**Caso de Uso — Saque**

**1) Identificação**

* **ID:** UC-MOV-002
* **Nome:** Saque em Conta
* **Objetivo:** Debitar um valor de uma conta **Ativa**, respeitando **saldo disponível** e **limite diário**, registrando a movimentação e atualizando o saldo de forma **atômica** e **idempotente**.

**2) Atores**

* **Primário:** Cliente autenticado (app/web) sacando da **própria conta**.
* **Secundário (opcional):** Operador/Administrador (canal interno).

**3) Pré-condições**

* Sistema e banco de dados disponíveis.
* Usuário autenticado e autorizado (**Sacar**) para a conta alvo (ownership) ou role interna.
* Conta existente.
* **Idempotência** via IdOperacao (header) definida.

**4) Disparador**

* Solicitação de saque informando contaId e valor (> 0).

**5) Regras de Negócio e Validações (MVP)**

* **Status da conta:** deve estar **Ativa**.
  + Inativa, Bloqueada ou Encerrada → **não permite** saque.
* **Valor:** valor > 0 (BRL, duas casas).
* **Saldo disponível:** saldo >= valor. *(MVP sem “reservas”/bloqueios)*
* **Limite diário de saque:** **R$ 2.000,00** por dia por conta.
  + “Dia” calculado no fuso **America/Fortaleza** (UTC-3).
  + Somar os **saques efetivados no dia** + valor ≤ **limite**.
* **Idempotência:**
  + Mesmo IdOperacao + mesma conta + mesmo payload → **mesma resposta** (200/201).
  + Mesmo IdOperacao + payload divergente → **409 CONFLITO\_IDEMPOTENCIA**.
  + Recomenda-se **unique** (ContaId, IdOperacao) em **Movimentacao**.
* **Transação atômica:** registrar **Movimentacao (débito)** e atualizar **Saldo** no **mesmo commit**.
  + Proteger com **row-level lock**/**SERIALIZABLE**/**UPDATE checado** para evitar corrida.
* **Auditoria:** registrar IdOperacao, CorrelationId (se houver), canal, timestamp local e UTC, usuário.

**Obs. técnica p/ limite diário (MVP):**  
Persistir também dataOperacaoLocal (YYYY-MM-DD em America/Fortaleza) na Movimentação para consultas de limite eficientes e sem ambiguidade de fuso.

**6) Fluxo Principal (Feliz)**

1. Ator envia contaId, valor e IdOperacao.
2. Sistema valida existência da conta e **Status = Ativa**.
3. Sistema valida valor > 0 e formato.
4. Sistema verifica **idempotência** (ContaId + IdOperacao).
5. Sistema inicia **transação** e obtém **lock** na conta.
6. Sistema calcula **saques do dia (local)** e checa limite diário remanescente.
7. Sistema valida **saldo >= valor**.
8. Sistema registra **Movimentacao** do tipo **Saque** (débito) com SaldoAnterior e SaldoPosterior.
9. Sistema atualiza o **Saldo** da conta (saldo = saldo − valor).
10. Sistema confirma a transação e registra auditoria.
11. Sistema retorna **201 Created** com dados do movimento e saldo atualizado.

**7) Fluxos Alternativos / Exceções**

* **A1 — Conta inexistente:**  
  2a. **404 Not Found** CONTA\_NAO\_ENCONTRADA.
* **A2 — Conta não ativa:**  
  2b. **409 Conflict** CONTA\_NAO\_ATIVA.
* **A3 — Valor inválido (≤ 0 ou formato):**  
  3a. **422 Unprocessable Entity** VALOR\_INVALIDO.
* **A4 — Idempotência (reenvio igual):**  
  4a. **200/201** com a **mesma resposta** (sem novo débito).
* **A5 — Idempotência (reenvio divergente):**  
  4b. **409 Conflict** CONFLITO\_IDEMPOTENCIA.
* **A6 — Limite diário excedido:**  
  6a. **409 Conflict** LIMITE\_DIARIO\_EXCEDIDO (informar restante).
* **A7 — Saldo insuficiente:**  
  7a. **409 Conflict** SALDO\_INSUFICIENTE.
* **A8 — Falha de persistência:**  
  10a. **500 Internal Server Error** ERRO\_PERSISTENCIA.
* **A9 — Permissão insuficiente:**  
  0a. **403 Forbidden** PERMISSAO\_NEGADA.

**8) Pós-condições**

* Movimentação de **Saque** criada.
* **Saldo** da conta reduzido.
* Auditoria registrada (IdOperacao, CorrelationId, usuário, canal).
* Limite diário atualizado implicitamente (via soma das movimentações do dia).

**9) Dados Persistidos (rascunho)**

* **Movimentacao:**
  + Id (GUID)
  + ContaId (GUID, FK)
  + Tipo (enum: **Saque**)
  + Valor (decimal(18,2))
  + SaldoAnterior (decimal(18,2))
  + SaldoPosterior (decimal(18,2))
  + Descricao (string, opcional)
  + IdOperacao (GUID/string) **[único por ContaId]**
  + CorrelationId (GUID/string, opcional)
  + DataOperacaoUtc (datetime), DataOperacaoLocal (date, America/Fortaleza)
  + CriadoEm (datetime), CriadoPor (string/userId)
* **Conta:**
  + Saldo atualizado
  + AtualizadoEm, AtualizadoPor (opcional)
* **Índices/Restrições:**
  + **Unique** (ContaId, IdOperacao) em **Movimentacao**.
  + Índices por ContaId + DataOperacaoLocal + Tipo (consulta de limite diário).

**10) Interface/Contrato de API (MVP)**

* **Endpoint:** POST /api/contas/{id}/saques
* **Headers:**
  + IdOperacao: "<uuid>" *(obrigatório)*
  + X-Correlation-Id: "<uuid>" *(recomendado)*
* **Request (exemplo):**

{

"valor": 180.00,

"descricao": "Saque via app"

}

* **Response 201 (exemplo):**

{

"movimentoId": "6cf7e20c-9c6e-4d3e-bb0a-1a23c45d9f00",

"contaId": "7f6a2e34-1c67-4b1a-9d23-1f23a1b9c0e3",

"tipo": "Saque",

"valor": 180.00,

"saldoAnterior": 500.00,

"saldoPosterior": 320.00,

"limiteDiarioRestante": 1820.00,

"criadoEm": "2025-08-15T01:30:00Z"

}

* **Erros comuns:** 403, 404, 409, 422, 500.

**11) Critérios de Aceitação (Given–When–Then)**

1. **Saque válido**
   * Given conta **Ativa**, valor > 0, saldo suficiente e dentro do limite
   * When POST /api/contas/{id}/saques com IdOperacao
   * Then **201** com movimento criado e saldo atualizado.
2. **Saldo insuficiente**
   * Given saldo < valor
   * Then **409 SALDO\_INSUFICIENTE**.
3. **Limite diário excedido**
   * Given soma dos saques do dia + valor > 2000
   * Then **409 LIMITE\_DIARIO\_EXCEDIDO**.
4. **Conta não ativa**
   * Given conta Bloqueada/Inativa/Encerrada
   * Then **409 CONTA\_NAO\_ATIVA**.
5. **Idempotência (reenvio igual)**
   * Given mesmo IdOperacao e payload
   * Then **200/201** com **mesma resposta** (sem novo débito).
6. **Valor inválido**
   * Given valor <= 0
   * Then **422 VALOR\_INVALIDO**.

**12) Observações de Segurança e Privacidade**

* Não logar valores/IDs sensíveis em nível **INFO** sem máscara.
* HTTPS obrigatório.
* Autorização por **ownership** (cliente só saca da própria conta) ou role interna.
* Considerar exigir **2FA/PIN** para saque via canais críticos (futuro).

**13) Métricas & Logs (MVP)**

* Taxa de sucesso; taxa de **409** por **limite** e **saldo**; tempo médio de processamento.
* Logs estruturados: ContaId, MovimentoId, IdOperacao, Valor, SaldoAnterior→SaldoPosterior, DataOperacaoLocal, canal.

**14) TODOs / Futuro**

* **[TODO-SAQUE-01]** Antifraude/limites adicionais por canal (valor/tx/dia/semana).
* **[TODO-SAQUE-02]** Definir **taxas/tarifas** de saque (se aplicável) e contabilização.
* **[TODO-SAQUE-03]** Suportar “reservas/bloqueios” para evitar disputa de fundos com outras operações concorrentes.
* **[TODO-SAQUE-04]** Exigir **2FA/PIN** em canais específicos.
* **[TODO-SAQUE-05]** Endurecer concorrência: **UPDATE com cláusula** WHERE Saldo >= :valor e verificação de rows affected para blindar corrida.