



DOCUMENTAÇÃO DE SISTEMA

Sistema Eletrônico de Sorteio — EDITAL DE CREDENCIAMENTO Nº
001/2025 – SVMA/UMAPAZ

1. Identificação do Sistema

Campo	Informação
Órgão	UMAPAZ — Coordenação de Educação Ambiental e Cultura de Paz
Sistema	Sistema Eletrônico de Sorteio para Edital de Credenciamento
Plataforma	Google Apps Script (GAS) + Google Sheets + Interface Web HTML/JS
Linguagem	JavaScript (backend GAS) e HTML/CSS/JavaScript (frontend)
Repositório de dados	Google Sheets — abas: 'sorteio' (origem) e 'resultado sorteio' (saída)
Controle de acesso	Autenticação por senha verificada no servidor (backend GAS)

2. Objetivo do Sistema

O sistema tem por finalidade realizar sorteios eletrônicos de participantes habilitados em processos de credenciamento público, organizados por linha programática (categoria). O resultado é registrado automaticamente em duas planilhas dedicadas, com ranking por linha programática, CPF, nome e timestamp de conclusão, garantindo rastreabilidade e transparência.

O sistema foi desenvolvido para substituir sorteios manuais, eliminando subjetividade e conferindo auditabilidade ao processo, em conformidade com os princípios da Administração Pública de impessoalidade, moralidade e publicidade.

3. Arquitetura e Componentes

3.1 Componentes Principais

Componente	Arquivo	Responsabilidade
Backend	Código .gs (Apps Script)	Leitura da planilha, validação de senha, cálculo de inscritos/habilitados, embaralhamento e persistência dos resultados
Frontend	pagina.html	Interface visual, animações de sorteio, exibição de resultados, histórico de sessão e comunicação com o backend via google.script.run
Dados de entrada	Aba 'sorteio'	Contém CPF (col. A), Nome (col. B), Status (col. F) e Linha Programática (col. N) dos participantes inscritos
Dados de saída	Aba 'resultado sorteio' na planilha 1 e planilha 2	Registra linha programática, ranking, nome e CPF de cada sorteado; linha 128/col. B recebe o timestamp de início e linha 129/col. B recebe o timestamp de conclusão

3.2 Fluxo de Execução

1. O operador acessa a URL pública do sistema via navegador.
2. Um modal de autenticação exige senha antes de qualquer interação. A senha é verificada no servidor (backend GAS) — nunca exposta no HTML.
3. Após autenticação, o sistema carrega automaticamente as linhas programáticas únicas diretamente da coluna N da aba 'sorteio'.
4. O operador aciona 'Sortear linhas programáticas': uma contagem regressiva de 5 segundos é exibida e uma animação percorre as linhas programáticas disponíveis, acelerando e desacelerando aleatoriamente.
5. Após o início do sorteio da primeira linha programática, o timestamp de início é gravado na célula B128 da aba 'resultado sorteio' em ambas planilhas.
6. A linha programática sorteada é destacada na interface e o sistema busca no backend os participantes habilitados dessa linha programática.
7. O operador aciona 'Iniciar sorteio de participantes': cada nome é sorteado individualmente com animação, exibindo um popup de confirmação a cada resultado.
8. Ao final de cada linha programática, o resultado completo (ordem de classificação) é salvo automaticamente nas abas 'resultado sorteio' com CPF, ranking, nome e linha programática.
9. Após o sorteio de todas as linhas programáticas, o timestamp de conclusão é gravado na célula B129 da aba 'resultado sorteio' em ambas planilhas.

4. Mecanismo de Aleatoriedade

O sorteio utiliza dois níveis de aleatoriedade, ambos baseados na função `Math.random()` nativa do JavaScript:

Etapa	Descrição
Sorteio de linha programática	A animação percorre aleatoriamente as linhas programáticas disponíveis em 20 iterações com velocidade variável. A linha programática destacada na última iteração é a sorteada.
Pré-embaralhamento	Antes da animação de nomes, cada participante habilitado recebe um número aleatório (<code>Math.random()</code>). A lista é ordenada por esse número, garantindo que a sequência de sorteio seja imprevisível desde o início.
Sorteio de nomes	A animação percorre 18 iterações sobre os nomes ainda não sorteados, com desaceleração progressiva. O nome destacado na última iteração é declarado vencedor da posição.

Nota técnica: `Math.random()` é um gerador pseudoaleatório (PRNG) baseado no algoritmo interno do motor V8 do Google (xorshift128+). Para fins de credenciamento público, este nível de aleatoriedade é considerado suficiente, pois: (a) a semente é baseada em entropia do sistema operacional e horário de execução; (b) o resultado é imprevisível para todos os envolvidos, inclusive o operador.

5. Controle de Acesso e Segurança

Mecanismo	Implementação
Autenticação por senha	A senha é inserida pelo operador e enviada ao backend GAS via <code>google.script.run</code> . A comparação ocorre exclusivamente no servidor. O HTML jamais contém ou exibe a senha correta.
Bloqueio de interface	Um modal de sobreposição (z-index 999) bloqueia toda a interface até a autenticação ser confirmada, impedindo interações não autorizadas.
Proteção contra erros	O botão de confirmação é desabilitado durante a verificação, prevenindo múltiplas tentativas simultâneas. Mensagens de erro são exibidas sem revelar detalhes sobre a senha.
Dados sensíveis	CPFs dos participantes são lidos diretamente da planilha no momento do sorteio e gravados na aba de resultado. Não são armazenados em memória além do necessário para a operação.

6. Descrição das Funções do Backend (.gs)

Função	Descrição
<code>doGet(e)</code>	Ponto de entrada da aplicação web. Carrega o template HTML e o serve como página pública com permissão de <code>iframe</code> .
<code>verificarSenha(senha)</code>	Recebe a senha digitada pelo operador e compara com o valor armazenado no servidor. Retorna booleano. A senha nunca trafega de volta ao cliente.
<code>getCategorias()</code>	Lê a coluna N da aba 'sorteio' e retorna array de linhas programáticas únicas, filtrando valores vazios ou inválidos. Elimina dependência de intervalos hardcoded.

getParticipantesDaCategoria(categoria)	Para a linha programática recebida, lê colunas A:N, conta inscritos totais e habilitados, filtra por status 'habilitado(a)' na col. F, embaralha e retorna a lista com contagens.
salvarResultado(categoria, participantes)	Grava na aba 'resultado sorteio' as colunas: Linhas Programática (A), Ranking (B), Nome (C) e CPF (D). Busca a primeira linha vazia a partir da linha 2 para não sobrescrever dados existentes.
salvarTimestampInicio()	Grava data e hora de início do sorteio na célula B128 da aba 'resultado sorteio em abas as planilhas', no formato dd/MM/yyyy HH:mm:ss, usando o fuso horário do script.
salvarTimestampFinal()	Grava data e hora de conclusão do sorteio na célula B129 da aba 'resultado sorteio' em ambas as planilhas, no formato dd/MM/yyyy HH:mm:ss, usando o fuso horário do script.

7. Mapeamento de Colunas da Planilha

7.1 Aba 'sorteio' (entrada)

Coluna	Campo	Uso no sistema
A	CPF	Gravado na aba de resultado para identificação do participante sorteado
B	Nome	Exibido na animação e gravado no resultado
C-E	(não utilizadas)	Ignoradas pelo sistema
F	Status	Filtragem: apenas registros contendo 'habilitado(a)' são incluídos no sorteio
G-M	(não utilizadas)	Ignoradas pelo sistema
N	Linha Programática	Categoria do sorteio. Usada para agrupar participantes e gerar a lista de categorias disponíveis

7.2 Aba 'resultado sorteio' (saída)

Coluna/Célula	Campo	Descrição
A	Linha Programática	Nome da linha programática sorteada
B	Ranking	Posição do participante dentro da categoria (1º, 2º, 3º...)
C	Nome	Nome completo do participante sorteado
D	CPF	CPF do participante, vinculado pelo nome via dicionário interno
B128 – B129	Timestamp	Data e hora de início e de conclusão do sorteio de todas as categorias (formato: dd/MM/yyyy HH:mm:ss)

8. Registro e Rastreabilidade

O sistema garante rastreabilidade por meio dos seguintes mecanismos:

- Persistência automática: cada categoria sorteada tem seus resultados gravados imediatamente na aba 'resultado sorteio', sem intervenção manual do operador.
- Identificação única: o CPF de cada participante sorteado é vinculado ao resultado, permitindo identificação inequívoca.
- Ordenação por ranking: a posição de cada participante dentro da linha programática é registrada numericamente, refletindo a ordem exata do sorteio.
- Timestamp de início: a data e hora de início do processo são gravadas em célula fixa (B128), criando um marco temporal verificável.
- Timestamp de conclusão: a data e hora de finalização do processo são gravadas em célula fixa (B129), criando um marco temporal verificável.
- Histórico de sessão: durante a execução, o operador pode consultar o histórico completo de todos os sorteios realizados na sessão via interface visual.

9. Limitações e Observações Técnicas

Sessão sem persistência: O histórico exibido na interface existe apenas durante a sessão ativa do navegador. Ao recarregar a página, o histórico visual é perdido (os dados na planilha permanecem).

10. Declaração de Conformidade

O sistema descrito neste documento foi desenvolvido com o objetivo de assegurar isonomia, transparência e impessoalidade no processo de credenciamento público conduzido pela UMAPAZ. O mecanismo de sorteio eletrônico substitui procedimentos manuais, reduzindo a possibilidade de interferência humana no resultado.

11. Equipe de Desenvolvimento

Este sistema foi desenvolvido pela Secretária Escolar da UMAPAZ:

- Carlos Alexandre Petito (servidor)
- Felipe da Silva Freitas (estagiário)
- José Eduardo Campos (estagiário)
- Roberto Couto Fernandes Mourão (servidor)

O sistema teve auxílio parcial de ferramentas de Inteligência Artificial.