regularly available on any device a User might have so it can be used when they need to |
| Account and personalised Dashboard |  |  |
| Event Management System – Similar to Edulink’s timetable System |  | By Implementing such a feature Cadets will have one simple place to look to know what they are doing each training session |

## Success criteria

Using the 4-stage table in the **problem decomposition** section I will grade:

If I only I achieve systems in **group 1** I would consider the project to be a failure as the skeleton of the program would be operational, but it would lack any useable functionality. If I only achieve systems in **group 1** and 2 then the program would be operational with limited functionality, I believe this would be a good beta version and a limited success as the program would be somewhat useful. If I achieve groups **1**,**2** and **3** the program would be operational with a good amount of functionality; I would consider the program largely successful. If I complete all four groups, I would consider the program greatly successful as functionality is achieved to a greater deal than what I have planned and shows the program is useful.

*\* I consider a system complete where it runs with minimal bugs and full planned functionality is operational.*

Other Goals

Other than what Implementing systems my other goals are for the code to be readable and easily understandable, this means extensively commenting code (ahhh!). The rationale behind this goal is as I will not all ways be at my cadet force and when I leave the system will still need to be maintained and if new features are desired, I hope others may be able to add to the program. This would also be useful if other cadet force wants to use the program as it will have to be adaptable to their needs.

## The Design

### **User interface:**

#### **Goals:**

#### **Must be Simple and Intuitive for the average User.**

#### **Must be Adaptable for All Devices**

#### **Use colour scheme that is Linked to User’s background I E army cadets**

##### Pages and flow:

Users will Log in from One central page, regardless of if CFAV or not.

This will Move them to a **Dashboard** where they will be able to see a small amount of relevant data such as the next event and the info for that. This will also be the page where the buttons that link to the function pages such as the Kit Request and the Virtual Stores pages.

Users will be able to go back to the Dashboard from any function pages. Users will not be able to go from one function page to another. It is worth noting that function pages are grouped in related sections, I.E., the page to make a request and see your requests will be in one function group and only has one button on the **Dashboard** that is linked to it; when I refer to a function page from the dashboard assume I am talking about all the pages in that group. There will be pages that Can only be accessed from a function page, I.E., an edit row page of a table, they will link back to their parent function page.

This System is very similar to Edulink and other systems like it; this is intentional as after using said programs I believe the system work well enough. This will help the user navigate My Program if it is similar to something that they are used to.

DFD to Show flow from Login to Function page.

Diagram

Description automatically generated

##### Pages Layout and Style

Like in the Pages and Flow section my design is similar to Edulink and other systems, this is for the same reasons stated previously.

###### Colours

I have Chosen Colours from the Army Cadet Website (see References) as it links My website to the Cadets and it effectively conveys what I want.

|  |  |  |  |
| --- | --- | --- | --- |
| Colour Hex code | Colour Name | Sample | Usage |
| 002B17 | **Dark Green** |  | Will be used for background of generic elements – taken from the army cadets’ website |
| FFFFFF | **White** |  | **Will be Used for Text on elements with Dark Green back grounds and for back grounds of Elements I want to Emphasize** |
| 000000 | **Black** |  | **For text where White isn’t appropriate** |
| F5EA00 | **Yellow** |  | **Used on text and some Elements where I want to highlight them, for instance when a user hovers over a button - taken from the army cadets’ website** |
| 032F52 | **Ben Blue** |  | **One Of the colours of my Cadet force used for dashboard button** |
| AE1E30 | **TJWA Red** |  | **The other colour of my Cadet force used for dashboard button** |

###### Layout

Graphical user interface, application

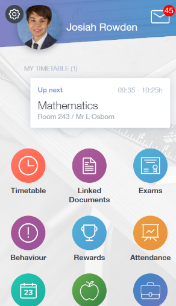
Description automatically generated**Concept of How I want the Dashboard to look – On a Laptop/ PC**

## 

Graphical user interface, chart, bubble chart

Description automatically generatedI want the Dashboard to be Simple, easy to use, yet still display the relevant information. As such I have Lage central body where the user attention will be attracted to that most because of the different colours. This is intentional as it is where almost everything the user will use to interact with the site is located.

1Edulink Computer Dashboard Layout

The Events section will be a table of some Kind with some key information about the displayed event and link to Orders Document. Previous or Later events will be viewed by pressing the respective buttons on the side of the events section.

The Log Out button and the Profile picture Positions are not finalized.

This is again very similar to Edulink One’s Page layout – especially on mobile on mobile, with an events section and Buttons leading to function pages. This is natural as Edulink is one of my main sites I’m taking inspiration from. Where it differs is the Size and number of buttons; I will have significantly less functions Pages than Edulink and as such I can afford to have lager buttons that feel the page, which should help users click the button they wanted first time.

2EduLink’s Mobile Dashboard

**On Mobile: Graphical user interface, text, application, chat or text message

Description automatically generated**

For Usage on mobile, I plan to just scale Everything Down. The Margin for the central body is removed to make space for the largest buttons as possible. This is another reason why having bigger buttons is important.

Most of the function pages will Follow the same Style and layout just adapting it to suit their functions.

##### Database Design

###### Chart, table Description automatically generatedPrevious System to store data

The images to the left are an example of how uniform data is currently stored and managed. The user manually updates each field when needed, excluding totals column that is update by a simple formular: the sum of in store and issued.

There is Some elements of this that works well, it is simple to read and understand: data can be read easily. Most of the columns store data atomically which will be helpful when transferring to a data base. As such I want to use a similar layout when displaying the data to make it more familiar to the user and to uses the features that work well.

Table

Description automatically generatedHowever, there are some issues, for Uniform that does not have three sizes there are blank column. And as the data is entered manually it is prone to human Error.

#### **Goals:**

#### **To Use a Similar Format to Clients Exiting Spreadsheet.**

#### **To Use a relational data base.**

#### **To have all data in 3rd Normal For**

#### **To have a consistent naming Scheme so the data base is easily understood by another Coder**

###### Requirements

* table(s) to store items in stock
* table(s) to store Requests
* table(s) to store users
* table(s) to store Events

###### A screenshot of a computer Description automatically generated with low confidenceDesign:

Diagram 1 Entity Relationships for CadetLink – excluding Events System

Rational

*Notes*

* *Most of the physical Items that are going to be virtually stored in these database tables are Military Uniform and Equipment – as such they all have a same size system (measured in cm) and NATO Stock Number, which is verry helpful. Size listed on the Items are all in centimetres and are formatted like 160/88 or 70/72/88 where each individual numbers represent a dimension like Chest or Inside Leg.*
* *When I Refer to an Item it is generally a piece of uniform but could also be something else military related like a TRF of a water Canteen.*

**Items and Sizes tables**

This table store the Items that the Cadet force is responsible for. This covers Items unreserved in Store, Items Issued, Items Reserved in store, and Items that have been Ordered as represented by their respected column names. There is not a total as in client spreadsheet column as this can be Calculated from the sum of other tables by the program displaying the data if necessary.

I decided that, like what is done on the user’s spreadsheet. I will split sizes from the items in its own format to avoid repeated data. Although it would be relatively easy to Store them together, I believe it would be easier for comparison if they were separated out. For instance, a User can search for an Item with chest of 160 and multiple items would appear; this would be a useful feature for fitting out a new cadet.

They are split into the Item and all the individual sizes that item may have; as such they are connected together with a one-to-many relationship; sizes have a foreign key ItemID which links to the ID of items. Sizes has another foreign key that link to a look up table for the different size types, Height, Chest etc. This was Done so that if I wanted just all the different size types I could have one table with them all. This relationship is one-to-one.

The Item Type table is a look up table for all the different types of Items stored in the Database, it is linked to from the Items and ItemRequest tables.

**ItemRequest and SizesRequest tables**

These tables store Item Requests from users making them through the system. As such they have a UserID foreign Key to denote what user is making the request. This is link to the users table with a one-to-one relationship. The StockID column is another foreign key and this links to the ID column of the ID table; this so that the number columns in Items table can be updated where an appropriate request is made. SizesRequest is to ItemRequest Exactly the same as Sizes is to Items; the reason I decided to have it and not just work off the sizes of the item in items that is linked to by the StockID is to reduce complexity and so a user can request an item with a size that may not be in the database.

The Purpose column is one requested by the Client, they want to make sure that every request is valid, so they requested a reason or purpose for each request be submitted when a user makes a Request.

DateNeeded is planned to be optional, the idea going forward within my cadet force is that Senior NCO’s start using their CFIT skills to actually take some lessons, as it would be helpful to have equipment for Such activities, Item Request is a way of doing that and being able to specify what day you need that equipment is verry handy. It in Date Time format so it can be as specific as possible.

DateRequested is a field that will be automatically filled out when the request is submitted, its purpose is so that the QM or whoever is issuing the Items can see the request that has been waiting the longest and hopefully get to them first.

Status is a column that will tell the user at what point their request is at. Planned Statuses are ISSUED, TO BE ISSUED, ORDERED, AWAITING ORDER. These roughly correspond to the number Columns in the Items table. This is intended so that a Request with an ORDERED status would increase the value NumOrdered in the Items table when made.

**User Table**

The Users table is fairly self-explanatory: Every user will have a Cadet Number and Password that, stored in the Cnum and Pwd column respectively. The Cadet Number will relate to their Winchester Number, which from the MOD Database system for managing cadets that works at a higher level than the scope of CadetLink – this was a feature requested by the Client. The Users Password will be set by a CFAV when setting up the account to a default value, the User will then be asked to change it when they log in for the first time. The value Stored for the Password will be the hashed and salted version of what the user enters

The columns fname, Lname, and rank store the first name, last name, and the Rank of the current user respectively. I decided to split these up so they can be updated easily, although first name and the last name shouldn’t need to be changed if initially correctly entered their rank may. Another reason for keeping the First name and the Last name separate is so for smaller screens I could have the option of only displaying one to save space on the screen.

Troop and section is the Groups the user are of, for example troop: Chard, section: 1. They will both have a separate look up table.

The Profile pic URL stores the location of the users Profile Picture.

**Event Tables:**

The event tables are there to store events and their relevant information.

# References

Army Cadet Website: armycadets.com

# Common abbreviations

* CFIT: Cadet Force Instructional Training
* NCO: Non-commissioned Officer

// notes and at the moment rejected paragraphs

* **A Login and account and personalised Dashboard system** would allow Cadets and CFAV’s to access the information specific to them such as what section or troop they are in, what training they are going to undergo or lead next cadet session, what Kit and equipment they need to bring, dates and times of upcoming trips, and anything else that might be necessary to communicate to cadets without physical presence.
* **A Uniform and Equipment Database** would allow CFAV’s and senior NCO’s who are planning a lesson to know what Equipment is available and book that for their lesson. Would also allow cadets to order new uniform at any time of day or any location, automatically check if their size is in stock and present that data to the Quartermaster so it can be ordered and or issued the next cadet session. The computerised database would be able to run real time comparisons and remove much of the human error inherent to a paper database.

The operation of cadets, like the army itself requires a certain amount of logistical support, The difference is in Cadets we don’t have a whole Corp supporting us and although the efforts of our CFAV’s (Cadet Force Adult Volunteer) keep the process mostly up to date, something sometimes does still get forgotten or missed. Having to wait 4 months for new Kit Is not great, especially when remembrance parade is coming up.

This new app would be able to take a lot of the annoying tasks out of the logistics of cadets. Information about trips, Orders for this week, what Kit to bring (and whenever its short sleeves or not), scores for section competitions and a way for uniform issues to be logged would all be displayed in an easy-to-use central display. Other features like talking to an NCO could be implemented. Each cadet would have their own log in. the uniform issues would be logged, cross referenced against the Quartermaster’s database and would have the option to be displayed in an excel spread sheet.

Solvable by Computational method?

Use algorithms

* Solve the problem consistently
* Efficient
* Completes in reasonable amount of time