



**Universidade São Judas Tadeu - Butantã - Noturno**

**Jessica Almeida Mesquita -824156980**

**Kaik Jose Rodrigues - 824159059**

**PROJETO A3**

**Turma: GQS-CCP1AN-BUE1**

## Sumário

1. Planejamento de Testes de Software.....	3
1.1. Cronograma de Atividades.....	3
1.2. Alocação de Recursos.....	3, 4
1.3. Marcos do Projeto.....	4
2. Documentos de Desenvolvimento de Software.....	4
2.1. Plano de Projeto.....	4
2.1.1. Planejamento do Projeto.....	4
2.1.2. Escopo.....	5
2.1.3. Recursos.....	5
2.1.4. Estimativas de Projeto.....	5, 6
2.2. Documento de Requisitos.....	6
2.3. Planejamento de Testes.....	7
2.3.1. Plano de Testes.....	7
2.3.1.1. Introdução.....	7
2.3.1.2. Escopo.....	7
2.3.1.3. Objetivos.....	7
2.3.1.4. Requisitos a Serem Testados.....	8
2.3.1.5. Estratégias, Tipos de Testes e Ferramentas a Serem Utilizadas.....	8
2.3.1.6. Recursos a Serem Empregados.....	8
2.3.1.7. Cronograma das Atividades.....	9
2.3.1.8. Definição dos Marcos do Projeto.....	9
2.3.2. Casos de Testes.....	10
2.3.3. Roteiro de Testes.....	10
3. Gestão de Configuração de Software.....	10,11
4. Repositório de Gestão de Configuração de Software.....	11

**CAPITULO 1. Planejamento de Testes de Software**

**1.1. Cronograma de Atividades**

<b>Atividade</b>	<b>Data de inicio</b>	<b>Data de término</b>	<b>Responsável</b>
Planejamento dos Testes	02/08/2024	07/08/2024	Equipe de QA
Preparação dos Casos de Teste	08/08/2024	10/08/2024	Analista de Testes
Execução de Testes de Performance	11/08/2024	17/08/2024	Desenvolvedores
Execução de Testes Funcionais	18/08/2024	22/08/2024	Equipe de QA
Testes de Recuperação e Backup	23/08/2024	27/08/2024	Administradores de DB
Análise de Resultados e Correções	28/08/2024	01/09/2024	Equipe de QA
Relatório Final de Testes	02/09/2024	05/09/2024	Equipe de QA

**1.2. Alocação de Recursos**

- **Recursos Humanos:** 2 analistas de QA, 4 desenvolvedores, 2 administradores de banco de dados, 1 gerente de projeto.

- **Hardware:** Servidores de banco de dados para ambientes de teste e produção, máquinas de desenvolvimento.

- **Software:** Ferramentas de teste de carga (ex: Apache JMeter, SysBench), ferramentas de monitoramento de desempenho (ex: Prometheus, Grafana), ferramentas de backup e recuperação.

### **1.3. Marcos do Projeto**

- Marcos Iniciais: Definição do escopo e planejamento do projeto de testes para o MySQL.

- Marcos Intermediários: Execução de testes de funcionalidade e carga.

- Marcos Finais: Validação final do sistema e criação de backup.

## **Capítulo 2. Documentos de Desenvolvimento de Software**

### **2.1. Plano de Projeto**

#### **2.1.1. Planejamento do Projeto**

O planejamento do projeto de teste do MySQL envolve validar melhorias de desempenho, novas funcionalidades de recuperação de dados e garantir que o sistema possa suportar grandes volumes de dados de forma eficiente. O foco está em verificar se o MySQL pode escalar horizontalmente, realizar backup de dados de maneira eficiente e fornecer alta disponibilidade.

### 2.1.2. Escopo

O escopo do projeto inclui a avaliação das seguintes melhorias no MySQL:

- Otimização de consultas complexas em grandes bancos de dados.
- Teste de novos algoritmos de indexação.
- Melhoria na escalabilidade e performance de replicação em clusters de MySQL.
- Verificação da confiabilidade dos backups automáticos e restauração de dados.

### 2.1.3. Recursos

- **Humanos:** Equipe de testes (2 pessoas), equipe de desenvolvedores (4), administradores de banco de dados (2).
- **Tecnológicos:** Ferramentas de teste como Apache JMeter para carga, ferramentas de monitoramento como Grafana, e ferramentas para backup como Percona XtraBackup.
- **Financeiros:** Orçamento de \$200.000 para testar e validar as novas melhorias e atualizações do MySQL.

### 2.1.4. Estimativas de Projeto

- **Desenvolvimento de melhorias e ajustes:** 2 meses.
- **Testes de desempenho e carga:** 1 mês.
- **Testes de recuperação e backup:** 0,5 mês.
- **Finalização e relatórios:** 0,5 mês.

Total estimado: 4 meses.

## **2.2. Documento de Requisitos**

Para o MySQL, os requisitos podem ser divididos entre funcionais e não funcionais:

### **- Requisitos Funcionais:**

1. O sistema deve suportar consultas SQL complexas sem degradação significativa de desempenho.
2. O sistema deve permitir a replicação de dados em tempo real entre servidores.
3. O MySQL deve fornecer backups incrementais e completos.

### **- Requisitos Não Funcionais:**

1. O MySQL deve ser capaz de processar pelo menos 1 milhão de consultas por minuto.
2. O tempo de recuperação após uma falha deve ser inferior a 15 minutos.
3. O sistema deve ser capaz de escalar horizontalmente para atender grandes volumes de dados sem perda de performance.

## **2.3. Planejamento de Testes**

### **2.3.1. Plano de Testes**

#### **2.3.1.1. Introdução**

O objetivo do plano de testes é garantir que as novas melhorias, como otimizações de desempenho e a implementação de algoritmos de backup, funcionem conforme esperado e que o MySQL possa ser escalado para grandes volumes de dados sem perder performance.

#### **2.3.1.2. Escopo**

Os testes envolverão:

- Testes de carga para validar o desempenho com grandes volumes de dados.
- Testes de recuperação de dados para garantir a eficiência dos backups.
- Testes de replicação para verificar se a sincronização entre servidores funciona corretamente.
- Testes de segurança para identificar possíveis vulnerabilidades em novas funcionalidades.

#### **2.3.1.3. Objetivos**

Garantir que o MySQL continue sendo uma solução robusta para gerenciamento de banco de dados, com alta disponibilidade e desempenho escalável, validando todos os requisitos funcionais e não funcionais definidos.

#### 2.3.1.4. Requisitos a Serem Testados

- Performance de consultas SQL em grandes volumes de dados.
- Eficiência de backups e recuperação de dados.
- Escalabilidade e performance em clusters de MySQL.
- Segurança e confiabilidade dos dados em transações.

#### 2.3.1.5. Estratégias, Tipos de Testes e Ferramentas a Serem Utilizadas

- **Estratégia:** Testes manuais e automáticos, focados principalmente em testes de desempenho e carga.

- **Tipos de Testes:**

- **Testes de Desempenho:** Validar o tempo de resposta de consultas SQL complexas.

- **Testes de Carga:** Simular um grande número de transações simultâneas e verificar a estabilidade do MySQL.

- **Testes de Recuperação e Backup:** Verificar a confiabilidade dos backups automáticos e a capacidade de restaurar rapidamente os dados.

- **Testes de Segurança:** Avaliar se há vulnerabilidades nas novas funcionalidades.

- **Ferramentas:**

- **Apache JMeter:** Para testar a carga e o desempenho em grandes volumes de dados.



- **Percona XtraBackup:** Para testar os backups e recuperação de dados.
- **SysBench:** Para testes de benchmarking no MySQL.
- **Grafana e Prometheus:** Para monitoramento de desempenho durante os testes.

#### **2.3.1.6. Recursos a Serem Empregados**

- **Recursos Humanos:** 2 analistas de QA, 2 administradores de banco de dados, 4 desenvolvedores.
- **Ambientes de Teste:** Instâncias de MySQL configuradas em clusters, servidores dedicados para testes de carga e recuperação.

#### **2.3.1.7. Cronograma das Atividades**

O cronograma de atividades foi detalhado na seção de planejamento de testes (1.1).

#### **2.3.1.8. Definição dos Marcos do Projeto**

- **Execução de Testes de Desempenho Concluída.**
- **Testes de Recuperação e Backup Finalizados.**
- **Validação de Replicação em Clusters.**
- **Testes de Segurança Concluídos.**

#### **2.3.2. Casos de Testes**

Exemplo de caso de teste para a replicação entre servidores:

IID	Descrição	Entrada	Saída Esperada	Status
TC 01	Verificar a replicação entre dois servidores	Atualizar dados no servidor primário	Dados replicados no servidor secundário	Aprovado
TC 02	Testar a falha de replicação com servidor inativo	Desconectar o servidor secundário	Servidor primário continua funcionando	Aprovado

### 2.3.3. Roteiro de Testes

O roteiro de testes detalha as etapas a serem seguidas durante a execução dos testes:

- **Passo 1:** Testar a replicação de dados entre dois servidores MySQL.
- **Passo 2:** Testar a recuperação de um banco de dados usando Percona XtraBackup.
- **Passo 3:** Validar o tempo de resposta de consultas SQL em um banco de dados com 10 milhões de registros.

## Capítulo 3. Gestão de Configuração de Software

A gestão de configuração do MySQL envolve o controle de versões do código-fonte e as alterações nas configurações de servidores. As versões do MySQL são gerenciadas usando **Git** e distribuídas por meio de **Docker**, permitindo um ambiente de desenvolvimento consistente.

## **Capítulo 4. Repositório de Gestão de Configuração de Software**

O repositório de gestão de configuração do MySQL é hospedado no **GitHub** e inclui o código-fonte do sistema, bem como scripts de configuração para servidores e clusters MySQL.