INVESTMENT APP

Gabriel Fonsaca

INTRODUCTION

The search for “investment” is constantly growing in Brazil, a research demonstrates the number of individual investors in the market is increasing each year. There is a lack in the market for an app that shows valuable insights and leads investors to make a better allocation of their money. Actually, the majority of the apps available in the market are focused on teaching, market indicators and informing with news, in addition to that, they have fewer functionalities regarding the investor wealth. In consequence of that, the project was born and it's driven by the vision of delivering investor insights. This document contemplates the description of a few functionalities, both in business and technical spectrum.

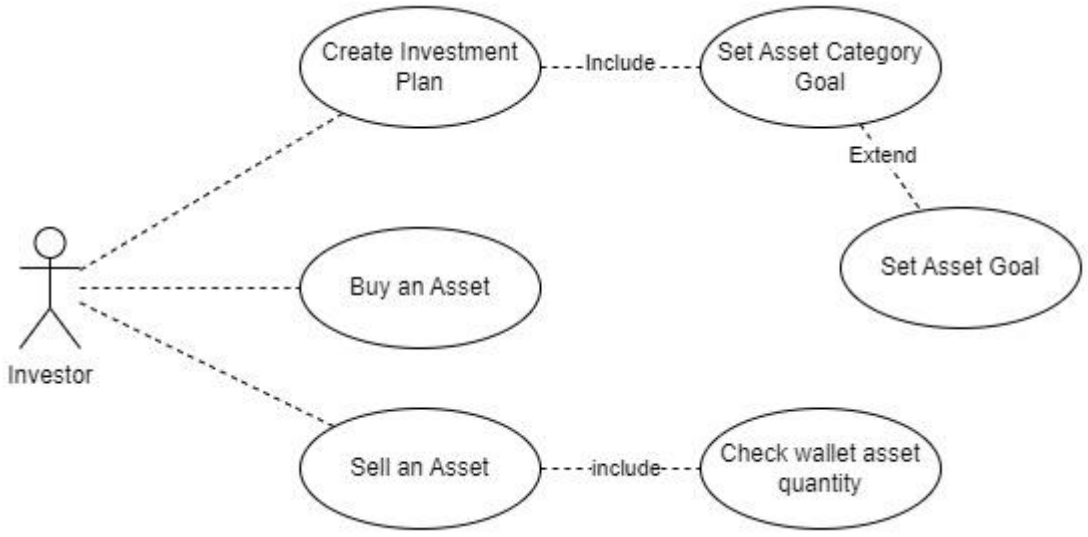
SCOPE

This project is a backend Rest API with the functionalities:

1. Manage investment goals;
2. Buy and sell assets.

The document doesn’t contemplate the entire project functionalities.

USE CASE



The investor can create an investment plan, buy and sell assets. When creating an investment plan, the investor must set the category goal. The categories and most assets are automatically fed by a system task which is not contemplated on this document. A goal for an asset is optional. When selling the asset must be in the investor's wallet. Available amount is not checked when buying, because the app does not control the investor earning, only his investments.

FUNCTIONAL REQUIREMENTS

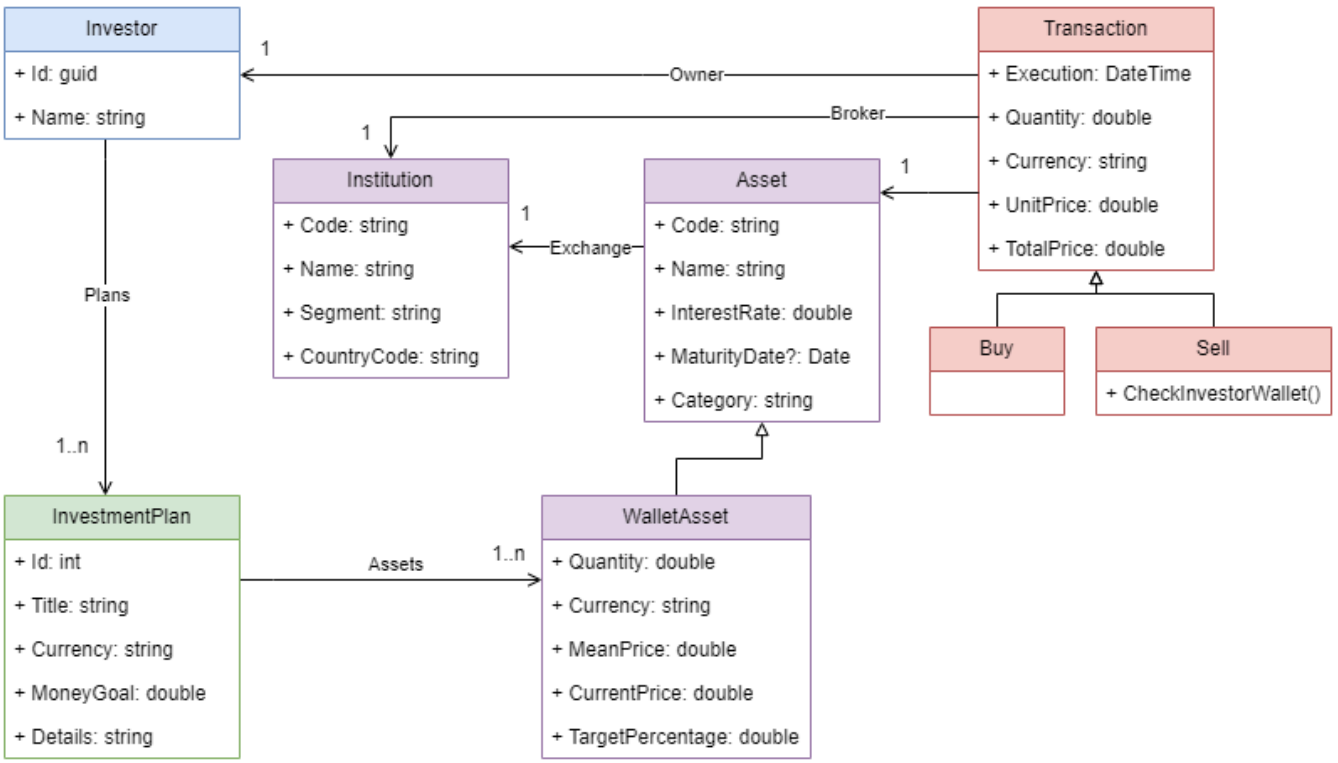
1. The system must require the creation of an investment plan.
2. The investor is required to set at least one goal.
3. The asset goal must be a percentage value.

4. The investor can buy any listed asset. He must inform the unit value and quantity.
5. The investor can only sell assets that are in his wallet. He must inform the unit value and quantity.
6. The investor must specify from which plan the asset was withdrawn.

NON-FUNCTIONAL REQUIREMENTS

1. Assets and institutions requests should have fast response, so they may be stored in memory.
2. The system may support thousands of users concurrently, so it must scale horizontally.
3. Get requests cannot take longer than 3 seconds.
4. Post, Put and Patch requests cannot take longer than 5 seconds.
5. The system must guarantee the consistency of wallet's assets when a transaction is executed.

CLASS DIAGRAM



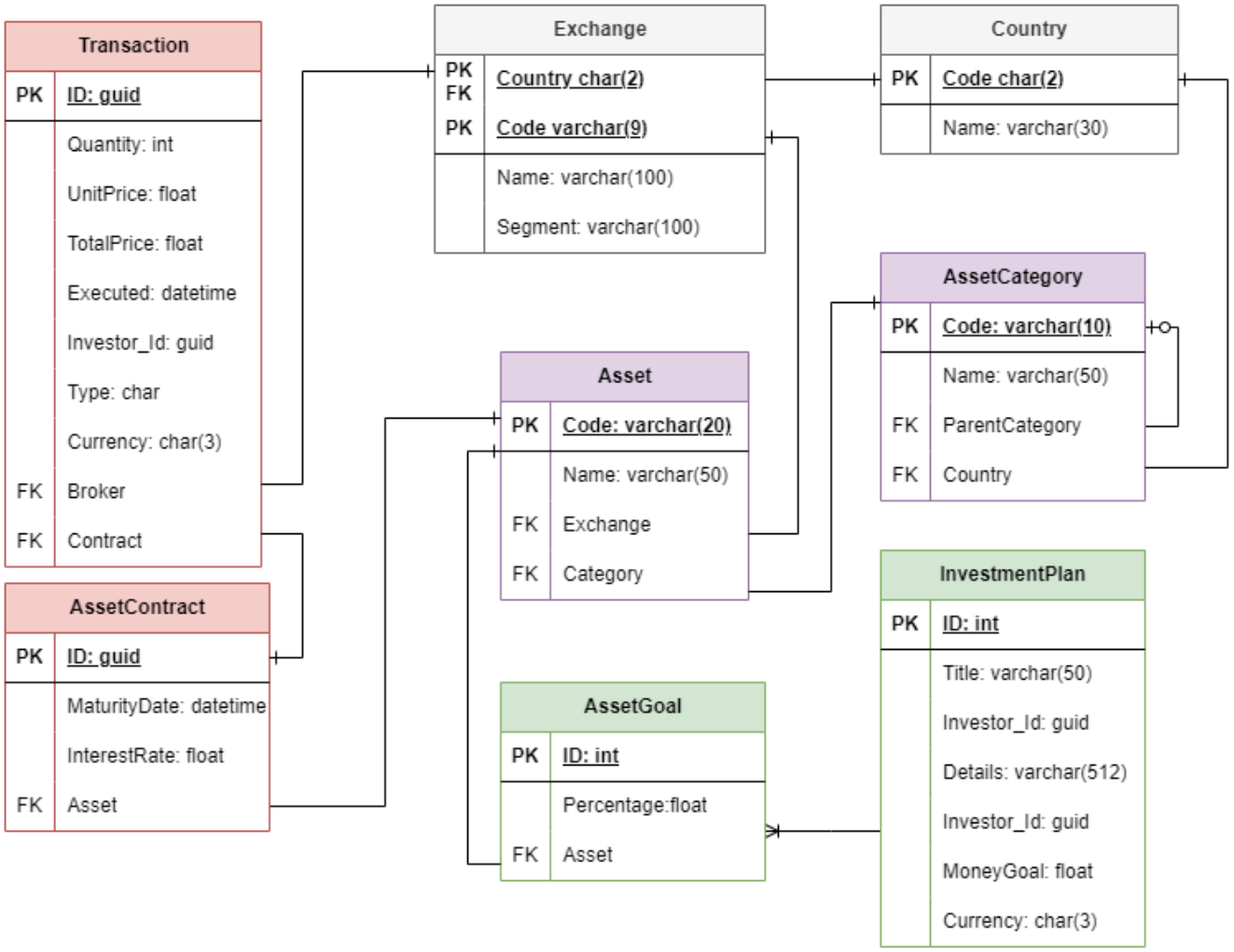
The investor has at least one investment plan with at least one asset in his wallet. One asset can have the quantity equal to zero but the target percentage must be set, which represents the amount of that asset in his wallet. Mean price of a wallet asset is the weighted average of the buy price. An asset is offered in an exchange, which is an institution.

The institution code is the join of the country code with its unique identifier, example: In Brazil the B3 exchange has the following id 09.346.601, so the institution code is BR09346601.

The asset code is the join of the institution code with the asset code. For example, in Brazil stock exchange B3 has listed Bradesco Stock: BBSA3, so the asset code is BR09346601BBSA3.

Every input information of an asset purchase or sell is a transaction.

SQL DATABASE DIAGRAM



Remarks:

- The wallet asset is compiled during each transaction and stored in a NoSQL database which is not in scope.
- Investor data is managed by a separated service, which is not in scope.

TECHNOLOGIES

- .NET