

APPENDICES

Table of Contents

Appendix A – System Flow Diagrams..... 2

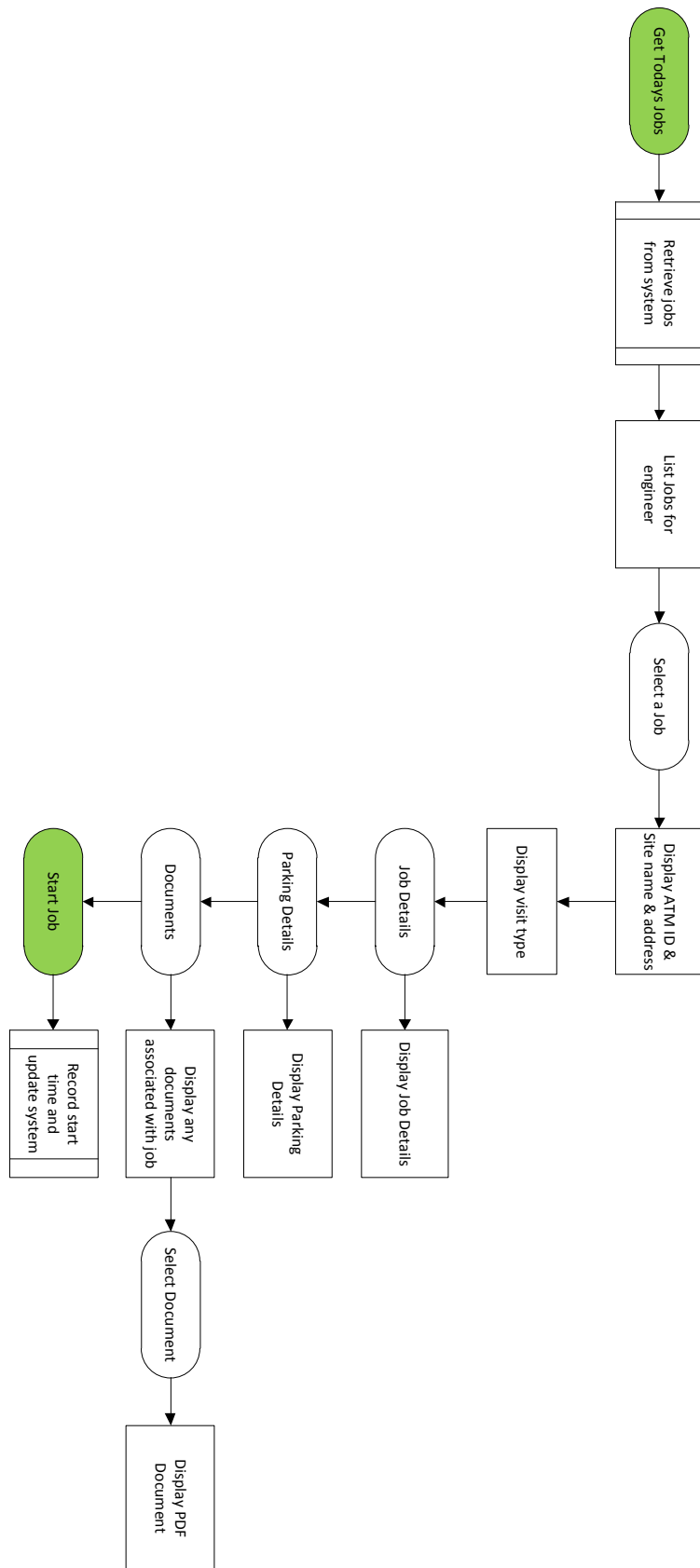
Appendix B – Barcode Examples..... 5

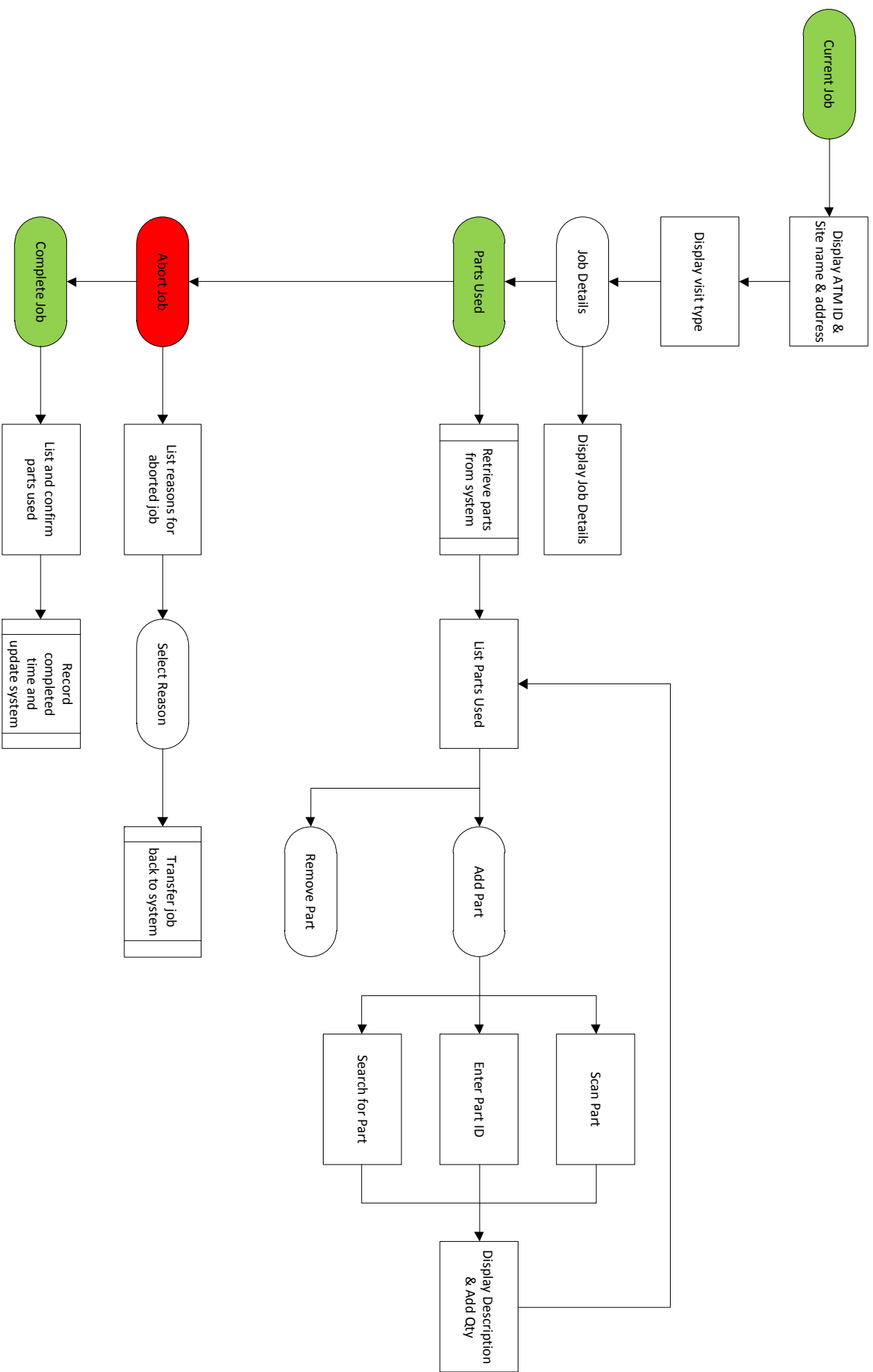
Appendix C – Prototype Screenshots..... 6

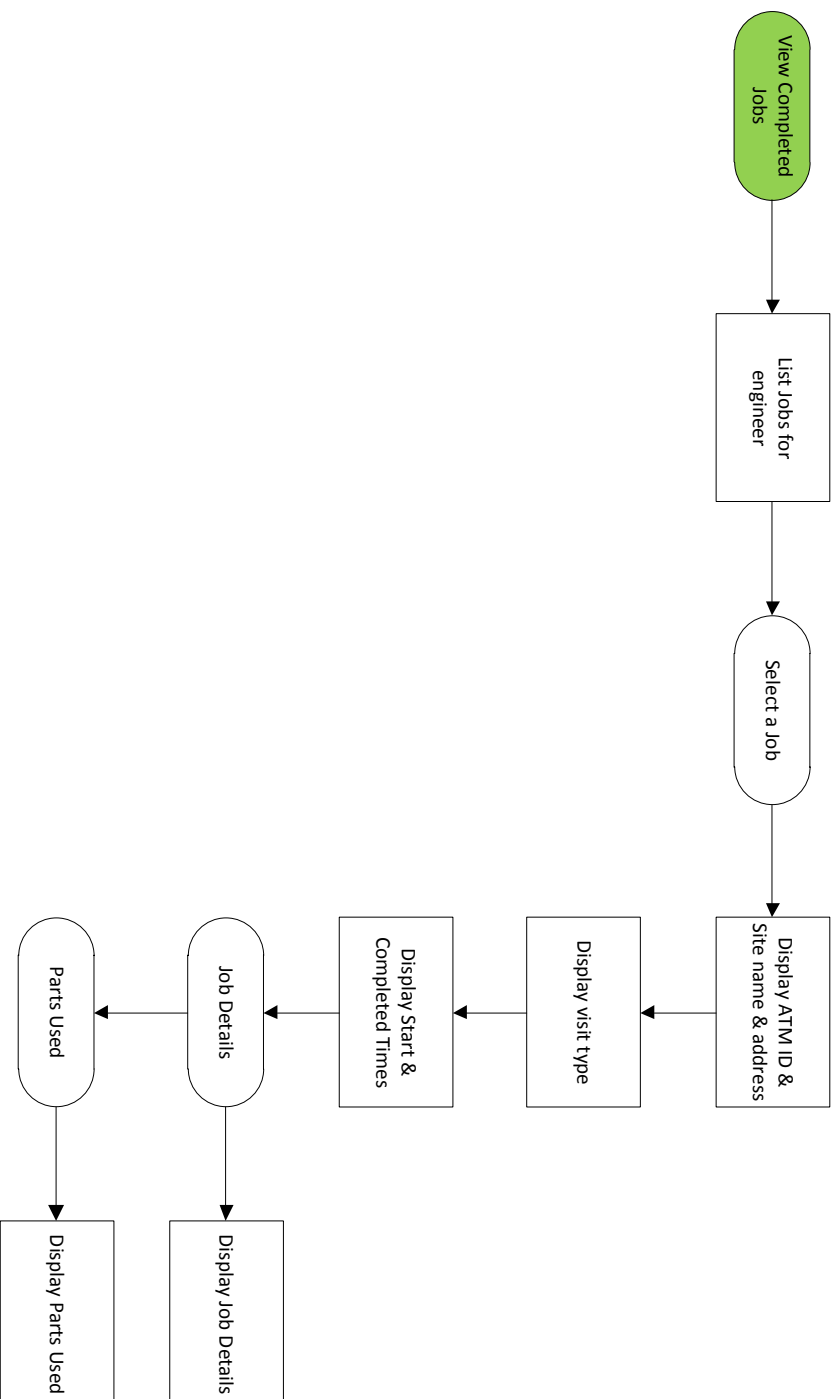
Appendix D – Implementation snippet from ZBar Prototype..... 8

Appendix E – Updated Timeplan 10

Appendix A – System Flow Diagrams







Appendix B – Barcode Examples

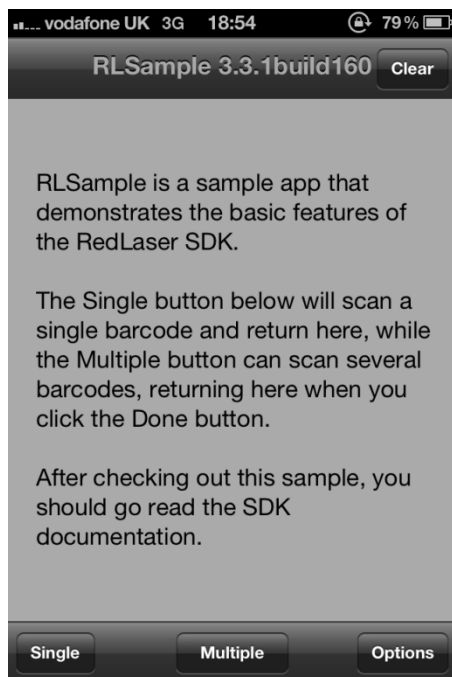


[Figure 1] Two examples of the larger style barcode labels.

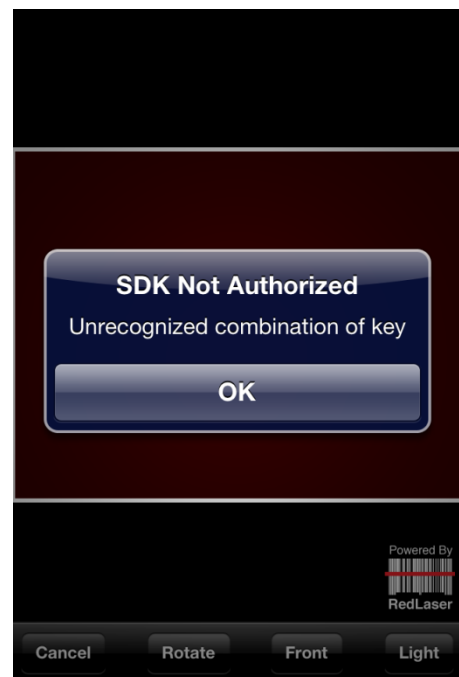


[Figure 2] An example of the smaller style barcode label.

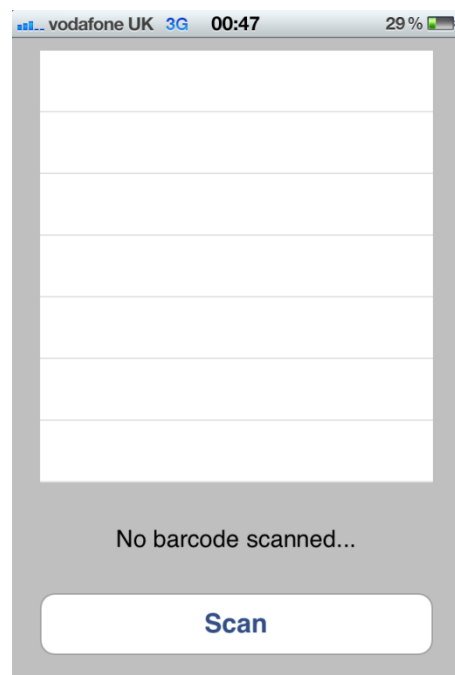
Appendix C – Prototype Screenshots



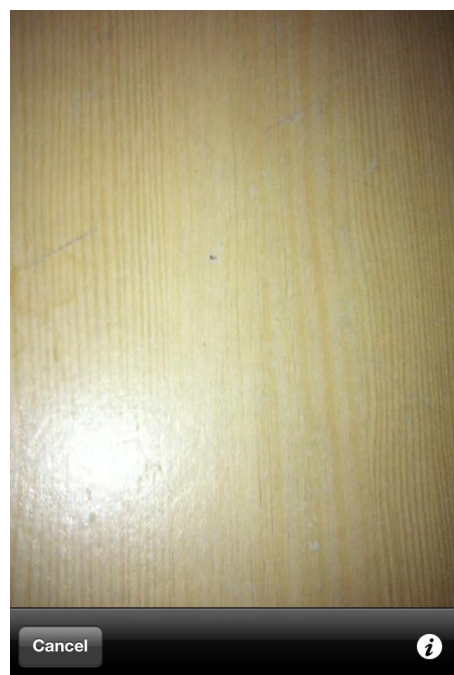
[Figure 1] The RedLaser sample application's main screen.



[Figure 2] An example of a common error message displayed by the SDK.



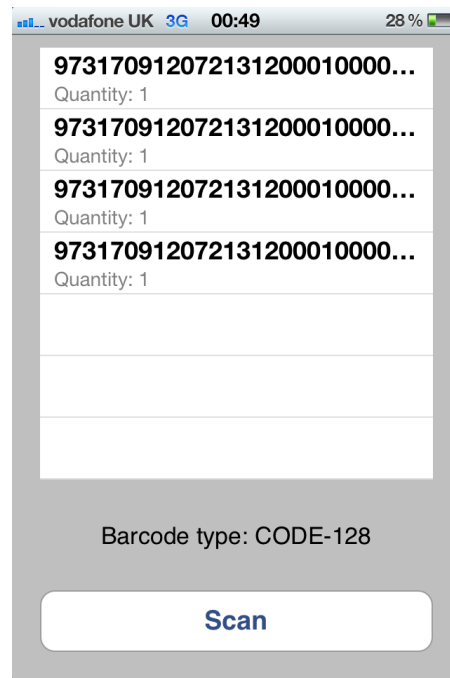
[Figure 3] The ZBar prototypes main screen.



[Figure 4] The scan page, a live update from the camera is shown.



[Figure 5] When a barcode is detected the scanner illustrates this with a green box.



[Figure 6] All scanned codes are added to the list. The type of barcode scanned is displayed above the Scan button.

Appendix D – Implementation snippet from ZBar Prototype

ReaderSampleViewController.m

```
#import "ReaderSampleViewController.h"

@implementation ReaderSampleViewController

@synthesize viewContents;
@synthesize scannedList;
@synthesize resultText;

- (void)viewDidLoad {
    [super viewDidLoad];
    self.viewContents = [[NSMutableArray alloc] init];
}

// Customize the number of rows in the table view.
- (NSInteger)tableView:(UITableView *)tableView
    numberOfRowsInSection:(NSInteger)section {
    return [self.viewContents count];
}

// Customize the appearance of table view cells.
- (UITableViewCell *)tableView:(UITableView *)tableView
    cellForRowAtIndexPath:(NSIndexPath *)indexPath {

    static NSString *CellIdentifier = @"Cell";

    UITableViewCell *cell = [tableView
        dequeueReusableCellWithIdentifier:CellIdentifier];
    if (cell == nil) {
        cell = [[UITableViewCell alloc]
            initWithStyle:UITableViewCellStyleSubtitle
            reuseIdentifier:CellIdentifier];
    }

    // Configure the cell.
    cell.textLabel.text = [self.viewContents objectAtIndex: [indexPath row]];
    cell.detailTextLabel.text = @"Quantity: 1";

    return cell;
}

- (IBAction) scanButtonTapped
{
    ZBarReaderViewController *reader = [ZBarReaderViewController new];
    reader.readerDelegate = self;
    reader.supportedOrientationsMask = ZBarOrientationMask(UIInterfaceOrientationPortrait);

    // Disable all other barcode types other than Code 128
    ZBarImageScanner *scanner = reader.scanner;
    [scanner setSymbology: 0
        config: ZBAR_CFG_ENABLE
        to: 0];
    [scanner setSymbology: ZBAR_CODE128
        config: ZBAR_CFG_ENABLE
        to: 1];

    // present and release the controller
    [self presentViewController: reader
        animated: YES];

    [reader release];
}

- (void) imagePickerController: (UIImagePickerController*) reader
    didFinishPickingMediaWithInfo: (NSDictionary*) info
{
    id<NSFastEnumeration> results =
        [info objectForKey: ZBarReaderControllerResults];
    ZBarSymbol *symbol = nil;
}
```



```

        for(symbol in results)
            // Just grab the first barcode
            break;

        // Set the text above the button to the barcode type
        NSString *type = @"Barcode type: ";
        resultText.text = [type stringByAppendingString: symbol.typeName];

        // Add the barcode data to the table view
        [self.viewContents addObject:symbol.data];
        [self.scannedList reloadData];

        [reader dismissModalViewControllerAnimated: YES];
    }

- (BOOL) shouldAutorotateToInterfaceOrientation: (UIInterfaceOrientation) orient
{
    return(YES);
}

- (void) dealloc {
    self.resultText = nil;
    [scannedList release];
    [super dealloc];
}

- (void)viewDidUnload {
    self.viewContents = nil;
    [self setScannedList:nil];
    [super viewDidUnload];
}
@end

```

Appendix E – Updated Timeplan

January

Christmas Break

Exam Period

Exam Period

Week 1 – Set up an SQL Server database and continue development of the web services prototype.

February

Week 2 – Storyboard the UI of the application.

Week 3 – Begin implementation of the job lifecycle framework.

Week 4

Week 5 – **Meet with client** (demonstrate app and get feedback)

March

Week 6 – Begin implementation of the scanning feature.

Week 7

Week 8

Easter Break – **Meet with client** (demonstrate app and get feedback)

April

Easter Break – Convert dummy data to live database data via web services.

Easter Break

Week 9 – **Meet with client** (final deliverable)

Week 10

May

Week 11 – Deadline for the final report.