Cloud Hosting Options for Node.js/Express API

Technical Evaluation Document

with Supabase PostgreSQL Database

Executive Summary

This document evaluates free cloud hosting options for deploying a Node.js/Express Web API that connects to a Supabase PostgreSQL database. With your GitHub Student Developer Pack, you have access to enhanced free tiers and credits across multiple platforms.

We'll examine four primary hosting platforms that offer free or credit-based options suitable for student projects and development work. Each platform has been evaluated based on ease of deployment, performance, scalability, and long-term viability.

Database Architecture: Supabase

Since you're using Supabase for your PostgreSQL database, your hosting platform only needs to run the Node.js/Express server. Supabase handles:

- PostgreSQL database hosting and management
- Automatic backups and point-in-time recovery
- Real-time subscriptions and webhooks
- Built-in authentication (optional)
- RESTful API auto-generation

Supabase Free Tier Specifications

- 500 MB database storage
- Unlimited API requests
- 50,000 monthly active users
- Social OAuth providers
- Projects pause after 1 week of inactivity (easily wake-able)

Your API server will connect to Supabase using a connection string and handle custom business logic, data validation, and any functionality beyond basic CRUD operations that Supabase's auto-generated API doesn't cover.

GitHub Student Developer Pack Benefits

As a verified GitHub student, you receive significant benefits:

- **DigitalOcean:** $200 credit (valid for 1 year)
- **Microsoft Azure:** $100 credit plus free services
- **Heroku:** Free credits for hosting (note: no longer offers a permanent free tier)
- **MongoDB Atlas:** $50 credit (not needed since you're using Supabase)

Evaluation Criteria

Each platform has been evaluated on the following factors:

- **Cost:** True free tier availability and sustainability
- **Ease of Deployment:** Setup complexity and learning curve
- **GitHub Integration:** Automatic deployment from Git repositories
- **Performance:** Response times and cold start behavior
- **Scalability:** Growth path as your application expands
- **Developer Experience:** Documentation, debugging tools, and support
- **Reliability:** Uptime guarantees and stability

Option 1: Render

Overview

Render is a modern cloud platform that has gained significant popularity as a Heroku alternative. It offers straightforward deployment with excellent developer experience and a genuinely free tier suitable for hobby projects and student work.

Free Tier Specifications

- **Instance Type:** 512 MB RAM, shared CPU
- **Bandwidth:** 100 GB per month
- **Build Minutes:** 500 minutes per month
- **Sleep Behavior:** Services spin down after 15 minutes of inactivity
- **Cold Start Time:** ~30 seconds to wake up
- **Custom Domains:** Supported with automatic SSL
- **Environment Variables:** Unlimited

Deployment Process

- Connect your GitHub repository
- Render auto-detects Node.js and reads package.json
- Specify build command (npm install) and start command (node server.js)
- Add Supabase connection string as environment variable
- Deploy with one click
- Automatic redeployment on every Git push

PROS

- ✓ **Truly Free Forever:** No credit card required, permanent free tier with no time limits
- ✓ **Excellent Documentation:** Clear guides specifically for Node.js/Express applications
- ✓ **Auto-Deploy from GitHub:** Push code and it automatically builds and deploys
- ✓ **Preview Environments:** Automatically creates preview deployments for pull requests
- ✓ **Built-in SSL:** Automatic HTTPS with free SSL certificates
- ✓ **Live Logs:** Real-time logging and debugging in the dashboard
- ✓ **Background Workers:** Support for cron jobs and background processing

- ✓ **Environment Groups:** Manage environment variables across multiple services
- ✓ **Docker Support:** Can deploy containerized applications if needed
- ✓ **Rollback Capability:** Easy rollback to previous deployments

CONS

- ✗ **Cold Starts:** 15-minute inactivity causes service to sleep; 30-second wake time affects first request
- ✗ **Limited Resources:** 512 MB RAM may be insufficient for memory-intensive applications
- ✗ **Shared CPU:** Performance can be inconsistent during high-load periods
- ✗ **Build Time Limits:** 500 minutes per month may run out with frequent deployments
- ✗ **Geographic Limitation:** Free tier limited to single region deployment

Best Use Cases

- Student projects and portfolio applications
- APIs with intermittent traffic (can tolerate cold starts)
- Development and staging environments
- Proof-of-concept applications
- Applications where first-request latency is acceptable

Pricing Path

If you outgrow the free tier, the Starter tier is $7/month and includes always-on instances (no sleep), 1 GB RAM, and dedicated resources.

Option 2: Railway

Overview

Railway is a developer-first infrastructure platform known for its exceptional user experience and magical deployment process. It automatically detects your tech stack and configures everything for you.

Free Trial Specifications

- **Trial Credit:** $5 one-time credit for new accounts
- **Usage:** Covers approximately 500 hours of basic instance time
- **Instance Resources:** 512 MB RAM, 1 vCPU (shared)
- **Sleep Behavior:** No automatic sleep - runs until credits deplete
- **Network:** 100 GB outbound data included in credit
- **After Trial:** Hobby Plan at $5/month required to continue

Deployment Process

- Connect GitHub account or repository
- Railway auto-detects Node.js project structure
- Automatically configures build and start commands
- Add environment variables for Supabase connection
- Deploy immediately - literally 2-3 clicks
- Automatic deployments on every commit

PROS

- ✓ **Best Developer Experience:** Industry-leading ease of deployment - truly 'magical' setup
- ✓ **No Cold Starts:** Services stay running 24/7 within credit limits
- ✓ **Monorepo Support:** Can deploy multiple services from single repository
- ✓ **Private Networking:** Services can communicate privately without consuming bandwidth
- ✓ **Ephemeral Environments:** Automatic PR preview environments for testing
- ✓ **Excellent Observability:** 90-day log retention, metrics, and monitoring built-in
- ✓ **Fast Deployments:** Typically completes in under 60 seconds
- ✓ **Team Collaboration:** Share projects easily with collaborators
- ✓ **CLI Tool:** Powerful command-line interface for advanced operations
- ✓ **Database Templates:** One-click PostgreSQL, MySQL, Redis, MongoDB deployment (though you're using Supabase)

CONS

- ✗ **No Permanent Free Tier:** $5 trial credit runs out after ~500 hours or ~3 weeks of continuous running
- ✗ **Requires Payment After Trial:** Must upgrade to $5/month Hobby plan to continue service
- ✗ **Credit Burn Rate:** High traffic can deplete credits faster than expected ($0.0002/CPU-second)
- ✗ **Limited Control:** Less infrastructure customization compared to VPS options
- ✗ **Account Verification:** Some users report issues with GitHub verification for trial credits

Best Use Cases

- Production-ready projects that need always-on availability
- Applications requiring real-time features (WebSockets)
- Testing a project for 2-3 weeks before committing to paid hosting
- Teams that value developer experience over cost
- Applications where cold starts are unacceptable

Pricing Path

After trial credits expire, the Hobby plan at $5/month includes $5 in usage credits, which typically covers 500+ hours of a basic Node.js API. Overages are billed at standard rates.

Option 3: Vercel (Serverless)

Overview

Vercel is a serverless platform optimized for frontend frameworks like Next.js but supports serverless Node.js functions. Unlike traditional hosting, your Express API would need to be structured as serverless functions rather than a long-running server.

Important Consideration

Vercel is fundamentally different from Render and Railway. It doesn't run a traditional Node.js/Express server. Instead, it runs serverless functions that execute on-demand. This requires restructuring your application.

Free Tier Specifications (Hobby Plan)

- **Serverless Function Executions:** Unlimited
- **Function Duration:** 10 seconds maximum per invocation

- **Function Memory:** 1024 MB
- **Bandwidth:** 100 GB per month
- **Builds:** 6,000 minutes per month
- **Deployments:** Unlimited
- **Custom Domains:** Unlimited with automatic SSL

Architecture Adjustment Required

Instead of a traditional Express server, you create serverless functions:

- Each API endpoint becomes a separate function file in an /api directory
- Functions execute independently on each request
- No persistent in-memory state between requests
- Database connections must be managed per-function

PROS

- ✓ **Truly Free Forever:** Generous free tier with no time limits or credit depletion
- ✓ **No Cold Start Concerns:** Serverless functions wake instantly (sub-second)
- ✓ **Global Edge Network:** Functions execute at edge locations closest to users
- ✓ **Automatic Scaling:** Handles traffic spikes effortlessly without configuration
- ✓ **Zero Maintenance:** No server management or infrastructure concerns
- ✓ **Preview Deployments:** Every Git branch gets its own URL for testing
- ✓ **Excellent Analytics:** Built-in performance monitoring and analytics
- ✓ **Superior Frontend Integration:** Perfect if you also host your React/Next.js frontend on Vercel
- ✓ **Environment Variables:** Encrypted and scoped per environment

CONS

- ✗ **Architecture Change Required:** Cannot deploy traditional Express apps without significant restructuring
- ✗ **10-Second Timeout:** Functions must complete within 10 seconds (60s on Pro plan)
- ✗ **No WebSocket Support:** Cannot maintain persistent connections
- ✗ **No Background Jobs:** Cannot run scheduled tasks or long-running processes
- ✗ **Stateless Only:** No in-memory caching or session storage between requests
- ✗ **Database Connection Overhead:** Each function invocation must establish database connection (can be mitigated with connection pooling)
- ✗ **Learning Curve:** Serverless paradigm requires understanding of function-based architecture

Best Use Cases

- Simple REST APIs with short-lived requests
- APIs primarily serving frontend applications also hosted on Vercel
- Projects requiring automatic global distribution
- APIs with highly variable traffic patterns
- You're already using Next.js and want backend integration

Pricing Path

Pro plan at $20/month per team member increases function timeout to 60 seconds, adds 1,000 GB bandwidth, and includes team collaboration features.

Option 4: DigitalOcean App Platform

Overview

DigitalOcean's App Platform is a fully managed PaaS that simplifies deployment while giving you the power of DigitalOcean's infrastructure. With your GitHub Student Developer Pack, you receive $200 credit valid for one year, providing essentially free hosting for student projects.

GitHub Student Pack Benefit

- **Credit Amount:** $200
- **Validity:** 12 months from signup
- **Monthly Equivalent:** ~$16.67 per month of services
- **Requirement:** Valid credit card required (for verification, not charged during credit period)

App Platform Pricing (What Your Credit Covers)

- **Basic Web Service:** $5/month - 512 MB RAM, 1 vCPU
- **Professional Web Service:** $12/month - 1 GB RAM, 1 vCPU

With $200 credit, you can run a Basic app for 40 months or Professional app for 16+ months

App Platform Features

- **No Sleep Mode:** Apps run 24/7 without sleeping
- **Auto-Deploy:** Deploys automatically from GitHub/GitLab
- **Scaling:** Horizontal and vertical scaling available
- **SSL Certificates:** Automatic HTTPS with Let's Encrypt
- **CDN:** Built-in global CDN for static assets

PROS

- ✓ **Substantial Free Credit:** $200 covers 12 months of student development work
- ✓ **No Cold Starts:** Traditional server model means instant response times
- ✓ **Reliable Infrastructure:** Backed by DigitalOcean's proven cloud infrastructure
- ✓ **Complete Ecosystem:** Access to DigitalOcean's full suite (Droplets, Spaces, Databases)
- ✓ **Better Resources:** Basic tier offers better performance than most free tiers
- ✓ **Multiple Apps:** Can run multiple apps within credit allowance
- ✓ **Professional Environment:** Production-grade features suitable for real-world applications
- ✓ **Background Workers:** Can run worker processes alongside web services
- ✓ **Docker Support:** Deploy containerized applications easily
- ✓ **Multiple Environments:** Separate staging and production deployments

CONS

- ✗ **Not Truly Free:** Requires credit card and eventually transitions to paid service
- ✗ **Time Limited:** Credit expires after 12 months
- ✗ **Credit Card Required:** Must provide payment method upfront

- ✗ **Auto-Billing After Credit:** Will automatically charge after credit depletes unless cancelled
- ✗ **Less Beginner-Friendly:** More options mean steeper learning curve than Render or Railway
- ✗ **Credit Management:** Need to monitor credit usage to avoid surprise charges

Best Use Cases

- Year-long student projects with production requirements
- Applications requiring always-on reliability
- Learning professional deployment practices
- Projects that may need scaling or multiple environments
- When you want to explore DigitalOcean's broader ecosystem

Cost Management Tips

- Set up billing alerts at $150, $175, and $190 usage
- Use the Basic tier ($5/month) to maximize credit duration
- Monitor the billing dashboard regularly
- Destroy unused resources promptly
- Set a calendar reminder for month 11 to cancel if not continuing

Side-by-Side Comparison

Quick Reference Table

Feature	Render	Railway	Vercel	DigitalOcean
Cost	Free forever	$5 trial, then $5/mo	Free forever	$200 credit (12 mo)
RAM	512 MB	512 MB	1024 MB	512 MB - 1 GB
Sleep Mode	Yes (15 min)	No	N/A (serverless)	No
Cold Start	~30 seconds	None	<1 second	None
Architecture	Traditional server	Traditional server	Serverless only	Traditional server
Ease of Deploy	Easy	Easiest	Requires restructure	Moderate
GitHub Integration	Excellent	Excellent	Excellent	Good
WebSockets	Yes	Yes	No	Yes
Background Jobs	Yes	Yes	No	Yes
Credit Card Required	No	No	No	Yes
Best For	Long-term free hosting	Best dev experience	Auto-scaling APIs	Production projects

Recommendations

For Your Movie Database API Project

Based on your requirements (Node.js/Express API connecting to Supabase), here are scenario-based recommendations:

BEST CHOICE FOR STUDENTS: Render

Why: Completely free with no credit card, no time limits, and no credit expiration. Perfect for learning, portfolio projects, and course assignments. The cold start issue (15-minute sleep) is acceptable for most student projects.

BEST FOR PRODUCTION/PORTFOLIO: Railway (trial) → DigitalOcean (long-term)

Why: Railway gives you 2-3 weeks of perfect hosting to develop and test. For long-term hosting of portfolio projects you want to showcase, transition to DigitalOcean using your $200 student credit for reliable, always-on hosting.

IF YOU'RE WILLING TO REFACTOR: Vercel

Why: If you're open to restructuring your Express app as serverless functions, Vercel offers the most generous free tier with exceptional performance. Best combined with a Next.js frontend.

NOT RECOMMENDED FOR THIS PROJECT: Railway as long-term solution (unless willing to pay $5/month)

Why: While Railway has the best developer experience, the $5/month cost after trial makes it less ideal for students compared to truly free alternatives.

Recommended Deployment Strategy

Phased Approach for Development and Learning

Phase 1: Development (Weeks 1-2)

- Platform: Render
- Purpose: Build and test your API with frequent deployments
- Benefit: Free, no credit card, quick setup
- Drawback: Cold starts acceptable during development

Phase 2: Testing/Showcase (Week 3+)

- Platform: Railway (using $5 trial credit)
- Purpose: Test production-like environment, showcase to professors/employers
- Benefit: No cold starts, excellent performance
- Duration: 2-3 weeks of continuous hosting

Phase 3: Portfolio/Long-term (Month 2+)

- Platform: DigitalOcean App Platform (using $200 student credit)
- Purpose: Professional hosting for resume/portfolio projects
- Benefit: 12 months of reliable hosting, production-grade infrastructure
- Cost: Approximately $5-12/month deducted from credit

Technical Setup: Connecting to Supabase

Regardless of which platform you choose, connecting your Node.js/Express API to Supabase follows the same pattern:

1. Install Supabase Client

npm install @supabase/supabase-js

2. Configure Environment Variables

Add to your hosting platform's environment variables:

- SUPABASE_URL=your-project-url.supabase.co
- SUPABASE_ANON_KEY=your-anon-key

3. Initialize Supabase Client in Your API

Create a supabase.js file with the client configuration. Your Express routes can then import this client to interact with your database.

4. Connection Pooling Considerations

- Supabase handles connection pooling automatically through its REST API
- Each platform handles connections differently but all work seamlessly with Supabase
- Serverless platforms (Vercel) benefit from Supabase's HTTP API rather than traditional PostgreSQL connections

Common Pitfalls to Avoid

- **Port Configuration:** Use process.env.PORT (not hardcoded 3000) as hosting platforms assign dynamic ports
- **Environment Variables:** Never commit .env files; use platform's environment variable system
- **Build Commands:** Ensure package.json has correct start script
- **CORS Configuration:** Configure CORS properly if your frontend is on a different domain
- **Health Check Endpoints:** Implement /health or / routes for platform health checks
- **Cold Start Awareness:** On Render, first request after sleep will be slow - design UI accordingly
- **Credit Monitoring:** On Railway and DigitalOcean, regularly check credit/usage to avoid surprises

Conclusion

For your Node.js/Express API with Supabase PostgreSQL database, you have excellent free hosting options available as a student:

Render offers the most sustainable free option with no credit card requirement and no time limits. The cold start behavior is a reasonable trade-off for truly free, permanent hosting.

Railway provides the smoothest development experience with no cold starts, perfect for 2-3 weeks of testing, but requires payment afterward.

Vercel is excellent if you're willing to refactor into serverless functions, offering unlimited executions with instant wake times.

DigitalOcean with your $200 student credit provides production-grade hosting for an entire year, ideal for portfolio projects you want to showcase long-term.

Our recommendation: Start with Render for development, use Railway's trial for testing, and transition to DigitalOcean when you need long-term, reliable hosting for your portfolio. This strategy maximizes your free resources while providing the best experience for each project phase.

All four platforms integrate seamlessly with Supabase and GitHub, providing automatic deployments and excellent developer experience. The choice ultimately depends on your specific needs for uptime, cold start tolerance, and long-term hosting requirements.

Additional Resources

- **GitHub Student Developer Pack:** education.github.com/pack
- **Render Documentation:** render.com/docs
- **Railway Documentation:** docs.railway.app
- **Vercel Documentation:** vercel.com/docs
- **DigitalOcean App Platform:** docs.digitalocean.com/products/app-platform
- **Supabase Documentation:** supabase.com/docs