Finance Valley

Group 5

Dylan Dayter , Nick Bequette , Aidan Amarante , Dominick Dupoux, Jian Zhong

Dylan Dater: Sections 3 & 4

Nick Bequette: Sections 5 & 6

Aidan Amarante: Sections 7 & 8

Dominick Dupoux: Sections 1 & 2

Jian Zhong: Sections 9 & 10

Volere

Group 5

Dylan Dayter , Nick Bequette , Aidan Amarante , Dominick Dupoux, Jian Zhong

Dylan Dater: sections 3 & 4

Nick Bequette: sections 5 & 6

Aidan Amarante: sections

Dominick Dupoux: sections

Jian Zhong: sections

Edition 20—2020

by James Robertson & Suzanne Robertson
principals of the Atlantic Systems Guild

The Volere Requirements Specification Template is intended for use as a basis for discovering and communicating your requirements. The template provides sections for each of the requirements types appropriate to today's software systems. You may download the template from the Volere site and adapt it to your requirements process and requirements tool. The template is process independent and can be used by Agile, Traditional, and Outsourced projects. The template can be used with any combination of automated tools you are using see http://www.volere.co.uk/tools.htm for a summary of available tools.

The Volere Requirements Knowledge Model (included with the download of this template) shows the formal structure of the template and the cross references between the components in the above table of contents.

Volere

Volere is the result of many years of practice, consulting, and research in requirements engineering and business analysis. We have packaged our experience in the form of a generic requirements process, requirements training, requirements consultancy, requirements audits, a variety of downloadable guides and articles, a requirements knowledge model and this requirements template. We also provide requirements specification-writing services.

The first edition of the Volere Requirements Specification Template was released in 1995. Since then, organizations from all over the world have saved time and money by using the template as the basis for discovering, organizing, and communicating their requirements.

The Volere web site www.volere.co.uk contains articles about the Volere techniques, experiences of Volere users and case studies, requirements tools, and other information useful to requirements practitioners.

The Volere requirements process is described in the book *Mastering the Requirements Process—Third Edition* by Suzanne Robertson and James Robertson, Addison-Wesley, 2012. ISBN 0-321-81574-2

Kindle and Safari editions are also available.

Other books about requirements by Suzanne Robertson and James Robertson:

Business Analysis Agility Addison-Wesley, 2019. ISBN 0-13-484706-7

Requirements Led Project Management Addison-Wesley, 2005. ISBN 0-321-65904-X

Updates to this template and instructions for downloading are available at http://www.volere.org

Public seminars on Volere are run on a regular basis in Europe, the United States, Australia, and New Zealand. For a schedule of courses, refer to http://www.volere.org

In-house courses and project clinics are run on request.

Video course on the Volere requirements process**: Requirements the Masterclass** is available at http://www.informit.com/store/requirements the-masterclass-livelessons-traditional-9780134189758

Requirements Types

For ease of use, we have found it convenient to think of requirements as belonging to a type. There are two reasons for the type: as an aid to discovering the requirements and to be able to group the requirements that are relevant to a specific expert specialty. Sometimes you might find it necessary to assign more than one type to a requirement.

Functional Requirements are the fundamental or essential subject matter of the product. They describe what the product has to do, the rules that it has to carry out or what processing actions it must take.

Non-functional Requirements are the properties that the functions must have, such as performance and usability. Do not be deterred by the unfortunate name for this kind of requirements, they are as important as the functional requirements for the product's success.

Constraints impose restrictions on the chosen solution. These restrictions might apply to the whole project, for example: budget, time, skills. Other

constraints relate to the technology to be used like: the product might have to be implemented in the hand-held device being given to major customers, or it might have to use the existing servers and desktop computers, or any other hardware, software, or business practice that must be conformed with and cannot be changed.

Project Drivers are the business-related forces. For example, the purpose of the project is a project driver, as are all of the stakeholders—each for different reasons.

Project Issues define the conditions under which the project will be done. Our reason for including them as part of the requirements is to present a coherent picture of all factors that contribute to the success or failure of the project and to illustrate how managers can use requirements knowledge as input to help to manage a project.

Testing Requirements

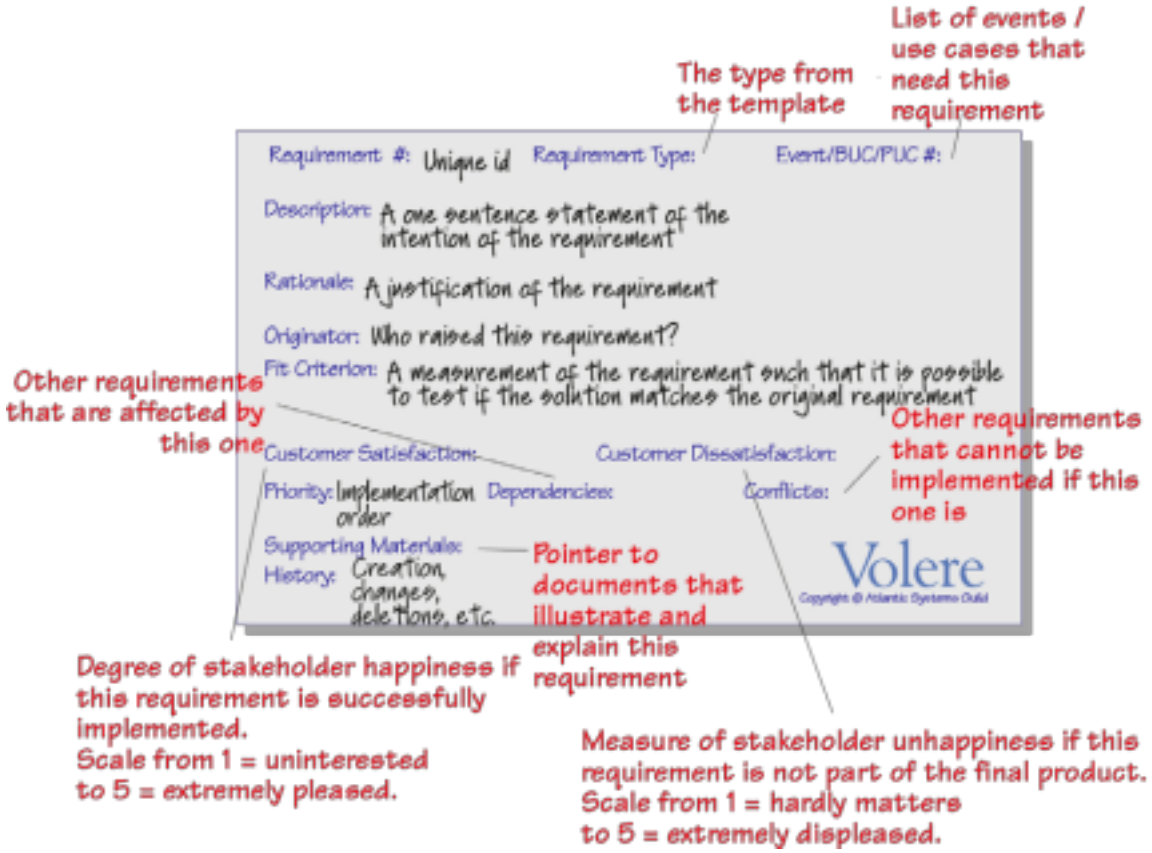
The Volere philosophy is to start testing requirements as soon as you start writing them. You make a requirement testable by adding its *fit criterion*. This fit criterion measures the requirement, making it possible to determine whether a given solution fits the requirement. If a fit criterion cannot be found for a requirement, then the requirement is either ambiguous or poorly understood. All requirements can be measured, and all should carry a fit criterion.

Atomic Requirements Shell

The requirements shell is a guide to writing each atomic requirement. The components of the shell (also called a "snow card") are identified below. An atomic requirement is made up of this collection of attributes.

You might decide to add some additional attributes to provide traceability necessary for your environment. For example: products that implement this requirement, version of the software that implements this requirement, departments who are interested in this requirement, etc. There are others but do not capriciously add attributes unless they really help you: every attribute you add needs to be maintained.

This requirements shell can, and should, be automated. When you download the template you will also find an Excel spreadsheet implementation of the snow card. You can also implement it using whatever requirements tool/s you have available.



The following discusses and provides examples for each of the sections of the Volere Requirements Specification Template. For each section, the Content, Motivation, Considerations, Examples *and* Form *provide the template user with some guidance for writing each type of requirement.*

1. The Purpose of the Project

The first section of the template deals with the fundamental reason your client asked you to build a new product. That is, it describes the business problem the client faces and explains how the product is intended to solve the problem.

1a. The User Business or Background of the Project Effort

Content

In this project we are going to be doing a game based around the aspect of finance, it will be a platformer that has the elements of teaching finance to the player we came upon this idea because we could all have something to add to its development in the financial aspect.

Motivation

The motivation is to teach/develop the user aspects of finance by using the mechanics developed within our game

Considerations

The business is not entirely serious as we want to put out a game that teaches players how to learn and navigate finance

Form

A platformer game that teaches the aspects of financial literacy basics

1b. Goals of the Project

Content

We want to provide the user with a complete and wholistic product that effectively builds on the education of finance

Motivation

We want to try and make the most fun and engaging game possible to drive sales

Examples

We want to give an immediate and complete response to customers who order our goods online.

To reduce road accidents by accurately forecasting and scheduling the de-icing of roads.

Measurement

- Service goal: Provide customer with an accurate understanding of finance

- Revenue goal: we plan to make a small profit charging a small price for the game

- Legal goal: To not infringe on copyright and not have to pay royalties for borrowed content. We will be making everything ourselves to avoid these

Form

We are creating this to develop/refresh skills in finance to the user

One advantage would be that our game could be used in schools,work environments and even recreational use so it is a multi fascinated game to develop in all areas

After play testing and feedback is reported we can accurately access how impactful the aspects of finance are implemented in the game

2. The Stakeholders

2a. The Client

Content

Professor Cornell

Motivation

Client motivation is to build an enjoyable product that can educate the user on finance

Considerations

Professor Cornell (marketing Department)

Form

Ceo- Aidan Ama

Coo-Dominick Dupoux

Cfo-Dylan Dayter

Cmo-Jian Zhong

Marketing manager- Professor Cornell

Client responsible for:

-Product Due Date

-Product Marketing

-Product guidelines

2b. The Customer

Content

Joe Smith

the people for whom he is responsible will adopt a new/changed product.

In the case of development of a mass-market product, this section contains a description of the persona developed as the archetypical customer for the product (See section 2e).

Motivation

Customer motivation is to find a product to build and expand their knowledge of finance from a beginner level of experience

Form

Customer responsibilities:

-Purchase Product

-Engage and Develop knowledge provided by product

2c. Other Stakeholders

Content

Examples of stakeholders:

- Sponsor: Professor Cornell
- Customer: Joe Smith
- Experts in Field: CEO & COO
- Marketing Team: Profesor Cornell
- Legal Experts: Jerry's Law Firm
- Designers and developers
- Software engineers:Dominick Dupoux,Aidan Ama
- Technology experts: Jian Zhong & Dylan Dayter

- System designer: Nick Bequette
- Stakeholder identification: Professor Cornel Marketing team leader in Albany Incorperated
- Knowledge needed: How to perform a professional rollout of a game
- The degree of involvement necessary for that stakeholder/knowledge combination: Not much hands on supervision mainly resources like connections to more experienced employees and funding
- The degree of influence for that stakeholder/knowledge combination:not much influence taken from stakeholder
- Agreement on how to address conflicts between stakeholders who have an interest in the same knowledge: majority shared decides the verdict

Motivation

CEO: John Smith | |-- **Sponsor: Professor Cornell** | - Knowledge Needed: Oversight of professional game rollout | - Degree of Involvement: Minimal hands-on supervision, mainly providing resources | - Degree of Influence: Limited influence COO: Jane Johnson | |-- Experts in Field: CEO & COO | - Knowledge Needed: Industry insights and strategic guidance | - Degree of Involvement: Occasional consultations and high-level decision-making | - Degree of Influence: Significant influence on project strategy |-- **Customer: Joe Smith** | - Knowledge Needed: Game requirements and user expectations | - Degree of Involvement: Active participation in defining project requirements | - Degree of Influence: High influence on project direction Head of HR: Emily White Head of Sales: Lisa Adams | |-- Marketing Team: Professor Cornell | - Knowledge Needed: Marketing strategies for game promotion | - Degree of Involvement: Leading marketing efforts for the game | - Degree of Influence: High influence on marketing aspects |-- Legal Experts: Jerry's Law Firm | - Knowledge Needed: Legal compliance and intellectual property rights | - Degree of Involvement: Legal counsel and risk assessment | - Degree of Influence: Moderate influence on legal matters Head of Marketing: David Garcia | |-- **Designers and Developers** | - Knowledge Needed: Game design and development expertise | - Degree of Involvement: Hands-on development and design work | - Degree of Influence: Significant influence on game design and functionality |-- **Software Engineers (Dominick Dupoux, Aidan Ama)** | - Knowledge Needed: Technical skills and coding expertise | - Degree of Involvement: Actively involved in coding and development | - Degree of Influence: Moderate influence on technical decisions |-- **Technology Experts (Jian Zhong & Dylan Dayter)** | - Knowledge Needed: Technology trends and innovations | - Degree of Involvement: Providing insights on tech-related decisions | - Degree of Influence: Moderate influence on technology choices |-- **System Designer: Nick** | - Knowledge Needed: System architecture and design principles | - Degree of Involvement: Leading system design efforts | - Degree of Influence: High influence on system design

2d. The Hands-On Users of the Product

Content

A list of a special type of stakeholder—the potential users of the product. For each category of user, provide the following information:

- User name/category: Most likely the name of a user group, such as clerical users, schoolchildren, road engineers, or project managers.
- User role: Summarizes the users' responsibilities.
- Subject matter experience: Summarizes the users' knowledge of the subject matter/business. Rate as novice, journeyman, or master.
- Technological experience: Describes the users' experience with relevant technology. Rate as novice, journeyman, or master.

- Other user characteristics: Describe any characteristics of the users that have an effect on the requirements and eventual design of the product. For example:

  Physical abilities/disabilities

  Intellectual abilities/disabilities

  Attitude toward job

  Attitude toward technology

  Physical location

  Education

  Linguistic skills

  Age group

  Gender

  Ethnic group/s

Motivation

Product is designed for users that want to be educated in finance while playing an enjoyable game

Form

-Inexperience/no experienced users

-School teachers, People with a personal interest in finance, business trying to provide training in basic finance

2e. Personas

Content

Daniel Mitchell, a 27-year-old financial analyst from San Francisco who's passionate about finance and always eager to enhance his financial knowledge. Recently, he stumbled upon a finance game that promised to make learning about investments engaging. Intrigued by the concept, Daniel, a tech-savvy enthusiast, gave it a shot. With a love for continuous learning, this game seemed like the perfect match to fuel his financial aspirations and make his journey towards financial success even more exciting.

Motivation

A professor in Finance

Form

Professor James Anderson, a 53-year-old finance expert. He's been teaching finance for over 20 years and doing research in the field. James is all about being careful and dependable. He wants a tool to use to tech his class on the basics of finance in a fun and engaging way

2f. Priorities Assigned to Users

Content

Attach a priority to each category of user. This identifies the importance and precedence of the user. Prioritize the users as follows:

- Key users: They are critical to the continued success of the product. Give greater importance to requirements generated by this category of user.
- Secondary users: They will use the product, but their opinion of it has no effect on its long-term success. Where there is a conflict between secondary users' requirements and those of key users, the key users take precedence.
- Unimportant users: This category of user is given the lowest priority. It includes infrequent, unauthorized, and unskilled users, as well as people who misuse the product.

The percentage of the type of user is intended to assess the amount of consideration given to each category of user.

Motivation

There is no large customer order for the product we are just putting it

out to the public

Form

User importance rating is very high as they drive the success of the product

2g. User Participation

Content

Sign here_____

To acknowledge the fact of user participation

Motivation

Sign here_____ to accept responsibility in development.

Form

Estimated user participation time: 20hours

2h. Maintenance Users and Service Technicians

Maintenance User

User Role	Maintenance Worker
User Name/Representative	Maintenance User
Relevant Characteristics	-Knowledgeable about video games -Documentation skills -Communication skills

Service Technician

User Role	Patches Bugs on the product
User Name/Representative	Service Technician
Relevant Characteristics	-Proficient in coding -Knowledgeable about video games -Documentation skills

3. Constraints

3a. Solution Constraints

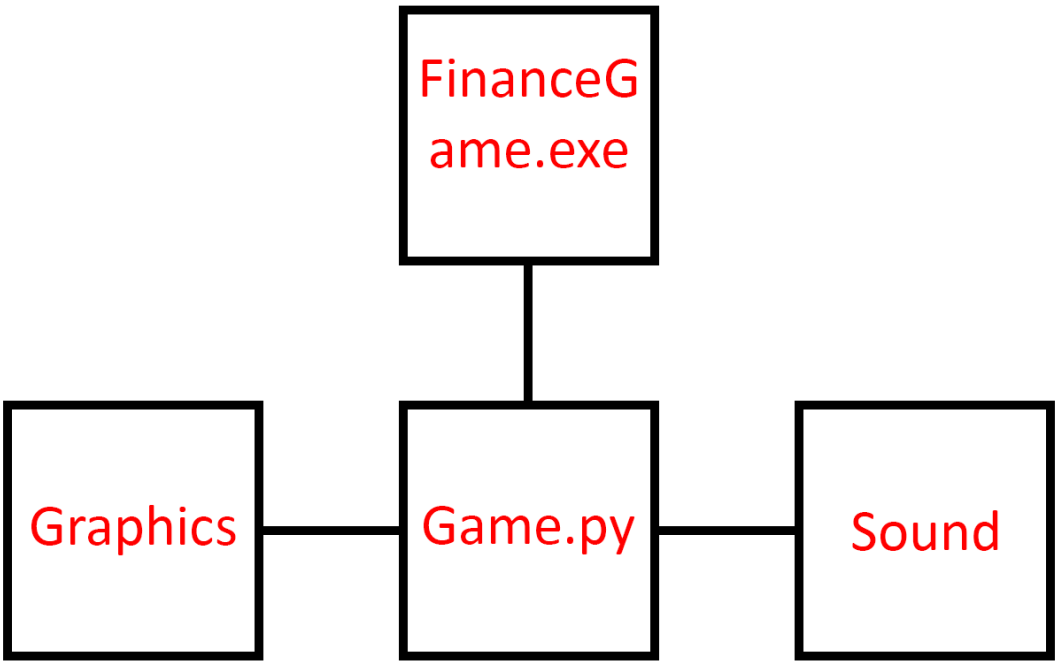
Requirement #1: Investing System	Requirement Type: Functional	Event/Use Case #'s:
Description: The players will need to be able to invest in stocks in order to make money in the game.		
Rationale: Investing in stocks in a core concept of the product, as such it is essential to the gameplay.		
Originator: Dylan Dayter		
Fit Criterion: The product will be tested to make sure the investing system is implemented properly.		
Customer Satisfaction: 5	Customer Dissatisfaction: 5	
Priority: High	Conflicts: None	
Supporting Materials: None		
History: Waiting to be implemented.		

Requirement #1: Go to Next Day	Requirement Type: Functional	Event/Use Case #'s:
Description: A system to go to the next day after the player has completed everything they wanted.		
Rationale: The player will go to sleep (in game) and wake up the next day to see how their investments went. This will also act as a save system for the game where after each day, progress is autosaved.		
Originator: Aidan Amarante		
Fit Criterion: The product will be tested to make sure the ability for the player to sleep is implemented properly.		
Customer Satisfaction: 5	Customer Dissatisfaction: 4	
Priority: High	Conflicts: The investment system needs to be completed before this one can be made and tested.	
Supporting Materials: None		
History: Waiting to be implemented.		

.

3b. Implementation Environment of the Current

System



3c. Partner or Collaborative Applications

We currently are not using any partner or collaborative Applications.

3d. Off-the-Shelf Software

Python is an off-the-shelf software that will be used to make the product. Python can be used to meet all the requirements of the product. It has access to a vast amount of libraries which all can be used to make a product faster and better. Some of the prominent ones for this product are pygame, Matplotlib, and SQLite. The game is going to run on Python as such, it is essential for the product to be made.

Pandas is another off-the-shelf software that will be used to make the product. Pandas will be used for data analysis. This is important for the "Investing System" requirement, because stocks are how the user will invest in the game. Pandas will be used for the data about the stocks.

3e. Anticipated Workplace Environment

There is no official workplace. All developers work on the product from their room. Due to this, the environment varies from developer to developer.

3f. Schedule Constraints

The deadline of our product is November 28th, 2023. This is due to UAlbany's showcase being the next day, which is where we have to present our product. Not meeting the deadline means an unfinished product will be presented.

3g. Budget Constraints

The budget is undefined, but all the things used to make this product are free to use, so it could be stated as $0. For the foreseeable future, this will not change.

3h. Enterprise Constraints

This section is not applicable. There is no enterprise investing in the product we are making.

4. Naming Conventions and Terminology

4a. Glossary of All Terms, Including Acronyms, Used by Stakeholders Involved in the Project

Auto-save: Short for automatically save. This is a feature where progress is automatically saved at either an interval or when a checkpoint is reached.

Python: A high-level programming language.

Bug: An unintentional function or result of a piece of code.

5. Relevant Facts and Assumptions

5a. Relevant Facts

Content

Relevant facts about our game may include the different types of finance concepts we would like to include, the different gameplay mechanics, or any external factors which affect the overall game development.

- The game will be built using PyGame
- The game's narrative will incorporate real-world scenarios/challenges, allowing players to make choices that

impact their financial situation
- The game will implement concepts like budgeting, saving, and investing
- It will be an open-world city where players can access a multitude of financial institutions (banks, stock exchanges, etc.)

Motivation

Relevant facts about the motivation behind things will provide background information and display why certain decisions were made in regards to design.

Examples

- Players may be different ages or have different experience with finances, influencing how we should introduce terms and concepts
- Our open-world concept will allow learning to be more immersive as players will actively engage with real financial institutions
- Using an in-game currency will allow players to actually apply the knowledge they are getting from the game

5b. Business Rules

Content

Business rules that will impact our project are more-so in-game. They will include things such as in-game transactions, character interactions, and the overall educational content we provide.

- Characters in the game must provide factual finance information (e.g. if there is a financial advisor, the advisement must be accurate)
- The player of the game cannot learn advanced concepts without first learning/knowing the basics (if the user does not know how to deposit money in the bank, they should not be trading stocks)

Motivation

Business rules here guide the development and ensure consistency in how financial concepts are presented and taught.

Examples

- Players cannot invest in anything before assessing the risk associated with that particular investment
- Any financial advice given in the game should be accurate and allow players to rely on said advice
- Players must keep a certain amount of money in their bank accounts to avoid penalties and fees, encouraging good finance management

Form

Risk Assessment: Players cannot invest in anything before assessing the risk associated with that particular investment.

Reason for the Rule: To emphasize the importance of informed decision-making in financial investments.

Authority for the Rule: In-game financial regulatory body.

Financial Advice Accuracy: Any financial advice given in the game should be accurate and allow players to rely on said advice.

Reason for the Rule: To provide players with reliable financial guidance that reflects real-world financial practices.

Authority for the Rule: The in-game financial advisory board.

Bank Account Balance: Players must keep a certain amount of money in their bank accounts to avoid penalties and fees, encouraging good finance management.

Reason for the Rule: To promote responsible financial behavior and mimic real-world banking practices.

Authority for the Rule: The virtual bank within the game.

5c. Assumptions

Content

Assumptions for the game are the factor we may take for granted, but should be made explicit

- Players will know how to work a computer game
- PyGame will be suitable throughout development of the whole project

Motivation

By declaring assumptions, we become aware of any project risks that may not have been obvious prior.

Examples

- We assume that the player will know how to work a computer game and will likely not educate them
- We assume the concept's of finance will remain relevant to the player's throughout the game

Considerations

- Are there any specific things that you, as stakeholders, would like us to implement into our game?
- What software tools do you as stakeholders expect to be available for our finance game?
- Should we consider leveraging existing products into our game?

Form

Familiarity with computer games:

- Written statement: It will be assumed that all players of the game will have a basic understanding of how to work a computer game (including input devices and navigating interfaces).
- Effect if assumption is false: If players do not know how to work a computer game, we must educate them on how and create a friendly UI that ensures they are able to work and enjoy the game.

PyGame suitability:

- Written statement: It will be assumed that PyGame will be effective throughout the entire development of our project
- Effect if assumption is false: If PyGame is unsuitable at all, we must use alternative frameworks and adapt to create our game. It could change our whole development approach, but we should be prepared for the absolute worst to stay on track.

6. The Scope of the Work

6a. The Current Situation

Content

This is an analysis of any existing business processes including both manual and autonated processes that may be replaced or changed by our new finance game. It will include business models that demonstrate

how financial processes are carried out within organizations.

Motivation

We must understand current financial processes in order to assess the impact certain changes will have and determine the best approach to integrate our game.

Form

Many different methods can be used to build business process models such as activity diagrams, dataflow diagrams, and business process models.

6b. The Context of the Work

Content

In this section, a work context diagram is developed to delineate the extent of the finance game's scope and its interconnectedness with various systems and entities. This encompasses hardware, software, individuals, and organizational components. Additionally, it encompasses the interfaces through which the finance game interacts with neighboring systems.

Motivation

Defining context allows to establish clear boundaries for the project and ensures that the finance game will be integrated perfectly

Considerations

Form

The above work context diagram is a graphical representation of the inputs and outputs that flow between the finance game and adjacent systems.

6c. Work Partitioning

Content

This is a list of all business events to which the finance game responds. The response to each event is called a business use case (known as a BUC); it represents a discrete piece of work that contributes to the total functionality of the work.

Motivation

Identifying and listing business events and their corresponding BUCs helps break down the project into manageable units and serves as a basis for discovering detailed requirements. Each BUC can be studied independently.

Considerations

- Event name: Real-World Financial Market Data Retrieval
- Input: User request for specific market data
- Output: Real-time financial market data
- Brief summary of business use case: Users request real-time financial market data for specific stocks, indices, or commodities, which is sourced from the "Real-World Financial Market Data" system to inform their in-game investment decisions.

- Classes of business data relevant to this event: (you won't know this early in the study of the event, as you go into detail you will start to understand the essential data and you can add it to the event list.)

- Event name: External Event Trigger
- Input: Real-world event data (e.g., news affecting markets)
- Output: Activation of corresponding in-game events or scenarios
- Brief summary of business use case: Real-world events, such as market news or economic reports, trigger in-game events or scenarios within your finance game, impacting market conditions and player strategies.
- Classes of business data relevant to this event: (you won't know this early in the study of the event, as you go into detail you will start to understand the essential data and you can add it to the event list.)

- Event name: In-Game Economic System Management
- Input: User interaction with in-game economic systems
- Output: In-game economic system updates, player progress
- Brief summary of business use case: Users engage with in-game economic systems, such as buying/selling assets, managing portfolios, or participating in virtual economic activities, leading to updates in the in-game economy and player progress.
- Classes of business data relevant to this event: (you won't know this early in the study of the event, as you go into detail you will start to understand the essential data and you can add it to the event list.)

- Event name: Access Educational Content
- Input: User request for specific educational content
- Output: Display of requested educational content

- Brief summary of business use case: Users can access educational content within the game, which is provided by the "Educational Content" system. This content helps them learn about finance concepts and strategies.
- Classes of business data relevant to this event: (you won't know this early in the study of the event, as you go into detail you will start to understand the essential data and you can add it to the event list.)

6d. Specifying a Business Use Case (BUC)

Content

Real-World Financial Market Data Retrieval

When the "Real-World Financial Market Data Retrieval" event occurs:

- The finance game should promptly respond to the user's request for specific market data, accurately capturing and validating user input.
- It should initiate communication with the "Real-World Financial Market Data" system to retrieve real-time financial market data.
- After receiving the data, the game must process and format it for user-friendly presentation.
- The formatted real-time financial market data, along with a timestamp, should be displayed for user decision-making.

External Event Trigger

When an external event (e.g., real-world news affecting markets) occurs:

- The finance game should activate corresponding in-game events or scenarios, continuously monitoring external sources for relevant events.
- It should assess the impact, update in-game market conditions, asset prices, and other relevant factors based on the event.
- Players should experience changes in market conditions and scenarios influencing their strategies.

In-Game Economic System Management

Users should be able to interact with in-game economic systems, such as buying/selling assets or participating in virtual economic activities.

- The game should accurately capture and process user interactions, validating and updating in-game economics.
- User participation should result in in-game rewards, progress, or achievements.
- A record of all economic system interactions should be maintained for user tracking.

Access Educational Content

Users should have the option to request and access educational content within the game to learn about finance concepts and strategies.

- The game should facilitate user requests and display requested educational content, ensuring accuracy and relevance.
- Educational content should be sourced from the "Educational Content" system, enhancing user understanding.
- These specifications ensure that each BUC responds effectively to its respective Business Event, enabling the finance game to function seamlessly. The actions and processes align with the inputs and outputs of each Business Event, allowing for precise and consistent implementation within the game.

Motivation

Real-World Financial Market Data Retrieval

The motivation behind these specifications is to ensure that the finance game can promptly respond to user requests for specific market data. Accurately capturing and processing user input, as well as retrieving and presenting real-time financial market data, is essential for informed decision-making within the game.

External Event Trigger

These specifications are motivated by the need for the finance game to respond to real-world events, such as market news or economic reports, by activating in-game events or scenarios. Monitoring and assessing external events and their impact on the in-game world are vital for creating a dynamic and engaging user experience.

In-Game Economic System Management

The motivation here is to allow users to interact with in-game economic systems effectively. Capturing and validating user actions, updating in-game economics, and maintaining a record of interactions are key for user engagement and progression within the game.

Access Educational Content

The motivation is to provide users with access to educational content within the game, enhancing their understanding of finance concepts and strategies. Facilitating user requests and ensuring the accuracy and relevance of educational content contribute to the game's educational value

Considerations

Real-World Financial Market Data Retrieval

Consideration: As the game interacts with external financial data sources, it's important to ensure data security and privacy for users. Safeguarding user inputs and the real-time financial market data is critical.

External Event Trigger

Consideration: Continuous monitoring of external events requires an efficient data source and monitoring system. Ensuring timely and accurate detection of relevant events is crucial for delivering a responsive gaming experience.

In-Game Economic System Management

Consideration: The accurate tracking of user interactions within the economic systems necessitates a robust data management and processing system. Proper data validation and transaction handling are essential for user satisfaction.

Access Educational Content

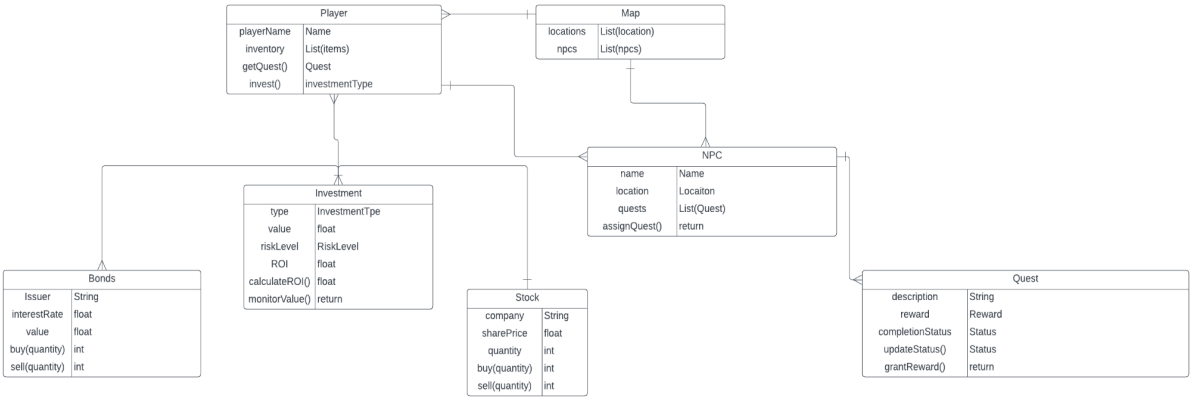
Consideration: When facilitating access to educational content, it's important to maintain content quality and relevance. Regular updates and content curation are key to providing valuable educational resources to users.

Form

A BUC can be specified using any combination of models that suits the analyst. The most common approaches are: activity diagrams, BUC scenarios, process flow diagrams, sequence diagrams, business stories, mind maps, etc. The key is to ensure that the inputs and outputs align precisely with the corresponding Business Event.

7. Business Data Model and Data Dictionary

7a. Business Data Model



7b. Data Dictionary

Name	Content	Type
Player	Player Name, Player Location, Player Inventory	Class
Map	List of interactable locations, List of interactable NPCs	Class
Investment	Type of investment (Stock/Bond), Initial value of investment, risk level of investment(dynamic by time and specific item), return on	Class

	investment.	
NPC(Non-Player character)	Name, location on map(Possibly dynamic), quests(List of quests available to player from specified NPC)	Class
Quest	Description of quest, reward for quest completion, status of quest(Completed, In Progress, not completed)	Class
Location	Place or position on game map Example: Location 1 : Stock Exchange Location 2: Bond office	Attribute
InvestmentType	Categorization of different investments, Stocks Bonds	Attribute
RiskLevel	A measure of risk associated with investments. Bonds have inherently low risk that is static. Stocks have fluctuating risk depending on certain aspects such as price, class, and others.	Attribute
Reward	Prize given to player for completion of quest. In the form of in game currency or item	Attribute
Status	Current state of a quest	Attribute

8. The Scope of the Product

8a. Product Boundary

The finance game we are creating is a python based game that provides an immersive medium for players to trade stocks and bonds while only risking in-game currency. It offers a comprehensive experience allowing players to navigate around a virtual city, engage with NPCs, invest in multiple mediums, complete quests, and converse with NPCs.

A map with multiple locations and NPCs that the player can interact with allows the player to make countless unique investments and complete unique quests.

Investments players can make currently include stocks and bonds, with plans to add real estate investments for a later game objective for players. The non-player characters offer a wide array of quests and tips for the players to follow to improve the experience.

8b. Product Use Case Table

Product Use Case (PUC) Summary Table

PUC No	PUC Name	Actors/Users	Input & Output
1	Explore Map	Players, NPCs	Player controls, preprogrammed NPC routes
2	Make investment	Player	Player investment type and amount. Return on Investment

3	NPC Interaction	Player, NPCs	Player Dialogue choices and actions
4	Complete Quest	Player	Fullfilment of quest requirements
5	Monitor Investment	Player	Access to investment portfolio

8c. Individual Product Use Cases

Use Case: Explore Map

Actor: Player

Scenario:

- Player loads into game and starts moving around map.
- Player walks into NPC or interactable location
- Action results in dialogue interaction with NPC, or being able to explore inside a building.
- Could trigger quest start or unlock new game mechanic.

Design Rationale:

Allowing players to explore an open world map will allow players to navigate the game at their own pace and learn mechanics at their own pace.

Making Stock Investment

Actor: Player

Scenario:

- Player meets an NPC or location that allows for investment.
- Game presents the player with a list of stock/bonds, current price, and risk associated with specific investment.

- Player chooses an option and continues to go throughout the game.
- Players then can decide to sell stock at fluctuating prices in hope to profit.

Design Rationale:

Allows players to take their own risks with quest rewards and gives the opportunity to grow in-game balance.

Interacting with NPCs

Actor: Player

Scenario:

- Player approaches NPC attaining quest
- NPC presents quest details and requirements
- Player accepts quest and attempts to fulfill requirements
- Upon completion of quest player goes back to NPC and collects rewards

Design Rationale:

Keeps the player engaged in the game. Quests offer a sort of game inside a game to further player progression and teach player game mechanics.

Monitoring Investments

Actor: Player

Scenario:

- Player accesses a personal portfolio either through inventory or can go to a map location.
- UI will display player investment and percent gain/loss as well as currency gain/loss
- Visual aids will graph player investments over time as well as allowing the option to sell.

9. Functional Requirements

9a. Functional Requirements

User Registration and Profile Management:

- User signup/login**,** Users can create accounts with unique usernames and passwords.
- User profile management, Users can manage their profiles, including viewing their current assets, investment history, and achievements.

In-Game Navigation:

- Players can navigate the game world represented as a rectangular area of town.
- Players can visit different places within the town, including the stock market, cryptocurrency exchange, real estate market, and precious metal shops.

Investment Options:

- Players can invest in stocks, cryptocurrencies, real estate properties, and precious metals.
- Each investment option has unique characteristics, risks, and potential returns.
- Players can view real-time market data and trends for informed decision-making.

Resource Management:

- Players start with $1000 and can earn or lose money based on their investment decisions.
- Players can track their funds, assets, and investment portfolios.
- Players receive notifications about investment outcomes and market changes.

Exploration and Events:

- Players can explore different parts of the town, encountering random events and challenges.
- Events can include economic fluctuations, investment opportunities, or financial emergencies.

Achievements and Rewards:

- Players can earn achievements based on their investment performance and in-game accomplishments.
- Achievements unlock rewards, such as unique investment opportunities or in-game items.

Motivation

- **Motivation for User Registration and Profile Management**: Ensures that players have personalized experiences within the game, allowing them to track their progress, achievements, and investments over time. User registration and profile management are essential for creating a sense of ownership and continuity in the game world.
- **Motivation for In-Game Navigation**: Facilitates player exploration and interaction with different aspects of the game world, enabling them to make informed investment decisions. Intuitive navigation is crucial for providing a seamless and immersive gameplay experience.
- **Motivation for Investment Options**: Provides players with diverse opportunities to simulate real-world investment scenarios. Offering various investment options allows players to strategize, take risks, and learn about different financial tools, enhancing their understanding of investment concepts.
- **Motivation for Resource Management**: Allows players to manage their finances and investments, promoting financial literacy and responsible decision-making. Tracking resources, assets, and notifications about market changes enable players to learn from their choices and adapt their investment strategies accordingly.
- **Motivation for Exploration and Events:** Introduces randomness and challenges into the game, reflecting real-world uncertainties. Exploration and encountering events simulate the dynamic nature of financial markets, encouraging players to adapt to unexpected situations and make decisions under various conditions.
- **Motivation for Achievement and Rewards**: Enhances player engagement and provides a sense of accomplishment. Achievements and rewards serve as incentives, motivating players to explore different investment strategies, overcome challenges, and aim for in-game milestones.

Fit Criterion

- **Test Cases for User Registration and Profile Management:**

Test Case 1: Verify that a new user can register with a unique username and password.

Steps:

1 Navigate to the registration page.

2 Enter a unique username and password.

3 Click on the "Register" button.

Expected Result: The system should create a new user account. The user should be able to log in with the provided credentials.

Test Case 2: Verify that users can view their profile, including current assets, investment history, and achievements.

Steps:

1 Log in to the user account.

2 Navigate to the profile section.

Navigate to the profile section.

Expected Result: The user should see their current assets, investment history, and achievements displayed accurately on the profile page.

- **Test Cases for In-Game Navigation:**

Test Case 1: Verify that the player can move within the town area using arrow keys or touch controls.

Steps:

1 Start the game.

2 Attempt to move in different directions using arrow keys or touch controls.

Expected Result: The player's character should move in the corresponding directions without any glitches or delays.

Test Case 2: Verify that the player can visit different places within the town.

Steps:

Move the character to various locations (stock market, cryptocurrency exchange, real estate market, precious metal shops).

Expected Result: The player should be able to navigate to different places within the town, triggering specific interactions at each location.

- **Test Cases for Investment Options:**

Test Case 1: Verify that the player can invest a specific amount of money in stocks.

Steps:

1 Go to the stock market.

2 Select a stock to invest in.

3 Enter the investment amount.

4 Confirm the investment values and shares values.

Expected Result: The player's funds should decrease by the invested amount, and the investment should be reflected in the player's portfolio.

Test Case 2: Verify that the player can invest in cryptocurrencies.

Steps:

1 Go to the cryptocurrency exchange.

2 Select a cryptocurrency to invest in.

3 Enter the investment amount.

4 Confirm the investment..

Expected Result: The player's funds should decrease by the invested amount, and the investment should be reflected in the player's portfolio.

- **Test Cases for Resource Management:**

Test Case 1: Verify that the system accurately tracks the player's funds, assets, and investment portfolios.

Steps:

1 Make multiple investments in different assets (stocks, cryptocurrencies, real estate, precious metals).

2 Monitor the player's funds and portfolio changes.

Expected Result: The player's funds and portfolio should be updated correctly after each investment, reflecting the current financial status accurately.

Test Case 2: Verify that players receive real-time notifications about market changes.

Steps:

1 Wait for market changes (stock price fluctuations, investment outcomes).

2 Verify if there are notifications and changes in stock price.

Expected Result: Players should receive notifications promptly when there are changes in the market, providing them with timely information about their investments.

- **Test Cases for Exploration and Events:**

Test Case 1: Verify that players can explore different parts of the town and encounter random events.

Steps:

1 Move the character to various town locations.

2 Encounter random events.

Expected Result: Players should be able to explore different areas of the town, and events should occur at random intervals, affecting the player's financial status accordingly.

Test Case 2: Verify that event outcomes are diverse and offer unique challenges or opportunities.

Steps:

1 Encounter multiple events with different descriptions and contexts.

Expected Result: Each event should have a distinct outcome, presenting players with varied challenges or opportunities, enhancing the game's complexity and realism.

- **Test Cases for Achievements and Rewards:**

Test Case 1: Verify that players receive achievements upon completing specific milestones.

Steps:

1 Accomplish in-game milestones (e.g., reaching a certain investment level, making profitable investments)

Expected Result: Players should receive corresponding achievements in their profiles upon completing the specified milestones.

Test Case 2: Verify that achievements unlock rewards, such as new investment options or in-game items.

Steps:

1 Encounter multiple events with different descriptions and contexts.

Expected Result: Verify that achievements unlock rewards, such as new investment options or in-game items.

Considerations:

- **Performance**: ensure the game is fast to load, would not consume overly large amount of memory.
- **Security**: Ensure that user data is encrypted and stored securely to protect user privacy.
- **Unique Identifiers**: Implement mechanisms to guarantee unique usernames to prevent conflicts and ensure individuality.
- **Data Integrity**: Verify that user profiles store data accurately, including assets,

investment history, and achievements.

- **Smooth Transition**: Ensure seamless movement between different areas of the town to enhance player experience.
- **Intuitive Controls**: Design controls that are intuitive and easy to understand, catering to both PC and mobile platforms.
- **Visual Consistency**: Maintain a consistent visual style across different town locations to provide a cohesive game world.
- **Balanced Opportunities**: Balance risk and reward for each investment option to provide diverse strategies for players.
- **Transaction Accuracy**: Ensure that investments are processed accurately, reflecting changes in player funds and portfolios immediately.
- **Realism**: Implement realistic market fluctuations and investment outcomes to provide an educational experience for players.
- **Timely Notifications**: Ensure that notifications about market changes are delivered promptly to inform players of their investment status.
- **Transparent Feedback**: Provide clear feedback on investment outcomes, indicating gains or losses, to help players understand their decisions.
- **Diversity**: Create a diverse set of events that simulate real-world economic scenarios, ensuring players face varied challenges and opportunities.
- **Randomness**: Implement random event triggers to maintain unpredictability, adding replay value to the game.
- **Impactful Events**: Design events that have meaningful consequences, affecting player finances and requiring strategic decision-making.
- **Meaningful Achievements**: Develop achievements that represent significant milestones, encouraging players to explore different aspects of the game.

Form

Spread sheet would be the form to manage the atomic requirements, atomic requirements might include the following:

- **Requirement ID:** Unique identifier for the requirement.
- **Requirement Description**: Clear and concise description of the requirement.
- **Rationale**: Why this requirement is necessary for the game.
- **Dependencies**: Any other requirements or components this requirement depends

on.

- **Acceptance Criteria**: Specific conditions that need to be met for this requirement to be considered complete.
- **Constraints**: Any limitations or restrictions related to this requirement.
- **Priority**: Importance level (e.g., High, Medium, Low).
- **Status**: Current status of the requirement (e.g., Proposed, In Progress, Completed).

10. Look and Feel Requirements

10a. Appearance Requirements

Visual Design:

- **Art Style**: The game will feature an enchanting art style inspired by the world of Pokémon, combining vibrant colors, captivating designs, and adorable characters to create a visually appealing and approachable learning environment.
- **Character Design**: Player characters, mentors, and financial experts will be designed in a charming and friendly manner, akin to Pokémon characters, to establish a reliable connection with players of all ages.
- **Investment Representation**: Stocks, cryptocurrencies, real estate properties, and precious metals will be creatively represented in the game, ensuring that investment concepts are visually intuitive and easy to understand, even for beginners.

Accessibility:

- **Color Contrast:** All UI elements and visuals will maintain clear color contrast, ensuring readability and accessibility for players with diverse color vision abilities.
- **Text Clarity:** Use easily readable fonts and incorporate clear speech bubbles for characters, ensuring that all in-game text and educational information are presented in a comprehensible manner.

Audio Design:

- **Background Music**: The game will feature a delightful soundtrack that captures the spirit of adventure and curiosity, mirroring the uplifting melodies found in Pokémon games (8-bit Music alike), enhancing the learning experience.

- **Sound Effects**: Interactive elements, such as selecting investment options, successful investments, and learning achievements, will be accompanied by cheerful and encouraging sound effects, providing positive reinforcement and feedback.

10b. Style Requirements

Consistency and Coherence:

- **Visual Consistency:** Maintain a consistent visual theme inspired by Pokémon throughout the game, ensuring that characters, animations, and investment options are visually coherent and create a unified game world.
- **Audio Consistency:** Maintain consistent audio quality and themes, aligning music and sound effects with the game's overall tone and educational objectives, fostering an immersive learning experience.

Animation and Effects:

- **Character Animation**: Characters and creatures will have lively animations, characters should show a happy face when an investment decision is making profit, a sad face when an investment is losing money, fostering a positive and engaging atmosphere.
- **Investment Feedbacks:** Engaging visual effects (e.g., confetti, cheerful animations) will also accompany successful investments and learning milestones, celebrating achievements and reinforcing the learning process. For instance,

Cultural Sensitivity:

- **Inclusivity:** Ensure that characters and environments represent diverse cultures and backgrounds.
- **Language Accessibility:** we can provide language localization options, allowing players from different regions to learn about investments in their native languages, fostering understanding and accessibility for learners worldwide.