**AlphaBet Backend Exercise**

You have a black box called "processor" that is used for money transfers between banks.

It's API supports two calls:

1 - **perform\_transaction(src\_bank\_account, dst\_bank\_account, amount, direction)**

direction = credit / debit

This call returns a transaction\_id

2 - **download\_report**

This downloads a daily report of transaction results.

It may contain info about transactions from the last 5 days.

The report format is:

transaction\_id, success/fail

Use the processor to define a billing system that supports a single API call:

**perform\_advance(dst\_bank\_account, amount)**

The system credits the customer with the amount.

In the following 12 weeks, the system performs debits of amount/12 once a week.

A failed debit is moved to the end of the repayment plan (a week from the last payment).

Please implement, define, and elaborate as much as you can about the architecture of such use case (Flows/Classes/Databases/Edge Cases, etc.)

Please upload solution to Github.

Good luck.