



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE

BACHARELADO EM TECNOLOGIA DA INFORMAÇÃO

# Definição do Projeto de Banco de Dados

## *Shop Of Amazing Products*

Janeto Erick da Costa Lima

João Gabriel Reis Saraiva de Andrade

Thiago Vinícius Azevedo de Oliveira

Natal - RN

Julho de 2023

# Sumário

|  |          |
|--|----------|
| <b>1. Descrição da Visão do Produto:</b>                         | <b>3</b> |
| <b>2. Descrição dos Atores Envolvidos e dos Usuários Finais:</b> | <b>4</b> |
| <b>3. Descrição do Ambiente do Usuário:</b>                      | <b>5</b> |
| <b>4. Principais Necessidades dos Usuários e Envolvidos:</b>     | <b>5</b> |
| <b>5. Alternativas e Concorrência:</b>                           | <b>6</b> |
| <b>6. Visão Geral do Produto:</b>                                | <b>6</b> |
| <b>7. Recursos do Produto:</b>                                   | <b>7</b> |
| <b>8. Outros Requisitos do Sistema:</b>                          | <b>7</b> |
| <b>9. Cronograma do projeto:</b>                                 | <b>8</b> |

# 1. Descrição da Visão do Produto:

O produto idealizado é a construção da lógica de um jogo sobre alquimia, chamado *Shop Of Amazing Products*, também referenciado através do acrônimo SOAP. Nele o jogador utilizará de ingredientes reais e fantásticos em um mundo de fantasia para tratar de diferentes condições dos clientes que procuram a sua loja.

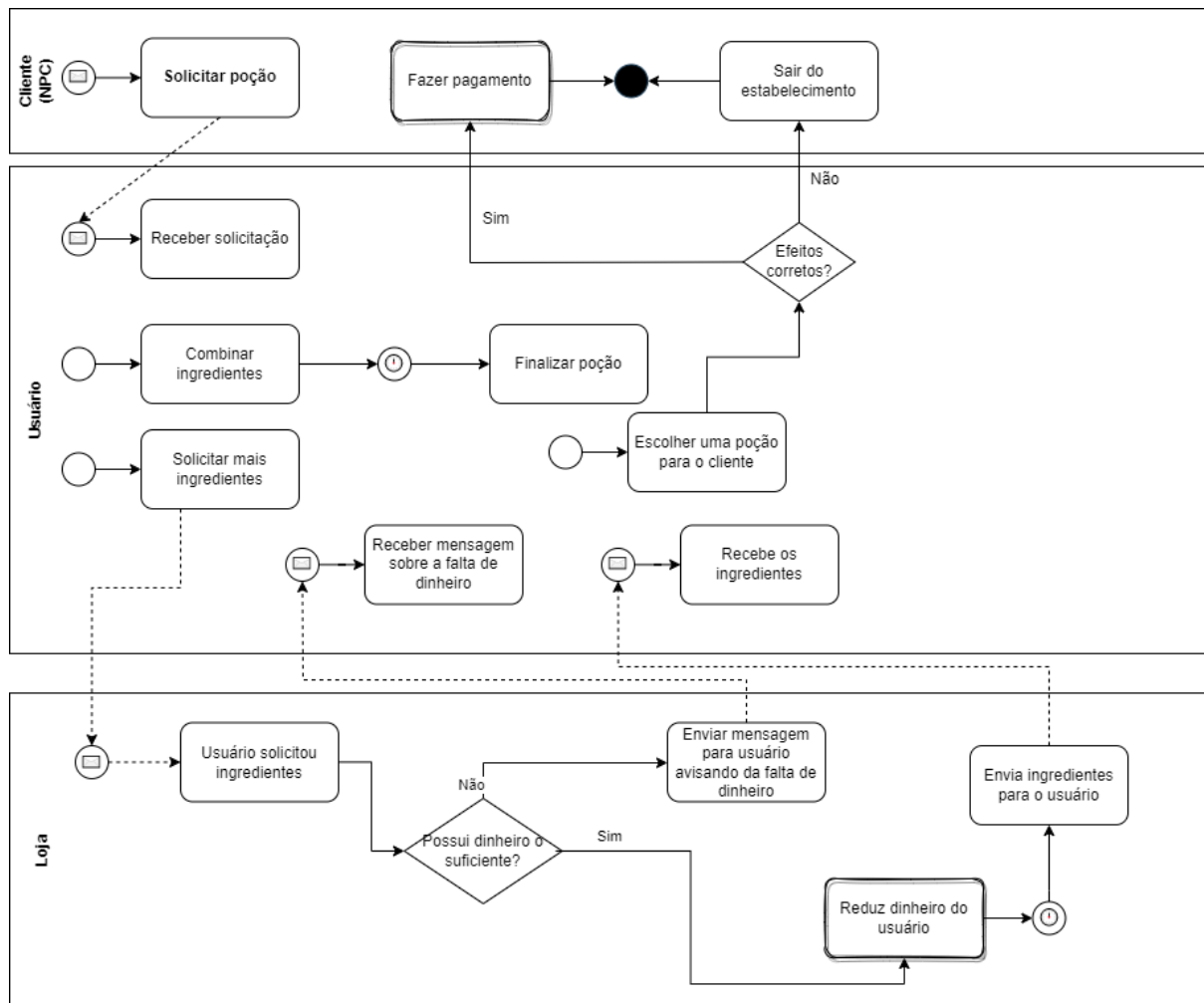
O projeto será o desenvolvimento de um banco de dados capaz de armazenar e gerenciar os dados do jogo, e a API contendo parte da lógica do jogo, para que seja possível apresentar uma prova de conceito sobre o sistema produzido. Ainda dentro do escopo estabelecido para o projeto, criaremos uma interface no terminal para poder simular todas as ações que os jogadores poderão realizar.

O jogo possibilitará, através de um sistema com um banco de dados, o uso de tempo real para o gerenciamento das mecânicas do jogo. Serão gerenciados aspectos como a chegada de novos clientes e o tempo de preparo de cada poção até ficarem disponíveis para o jogador vender.

Uma forte questão na área de jogos digitais é que muitos jogos não suportam uma mesma sessão em múltiplas plataformas diferentes. Com o uso de um sistema online com banco de dados, é possível realizar o gerenciamento de múltiplos jogadores, cada um com uma sessão de jogo, que pode ser acessada através de diferentes dispositivos. Portanto, um jogador pode jogar pelo seu celular enquanto está na rua e ao chegar em casa pode continuar sua sessão de jogo através do computador.

Outra vantagem é que esse banco de dados pode ser agregado a um jogo maior que utilize o esquema de poções implementado no sistema. Dessa forma, os desenvolvedores podem utilizar este produto apenas para gerenciar o módulo do jogo que desejam.

Abaixo, encontrasse um diagrama que ilustra bem os fluxos que serão oferecidos pelo jogo, com todas as funcionalidades que os usuários poderão explorar. Essas funcionalidades estão descritas com mais detalhes na seção 7.



## 2. Descrição dos Atores Envolvidos e dos Usuários Finais:

Os próprios desenvolvedores do jogo são elementos envolvidos de fundamental importância, que terão como foco a implementação do sistema, desde o banco de dados, até a lógica do jogo na API. Estes são Thiago Vinícius Azevedo de Oliveira, João Gabriel Reis Saraiva de Andrade e Janeto Erick da Costa Lima. Todos estes terão como responsabilidade a modelagem do banco, a criação de fato de todas as entidades e relacionamentos do banco de dados, e a lógica do jogo na API.

Como elementos envolvidos e interessados no desenvolvimento do sistema proposto, podemos identificar também os usuários finais, que de fato irão jogar o SOAP. Podemos exemplificar um usuário final, Jorge, como um usuário que possui apenas dispositivos portáteis ao seu dispor. Pelo fato do jogo ser implementado com

um banco de dados e uma API, será possível utilizar esse mesmo sistema para implementar o jogo para múltiplas plataformas, e assim, Jorge poderia jogá-lo normalmente em seu dispositivo móvel.

Podemos considerar como outro possível usuário final, Carlos, que gosta apenas de jogar o jogo em um navegador. Ele também poderia jogar sem precisar instalar nada em seu computador, e até mesmo continuar seu progresso no jogo em uma plataforma diferente, caso deseje.

Além desses casos específicos, podemos considerar que o projeto poderá alcançar qualquer usuário de jogos que se interesse pelo gênero proposto para o SOAP. Todos esses interesses das personas apresentadas como usuários finais deverão ser satisfeitos pelos desenvolvedores do sistema proposto.

### **3. Descrição do Ambiente do Usuário:**

A proposta do SOAP é de ser uma experiência de um jogador por sessão sem interação entre jogadores. A ideia é que o jogador tenha sessões curtas de jogo, entre 5 a 10 minutos, durante múltiplos momentos do dia.

Pretendemos contornar a restrição de uma sessão de jogo por plataforma ao usar um sistema online que gerencie a sessão do usuário e assim permita a utilização de múltiplos dispositivos para jogar. Portanto, o ambiente em que o usuário-alvo irá interagir com o nosso sistema depende apenas de sua escolha e da existência de uma implementação de interface para a plataforma escolhida.

O jogo a ser desenvolvido não dependerá de sistemas alternativos, e funcionará apenas com base nas bibliotecas que virão a ser necessárias.

### **4. Principais Necessidades dos Usuários e Envolvidos:**

Um problema bastante comum no desenvolvimento de jogos de pequeno e médio porte é a possibilidade de montar um jogo que funcione em múltiplas plataformas, principalmente pela falta de tempo e recursos que essas equipes podem investir em portabilidade. Ao criar uma API e um banco de dados comum, é possível utilizá-los para criar o aspecto frontal do jogo em qualquer plataforma, mudando apenas a implementação gráfica do jogo, que de fato dependerá do dispositivo. Isso é de grande importância para os desenvolvedores, que podem montar a lógica do jogo apenas uma vez, e então replicar os visuais para quantas plataformas desejarem, reduzindo o tempo investido no desenvolvimento do projeto. O atendimento desta necessidade é visto com prioridade alta pela equipe do projeto.

Outra questão negativa que pode ser observada em alguns jogos é o incentivo ao jogador estar sempre jogando seu jogo. Em alguns casos, isso pode

levar até a um certo vício por parte dos jogadores. Como o SOAP funciona em tempo real, o jogador só vai precisar jogar quando de fato acontecer algo novo no jogo, como uma nova poção ficar pronta, ou novas demandas de poções surgirem. É um aspecto interessante para trazer para o jogo desenvolvido, pois o jogador tende a ficar menos desgastado do que ficaria se passasse muito tempo jogando. Também tratamos esta necessidade com uma alta prioridade, pois faz parte da identidade que queremos para o nosso jogo.

## **5. Alternativas e Concorrência:**

A concorrência vem por parte de jogos casuais como *candy crush* que servem para partidas de jogo curtas e o *Potion Craft: Alchemist Simulator*, um jogo com temática similar.

Elaborando sobre *Candy Crush* e jogos similares eles possuem como pontos positivos: ser fácil de jogar e custo de entrada nulo. Enquanto seus pontos negativos são: táticas predatórias em relação a seus jogadores, levando a vícios e falta de profundidade em mecânicas, o que pode levar alguns jogadores a não se interessarem pelo jogo.

Sobre o *Potion Craft: Alchemist Simulator*, os pontos positivos dele é o estilo visual, não uso de táticas predatórias e preço competitivo. Enquanto os pontos negativos são: falta de múltiplas plataformas, estando atualmente disponível somente para Windows e, dependendo da perspectiva do jogador, sessões de jogo longas.

## **6. Visão Geral do Produto:**

Nosso produto está sendo proposto com o foco em ser inteiramente independente, ou seja, não necessitará de sistemas adicionais para tornar-se funcional. Portanto, também é da intenção deste projeto que o produto possa ser acoplado a um sistema ainda maior, caso o usuário final deseje. Toda a lógica do jogo será processada através de uma API, que se comunicará com o banco de dados diretamente. Caso o usuário deseje, este poderá obter os executáveis para rodar a API em seu próprio sistema local, e utilizá-la para construir um jogo próprio, que utilize o mesmo modelo de jogo que seguimos para o SOAP. Dessa forma, um outro desenvolvedor poderá usar nossa API para construir um novo jogo, e que incorpore o SOAP como uma de suas fases, por exemplo, e em qualquer plataforma que deseje.

Nossa intenção com o projeto encontra-se em criar uma API com Java que possa ser acessada através de diferentes clientes que implementam uma interface para o jogo. Para efeitos do projeto pretendemos implementar uma interface simples em terminal que exemplifica as mecânicas e processos do jogo.

## **7. Recursos do Produto:**

A ideia central do jogo é habilitar que os jogadores possam misturar ingredientes de diversos tipos, para criarem poções, que serão aplicadas nas pessoas, com base na necessidade delas. Cada ingrediente possui seus próprios efeitos, e quando combinados em uma poção, seus efeitos são somados, podendo gerar uma poção mais forte.

No jogo, as pessoas são personagens genéricos e não jogáveis. Elas possuem como identificador um id, além de possuírem nome e condições que listam os seus estados. Com base nisso, as pessoas solicitam poções ao jogador, que deve combinar os ingredientes para montar a poção que melhor atenda a necessidade da pessoa. Além disso, as pessoas possuem um tempo de vida restante, e caso a doença não seja tratada pelo jogador, o personagem que fez o pedido irá para outro estabelecimento.

Outro fator importante são as alergias da pessoa, um fator que impedirá que o jogador utilize certos ingredientes ao construir a poção. Caso seja dada para uma pessoa uma poção com algum ingrediente que ela possua alergias, ela não poderá ser aplicada. Também é importante salientar que todos os fatores de tempo do jogo serão calculados em tempo real, ou seja, caso uma pessoa tenha 1 hora de espera restante, isso de fato corresponde a uma hora no mundo real.

A verba das pessoas possui um valor base, um multiplicador, e a periodicidade. Dessa forma, ao se passar certo tempo (específico para cada uma), a verba da pessoa irá aumentar, permitindo que o jogador tenha mais alternativas para criação de poções. Uma estratégia interessante seria, em alguns casos, por mais que o jogador já possua ingredientes para montar a poção certa, caso a pessoa ainda tenha tempo de vida suficiente, ele pode esperar a verba da pessoa aumentar, para que possa construir uma poção mais cara, e ganhar mais dinheiro.

Por fim, os ingredientes possuem efeitos atrelados, que se combinarão para montar as poções. Um ingrediente não funciona por si só. Além disso, os ingredientes possuem um custo. Cabe ao jogador gerenciar sua verba para comprar os ingredientes de maneira efetiva, e poder lucrar com a venda de poções.

## **8. Outros Requisitos do Sistema:**

Como requisito para execução do sistema, o usuário final deverá possuir idealmente um sistema operacional Windows 10 ou superior. Como o jogo será executado de um terminal Java, não será necessário nenhum hardware avançado para poder jogar o SOAP, mas será preciso que o usuário possua o Java instalado em sua máquina, na versão 17 ou maior. Isso será o bastante para que o jogo execute qualquer ação o mais imediatamente possível, apenas com o gargalo para realizar a comunicação entre a API e banco de dados.

O jogo deverá ser capaz de aceitar qualquer entrada do usuário, sem que sua execução seja encerrada repentinamente. Além disso, caso a execução do jogo seja interrompida em algum momento, é extremamente necessário que o usuário não perca nenhuma parte de seu progresso. O sistema deve assegurar que todos os dados estão sendo guardados no banco, e que as operações de fato sejam atômicas e seguras.

Todas as opções para utilização das funcionalidades do sistema deverão ser claras, informando ao jogador exatamente o que irá ocorrer após cada ação realizada. O design do jogo deverá ser construído apenas através do terminal. Por fim, o projeto final deverá conter um breve manual de usuário, para que todas as funcionalidades do sistema estejam descritas de maneira clara, assim como o que será necessário para realizar a instalação do sistema.

## 9. Cronograma do projeto:

| Entrega    | Item                                      | Descrição  |
|------------|---|--|
| 04/04/2023 | Documento de Visão Geral do Produto.      | Documento contendo todos os itens que foram planejados para o desenvolvimento do projeto, incluindo todos os fatores necessários para dar início a modelagem do sistema. |
| 14/04/2023 | Diagrama ER do banco de dados.            | Diagrama contendo todas as entidades que farão parte do banco de dados do sistema, incluindo suas relações e quaisquer especificações necessárias.                       |
| 29/05/2023 | Mapeamento para Relacional e Normalização | Realizar o mapeamento para relacional, e a normalização.   |
| 12/07/2023 | Finalização do projeto                    | Entregar o projeto completo e funcional, incluindo o projeto de banco de dados e a API com a lógica do jogo.   |