# InvoiceGen – Requirements and design specifications

## Purpose

To provide a simple system of generating monthly invoices, as well as generic invoices, as spreadsheets, and to email them to a recipient. Will also be capable of storing historical records, so they can be generated again.

## Overview and requirements

The application will be capable of saving a list of items entered by the user into the form as a spreadsheet.

The application will be able to save the spreadsheet to disk, as well as send it via email to a nominated recipient.

The application will be able to keep a history of invoices in a database or similar, so they can be retrieved and viewed, and the spreadsheets or emails generated again.

### Data Flow

A picture containing bird

Description automatically generated

Figure 1: Level-0 Data Flow Diagram

A picture containing game

Description automatically generated

Figure 2: Level-1 Data Flow Diagram

## Architecture design

The system will be comprised of a single, portable desktop application. It will be written in C# .NET using WinForms.

The application will be structured as a Model-View-Presenter (MVP) design. This pattern is known to be a sound choice for WinForms applications, and provides a high degree of testability, especially for the user interface. This is important for this application, as it is anticipated that the behaviour of some of the user interface controls will be complex enough that manual testing may not uncover as many bugs as with unit testing.

## Data design

Invoice data will be stored on disk as an XML file, with entries for each invoice, and the items contained within.

In the application, classes will be used to represent Invoices, as well as each item within an invoice.

## Interface design

A screenshot of a social media post

Description automatically generated

Figure 3: Screenshots

## Use cases

### 6.1 New Invoice

Table New Invoice Use Case

|  |  |
| --- | --- |
| Main success scenario | 1. “New” invoice button clicked. Monthly controls in “Title” group are enabled. 2. If a valid title is entered, all controls except the “Add” button in the “New Item” group are enabled. 3. When the “Description” and “Amount” and “Quantity” entries have valid values, the “Add” item button is enabled. 4. When the “Add” button is clicked, the item is added to the item ListView. If the item already exists, the quantity is updated. Otherwise, a new entry is added for the item. Repeat as many times as required. |
| Variation (alternative scenario) #1 – “Cancel “button clicked. | 1. In steps 2-4, the “Cancel” button is clicked. The “View or Generate” tab of the main window is returned to the “ready” state. The status bar colour is set to grey, and the status bar text is set to “Ready”. |
| Variation (alternative scenario) #2 – “Save and Email” button clicked. | 1. “Save and Email” button clicked. 2. New invoice saved to records. Status bar colour set to “in progress” colour, status bar text set to “Saving to Records In Progress”. 3. Spreadsheet is generated. Status bar text set to “Creating Spreadsheet In Progress”. 4. Spreadsheet is sent as an email attachment to the recipient. Status bar text set to “Sending Email In Progress”. 5. Status bar colour set to “successful” colour. Status bar text set to “Sending Email Completed Successfully”. |
| Variation (alternative scenario) #3 – “Save and Export XLSX” button clicked. | 1. “Save and Export XLSX” button clicked. 2. A FolderBrowserDialog is shown. If this operation is cancelled, return the window to the same state as in step 4. If a folder is selected, note that directory. 3. New invoice saved to records. Status bar colour set to “in progress” colour, status bar text set to “Saving to Records In Progress”. 4. Spreadsheet is created and saved to the selected directory. Status bar text set to “Exporting Spreadsheet In Progress”. 5. Status bar colour set to “successful” colour. Status bar text set to “Exporting Spreadsheet Completed Successfully”. |
| Extension (error scenario) #1 – error saving to records | 1. Status bar colour set to “error”. Status bar text set to “Error Saving to Records”. An error dialog box is shown with a short message, and “Retry” and “Cancel” buttons. 2. If the “Retry” button is clicked, the operation to attempt to save to records is carried out again. If the “Cancel” button is clicked, clear the window and reset it, set the status bar colour to default and the status bar text to “Ready”. |
| Extension (error scenario) #2 – error saving (exporting) spreadsheet | 1. Status bar colour set to “error”. Status bar text set to “Error Exporting Spreadsheet”. An error dialog box is shown with a short message, and “Retry” and “Cancel” buttons. 2. If the “Retry” button is clicked, the operation to attempt to export the spreadsheet is carried out again. If the “Cancel” button is clicked, clear the window and reset it, set the status bar colour to default and the status bar text to “Ready”. |
| Extension (error scenario) #3 – error sending email | 1. Status bar colour set to “error”. Status bar text set to “Error Sending Email”. An error dialog box is shown with a short message, and “Retry” and “Cancel” buttons. 2. If the “Retry” button is clicked, the operation to attempt to send the email is carried out again. If the “Cancel” button is clicked, clear the window and reset it, set the status bar colour to default and the status bar text to “Ready”. |