

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS

PUC Minas Virtual

Pós-graduação *Lato Sensu* em Engenharia de *Software*

Projeto Integrado

Relatório Técnico

Save for You

Larissa Quintão Lana

Belo Horizonte
Junho 2023

Projeto Integrado

Sumário

Projeto Integrado	2
1. Cronograma de Trabalho	3
2. Introdução	5
3. Definição Conceitual da Solução	7
3.1 Diagrama de Casos de Uso	7
3.2 Requisitos Funcionais	8
3.3 Requisitos Não-funcionais	10
4. Protótipo Navegável do Sistema	11
5. Diagrama de Classes de Domínio	12
6. Arquitetura da Solução	16
6.1 Padrão Arquitetural	16
6.2 C4 model - Diagrama de Contexto	16
7. Frameworks de Trabalho	19
8. Estrutura Base do Front End	21
9. Modelo Relacional ou Projeto de Banco de Dados NoSQL	31
10. Plano de Testes	35
11. Apropriação de Horas no Projeto	37
12. Código da Aplicação	39
13. Avaliação Retrospectiva	41
13.1 Objetivos Estimados	41
13.2 Objetivos Alcançados	41
13.3 Lições aprendidas	41
14. Referências	43

1. Cronograma de Trabalho

Datas		Atividade / Tarefa	Produto / Resultado
De	Até		
14/01/2023	15/01/2023	1. Realizar estudos e pesquisas para definição e entendimento do contexto do projeto e seu público-alvo	Rascunho inicial das funcionalidades e identificação dos possíveis stakeholders
16/01/2023	16/01/2023	2. Relatório Técnico: preencher seção “1. Cronograma de Trabalho”	Seção “1. Cronograma de Trabalho” do Relatório Técnico preenchida
17/01/2023	22/01/2023	3. Identificar os objetivos do projeto, definir o escopo e realizar o levantamento dos requisitos	Lista contendo os objetivos e as funcionalidades a serem implementadas
22/01/2023	01/02/2023	4. Prototipar interface, em papel, para validar e testar o escopo definido	Escopo validado e rascunho do protótipo da interface
02/02/2023	05/02/2023	5. Ajustar escopo do projeto	Escopo ajustado e revisado
06/02/2023	10/02/2023	6. Relatório Técnico: preencher seção “2. Introdução”	Seção “2. Introdução” do Relatório Técnico preenchida
11/02/2023	15/02/2023	7. Criar Diagrama de Casos de Uso	Diagrama de Casos de Uso
26/02/2023	28/02/2023	8. Relatório Técnico: preencher seção “3. Definição Conceitual da Solução”	Seção “3. Definição Conceitual da Solução” do Relatório Técnico preenchida
01/03/2023	18/03/2023	9. Criar o protótipo da interface do sistema no Figma	Protótipo navegável
19/03/2023	19/03/2023	10. Avaliar o protótipo criado tomando por base as Heurísticas de Nielsen	Protótipo navegável validado
20/03/2023	25/03/2023	11. Ajustar protótipo	Protótipo navegável ajustado e revisado
26/03/2023	01/04/2023	12. Criar Diagrama de Classes de domínio	Diagrama de Classes de Domínio
02/04/2023	02/04/2023	13. Relatório Técnico: preencher seção “5. Diagrama de Classes do Domínio”	Seção “5. Diagrama de Classes do Domínio” do Relatório Técnico preenchida
02/04/2023	04/04/2023	14. Gravar e editar o vídeo de apresentação do protótipo navegável	Vídeo de apresentação do protótipo navegável
05/04/2023	08/04/2023	15. Relatório Técnico: preencher seção “4. Protótipo Navegável do Sistema” e realizar revisão textual	Seção “4. Protótipo Navegável do Sistema” do Relatório Técnico preenchida e texto ajustado e revisado
08/04/2023	09/04/2023	16. Realizar entrega da Etapa 1	Etapa 1 entregue e finalizada
10/04/2023	15/04/2023	17. Realizar estudos e pesquisas para definição da arquitetura, dos frameworks, linguagens e demais tecnologias a serem utilizadas para desenvolvimento da solução	Listagem das possíveis soluções técnicas a serem utilizadas
16/04/2023	18/04/2023	18. Definir a arquitetura da solução, os frameworks a serem empregados, linguagens e demais tecnologias envolvidas	Detalhes técnicos definidos
19/04/2023	22/04/2023	19. Criar o Diagrama C4 model	Diagrama C4 model
23/04/2023	25/04/2023	20. Relatório Técnico: preencher as seções “6. Arquitetura da Solução” e “7. Frameworks de Trabalho”	Seções 6. Arquitetura da Solução” e “7. Frameworks de Trabalho” do Relatório Técnico preenchidas
26/04/2023	28/04/2023	21. Criar Diagrama do Modelo Relacional	Diagrama do Modelo Relacional

Datas		Atividade / Tarefa	Produto / Resultado
De	Até		
29/04/2023	30/04/2023	22. Criar banco de dados e script dos schemas e tabelas	Banco de dados, schemas e tabelas criados
01/05/2023	06/05/2023	23. Criar Diagrama do Projeto de Banco de Dados NoSQL	Diagrama do Projeto de Banco de Dados NoSQL
07/05/2023	07/05/2023	24. Relatório Técnico: preencher seção “9. Modelo Relacional e Projeto de Banco de Dados NoSQL”	Seção “9. Modelo Relacional e Projeto de Banco de Dados NoSQL” do Relatório Técnico preenchida
08/05/2023	13/05/2023	25. Relatório Técnico: preencher seção “10. Planos de Teste”	Seção “10. Planos de Teste” do Relatório Técnico preenchida
14/05/2023	31/05/2023	26. Desenvolver o Back End da aplicação	Back End da aplicação implementado
01/06/2023	11/06/2023	27. Desenvolver o Front End da aplicação	Front End da aplicação implementado
12/06/2023	12/06/2023	28. Relatório Técnico: preencher seção “8. Estrutura Base do Front End”	Seção “8. Estrutura Base do Front End” do Relatório Técnico preenchida
13/06/2023	17/06/2023	29. Realizar o deploy da aplicação	Aplicação disponibilizada em servidor
14/06/2023	17/06/2023	29. Testar aplicação e realizar correções	Aplicação testada
17/06/2023	17/06/2023	30. Disponibilizar o código da aplicação e demais artefatos em repositório público	Código da aplicação em repositório público
18/06/2023	20/06/2023	31. Gravar e editar vídeo de apresentação do projeto	Vídeo de apresentação do projeto
21/06/2023	23/06/2023	32. Relatório Técnico: preencher seções restantes do relatório técnico, adequar referências às normas ABNT e atualizar diagramas	Relatório Técnico preenchido
24/06/2023	24/06/2023	33. Relatório Técnico: realizar revisão textual	Relatório Técnico revisado e finalizado
25/06/2023	25/06/2023	34. Realizar entrega da Etapa 2	Etapa 2 entregue e projeto finalizado

2. Introdução

O hábito de poupar dinheiro é um tema recorrente no mundo financeiro e sua importância é inegável, especialmente por ser um dos pilares fundamentais para se alcançar uma vida financeira saudável e estável, reduzir dívidas, construir riqueza e garantir um futuro tranquilo.

No entanto, apesar da importância, muitas pessoas ainda encontram dificuldades em manter esse hábito. Conforme uma pesquisa realizada em 2018 pela Confederação Nacional de Dirigentes Lojistas (CNDL) e pelo Serviço de Proteção ao Crédito (SPC Brasil), apenas 35% dos brasileiros costumam poupar regularmente. De acordo com essa mesma pesquisa, o número de poupadores é maior nas classes A e B, chegando a 39%, mas tal número chama atenção, pois revela que até mesmo nos extratos mais elevados de renda o volume de poupadores ainda é relativamente baixo (CNDL/SPC BRASIL, 2018).

Tal realidade foi agravada ainda mais pela instabilidade econômica decorrente da pandemia de COVID-19, uma vez que essa trouxe consigo uma série de desafios e incertezas para a vida financeira das pessoas (FMI, 2020).

Nesse contexto, ter objetivos financeiros bem definidos - sejam eles de curto, médio ou longo prazo, como ter uma reserva de emergência ou garantir a aposentadoria-, e possuir uma estratégia para atingi-los, pode ser um primeiro passo importante para adquirir o hábito e, conseqüentemente, estar mais próximo da estabilidade financeira. Isso porque a motivação para alcançar esses objetivos pode ajudar as pessoas a manterem o foco e a disciplina necessária para tomar decisões mais inteligentes em relação ao dinheiro. Ao estabelecer metas específicas, os indivíduos se sentem mais motivados e engajados a criar um plano para atingi-las, o que pode envolver estabelecer um orçamento, reduzir despesas, aumentar a receita ou investir em oportunidades financeiras. Além disso, alcançá-las pode trazer uma sensação de realização e sucesso, aumentando a autoconfiança e a autoestima. Essa sensação de conquista pode, por sua vez, motivá-las a continuar trabalhando em direção a objetivos financeiros ainda mais ambiciosos (VIRGINIA CREDIT UNION, 2023; M1 FINANCE, 2023).

Contudo, muitas pessoas encontram dificuldades em definir e cumprir esses objetivos, seja por falta de organização, de disciplina ou simplesmente por se sentirem perdidas em relação a como alcançá-los, o que as leva, por muitas vezes, a um ciclo vicioso de frustração, desistência ou falta de comprometimento. E é nesse contexto que surge a necessidade de uma ferramenta capaz de tornar tal atividade mais simples e eficaz.

A motivação para resolver esse problema é tanto social quanto mercadológica. Social, porque uma solução eficaz de gerenciamento financeiro pode ajudar as pessoas a alcançar seus objetivos financeiros e, consequentemente, melhorar sua qualidade de vida. Mercadológica, pois há uma grande demanda por soluções desse tipo.

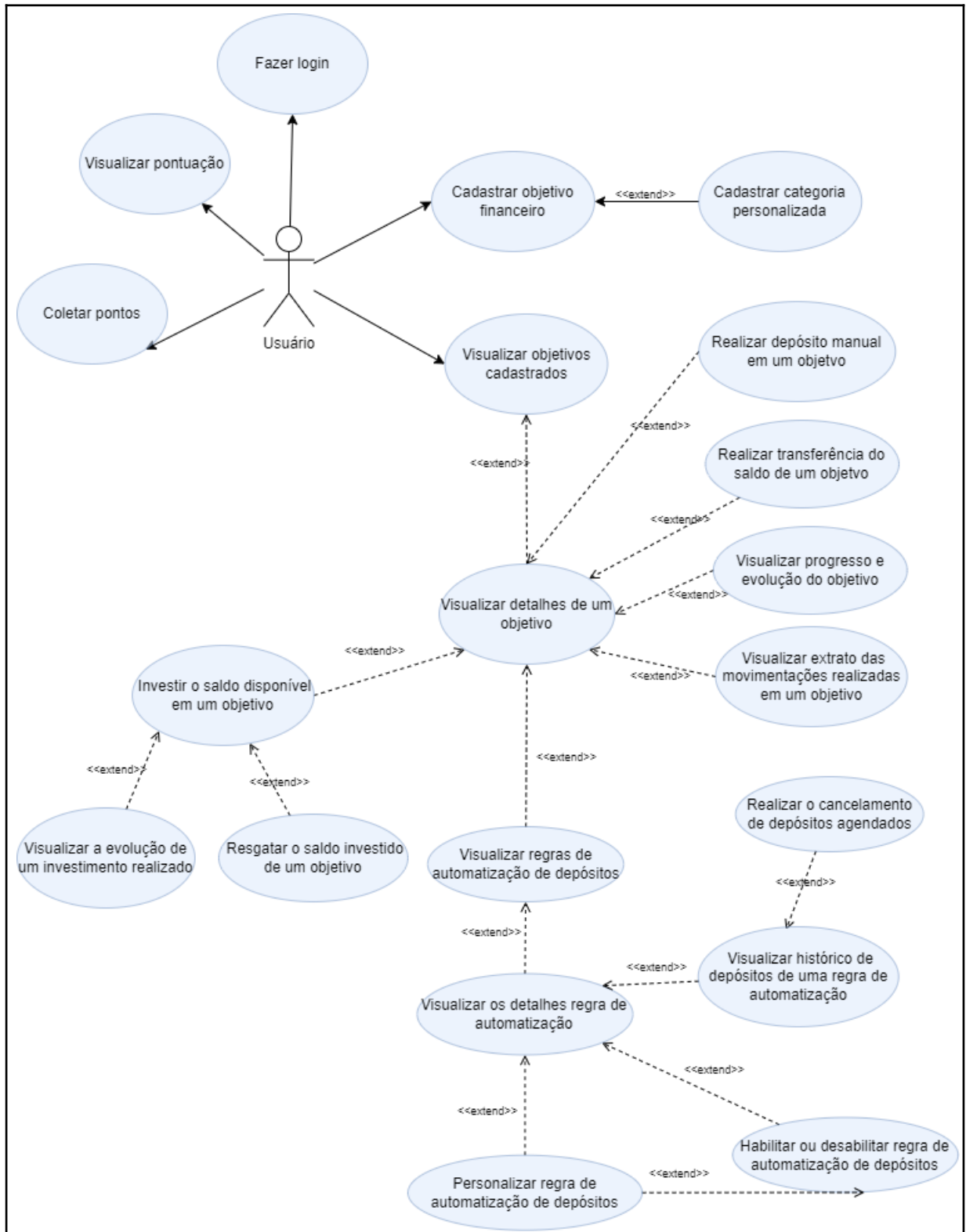
Nesse sentido, o objetivo geral deste trabalho é desenvolver um aplicativo para dispositivos móveis que ajude as pessoas a estabelecerem e alcançarem seus objetivos financeiros de forma prática e eficiente, bem como auxiliá-las a manter uma rotina de poupança regular, a terem maior controle e organização sobre suas finanças pessoais e a melhorarem sua saúde financeira como um todo. Dentre os objetivos específicos incluem:

- Realizar estudos e pesquisas para definição e entendimento do contexto do projeto bem como para identificar as reais necessidades e anseios de seu público-alvo;
- Descrever os requisitos do aplicativo, bem como os demais artefatos necessários para seu desenvolvimento;
- Desenvolver o aplicativo com uma interface amigável e intuitiva, facilitando seu uso por pessoas com diferentes níveis de conhecimento financeiro ou tecnológico.

Espera-se que este aplicativo possa contribuir para o aumento da educação financeira da população brasileira, tornando o hábito de poupar dinheiro mais acessível e fácil de ser incorporado no dia a dia. Espera-se também que ele contribua para que os brasileiros possam estar melhor preparados para enfrentar crises econômicas e realizar seus sonhos e projetos de vida.

3. Definição Conceitual da Solução

3.1 Diagrama de Casos de Uso



3.2 *Requisitos Funcionais*

ID	Descrição Resumida	Dificuldade (B/M/A)*	Prioridade (B/M/A)*
RF01	O sistema deve permitir ao usuário realizar o cadastro (inserção, alteração e exclusão) de um objetivo financeiro. Para cada objetivo devem ser informados: nome, categoria, valor desejado e uma data limite para alcançá-lo.	M	A
RF02	O sistema deve permitir ao usuário realizar o cadastro (inserção, alteração e exclusão) de uma categoria personalizada. Para cada categoria personalizada devem ser informados: nome e um ícone associado.	B	B
RF03	O sistema deve permitir ao usuário visualizar todos os seus objetivos cadastrados separados conforme a situação dos mesmos: “Em progresso”, “Pausado”, “Alcançado”.	B	A
RF04	O sistema deve permitir ao usuário acompanhar o seu progresso em relação a um objetivo financeiro, mostrando o quanto já guardou e quanto ainda falta para atingir o valor desejado.	B	A
RF05	O sistema deve permitir ao usuário habilitar ou desabilitar uma ou mais regras para automatização de depósitos em um objetivo. Para isso deve fornecer uma relação de regras já pré-cadastradas e pré-configuradas.	M	A
RF06	O sistema deve permitir que o usuário personalize as regras de automatização de depósitos definindo a frequência desejada e um valor ou percentual que será depositado automaticamente em determinado objetivo.	A	A
RF07	O sistema deve exibir notificações aos usuários para avisá-los sobre os depósitos automáticos realizados, os que estão próximos de serem realizados bem como lembretes para ajudá-los a manter o foco em suas metas e objetivos financeiros.	M	A
RF08	O sistema deve permitir ao usuário realizar o cancelamento de depósitos agendados.	M	M
RF09	O sistema deve permitir ao usuário visualizar os detalhes de uma regra de automatização de depósito bem como o histórico de depósitos registrados para essa regra, exibindo detalhes como valor, data e categoria.	M	M
RF10	O sistema deve exibir dicas e sugestões aleatórias sobre Educação Financeira e como alcançar determinado objetivo	M	M

ID	Descrição Resumida	Dificuldade (B/M/A)*	Prioridade (B/M/A)*
	mais rapidamente. Essas dicas podem incluir sugestões de investimento, maneiras de economizar dinheiro em despesas cotidianas e outras estratégias úteis.		
RF11	O sistema deve permitir ao usuário realizar depósitos manuais em um objetivo. Para isso devem ser disponibilizadas opções de depósitos via boleto bancário, PIX, TED ou DOC (disponibilizando os dados de sua conta corrente) ou via saldo em conta corrente.	M	M
RF12	O sistema deve permitir ao usuário transferir o saldo disponível em um objetivo. Para isso devem ser disponibilizadas opções para transferências de um objetivo para outro já cadastrado, para conta corrente ou para uma de suas contas de outra instituição financeira.	M	M
RF13	O sistema deve permitir ao usuário visualizar a lista de investimentos disponíveis separados por categoria. As categorias disponibilizadas serão: Renda Fixa, Ações, Fundos de Investimentos e Tesouro Direto.	M	M
RF14	O sistema deve permitir ao usuário investir o saldo disponível em um objetivo.	A	M
RF15	O sistema deve permitir ao usuário resgatar o valor investido de um objetivo.	A	M
RF16	O sistema deve permitir ao usuário visualizar a evolução de um investimento realizado ao longo do tempo.	A	M
RF17	O sistema deve permitir ao usuário visualizar o extrato das movimentações realizadas em um objetivo. Para isso deve listar as movimentações referentes aos depósitos, transferências, agendamentos de depósitos, aplicação em investimentos, resgate de investimentos e quaisquer outras movimentações financeiras.	B	M
RF18	O sistema deve disponibilizar mecanismos de gamificação como forma de engajar os usuários e fortalecer o aprendizado sobre saúde financeira.	M	A

* B = Baixa, M = Média, A = Alta.

3.3 Requisitos Não-funcionais

ID	Descrição	Prioridade B/M/A
RNF01	O sistema deve ser compatível com diferentes smartphones e sistemas operacionais, executando tanto em plataforma iOS quanto Android.	A
RNF02	O sistema deve ser confiável e estar disponível 24 horas por dia, 7 dias por semana. O tempo de indisponibilidade deve ser mínimo (somente 0,01%) e deve ser comunicado com antecedência aos usuários.	M
RNF03	O aplicativo deve responder rapidamente aos comandos dos usuários. O tempo de resposta do aplicativo deve ser de no máximo 3 segundos para garantir a satisfação do mesmo.	M
RNF04	O aplicativo deve funcionar corretamente e sem falhas, evitando perda de dados e interrupções no serviço. O aplicativo deve ter um tempo médio entre falhas (MTBF) de pelo menos 3.000 horas.	M
RNF05	O sistema deve proteger as informações pessoais e financeiras do usuário, garantindo que as informações sejam criptografadas e armazenadas com segurança.	A
RNF06	O sistema deve ser escalável, sendo capaz de suportar, pelo menos, 200 usuários conectados simultaneamente sem problemas de desempenho, disponibilidade ou estabilidade. Além disso, ele deve ser dimensionado adequadamente, para garantir que possa lidar com picos de tráfego e crescimento futuro.	A
RNF07	O sistema deve ser fácil de manter e atualizar, permitindo que os desenvolvedores possam corrigir problemas e adicionar novos recursos rapidamente. O código deve ser bem organizado e documentado e deve seguir as melhores práticas de programação para facilitar sua evolução e manutenção futura.	M

4. Protótipo Navegável do Sistema

O protótipo navegável e interativo do sistema pode ser acessado e visualizado de duas maneiras distintas:

- Por meio do download do vídeo em formato MP4 disponibilizado no GitHub:

<https://github.com/larissalana/save-for-you-docs/blob/main/docs/Etapa%2001/prototipo-navegavel-save-for-you.mp4>

- Ou por meio da ferramenta Figma:

[https://www.figma.com/proto/dmmJBYJWZSJhbamhfSRZBp/Save-for-You-\(Protótipo-navegável\)?node-id=909-2097&starting-point-node-id=909%3A2097&show-proto-sidebar=1](https://www.figma.com/proto/dmmJBYJWZSJhbamhfSRZBp/Save-for-You-(Protótipo-navegável)?node-id=909-2097&starting-point-node-id=909%3A2097&show-proto-sidebar=1)

No protótipo é possível visualizar os principais fluxos da aplicação, representados pela Tela Inicial e os Casos de Uso: “Cadastrar objetivo financeiro”, “Visualizar objetivos cadastrados”, “Visualizar detalhes de um objetivo”, “Visualizar regras de automatização de depósitos”, “Personalizar regra de automatização de depósitos”, “Habilitar ou desabilitar regra de automatização de depósitos”, “Visualizar pontuação” e “Coletar pontos”. Ele foi desenvolvido também visando representar todos os Requisitos Funcionais classificados como prioridade alta e que mais agregam valor ao usuário, quais sejam: RF01, RF03, RF04, RF05, RF06, RF07 e RF18.

Save for you

5. Diagrama de Classes de Domínio

Conforme descrito na seção "Arquitetura da Solução", o padrão arquitetural escolhido foi o de Microsserviços, sendo assim, cada uma das figuras seguintes representa o Diagrama de Classe de Domínio de um determinado Microsserviço. Eles também podem ser acessados no repositório do projeto por meio do link:

- <https://github.com/larissalana/save-for-you-docs/tree/main/docs/Etapa%2002/Diagramas%20de%20Classe%20de%20Dom%C3%ADnio>

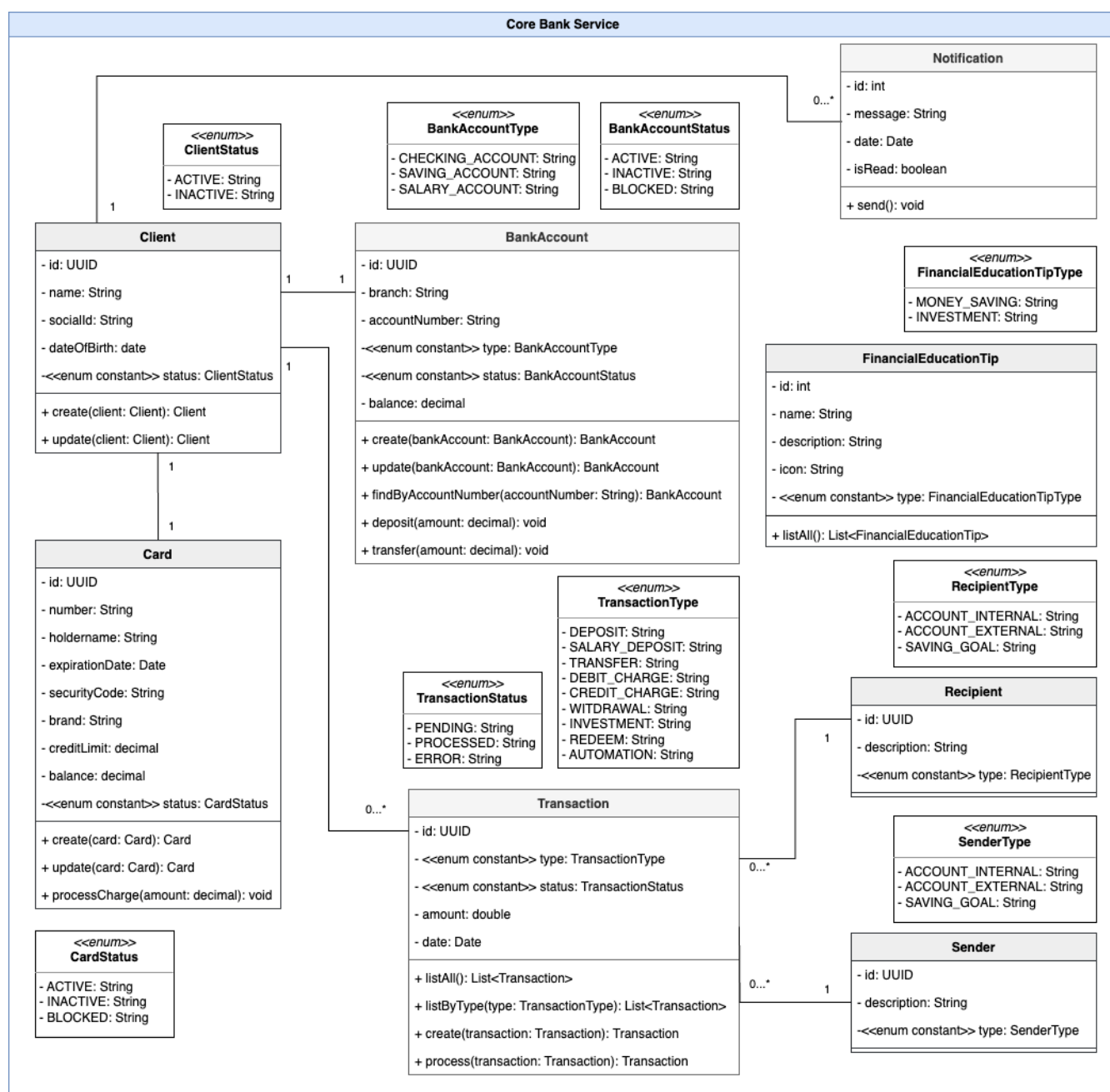


Figura 2 - Diagrama de Classe de Domínio - Core Bank Service

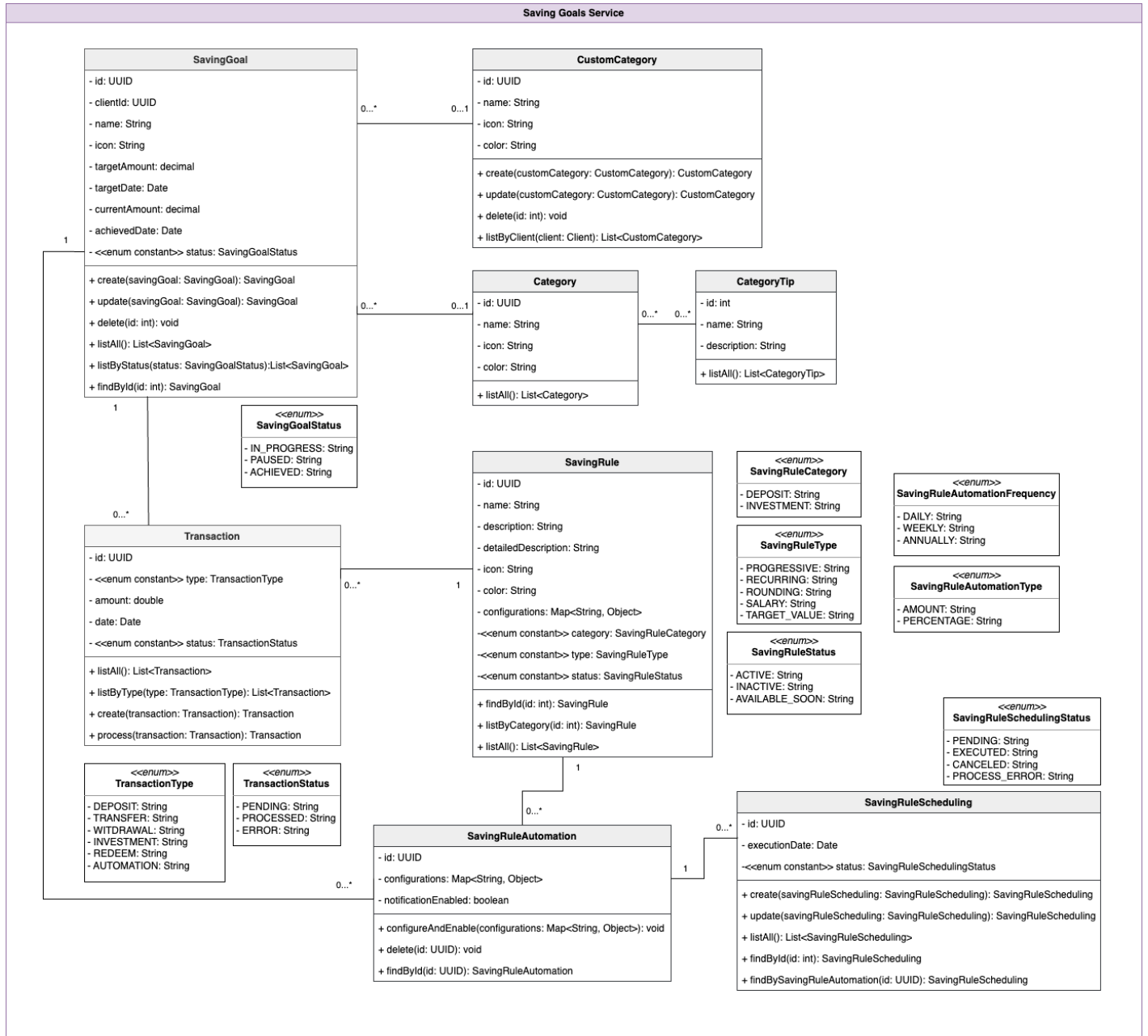


Figura 3 - Diagrama de Classe de Domínio - Saving Goals Service

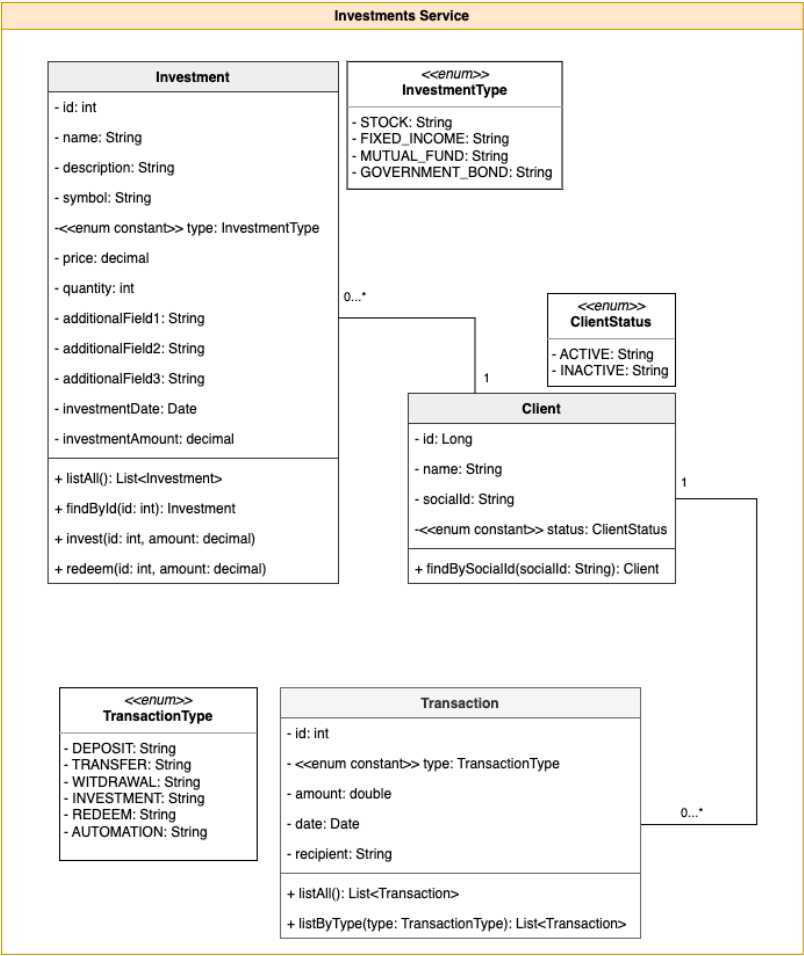


Figura 4 - Diagrama de Classe de Domínio - Investments Service

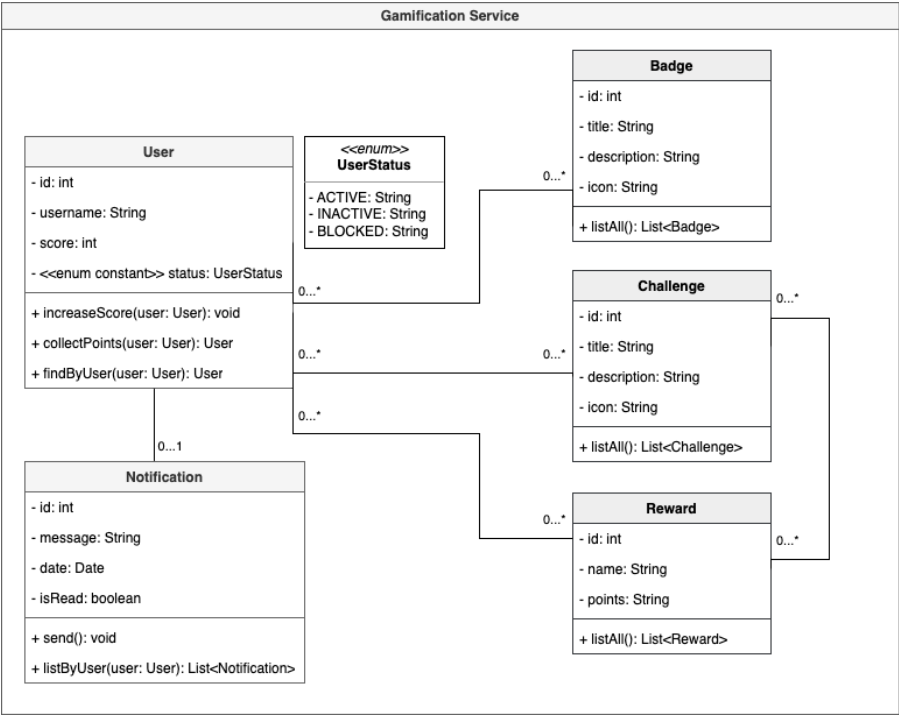


Figura 5 - Diagrama de Classe de Domínio - Gamification Service

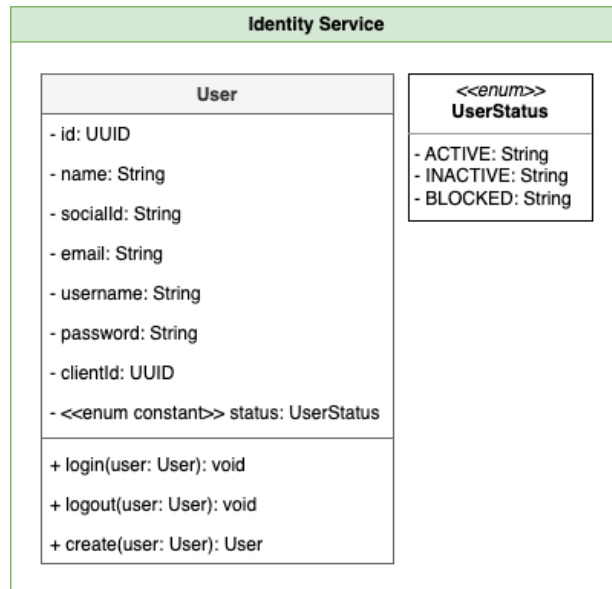


Figura 6 - Diagrama de Classe de Domínio - Identity Service

6. Arquitetura da Solução

6.1 Padrão Arquitetural

A arquitetura escolhida para a solução foi a Arquitetura de Microsserviços. Isso porque tal arquitetura é comumente utilizada para soluções complexas, como do aplicativo proposto, que apresenta uma série de domínios diferentes, pois permite que cada um desses domínios sejam desenvolvidos de forma separada, o que diminui a complexidade e favorece o desenvolvimento ágil e assertivo, uma vez que cada serviço pode se concentrar em uma funcionalidade específica. Além disso, ela torna possível o uso de tecnologias diferentes em cada microsserviço, o que pode ser vantajoso para aproveitar as melhores ferramentas e linguagens em cada contexto da presente solução.

6.2 C4 model - Diagrama de Contexto

No topo do diagrama encontra-se um desenho representando o usuário do aplicativo da solução que foi nomeado como "Save for You". Logo abaixo dele, é possível visualizar o aplicativo móvel que é a interface que os usuários instalam em seus dispositivos móveis para se comunicar com a aplicação.

Essa interface se comunica com uma API Gateway após uma requisição ter sido feita pelo usuário. A API Gateway atua como ponto de entrada para o sistema e é responsável por distribuir as chamadas recebidas entre os diferentes microsserviços da solução, quais sejam:

- **Identity Service:** responsável pela autenticação e autorização dos usuários. Ele valida as credenciais do usuário e emite tokens de acesso para uso posterior em chamadas subsequentes;
- **Core Bank Service:** gerencia a funcionalidade de conta bancária do aplicativo. Lida com a criação de contas, consultas de saldo, transferências de fundos e outras transações relacionadas à conta bancária;
- **Saving Goals Service:** responsável por gerenciar os objetivos financeiros do usuário, bem como as regras de automatização e transações relacionadas;

- **Investments Service:** responsável por oferecer informações e recursos relacionados a investimentos, como análises de mercado, opções de investimento e acompanhamento de desempenho;
- **Gamification Service:** responsável pela funcionalidade de gamificação do aplicativo. Ele oferece recursos para definir e rastrear objetivos financeiros, recompensas, desafios e conquistas para incentivar os usuários a alcançarem seus objetivos financeiros.

As comunicações entre o aplicativo, a API Gateway e os microsserviços são baseadas em protocolo HTTP/REST, permitindo a troca de dados no formato JSON e a realização das operações desejadas.

O aplicativo instalado no celular do usuário foi desenvolvido em React Native e TypeScript. Enquanto que cada um dos microsserviços e a API Gateway foram construídos utilizando Spring Boot e Java. Além disso, por ser uma arquitetura baseada em Microsserviços e como tal arquitetura propõe, existe um banco de dados específico para cada um dos serviços:

- **Banco de Dados MongoDB:** Utilizado para armazenar dados não relacionais dos microsserviços Saving Goals Service (saving-goals-collection), Investments Service (investments-collection) e Gamification Service (gamification-collection). Tal escolha foi feita pois o MongoDB é um banco de dados NoSQL orientado a documentos, que oferece alta escalabilidade e flexibilidade para lidar com dados não estruturados ou em constante mudança, como são os dados tratados por esses microsserviços.
- **Banco de Dados MySQL:** Usado para armazenar dados relacionais dos microsserviços Identity Service (identity-db) e Core Bank Service (core-bank-db). Tal escolha foi feita pois o MySQL, é um banco de dados relacional amplamente utilizado em aplicativos de missão crítica, fornecendo recursos de ACID (Atomicidade, Consistência, Isolamento e Durabilidade) e suporte a transações que são recursos indispensáveis para esses microsserviços.

Esses componentes juntos formam a arquitetura de microsserviços do aplicativo proposto. Essa abordagem modular permite que cada microsserviço seja desenvolvido e implantado independentemente, promovendo a escalabilidade e a flexibilidade do sistema.

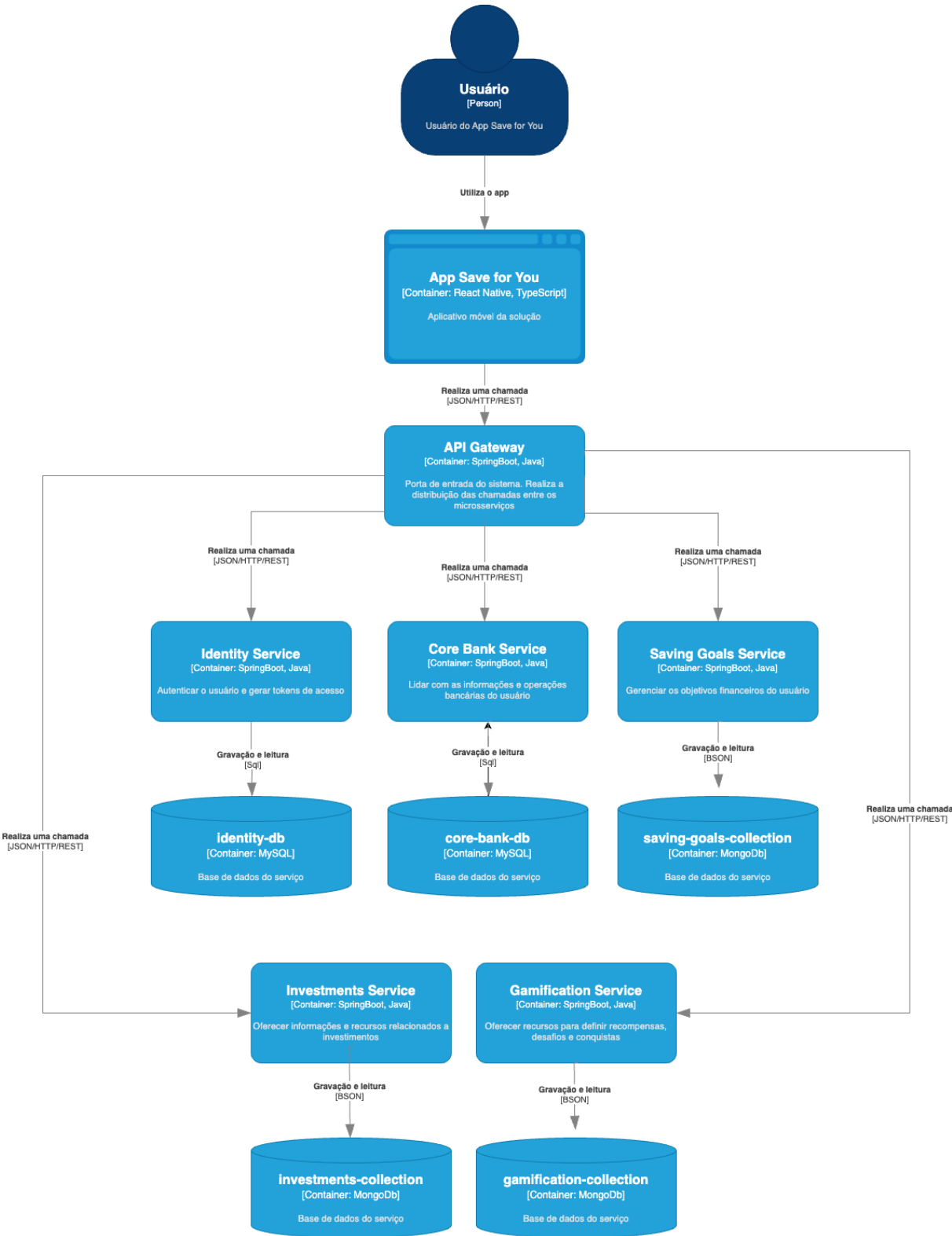


Figura 7 - Visão Geral do Contexto do Sistema

7. *Frameworks de Trabalho*

Dentre os principais frameworks e tecnologias escolhidos para o processo de implementação e implantação da solução encontram-se:

- **Spring Boot**: framework para desenvolvimento do back-end em Java.
- **Java**: linguagem de programação utilizada no desenvolvimento do back-end.
- **Maven**: ferramenta de automação de construção e gerenciamento de dependências do back-end em Java.
- **Swagger**: utilizado para documentar e visualizar as APIs do aplicativo.
- **JUnit**: framework utilizado para escrever testes automatizados no back-end.
- **IntelliJ IDEA**: IDE utilizada para desenvolvimento do código-fonte do backend.
- **React Native**: framework para desenvolvimento do front-end do aplicativo móvel.
- **TypeScript**: linguagem de programação utilizada no desenvolvimento do front-end.
- **Expo**: ferramenta que permite o fácil acesso às APIs nativas do dispositivo móvel sem precisar instalar qualquer dependência ou alterar código nativo e ajudando construir, rapidamente, aplicativos iOS e Android a partir da mesma base de código JavaScript/TypeScript.
- **Tamagui**: biblioteca de interface do usuário para React Native que fornece uma ampla variedade de componentes personalizáveis e já prontos para uso.
- **Axios**: biblioteca utilizada para integração do projeto React Native com as APIs do back-end.
- **Node.js**: ambiente de execução do JavaScript utilizado no desenvolvimento do front-end.
- **npm**: gerenciador de pacotes do Node.js utilizado para instalar e gerenciar as dependências do projeto.
- **Jest**: framework utilizado para escrever testes automatizados no front-end.
- **Visual Studio Code**: IDE utilizada para desenvolvimento do código-fonte do front-end.
- **MongoDB**: banco de dados NoSQL utilizado para persistência de dados.

Save for you

- **MySQL**: banco de dados relacional utilizado para persistência de dados.
- **Github**: sistema de controle de versão utilizado para gerenciar o código-fonte do projeto.
- **Docker**: plataforma utilizada para facilitar o empacotamento e a implantação dos microsserviços em contêineres.
- **Docker Hub**: plataforma utilizada como repositório das imagens containerizadas dos microsserviços.
- **Postman**: utilizado para testar as APIs e realizar requisições HTTP.
- **AWS**: serviços da Amazon Web Services utilizados para hospedar a infraestrutura do aplicativo.

8. *Estrutura Base do Front End*

Nas figuras seguintes é possível visualizar o layout mestre do sistema bem como o menu de opções disponíveis. Cada uma delas também podem ser acessadas no repositório do projeto por meio do link:

- <https://github.com/larissalana/save-for-you-docs/tree/main/docs/Etapa%2002/Layout%20Mestre>



Figura 8 - Layout Mestre - Tela Inicial

Save for you

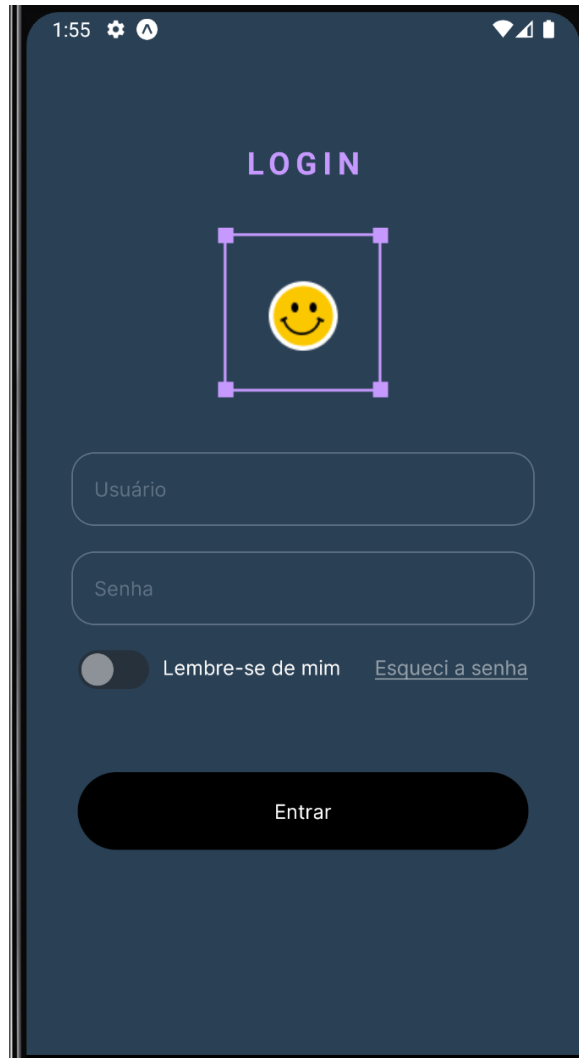
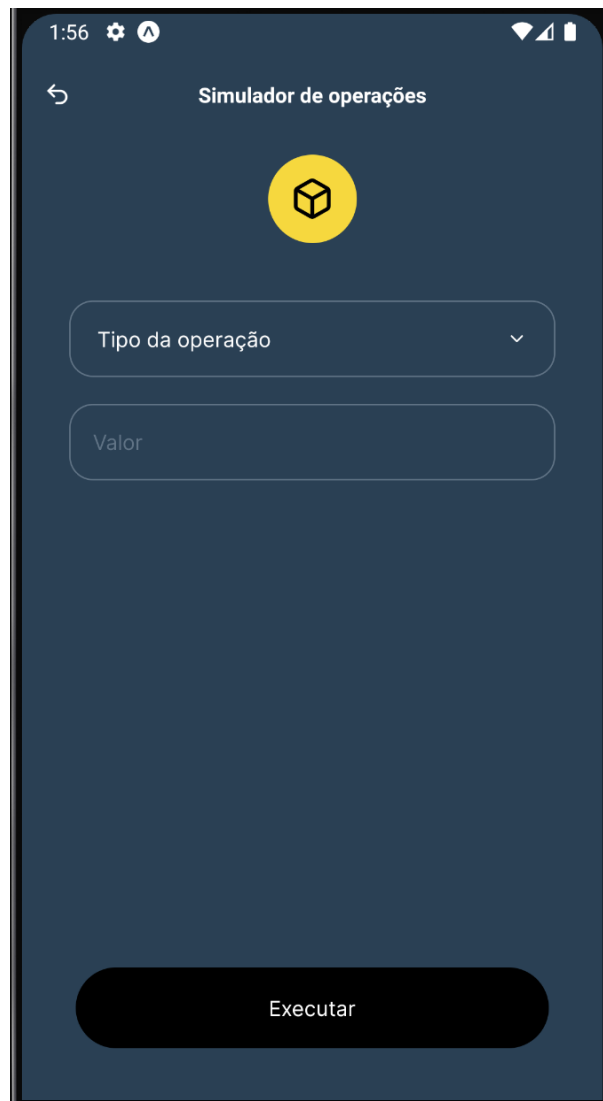


Figura 9 - Layout Mestre - Tela de Login



Figura 10 - Layout Mestre - Tela Principal

Save for you



The image shows a mobile application interface titled "Simulador de operações". At the top, there is a status bar with the time "1:56", a settings gear icon, a notification bell icon, and standard Android connectivity icons (Wi-Fi, signal strength, battery). Below the status bar is a navigation bar with a back arrow on the left and the title "Simulador de operações" in the center. The main content area has a dark blue background. In the center, there is a yellow circular icon containing a black cube. Below this icon are two input fields: the first is a dropdown menu labeled "Tipo da operação" with a downward arrow, and the second is a text input field labeled "Valor". At the bottom of the screen, there is a large, rounded black button with the white text "Executar".

Figura 11 - Layout Mestre - Simulador de Operações



Figura 12 - Layout Mestre - Tela de Extrato

Save for you

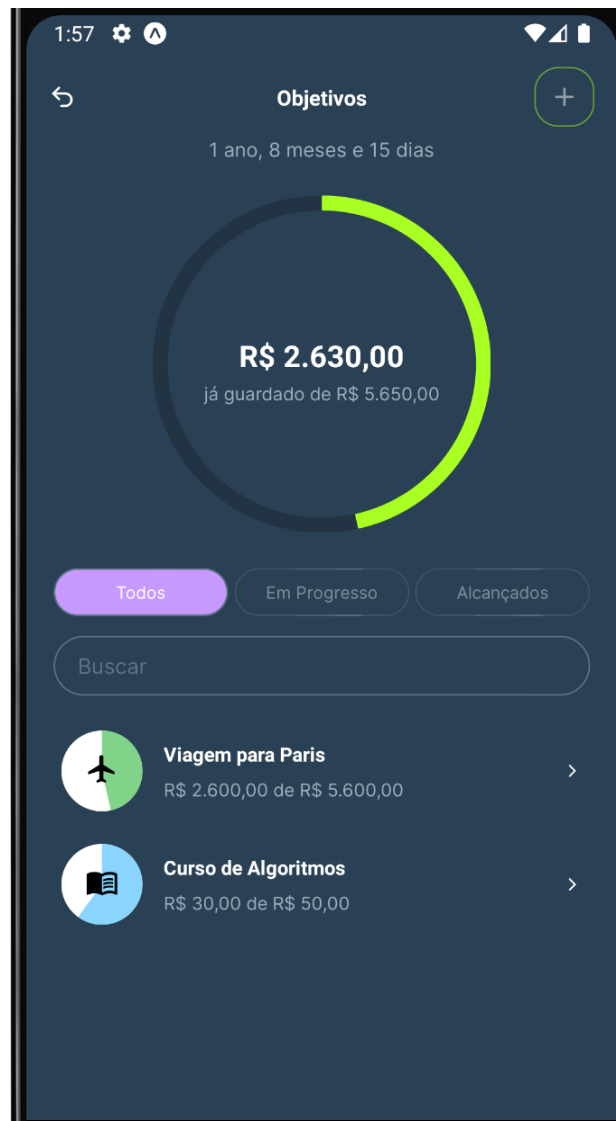


Figura 13 - Layout Mestre - Tela Lista de Objetivos Cadastrados



Figura 14 - Layout Mestre - Tela Detalhes do Objetivo



Figura 15 - Layout Mestre - Tela Lista de Regras Ativas de um Objetivo



Figura 16 - Layout Mestre - Tela Descrição detalhada de uma Regra



Figura 17 - Layout Mestre - Tela Detalhes de uma Regra Ativa de um Objetivo

9. Modelo Relacional ou Projeto de Banco de Dados NoSQL

Conforme apresentado na seção 6.2 do presente trabalho e detalhado nas figuras abaixo, a proposta é que os microsserviços Identity Service e Core Bank Service sejam desenvolvidos utilizando um banco de dados relacional, enquanto que os demais fazem uso de um banco de dados NoSQL.

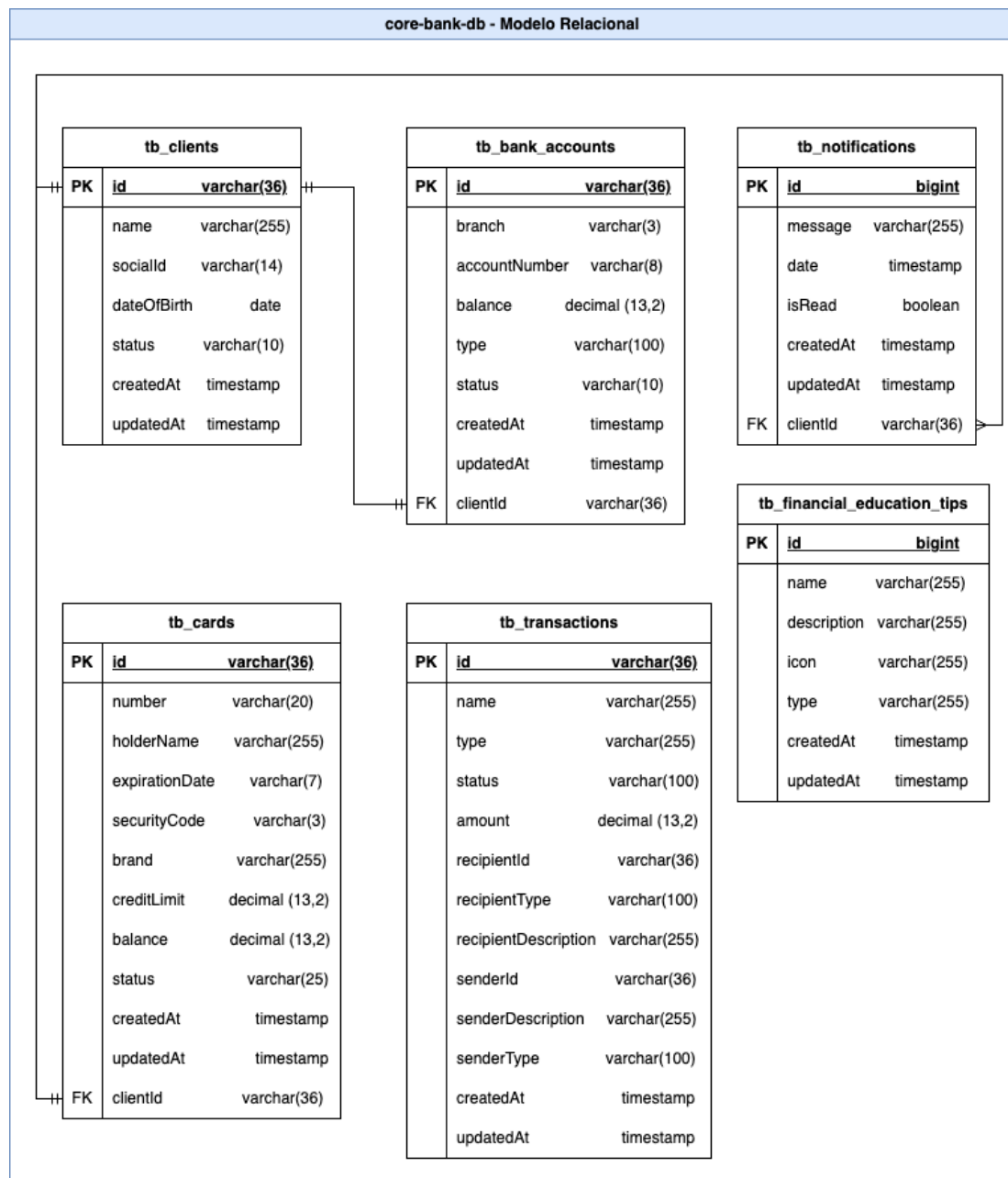


Figura 18 - Modelo Relacional - Core Bank Service

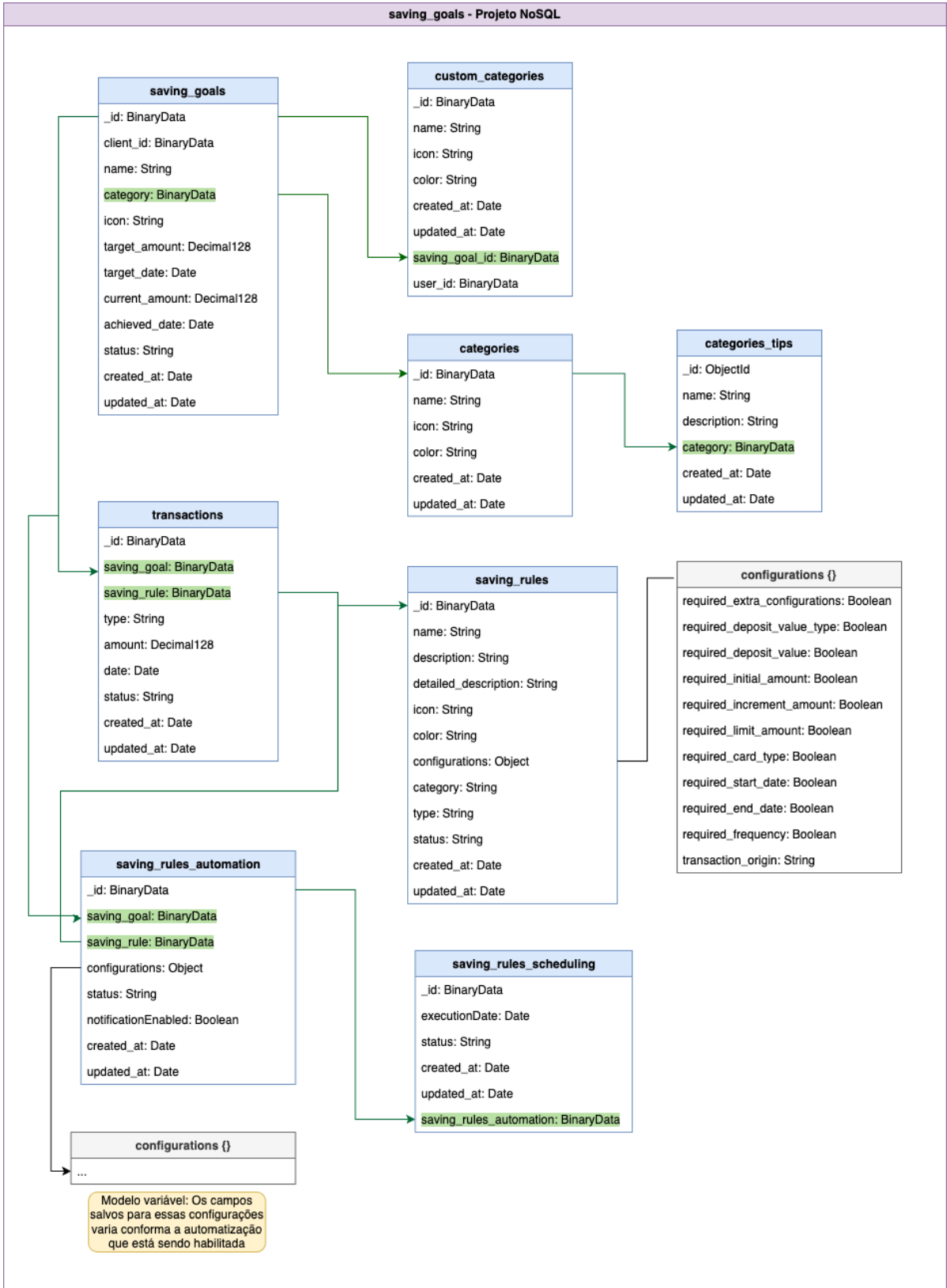


Figura 19 - Projeto de Banco de Dados NoSQL - Saving Goals Service

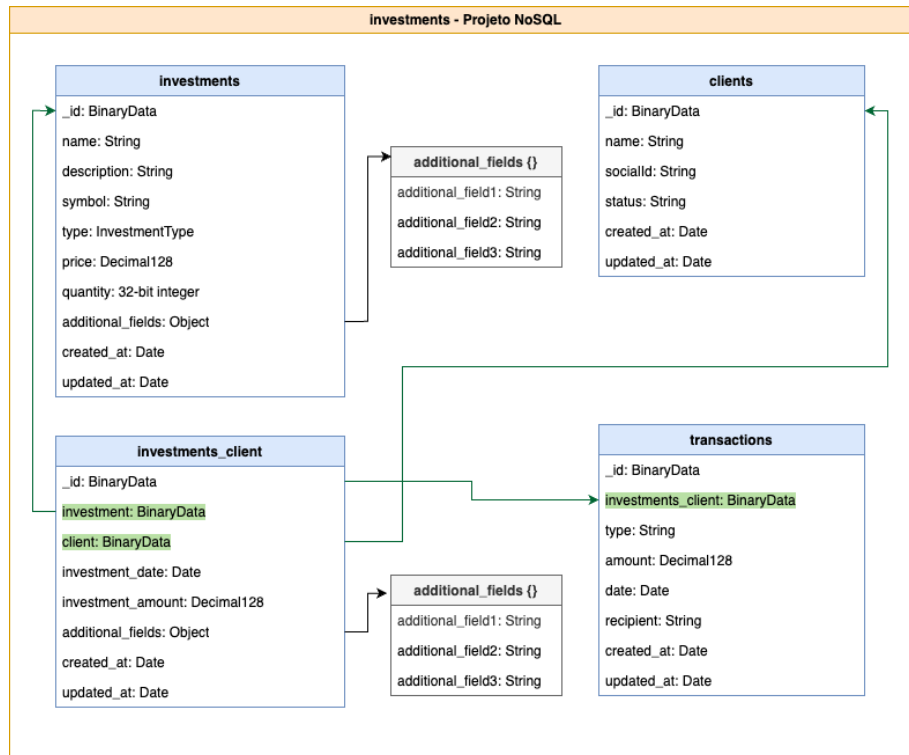


Figura 20 - Projeto de Banco de Dados NoSQL - Investments Service



Figura 21 - Projeto de Banco de Dados NoSQL - Gamification Service

Save for you

identity-db - Modelo Relacional		
tb_users		
PK	id	varchar(36)
	name	varchar(255)
	socialId	varchar(14)
	email	varchar(100)
	username	varchar(100)
	password	varchar(100)
	status	varchar(10)
	clientId	varchar(36)
	createdAt	timestamp
	updatedAt	timestamp

Figura 22 - Modelo Relacional - Identity Service

10. Plano de Testes

Nº	Caso de uso	Objetivo do caso de teste	Entradas	Resultados esperados
1	Cadastrar objetivo financeiro	Verificar se o usuário pode cadastrar um novo objetivo financeiro no aplicativo com sucesso.	- Nome: Viagem para Paris - Categoria: Férias - Valor alvo: R\$: 15.000,00 - Data alvo: 17/12/2025	O objetivo financeiro é cadastrado com sucesso e é exibido na lista de objetivos do usuário.
2	Cadastrar objetivo financeiro	Verificar se o usuário pode cadastrar um novo objetivo financeiro no aplicativo com mesmo nome de outro já cadastrado.	- Nome: Viagem para Paris - Categoria: Férias - Valor alvo: R\$: 15.000,00 - Data alvo: 17/12/2025	Uma mensagem é exibida informando que já existe um objetivo financeiro com o nome informado.
3	Cadastrar objetivo financeiro	Verificar se o usuário pode cadastrar um novo objetivo financeiro no aplicativo com o nome em branco.	- Nome: [vazio] - Categoria: Férias - Valor alvo: R\$: 15.000,00 - Data alvo: 17/12/2025	O sistema deve exibir uma mensagem de erro indicando que o nome do objetivo é obrigatório.
4	Cadastrar objetivo financeiro	Verificar se o usuário pode cadastrar um novo objetivo financeiro no aplicativo com o valor alvo negativo.	- Nome: Férias dos sonhos - Categoria: Férias - Valor alvo: R\$: -10.000,00 - Data alvo: 17/12/2025	O sistema deve exibir uma mensagem de erro indicando que o valor do objetivo deve ser maior que zero.
5	Visualizar objetivos cadastrados	Verificar se o aplicativo lista corretamente os objetivos do usuário, separando-os por status	Nenhum parâmetro de entrada é necessário para este caso de teste, apenas clique sobre cada uma das categorias exibidas.	Os objetivos são listados corretamente em suas respectivas categorias
6	Visualizar objetivos cadastrados	Verificar situações em que não existe objetivo financeiro para determinado status.	Nenhum parâmetro de entrada é necessário para este caso de teste, apenas clique sobre cada uma categoria que não possui objetivos vinculados.	O sistema deve exibir uma mensagem informativa quando não há objetivos cadastrados, em vez de exibir uma lista vazia.
7	Visualizar detalhes de um objetivo	Verificar se o sistema exibe corretamente os detalhes de um objetivo selecionado.	Nenhum parâmetro de entrada é necessário para este caso de teste, apenas seleção de um objetivo específico na lista de objetivos.	Os detalhes do objetivo estão corretos e correspondem às informações previamente cadastradas.

Nº	Caso de uso	Objetivo do caso de teste	Entradas	Resultados esperados
8	Visualizar regras de automatização de depósitos	Verificar se o sistema exibe corretamente as regras de automatização de depósitos configuradas pelo usuário.	- Usuário autenticado no sistema. - Configurações de automatização de depósitos previamente definidas pelo usuário.	O sistema deve exibir as regras de automatização de depósitos configuradas.
9	Visualizar os detalhes regra de automatização	Verificar se o sistema exibe corretamente os detalhes da regra de automatização quando solicitado pelo usuário.	Nenhum parâmetro de entrada é necessário para este caso de teste, apenas seleção de uma regra específica na lista de regras ativas.	O sistema deve exibir os detalhes da regra de automatização, incluindo todas as informações cadastradas.
10	Visualizar os detalhes regra de automatização	Verificar se o sistema não exibe a tela contendo detalhes de uma regra quando ela não está cadastrada ou configurada.	Nenhum parâmetro de entrada é necessário para este caso de teste, apenas seleção de uma regra específica na lista de regras que ainda não está habilitada.	O sistema não faz nada. Não abre nenhuma tela de detalhes, pois a regra ainda não foi habilitada.
11	Visualizar histórico de depósitos de uma regra de automatização	Verificar se o sistema exibe corretamente os depósitos de uma regra de automatização configuradas pelo usuário.	Nenhum parâmetro de entrada é necessário para este caso de teste, apenas seleção de uma regra específica na lista de regras ativas.	O sistema deve exibir os detalhes da regra de automatização, incluindo todas as informações cadastradas, bem como a lista das transações realizadas para essa regra.
12	Visualizar histórico de depósitos de uma regra de automatização	Verificar situações em que não existem depósitos para determinada regra de automatização.	Nenhum parâmetro de entrada é necessário para este caso de teste, apenas seleção de uma regra específica na lista de regras ativas.	O sistema deve exibir uma mensagem informando que não há depósitos realizados, em vez de exibir uma lista vazia.

11. Apropriação de Horas no Projeto

Histórico de apropriação de horas		
Data do registro	Atividade	Quantidade de horas
15/01/2023	1. Realizar estudos e pesquisas para definição e entendimento do contexto do projeto e seu público-alvo	8
16/01/2023	2. Relatório Técnico: preencher seção “1. Cronograma de Trabalho”	2
22/01/2023	3. Identificar os objetivos do projeto, definir o escopo e realizar o levantamento dos requisitos	4
01/02/2023	4. Prototipar interface, em papel, para validar e testar o escopo definido	8
05/02/2023	5. Ajustar escopo do projeto	4
10/02/2023	6. Relatório Técnico: preencher seção “2. Introdução”	6
15/02/2023	7. Criar Diagrama de Casos de Uso	2
28/02/2023	8. Relatório Técnico: preencher seção “3. Definição Conceitual da Solução”	4
18/03/2023	9. Criar o protótipo da interface do sistema no Figma	30
19/03/2023	10. Avaliar o protótipo criado tomando por base as Heurísticas de Nielsen	2
25/03/2023	11. Ajustar protótipo	4
01/04/2023	12. Criar Diagrama de Classes de domínio	4
02/04/2023	13. Relatório Técnico: preencher seção “5. Diagrama de Classes do Domínio”	0,5
02/04/2023	14. Gravar e editar o vídeo de apresentação do protótipo navegável	2
08/04/2023	15. Relatório Técnico: preencher seção “4. Protótipo Navegável do Sistema” e realizar revisão textual	2
08/04/2023	16. Realizar entrega da Etapa 1	0,25
15/04/2023	17. Realizar estudos e pesquisas para definição da arquitetura, dos frameworks, linguagens e demais tecnologias a serem utilizadas para desenvolvimento da solução	4
18/04/2023	18. Definir a arquitetura da solução, os frameworks a serem empregados, linguagens e demais tecnologias envolvidas	4
22/04/2023	19. Criar o Diagrama C4 model	2
25/04/2023	20. Relatório Técnico: preencher as seções “6. Arquitetura da Solução” e “7. Frameworks de Trabalho”	3
28/04/2023	21. Criar Diagrama do Modelo Relacional	4
30/04/2023	22. Criar banco de dados e script dos schemas e tabelas	3
06/05/2023	23. Criar Diagrama do Projeto de Banco de Dados NoSQL	2
07/05/2023	24. Relatório Técnico: preencher seção “9. Modelo Relacional e Projeto de Banco de Dados NoSQL”	0,5
13/05/2023	25. Relatório Técnico: preencher seção “10. Planos de Teste”	2

Histórico de apropriação de horas		
Data do registro	Atividade	Quantidade de horas
31/05/2023	26. Desenvolver o Back End da aplicação	40
11/06/2023	27. Desenvolver o Front End da aplicação	30
12/06/2023	28. Relatório Técnico: preencher seção “8. Estrutura Base do Front End”	1
17/06/2023	29. Realizar o deploy da aplicação	8
17/06/2023	30. Testar aplicação e realizar correções	8
17/06/2023	31. Disponibilizar o código da aplicação e demais artefatos em repositório público	2
20/06/2023	32. Gravar e editar vídeo de apresentação do projeto	4
23/06/2023	33. Relatório Técnico: preencher seções restantes do relatório técnico, adequar referências às normas ABNT e atualizar diagramas	4
24/06/2023	34. Relatório Técnico: realizar revisão textual	2
25/06/2023	35. Realizar entrega da Etapa 2	0,25

12. Código da Aplicação

O código da aplicação, back-end e front-end, encontram-se disponíveis, respectivamente, nos seguintes links:

- <https://github.com/larissalana/save-for-you-back-end>
- <https://github.com/larissalana/save-for-you-front-end>

Imagens do back-end da aplicação também podem ser obtidas pelo Docker Hub em:

- <https://hub.docker.com/repositories/larissalana01>

12.1 Instruções para acesso a aplicação

Para acessar a aplicação será necessário fazer inicialmente o download do aplicativo cliente móvel chamado Expo nas lojas oficiais de aplicativos:

- Play Store:
 - https://play.google.com/store/apps/details?id=host.exp.exponent&hl=pt_BR&gl=US&pli=1
- App Store:
 - <https://apps.apple.com/br/app/expo-go/id982107779>

Após a instalação do aplicativo no celular deverá ser seguido, ainda, uma das instruções abaixo conforme sistema operacional do dispositivo.

12.1.1 Instruções para Android

Para celulares Android, basta realizar a leitura do QRCode disponibilizado no link abaixo pela câmera do celular:

- <https://expo.dev/@saveforyou2023/save-for-you-app?serviceType=classic&distribution=expo-go>

Feita a leitura, o aplicativo Expo será aberto e uma versão do aplicativo Save for You será automaticamente carregada e estará disponível para uso.

12.1.2 Instruções para iOS

Para celulares iOS, devido a uma política de segurança da Apple, é necessário fazer login com um usuário com permissão de acesso, antes de fazer a leitura do QRCode.

Para isso, abra o aplicativo Expo após baixado, clique no canto superior direito para fazer login e introduza as seguintes entradas:

- Usuário: saveforyouuser
- Senha: 123456@.

Após o login com sucesso, prossiga com a leitura do QRCode disponibilizado no link abaixo pela câmera do celular:

- <https://expo.dev/@saveforyou2023/save-for-you-app?serviceType=classic&distribution=expo-go>

Feita a leitura, clique em "Abrir com Expo Go". O aplicativo Expo será aberto e uma versão do aplicativo Save for You será automaticamente carregada. Após conclusão do carregamento, o aplicativo estará disponível para uso.

12.2 Credenciais de acesso

As credenciais de acesso da aplicação "Save for You" podem ser obtidas por meio da tabela abaixo, estando divididas conforme seu propósito:

Credenciais de Acesso			
Nº	Usuário	Senha	Descrição e propósito
1	marie.curie	654321	Para esse usuário, nenhum objetivo financeiro foi cadastrado e o mesmo é utilizado para testar se ao realizar alguma operação financeira (depósito, depósito de salário, cobrança no crédito ou no débito), apenas o saldo em conta sofre alguma alteração e se somente um registro de transação é gerado conforme a operação realizada.
2	ada.lovelace	123456	Para esse usuário foi cadastrado dois objetivos financeiros: "Viagem para Paris" e "Curso de Algoritmos". Cada um dos objetivos possui uma regra de automatização vinculada.

Credenciais de Acesso			
Nº	Usuário	Senha	Descrição e propósito
			<ul style="list-style-type: none"> • "Viagem para Paris": está vinculado a regra "Depósito de Salário", de forma que toda vez que for realizado um depósito de salário de R\$250,00 ou mais, uma transferência automatizada desse valor será realizada para esse objetivo. • "Curso de Algoritmos": está vinculado a regra "Depósito por arredondamento", de forma que toda vez que uma cobrança de cartão de crédito ou débito for realizada, uma nova cobrança automatizada também deverá ser feita, e o valor dessa nova cobrança transferido para o objetivo.

12.3 Vídeo de apresentação

O vídeo de apresentação do projeto pode ser obtido por meio do download pelo repositório de documentação do projeto no GitHub:

- <https://github.com/larissalana/save-for-you-docs/blob/main/docs/Etapa%202/Apresentac%CC%A7a%CC%83o%20do%20projeto.mp4>

13. Avaliação Retrospectiva

13.1 Objetivos Estimados

- Realizar estudos e pesquisas para definir e entender o contexto do projeto, bem como identificar as necessidades e anseios do público-alvo;
- Descrever os requisitos do aplicativo e os demais artefatos necessários para o seu desenvolvimento;
- Desenvolver o aplicativo com uma interface amigável e intuitiva, facilitando seu uso por pessoas com diferentes níveis de conhecimento financeiro ou tecnológico.

13.2 Objetivos Alcançados

- Foram realizados estudos e pesquisas para compreender o contexto do projeto e as necessidades do público-alvo, garantindo uma base sólida para o desenvolvimento do aplicativo.
- Os requisitos do aplicativo foram definidos e documentados, fornecendo uma orientação clara para o desenvolvimento.
- O aplicativo foi desenvolvido com uma interface colorida visando ser amigável e intuitivo. Entretanto, observa-se que dada as cores escolhidas, tal objetivo foi alcançado apenas parcialmente, pois o público alvo ficou bastante restrito aos jovens e jovens adultos. Dessa forma, espera-se que ajustes sejam feitos na interface para que seja também útil e agradável visualmente a outros diferentes perfis de usuários.

13.3 Lições aprendidas

	Retrospectiva (Lições Aprendidas)	
	Descrição da Lição	Classificação
1	Aplicação prática dos conceitos aprendidos durante o curso.	Positiva
2	A importância de se ter um escopo bem definido antes de tomar qualquer decisão arquitetural ou tecnológica.	Positiva

	Retrospectiva (Lições Aprendidas)	
	Descrição da Lição	Classificação
3	A percepção prática de que é essencial definir claramente os limites dos microsserviços e garantir uma comunicação eficiente entre eles.	Positiva
4	A percepção de que é essencial definir uma documentação clara e consistente para facilitar a compreensão, desenvolvimento e o uso de APIs.	Positiva

14. Referências

SERVIÇO DE PROTEÇÃO AO CRÉDITO. **Apenas 18% dos brasileiros conseguiram poupar em junho, revela indicador da CNDL/SPC Brasil**. São Paulo: SPC Brasil, 2018. Disponível em: <<https://www.spcbrasil.org.br/pesquisas/indice/5136>>. Acesso em: 10 fev. 2023.

Fundo Monetário Internacional. **A crise da COVID-19 é uma ameaça para a estabilidade financeira**. Blog do FMI, 14 abr. 2020. Disponível em: <<https://www.imf.org/pt/Blogs/Articles/2020/04/14/blog-gfsr-covid-19-crisis-poses-threat-to-financial-stability>>. Acesso em: 09 fev. 2023.

VIRGINIA CREDIT UNION. **10 Steps to Financial Success**. Disponível em: <<https://www.vacu.org/learn/financial-management/managing-your-money/10-steps-to-financial-success>>. Acesso em: 09 fev. 2023.

M1 FINANCE. **What is financial success?** M1 Finance. Disponível em: <<https://m1finance.com/articles-1/financial-success/>>. Acesso em: 10 fev. 2023.