



Temporary Store of Value

Out of the Ordinary

.....
Hello! 🖐️
.....

Lishen Ramsudh

Temporary Store of Value - Wallets

- Trying to streamline business operations and open up product possibilities through the use of a temporary store of value
- The temp store of value and automation of certain operations opens up a different trust relationship with a process being the custodian instead of hoping that a transaction is upheld
- C# DotNet Core 3.1, Postgres, Redis, Xamarin Forms, (eventually include Docker, AWS ECS Fargate and API Gateway, AWS Aurora)

Out of the Ordinary

**“ With first principles,
you break things
down to their
fundamental truths,
and then reason from
there - Elon Musk**

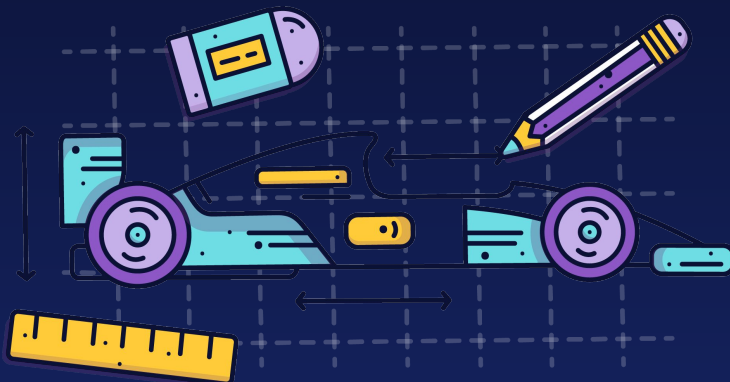


Out of the Ordinary

Use Cases

- Escrow type services - system becomes trusted store of value as well as allowing free wallets to enable escrow that can pay out to any bank account once the transaction is completed
- Inter-department budgeting - have a wallet per department
- Tender processing - create wallet for Tender and add strict workflow, process control on top of this such that the money only leaves the wallet AFTER all steps have been completed

Out of the Ordinary



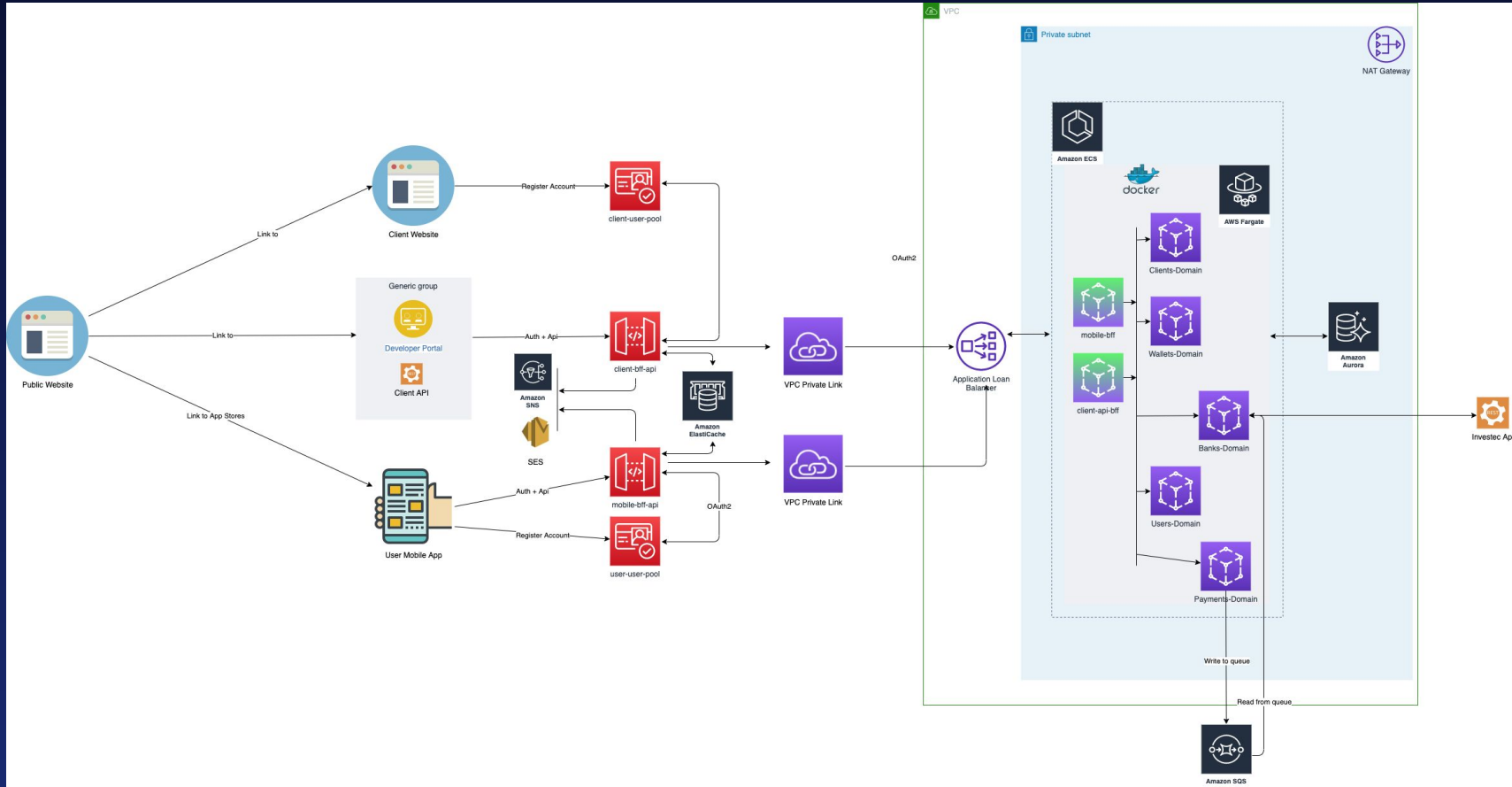
How it works

Out of the Ordinary

How it works

- Focussed services to handle different parts of the system
- Individually scalable
- Easy to update
- Bank Service is scheduled to ping the Investec Api every 30min
- Caches the main pieces of the previous run
- Then uses the payment reference description to find a wallet associated with the deposit
- Allocates funds accordingly

Out of the Ordinary



Out of the Ordinary

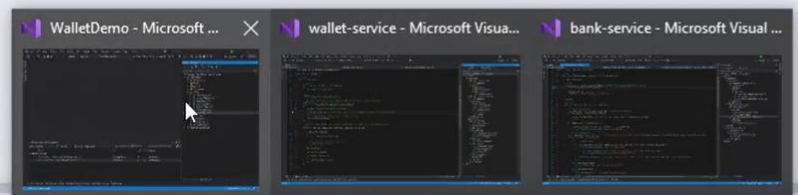
```
MINGW64/c/Projects/open-source/investec-api-wallet-service/wallet-service...
$ dotnet run
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: https://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\Projects\open-source\investec-api-wallet-service\wallet-service.api
```

- FTS Configurations
- FTS Dictionaries
- Aa FTS Parsers
- FTS Templates
- Foreign Tables
- Functions
- Materialized Views
- 1.3 Sequences
- Tables (2)
 - transactions
 - Columns
 - Constraints
 - Indexes
 - RLS Policies
 - Rules
 - Triggers
 - wallets
 - Columns

```
postgres@postgres-local * wallet-service/...
postgres@postgres-local
History Scratch Pad
"SystemRefId", "WalletId", "TransactionType", "Amount", "Currency",
lic.transactions;
```

Data Output Explain Messages Notifications

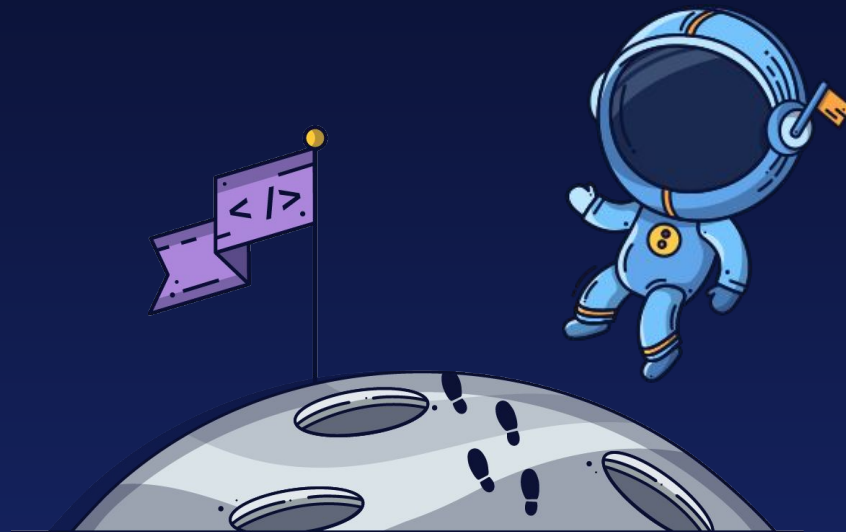
Id	SystemRefId	WalletId	TransactionType	Amount	Currency	CreationDate	TransactionTrigger
[PK] bigint	uuid	bigint	smallint	numeric (15,10)	character (3)	date	smallint



Challenges encountered

- Had to learn a lot of tech that I am not exposed to on a daily basis
 - Dotnet Core
 - Postgres
 - Docker
 - Xamarin
- Kept one part in common and that was the language: C#
- Working on focussed smaller components helped to break down the pieces.
- Cloudformation iterations were tough

Out of the Ordinary



Next Steps

Out of the Ordinary

 **Investec**
Private Banking

OFFER ZEN

Next steps

- Create dotnet new template project to help create additional domain services
- This is part of a much larger project that relies on temporary store of value
- Configure build and deploy pipelines
- Clean up code
- Investigate Event Sourcing using Redis Message Bus for inter-service communication
- Place Wallet and Bank Services on DockerHub

Out of the Ordinary



Get Involved!

Out of the Ordinary

 **Investec**
Private Banking

OFFER ZEN

Get in touch

- Wallet, Bank, SDK and Domain Service Template Generator are all open-source
- <https://github.com/ihiresolutions>
- Email: lishen.ramsudh@gmail.com

Out of the Ordinary



iHire Solutions
ihiresolutions

Edit profile

Overview

Repositories 4

Projects

Packages

Popular repositories

Customize your pins

[investec-api-csharp-sdk](#)

C# wrapper for the Investec Programmable Banking Api

C#

[investec-api-wallet-service](#)

Works in tandem with the Investec Bank Service to manage multiple wallets within an Investec Bank Account

C#

[investec-api-bank-service](#)

Works in tandem with the Investec Api Wallet Service to help manage multiple wallets within a single Investec Bank Account

C#

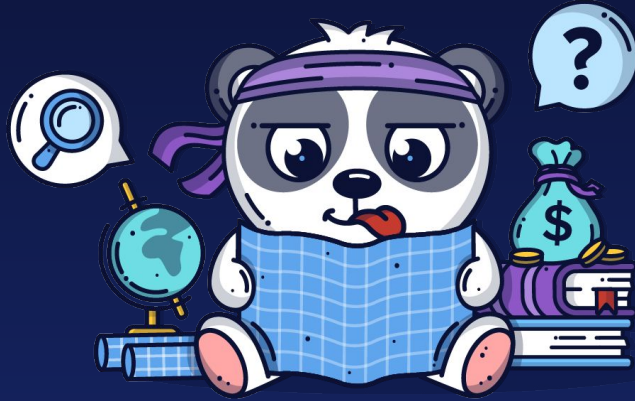
[domain-service-template](#)

Dotnet Core 3.1 template that creates a typical domain service with integration into Postgres

Out of the Ordinary

 **Investec**
Private Banking

OFFER ZEN



Questions?

Out of the Ordinary