# Plastering Management Application

Kyle Kirkby

September 27, 2014

# Contents


**1 Analysis** **5**

1.1 Introduction . . . 5
1.1.1 Client Identification . . . 5
1.1.2 Define the current system . . . 5
1.1.3 Describe the problems . . . 6
1.1.4 Section appendix . . . 7
1.2 Investigation . . . 7
1.2.1 The current system . . . 7
1.2.2 The proposed system . . . 10
1.3 Objectives . . . 10
1.3.1 General Objectives . . . 10
1.3.2 Specific Objectives . . . 10
1.3.3 Core Objectives . . . 10
1.3.4 Other Objectives . . . 10
1.4 ER Diagrams and Descriptions . . . 10
1.4.1 ER Diagram . . . 10
1.4.2 Entity Descriptions . . . 10
1.5 Object Analysis . . . 10
1.5.1 Object Listing . . . 10
1.5.2 Relationship diagrams . . . 10
1.5.3 Class definitions . . . 10
1.6 Other Abstractions and Graphs . . . 10
1.7 Constraints . . . 10
1.7.1 Hardware . . . 10
1.7.2 Software . . . 10
1.7.3 Time . . . 10
1.7.4 User Knowledge . . . 10
1.7.5 Access restrictions . . . 10
1.8 Limitations . . . 10
1.8.1 Areas which will not be included in computerisation . . . 10
1.8.2 Areas considered for future computerisation . . . 10
1.9 Solutions . . . 10
1.9.1 Alternative solutions . . . 10

1.9.2 Justification of chosen solution . . . . . 10

**2 Design** **11**
2.1 Overall System Design . . . . . 12
2.1.1 Short description of the main parts of the system . . . . . 12
2.1.2 System flowcharts showing an overview of the complete system . . . . . 12
2.2 User Interface Designs . . . . . 12
2.3 Program Structure . . . . . 12
2.3.1 Top-down design structure charts . . . . . 12
2.3.2 Algorithms in pseudo-code for each data transformation process . . . . . 12
2.3.3 Object Diagrams . . . . . 12
2.3.4 Class Definitions . . . . . 12
2.4 Prototyping . . . . . 12
2.5 Definition of Data Requirements . . . . . 12
2.5.1 Identification of all data input items . . . . . 12
2.5.2 Identification of all data output items . . . . . 12
2.5.3 Explanation of how data output items are generated . . . 12
2.5.4 Data Dictionary . . . . . 12
2.5.5 Identification of appropriate storage media . . . . . 12
2.6 Database Design . . . . . 12
2.6.1 Normalisation . . . . . 12
2.7 Security and Integrity of the System and Data . . . . . 12
2.7.1 Security and Integrity of Data . . . . . 12
2.7.2 System Security . . . . . 12
2.8 Validation . . . . . 12
2.9 Testing . . . . . 12
2.9.1 Outline Plan . . . . . 13
2.9.2 Detailed Plan . . . . . 13

**3 Testing** **14**
3.1 Test Plan . . . . . 14
3.1.1 Original Outline Plan . . . . . 15
3.1.2 Changes to Outline Plan . . . . . 15
3.1.3 Original Detailed Plan . . . . . 15
3.1.4 Changes to Detailed Plan . . . . . 15
3.2 Test Data . . . . . 16
3.2.1 Original Test Data . . . . . 16
3.2.2 Changes to Test Data . . . . . 16
3.3 Annotated Samples . . . . . 16
3.3.1 Actual Results . . . . . 16
3.3.2 Evidence . . . . . 16
3.4 Evaluation . . . . . 17
3.4.1 Approach to Testing . . . . . 17
3.4.2 Problems Encountered . . . . . 17

3.4.3 Strengths of Testing . . . 17
3.4.4 Weaknesses of Testing . . . 17
3.4.5 Reliability of Application . . . 17
3.4.6 Robustness of Application . . . 17

**4 System Maintenance 18**
4.1 Environment . . . 19
4.1.1 Software . . . 19
4.1.2 Usage Explanation . . . 19
4.1.3 Features Used . . . 19
4.2 System Overview . . . 19
4.2.1 System Component . . . 19
4.3 Code Structure . . . 19
4.3.1 Particular Code Section . . . 19
4.4 Variable Listing . . . 19
4.5 System Evidence . . . 19
4.5.1 User Interface . . . 19
4.5.2 ER Diagram . . . 19
4.5.3 Database Table Views . . . 19
4.5.4 Database SQL . . . 19
4.5.5 SQL Queries . . . 19
4.6 Testing . . . 19
4.6.1 Summary of Results . . . 19
4.6.2 Known Issues . . . 19
4.7 Code Explanations . . . 19
4.7.1 Difficult Sections . . . 19
4.7.2 Self-created Algorithms . . . 19
4.8 Settings . . . 19
4.9 Acknowledgements . . . 19
4.10 Code Listing . . . 19
4.10.1 Module 1 . . . 20

**5 User Manual 21**
5.1 Introduction . . . 22
5.2 Installation . . . 22
5.2.1 Prerequisite Installation . . . 22
5.2.2 System Installation . . . 22
5.2.3 Running the System . . . 22
5.3 Tutorial . . . 22
5.3.1 Introduction . . . 22
5.3.2 Assumptions . . . 22
5.3.3 Tutorial Questions . . . 22
5.3.4 Saving . . . 22
5.3.5 Limitations . . . 22
5.4 Error Recovery . . . 22
5.4.1 Error 1 . . . 22

5.4.2 Error 2 . . . 22
5.5 System Recovery . . . 22
5.5.1 Backing-up Data . . . 22
5.5.2 Restoring Data . . . 22

**6 Evaluation** **23**
6.1 Customer Requirements . . . 24
6.1.1 Objective Evaluation . . . 24
6.2 Effectiveness . . . 24
6.2.1 Objective Evaluation . . . 24
6.3 Learnability . . . 24
6.4 Usability . . . 24
6.5 Maintainability . . . 24
6.6 Suggestions for Improvement . . . 24
6.7 End User Evidence . . . 24
6.7.1 Questionnaires . . . 24
6.7.2 Graphs . . . 24
6.7.3 Written Statements . . . 24

# Chapter 1

# Analysis

## 1.1 Introduction

### 1.1.1 Client Identification

My client is 30 year old plasterer Dan Austin who runs his own plastering business known as DnA Plastering. Dan mainly uses his Toshiba laptop (Dual Core Intel with 6 GB Ram and running Windows 8 64 bit) to do basic tasks such as social networking and receiving/sending emails.

The current system is a paper based method where he records the prices and measurements of the plastering/screening/rendering jobs he undertakes. Dan works in an around the Suffolk/Essex area but occasionally takes on larger jobs further afield in places such as London or Epping. All the recording and calculations are done by Dan himself and does not require additional assistance in completing these tasks but is looking for a digital solution to the organisation problems faced with the current manual paper method.

Dan is looking to introduce a computer based system to replace the current one in order to make keeping track of jobs and pricing up new jobs easier and more efficient. Alongside this he would like to be able to keep information on all of his customers so he can simply search for clients' details and contact information all in one location. He will also be able to look up the jobs that he has done for them to make sending invoices easier and manageable.

### 1.1.2 Define the current system

The current system in place is a paper/notebook based system where details of clients are stored along with prices of jobs and cost of materials needed etc.

The details of the clients include their address, phone number, email, first name and surname. The infromation about the job usually includes the measurements of what needs to be plastered along with how long it will take to complete and if he is taking any labourers to too. Calculations are often also made to work out how much to charge depending on the price he is charging per square meter. This rate often changes depending on the current economy.

Once all the calculations are made, he works out how much the materials are going to cost and also how long it will take him to complete the job. Once all these calculations and prices have been evaluated he notifies the client of the price; when the price is confirmed the job is undertaken.

Finally, Dan writes out an invoice using a standard invoice book purchased from a stationary store to inform the client of the costs and charges of the job. The current folder containing the invoices for his clients is not organised and offers another problem whereby finding information for jobs is difficult due to the inability to search quickly for any given customer.

### 1.1.3 Describe the problems

Problems are plentiful in the current system. One of the main problems is keeping valuable client data from being lost or damaged as there is only one hard copy made in a notebook. Another problem with the notebook is not being able to easily search through the details of all the clients to find specific phone numbers or contact details. Using a computer based system would allow Dan to search through his clients efficiently and allow him to make backups of the valuable client and job data.

**Interview with client**

1. **Study of life?**

   Helelwjfnwejfnwefnwelfnwef

2. **Studywefwefwf of life?**

   fewefwefwefwef

3. **Stuwfwefdy of life?**

   wfjiwenfjiwenfjinwef

### 1.1.4 Section appendix

## 1.2 Investigation

### 1.2.1 The current system

#### Data sources and destinations

There are four main data sources within the current system - The plasterer, the client, the builders merchant and visting the clients job. A client contacts Dan through a phone call placed to Dan's mobile. Sometimes a client may leave Dan a voicemail message if he is too busy to answer the call at that given moment. If this is the case then Dan will get back to the client as soon as possible. Most of the data in the current system will come from the client or the clients job - this data will be the job measurements and the clients contact information. The main data destinations are the forms given to the client i.e the quote and the invoice document.

| Source | Data | Example Data | Destination |
|---|---|---|---|
| Client | Client Contact information First-name - Lastname - phoneNumber - AddrLine1 - Addr-Line2 - AddrLine3 - AddrLine4 - PostCode - Email - JobType | John - Smith - 07809726812 - 15 - Crowley Road - Haverhill - Suf-folk - CB90DJ - john@gmail.com - Plastering Bed-room | Appointment and Client Book. |
| Plasterer | Appointment Time and Place | 16:00 at 15 Crowley Road, Haverhill | Client Calendar or Diary |
| Visiting Job Site | Measurements of Job Size and Mate-rials that need to be purchased | 4m x 5m x 3m = $60m^2$ 10 Bags of Plaster | Work Notebook |
| Plasterers Calcula-tions | Quote for the work that needs doing and agree a date it can be done. | £600, 1 Day, 15th October | Quote written out on paper or agree in person. |
| Plasterers calcula-tions for the mate-rials needed for the job | Quantity of materi-als needed for the job | 25 bags of plaster and 12m of angle beading | Builders Merchant |
| Builders Merchant | A price for the ma-terials needed | £350 for the bags of plaster and angle beading | Plasterer |
| Plasterer | Total cost of the job broken down - cost of parts,labouring and vat. Date of Job | £600 - £350 mate-rials - £50 VAT - 14/08/14 | Client. |

**Algorithms**

There are three main algorithms utilised in the current system. The first is an algorithm to agree the price of the job with the client.

**Algorithm 1** Agreeing a price Algorithm

```
1: SET agreed TO false
2: WHILE agreed = False DO
3:     IF "Client does not agree with quoted price" THEN
4: Discuss price and change quote if new price is agreed upon.
5:     ELSE
6:         SET agreed TO true
7: Arrange a date for the work to be started on.
8:     END IF
9: END WHILE
```

The second algorithm currently being used in the system is an algorithm used to calculate the price for the job.

**Algorithm 2** Calculating the price

```
1: SET pricePerSquareMeter TO 10
2: SET calculated TO false
3: WHILE calculated = false DO
4:     SEND "Please enter the measurements of the job." TO DISPLAY
5:     RECEIVE measurements FROM KEYBOARD Calculate the
   square meter figure of the job using measurements
6:     SET calculatedMeterPrice TO squareMeterFigure *
   pricePerSquareMeter
7: END WHILE
```

**Data flow diagram**

**Input Forms, Output Forms, Report Formats**

### 1.2.2 The proposed system

**Data sources and destinations**

**Data flow diagram**

**Data dictionary**

**Volumetrics**

## 1.3 Objectives

### 1.3.1 General Objectives

### 1.3.2 Specific Objectives

### 1.3.3 Core Objectives

### 1.3.4 Other Objectives

## 1.4 ER Diagrams and Descriptions

### 1.4.1 ER Diagram

### 1.4.2 Entity Descriptions

## 1.5 Object Analysis

### 1.5.1 Object Listing

### 1.5.2 Relationship diagrams

### 1.5.3 Class definitions

## 1.6 Other Abstractions and Graphs

## 1.7 Constraints



### 1.7.1 Hardware

### 1.7.2 Software

### 1.7.3 Time

### 1.7.4 User Knowledge

# Chapter 2

# Design

## 2.1 Overall System Design

### 2.1.1 Short description of the main parts of the system

### 2.1.2 System flowcharts showing an overview of the complete system

## 2.2 User Interface Designs

## 2.3 Program Structure

### 2.3.1 Top-down design structure charts

### 2.3.2 Algorithms in pseudo-code for each data transformation process

### 2.3.3 Object Diagrams

### 2.3.4 Class Definitions

## 2.4 Prototyping

## 2.5 Definition of Data Requirements

### 2.5.1 Identification of all data input items



### 2.5.2 Identification of all data output items

### 2.5.3 Explanation of how data output items are generated

### 2.5.4 Data Dictionary

### 2.5.5 Identification of appropriate storage media

### 2.9.1 Outline Plan

| Test Series | Purpose of Test Series | Testing Strategy | Strategy Rationale |
|---|---|---|---|
| Example | Example | Example | Example |

### 2.9.2 Detailed Plan

| Test Series | Purpose of Test | Test Description | Test Data | Test Data Type (Normal/ Erroneous/ Boundary) | Expected Result | Actual Result | Evidence |
|---|---|---|---|---|---|---|---|
| Example | Example | Example | Example | Example | Example | Example | Example |

# Chapter 3

# Testing

## 3.1 Test Plan

### 3.1.1 Original Outline Plan

| Test Series | Purpose of Test Series | Testing Strategy | Strategy Rationale |
|---|---|---|---|
| Example | Example | Example | Example |

### 3.1.2 Changes to Outline Plan

| Test Series | Purpose of Test Series | Testing Strategy | Strategy Rationale |
|---|---|---|---|
| Example | Example | Example | Example |

### 3.1.3 Original Detailed Plan

| Test Series | Purpose of Test | Test Description | Test Data | Test Data Type (Normal/ Erroneous/ Boundary) | Expected Result | Actual Result | Evidence |
|---|---|---|---|---|---|---|---|
| Example | Example | Example | Example | Example | Example | Example | Example |

### 3.1.4 Changes to Detailed Plan

| **Test Series** | **Purpose of Test** | **Test Description** | **Test Data** | **Test Data Type (Normal/ Erroneous/ Boundary)** | **Expected Result** | **Actual Result** | **Evidence** |
|---|---|---|---|---|---|---|---|
| Example | Example | Example | Example | Example | Example | Example | Example |

## 3.2 Test Data

### 3.2.1 Original Test Data

### 3.2.2 Changes to Test Data

## 3.3 Annotated Samples

### 3.3.1 Actual Results

### 3.3.2 Evidence

## 3.4 Evaluation

### 3.4.1 Approach to Testing

### 3.4.2 Problems Encountered

### 3.4.3 Strengths of Testing

### 3.4.4 Weaknesses of Testing

### 3.4.5 Reliability of Application

### 3.4.6 Robustness of Application

# Chapter 4

# System Maintenance

## 4.1 Environment

### 4.1.1 Software

### 4.1.2 Usage Explanation

### 4.1.3 Features Used

## 4.2 System Overview

### 4.2.1 System Component

## 4.3 Code Structure

### 4.3.1 Particular Code Section

## 4.4 Variable Listing

## 4.5 System Evidence

### 4.5.1 User Interface

### 4.5.2 ER Diagram

### 4.5.3 Database Table Views

### 4.5.4 Database SQL

### 4.5.5 SQL Queries

## 4.6 Testing

### 4.10.1 Module 1

# Chapter 5

# User Manual

## 5.1 Introduction

## 5.2 Installation

### 5.2.1 Prerequisite Installation

#### Installing Python

#### Installing PyQt

#### Etc.

### 5.2.2 System Installation

### 5.2.3 Running the System

## 5.3 Tutorial

### 5.3.1 Introduction

### 5.3.2 Assumptions

### 5.3.3 Tutorial Questions

#### Question 1

#### Question 2



### 5.3.4 Saving

### 5.3.5 Limitations

## 5.4 Error Recovery

### 5.4.1 Error 1

# Chapter 6

# Evaluation

## 6.1 Customer Requirements

### 6.1.1 Objective Evaluation

## 6.2 Effectiveness

### 6.2.1 Objective Evaluation

## 6.3 Learnability

## 6.4 Usability

## 6.5 Maintainability

## 6.6 Suggestions for Improvement

## 6.7 End User Evidence

### 6.7.1 Questionnaires

### 6.7.2 Graphs

### 6.7.3 Written Statements