School of Computing and Digital Technologies

Software Projects (55-407815-AF-20245)

Currency Transfer Application

Name: Jason Izuchukwu
Student ID: 34052600
Degree Route: Cs wAI

Table of Contents

1. Requirements Gathering Stage

1.1 Project Description and Users

Briefly outline your project in the context of a business. Who are the users in this context? Identify two users. If you identify more than two users, this will be fine, e.g.

Name	Description
Personal User	Access the system's features to transfer currency between accounts
Business User	Access the system's extended features to transfer currency between accounts.
Customer Service User	Assist all users with their queries
System Admin User	Monitor all user accounts and perform CRUD operations if needed

1.2 User Stories and Acceptance Tests

Provide three user stories for each user you identified for the project. Provide a set of acceptance tests for each user story you identified, e.g.

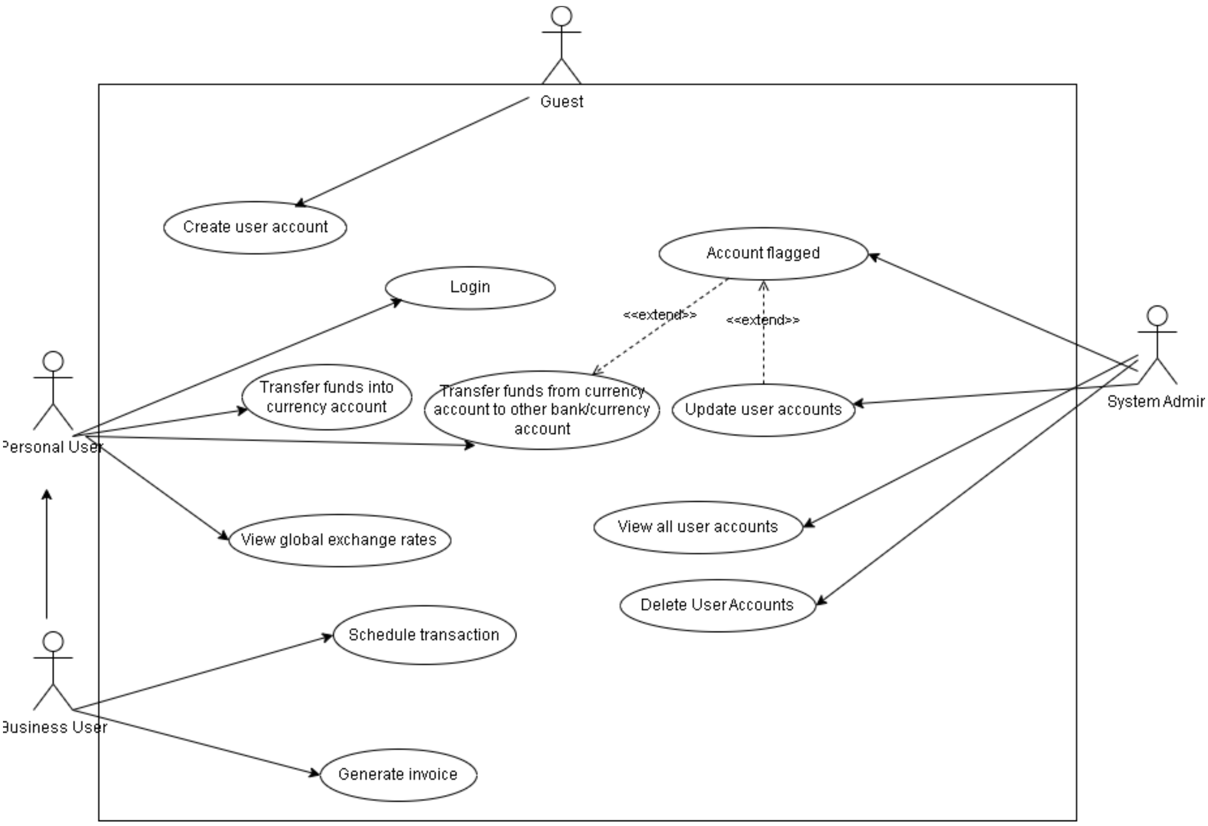
ID	User Story	Notes	Acceptance Test
PU-US-1	As a personal user, I want to create an account in my chosen currency so I can use the system.	N/A	• Verify that the system allows the user user to successfully create an account in their chosen currency when all required information has been provided. • Verify that the system denies new user's account creation when required information has not been provided.
PU-US-2	As a personal user, I want to be transfer funds from my UK bank account into my currency account so I can send money to my family aborad.	There should be a cap on the transfer amount	• Verify that the system permits the transfer if the entered details match a valid UK bank account. • Verify that the system permits the transfer if the transfer amount is less than or equal to the balance of the UK bank account. • Verify that the system denies the transfer if the details do not match a current UK bank account. • Verify that the system denies the transfer if the transfer amount is greater than the balance of the UK bank account.

			• Verify that, once a transfer is permitted, the system is able to convert the amount from GBP to the chosen account's currency. • Verify that the system does no conversion if the chosen currency account is already in GBP. • Verify that the system denies the transfer if it exceeds the transfer cap. • Verify, if the system flags the transfer, the account is held in a state of suspension • Verify that the system will action the transfer the following day if the transfer is made out of office hours.
PU-US-3	As a personal user, I want to be able to transfer funds between my currency account to another currency account or another bank account in the world.	Daily limit for transfer depending on the currency being transferred. Transfer fees incurred. Should be able to refund transfers if transfer flagged*	• Verify that the system displays global exchange rates. • Verify that the system permits the transfer if the transfer amount is less than or equal to the balance of the user's currency account • Verify that the system permits the transfer if the entered details match an existing bank account or currency account. • Verify that the system denies the transfer if the transfer amount is greater than the balance of the currency account. • Verify that the system denies the transfer if the transferee details do not match an existing bank or currency account. • Verify that, once a transfer is permitted, the system can convert the currency of the transferer to that of the transferee using global exchange rates.
BU-US-1	As a business user, I want to schedule a transaction between my currency account and another bank account.		• Verify that the system schedules the transfer if the date of the transfer is on or after the current date

			• Verify the system does not schedule the transfer if the date is before the current day.
BU-US-2	As a business user, I wish to be able to generate invoices for every transaction.		• Verify that the system generates an invoice after a transaction has been successful • Verify that the system does not generate an invoice if the transaction has not been successful
BU-US-3	As a business user, I wish to be able to transfer a larger sum of money than I would be able to in a personal account.		• Verify that the system increases the transfer cap if the user has a business account • Verify that the system keeps the transfer cap the same if the user does not have a business account.
SA-US-1	As a system admin I want to be able to view all user accounts		• Verify that that the system permits the admin to see all user accounts • Verify that the system doesn't display anything if there are no user accounts
SA-US-2	As a system admin I want to be able to delete user accounts		• Verify that the system allows the admin to delete an account when a button is pressed • Verify that if the right button isn't pressed the system does not delete any user accounts
SA-US-3	As a system admin I want to be able unblock accounts		• Verify that blocked accounts can be unblocked after a button is pressed • Verify that the system does not unblock accounts that aren't already blocked

1.3 Use Case Diagram

Insert your diagram here in **png** format and provide a brief annotation, e.g.,

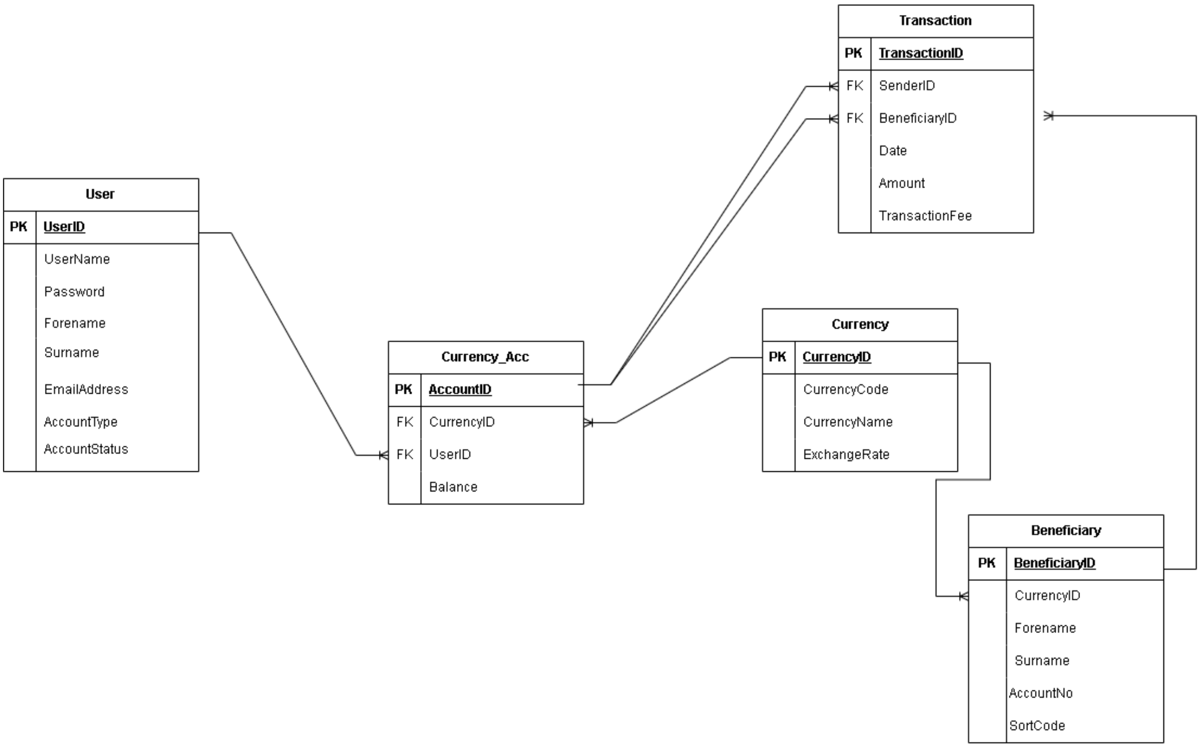


So here we have 3 users. The personal user can access most of the basic features of the application such as viewing global exchange rates and transferring funds between different currency accounts. The business user inherits all of the base functions of the personal user with the added benefit of being able to schedule transactions for a later date(as businesses do especially regarding payroll) as well as generating invoices. The system admins functions are different but simple; being able to view all existing users, being able to delete their accounts if needed and also being able to unblock or unsuspend their accounts if they've ben blocked for going over the limit.

2. Design Stage

2.1 Entity Relationship Diagram

The design must be annotated outlining the purpose of each entity within the design, e.g.,



AccountNo
SortCode

The ERD Diagram that models my application is rather simple. A user can create an acccount and from there create a currency account which stores the user ID as a foreign key. A currency account is linked to a currency table where I store all the currencies I want to use for the sake of the application. The Beneficiary ID exists for the feature of being able to transfer money from a user's account to an external bank account, not a different currency account that it owns.

3. Development Stage

3.1 Transitioning a Prototype to Production-Quality Software

Provide evidence of how you transitioned your prototype to a production-quality (where possible) software system systematically. This may include use of a systematic approach to convert an entity-relationship diagrams to a corresponding database implementation. You can also include such details as how you interfaced to a database system within an application (if applicable to your project).

https://rp.mockplus.com/rps/ovbYGGM3dF? Please view "Currency Transfer"

I'll first give a quick breakdown of how my use cases became ERD components:

Key Use Cases → ERD Components:

- "Create user account" → `User` table with account details
- "Transfer funds" → `Transaction` table and relationships between accounts
- "View global exchange rates" → `Currency` table with exchange rates
- "View all user accounts" → `Currency_Acc` table linked to `User`
- Beneficiary management → Created the beneficiary table later on when I realised I wanted to store the details of the “external bank account” that the user has the option to transfer to

The use cases I devised show the need for tracking transactions between different currency accounts, which I then reflected in the ERD relationships with the foreign keys.

ERD Components → Database Tables:

- `User` → `Users` table with all specified fields
- `Currency_Acc` → `Currency_Acc` table with AccountID, Balance, UserID, CurrencyID
- `Transaction` → `Transaction` table with all transactions and appropriate relationships
- `Currency` → `Currency` table with exchange rates
- `Beneficiary` → `Beneficiary` table with account details of said beneficiary

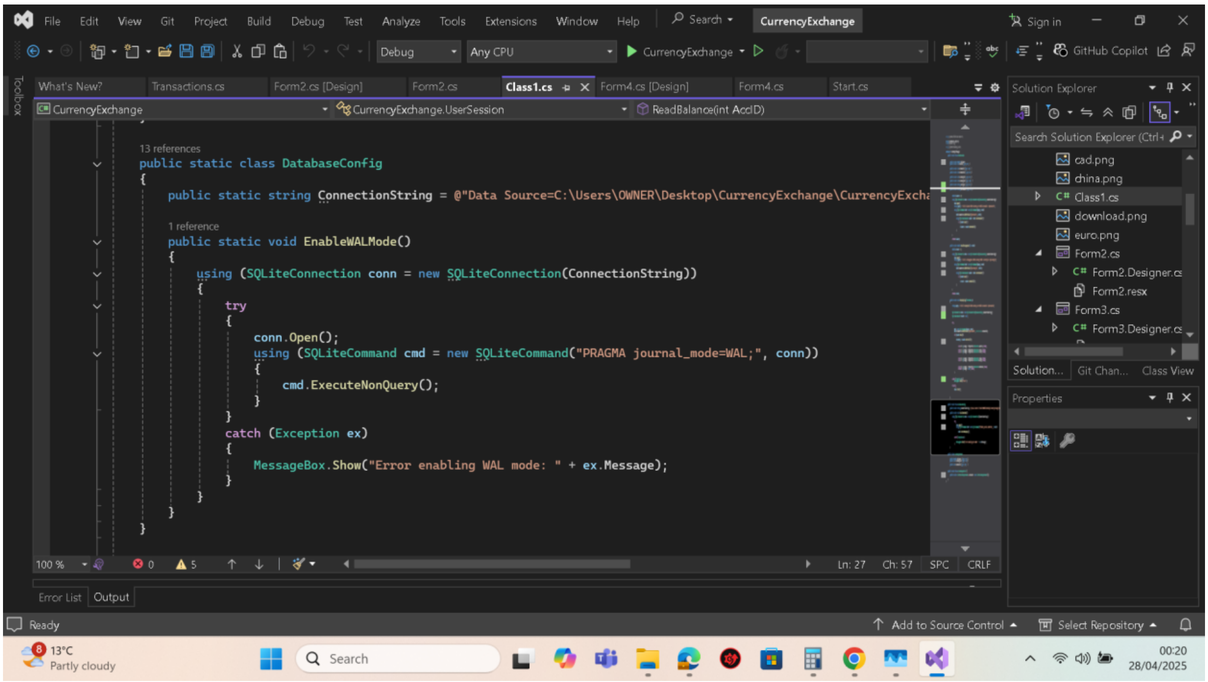
Relationship Implementation:

These are all the relationtships that exist within my different tables or entities.

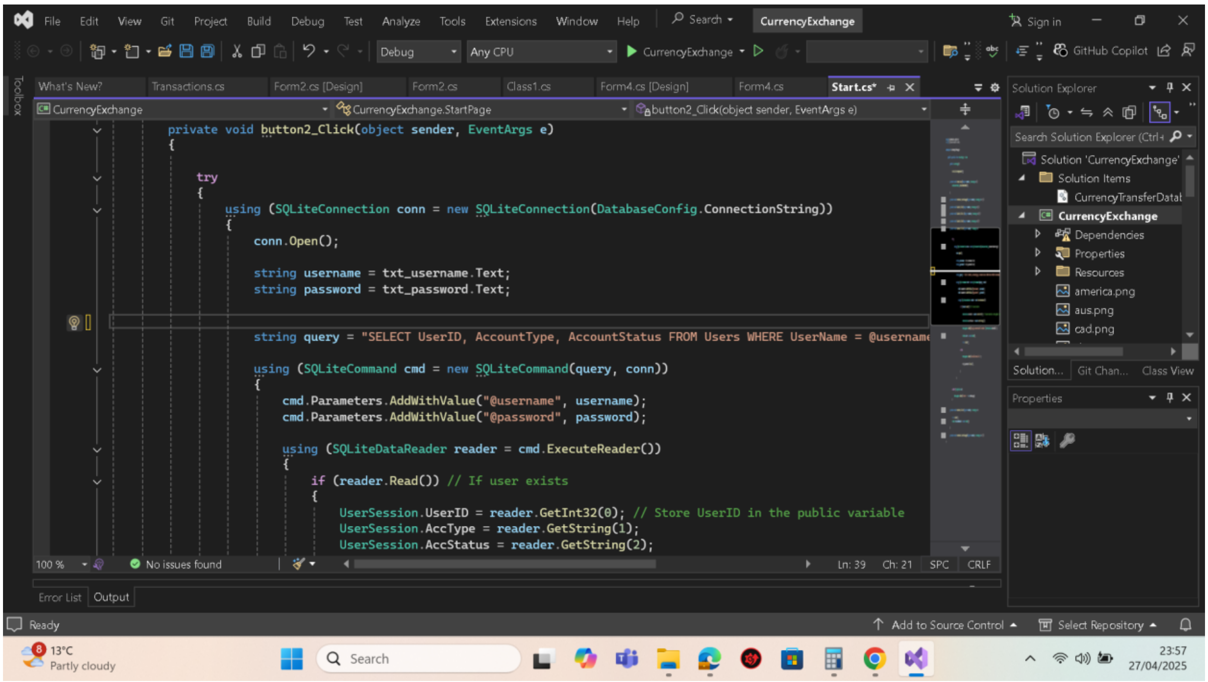
- Foreign keys in `Currency_Acc` linking to `Users` and `Currency`
- Foreign keys in `Transaction` linking to sender/receiver accounts and beneficiaries

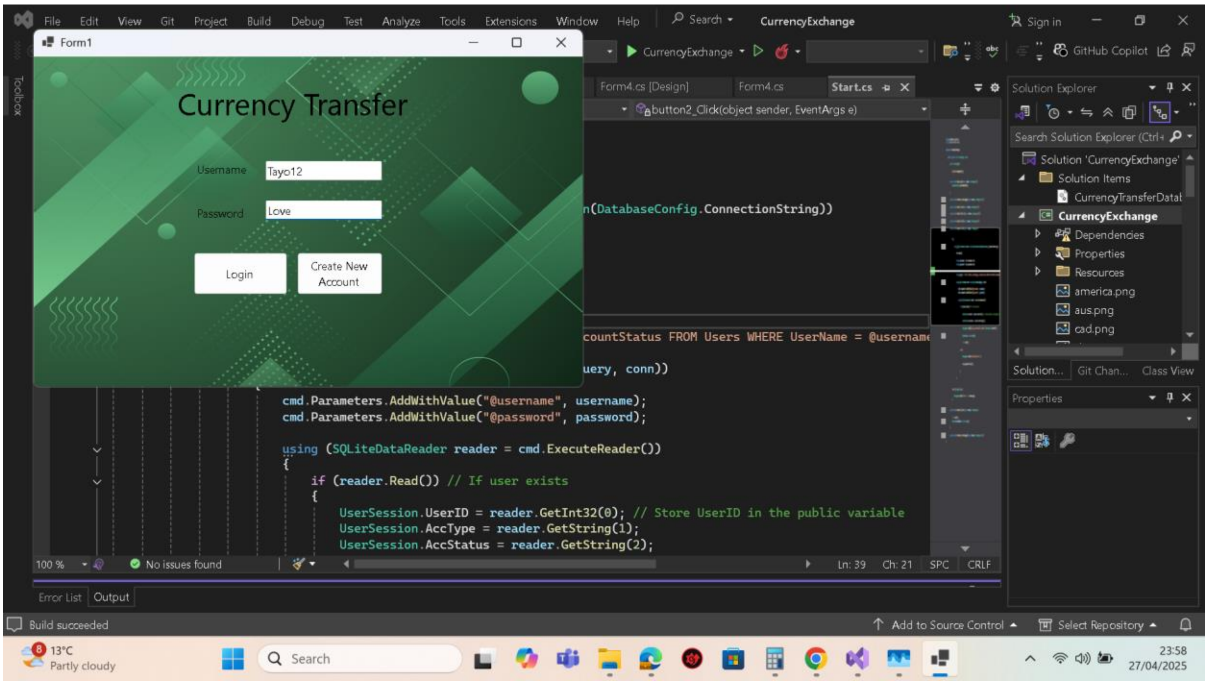
Once I made my protype I decided to start small. I had already decided on my approach to use windows forms c# to create an application rather than website as I had never done so before and wanted the challenge and experience. With the resources provided by the uni. I was able to use visual studio to drag and drop controls into to recreate the first page which is the login form. I used a website called freepiks to find an license free image that I used as my background as I realised my prototype was looking rather bland.

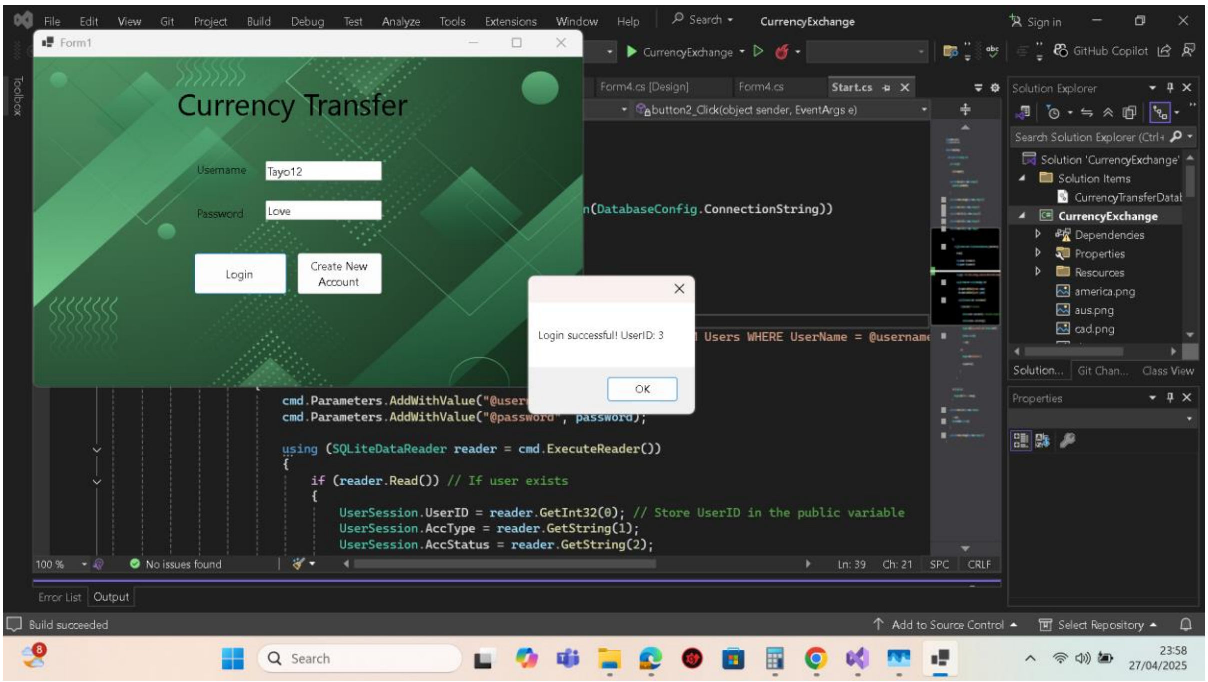
I had already devised a database file and I took the relevant steps to connect it to the project by adding it as an item in visual studio. It could then be used in datagrid views that I end up making use of alot. I also needed to note the file location of the database, which at first I just made a string object called connectionString to link to the database but as my project got bigger and bigger I made a new class called Database config in which the instance variable was a string that I could change or call whenever I needed to setup a SQLite command and saved me a lot of time later on. I also enabled the pragma mode to WAL or write ahead logging as I ran into several issues late along the line while having functions that were writing and reading several times as well as making use of transactions.



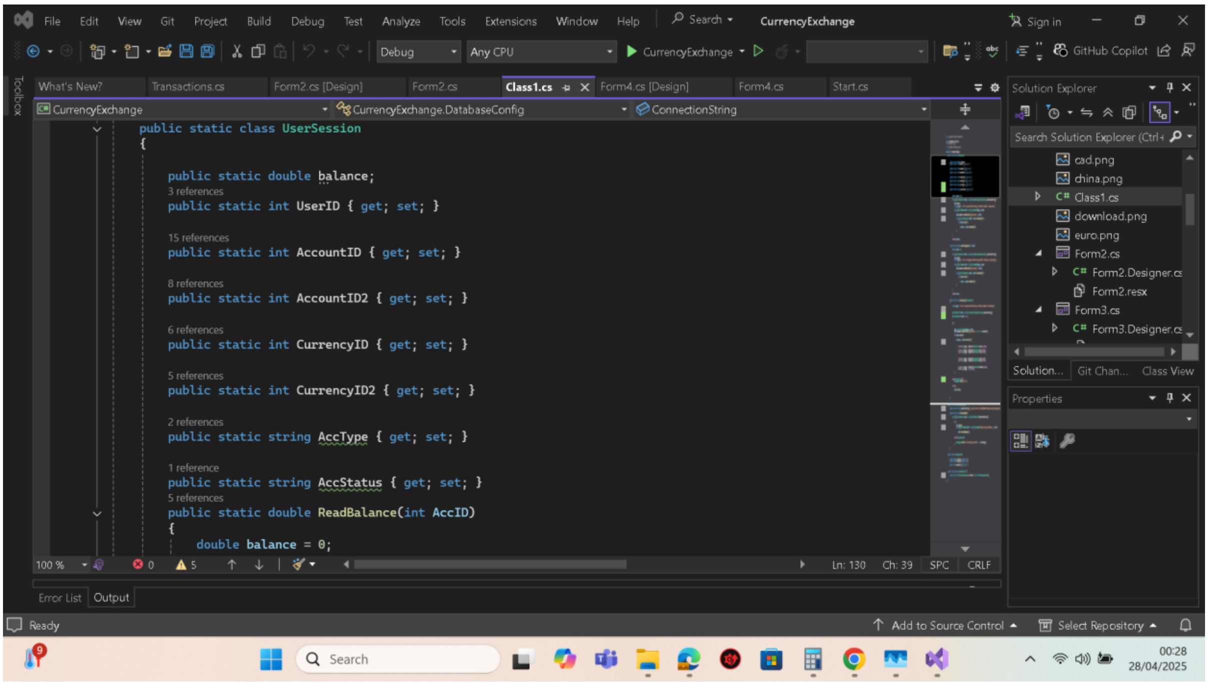
Afterwards I made the login button take the username and password textboxes as input and check if it matched up with what was in the databases using a SQLite reader object.



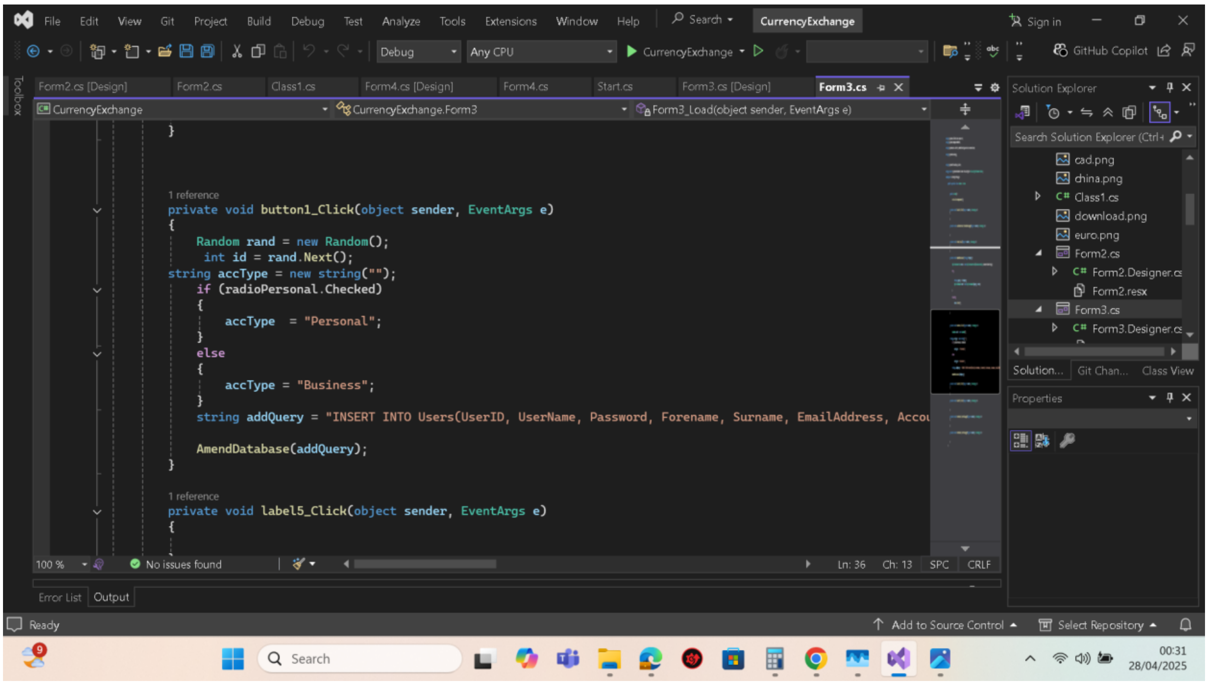


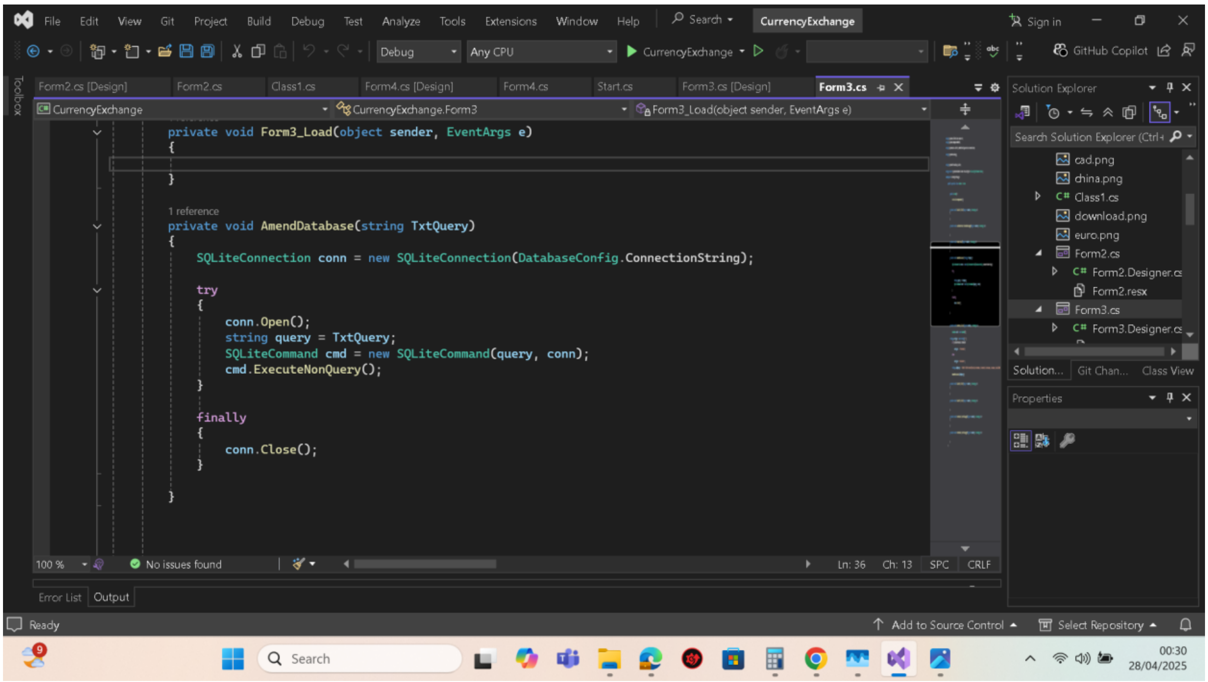


The details are stored in a separate public class called UserSession which records the necessary information of the user after login which is then used in almost every single form afterwards that needs to know a specific user's details.



After this I performed a similar operation to the login button with the create new account button which prompted the user to a new form or screen and took the details in textboxes to create a new user with a randomly generated userID which is then stored in the database.





After this I then devised a system to open a second form which in then turn displayed a list of currency accounts and allowed the use to pick which currency account they wanted to display. From there they are greeted by the general UI which allows them to view currency exchange rates as well as their current account balances and a list of buttons that can take them to different forms such as switching currency accounts, transferring between currency accounts, transferring to an external bank account and even topping up the existing currency account.

3.2 Test Description

In this sub-section, you are required to outline your test description with evidence (i.e., test results).

User Story ID	Test ID	Acceptance Test	Test Description	Expected Result	Test Result
RU-US-1	T001-RU-US-1	Verify that the system allows returning user to successfully login when valid credentials are provided.	1. Double click Chrome browser icon on desktop. 2. Type *localhost/project_enigma/login.php* on the URL address bar. 3. A login form will be displayed. 4. Enter the following details: a. Username: *user123* b. Password: *Us3r*!23 5. Click '*login*' button 6. A page will be displayed with message that says "*Login is successful!*"	User will be successfully logged in.	Pass / Fail
	T002-RU-US-1	Verify that system won't allow returning user to successfully login when invalid credentials are provided.	1. Repeat steps (1) through (3) in T001-RU-US-1 2. Enter the following details: a. Username: *user123* b. Password: *Us3r*!2332 3. Click '*login*' button 4. A page will be displayed with message that says "*Login is unsuccessful!*"	User will not be successfully logged in.	
PU-US-1	T001-PU-US-1	Verify that the system allows returning user to successfully login when valid credentials are provided.	1. Open the application 2. Enter username and password; Tayo12 and Love 3. Click login button 4. A message will be displayed saying login successful with the user ID of the account	User will be successfully logged in	Pass

PU-US-1	T002-PU-US-1			1. Repeat steps 1 to 3 above. 2. A message box saying invalid details should appear	User will not be succesfully logged in	Pass
PU-US-2	T001-PU-US-2			5. Login and select a currency account 6. Click add currency 7. Fill out the details provided 8. You will be taken back to the home screen with a new balance depending on the amount input	The updated balance should be the old balance + the amount input in the previous page	Pass
PU-US-2	T002-PU-US-2					
PU-US-3	T001-PU-US-3			1. Login 2. Click ‘transfer between currency accounts’ 3. Pick the currency account u wish to transfer to 4. Input the amount and lick ‘confirm transaction’ 5. A message box saying ‘ Currency Exchange successful’ should pop up	The sending account should have the amount + fee removed from it and the receiving account should have the a converted or exchanged amount added to it	Pass
PU-US-3	T002-PU-US-3			9. Repeat steps 1-4 above 10. A messagebox saying ‘Error: Insufficient funds’ should pop up	If the amount in the sending account is less than the amount to be sent the error message should pop up	Pass
BU-US-1	T001-BU-US-1			11. Login 12. Click the view transactions button 13. Select a transaction then hit ‘Generate invoice’ button 14. File explorer should pop up	The user should see an invoice pdf on their system	Pass

				15. Save the name of the new pdf		
BU-US-1	T002-BU-US-1			16. Login 17. Click Generate Invoice	A messagebox should pop up saying an error if their is no transaction selected in the datagridview	Pass
BU-US-2	T001-BU-US-2			18. Login 19. Select either 'transfer between currency accounts' or 'transfer to external bank account' 20. Make a transfer under 5000 GBP or the equivalent	The transaction should go through and a message saying transction successful	Pass
BU-US-2	T002-BU-US-2			21. Repeat steps 1-2 Above 22. Make a transfer exceeding 5000 GBP or equivalent	An error message saying exceeded transfer cap should pop up and the account should be blocked	Pass
BU-US-3	T001-BU-US-3			23.		
BU-US-3	T002-BU-US-3			24.		
SA-US-1	T001-SA-US-1			1. Login with the right system admin account 2. Click the 'users' button	You should be directed to page where it shows all existing user accounts	Pass
SA-US-1	T002-SA-US-1			1. Repeat steps 1-2 above	No user accounts should show up in the datagridview	Pass

				if there are none	
			25.		
			26.		

4. Software and Its Presentation

4.2 The Production-Quality Software

You are expected to submit the project, including all its components (e.g., codebase), compressed in a zip file (or 7z). The file should be named "*IndividualProject_StudentNo*" and must be uploaded to Blackboard as directed in the relevant submission point. (e.g., *IndividualProject_32212112.zip*)

4.3 Video Presentation

The project must be showcased in a video recording of up to 15 minutes. We will stop watching after the 15th minute.

https://shu.cloud.panopto.eu/Panopto/Pages/Viewer.aspx?id=b35aedd7-0a14-401c-b664-b26b00ba566b

5. Evaluative Report on Legal, Social, Ethical and Professional Issues

5.1 Relevant Issues

Identify two or three issues that specifically relate to your project (this could be GDPR, copyright, accessibility, testing, etc.), and briefly explain their relevance to your project.

To start with I realise I am working with a virtual or conceptual system that cannot actually utilise real-word bank accounts. That being said, I attempted to put myself in the mindset of a real-life developer developing a system that worked in said way. One thing I realised, after development, was how I stored my passwords in plaintext which could make me face fines or legal actions for me mishandling data like this in the real word.

To remedy this I could've used password hashing which I'm sure .NET has a feature or package that accommodate that.

Other than that I noticed a couple other issues with my codebase:

- No **keyboard navigation** (users can't tab between fields).
- No **screen reader support** (blind users can't interact with buttons/dropdowns).
- No **error feedback** for certain invalid inputs (I tried my best to include as many error inputs and try statements as I could but I know I missed a couple).

Why It Matters:

- **Social Exclusion**: Over 1 billion people live with disabilities. Ignoring accessibility excludes them, which is unethical.
- **Reputation Risk**: If this were a real product, bad accessibility could lead to public criticism (e.g., lawsuits like the 2019 Domino's Pizza case).
- **Professional Stanadards:** I could've used more unit testing and error-handling within my code to make it more readable and valid to be used in a corporate setting

Apart from all of that, certain matters came to mind during development. The introduction of a transfer cap was important for real world settings because its extremely important for currency exchange/transfer applications to not facilitate any form of money laundering and so a transfer can help deter potential money launderers.

Another issue I could've potentially addressed is the lack of a wider variety of currencies. I've got roughly 11 currencies hardcoded into my codebase but in a real-world application used by people around the world then alot more currencies would be needed.

5.2 Discussion

Discuss what impact these will have on the project. Specifically, you may discuss how these issues will impact on the way you will transition your prototype / design you developed in Requirements and Design stage to a production-quality (where possible). As well as supporting your discussion with references, throughout your work you are also expected to identify recent public examples that have been reported in the news (or other reputable sources), for example if you are creating an application that will store personal data, a useful example would be to mention the fine British Airways received for being in breach of GDPR, all of which should be cited using the APA format.

6. Incorporation of formative feedback

Provide evidence of how you evaluated and acted on the formative feedback you received from your tutors, during the IT sessions.

Individual Project		
	Feedback Received	**Action Taken**
Week 1	I was suggested to add more functional requirements such as daily limit for transfer depending on the currency being transferred. If the account is flagged it should be put into a state of suspension that the admin can remove and also the user can refund the transaction if it is flagged.	I added all of these things as user stories.
Week 2	In my use case diagram I didn't have guest actor. I also imposed an order of events linking ovals to other ovals which I was told wasn't the right approach.	I created a guest actor and separated the linked ovals and made sure all ovals linked directly to an actor.
Week 3	I had a currency attribute in my user entity when there was no need as users could have multiple currency accounts. Also I struggled with the idea of a global exchange rate entity but as I planned it to be static was told it was better off being stored in code rather than a database.	I instead made sure there was a currency account entity and that currency type was one of the attributes.
Week 4		
Week 5		
Week 6		

7. References

Your reference list should contain citations to external sources that have been relied on throughout your project's development and writing this portfolio. The citations should conform to the APA referencing system[1], e.g.,

[1] https://libguides.shu.ac.uk/ld.php?content_id=33803668

Fitzgerald, J., & Hayward, P. (2009). Inflamed: Synthetic folk music and paganism in the island world of The Wicker Man. In P. Hayward (Ed.), Terror tracks: Music, sound and horror cinema (pp. 101-111). London: Equinox.

Melchers, G., Shaw, G., & Shaw, P. (2013). World Englishes (2nd ed.). Retrieved from http://lib.myilibrary.com

Miller, D. (2016). Social media in an English village. https:// doi.org/10.14324/111.9781910634431

TED. (2007, January 6). Sir Ken Robinson: Do schools kill creativity? [Video file]. Retrieved from https:// www.youtube.com/watch?v=iG9CE55wbtY