REVIEW

Synoptic Assignment Task 5

Albion
Student ID: 11095480

Contents

Application review summary

The development of this prototype application was a success, all the client's needs were addressed, as described in the client brief. The application is functional, and the controls are clearly laid out making it easy and simple to use, due to being a prototype the recorded data is only stored locally within the application and so there is no persistence between launches of the app, the data needed for the app to work is loaded from txt files rather than a full database.

Visual studio 2017 was a great IDE Integrated Development Environment, it enabled me to more quickly and accurately write the code, due to the predictive features showing me the existing variables and methods etc. that I might be writing allowing me to press Tab and auto fill, this along with automated indentations are what makes IDE's so useful, making life so much easier for programmers.

Purpose

This applications purpose is to record expenses incurred by the company's sales representatives, recording the NET, Gross, and VAT Values of multiple purchases, providing the user with a summary showing the total values for multiple invoices/purchases.

Development Process

The life cycle used for this prototype was the waterfall model, this was effective because it provided a very quick and focused structure to the development, this enable me to finish this prototype more quickly giving the client something to work with and review sooner.

The requirement analysis phase went smoothly, the clients brief provided me with all I needed to know to create a sufficient prototype.

The plan/design phase was done through creating a few sketches and diagrams and then writing a plan and design document that discussed the development process and test plans for the application, establishing classes to be created and establishing the data flow of the application, this made the implementation phase much more effective than it would have been without such a detailed plan, because it gave me a clear view of the structure of the application and the flow of data providing me with guidance for writing the code.

Implementation and Testing (writing code and creating app) the implementation phase involved testing the features throughout the process, taking note of any significant issues, I used visual studio code 2017, programming with c#, creating a Windows Presentation Foundation WPF application using xaml to design the pages, object oriented programming paradigm was used throughout, creating classes to be used throughout the application, this was effect as it made using the sections of code within the classes in multiple pages or classes easier, it will also make future development easier as the classes can easily be re-used and child classes can be made to inherit features from existing classes.

Review This document is part of the review process, it will discuss not only how the prototype development went but will look at suggestions for the development of the next version, if the client approves of the prototype and wants to take this project further.

Usage

This prototype will be used by our client's sales representatives to record their expenses in order to claim them back, and so they will supply the application with the sufficient details, Date incurred, expense type, supplier name, invoice number, invoice value (price), and any comments to describe the purchase, and select either VAT inclusive or exclusive, once they have placed all the data in they can click the VAT button which will calculate the NET, Gross, and VAT content of the invoice by identifying the vat rate of the expense type e.g. food, accommodation, and fuel, once this has been done they can add this invoice to the list box giving them an overview of the invoice, multiple expenses can be recorded and placed into the list box at a time, once the user has recorded all their expenses they can click on the show summary button and they will be given the total values for their expenses.

The application is likely to be used across multiple mobile platforms due to the nature of it's usage the users will find it useful to be able to record their expenses shortly after they are taken, this will help them to stay on top of their expense claims, due to this the application was made using the Windows Foundation Presentation WPF which creates grids for the application that controls are anchored to meaning that as the page changes size and shape the forms adjust accordingly, rather than being anchored to the location 0, 0 top left of the page meaning that it would keep the controls at the same place and size, when the app gets stretch resulting in a very unprofessional looking app, using WPF will keep this application in proportion making it professional and easily usable on a range of screen sizes.

Features

The features of this application were all implemented to ensure that the app met the client needs and to ensure that the application can be used as intended, there are a few features that were not specifically asked for in the brief but that I believe enhance the features requested, the features will be separated into the pages they run

Login page

Page used to prevent unauthorized persons from using the application, this feature was included to provide some integrity and security to the data of the application, this prototypes login is merely a demonstration of the login page, for the release version a database will need to be implemented to store user login details, a discussion will need to be held to decide on either server based or cloud based storage for the application, if the application is only going to be used on devices in the office then it would be easy to manage it all locally through a server, but if users need to be able to use the application on the go with internet access a database that can be accessed online would allow that.

Clear button clears the controls of text, this feature was added to provide users with a quick way to empty the controls, it has very small impact on the application but it's the small things that make an app feel like a fully thought out product.

This page has a help button this hasn't been fully implemented for the prototype, for the release version the end user documents will be accessible using this button.

Purchase record page

This is the main page it handles the recording of expenses/purchases, as well as calculating the NET, Gross, and VAT content.

Preset expenses/purchases, this feature provides the user with a selection of preset choices that all have corresponding VAT rates that are provided for the application to calculate the VAT content. These are loaded from text files. This feature was included to provide the user with an easy way to provide the necessary data for the application, not including this would require the users to manually input the type of expense and the VAT rate of it, this would waste a lot of their time. Alternatively, I could have provided preset VAT rates and allowed the users to type in the name of the expense, and select the correct rate, this would provide opportunities for mistakes in the VAT rates and results of the users expense claim, potentially costing the client money, I believe the way that this has been done was the optimal way out of the current options.

Clear invoice clears the controls, this feature was added to provide users with a quick way to empty the controls, it has very small impact on the application but it's the small things that make an app feel like a fully thought out product, and anything that makes the users job easier makes the application better, allowing them to more quickly record multiple expenses. This feature is used when the user records an expense and when they click the clear invoice button.

The user's expenses are placed into a list box as a list item, this allows the user to view all of their recently recorded expenses before viewing the totals in the summary page, users will be able to notice any abnormalities or mistakes in the expenses they have recorded. This feature was included with future development in mind, in the current prototype version any mistakes noticed can' t be amended, but in future version features for deleting and editing expenses will be implemented. This feature will reduce the risk of incorrect expense claims being recorded and will allow the users to more easily record multiple expenses at a time as they will be able to quickly view the list and note how many expenses they have provided and how many more they must do.

Summary page

This page is purely for display purposes it calculates the totals for the expenses recorded displaying the number of expenses/purchases, VAT, Gross, and NET values, this provides the user with an idea of the total amount they have claimed and serves as confirmation that their expense claims have been recorded, not having this feature would require the users to calculate the totals themselves and double check their recorded expenses.

Style

The presentation of this prototype is vary basic, using default colors and fonts, as it doesn't affect the functionality of the application and so the client was provided with a functioning prototype allowing them to suggest any additional features, for the future development a meeting will be arranged to discuss a theme for the application, establishing what the businesses running theme is, finding fonts and colors to be used in the application, using the business logo within the application will require a logo of the correct size, going through all of that this early on in the development of the application could have been a waste of time, money, and resources, if the client was to decide that they didn't want the app in the end then it would be wasted or if they wanted some drastic changes made to it a lot of that work would have to be redone, and so I made the decision to hold fire on the presentation and style of the application during this first prototype version.

The style of the application can safely be implemented later with little fear of it effecting the functionality of the app.

Improvements

Recommendations for features and improvements to be implemented in the development of future versions and or the production version.

Implementing databases –

Switching from local storage to using databases to store expenses and user credentials etc.

The current prototype doesn't retain any data after the application is closed, this will need to be addressed for the production version, this is a key improvement that will provide opportunities for other great features that will improve the application.

This can be done using either a network to controls access to the databases or an online system that can control access, using databases with online features that allow the application to use them with any internet connection rather than just the office connection, will enable the users to record their expenses as soon as possible after making the payment as they will be able to record them anywhere they have data coverage. This will greatly improve the usefulness of the application and make the users more efficient in their work.

Having a database that can be accessed online by users will require extra security measures, the user login page of the application will ensure that only users with valid login credentials will be able to use the application, but extra measure will need to be take in order to make the application more secure, using encryption etc.

Homepage, with list of expense history. –

Once a database system is in place for the application I would suggest creating a homepage before the expense/purchase record page, this page will display the users profile, providing them with an over view of their expense allowing them to view their total expenses for the last week, month, 3 months, etc. they will be able to see if an expense claim has gone through or if it is still waiting to be viewed and claimed.

From this page the user will be able to click on a button to open the purchase/expense record page.

Creating this additional page and feature, will require another prototype, and so I have suggested using the prototype life cycle model (SDLC or development Method) this will involve iterations of development that produce prototypes, this will allow the development team to create new features such as this and review them with the client before creating the production version. Providing the both the developers and client ample opportunity for suggestions and ideas to be developed, resulting in a product that the client is pleased with.

This feature will improve the application in a few ways, it will provide the users with one place to view all their recorded claims, it will also enable them to know when their expenses have been claimed, via a notification system. This will help the users to more efficiently manage their expense claims, not having this feature would require the client's employees to contact other departments or check their accounts to get confirmation of an expense being claimed, they would also need to request access to the database and view it to check what expenses they have recorded, this could result in them not claiming an expense or accidently claiming more than once for an expense. The invoice number will prevent duplicate expenses form being processed, but it would still cause delays or issues in the process.

Feature for editing or deleting expenses. –

This feature will allow the users to edit or delete expenses that haven't been processed yet, this will be possible from the homepage by selecting an expense, (they will likely be shown in a list box similar to the way they are in the purchase record page) once an expense has been selected the expense/purchase record page will be opened with the selected expenses data in the controls, this will allow the user to see what they provided in the first record and make any necessary changes easily.

It will also be possible for the user to delete any expense claims that haven't been processed yet, this will be a much quicker process with a simple select and then click on a delete button, a confirmation message box will pop up, requiring the user to confirm with their password, this password required confirmation will prevent anyone unwanted deletion.

Implementing this feature will reduce the risk of invalid or incorrect expense claims from being processed if the user is able to easily edit them then as soon as they notice anything that is wrong with their recorded expenses they will correct them or delete the claim. Not having this feature would provide opportunities for mistakes to go unchecked resulting in wasted time and budget for the client and their employees.

Implementing this feature will require having the two previous features and will require a development team that is experienced in using databases and C# together in application development.