



Projeto 16

Cloud360

Documentação Técnica

MARÇO/2024

INTEGRANTES DO GRUPO

Nome Completo	Linkedin
Antonio Herisson Silva Morais	https://www.linkedin.com/in/herisson-silva-7275a0187/
Antonio Luiz Scarpe	https://www.linkedin.com/in/antonioluizscarpe/
Carlos Reynaldo Fernandes Junior	https://www.linkedin.com/in/junior-fernandes-35006228/
Carlos Roberto Borzi	https://www.linkedin.com/in/carlosborzi/
Filipi Meirelles	https://www.linkedin.com/in/filipimeirelles
Haroldo Abreu	https://www.linkedin.com/in/haroldoabreudevops/
Henrique de Lacerda Barros	https://www.linkedin.com/in/henrique-lacerda-barros-3b19b7288/
Joel Albino de Oliveira	https://www.linkedin.com/in/joel-oliveira-b6a91226/
José Américo Rodrigues	https://www.linkedin.com/in/joseamericorodrigues/
Juan Versolato Lopes	https://www.linkedin.com/in/juan-versolato-lopes-10b238159/
Loan Alencar Schlemmer	https://www.linkedin.com/in/loanschlemmer/
Maicon Jose Duarte	https://www.linkedin.com/in/maiconjoseduarte/
Marcos Cabisieri	https://www.linkedin.com/in/marcos-cabisieri/
Marcos Galindo da Costa	https://www.linkedin.com/in/marcos-galindo-da-costa-b5a90128/
Rafael Gomes dos Santos	
Rafael Machado	https://www.linkedin.com/in/rafaelmachado-260384/
Renato José da Silva	https://www.linkedin.com/in/renato-silva-cloud-aws-tecnologia
Roberto Cosmo Sobrinho	https://www.linkedin.com/in/robertocosmo/
Walter Gargarela Junior	https://www.linkedin.com/in/waltergargarela/

SUMÁRIO

1. INTRODUÇÃO	3
1.1. Quem Somos	3
1.2. Objetivo Deste Documento	4
1.3. Resumo Executivo do Problema	4
2. CLÁUSULA DE CONFIDENCIALIDADE	5
3. PROPOSTA TÉCNICA	6
3.1. Solução Proposta	6
3.2. Serviços Utilizados	7
3.2.1. Termos, nome, endereços e atribuições	15
3.3. Migração do Banco de Dados	18
3.3.1. Motivos Para Migração	18
3.3.2. Processo de Migração	19
3.3.2.1. Configuração do AWS DMS	19
3.3.2.2. Configuração da Tarefa de Migração	20
3.3.2.3. Execução da Tarefa de Migração, Validação e Testes	20
3.3.3. Considerações Adicionais Sobre a Migração	21
3.4. Automatização da Aplicação	21
4. TESTES DE CARGA	24
5. CRONOGRAMA DO PROJETO	27

1. INTRODUÇÃO

1.1. Quem Somos

Na Cloud360, somos apaixonados por desbravar os limites da tecnologia e impulsionar a inovação na era da computação em nuvem. Nossa equipe é composta por especialistas em nuvem altamente qualificados e motivados, dedicados a fornecer soluções de ponta.

Missão

Oferecer serviços e soluções excepcionais, alinhadas com as melhores práticas da indústria, para impulsionar a transformação digital e promover a agilidade, inovação e eficiência operacional de nossos clientes.

Valores

1. **Excelência Técnica:** Buscamos constantemente a excelência em nossos serviços, investindo em conhecimento e habilidades para oferecer soluções de alta qualidade e desempenho;
2. **Parceria Colaborativa:** Valorizamos a colaboração e o trabalho em equipe, tanto internamente quanto com nossos clientes, parceiros e comunidade, para alcançar resultados excepcionais juntos;
3. **Inovação Contínua:** Estamos comprometidos com a inovação constante, buscando novas maneiras de melhorar e otimizar nossos processos, serviços e soluções para atender às necessidades em constante evolução de nossos clientes;
4. **Integridade e Transparência:** Operamos com integridade e transparência em todas as nossas interações, mantendo altos padrões éticos e sendo honestos e confiáveis em tudo o que fazemos.

1.2. Objetivo Deste Documento

Este documento tem como objetivo apresentar uma solução para resolver os desafios enfrentados pela Fintech, oferecendo uma abordagem baseada em nuvem. Com computação elástica, segura e escalável, esta proposta visa atender às demandas apresentadas pela Fintech. Ao adotar essa solução em nuvem, a Fintech pode garantir uma infraestrutura flexível e adaptável às suas necessidades específicas, proporcionando um ambiente confiável e eficiente para suas operações.

1.3. Resumo Executivo do Problema

A Fintech é uma startup em crescimento que possui uma aplicação financeira em produção em pequena escala. Atualmente possuem 1000 usuários utilizando a aplicação e pretendem chegar a 200 mil usuários.

A aplicação atualmente está rodando em PHP 7.4 utilizando uma instância EC2 e em um banco de dados MySQL 5.7 no RDS, a aplicação também possui integração com o S3 para armazenamento de imagens.

A Fintech precisa realizar a migração da aplicação para que sejam atendidos principalmente os seguintes objetivos:

- *Aplicação com alta disponibilidade;*
- *Melhoria na segurança da aplicação e infraestrutura;*
- *Atualizações automáticas pois atualmente todas as atualizações são manuais.*

2. CLÁUSULA DE CONFIDENCIALIDADE

Este documento contém informações confidenciais e proprietárias da Cloud360 e destina-se apenas para uso interno. Sua divulgação não autorizada é estritamente proibida e sujeita a medidas legais. Qualquer compartilhamento externo deste documento requer prévia autorização por escrito da Cloud360.

A Cloud360 compromete-se a proteger a confidencialidade de todas as informações fornecidas pela Fintech para a elaboração desta proposta. Nenhuma informação será compartilhada ou divulgada a terceiros sem consentimento expresso.

Esta proposta técnica está oficialmente registrada sob o número PROPTEC-0603/2024 e uma cópia idêntica é mantida em nossos arquivos de conformidade. Quaisquer dúvidas ou preocupações sobre a conduta de nossa equipe devem ser encaminhadas aos nossos canais de conformidade, disponíveis em www.compliance.cloud360.com.br ou por e-mail para compliance@cloud360.com.br.

3. PROPOSTA TÉCNICA

Apresentamos uma solução tecnológica que combina eficiência, flexibilidade e vantagens financeiras, promovendo o alcance eficaz do público-alvo da Fintech. Nosso objetivo é elevar os padrões de segurança da aplicação e infraestrutura, garantir alta disponibilidade e automatizar as atualizações, proporcionando uma experiência excepcional e contínua.

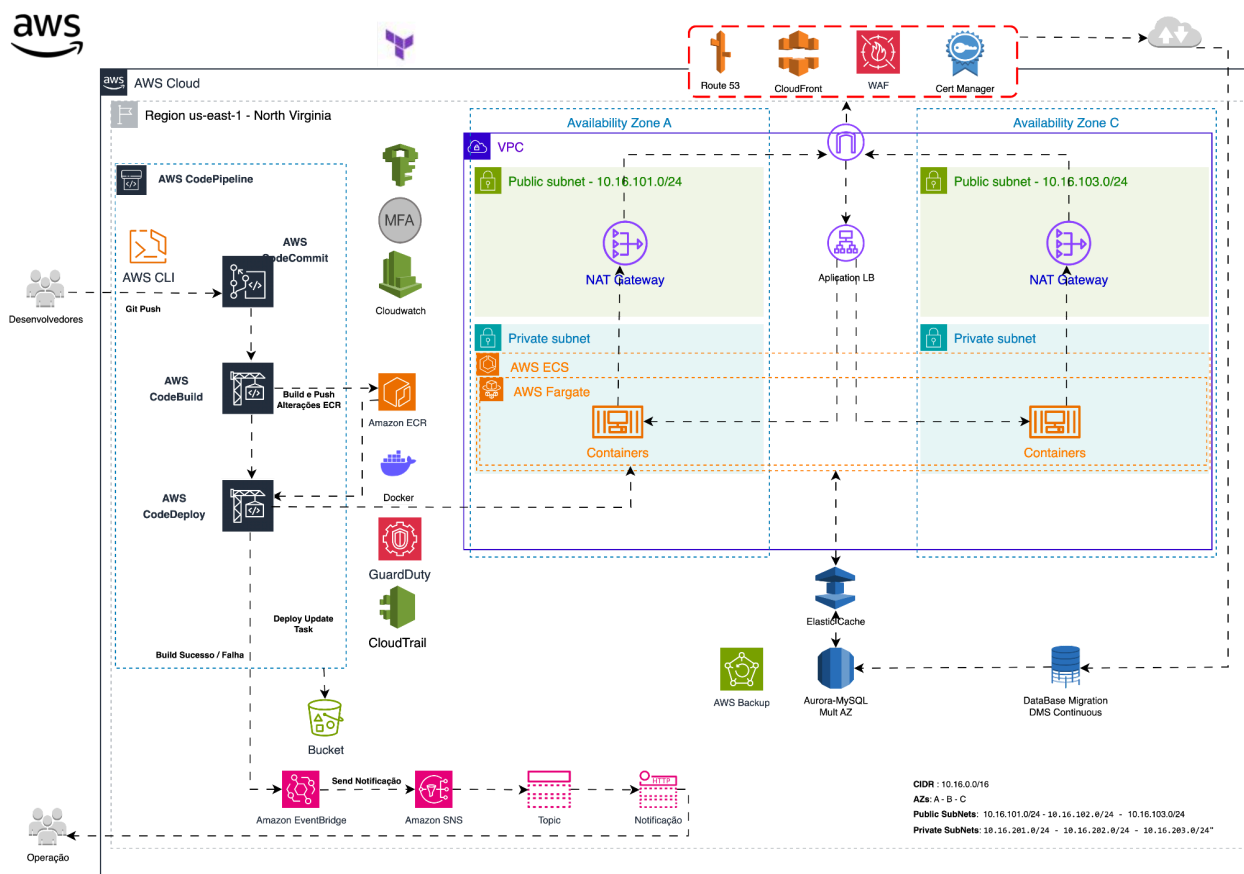
3.1. Solução Proposta

Nossa abordagem se baseia em uma arquitetura na nuvem, estruturada na AWS através do Terraform, integrando recursos poderosos para garantir um desempenho excepcional, segurança robusta e gerenciamento simplificado.

Para maximizar a flexibilidade e escalabilidade da solução, optamos por executar a aplicação em containers Docker. Essa escolha estratégica permite uma implantação ágil e eficaz do sistema, além de facilitar a manutenção e aprimoramento contínuos. Com o Docker, garantimos não apenas uma operação fluida, mas também a capacidade de realizar atualizações rápidas e seguras, mantendo a infraestrutura sempre atualizada e adaptável às demandas do ambiente.

Abaixo, destacamos os elementos fundamentais dessa estrutura:

Figura 1: Diagrama da arquitetura na AWS proposta.



3.2. Serviços Utilizados

Os serviços que serão empregados na solução proposta são os seguintes:

- **AWS CLI:** A AWS Command Line Interface (AWS CLI) é uma ferramenta unificada para o gerenciamento de seus produtos da AWS. Com apenas uma ferramenta para baixar e configurar, você poderá controlar vários produtos da AWS pela linha de comando e automatizá-los usando scripts.
- **Amazon VPC (VPC/subnets):** Amazon VPC (Virtual Private Cloud) é uma solução que permite um controle total sobre o ambiente de produção e suas

sub-redes (subnets) na nuvem. Com a VPC, é possível posicionar recursos, estabelecer conectividade e garantir a segurança de forma personalizada e eficiente.

- **Amazon VPC Peering:** É uma forma de conectar duas VPC's na AWS de forma segura e privada. Esse recurso permite que as instâncias em ambas as VPC's se comuniquem entre si como se estivessem na mesma rede local, mesmo que estejam em regiões AWS diferentes.
- **NAT Gateway:** O NAT Gateway da AWS é um serviço que permite que instâncias em redes privadas acessem a Internet com segurança, atuando como intermediário e protegendo contra conexões não solicitadas. É essencial para garantir a conectividade e segurança de instâncias em redes privadas na AWS.
- **Security Group:** O Security Group atua como um firewall virtual para as instâncias da AWS, como contêineres ECS, banco de dados RDS e outros recursos, controlando o tráfego de entrada e saída de dados para garantir a segurança em sua arquitetura na nuvem.
- **Network Access Control List (Network ACL):** É uma Lista de Controle de Acesso que permite controlar o tráfego de entrada e saída de dados em nível de sub-rede. Ela atua como uma camada de segurança, permitindo que sejam definidas regras personalizadas para permitir ou negar o tráfego com base em endereços IP, portas e protocolos. As Network ACL's são essenciais para proteção de sub-redes na nuvem, melhorando a segurança e o controle de acesso aos recursos.

- **AWS Identity and Access Management (IAM):** Serviço Web da AWS que controla o acesso aos recursos de forma segura, tanto de instância como de usuários.
- **AWS Secrets Manager:** É um serviço que ajuda a gerenciar, recuperar e alternar credenciais de banco de dados, chaves de API e outros segredos ao longo de seus ciclos de vida.
- **Amazon Aurora (RDS):** O Amazon Aurora é um mecanismo de banco de dados relacional gerenciado compatível com o MySQL e o PostgreSQL. Com a migração para Amazon Aurora iremos ter a performance aprimorada, pois o Amazon Aurora é projetado para oferecer uma performance até cinco vezes melhor que o MySQL padrão, devido à sua arquitetura de armazenamento otimizada e mecanismos de processamento paralelo.
- **AWS Database Migration Service (AWS DMS):** O AWS Database Migration Service (AWS DMS) é um serviço de replicação e migração gerenciado que ajuda a mover workloads analíticos e bancos de dados para a AWS rapidamente, de forma segura e com o mínimo possível de inatividade e zero perda de dados.
- **Amazon Route 53:** É um serviço de DNS gerenciado que oferece aos clientes a possibilidade de registrar nomes de domínio, direcionar o tráfego da Internet para os recursos de domínio do cliente e verificar a integridade desses recursos.

- **AWS Certificate Manager:** É um serviço que facilita a obtenção de certificados de segurança SSL para garantir a entrega segura de sites, habilitando o protocolo HTTPS.
- **Amazon CloudWatch:** É uma ferramenta essencial para criar métricas e monitorar recursos da AWS, fornecendo insights valiosos sobre o desempenho dos mesmos.
- **AWS WAF:** O AWS WAF é um firewall de aplicação web projetado para rastrear solicitações de usuários finais aos seus aplicativos. Ele oferece controle de acesso, proteção contra ameaças comuns da web e bots maliciosos, garantindo a disponibilidade e segurança dos seus recursos, ao mesmo tempo em que evita o consumo excessivo de recursos.
- **AWS CloudTrail:** É um serviço que monitora e registra a atividade da conta por toda a infraestrutura da AWS, oferecendo controle sobre o armazenamento, análise e ações de remediação.
- **AWS GuardDuty:** É um serviço de detecção de ameaças que monitora continuamente suas contas e workloads da AWS para detectar atividades mal-intencionadas e entregar descobertas de segurança detalhadas, permitindo visibilidade e correção.
- **AWS Shield:** É um serviço gerenciado de proteção contra DDoS que protege as aplicações em execução na AWS.

- **Application Load Balancer:** O Application Load Balancer opera no nível da solicitação (camada 7), direcionando o tráfego para destinos (instâncias EC2, contêineres, endereços IP e funções Lambda) com base no conteúdo da solicitação. Ideal para balanceamento de carga avançado de tráfego HTTP e HTTPS, o Application Load Balancer fornece roteamento avançado de solicitações, sendo especialmente adequado para arquiteturas de aplicativos modernos, incluindo microsserviços e aplicativos baseados em contêineres.
- **Amazon Simple Notification Service (SNS):** O Amazon Simple Notification Service (SNS) é um serviço de mensagens e notificações totalmente gerenciado pela AWS. Ele permite que você envie mensagens, notificações e alertas para uma ampla gama de destinos, como aplicativos móveis, e-mail, SMS, HTTP e muito mais. O SNS facilita a comunicação eficiente e escalável com seus clientes, usuários e sistemas, sendo uma ferramenta essencial para notificações em tempo real e integração de aplicativos na nuvem.
- **AWS CloudFront:** O AWS CloudFront é um serviço de entrega de conteúdo (CDN) da Amazon Web Services que distribui eficientemente imagens, vídeos, arquivos e páginas da web aos usuários finais. Ele aprimora a entrega, reduzindo latência e aumentando a disponibilidade, melhorando o desempenho e a confiabilidade na entrega de recursos na web.
- **AWS S3:** O AWS S3 (Simple Storage Service) é um serviço de armazenamento em nuvem da Amazon Web Services que fornece uma plataforma escalável, durável e altamente disponível para armazenar eficientemente uma variedade de dados, como imagens, vídeos, backups e conteúdo digital.

- **AWS Backup:** O AWS Backup é um serviço da AWS que simplifica a cópia de segurança e a recuperação de dados na nuvem. Ele automatiza a proteção e retenção de informações críticas, abrangendo recursos como instâncias EC2 e bancos de dados RDS, tornando a gestão de dados na AWS mais eficiente e confiável.
- **Auto Scaling:** É um serviço que permite o crescimento da aplicação horizontalmente, em questão de minutos ocorre a escalabilidade automática dos recursos da AWS que fazem parte da aplicação.
- **AWS Fargate:** O AWS Fargate é compatível com o Amazon Elastic Container Service (Amazon ECS) e com o Amazon Elastic Kubernetes Service (Amazon EKS). O AWS Fargate é um mecanismo de computação com tecnologia sem servidor que permite que você se concentre no desenvolvimento de aplicações sem a necessidade de gerenciar servidores.
- **Elastic Container Service (ECS):** É um serviço de orquestração de contêineres totalmente gerenciado que simplifica a implantação, o gerenciamento e a escalabilidade de aplicativos containerizados. Basta descrever sua aplicação e os recursos necessários, e o Amazon ECS irá executar, monitorar e dimensionar a aplicação em opções flexíveis de computação com integrações automáticas com outros serviços de suporte da AWS que sua aplicação precisa.
- **Elastic Container Registry (ECR):** É um serviço que oferece um registro de contêineres gerenciado para armazenar, gerenciar e implantar imagens de contêineres com facilidade. Ele permite que você armazene, compartilhe e integre imagens de contêineres de maneira segura, acelerando o

desenvolvimento e implantação de aplicativos baseados em contêineres na plataforma AWS.

- **Amazon Elastic Cache:** É um serviço totalmente gerenciado, compatível com Redis e Memcache, que oferece desempenho econômico e em tempo real para aplicações modernas. Ele simplifica a implementação e a administração de caches na nuvem, melhorando o desempenho e a escalabilidade de aplicativos ao reduzir a carga nos bancos de dados e, ao mesmo tempo, oferecendo recursos avançados de segurança e monitoramento.
- **AWS Code Build:** É um serviço de integração contínua totalmente gerenciado que compila código-fonte, executa testes e produz pacotes de software prontos para implantação.
- **AWS Code Pipeline:** É um serviço totalmente gerenciado de entrega contínua que ajuda a automatizar pipelines de lançamento para oferecer atualizações rápidas e confiáveis de aplicações e infraestruturas.
- **AWS Code Commit:** É um serviço de controle de código-fonte totalmente gerenciado, seguro e altamente escalável que hospeda repositórios privados do Git.
- **Amazon EventBridge:** É um serviço de integração de eventos e roteamento da AWS (Amazon Web Services) que simplifica a construção de arquiteturas de microsserviços, aplicações serverless e pipelines de dados. Ele permite que você colete e processe dados de diferentes fontes, como aplicativos,

serviços da AWS, sistemas SaaS (Software as a Service) e sistemas locais, e os encaminhe para destinos específicos, como funções Lambda, filas SQS (Simple Queue Service) ou tópicos SNS (Simple Notification Service), de forma confiável e escalável. Com o EventBridge, você pode criar regras para filtrar eventos específicos e automatizar ações em resposta a esses eventos, facilitando a construção de arquiteturas reativas e altamente integradas na nuvem.

- **Terraform:** O Terraform é uma ferramenta de infraestrutura como código que permite automatizar a criação e gestão de recursos na AWS por meio de código declarativo, facilitando o provisionamento e a manutenção de infraestrutura de maneira eficiente e consistente.
- **Docker:** O Docker é uma plataforma de software que permite que você empacote, distribua e execute aplicativos em contêineres. Os contêineres são ambientes leves e isolados que incluem tudo o que um aplicativo precisa para ser executado, como código, bibliotecas e dependências.

Um plano de backup será desenvolvido seguindo rigorosamente as melhores práticas e diretrizes estabelecidas pelas normas técnicas de TI, como ITIL e os padrões da família ISO 27000.

A implementação de backups será automatizada, utilizando métricas predefinidas no serviço de backup do Amazon RDS. A equipe de suporte local ou os serviços fornecidos aos clientes serão responsáveis por monitorar continuamente e ajustar o plano de backup conforme necessário. Além disso, esses serviços irão englobar a restauração dos backups realizados na infraestrutura da AWS, assegurando a integridade dos dados. Este processo garantirá a segurança e a disponibilidade dos dados, essenciais para a continuidade dos negócios e a conformidade com os padrões de segurança.

3.2.1. Termos, nome, endereços e atribuições

A infraestrutura é organizada em três zonas de disponibilidade, conforme ilustrado no diagrama da Figura 1, na página 7.

Os elementos a seguir são apresentados para simplificar a implementação e a manutenção contínua do ambiente de nuvem em produção.

- **Recurso: VPC**

Nome: vpc_fintech

Endereço IPv4 CIDR: 10.16.0.0/16

Com o intuito de aprimorar a gestão do ambiente virtual e reforçar a segurança, foi concebida uma VPC customizada. A escolha recaiu sobre a região Norte da Virgínia (us-east-1) visando a otimização dos custos. Dentro desta VPC, foram configuradas duas sub-redes públicas, destinadas à operação do NAT Gateway, e outras duas sub-redes privadas, reservadas para a alocação dos recursos utilizados pelo ECS.

- **Subnets**

Nome: sn_fintech_public1

Endereço IPv4 CIDR: 10.16.101.0/24

Nome: sn_fintech_public2

Endereço IPv4 CIDR: 10.16.102.0/24

Nome: sn_fintech_public3

Endereço IPv4 CIDR: 10.16.103.0/24

Nome: sn_fintech_private1

Endereço IPv4 CIDR: 10.16.201.0/24

Nome: sn_fintech_private2

Endereço IPv4 CIDR: 10.16.202.0/24

Nome: sn_fintech_private3

Endereço IPv4 CIDR: 10.16.203.0/24

- **Route Tables**

Nome: rt_fintech_public

Descrição: Associada às subnets públicas.

Nome: rt_fintech_private

Descrição: Associada às subnets privadas.

- **NAT Gateway**

Nome: nat_gw_fintech

- **Network ACL's**

Nome: acl_fintech_public

Descrição: Associada às subnets públicas.

- ❖ **Inbound Roles**

Portas 80 e 443 - Libera todo tráfego HTTP e HTTPS, qualquer origem.

Porta 22 - Libera todo tráfego SSH, qualquer origem.

Portas Efêmeras 1024 a 65535 - Todo tráfego, qualquer origem.

- ❖ **Outbound Roles**

Liberado para todas as portas, para todos os protocolos, para qualquer destino.

Nome: acl_fintech_private

Descrição: Associada às subnets privadas.

❖ **Inbound Roles**

Libera todo o tráfego que vem somente das subnets.

❖ **Outbound Roles**

Tudo liberado.

- **Security Groups**

Nome: sg_alb_fintech

Descrição: Grupo de segurança com portas 80 e 443 liberadas, liberando tráfego HTTP e HTTPS para acesso web à aplicação.

Nome: sg_ecs_fintech

Descrição: Grupo de segurança para as Tasks do ECS, liberando acesso de entrada na porta HTTP(80) apenas para tráfego vindo do Load Balancer.

Nome: sg_db_fintech

Descrição: Grupo de segurança para o banco de dados RDS, liberando acesso de entrada na porta Amazon Aurora apenas para tráfego vindo das Tasks ECS.

- **Amazon Simple Notification Service (SNS)**

Nome: sns_fintech

- **Application Load Balancer**

Nome: alb_fintech

- **Route53**

Nome de domínio: fintech.net.br

- **AWS S3**

Nome do bucket: s3-fintech

Acesso: Privado

- **RDS**

Banco de dados: Amazon Aurora

Tipo de instância: db.r6g.large - 2vCPU e 16 Gb de memória, interface de rede 10 gigabit

Armazenamento: 500 Gb

Armazenamento de backup: 2 Tb

- **AWS ECS**

Tasks: Linux - 2vCPU e 4Gb de memória.

3.3. Migração do Banco de Dados

Atualmente a aplicação está rodando em um banco de dados MySQL 5.7 no RDS. Após análise e visando melhorias, durante o procedimento de migração, realizamos a migração para o Amazon Aurora no RDS compatível com MySQL.

O Amazon Aurora é um banco de dados relacional totalmente gerenciado, projetado para oferecer desempenho, escalabilidade e disponibilidade superiores em comparação com o MySQL convencional.

3.3.1. Motivos Para Migração

- **Desempenho Aprimorado:** O Amazon Aurora oferece até cinco vezes o desempenho do MySQL convencional devido à sua arquitetura distribuída e otimizações de armazenamento.

- **Alta Disponibilidade e Durabilidade:** Com a replicação automática de dados em várias zonas de disponibilidade e backups contínuos, o Amazon Aurora garante alta disponibilidade e durabilidade dos dados.
- **Escalabilidade Automática:** A capacidade de dimensionamento automático do Amazon Aurora permite lidar com cargas de trabalho crescentes sem a necessidade de intervenção manual.
- **Gerenciamento Simplificado:** O Amazon RDS oferece ferramentas de monitoramento, alarmes e gerenciamento automatizado de backups, reduzindo a carga operacional para administradores de banco de dados.

3.3.2. Processo de Migração

Para realizar a migração do banco de dados, utilizamos o serviço DMS (Database Migration Service) da AWS, o DMS é uma ferramenta que facilita a migração de bancos de dados para a AWS de forma rápida e sem interrupções. Com a utilização do DMS temos as seguintes etapas:

3.3.2.1. Configuração do AWS DMS

Para a criação do AWS DMS, é necessária a criação de alguns recursos do DMS, sendo eles:

- **Criação de um Endpoint de Origem:** Foi configurado um endpoint de origem para o banco de dados de origem, que no nosso caso é o MySQL 5.7. Durante o processo de criação, fornecemos as informações de acesso do banco de dados.
- **Criação de um Endpoint de Destino:** Foi configurado um endpoint de destino para o Amazon Aurora. Durante o processo de criação, fornecemos as informações de acesso do banco de dados.

- **Criação de uma instância de replicação:** Uma instância de replicação no AWS DMS é utilizado para replicar dados entre diferentes fontes de dados. É na instância de replicação que ocorre a transformação e a transferência dos dados entre a fonte de dados de origem e o destino.

Durante o processo de replicação, a replicação é feita para o banco de dados em uma VPC privada, para que ocorra a comunicação entre a VPC pública onde está o banco de dados de origem com a VPC privada onde está o banco de dados de destino, é feita uma VPC Peering, com ela é aberta uma conexão direta entre as duas VPC's.

3.3.2.2. Configuração da Tarefa de Migração

Após realizar a criação dos recursos necessários para a migração, foi realizada a criação de uma nova tarefa de migração. Na tarefa de migração foram especificados os endpoints de origem (MySQL 5.7) e destino (Amazon Aurora).

Durante a configuração da tarefa de migração foi definido o mapeamento de tabelas entre os banco de dados de origem e destino, incluindo a seleção das tabelas a serem migradas e definição do esquema de destino no Amazon Aurora.

Ainda durante o processo de configuração da tarefa de migração, foi definido o tipo de migração, para o nosso cenário foi selecionado o tipo **Migrar dados existentes e replicar alterações contínuas**, selecionando essa opção a migração irá ocorrer dos dados já existentes e caso ocorra adição de informações no banco de dados de origem, será replicado automaticamente para o banco de dados de destino.

3.3.2.3. Execução da Tarefa de Migração, Validação e Testes

Após todas as configurações realizadas, é iniciada a tarefa de migração no AWS DMS, durante o processo de migração é feito o monitoramento. Após a

migração concluída, verificamos os dados no Amazon Aurora para garantir que os dados foram migrados corretamente, foi feita adição de informações no banco de dados de origem para realização de testes e verificação se seriam migrados automaticamente para o banco de dados de destino.

3.3.3. Considerações Adicionais Sobre a Migração

Além das vantagens citadas anteriormente, podemos destacar também os seguintes pontos:

- **Compatibilidade com MySQL:** O Amazon Aurora é compatível com MySQL, o que simplifica o processo de migração e minimiza a necessidade de modificar o código da aplicação.
- **Custos:** Embora o Amazon Aurora possa parecer inicialmente mais caro do que o MySQL padrão, suas vantagens de desempenho, escalabilidade e gerenciamento simplificado podem resultar em custos totais mais baixos a longo prazo.

3.4. Automação da Aplicação

Atualmente ocorrem atualizações de forma manual na aplicação, e isso pode apresentar desvantagens significativas em termos de eficiência, confiabilidade e segurança em comparação com abordagens automatizadas, como a integração contínua e a entrega contínua (CI/CD).

Visando a automação da aplicação, utilizamos os serviços de CI/CD da AWS, sendo eles Code Commit, Code Build e Code Pipeline. Esses serviços trazem uma série de vantagens para a atualização da aplicação, podemos citar:

- **Integração contínua e entrega contínua (CI/CD):** Automatizar o processo de integração, compilação, teste e entrega permite atualizações rápidas e consistentes na aplicação. Isso reduz o tempo de espera entre as alterações

de código e a implantação em produção, promovendo uma entrega contínua e confiável.

- **Eficiência operacional:** Ao automatizar tarefas repetitivas, como compilação, teste e implantação, é reduzido o esforço manual necessário e minimizada a possibilidade de erros humanos. Isso libera tempo para os desenvolvedores se concentrarem em atividades de maior valor, como desenvolvimento de novos recursos.
- **Padronização e consistência:** Ao definir um pipeline de CI/CD, são estabelecidos padrões claros para o processo de desenvolvimento e implantação. Isso garante que cada alteração de código passe pelos mesmos estágios de teste e implantação, promovendo a consistência e reduzindo a chance de problemas devido a diferenças no ambiente.
- **Feedback rápido:** Com testes automatizados integrados ao pipeline, os desenvolvedores recebem feedback imediato sobre a qualidade das alterações de código. Isso permite identificar e corrigir problemas mais cedo no ciclo de desenvolvimento, economizando tempo e recursos.
- **Escalabilidade:** Os serviços da AWS, como CodeBuild e CodePipeline, são altamente escaláveis e podem lidar com cargas de trabalho variáveis e demandas de tráfego. Isso garante que o processo de CI/CD possa lidar com picos de atividade sem comprometer o desempenho ou a confiabilidade.
- **Segurança:** A integração com serviços como AWS Identity and Access Management (IAM) permite controlar de forma granular quem pode acessar e modificar seus repositórios de código e pipelines de CI/CD. Isso ajuda a garantir que apenas usuários autorizados possam modificar o código e iniciar implantações.

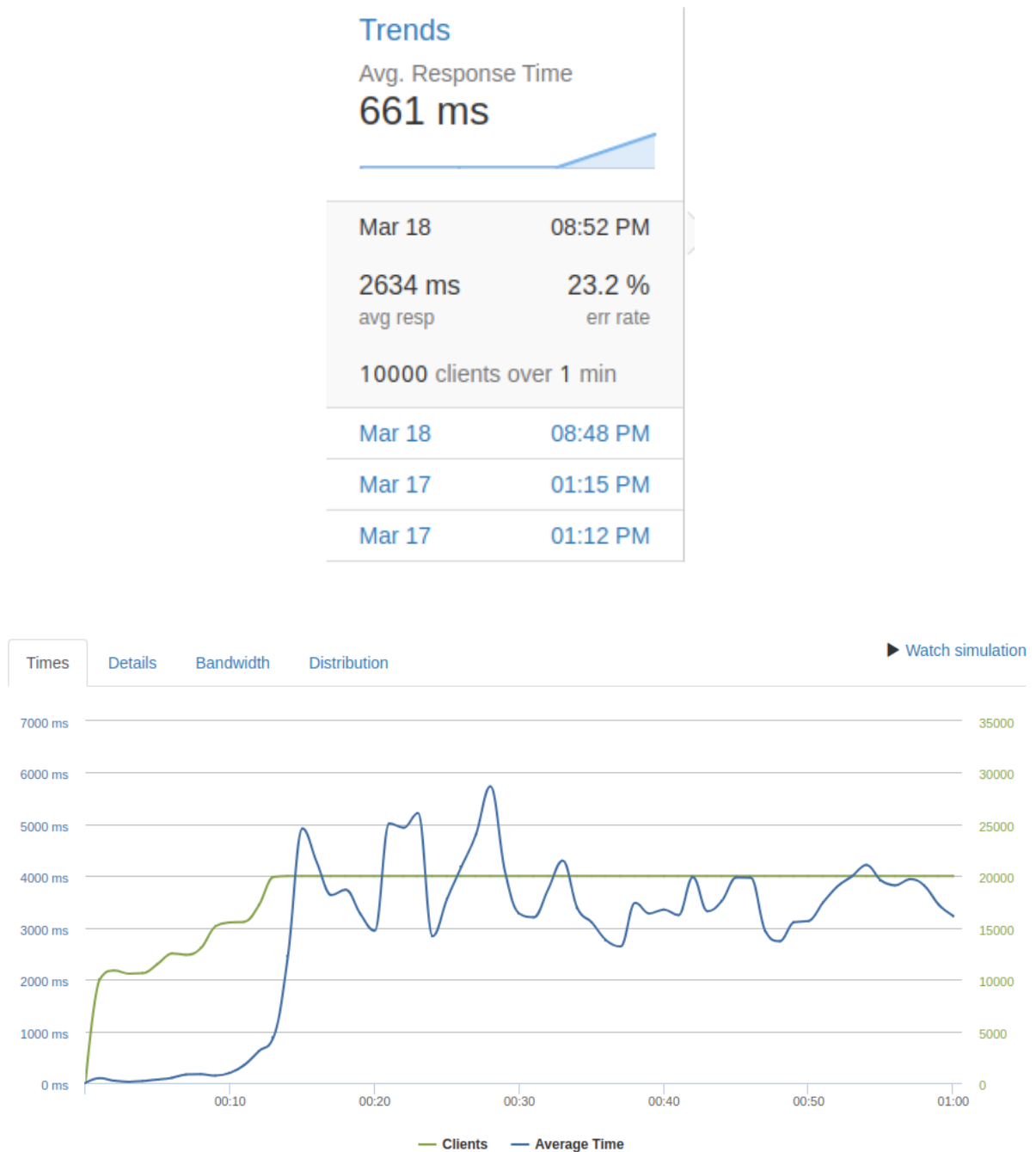
- **Flexibilidade:** Os pipelines de CI/CD podem ser configurados para se integrar com uma variedade de ferramentas e serviços de terceiros, permitindo personalizar o processo de desenvolvimento de acordo com as necessidades específicas do seu projeto.

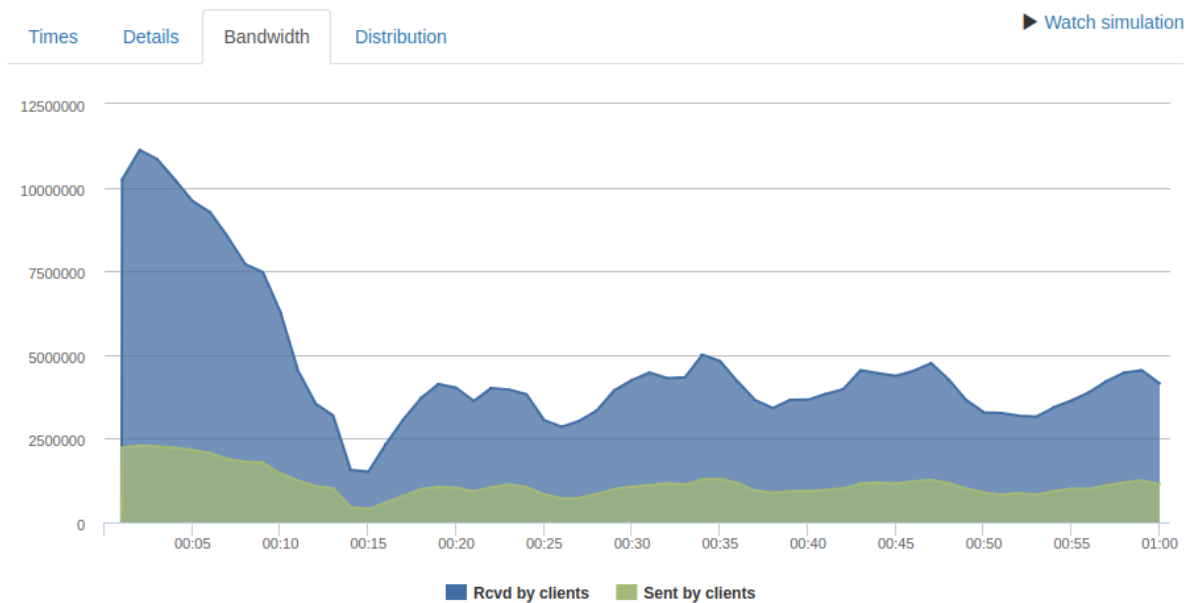
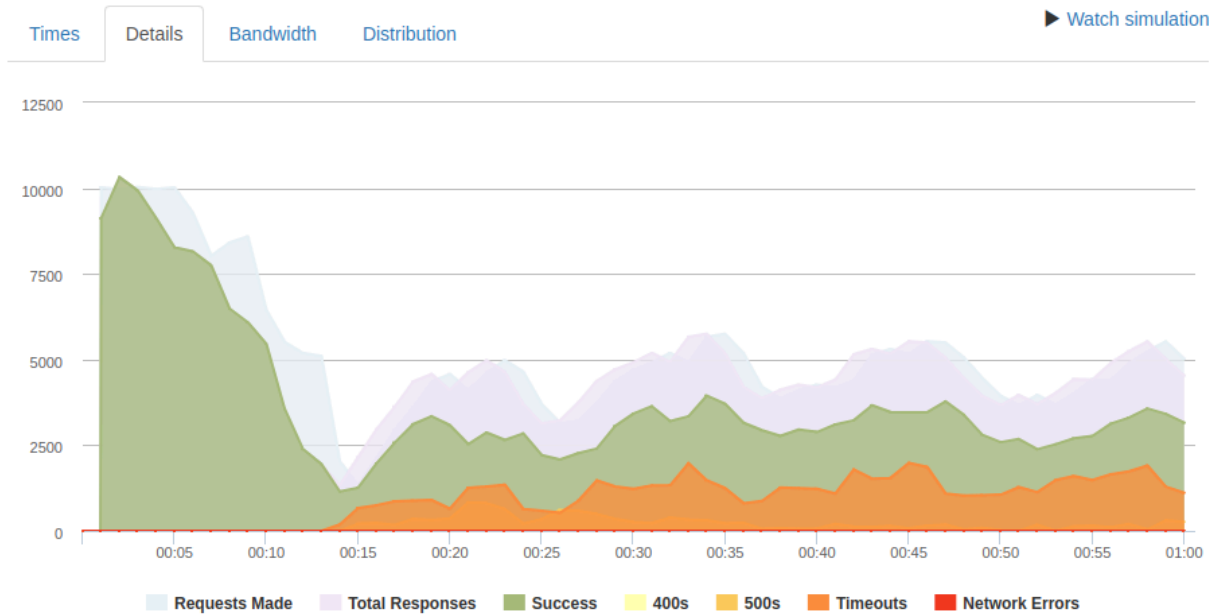
4. TESTES DE CARGA

Aplicação utilizada: Loader.io

Objetivo: Medir a capacidade do servidor da aplicação de responder rapidamente a solicitações de leitura e avaliar a latência e a taxa de transferência.

Método: 10 mil usuários simultâneos por segundo.





Response Times

Average 2634 ms
Min/Max 2 / 16992 ms

Response Counts

Success 226635 Timeout 56986
400/500 0 / 11487 Network 0

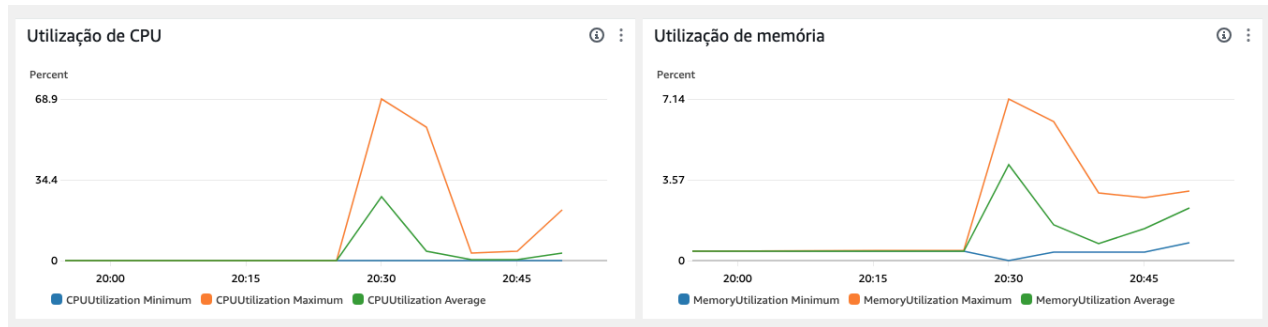
Bandwidth

Sent 66.94 MB
Received 266.97 MB

Redirects

Valid 310576
Invalid 0

Durante os testes foi monitorada a infraestrutura da AWS através do CloudWatch e tivemos os seguintes resultados:



Durante os testes de carga foi possível identificar que a infraestrutura proposta atende a demanda da Fintech.

5. CRONOGRAMA DO PROJETO

Este cronograma detalha as etapas essenciais para migrar com sucesso a aplicação financeira da empresa Fintech para uma infraestrutura em alta disponibilidade, automatizada e altamente segura na AWS. A migração incluirá a transição da aplicação e do banco de dados, visando proporcionar maior escalabilidade, segurança e eficiência operacional.



Semana 1:

- **Validação dos requisitos:** Revisão detalhada dos requisitos de migração, garantindo que todos os aspectos estejam documentados e compreendidos.
- **Implementação da infraestrutura:** Configuração inicial da infraestrutura na AWS, incluindo a criação de redes, servidores, armazenamento e demais recursos necessários para suportar a aplicação financeira.

Semana 2:

- **Migração dos dados:** Transferência dos dados da aplicação financeira para a nova infraestrutura, garantindo integridade e consistência.
- **Validação da estrutura:** Testes rigorosos para verificar se a infraestrutura está funcionando conforme o esperado, identificando e corrigindo eventuais problemas.

Semana 3:

- **Entrega da infraestrutura:** Finalização da configuração da infraestrutura e entrega oficial aos responsáveis pela aplicação financeira.
- **Acompanhamento:** Monitoramento inicial da nova infraestrutura para garantir estabilidade e disponibilidade, além de fornecer suporte e assistência conforme necessário.

Semana 4:

- **Monitoramento:** Monitoramento contínuo da infraestrutura e da aplicação financeira para identificar possíveis problemas de desempenho ou segurança.
- **Ajustes:** Realização de ajustes e otimizações com base nos dados de monitoramento, garantindo que a infraestrutura opere de forma eficiente e segura.
- **Entrega do projeto:** Documentação final do projeto de migração, incluindo lições aprendidas, recomendações e próximos passos, e entrega formal ao cliente ou à equipe interna.