בס"ד
Methods in software engineering
תשפ"א, סמסטר א'

Software Design Document

SDD

Project name: CPM

Developers: David Tsibulsky & Omri Haham

Content

בס"ד

CPM

Crypto Portfolios Manager

1. Introduction

a. System Overview

CPM provides crypto currency brokers and traders a set of tools – which helps them to easily manage several cryptocurrency portfolios – under one main account.
The program will show the broker the information, profits, losses and more statistics about each of his/her clients (investors) - while maintaining a reliable user interface that will make trading and managing portfolios - a much easier task, by monitoring and controlling each and every investor balance.

b. Purpose

Giving crypto currency experts a useful tool that will help them manage several crypto investors portfolios – with one single account.

c. Scope

- CPM will allow to manage no more than 8 portfolio
- The basic architecture of the software is a Python-based user interface which allows him to trade for several user at Binance.com – The world's most known cryptocurrency trading platform.
- The software will require an internet connection - using Binance API to get his clients portfolios information.
- The main libraries will be:
 python-Binance, json, os, time,datetime tkinter, matplotlib, schedule.

d. Constraints

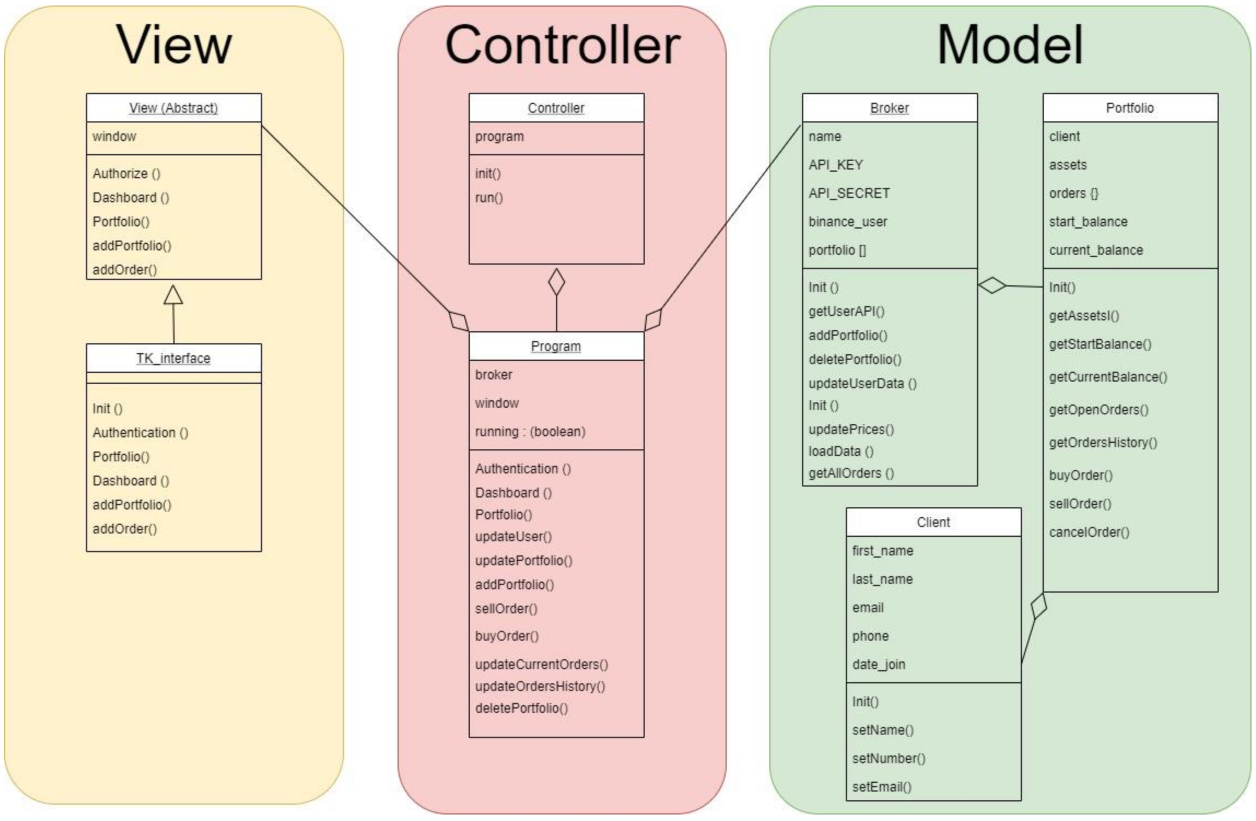
Our Broker customers must have a crypto currency which traded at Binance, and a basic knowledge in crypto currency transactions

2. Design

a. **Data Design** - Database Description

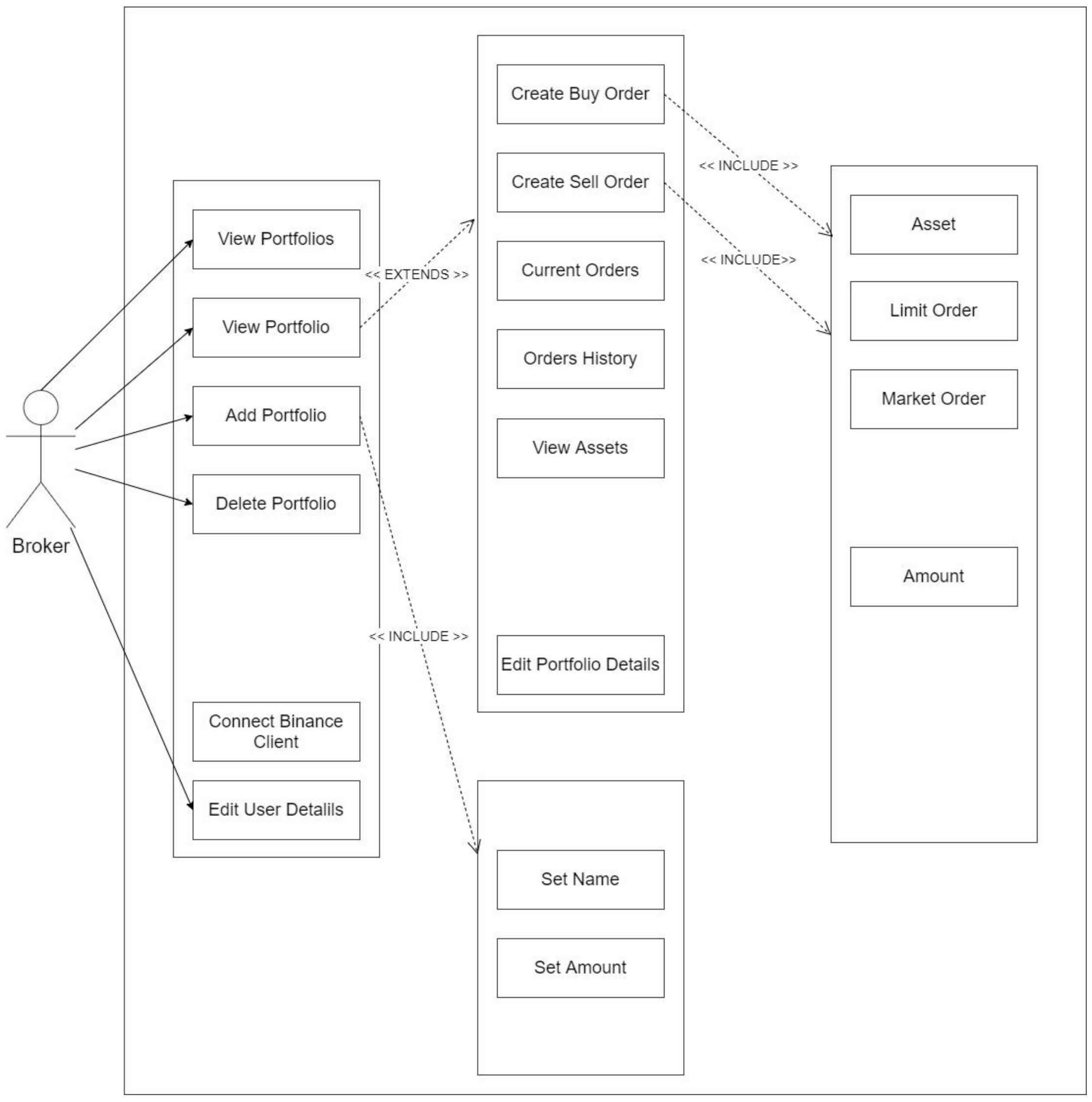
JSON files – which holds the main user data and the customers information, trading data and etc.

b. **Structural Design** - Class Diagram

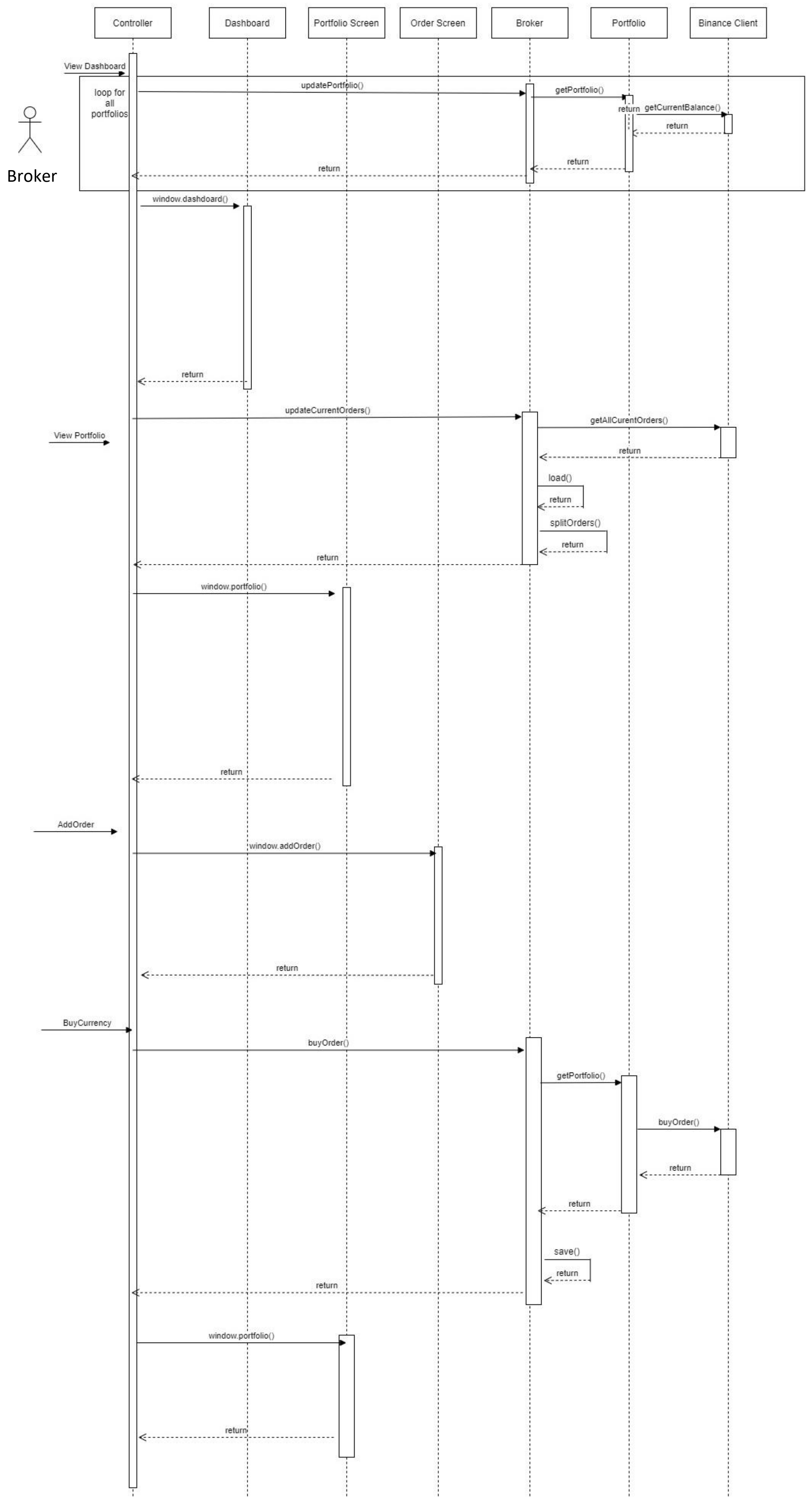


c. **Interactions Design**

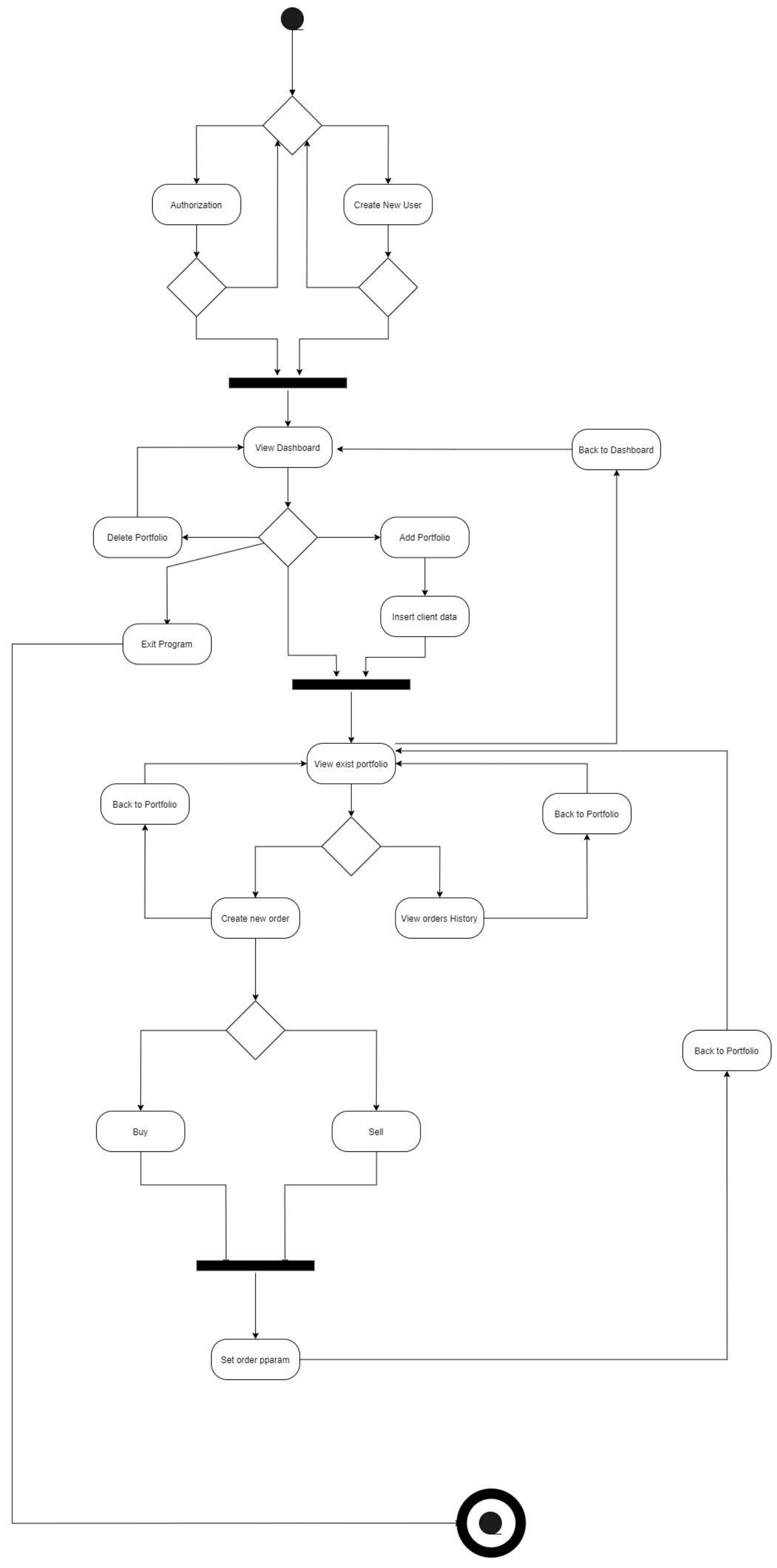
i. Use Cases



ii. Sequence Diagram

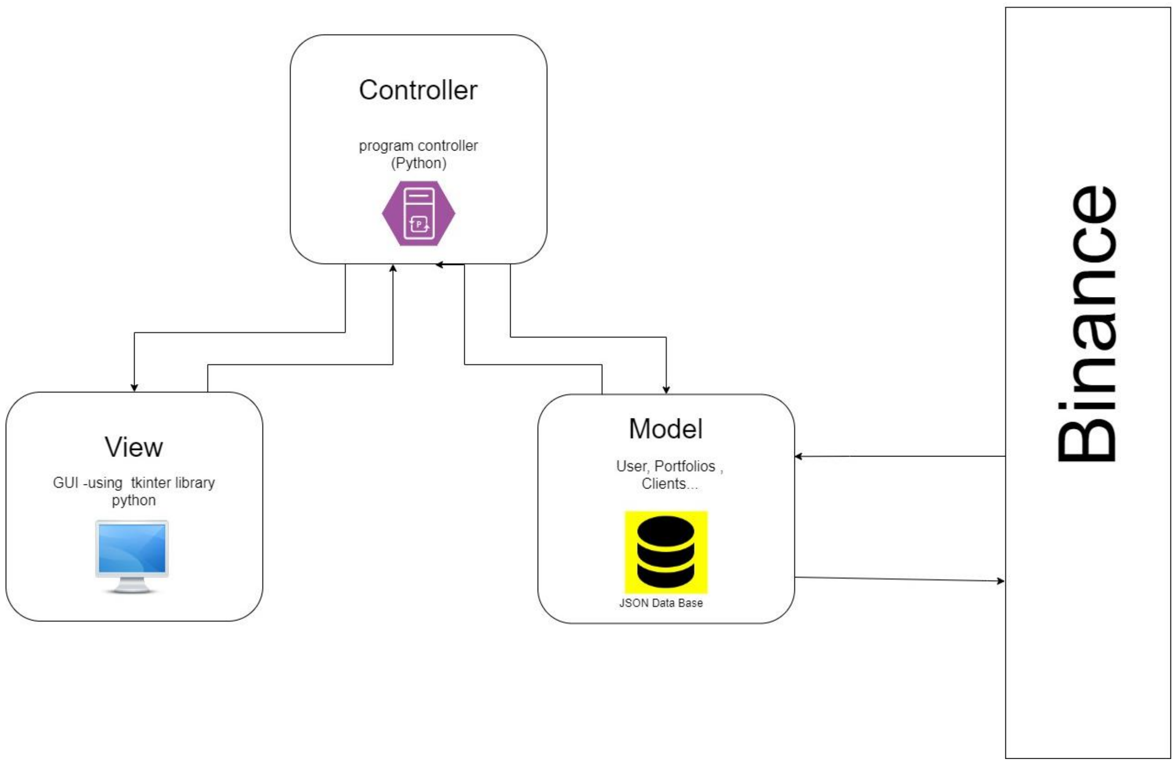


iii. Activity Diagram / State / Processes



d. **Software Architecture Pattern**

i. N-tier: Data, Logic, Service, Presentation tiers etc.


Controller
program controller (Python)
View
GUI -using tkinter library python
Model
User, Portfolios , Clients...
JSON Data Base
Binance

Verification

e. **<u>Validation and Evaluation Plan</u>**

1. ***Top Priority Validation*** – <u>Assets Division</u> - Validating that the total assets and their value at the main user account, will be divided precisly for each portfolio.
2. *<u>Orders Validation</u>* – each order will be attached to every portfolio.
3. *<u>GUI & QA</u>* - Each button leads the main user to the wanted window, and each error (such as "insseficient balance for an order") will show up when needed.

f. <u>Testing Platform</u>

Visual Studio Code and Pycharm Debuggers for Python

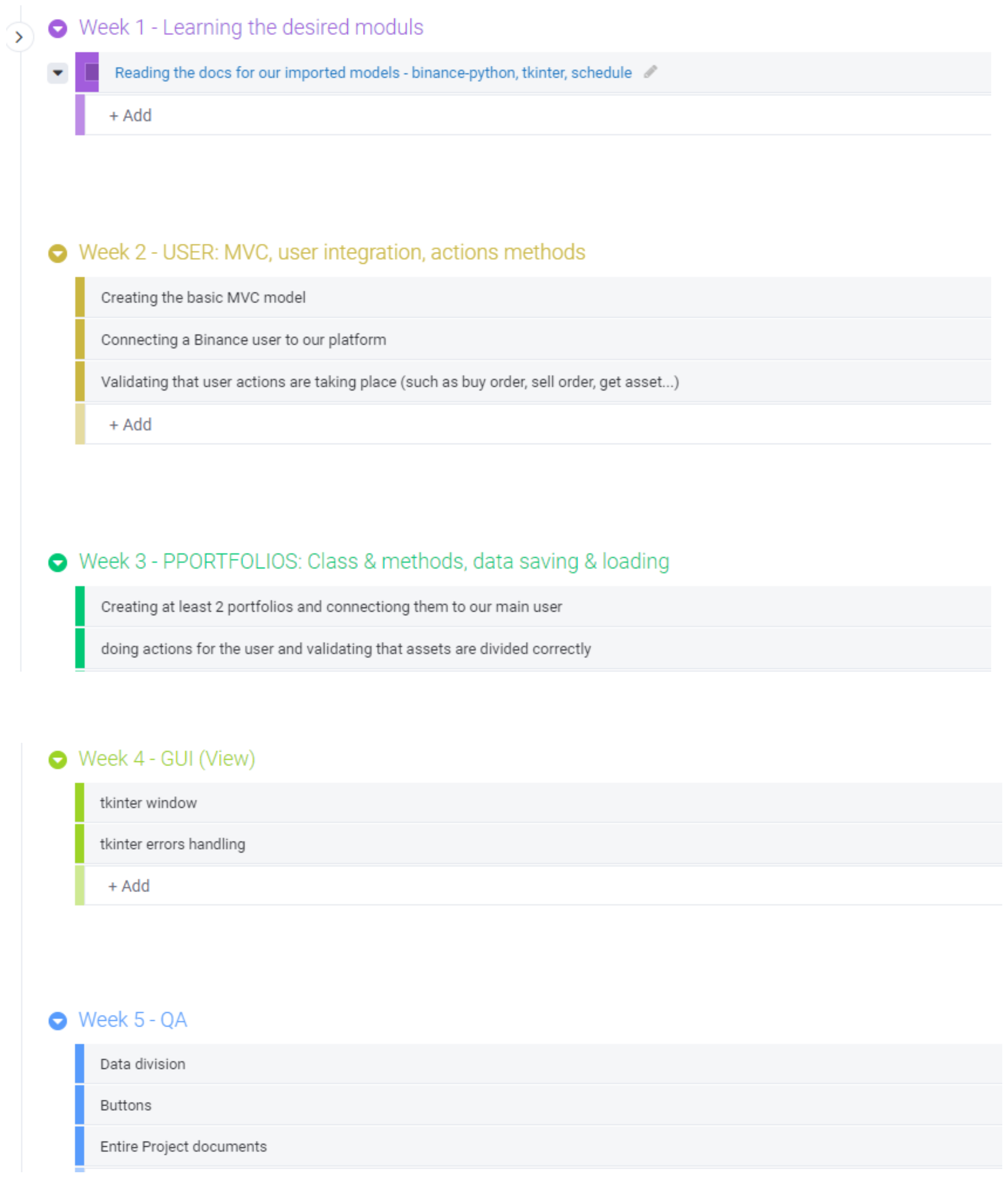
(David is using Pychamr, Omri is using Vscode)

- Note: Orders Testing will be made with specific methods which intended for test orders in binance-python library.

בס"ד

3. Project Management

a. Schedule



Week 1 - Learning the desired moduls

Reading the docs for our imported models - binance-python, tkinter, schedule

+ Add

Week 2 - USER: MVC, user integration, actions methods

Creating the basic MVC model

Connecting a Binance user to our platform

Validating that user actions are taking place (such as buy order, sell order, get asset...)

+ Add

Week 3 - PPORTFOLIOS: Class & methods, data saving & loading

Creating at least 2 portfolios and connectiong them to our main user

doing actions for the user and validating that assets are divided correctly

Week 4 - GUI (View)

tkinter window

tkinter errors handling

+ Add

Week 5 - QA

Data division

Buttons

Entire Project documents

b. Team Roles – final

Omri – GUI, libraries, files handling, scheduled tasks code
David – MVC, Testing, time management