

MZGrid

Powering the Intelligence Economy.

Whitepaper

Um marketplace aberto de computação em GPU na Base, em que cada hora de GPU é precificada, mantida em custódia, liquidada e colateralizada no token MZG.

Versão 1.2 · novembro de 2026

MZGrid Labs

Aviso importante

Este whitepaper descreve o design do protocolo MZGrid e do token MZG na data de sua publicação. Ele é fornecido apenas para fins informativos. Não constitui prospecto, oferta ou solicitação de venda ou compra de qualquer token ou valor mobiliário, nem aconselhamento de investimento, jurídico, tributário ou financeiro. O MZG é um token utilitário usado para pagar e fornecer computação na rede MZGrid. Nada neste documento constitui promessa de preço, rendimento, liquidez, listagem em corretora ou retorno de qualquer natureza. Os parâmetros aqui descritos podem mudar antes ou depois do lançamento; os smart contracts implantados e a documentação publicada prevalecem. Declarações sobre planos futuros são intenções, não compromissos, e dependem de fatores técnicos, comerciais e regulatórios fora do controle dos autores. Leia a Seção 12 (Riscos) antes de interagir com o protocolo. A aquisição ou a posse de MZG pode estar sujeita a restrições na sua jurisdição. Este documento é uma tradução do original em inglês; em caso de qualquer divergência, prevalece o original em inglês.

Sumário

1. Resumo
2. O problema: a computação é o gargalo
3. O MZGrid em resumo
4. Arquitetura
5. Ciclo de vida do job, custódia e liquidação
6. Colateral do provedor: o bloqueio de 20%
7. Verificação, reputação e agendamento
8. O token MZG
9. Preços
10. Modelo de segurança e confiança
11. Roadmap e status atual
12. Riscos

Apêndice A. Parâmetros do protocolo

Apêndice B. Interface dos contratos

Apêndice C. Glossário

1 Resumo

A inteligência artificial roda em GPUs, e a capacidade de GPU é cara, escassa e concentrada em um punhado de provedores de nuvem. Ao mesmo tempo, um grande número de GPUs capazes — em data centers independentes, estúdios de renderização, laboratórios e residências — fica ocioso durante boa parte do dia. O MZGrid é um marketplace aberto que conecta esses dois lados: **provedores**, que contribuem com máquinas de GPU verificadas, e **clientes** — desenvolvedores, startups, pesquisadores e agentes autônomos de IA —, que as alugam por segundo para executar endpoints de inferência, fine-tuning, geração de imagens e jobs em lote.

O MZGrid separa a **execução** da **economia**. As cargas de trabalho rodam off-chain em contêineres isolados no hardware dos provedores, coordenadas por um scheduler que associa jobs a máquinas com base em preço, confiabilidade e localização. O dinheiro nunca passa pelo operador: todo job é pago por meio de um contrato de custódia imutável na Base. O cliente deposita MZG antes de o job começar; quando o job termina, o contrato paga ao provedor os segundos efetivamente usados, envia uma taxa do protocolo de 7% para a tesouraria e reembolsa o restante ao cliente.

Em vez de staking, o MZGrid forma o colateral dos provedores a partir do trabalho efetivamente realizado: **20% de cada pagamento ao provedor fica bloqueado** no registro de provedores e só pode ser sacado quando o provedor deixa a rede, após um **prazo de saída de 10 dias**. Um provedor que abandona jobs, fica offline no meio de um job ou falha na verificação tem seu bloqueio penalizado, e metade do valor penalizado compensa o cliente afetado. Um limite por job vinculado ao bloqueio permite que novos provedores comecem com jobs pequenos e conquistem acesso a jobs maiores à medida que seu colateral cresce.

O MZG é a moeda única da rede. Os preços são cotados em MZG por hora de GPU; não há opção de pagamento em stablecoin, nem oráculo de preços, nem mecânica de token de governança. A oferta é fixa em 1,000,000,000 MZG, emitidos uma única vez, com vesting aplicado on-chain.

O MZGrid foi lançado como **beta na mainnet da Base em 17 de novembro de 2026**. Esta versão do documento descreve o protocolo tal como foi implantado e o estado da rede em novembro de 2026 (Seção 11).

Somente MZG	7%	20%	10 dias
Moeda única de pagamento e liquidação	Taxa do protocolo, incluída no preço	De cada pagamento ao provedor fica bloqueado como colateral	Prazo de saída até a liberação do bloqueio

2 O problema: a computação é o gargalo

2.1 Oferta concentrada

Um pequeno número de nuvens de hiperescala controla a maior parte das GPUs de data center disponíveis para aluguel. O acesso é condicionado a cotas, contratos de longo prazo e compromissos mínimos que favorecem grandes clientes. Desenvolvedores independentes, equipes pequenas e pesquisadores enfrentam listas de espera, escassez em determinadas regiões e preços que não podem negociar.

2.2 Capacidade ociosa

Fora dos hiperescaladores existe uma capacidade de GPU considerável — placas de workstation, pequenos clusters de GPU, render farms, máquinas de universidades, antigos rigs de mineração — cuja utilização é baixa e irregular. Seus donos não têm uma forma simples e confiável de vender horas ociosas a desconhecidos: precisariam encontrar clientes, precificar seu hardware, receber os pagamentos e se proteger contra a inadimplência.

2.3 Falta de uma camada de confiança

Um marketplace entre desconhecidos precisa responder a três perguntas que as ofertas de GPU peer-to-peer existentes muitas vezes deixam em aberto:

- **Vou receber?** Os provedores precisam ter a garantia de que os fundos do cliente existem antes de iniciar um job.
- **Vou receber o que paguei?** Os clientes precisam ter a garantia de que a GPU é o modelo anunciado, de que ela permanece online e de que pagam apenas pelo tempo efetivamente entregue.
- **O que acontece quando algo dá errado?** Os dois lados precisam de um desfecho previsível e automático quando um job falha, em vez de um chamado de suporte.

2.4 Fragmentação

Preços, descrições de hardware e alegações de desempenho não são padronizados entre os vendedores, o que torna a comparação lenta e sujeita a erros. Os agentes de IA — uma nova classe de cliente que precisa adquirir computação de forma programática — não têm uma interface uniforme para comprá-la.

> **Objetivo de design.** Tornar o aluguel de uma GPU de um provedor desconhecido tão seguro e previsível quanto o aluguel em uma grande nuvem, garantindo o pagamento e a responsabilização por meio de código, e não de contratos e centrais de suporte.

3 O MZGrid em resumo

3.1 Participantes

Participante	Papel	O que obtém
Provedores de GPU	Executam o agente do provedor em suas máquinas, publicam ofertas e executam jobs	Receita em MZG com hardware que, de outra forma, ficaria ocioso; acesso a jobs maiores à medida que o bloqueio cresce
Clientes	Desenvolvedores, startups, pesquisadores e empresas implantam cargas de trabalho	Cobrança por segundo, hardware verificado, reembolsos automáticos, sem contratos
Agentes de IA	Software que adquire computação pela API e paga em MZG	Acesso programático e permissionless a GPUs
Coordenador	Serviço off-chain que faz o matching, despacha, monitora e verifica jobs e envia as liquidações	Operado pela MZGrid Labs; limitado pelos contratos (Seção 10)
Tesouraria	Multisig que recebe as taxas do protocolo e metade das penalizações	Financia infraestrutura, segurança, incentivos e grants

3.2 Produtos

Produto	Descrição
MZGrid Marketplace	Descubra, compare e alugue ofertas de GPU verificadas; implante a partir de templates; gerencie jobs em execução, endpoints e logs.
MZGrid Provider	O agente `mzgrid-provider`: detecção de GPUs, benchmark, registro, execução de jobs em contêineres isolados, relatório de uso, saída e saque.
MZGrid Developer API	API REST e SDK TypeScript para buscar ofertas, criar e financiar jobs, transmitir logs e parar jobs; endpoints de inferência compatíveis com OpenAI.
MZGrid Intelligence	Mecanismo de matching e verificação: pontuação de ofertas, benchmark de hardware, monitoramento de uso e cálculo de reputação.
MZGrid Protocol	Smart contracts na Base: token MZG, custódia, registro de provedores (bloqueio, saída e penalização) e vesting.

3.3 Cargas de trabalho suportadas

Inicialmente, a rede se concentra em cargas de trabalho que rodam em uma única máquina e são fáceis de verificar:

Template	Tipo	VRAM mín.
Endpoint vLLM compatível com OpenAI (qualquer modelo do Hugging Face)	Endpoint de inferência	16 GB
Llama 3.1 8B Instruct	Endpoint de inferência	24 GB
Llama 3.3 70B Instruct FP8	Endpoint de inferência	80 GB
Qwen2.5 32B Instruct AWQ	Endpoint de inferência	24 GB
ComfyUI + SDXL / FLUX.1-schnell	Geração de imagens	16 GB
Whisper large-v3	Endpoint de transcrição	10 GB
PyTorch + CUDA JupyterLab	Interativo	8 GB
Imagem Docker personalizada / job em lote	Personalizado	qualquer

O treinamento distribuído entre máquinas não relacionadas está deliberadamente fora do escopo das primeiras fases: ele exige interconexões de alta largura de banda e um modelo de verificação diferente. Trata-se de um item de pesquisa do roadmap.

4 Arquitetura

O MZGrid é um sistema híbrido. A blockchain é usada para aquilo em que é boa — guardar e movimentar dinheiro sob regras fixas e registrar compromissos —, enquanto a execução em GPU, o agendamento e o monitoramento acontecem off-chain, onde são rápidos e baratos.

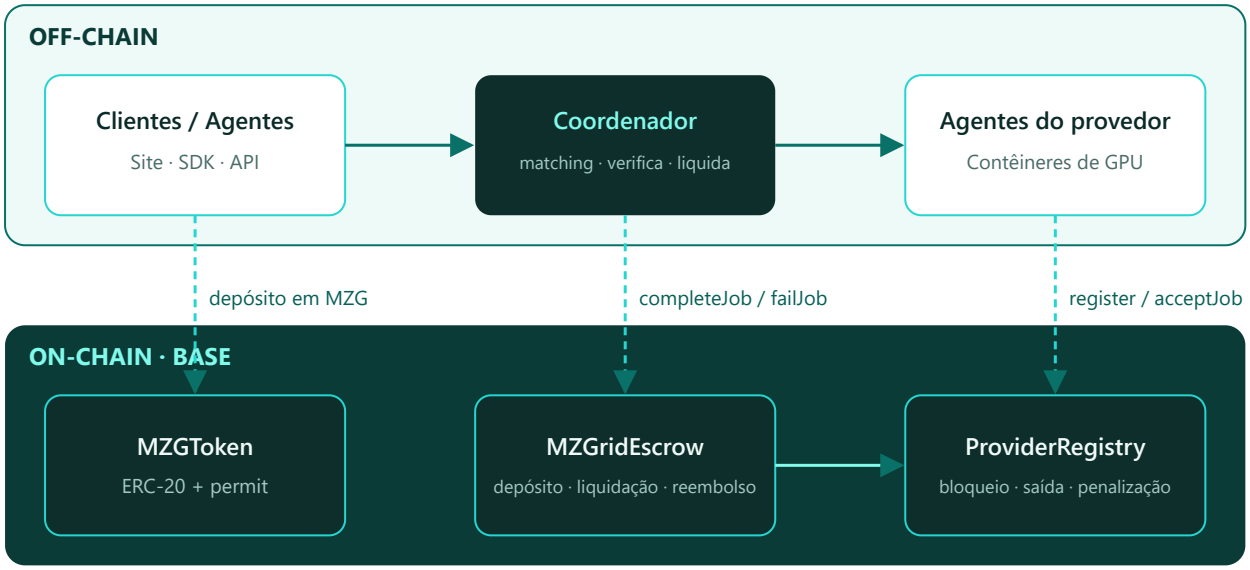


Figura 1 — A execução acontece off-chain; os fundos só se movimentam pelos contratos na Base.

4.1 Camadas

Camada	Responsabilidade
Interface	Site, painéis de clientes e de provedores, SDK TypeScript, API REST com descrição OpenAPI
Orquestração	Índice de ofertas, scheduler, despacho de jobs (long-poll para os agentes dos provedores), streaming de logs
Computação	Máquinas dos provedores executando jobs em contêineres isolados (sem rede do host, sem modo privilegiado, armazenamento por job, limites de recursos)
Verificação	Benchmarks no registro, amostras periódicas de uso, verificações pontuais com jobs canário, pontuação de reputação
Liquidação	MZGridEscrow e ProviderRegistry na Base; um indexador da blockchain espelha seus eventos no coordenador

4.2 Por que a Base

- Compatibilidade com EVM e ferramentas, carteiras e indexadores maduros.
- Custos de transação baixos, o que importa porque cada job gera pelo menos três transações (financiamento, aceite e liquidação).

- Blocos rápidos, de modo que uma implantação passa do financiamento à execução em menos de um minuto.
- Os permits EIP-2612 permitem que os clientes financiem um job em uma única transação.

5 Ciclo de vida do job, custódia e liquidação

5.1 Ciclo de vida

1. **Registro.** O provedor executa o agente, que detecta suas GPUs, roda um benchmark e registra o endereço do provedor on-chain e as máquinas no coordenador.
2. **Matching.** O cliente escolhe uma oferta, ou pede ao scheduler que escolha uma, além de um template e de uma duração máxima.
3. **Financiamento.** Antes de qualquer execução, o cliente deposita `pricePerHour × maxDuration` MZG no MZGridEscrow (com um permit ou uma aprovação). O contrato verifica se o provedor está ativo e se o depósito está dentro do limite por job do provedor.
4. **Aceite.** O provedor precisa aceitar on-chain em até **10 minutos**. O aceite inicia a contagem do tempo do job e define o prazo. Se o provedor nunca aceitar, o cliente pode cancelar e receber o reembolso integral.
5. **Execução.** O agente baixa a imagem, inicia o contêiner e envia amostras de uso pelo menos a cada 90 segundos. O cliente recebe a URL do endpoint e os logs ao vivo.
6. **Liquidação.** Quando o job termina ou o cliente o interrompe, o coordenador envia `completeJob(usedSeconds)`. A custódia faz o pagamento e o reembolso em uma única transação.

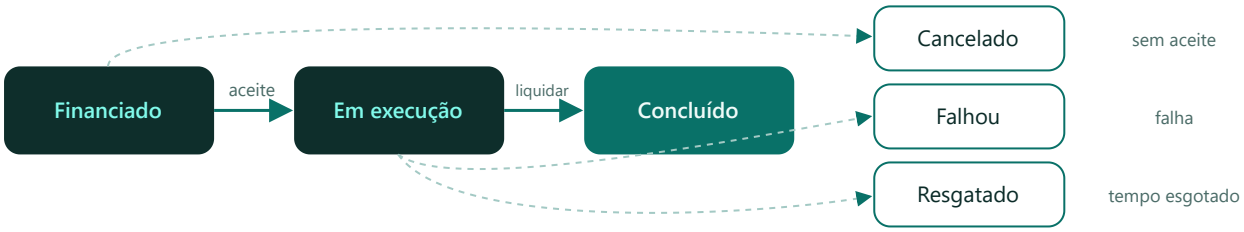


Figura 2 — Estados do job na custódia. Todo estado final diferente de Concluído devolve o depósito integral ao cliente.

5.2 Fórmula de liquidação

```
billedSeconds  = min(usedSeconds, block.timestamp − startedAt)
charged        = min(deposit, pricePerHour × billedSeconds / 3600)
fee            = charged × 7%                      → treasury
providerShare  = charged − fee                     (93%)
locked         = providerShare × 20%               → ProviderRegistry (provider's lock)
liquid         = providerShare − locked            → provider wallet
refund         = deposit − charged                 → customer
```

A cobrança é por segundo. O oráculo não pode cobrar mais tempo do que o decorrido on-chain desde o aceite, nem nunca mais do que o depósito. Todas as partes da divisão somam exatamente o depósito; o arredondamento é limitado a um wei por termo e é verificado por testes de fuzz e de invariantes.

5.3 Exemplo prático

Um cliente aluga uma oferta de RTX 4090 a 150 MZG/hora, reserva 3 horas e para após 2.5 horas.

Item	MZG
Depósito (150 × 3)	450.00
Cobrado (150 × 2.5)	375.00
Taxa do protocolo, 7% → tesouraria	26.25
Parte do provedor, 93%	348.75
— líquido para o provedor (80%)	279.00
— bloqueado no registro (20%)	69.75
Reembolso ao cliente	75.00

5.4 Cenários de falha

Situação	Resultado para o cliente	Resultado para o provedor
O provedor não aceita em até 10 minutos	Cancela; reembolso integral	Sem penalidade
O provedor fica offline ou falha na verificação	O coordenador chama `failJob`; reembolso integral, mais 50% do valor penalizado	Bloqueio penalizado (Seção 6.3)
O coordenador nunca liquida	Após o prazo + 1 hora, `reclaimTimeout`; reembolso integral	Não é pago pelo job

> **Os fundos nunca ficam com o operador.** Os depósitos ficam no contrato de custódia. O coordenador só pode chamar `completeJob` (pagando pelo tempo decorrido ao preço acordado) ou `failJob` (reembolsando o cliente). Ele não pode redirecionar fundos para nenhum outro endereço.

6 Colateral do provedor: o bloqueio de 20%

Muitas redes de computação exigem que os provedores façam stake de tokens antecipadamente. Isso exclui provedores que têm bom hardware, mas não têm tokens, e recompensa quem detém tokens em vez de quem trabalha. O MZGrid adota a abordagem oposta: o colateral é acumulado a partir dos próprios ganhos do provedor.

6.1 Bloqueio e saída

- **Bloqueio.** 20% de cada pagamento ao provedor é transferido para o ProviderRegistry e registrado como saldo bloqueado desse provedor.
- **Saída.** O provedor chama `requestExit()`, o que interrompe a atribuição de novos jobs. Após **10 dias**, e desde que não haja jobs em execução, `withdrawLocked()` devolve todo o saldo bloqueado.
- **Por que um prazo.** Às vezes, as falhas só são detectadas depois do fato. A janela de 10 dias garante que um provedor não possa agir de má-fé e sacar imediatamente seu colateral; a penalização continua possível durante a espera.

6.2 Limite por job

O depósito máximo que um provedor pode aceitar em um único job é

```
jobCap = 25 MZG + 4 × locked
```

Um novo provedor sem nada bloqueado pode aceitar jobs de até 25 MZG. Cada job concluído aumenta seu bloqueio e, portanto, seu limite; um provedor com 1,000 MZG bloqueados pode aceitar jobs de até 4,025 MZG. O limite mantém o dano potencial de qualquer job individual proporcional ao colateral que pode ser penalizado, sem exigir um stake inicial.

6.3 Penalização (slashing)

Motivo	Gatilho	Valor
1 · Abandono	Job aceito, mas não entregue até o prazo	min(bloqueado, 2 × depósito do job)
2 · Offline durante o job	Nenhum relatório de uso por mais de 10 minutos durante a execução	min(bloqueado, 1 × depósito do job)
3 · Falha na verificação	Divergência na saída de um job canário ou benchmark falsificado	100% do bloqueado

O MZG penalizado é dividido em **50% para o cliente afetado** e **50% para a tesouraria**, além do reembolso integral ao cliente. Um provedor nunca lucra com uma falha, e o cliente é compensado para além da simples recomposição do prejuízo.

6.4 Ilustração da economia do provedor

Uma RTX 4090 com preço de 150 MZG/hora e alugada por 504 horas em um mês de 30 dias (70% de utilização) gera o seguinte. Trata-se de aritmética, não de uma previsão; a utilização real depende da demanda.

Item	MZG / mês
Cobrado dos clientes (150 × 504)	75,600.00
Taxa do protocolo (7%)	5,292.00
Parte do provedor (93%)	70,308.00
Líquido (80%)	56,246.40
Adicionado ao bloqueio (20%)	14,061.60
Limite por job após um mês assim (25 + 4 × 14,061.60)	56,271.40

7 Verificação, reputação e agendamento

A custódia garante o pagamento; a verificação responde se o provedor entregou aquilo pelo que foi pago. O MZGrid combina quatro verificações de custo crescente.

7.1 Benchmark no registro

No registro, o agente executa um benchmark de multiplicação de matrizes FP16 em cada GPU. A vazão medida deve ficar entre **70–115%** da referência do catálogo para o modelo declarado. Resultados fora dessa faixa marcam a máquina como não verificada; ela não pode receber jobs. Isso detecta placas rotuladas incorretamente, limites de energia agressivos, thermal throttling e links PCIe severamente restritos.

7.2 Amostras de uso

Enquanto um job está em execução, o agente envia amostras de utilização da GPU pelo menos a cada 90 segundos. A liquidação usa o menor valor entre o tempo de execução declarado pelo provedor e a janela de relatórios observada. Uma lacuna de mais de 10 minutos durante um job em execução faz o job falhar e aciona uma penalização por *offline durante o job*.

7.3 Jobs canário

Cerca de 2% dos jobs são canários: cargas de trabalho cuja saída correta é conhecida de antemão, financiadas a preços normais para que os provedores não consigam distingui-las. Um resultado divergente aciona uma penalização por *falha na verificação* de todo o bloqueio. Como os provedores não conseguem identificar os canários, o custo esperado de simular trabalho supera o benefício.

7.4 Prazos

Um job aceito que não é entregue até o prazo é tratado como abandonado e penalizado em duas vezes o depósito do job, limitado ao bloqueio.

7.5 Reputação

Cada provedor tem uma pontuação de confiabilidade de 0 a 100, exibida em todas as ofertas:

Componente	Peso	Janela
Taxa de conclusão de jobs	40%	90 dias
Uptime das máquinas online	30%	30 dias
Consistência do benchmark	20%	Novos benchmarks vs. referência
Tempo na rede	10%	Linear, limitado a 180 dias

7.6 Agendamento

Os clientes podem escolher uma oferta manualmente ou deixar que o scheduler escolha uma. As ofertas elegíveis precisam ter VRAM suficiente para o template, um provedor ativo e um limite por job pelo menos igual ao depósito. Elas são classificadas por

```
score = 0.50 × normalised price + 0.35 × reliability / 100 + 0.15 × proximity
```

Dar o maior peso ao preço mantém o mercado competitivo; dar o segundo maior peso à confiabilidade recompensa os provedores que mantêm suas máquinas em bom estado.

> **Limites da verificação.** Benchmarks, amostras de uso e canários tornam a fraude detectável e custosa; eles não provam criptograficamente que determinada computação foi executada em determinada GPU. Tecnicamente, os provedores podem observar os dados processados em suas máquinas. Clientes com dados confidenciais não devem presumir garantias de computação confidencial, que o MZGrid não oferece (Seção 10).

8 O token MZG

8.1 Propriedades

Propriedade	Valor
Nome / símbolo	MZGrid / MZG
Padrão	ERC-20 com permit EIP-2612
Rede	Base
Casas decimais	18
Oferta	1,000,000,000 MZG, emitidos uma única vez na implantação
Emissão / queima	Sem função de mint; sem queima pelo protocolo
Atualização	Nenhuma — os contratos são imutáveis
Endereço do contrato (Base)	`0x0018b7E238DBfB210645B88072F32a77bd00C66e`
Governança	Nenhuma

8.2 Utilidade

Uso	Mecanismo
Pagamento de computação	Único meio de pagamento aceito; depositado em custódia antes de cada job
Ganhos	Os provedores recebem 93% de cada valor cobrado em MZG
Colateral	20% de cada pagamento fica bloqueado; o bloqueio define o limite por job do provedor
Receita do protocolo	7% de cada valor cobrado vai para a tesouraria em MZG
Compensação	Os clientes recebem metade de qualquer penalização aplicada ao seu job

Usar uma moeda única mantém o protocolo simples e seus contratos livres de oráculos de preço, uma fonte comum de exploits. Toda unidade de demanda por computação é denominada em MZG, e todo provedor é pago em MZG.

8.3 Alocação e vesting

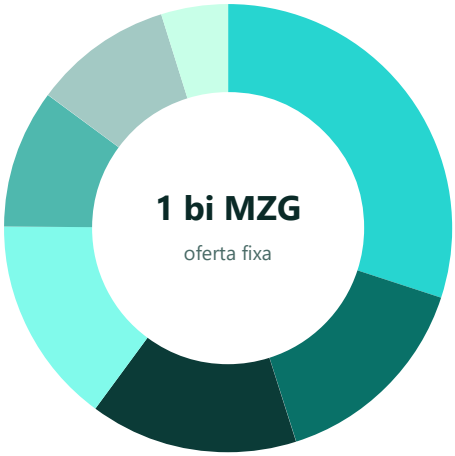


Figura 3 — Alocação da oferta

Categoria	%
Incentivos da rede	30
Ecossistema e grants	15
Tesouraria e operações	15
Equipe e colaboradores	15
Investidores e parceiros	10
Liquidez e desenv. de mercado	10
Lançamento para a comunidade	5

Categoria	MZG	Vesting
Incentivos da rede	300,000,000	Linear ao longo de 48 meses a partir do TGE
Grants para o ecossistema e desenvolvedores	150,000,000	Linear ao longo de 36 meses
Tesouraria e operações	150,000,000	25% no TGE, o restante linear ao longo de 24 meses
Equipe e colaboradores	150,000,000	Cliff de 12 meses, depois linear até o mês 48
Investidores e parceiros estratégicos	100,000,000	Cliff de 6 meses, depois linear até o mês 24
Liquidez e desenvolvimento de mercado	100,000,000	Desbloqueado no TGE
Lançamento para a comunidade	50,000,000	Desbloqueado no TGE

O vesting é garantido por contratos de vesting wallet da OpenZeppelin implantados junto com o token; nenhuma parte, nem mesmo a equipe, pode movimentar antecipadamente uma categoria em vesting. Os valores acumulados antes de um cliff passam a ser resgatáveis no cliff. **A oferta circulante no TGE é de 187.5M MZG (18.75%)**: liquidez 100M, comunidade 50M e a parcela de 37.5M da tesouraria liberada no TGE. O TGE ocorreu com a implantação do beta na mainnet em **17 de novembro de 2026**, que é a data de início de todos os cronogramas de vesting: o cliff dos investidores termina em 19 de maio de 2027 e o da equipe em 17 de novembro de 2027.

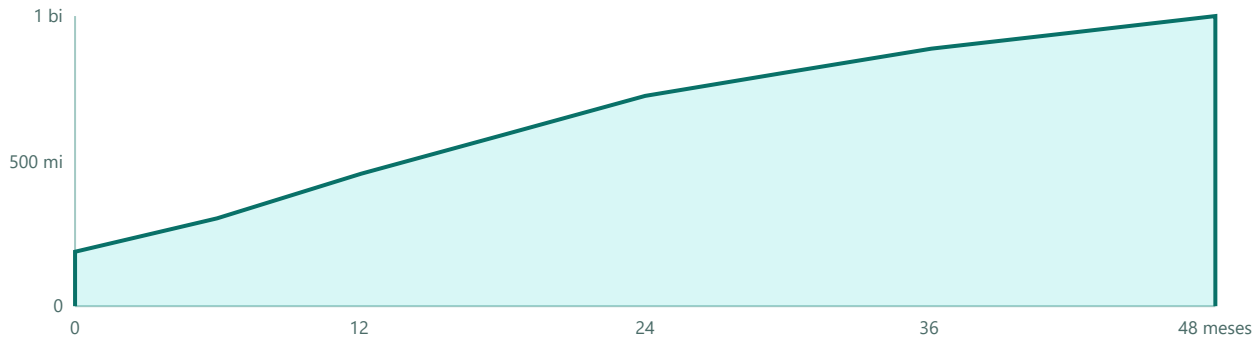


Figura 4 — Oferta desbloqueada acumulada ao longo de 48 meses (esquemático; calculado a partir da tabela de vesting).

8.4 Incentivos da rede

A categoria de incentivos não é distribuída como rendimento passivo. Ela é reservada para bônus de entrada de provedores, pagos somente após jobs verificados, créditos de primeira implantação para clientes e recompensas de confiabilidade para provedores na faixa de reputação mais alta. Todo pagamento de incentivo é publicado pela tesouraria.

8.5 Tesouraria

A tesouraria é uma multisig que recebe as taxas do protocolo, metade de todas as penalizações e sua própria alocação. Ela financia infraestrutura (coordenador, indexador, RPC), revisões de segurança e auditorias, incentivos, grants e operações, e publica um relatório trimestral.

8.6 O que o MZG não faz

- Não confere direitos de governança nem de voto.
- Não oferece recompensas de staking nem rendimento prometido.
- Não há recompras, suporte de preço nem compromissos de listagem.
- Não confere direito sobre receitas, lucros ou ativos da MZGrid Labs.

9 Preços

Os preços são cotados em **MZG por hora de GPU** e cobrados por segundo. Cada provedor define o preço de suas próprias ofertas. A rede publica um preço de referência por modelo de GPU; os provedores podem precificar entre **0.85× e 1.20×** a referência. A taxa do protocolo de 7% está incluída no preço anunciado.

GPU	VRAM	Arquitetura	Referência MZG / hora de GPU
NVIDIA B200	192 GB	Blackwell	1,500
NVIDIA H200 SXM	141 GB	Hopper	1,000
NVIDIA H100 SXM	80 GB	Hopper	750
NVIDIA H100 PCIe	80 GB	Hopper	640
NVIDIA A100 SXM 80GB	80 GB	Ampere	460
NVIDIA A100 PCIe 40GB	40 GB	Ampere	320
NVIDIA L40S	48 GB	Ada	290
NVIDIA RTX 6000 Ada	48 GB	Ada	240
GeForce RTX 5090	32 GB	Blackwell	220
NVIDIA RTX A6000	48 GB	Ampere	160
GeForce RTX 4090	24 GB	Ada	150
GeForce RTX 4080 SUPER	16 GB	Ada	90
GeForce RTX 3090	24 GB	Ampere	75

9.1 Formação de preços e volatilidade do MZG

Como os preços são denominados em MZG, uma variação no valor de mercado do MZG altera o custo real da computação para os clientes e a receita real dos provedores. Os provedores ajustam suas próprias ofertas a qualquer momento; as mudanças de preço valem apenas para novos jobs, de modo que o preço de um job em execução fica fixado no financiamento. Os preços de referência foram definidos em novembro de 2026, quando um MZG era negociado a aproximadamente US$0.003; a essa cotação, uma hora de RTX 4090 (150 MZG) custava cerca de US$0.45. Isso é uma ilustração, não um compromisso de preço. A tabela de referência é revisada quando a mediana das ofertas ao vivo de um modelo se desvia mais de 25% de sua referência por 14 dias consecutivos, e as mudanças são anunciadas com antecedência.

10 Modelo de segurança e confiança

10.1 O que não depende de confiança

- Os depósitos dos clientes ficam exclusivamente no contrato de custódia e só podem ir para o cliente, para o provedor designado, para o bloqueio no registro ou para a tesouraria, de acordo com a fórmula fixa.
- A taxa (7%), a parcela bloqueada (20%), o prazo de saída (10 dias), a janela de aceite (10 minutos), a carência de timeout (1 hora) e o limite por job são constantes em contratos imutáveis.
- Se o coordenador desaparecer, os clientes resgatam seus depósitos após o prazo mais uma hora, e os provedores sacam seus bloqueios pela saída normal.

10.2 O que depende de confiança

Função	Detida por	Pode	Não pode
Oráculo (liquidação)	Chave do coordenador em um armazenamento de chaves baseado em hardware	Liquidar jobs pelo tempo decorrido; falhar jobs (reembolso)	Cobrar além do tempo decorrido on-chain ou do depósito; enviar fundos a outro destino
Slasher	Chave do coordenador	Penalizar o bloqueio de um provedor pelos motivos 1–3	Penalizar além do bloqueio; ficar com os fundos penalizados (divisão fixa)
Admin	Multisig	Conceder e revogar funções	Emitir tokens, alterar constantes, atualizar contratos, movimentar fundos de usuários

O coordenador é, portanto, uma parte confiável, porém limitada: um oráculo comprometido poderia subdeclarar ou superdeclarar o uso dentro do depósito e do tempo decorrido, ou aplicar penalizações injustas dentro das regras. Esses riscos são mitigados pela custódia das chaves, por eventos públicos on-chain que tornam cada liquidação e cada penalização auditáveis e por uma janela de revisão de 30 dias para penalizações contestadas. Reduzir essa dependência de confiança — com múltiplos verificadores independentes e atestações de uso mais robustas — é um item do roadmap.

10.3 Isolamento de cargas de trabalho e privacidade

Os jobs rodam em contêineres sem rede do host nem modo privilegiado, com armazenamento por job e limites de recursos; os segredos são repassados apenas ao contêiner do job. Isso protege os provedores do código dos clientes e os clientes uns dos outros. Isso **não** oculta os dados do dono da máquina: um provedor com acesso físico pode inspecionar a memória e o armazenamento. O MZGrid não oferece computação confidencial. Cargas de trabalho sensíveis devem ser avaliadas levando isso em conta.

10.4 Garantias até o momento

- Smart contracts: testes unitários, de fuzz (1,000 execuções por propriedade) e de invariantes (conservação de tokens, saldo da custódia igual aos depósitos em aberto, saldo do registro igual ao total bloqueado); 100% de cobertura de linhas, branches e funções dos contratos principais; análise estática com Slither, com os achados triados.
- Testes de ponta a ponta cobrindo registro, financiamento via permit, execução, divisão da liquidação, penalização por offline e saída, executados antes de cada release.
- Beta na mainnet da Base desde 17 de novembro de 2026; todos os contratos estão verificados no Basescan.
- **Nenhuma auditoria externa independente foi concluída até o momento.** Ela é um requisito para sair do beta (Seção 11), e seu relatório será publicado.

11 Roadmap e status atual

As fases são orientadas por marcos, não por datas. Um recurso só é anunciado como disponível depois de implementado, testado e implantado.

11.1 Marcos

Data	Marco
2026	Protocolo, coordenador, agente do provedor, SDK e site desenvolvidos; contratos testados com suítes unitárias, de fuzz e de invariantes e execuções de ponta a ponta em uma blockchain local
17 de novembro de 2026	Beta na mainnet da Base: MZGToken, ProviderRegistry, MZGridEscrow e carteiras de vesting implantados; TGE; marketplace aberto com liquidação por segundo, depósitos via permit, verificações com jobs canário e o conjunto completo de templates

11.2 Fases

Fase	Escopo	Status
1 · Fundação	Marca e posicionamento; site e documentação; design do token MZG; smart contracts principais; registro de provedores; marketplace	Concluída
2 · Produto funcional	Agente do provedor; agendamento; templates de inferência de IA; pagamentos em MZG via custódia; bloqueio de 20%, saída e penalização; reputação; API, SDK e documentação para desenvolvedores; testes de desempenho e segurança	No ar (beta na mainnet)
3 · Expansão da rede	Mais modelos de GPU e regiões; matching avançado; serviços corporativos; mais templates; integrações para desenvolvedores	Em andamento
Saída do beta	Auditoria externa independente; procedimentos publicados de custódia das chaves do oráculo e da multisig; atualização automática do agente	Próxima
4 · Computação inteligente	Otimização de recursos orientada por IA; compras por agentes autônomos; múltiplos verificadores independentes; atestação de uso mais robusta; pesquisa em treinamento distribuído	Pesquisa

11.3 A rede em novembro de 2026

Números aproximados da primeira semana após o lançamento (novembro de 2026), obtidos a partir dos eventos de liquidação on-chain e dos registros do coordenador.

Métrica	Valor
Provedores	~64
GPUs registradas / online	~260 / ~220
Países	12
Faixa de utilização diária	30–38%
Jobs concluídos desde o lançamento	~1,700
Horas de GPU entregues desde o lançamento	~8,500
MZG liquidados desde o lançamento	~4.6M
MZG bloqueados pelos provedores	~0.85M
Clientes ativos (30 dias)	~190
Confiabilidade média dos provedores	~87

Os números em tempo real são publicados na página da rede no site. O crescimento passado da rede não indica crescimento futuro.

> **Métrica de sucesso.** O projeto se avalia pela recorrência com que os clientes compram computação e os provedores continuam a oferecê-la — uso recorrente, horas de GPU concluídas e retenção de provedores —, e não por métricas do token.

12 Riscos

- **Risco de smart contracts.** Apesar dos testes, os contratos podem conter defeitos. Eles são imutáveis; um defeito não pode ser corrigido no próprio contrato e exigiria a migração para novos contratos.
- **Risco do operador.** O coordenador detém as funções de oráculo e de slasher (Seção 10.2). Erros ou comprometimento poderiam causar liquidações ou penalizações incorretas dentro dos limites dos contratos.
- **Risco de preço do token.** Todos os preços são em MZG. O valor do MZG pode cair ou subir acentuadamente, alterando o custo real da computação e o valor real dos ganhos e bloqueios dos provedores. O MZG pode ter pouca ou nenhuma liquidez.
- **Risco de desbloqueio da oferta.** As alocações em vesting são desbloqueadas ao longo de 48 meses e podem ser vendidas.
- **Risco de mercado.** A rede pode não atrair clientes ou provedores suficientes; as nuvens estabelecidas e outras redes descentralizadas competem em preço e recursos.
- **Risco de hardware e de verificação.** A verificação torna a fraude custosa, mas não consegue provar cada computação. O hardware dos provedores pode falhar; os jobs podem ser interrompidos.
- **Risco de dados.** Os provedores podem acessar os dados processados em suas máquinas (Seção 10.3).
- **Risco da blockchain.** Indisponibilidades do sequenciador da Base, reorganizações ou falhas de RPC podem atrasar o financiamento e a liquidação.
- **Risco regulatório.** O tratamento jurídico de tokens e de marketplaces de computação varia conforme a jurisdição e pode mudar, afetando a disponibilidade do token ou do serviço.
- **Risco de chaves.** Usuários que perderem as chaves de suas carteiras perdem o acesso aos seus MZG, inclusive aos saldos bloqueados de provedores.

A Parâmetros do protocolo

Parâmetro	Valor	Onde é aplicado
Taxa do protocolo	7% (`FEE_BPS = 700`)	MZGridEscrow, constante
Parcela bloqueada do provedor	20% da parte do provedor (`LOCK_BPS = 2000`)	MZGridEscrow, constante
Janela de aceite	10 minutos	MZGridEscrow, constante
Carência de timeout	1 hora após o prazo	MZGridEscrow, constante
Duração máxima do job	365 dias	MZGridEscrow, constante
Prazo de saída	10 dias	ProviderRegistry, constante
Limite por job	25 MZG + 4 × bloqueado	ProviderRegistry, constantes
Divisão da penalização	50% cliente / 50% tesouraria	ProviderRegistry
Valores de penalização	2× / 1× o depósito (limitado ao bloqueio); 100% do bloqueio	Política do coordenador, limitada on-chain ao bloqueio
Faixa do benchmark	70–115% da vazão FP16 de referência	Coordenador
Intervalo de relatório de uso	≤ 90 s; offline após 10 min	Coordenador
Taxa de jobs canário	~2% dos jobs	Coordenador
Faixa de preço das ofertas	0.85–1.20× a referência	Coordenador
Oferta total	1,000,000,000 MZG	MZGToken, constante

B Interface dos contratos

MZGridEscrow

```
createJob(jobId, provider, pricePerHour, maxDuration, specHash)
createJobWithPermit(jobId, provider, pricePerHour, maxDuration, specHash, deadline, v, r, s)
acceptJob(jobId)                                  // provider, within 10 min
cancelJob(jobId)                                  // customer, if not accepted in 10 min
completeJob(jobId, usedSeconds, resultHash)       // ORACLE_ROLE
failJob(jobId, reason)                            // ORACLE_ROLE, full refund
reclaimTimeout(jobId)                             // customer, after deadline + 1 h
events: JobFunded, JobAccepted, JobCompleted, JobCancelled, JobFailed, JobReclaimed
```

ProviderRegistry

```
register(metadataURI) · updateMetadata(metadataURI)
requestExit() · withdrawLocked()                  // after 10 days, no running jobs
jobCap(provider) · isActive(provider) · getProvider(provider)
addLock(provider, amount)                         // ESCROW_ROLE
slash(provider, amount, jobId, reason, client)    // SLASHER_ROLE, reasons 1–3
events: Registered, LockAdded, ExitRequested, LockWithdrawn, Slashed
```

Endereços implantados (mainnet da Base, chain 8453)

```
MZGToken          0x0018b7E238DBfB210645B88072F32a77bd00C66e
ProviderRegistry  0x12b555e28C1756CEEDCa9c0e2b920aCf60D8e477
MZGridEscrow      0xD41600dA905E8327f7D8a9572cC10C6930Fa2bb9
```

MZGToken

```
ERC-20 "MZGrid" / "MZG", 18 decimals, EIP-2612 permit, fixed supply, no owner
```

C Glossário

Termo	Significado
Agente	O software `mzgrid-provider` que roda nas máquinas dos provedores
Canário	Job com saída correta conhecida, usado para detectar trabalho falsificado
Coordenador	Serviço off-chain de matching, despacho, verificação e envio de liquidações
Depósito	MZG colocado em custódia antes de um job: preço por hora × duração máxima
Custódia (escrow)	O contrato MZGridEscrow, que guarda os depósitos e executa as liquidações
Prazo de saída	Espera de 10 dias entre `requestExit` e `withdrawLocked`
Limite por job	Depósito máximo que um provedor pode aceitar: 25 MZG + 4 × bloqueado
Bloqueio	20% de cada pagamento ao provedor, mantido no registro como colateral
Oráculo	Função autorizada a liquidar jobs ou declará-los como falhos; detida pelo coordenador
Permit	Assinatura EIP-2612 que autoriza uma transferência de tokens sem uma transação de aprovação separada
Preço de referência	Preço publicado em MZG por hora de GPU para cada modelo de GPU; as ofertas devem ficar entre 0.85–1.20×
Penalização (slash)	Perda de parte ou de todo o bloqueio de um provedor por uma falha definida
TGE	Token generation event (evento de geração do token): implantação do token e início do vesting

© 2026 MZGrid Labs. Este documento pode ser redistribuído sem modificações. MZGrid, MZG e o logotipo do MZGrid são marcas da MZGrid Labs. Este documento é uma tradução do original em inglês; em caso de qualquer divergência, prevalece o original em inglês.