B

**A2 Computing Project**

**Building Job Invoice Generator**

****

**Ted Eriksson**

Contents

[Analysis 5](#_Toc283891926)

[Background 5](#_Toc283891927)

[Current system 5](#_Toc283891928)

[Problem definition 5](#_Toc283891929)

[So what? 5](#_Toc283891930)

[Feasibility analysis 6](#_Toc283891931)

[Fact finding 6](#_Toc283891932)

[Requirements analysis 7](#_Toc283891933)

[Requirements specification 8](#_Toc283891934)

[DFDs 9](#_Toc283891935)

[Level 0 9](#_Toc283891936)

[ERDs 9](#_Toc283891937)

[Research of alternative solutions 9](#_Toc283891938)

[Realistic appraisal of feasibility 9](#_Toc283891939)

[Justification of chosen solution 10](#_Toc283891940)

[Agreed system objectives and scope 10](#_Toc283891941)

[Design 10](#_Toc283891942)

[Outline system design 10](#_Toc283891943)

[System flow chart 10](#_Toc283891944)

[User interface design 11](#_Toc283891945)

[UI Overview 11](#_Toc283891946)

[Login and Daily Form 12](#_Toc283891947)

[New Job 13](#_Toc283891948)

[Results 14](#_Toc283891949)

[Edit Details 15](#_Toc283891950)

[Hardware specification 16](#_Toc283891951)

[Program structure 16](#_Toc283891952)

[Hierarchy charts 16](#_Toc283891953)

[Structure charts 16](#_Toc283891954)

[Design data dictionary 16](#_Toc283891955)

[Object diagrams and class definitions 16](#_Toc283891956)

[Data structures 16](#_Toc283891957)

[File organisation 16](#_Toc283891958)

[Entity-Relationship diagram 16](#_Toc283891959)

[Normalised database tables 16](#_Toc283891960)

[Algorithms 16](#_Toc283891961)

[Queries 16](#_Toc283891962)

[Detailed design of printed output 16](#_Toc283891963)

[Preliminary test plan 16](#_Toc283891964)

[Detailed test data 16](#_Toc283891965)

[Implementation 16](#_Toc283891966)

[Coding 16](#_Toc283891967)

[Testing 17](#_Toc283891968)

[Test strategy 17](#_Toc283891969)

[Test plan 17](#_Toc283891970)

[Test evaluation 17](#_Toc283891971)

[Test evidence/results 17](#_Toc283891972)

[Implementation 17](#_Toc283891973)

[Maintenance 17](#_Toc283891974)

[User Guide 17](#_Toc283891975)

[Appraisal 17](#_Toc283891976)

[Appendices 17](#_Toc283891977)

[Interview transcripts 17](#_Toc283891978)

[Questionnaires 17](#_Toc283891979)

[A.1 - COMPANY Background Questionaire 17](#_Toc283891980)

[Original system documents 18](#_Toc283891981)

[Source code 18](#_Toc283891982)

[Test data 18](#_Toc283891983)

[Acknowledgements 18](#_Toc283891984)

# Analysis

## Background

The company was founded on January 1st 1995, and later taken into a limited company in March 2006. The company currently has two full time employees and one to fifteen subcontractors at a time. The annual turnover for the company is approximately £250,000. It is based near Hailsham, East Sussex. The business provides general building work from a kitchen to oak-framed barns.

## Current system

At present Eriksson Building Services ltd use spreadsheets to work out the pay for each worker and the income and outgoings for the business.

Eriksson Building Services ltd employs up to 15 people at any one time, at the end of the day each workers’ hours and the task they were doing must be logged and stored in a database. Also any material bought for the jobs must be logged as well as where it was bought from so as to work out the running costs as well as the quarterly tax returns for the business.

|  |  |
| --- | --- |
| Pros | Cons |
| The user knows how to use this system  Spreadsheets are easily printable | They take a long time to fill out  Information has to be inputted several times in order for everything to correspond |

## Problem definition

Currently the company uses spreadsheets to input this data; this has to be done manually and requires some training in order for them to use it correctly. If information is inputted incorrectly then financial errors could be made causing a customer to lose their faith in the company therefore damaging the company’s reputation. If the user does have training, there is still a high chance for human error, so each record is checked for mistakes. This is very time consuming and is a problem especially when the company needs to keep to tight deadlines.

### So what?

What’s the problem with the current system?

* Using spreadsheets
* They have to be completed manually
* They require training to be of any use
* Only specific people with the training can get information to and from them
* Information could be inputted incorrectly
* Company records will be incorrect
* Financial errors made
* Customer loses faith in company
* Company reputation is tarnished

So what if the user has the training?

* It still takes a long amount of time to input data into spreadsheets
* User has to spend time checking records for errors
* This time could be better spent doing other work for the customer
* Deadline missed to inadequate system
* Customer dissatisfied
* Company reputation is tarnished

## Feasibility analysis

The problem is better solved by a computer because:

* if the system was paper based data could easily be lost
* writing data down would be a very long laborious task
* hard to backup the data

A new system would be feasible because:

* currently the repetitive nature of putting data into the spreadsheets takes a larger amount of time than it would with a more streamlined system

## Fact finding

I compiled a questionnaire to find out more about the company background (see A.1 in the appendices)

I also had an informal and formal meeting to discuss the current system and problem definition.

## Requirements analysis

|  |  |
| --- | --- |
| MoSCoW |  |
| Mo | Must have |
| S | Should have |
| Co | Could have |
| W | Would like |

Key:

FML2 = Formal meeting 2

F = Functional

S = Security

Rel= Reliability

P = Performance

R = Reporting

Pr = Processing

1 done

2 part done

3 not done but need to

4 Nahhh

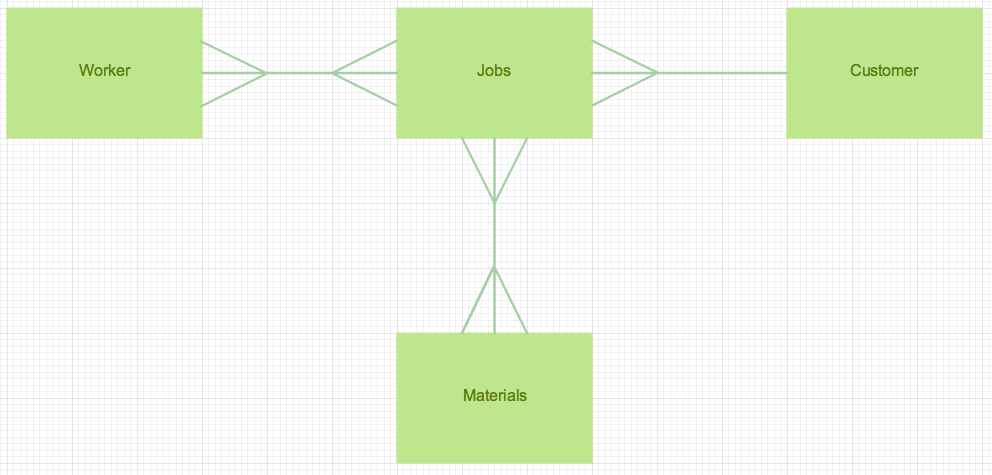
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Reference | Requirement | Source | Priority | Status |
| F1 | The system shall be able to use a drop down list for workers within the daily data form | FML2 | M | 1 |
| F2 | The system shall be able to input the amount of hours worked by a worker | FML2 | M | 1 |
| F3 | The system shall be able to input the wage of each worker | FML2 | M | 1 |
| F4 | The system shall be able to remember the details (name, address) of each worker | FML2 | M | 1 |
| F5 | The system shall be able to input what the worker was doing on a specific day | FML2 | M | 1 |
|  | The system shall be able to input the cost of materials bought for a job | FML2 | M | 1 |
| R1 | The system shall be able to easily locate all past data inputted into the system | FML2 | M | 3 |
| R2 | The system shall be able to view previous data in an easy to read format | FML2 | M | 3 |
| Pr1 | The system shall be able to to export data to a spreadsheet program (.xls) | FML2 | S | 3 |
| F6 | The system shall be able to log material costs separately from workers | FML3 | M | 1 |
| R3 | The system shall be able to give a VAT return report | FML3 | M | 3 |
| R4 | The system shall be able to calculate CIS return | FML3 | M | 4 |
| R5 | The system shall be able to produce a report containing information about the progress of a job | FML3 | M | 2 |
| R6 | The system shall be able to produce an invoice for a customer | FML3 | M | 3 |
| R7 | The system shall be able to produce an quote for a customer | FML3 | M | 3 |
| R8 | The system shall be able to be able to work out how accurate a quote is/was | FML3 | M | 4 |
| P1 | The system will run |  |  |  |
|  |  |  |  |  |

## Requirements specification

## DFDs

### Level 0

## ERDs



## Research of alternative solutions

Spreadsheet Software: This is sometimes a very good solution but spreadsheets can be confusing to new users and difficult and time consuming to use.

A paper based solution: This would be very impractical as it would require constant maintenance as well as taking up a massive amount of room as every item would have to be stored. Also it would be incredibly difficult to back up.

Business Management Software: These are powerful programs, but while they have a lot of functionality, they are expensive. Also in a bid to make them seem easier to use they do a lot of work behind the scenes, which can make them difficult for some users to trust.

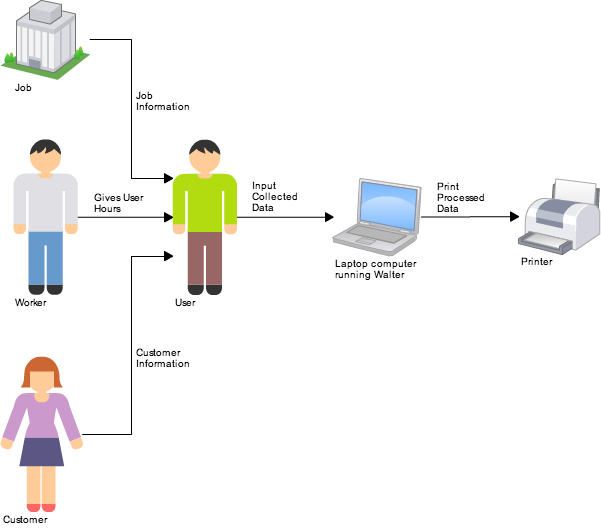
## Realistic appraisal of feasibility

## Justification of chosen solution

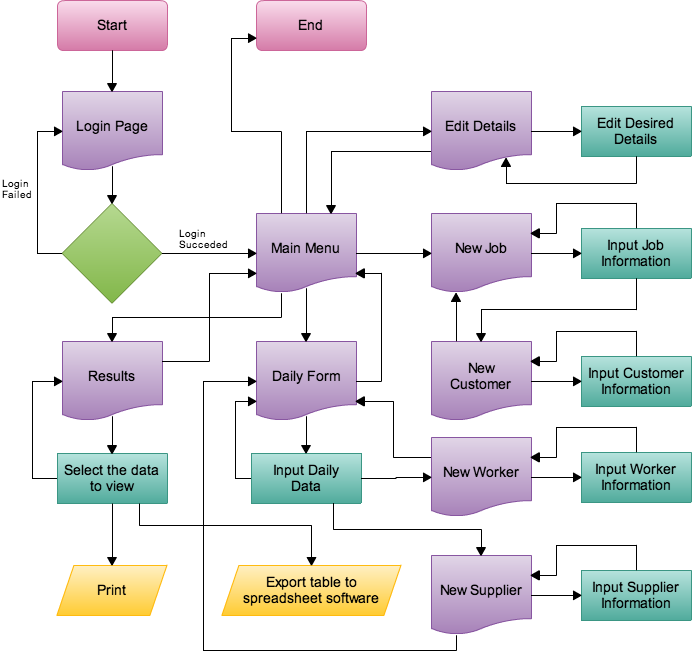
## Agreed system objectives and scope

# Design

## Outline system design

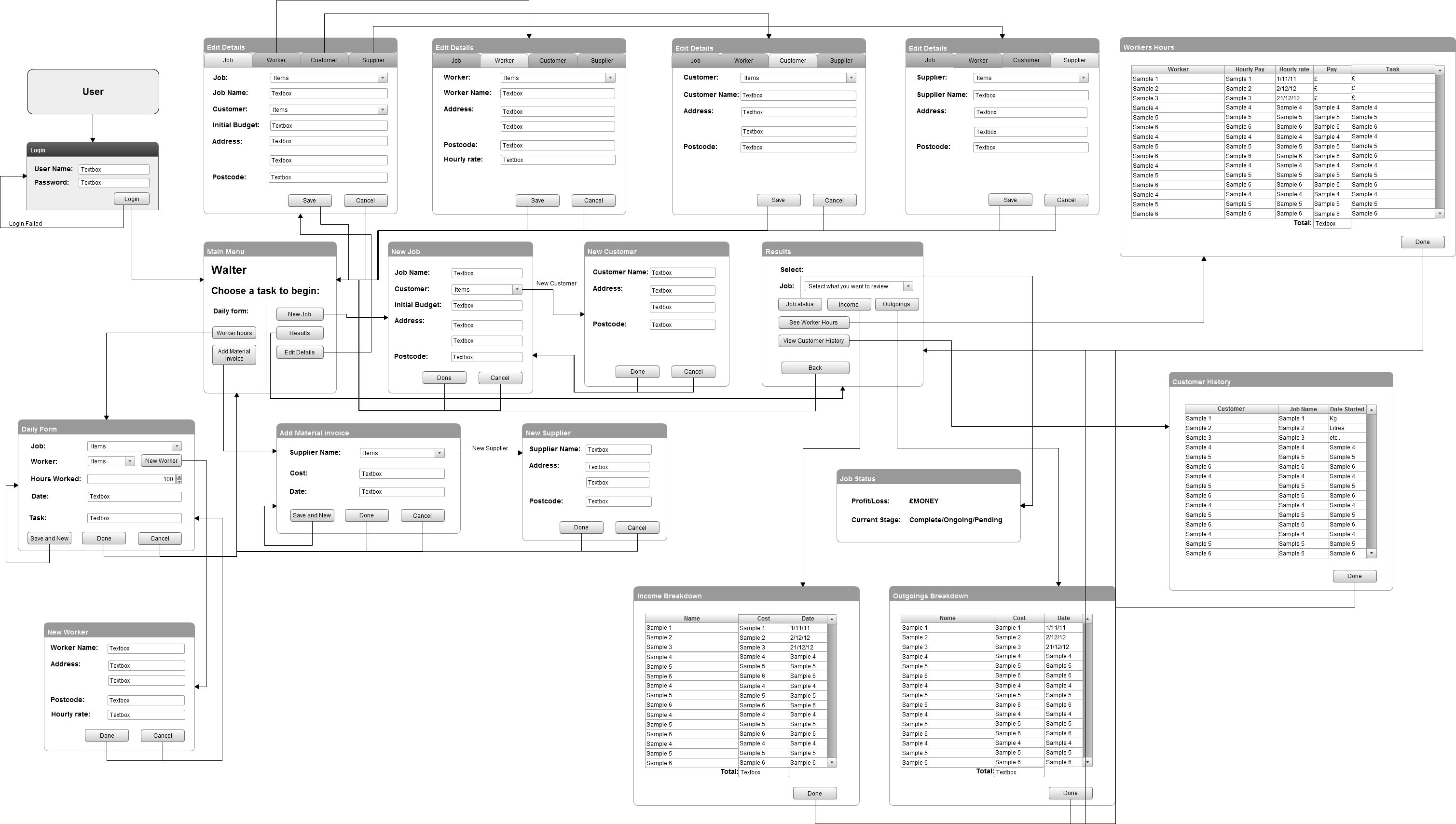


## System flow chart



## User interface design

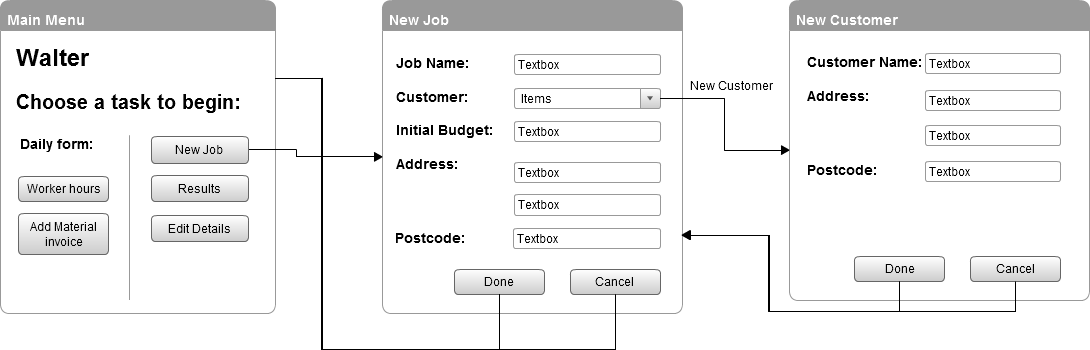
### UI Overview



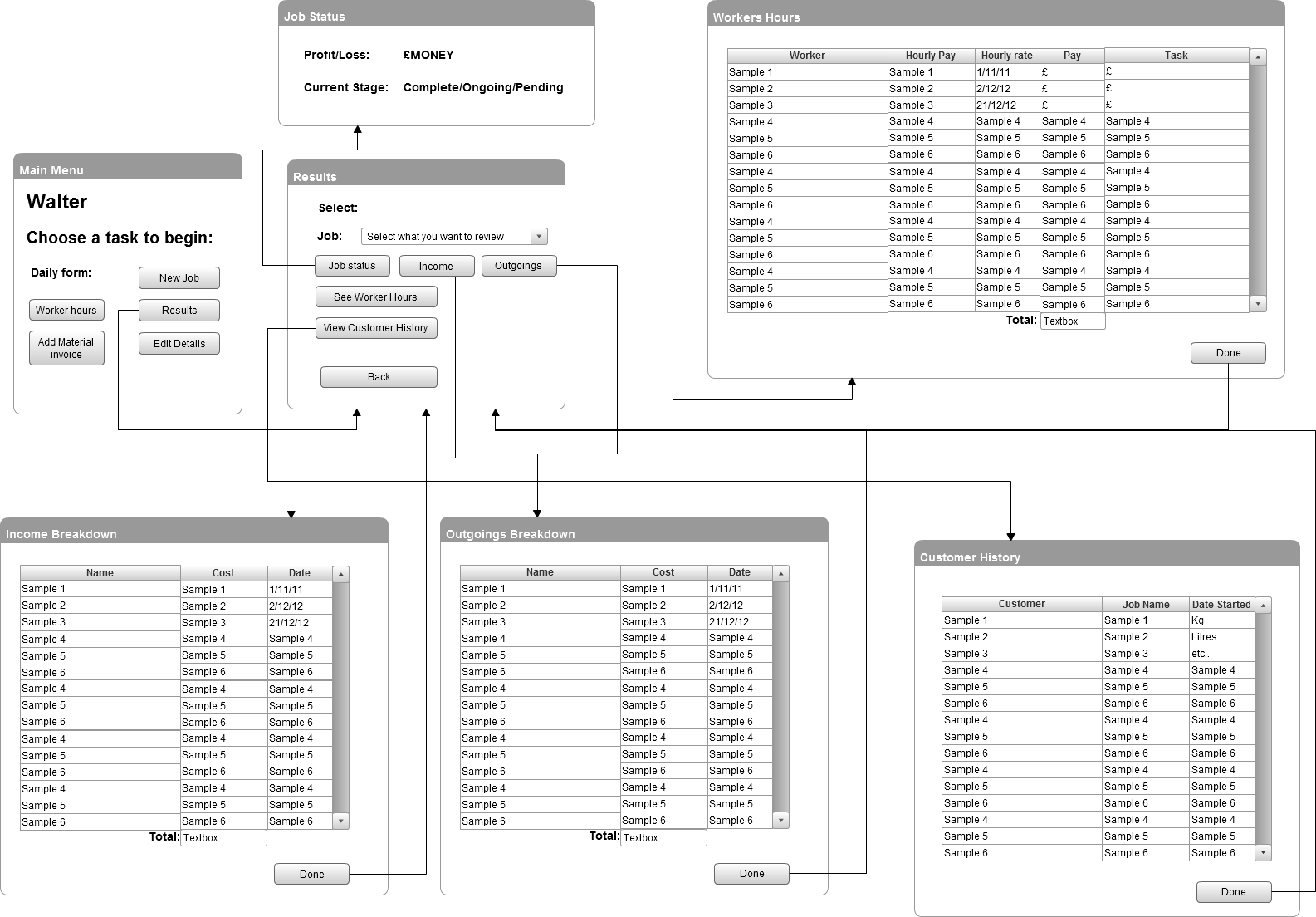
### Login and Daily Form



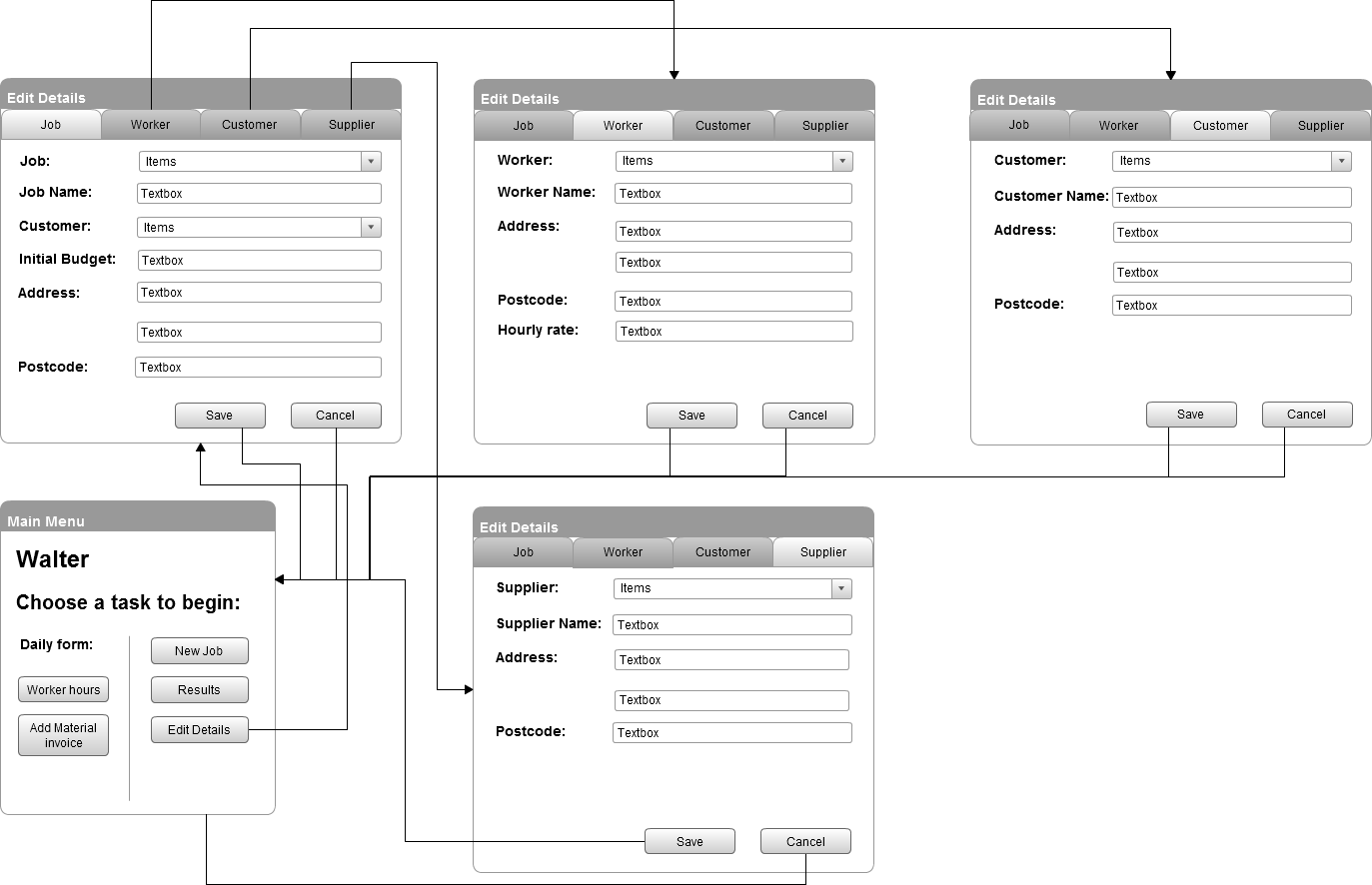
### New Job



### Results



### Edit Details



## Hardware specification

## Program structure

## Hierarchy charts

## Structure charts

## Design data dictionary

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Table | Fieldname | Contents | Type | Format | Range | Required | PK/FK | Reference |
| Jobs | Job\_ID  Job\_Name  CustomerID  Initial\_Budget  Address1  Address2  Postcode | ID for a job  Name of the job  ID for a Customer  Initial budget for a job  Address line 1 for a job  Address line 2 for a job  Postcode of a jobs address | Number  Text  Number  Currency  Text  Text  Text |  |  |  |  |  |
| Customer | CustomerID  Customer\_Name  Address1  Address2  Postcode | ID for a customer  Name of a Customer  Address line 1 for a customer  Address line 2 for a customer  Postcode for a customer | Number  Text  Text  Text  Text |  |  |  |  |  |
| Worker | WorkerID  Worker\_Name  Address1  Address2  Postcode  Hourly\_Rate |  | Number  Text  Text  Text  Text  Text |  |  |  |  |  |
| Worker\_hours | ID  WorkerID  JobID  Date\_worked  Hours\_worked |  | Number  Number  Number  Date/Time  Number |  |  |  |  |  |
| Supplier | SupplierID  Supplier\_Name  Address1  Address2  Postcode |  | Number  Text  Text  Text  Text |  |  |  |  |  |
| Supplier\_outgoings | ID  SupplierID  JobID  Cost  Date\_of\_invoice |  | Number  Number  Number  Currency  Date/Time |  |  |  |  |  |
| Workers\_hours Query |  |  |  |  |  |  |  |  |

## Object diagrams and class definitions

## Data structures

## File organisation

## Entity-Relationship diagram

## Normalised database tables

## Algorithms

## Queries

## Detailed design of printed output

## Preliminary test plan

## Detailed test data

# Implementation

## Coding

# Testing

## Test strategy

## Test plan

## Test evaluation

## Test evidence/results

# Implementation

# Maintenance

# User Guide

# Appraisal

# Appendices

## Interview transcripts

## Questionnaires

### A.1 - COMPANY Background Questionaire

#### Q: When and how was the company established?

A: It was a partnership that started January 1st 1995, and taken into a limited company in March 2006.

#### Q: How many employees are currently working for Eriksson Building Services LTD?

A: Only two employees, the rest (1-15 at a time) are sub contractors.

#### Q: What is the annual Turnover for the company?

A: Approximately £250,000

#### Q: Where is the company based?

A: the company is based at apple barn, old road, magham down, bn27 1pr

#### Q: In a nutshell, what service does the company provide?

A: General building services, anything from a kitchen to an oak-framed barn.

## Original system documents

## Source code

## Test data

## Acknowledgements