**Psicool**

**Especificação de Objetivos e Requisitos**

**PSC-EOR-002**

Goiânia

Outubro de 2025

**Revisões**

A primeira versão deste documento é criada após a sua aprovação e é vinculada a uma baseline do software. Esta versão, portanto, não pode ser modificada. As modificações que se fizerem necessárias após a criação da baseline farão parte de uma versão seguinte que será vinculada a outra baseline. Este procedimento pode se repetir sucessivamente. As modificações introduzidas em cada versão devem ser registradas seguindo o modelo do quadro abaixo. Desta forma, será possível perceber as diferenças entre as diversas versões geradas.

|  |  |  |
| --- | --- | --- |
| *Data* | *Descrição* | *Autor* |
| 12/09/2025 | Criação inicial do documento | Danilo Matias Barcelos,  Lucas Teixeira Correia |
| 14/09/2025 | Criação dos primeiros Requisitos Não-Funcionais | Danilo Matias Barcelos |

**Conteúdo**

1. Introdução 3
   1. Objetivos 3
   2. Público Alvo 3
   3. Organização do documento 3
2. Descrição do problema e do sistema 4
   1. Identificação e missão do Sistema 4
   2. Domínio do problema e contexto de sua aplicação 4
   3. Objetivos e Benefícios Esperados do Sistema 4
   4. Características Essenciais do Sistema 4
   5. Descrição dos interessados do sistema 5
   6. Diagnostico da Situação Atual 5
      1. Áreas de negócio atendidas 5
      2. Áreas de negócio não atendidas 5
      3. Integração com outros sistemas da empresa 5
      4. Integração com outros sistemas de terceiros 5
      5. Pontos negativos 5
      6. Pontos positivos 6
      7. Sugestões de melhorias 6
3. Casos de Uso 6
   1. Atores 6
   2. Diagramas de Caso de Uso e Lista de casos de uso 6
   3. Descrição de Casos de Uso 6
4. Requisitos e restrições funcionais (RFUN) 7
5. Requisitos e restrições não funcionais 10
   1. Requisitos e restrições de informação (RINF) 10

5.2 Requisitos e restrições de interface Homem-Computador (RHIC) 10

5.3 Requisitos de Interface Externa (RIEX) 11

5.4 Requisitos e Restrições de Projeto (RPRO) 12

5.5 Requisitos e restrições de arquitetura de software (RARQ) 12

5.6 Requisitos e restrições de plataforma de hardware (RPHW) 12

5.7 Requisitos e restrições de plataforma de software (RPSW) 12

5.8 Requisitos e restrições de desempenho (RDES) 13

5.9 Requisitos e restrições de disponibilidade (RDIS) 13

5.10 Requisitos e restrições de segurança (RSEG) 14

5.11 Requisitos e restrições de manutenibilidade (RMAN) 14

5.12 Requisitos e restrições de portabilidade (RPOR) 15

5.13 Requisitos de documentação (RDOC) 15

1. Requisitos Futuros (RFUT) 15
2. Referências cruzadas complementares 16
3. Aprovação Formal 17
4. Bibliografia 17

10. Anexos 17

**Figuras e Tabelas**

Incluir índice de figuras e tabelas mostradas no documento usando o mesmo modelo do índice de conteúdo.

# Introdução

Apresentar o documento ao leitor, descrevendo sucintamente o software que é objeto deste projeto e as informações contidas neste documento.

## Objetivos

Definir os objetivos do documento.

*Exemplo:*

*Este documento descreve todos os casos de uso, requisitos que deverão ser atendidos pelo Sistema XXXX que será desenvolvido para a empresa XXXXXXXX.*

## Público Alvo

Identificar o público alvo do documento, isto é, os órgãos e pessoas que poderão usá-lo.

## Organização do documento

Descrever a organização e o conteúdo de cada seção do documento.

# Descrição do problema e do sistema

Nesta seção o sistema objeto da especificação de requisitos deve ser descrito. O domínio do problema que deverá ser resolvido pelo sistema deve ser explicado e também as características específicas do problema no contexto da empresa em que o sistema deverá ser utilizado. As subseções seguintes podem ser colocadas no documento EOR para separar cada conteúdo específico, mas também pode ser usado um estilo de texto sem separação de subseções desde que todo o conteúdo seja coberto.

## Identificação e missão do Sistema

Identificar o sistema objeto da especificação de requisitos e definir em poucas palavras qual é a missão do sistema que está sendo considerado neste documento.

## Domínio do problema e contexto de sua aplicação

A descrição do domínio do problema deve complementar e detalhar a visão geral do domínio do problema apresentada no documento de ativação do projeto (APD), identificando o contexto deste problema no ambiente alvo (ambiente do cliente alvo).

*Exemplo:*

*Suponha que o domínio do problema seja a venda de produtos em um comércio varejista. Esta seção deverá explicar em que consiste o problema de vender produtos em uma empresa que atua no comércio varejista, ou seja, descrever genericamente quais são os processos de negócio envolvidos e seus objetivos. Além disso, o problema deve ser contextualizado mostrando as características específicas do problema na empresa alvo.*

## Objetivos e Benefícios Esperados do Sistema

Esta seção deve refinar a definição do produto contida no documento APD, ajudando a estabelecer o escopo do sistema a ser desenvolvido. Listar todos os objetivos a serem alcançados com o desenvolvimento do sistema e os respectivos benefícios esperados.

Exemplo:

|  |  |
| --- | --- |
| ***Objetivo a ser alcançado*** | ***Benefício adquirido*** |
| *Atualizar automaticamente o estoque de produtos após a realização de uma venda.* | *Permitir que os vendedores tenham informações atualizadas sobre os produtos disponíveis em estoque;* |

## Características Essenciais do Sistema

Descrever todas as características que devem ser consideradas para avaliar o sucesso do produto construído. Os detalhes necessários a essa avaliação devem ser especificados. Por exemplo, se a capacidade do sistema ser executado em diferentes plataformas computacionais é especificada como uma característica do sistema, deve ser definido um conjunto de plataformas que serão usadas para avaliar essa característica.

O produto final do processo de desenvolvimento será validado com relação à missão, aos objetivos e às características descritas neste documento.

## Descrição dos interessados do sistema

Descrever os perfis de cada interessado envolvido com o sistema.

|  |  |  |
| --- | --- | --- |
| *Interessado(s)* | *Descrição* | *Necessidades Básicas* |
| Nome do interessado. Por exemplo: (cliente, atendente, Almoxarifado, etc.) | Descrever o interessado e sua interação com o sistema | Indicar quais as necessidades básicas que o interessado tem em relação ao sistema. Por exemplo: o vendedor precisa de um sistema que lhe permita registrar uma venda feita com segurança e rapidez. |

## Diagnostico da Situação Atual

Descrever a situação atual do sistema que será substituído e/ou informatizado. O software a ser desenvolvido pode substituir outro software, complementar as funcionalidades de outro software ou pode servir para informatizar um sistema manual. Em ambos os casos, é preciso analisar o sistema que será modificado pela implantação do novo software.

### Áreas de negócio atendidas

Descrever as áreas de negócio (unidades organizacionais) dos clientes (ou da própria empresa, no caso de sistemas de uso interno), que utilizam o sistema atual. Para cada área de negócio identificada deverão ser indicados quais os processos de negócio realizados e como cada processo é executado (para quê, quando e como). Além disso, devem ser identificadas as informações manipuladas no processo.

### Áreas de negócio não atendidas

Descrever os processos de negócio que deveriam ou poderiam ser realizados no sistema atual, mas não o são. Identificar que tipo de suporte deveria ser fornecido pelo sistema (o quê deveria ser feito, quando e como) e que tipo de informações seriam necessárias para esse fim.

### Integração com outros sistemas da empresa

Explicar como o sistema atual se integra com outros sistemas da empresa (que tipo de integração existe, como, quando e por que ela é feita).

### Integração com outros sistemas de terceiros

Explicar como o sistema atual se integra com sistemas desenvolvidos por outras empresas (que tipo de integração existe, como, quando e por que ela é feita).

### Pontos negativos

Identificar os problemas e dificuldades que o sistema atual apresenta em termos de funcionalidades, desempenho ou qualquer outro aspecto relevante.

### Pontos positivos

Identificar as virtudes e facilidades que o sistema atual apresenta em termos de funcionalidades, desempenho ou qualquer outro aspecto relevante.

### Sugestões de melhorias

Listar as principais sugestões de melhorias para o sistema atual, que foram coletadas com clientes, usuários e colaboradores da empresa. Essas sugestões deverão ser classificadas em níveis de prioridade (por exemplo: alta, média, baixa, e descartada).

# Casos de Uso

Nesta seção devem ser identificados os casos de uso do software que será desenvolvido, os atores neles envolvidos e os requisitos funcionais e não funcionais neles tratados.

## CSU1 – Agendar Consulta (Cliente)

|  |  |
| --- | --- |
| Nome do Caso de Uso | Agendar consulta |
| Escopo | Sistema de gerenciamento de consultas psicológicas (plataforma online). |
| Nível | Objetivo do usuário. |
| Ator principal | Cliente (paciente) |
| Interessados e interesses | - Cliente: deseja reservar um horário disponível com um psicólogo de sua escolha.  - Psicólogo: deseja manter a agenda atualizada e evitar conflitos de horários.  - Sistema: precisa garantir integridade das agendas e registrar logs para auditoria. |
| Pré-condições | - Cliente autenticado no sistema.  - Psicólogos cadastrados e com horários disponíveis. |
| **Garantia de sucessos** | Consulta registrada com sucesso no banco de dados, (RFUN32)log gerado e notificação enviada. |
| **Cenário de sucesso principal** | 1. O cliente acessa a tela de agendamento. 2. O sistema exibe a lista de psicólogos disponíveis. 3. O cliente seleciona o profissional desejado. 4. O sistema mostra as datas e horários livres. 5. O cliente escolhe o horário e confirma. 6. O sistema valida disponibilidade e consistência dos dados. 7. (RFUN32)O sistema grava o agendamento, gera log e envia confirmação ao cliente. |
| **Extensões** | - 4a. Nenhum horário disponível → o sistema exibe mensagem e sugere outras datas.  - 6a. Falha de comunicação com o servidor → o sistema solicita nova tentativa. |
| **Requisitos especiais** | - (RFUN32)Log obrigatório de criação de agendamento.  - Interface responsiva e compatível com dispositivos móveis.  - Validação automática de conflitos de horário. |
| **Lista de variantes tecnológicas e de dados** | - Entrada via formulário web / aplicativo mobile.  - Armazenamento em banco de dados SQL. |
| **Frequência de ocorrência** | Alta. |
| **Diversos** | Pode gerar lembrete automático via e-mail ou notificação push. |

## CSU2 – Agendar Consulta (Empresa / Psicólogo)

|  |  |
| --- | --- |
| Nome do Caso de Uso | Agendar consulta manualmente |
| Escopo | Sistema de gerenciamento de consultas psicológicas. |
| Nível | Sub-função administrativa. |
| Ator principal | Psicólogo ou administrador da clínica. |
| Interessados e interesses | - Psicólogo: inserir manualmente um agendamento para um paciente.  - Cliente: ser informado da consulta criada.  - Sistema: garantir integridade e registro do log administrativo. |
| Pré-condições | • Psicólogo autenticado.  • Cliente cadastrado. |
| **Garantia de sucessos** | Agendamento salvo corretamente com (RFUN32)log administrativo e notificação enviada. |
| **Cenário de sucesso principal** | 1. O psicólogo acessa a agenda. 2. Seleciona o paciente. 3. Define data e horário da consulta. 4. Confirma a criação. 5. O sistema verifica disponibilidade. 6. (RFUN32)O sistema grava o agendamento e gera o log. 7. O sistema envia notificação ao paciente. |
| **Extensões** | - 5a. Horário já ocupado → o sistema alerta e solicita nova escolha.  - 6a. Falha de gravação → o sistema exibe erro e registra tentativa. |
| **Requisitos especiais** | - Restrição de acesso para usuários com permissão “psicólogo”.  - (RFUN32)Logs de todas as ações administrativas. |
| **Lista de variantes tecnológicas e de dados** | • Painel web administrativo.  Integração com módulo de notificações. |
| **Frequência de ocorrência** | Média. |
| **Diversos** | Pode gerar relatório mensal de consultas criadas manualmente. |

## CSU3 – Cancelar Consulta (Cliente)

|  |  |
| --- | --- |
| Nome do Caso de Uso | Cancelar consulta |
| Escopo | Sistema de gerenciamento de consultas psicológicas. |
| Nível | Objetivo do usuário. |
| Ator principal | Cliente (paciente). |
| Interessados e interesses | - Cliente: deseja cancelar a consulta dentro do prazo permitido.  - Psicólogo: deve ser notificado do cancelamento.  - Sistema: aplicar políticas de multa e registrar log da operação. |
| Pré-condições | • Cliente autenticado.  • Consulta existente e futura. |
| **Garantia de sucessos** | Consulta cancelada com log gravado e regra de multa aplicada corretamente. |
| **Cenário de sucesso principal** | 1. O cliente acessa a agenda de consultas. 2. Seleciona uma consulta agendada. 3. Solicita cancelamento. 4. O sistema valida o prazo e as regras de multa. 5. O sistema aplica multa (se necessário). 6. (RFUN32)O sistema registra o cancelamento e grava log. 7. O sistema envia confirmação ao cliente e notifica o psicólogo. |
| **Extensões** | - 4a. Cancelamento fora do prazo → o sistema bloqueia a ação e exibe mensagem explicativa.  - 5a. Falha no pagamento da multa → o sistema mantém o agendamento até regularização. |
| **Requisitos especiais** | - Aplicação automática das regras de negócio.  -(RFUN32)Registro de logs transacionais |
| **Lista de variantes tecnológicas e de dados** | - Interface web / aplicativo.  - Integração com gateway de pagamento. |
| **Frequência de ocorrência** | Média. |
| **Diversos** | Pode permitir justificativa do cliente para cancelamentos frequentes. |

## CSU4 – Cancelar Consulta (Empresa / Psicólogo)

|  |  |
| --- | --- |
| Nome do Caso de Uso | Cancelar consulta (administrativo) |
| Escopo | Sistema de gerenciamento de consultas psicológicas. |
| Nível | Objetivo do usuário. |
| Ator principal | Psicólogo ou administrador. |
| Interessados e interesses | - Psicólogo: precisa cancelar uma consulta por imprevisto.  - Cliente: deve ser informado e receber estorno, se aplicável.  - Sistema: precisa garantir consistência e registrar o evento. |
| Pré-condições | - Psicólogo autenticado.  - Consulta existente. |
| **Garantia de sucessos** | Consulta cancelada, (RFUN32)log gravado e estorno processado (quando aplicável). |
| **Cenário de sucesso principal** | 1. O psicólogo acessa sua agenda de consultas. 2. Seleciona a consulta a cancelar. 3. Solicita o cancelamento. 4. O sistema solicita confirmação. 5. (RFUN32)O sistema registra cancelamento e log. 6. O sistema processa o estorno e envia notificações. |
| **Extensões** | - 6a. Falha no estorno → o sistema informa erro e aciona o suporte financeiro. |
| **Requisitos especiais** | - (RFUN32)Registro completo de logs administrativos.  - Regras automáticas de estorno conforme política da clínica. |
| **Lista de variantes tecnológicas e de dados** | - Painel administrativo web.  - Integração com módulo financeiro. |
| **Frequência de ocorrência** | Baixa. |
| **Diversos** | Pode gerar relatório de cancelamentos por profissional. |

## CSU5 – Check-in (Cliente)

|  |  |
| --- | --- |
| Nome do Caso de Uso | Realizar check-in |
| Escopo | Sistema de gerenciamento de consultas psicológicas. |
| Nível | Objetivo do usuário. |
| Ator principal | Cliente (paciente). |
| Interessados e interesses | - Cliente: confirmar presença antes do início da consulta.  - Psicólogo: visualizar a confirmação em tempo real.  - Sistema: atualizar o status da consulta e armazenar horário de confirmação. |
| Pré-condições | • Consulta agendada e dentro do prazo permitido para check-in |
| **Garantia de sucessos** | Status da consulta atualizado para “confirmada” e registro de data/hora armazenado. |
| **Cenário de sucesso principal** | 1. O cliente acessa as consultas marcadas. 2. O sistema exibe as consultas. 3. O cliente seleciona a consulta e solicita check-in. 4. O sistema valida horário e status. 5. O sistema atualiza o status para “confirmada”. 6. (RFUN32)O sistema registra o log e mostra confirmação visual ao cliente. |
| **Extensões** | - 4a. Tentativa fora do horário permitido → sistema nega o check-in.  - 5a. Falha de conexão → sistema permite tentativa posterior até o início da consulta. |
| **Requisitos especiais** | - Compatibilidade com mobile.  - Confirmação visual imediata (ex.: ícone verde de check). |
| **Lista de variantes tecnológicas e de dados** | - Aplicativo mobile / web.  - Registro em banco de dados transacional |
| **Frequência de ocorrência** | Alta. |
| **Diversos** | Dados de check-in podem alimentar relatórios de assiduidade e pontualidade. |

## CSU6 -- Criação de prontuário

|  |  |
| --- | --- |
| **Nome do caso de uso** | Criar prontuário |
| **Escopo** | Sistema de gerenciamento da clínica psicológica |
| **Nível** | Objetivo de usuário |
| **Ator principal** | Psicólogo (ou profissional autorizado) |
| **Interessados e interesses** | - Psicólogo: deseja registrar dados clínicos do paciente de forma completa e segura.  - Paciente: quer que seu prontuário seja correto, privado e acessível apenas por pessoas autorizadas.  - Clínica: precisa de prontuários consistentes e rastreabilidade para auditoria. |
| **Pré-condições** | - Psicólogo autenticado e autorizado.  - Paciente previamente cadastrado (ou opção de cadastro durante o fluxo).  - Sistema disponível e com conexão ao banco de dados. |
| **Garantia de sucessos** | - Prontuário criado e persistido com identificador único.  - Registro de log da criação gravado (RFUN31): contém usuário, timestamp e resumo da ação. |
| **Cenário de sucesso principal** | 1. Psicólogo seleciona “Criar prontuário.” 2. Sistema exibe formulário 3. Psicólogo preenche e envia o formulário 4. Sistema salva novo prontuário no banco de dados e gera identificador. 5. Sistema gera automaticamente log de criação com: usuário, id do prontuário, operação, horário da ação, campos modificados/resumidos. 6. Sistema exibe confirmação e permite acessar o prontuário recém-criado. |
| **Extensões** | * 3c. Anexo corrompido ou não suportado   →Usuário tenta anexar arquivo (ex.: PDF, imagem) corrompido ou de tipo não permitido.  Sistema rejeita o upload e informa o motivo (“formato não suportado” / “arquivo corrompido”).   * 4a. Dados inconsistentes com o cadastro do paciente   →O sistema detecta divergência entre o prontuário e os dados cadastrais (ex.: CPF, inválido, paciente inativo).  Sistema alerta o usuário e solicita correção ou revalidação do paciente.   * 5a. Falha de persistência (DB indisponível)   →Sistema registra falha local e exibe mensagem; operação pode ser enfileirada para retrial.  Cria registro de erro e indica tentativa   * 6a. Gera versão histórica |
| **Requisitos especiais** | - Criptografia em repouso para campos sensíveis.  - Geração obrigatória de log (RFUN31).  - Retenção de logs por 20 anos.  - Conformidade com confidencialidade profissional (CFP / LGPD). |
| **Lista de variantes tecnológicas e de dados** | - Entrada: formulário web (JSON); formatos: texto, anexos (PDF).  - Saída: id do prontuário, notificação de sucesso, evento de log.  - Armazenamento: banco relacional com versionamento de prontuário. |
| **Frequência de ocorrência** | Várias vezes por dia. |
| **Diversos** | Logs visíveis apenas para perfis administrativos. |

## CSU7 – Edição de prontuário

|  |  |
| --- | --- |
| **Nome do caso de uso** | Editar prontuário |
| **Escopo** | Sistema de gerenciamento de clínica psicológica |
| **Nível** | Subfunção / objetivo do usuário (atualizar um recurso existente) |
| **Ator principal** | Psicólogo |
| **Interessados e interesses** | - Psicólogo: atualizar dados e observações  - Clínica: manter histórico e rastreabilidade das mudanças.  - Paciente: garantir integridade e histórico das informações. |
| **Pré-condições** | - Usuário autenticado e autorizado  - Prontuário existente.  - Sistema disponível |
| **Garantia de sucessos** | Prontuário atualizado com novos dados; um registro de log detalhado é gerado (RFUN31) mostrando usuário, timestamp, campos alterados e valores anteriores/novos, e versão anterior salva. |
| **Cenário de sucesso principal** | 1. Psicólogo abre o prontuário do paciente. 2. Sistema exibe versão atual com opção “Editar”. 3. Psicólogo altera campos e salva. 4. Sistema valida mudanças segundo regras (consistência). 5. Sistema grava nova versão e armazena metadados de versão. 6. Sistema gera log de edição (RFUN31) com: usuário, id do prontuário, operação=edição, timestamp, lista de campos alterados, valores antigos (se necessário) e novos. 7. Sistema salva versão anterior no banco de dados. 8. Sistema confirma atualização ao usuário. |
| **Extensões** | * 3a. Tentativa de alterar campo restrito   → Usuário tenta alterar dados pessoais do cliente (CPF, nome, data de nascimento).  Sistema bloqueia e solicita autorização extra.   * 5a. Falha ao salvar   →Usuário tenta fazer o salvamento mas o sistema está fora do ar.  Sistema notifica e registra erro; se houver retries, registrar tentativas em log. |
| **Requisitos especiais** | - RFUN31: obrigatoriedade de geração de log de edição.  - Controle de versão |
| **Lista de variantes tecnológicas e de dados** | - Frontends: web; comunicação via API REST (JSON).  - Dados sensíveis: tratamento e encriptação. |
| **Frequência de ocorrência** | Alta – durante e entre atendimentos |
| **Diversos** | Acesso a histórico restrito a perfis com permissão. |

## CSU8 – Armazenamento de prontuário (versões)

|  |  |
| --- | --- |
| **Nome do caso de uso** | Armazenar versões de prontuário |
| **Escopo** | Módulo de versionamento |
| **Nível** | Subfunção (suporte automático ao ciclo de vida do prontuário) |
| **Ator principal** | Sistema (processo interno) |
| **Interessados e interesses** | - Clínica: manter cópias históricas para auditoria e integridade.  - Psicólogo: reverter alterações ou consultar versões anteriores.  - Auditor/Legislação: garantir rastreabilidade. |
| **Pré-condições** | - Evento de criação ou edição do prontuário concluído (gatilho).  - Mecanismo de armazenamento de versões configurado. |
| **Garantia de sucessos** | Versão anterior do prontuário é armazenada e indexada; log de armazenamento gerado (RFUN31) indicando operação e metadados de versão. |
| **Cenário de sucesso principal** | 1. Após operação de criação/edição, sistema gera evento de versionamento. 2. Subsistema de versões captura snapshot do prontuário. 3. Snapshot é persistido em repositório de versões (p.ex. tabela de histórico ou S3). 4. Sistema grava metadados de versão (autor, timestamp, id da versão). 5. Sistema gera log de armazenamento (RFUN31) registrando o evento. 6. Confirmação de armazenamento é feita e disponível para consultas administrativas. |
| **Extensões** | * 2a. Falha ao gerar snapshot   →Sistema tenta gravar porém não há a gravação.  Sistema tenta em retry; se falhar, registra erro crítico e alerta administrador.   * 3a. Política de retenção acionada   →Tempo máximo de retenção do prontuário de um paciente é atingido.  Versão mais antiga arquivada para cold storage; ação registrada em log. |
| **Requisitos especiais** | - RFUN31: logs obrigatórios em cada operação de versionamento.  - Políticas de retenção e arquivamento configuráveis.  - Integridade: checksums e verificação periódica. |
| **Lista de variantes tecnológicas e de dados** | - Snapshot completo (JSON) ou delta; armazenamento em banco relacional, NoSQL ou object storage.  - Metadados para pesquisa (idProntuário, idVersão, autor, timestamp). |
| **Frequência de ocorrência** | Sempre que ocorrer criação/edição (alto). |
| **Diversos** | Deve existir interface administrativa para restauração de versões (caso seja necessário — pode ser outro caso de uso). |

## CSU9

|  |  |
| --- | --- |
| **Nome do caso de uso** | Exportar prontuário |
| **Escopo** | Sistema de gerenciamento de clínica psicológica |
| **Nível** | Objetivo do usuário |
| **Ator principal** | Psicólogo ou Administrador |
| **Interessados e interesses** | - Psicólogo: obter cópia para encaminhamento, relatório ou arquivo.  - Paciente: direito de obter cópia quando solicitado (conforme políticas).  - Clínica: garantir exportação segura e registrada. |
| **Pré-condições** | - Usuário autenticado e autorizado para exportar.  - Prontuário existente. |
| **Garantia de sucessos** | Arquivo exportado (PDF, XML, JSON) gerado e disponibilizado para download; evento registrado em log (RFUN31) contendo quem exportou, formato, id do prontuário e timestamp. |
| **Cenário de sucesso principal** | 1. Usuário seleciona prontuário e escolhe “Exportar”. 2. Sistema apresenta opções de formato e escopo (ex.: completo, apenas anotações). 3. Usuário confirma parâmetros e solicita exportação. 4. Sistema gera o arquivo no formato escolhido (p. ex. PDF com assinatura digital, ou XML). 5. Sistema grava registro de exportação no log (RFUN31) com detalhes de autorização, formato e destino. 6. Sistema disponibiliza arquivo para download ou envia via canal seguro (e-mail cifrado/portal). |
| **Extensões** | * 3a. Usuário sem permissão   →Usuário com baixa permissão tenta fazer exportação.  Acesso negado; evento de tentativa registrado em log de segurança.   * 4a. Falha na geração do arquivo   →Sistema não consegue efetuar a geração do arquivo no formato escolhido.  Sistema registra erro e sugere tentativa posterior. |
| **Requisitos especiais** | - RFUN31: geração de log obrigatória.  - Criptografia para arquivos sensíveis; watermark/assinatura para rastreabilidade.  - Auditoria para downloads e tentativas. |
| **Lista de variantes tecnológicas e de dados** | - Formatos: PDF (com marca d’água), XML (interoperabilidade), JSON (integração).  - Geração assíncrona para arquivos grandes (fila/worker). |
| **Frequência de ocorrência** | Baixa a moderada (sob demanda). |
| **Diversos** | Rastreabilidade completa para solicitações legais ou auditorias. |

## CSU10 – Agenda (visualização / edição de datas e horários/ integração)

|  |  |
| --- | --- |
| **Nome do caso de uso** | Gerenciar agenda |
| **Escopo** | Sistema de gerenciamento de clínica psicológica. |
| **Nível** | Objetivo do usuário. |
| **Ator principal** | Psicólogo / Secretário(a) / Paciente / Administrador |
| **Interessados e interesses** | - Paciente: deseja marcar uma consulta com o psicólogo em um horário disponível, podendo escolher entre atendimento online ou presencial.  - Psicólogo: deseja ter sua agenda organizada e atualizada com os horários de suas consultas.  - Secretário: quer facilitar o agendamento em nome do cliente, mantendo controle sobre horários.  - Administrador: busca garantir o funcionamento correto da agenda e a geração dos logs. |
| **Pré-condições** | - O ator deve estar autenticado no sistema.  - O profissional deve estar ativo e possuir horários disponíveis.  - O sistema de agenda deve estar acessível. |
| **Garantia de sucessos** | - A consulta é registrada corretamente na agenda do profissional.  - O sistema gera logs de criação e integração (RFUN31 e RFUN32).  - O cliente e o psicólogo são notificados do agendamento. |
| **Cenário de sucesso principal** | 1. Usuário acessa o módulo de agenda. 2. Sistema exibe visualização do calendário (dia/semana/mês) com status de cada slot (livre, reservado, bloqueado). 3. Usuário seleciona um horário para criar, editar, bloquear ou reagendar uma consulta. 4. No caso de agendamento, o sistema solicita a modalidade da consulta (online ou presencial). 5. Usuário confirma a ação. 6. Sistema valida disponibilidade e regras de negócio (evita sobreposição e horários inválidos). 7. Sistema persiste a alteração e, se configurado, sincroniza com sistemas externos. 8. Sistema gera log da ação (RFUN32) com informações do usuário, tipo de operação, slot alterado e timestamp. 9. Sistema confirma a atualização e envia notificações automáticas aos envolvidos (paciente, psicólogo, etc.). |
| **Extensões** | * 3a. Tentativa de reserva em horário ocupado   Sistema sugere horários alternativos   * 7a. Falha na integração com API externa   →Falha na comunicação com software que oferece serviços fora da plataforma  Sistema enfileira sincronização e registra a falha em log. |
| **Requisitos especiais** | - RFUN32: geração obrigatória de log de integração/alteração da agenda.  - Validação em tempo real para evitar duplo agendamento. Envio automático de notificações após qualquer alteração. |
| **Lista de variantes tecnológicas e de dados** | - Frontend: calendário interativo com filtros de visualização (por profissional, dia, semana, mês).  - Backend: API REST para operações CRUD e sincronização externa.  - Integração: Google Calendar, Outlook e WhatsApp (para notificações).  - Dados: idProfissional, idSlot, status (livre/reservado/bloqueado), tipoConsulta (online/presencial), timestamps. |
| **Frequência de ocorrência** | Diária (alta frequência de operações de visualização e agendamento). |
| **Diversos** | - Logs de agenda integrados ao histórico geral do sistema (RFUN32).  - Possibilidade de auditoria de histórico de modificações para fins legais e administrativos. |

## 

## 

## 

## 

## 

## 

## 

Listar todos os atores do software. Ator é tudo aquilo que interage com o software, por exemplo, pessoas, órgãos, software, máquinas, etc.

*Exemplo:*

1. Cliente
2. Software de Faturamento
3. Atendente
4. Telefonista
5. Supervisor

## Diagramas de Caso de Uso e Lista de casos de uso

Mostrar os casos de uso do sistema através de diagrama(s) de caso de uso identificados. Listar todos os casos de uso do software identificando sua categoria: primário, secundário ou opcional. Casos de uso primários são aqueles que representam processos comuns principais; casos de uso secundários representam processos menos importantes ou raros; casos de uso opcionais representam processos que talvez não sejam considerados. Quando define a categoria de um caso de uso é mais fácil de perceber quais casos de uso deverão ser expandidos primeiramente.

*Exemplo:*

|  |  |  |  |
| --- | --- | --- | --- |
| **Ref.** | **Descrição** | **Atores** | **Categoria** |
| CSU1 | Registrar ligação | Telefonista | Primário |
| CSU2 | Solicitar impressão de dados de clientes | Atendente | Secundário |

## Descrição de Casos de Uso

Identificar e descrever cada caso de uso listado anteriormente.

Os casos de uso deverão ser descritos preferencialmente no formato de alto nível, mas dependendo da sua importância ou da sua complexidade eles poderão ser descritos no formato essencial expandido. Casos de uso essenciais são aqueles que não referenciam aspectos de soluções tecnológicas adotadas ao contrário dos casos de uso em formato concreto que referenciam as estruturas de interface da solução projetada para o sistema e outros aspectos tecnológicos do sistema.

# Requisitos e restrições funcionais (RFUN)

Elaborar uma lista das funções que o software deve prover. Considerar não apenas as funções intrínsecas (essenciais) do software, mas também as de suporte, tais como administração da utilização do software, cópias de segurança, auditoria, e controle de acesso.

Indicar para cada função uma referência, descrição, categoria, prioridade e casos de uso em que ela é tratada.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ref.** | **Função** | **Categoria** | **Prioridade** | **Casos de Uso** |
| RFUN1 | CRUD do cliente | Evidente | Alta |  |
| RFUN2 | Agendamento de consulta pelo cliente | Evidente | Alta |  |
| RFUN3 | Cancelamento de consulta pelo cliente | Evidente | Alta |  |
| RFUN4 | Lembretes de consulta | Evidente | Média |  |
| RFUN5 | Aviso de pagamento | Evidente | Alta |  |
| RFUN6 | Check-in de consulta | Evidente | Média |  |
| RFUN7 | Criação de prontuário | Evidente | Alta |  |
| RFUN8 | Edição do prontuário | Evidente | Alta |  |
| RFUN9 | Exportação do prontuário | Evidente | Alta |  |
| RFUN10 | Emitir cobranças | Evidente | Alta |  |
| RFUN11 | Emissão de recibo | Evidente | Média |  |
| RFUN12 | Acompanhamento de inadimplência | Oculto | Alta |  |
| RFUN13 | CRUD de funcionários | Evidente | Alta |  |
| RFUN14 | Agendamento de consulta pela secretaria | Evidente | Alta |  |
| RFUN15 | Cancelamento de consulta pela secretaria | Evidente | Alta |  |
| RFUN16 | Controle despesas | Evidente | Alta |  |
| RFUN17 | Visualização de agenda | Evidente | Alta |  |
| RFUN18 | Edição de agenda | Evidente | Alta |  |
| RFUN19 | Fluxo de caixa | Evidente | Alta |  |
| RFUN20 | Emissão de relatórios financeiros em períodos regulares | Evidente | Alta |  |
| RFUN21 | Previsão Financeira | Evidente | Alta |  |
| RFUN22 | Avaliação do profissional por parte do paciente | Evidente | Média |  |
| RFUN23 | Visualização dos dados de avaliações | Evidente | Média |  |
| RFUN24 | Criação de desconto com base em condições | Evidente | Média |  |
| RFUN25 | Alerta de desconto | Evidente | Média |  |
| RFUN26 | Exportação de dados do paciente | Evidente | Média |  |
| RFUN27 | Teleatendimento | Evidente | Baixa |  |
| RFUN28 | Lista de espera | Evidente | Média |  |
| RFUN29 | Integração com WhatsApp | Evidente | Alta |  |
| RFUN30 | Integração com Google Agenda | Evidente | Baixa |  |
| RFUN31 | Criação de logs de acessos ao prontuário | Oculto | Alta |  |
| RFUN32 | Criação de logs de agendamento | Oculto | Média |  |
| RFUN33 | Visualização de logs | Evidente | Alto |  |
| RFUN34 | Termo de consentimento digital | Evidente | Médio |  |
| RFUN35 | Retenção de dados do paciente por no mínimo 20 anos | Oculto | Alto |  |
| RFUN36 | Emissão de relatórios de atendimentos em períodos regulares | Evidente | Médio |  |
| RFUN37 | Integração com sistema de pagamento | Evidente | Baixo |  |
| RFUN38 | Acesso a suporte | Evidente | Baixa |  |
| RFUN39 | Validação do contrato digital pelo paciente | Evidente | Alta |  |
| RFUN40 | Armazenamento do contrato legal | Evidente | Alta |  |
| RFUN41 | Lembrete de multas caso cancelamento segundo regras de negócio | Evidente | Alta |  |
| RFUN42 | Armazenar histórico de versões do prontuário | Oculto | Alta |  |
| RFUN43 | Bloqueio de agenda | Oculto | Alta |  |
| RFUN44 | Emissão de relatórios de cancelamentos em períodos regulares | Evidente | Médio |  |
| RFUN45 | Formulário personalizável | Evidente | Alto |  |

# Requisitos e restrições não funcionais

Elaborar uma lista de todos os requisitos não funcionais. Considerar requisitos de informação, de interface, de projeto, de arquitetura de software, de plataforma de hardware, de plataforma de software, de plataforma de comunicação, de desempenho, de disponibilidade, de segurança, de manutenibilidade, de portabilidade e de documentação. A lista poderá ser dividida por tipo de requisito, mas é importante que os requisitos tenham uma identificação única para que possam ser referenciados sem ambigüidades no futuro.

## Requisitos e restrições de informação (RINF)

Elaborar uma lista de todas as necessidades de informação que o software não pode deixar de atender. Esta lista deverá ser classificada em informações cadastrais e informações gerenciais.

|  |  |  |  |
| --- | --- | --- | --- |
| **Ref.** | **Tipo** | **Descrição** | **Casos de Uso** |
| RINF1 | Cadastral | Administrador: CPF, nome, e-mail, telefone, nome da clínica, logo da clínica, endereço da clínica. |  |
| RINF2 | Cadastral | Secretário(a): CPF, nome, salário, conta bancária, endereço, telefone. |  |
| RINF3 | Cadastral | Psicólogo(a): CPF, nome, CIP, salário, conta bancária, endereço, telefone, Agenda, Lista de Espera, :Avaliações. |  |
| RINF4 | Cadastral | Cliente: CPF, nome, endereço, telefone, e-mail, :Prontuário. |  |
| RINF5 | Cadastral | Consulta: :Psicólogo(a), :Cliente, data/hora, confirmação, presença, cancelada, :Secretário(a) (opcional). |  |
| RINF6 | Cadastral | Pagamento: :Cliente, :Consulta, quantia base, multa, desconto, data/hora, forma de pagamento, recibo. |  |
| RINF7 | Cadastral | Prontuário: avaliação da demanda, objetivos do tratamento, registros de evolução das sessões, procedimentos técnicos e científicos adotados, encaminhamentos e encerramento do caso, e cópias de documentos complementares.[1]  Log de Acesso, :Funcionario (Quem criou esta versão), :Prontuário (versão anterior). |  |
| RINF8 | Cadastral | Despesa: Tipo, quantia, recibo. |  |
| RINF9 | Cadastral | Avaliação: :Cliente, nota, descrição |  |
| RINF10 | Cadastral | Desconto: valor, condições. |  |
| RINF11 | Cadastral | Contrato Legal: Cláusulas. |  |
| RINF12 | Gerencial | Agrupamento de despesas e rendas em certos períodos regulares de tempo para geração de relatórios financeiros. |  |
| RINF13 | Gerencial | Agrupamento de avaliações de um(a) psicólogo(a) para geração de nota geral. |  |
| RINF14 | Gerencial | Agrupamento de notas e consultas de um(a) psicólogo(a) para geração de relatório de performance do profissional. |  |

Estes requisitos de informação são importantes para verificar a qualidade da modelagem de dados que for feita.

## 5.2 Requisitos e restrições de interface Homem-Computador (RHIC)

Definir todos os aspectos de Interface Homem Computador (IHC) incluindo: conteúdo de informações, fatores ergonômicos, dispositivos de interação, formato de apresentação, tipo de diálogo, e mecanismos de ajuda alocados a cada perfil/grupo/tarefa de usuário. Descrever, em particular, os requisitos de usabilidade para cada perfil/grupo/tarefa de usuário. Por exemplo, pode-se definir como requisito que as opções de *menu* do sistema tenham teclas de atalho associadas..

É recomendável definir diagramas de interface (telas e relatórios) para as funções previstas para o software. Devem ser estabelecidas, no diagrama, as áreas da janela ou do relatório destinadas a cada tipo de informação.

Existem diversos padrões para definição de interface de usuário. Exemplos destes padrões predefinidos são *Motif* e *Windows*. Esses padrões de interface definem desde diretivas para diagramação até os tipos de objetos de interface que podem ser adotados, com seus respectivos atributos e valores default.

No caso de se adotar algum padrão predefinido de interface, pode-se apenas referenciá-lo nesta seção, já que a diagramação de todas as janelas e relatórios deve seguir este padrão. Caso contrário, para cada área funcional identificada na diagramação, deve-se especificar a sua finalidade, suas dimensões e seu posicionamento relativo na janela ou relatório. Restrições sobre o tamanho e posição relativos entre as diversas áreas do diagrama também devem ser especificadas.

Cada área do diagrama pode ser recursivamente subdivida em áreas menores. Nestes casos devem ser especificadas as mesmas informações definidas para as áreas principais, ou seja, a identificação, a finalidade, o tamanho e o posicionamento relativo.

Exemplos típicos de áreas funcionais de janelas incluem: área de mensagens; área de comandos; área de respostas a comandos; área de desenho; área de menus; e área de identificação da Janela. As áreas típicas de composição de um relatório são: cabeçalhos, corpo, linhas de detalhe, linhas de totalização, e rodapé.

*Exemplo:*

|  |  |  |
| --- | --- | --- |
| **Ref.** | **Descrição** | **Casos de Uso** |
| RIHC1 | Para facilitar a usabilidade na transação de venda, pede-se que a tela de vendas tenha uma fonte (tipo e tamanho de letra) que permita uma fácil visualização a uma distância de 2 metros do monitor porque desta forma o cliente poderá visualizar as informações da venda da sua posição. | Todos |

## 5.3 Requisitos de Interface Externa (RIEX)

Identificar e descrever as interfaces com outros softwares/sistemas que o software deverá prover. Por exemplo, um software comercial deve gerar informações para o Sistema de Arrecadação da Secretaria da Fazenda Estadual. O formato dessas informações e o protocolo de envio são definidos pela própria secretaria, e atender essas definições é um requisito do software.

|  |  |  |
| --- | --- | --- |
| **Ref.** | **Descrição** | **Casos de Uso** |
| RIEX1 | O software deve exportar os dados do usuário (incluindo o prontuário) em formato JSON, pois esse é o formato de troca de dados padrão entre aplicações Web. |  |
| RIEX2 | Em caso de exportação das consultas, o formato deve ser .ics (Icalendar), pois esse é o formato universal de eventos em calendários.[2] |  |

## 5.4 Requisitos e Restrições de Projeto (RPRO)

Nesta seção serão especificados todos os requisitos e restrições associados a condução do projeto de desenvolvimento e que podem limitar ou definir ações que serão executadas.

DNK 14/10/2025: O que eu coloco aqui?

*Exemplo:*

|  |  |  |
| --- | --- | --- |
| **Ref.** | **Descrição** | **Casos de Uso** |
| RPRO1 | O cliente solicitou que o módulo de contabilidade fosse entregue até o dia 10 do último mês do ano corrente (10/12/05), para testes em ambiente real. | CSUi, CSUj, CSUk |

## 5.5 Requisitos e restrições de arquitetura de software (RARQ)

Se o software tiver de ser desenvolvido em uma arquitetura específica, então essa arquitetura deverá ser descrita.

|  |  |  |
| --- | --- | --- |
| **Ref.** | **Descrição** | **Casos de Uso** |
| RARQ1 | O software deverá ser desenvolvido com uma arquitetura de cliente-servidor, pois será uma aplicação Web. | Todos |

## 5.6 Requisitos e restrições de plataforma de hardware (RPHW)

Identificar e descrever requisitos e restrições relacionadas com a plataforma de hardware que será utilizada pelo software:

|  |  |  |
| --- | --- | --- |
| **Ref.** | **Descrição** | **Casos de Uso** |
| RPHW1 | O front-end do software deve ser capaz de rodar em qualquer navegador moderno, em qualquer computador capaz de executar esses navegadores. |  |

## 5.7 Requisitos e restrições de plataforma de software (RPSW)

Se o software tiver que ser executado em plataformas de software específicas, essas plataformas de software deverão ser definidas:

* Sistema Operacional: identificar e descrever o sistema operacional em que o software deverá ser executado;
* Softwares Básicos: identificar SGBD, linguagem de programação, ferramentas CASE e outros.

Se houver mais de uma plataforma de software, deve-se especificar qual a plataforma principal e em que situações as outras plataformas podem ser utilizadas.

|  |  |  |
| --- | --- | --- |
| **Ref.** | **Descrição** | **Casos de Uso** |
| RPSW1 | O software deverá usar o SGBD PostgreSQL, pois esse é o único que a equipe de desenvolvimento tem experiência. | Todos |
| RPSW2 | O front-end do software usará o stack padrão para desenvolvimento Web (HTML, CSS e Javascript). |  |
| RPSW3 | A lógica do back-end será implementada em PHP, com ferramentas adicionais escritas em C/C++ caso houver necessidade. |  |
| RPSW4 | O back-end deve ser capaz de rodar em máquinas Windows e em qualquer distribuição do Linux com glibc. |  |

## 5.8 Requisitos e restrições de desempenho (RDES)

Identificar e descrever os requisitos e restrições de desempenho do software.

|  |  |  |
| --- | --- | --- |
| **Ref.** | **Descrição** | **Casos de Uso** |
| RDES1 | A interface web não deve apresentar queda de performance em dispositivos suportados. | Todos |
| RDES2 | O tempo de resposta máximo permitido para requisições é de 2 segundos |  |

## 5.9 Requisitos e restrições de disponibilidade (RDIS)

Especificar os requisitos de disponibilidade necessários para o software de uma forma global:

* Período de disponibilidade: horário comercial, 24 horas por dia, etc.
* Período máximo para recuperação do software em caso de falha.

Devem ser definidos os tipos de falha e a tolerância aceitável para cada tipo de falha. Os tipos de falha podem ser definidos em função dos requisitos funcionais e de dados, mas não se restringem a estes. Por exemplo: a função “Registrar Venda” deve ter um tempo para recuperação de falha de no máximo uma hora (o que significa que esta função não poderá ficar mais do que uma hora indisponível para o usuário em nenhuma circunstância).

|  |  |  |
| --- | --- | --- |
| **Ref.** | **Descrição** | **Casos de Uso** |
| RDIS1 | O software deverá estar disponível 24 horas por dia | Todos |
| RDIS2 | O software pode ter downtime anual máximo de 1% | Todos |
| RDIS3 | O software não pode ficar mais que 4 horas consecutivas fora do ar | Todos |

## 5.10 Requisitos e restrições de segurança (RSEG)

Especificar os requisitos de segurança necessários para controle de acesso ao software. Definir a necessidade de:

* Verificação de senha;
* Criptografia de dados;
* Registro das operações efetuadas;
* Habilitação de funções por perfil de usuário;
* Acesso seletivo aos dados e funções.

*Exemplo:*

|  |  |  |
| --- | --- | --- |
| **Ref.** | **Descrição** | **Casos de Uso** |
| RSEG1 | Os prontuários dos clientes devem ser anonimizados quando em armazenamento. |  |
| RSEG2 | Dados de pessoas cadastradas devem ser criptografados em armazenamento. |  |
| RSEG3 | Prontuários devem ser exportados sem os dados do paciente. |  |
| RSEG4 | Psicólogos não podem ver as agendas de outros psicólogos. |  |

## 5.11 Requisitos e restrições de manutenibilidade (RMAN)

Especificar os requisitos que visam facilitar a manutenção posterior do software, tais como:

* Requisitos de reutilização (exemplo: uso de implementação orientada a objetos; bibliotecas de classes e padrões de projeto);
* Requisitos de modularização (exemplo: valores para métricas de acoplamento entre módulos; máximo de pontos de função por módulo);
* Requisitos de configuração (exemplo: regras para controle de versões);
* Requisitos de documentação (exemplo: documentação de programa)

*Exemplo:*

|  |  |  |
| --- | --- | --- |
| **Ref.** | **Descrição** | **Casos de Uso** |
| RMAN1 | O sistema irá usar o módulo Bootstrap para o front-end | Todos |

## 5.12 Requisitos e restrições de portabilidade (RPOR)

Identificar as diversas plataformas de software e hardware com as quais o software deve ser compatível. Devem ser consideradas tanto plataformas de desenvolvimento como plataformas de produção. Outros exemplos de requisitos de portabilidade são:

* Percentual de componentes que podem ter código dependente da plataforma hospedeira;
* Percentual de código que pode ser dependente da plataforma hospedeira;
* Uso de uma linguagem reconhecidamente portável.

|  |  |  |
| --- | --- | --- |
| **Ref.** | **Descrição** | **Casos de Uso** |
| RPOR1 | O back-end do software deve suportar apenas Windows 11 e distribuições modernas do Linux. | Todos |

## 5.13 Requisitos de documentação (RDOC)

Especificar os requisitos de documentação do produto de software que será desenvolvido.

*Exemplo:*

|  |  |  |
| --- | --- | --- |
| **Ref.** | **Descrição** | **Casos de Uso** |
| RDOC1 | Um tutorial da interface deve ser oferecido quando o usuário abrir a aplicação pela primeira vez |  |
| RDOC2 | O software deve prover ajuda on-line sensível ao contexto em todas as suas telas | Todos |

# Requisitos Futuros (RFUT)

Descrever os requisitos que poderão ser especificados em uma nova versão do produto.

*Exemplo:*

|  |  |  |
| --- | --- | --- |
| **Ref.** | **Descrição** | **Casos de Uso** |
| RFU1 | Em um futuro próximo o software de atendimento de clientes deverá ser integrado com o software do sistema de faturamento para que o atendente possa identificar o perfil de negócios do cliente | CSUx |

# Referências cruzadas complementares

Nesta seção são colocadas algumas referências cruzadas que podem ajudar o rastreamento futuro dos requisitos. Estes mapeamentos podem ser feitos em forma de matrizes de rastreabilidade como mostram os exemplos a seguir:

|  |  |
| --- | --- |
| **Requisitos Funcionais** | **Requisitos não funcionais vinculados** |
| *Colocar a identificação do requisito funcional* | *Colocar a identificação do requisito não funcional vinculado* |

|  |  |
| --- | --- |
| **Caso de Uso** | **Requisitos funcionais vinculados** |
| *Colocar a identificação do caso de uso* | *Colocar a identificação do requisito vinculado* |

|  |  |
| --- | --- |
| **Requisito Funcional** | **Origem do Requisito** |
| *Colocar a identificação do requisito funcional* | *Colocar a origem do requisito. Pode ser uma entrevista, um questionário, ou outra técnica qualquer de elicitação de requisito aplicada.* |

|  |  |
| --- | --- |
| **Requisito Não Funcional** | **Origem do Requisito** |
| *Colocar a identificação do requisito não funcional* | *Colocar a origem do requisito. Pode ser uma entrevista, um questionário, ou outra técnica qualquer de elicitação de requisito aplicada.* |

|  |  |
| --- | --- |
| **Caso de Uso** | **Origem do caso de uso** |
| *Colocar a identificação do requisito funcional* | *Colocar a origem do caso de uso. Pode ser uma entrevista, um questionário, ou outra técnica qualquer de elicitação de requisito aplicada.* |

Observações:

1. Os exemplos acima não esgotam as possibilidades de referências cruzadas.
2. Sugere-se que as referências sejam feitas usando *hiperlinks* sempre que possível.

# Aprovação Formal

O EOR deve ser datado e assinado de acordo com a definição contida no MPD. O documento aprovado se torna uma *baseline* do projeto que passa a ser controlada pela Gerência de Configuração de Software.

# Bibliografia

Detalhar todas as fontes de informação citadas no documento ou usadas de alguma forma para sua elaboração, incluindo título, autor, data, fonte de obtenção, órgão responsável pela publicação e qualquer outra informação que possa facilitar a localização e obtenção dessas fontes pelo leitor. Nesta seção cada fonte citada deve ser associada a um número identificador. Junto de cada citação deve ser colocada também uma explicação sucinta de como a fonte foi usada na elaboração do documento.

Referenciar todas as fontes de informações citadas no EOR, explicando o motivo de sua citação. Devem ser incluídos, por exemplo:

* Normas e Padrões utilizados;
* Livros e artigos (autores, título, editora, páginas, volume, data e local da publicação, ISBN/ISSN, etc);
* Manuais de outros Produtos.

1- [Prontuário psicológico: O que é, como fazer e o que deve conter](https://portaltelemedicina.com.br/prontuario-psicologico) – Portal Telemedicina

2- [Import events to Google Calendar](https://support.google.com/calendar/answer/37118?hl=en) – Suporte Google

# 10. Anexos

Anexar ao EOR documentos considerados importantes para sua compreensão. Por exemplo, documentos e leis usadas na especificação dos requisitos podem ser colocados nos anexos para facilitar o acesso do leitor às fontes dos requisitos do software. Cada anexo deverá receber uma folha de rosto indicando seu conteúdo.

!!!!!!!!!!!!!!!!!!!!!!REQUISITO DESCOBERTO

VISUALIZAÇÃO DE CONTRATO.

EDIÇÃO DE CONTRATO.