VKNG Render Deployment

VKNG AI Public Dashboard - Render Deployment Technical Document

VKNG AI Public Dashboard - Render Deployment Technical

Document

Document Version: 1.0
Created: January 23, 2026
Purpose: Technical specifications for VKNG public dashboard hosted on FIDUS Render platform

---

1. DEPLOYMENT SUMMARY

Field	Value
Public URL	https://fidus-investment-platform.onrender.com/getvkng
Authentication	None required (public access)
Platform	Render.com
Frontend Service	fidus-investment-platform
Backend Service	fidus-api

2. INFRASTRUCTURE DETAILS

2.1 Render Services

Frontend Service: | Field | Value | |——-|——-| | Service Name | fidus-investment-platform | | Service Type | Static Site | | URL | https://fidus-investment-platform.onrender.com | | Build Command | `npm run build` | | Publish Directory | `build` | | Auto-Deploy | Yes (on GitHub push to main) |

Backend Service: | Field | Value | |——-|——-| | Service Name | fidus-api | | Service Type | Web Service | | URL | https://fidus-api.onrender.com | | Start Command | `uvicorn server:app --host 0.0.0.0 --port 10000` | | Auto-Deploy | Yes (on GitHub push to main) |

2.2 IP Addresses

Service	IP Address	Notes
Render Outbound	Dynamic	Render uses dynamic IPs
MongoDB Atlas	Whitelist 0.0.0.0/0	Or use Render static IPs (paid)
Preview Environment	34.16.56.64	Current preview instance

Render Static Outbound IPs (if using paid tier): - Refer to Render dashboard for static IP assignment - Required for MongoDB Atlas IP whitelisting in production

3. ROUTES & ENDPOINTS

3.1 Frontend Routes

Route	Component	Authentication	Description
/getvkng	GetVKNGPublic	**None**	Public VKNG dashboard
/viking	VikingApp	Required	Admin VKNG dashboard
/	FidusApp	Required	Main FIDUS platform

3.2 Backend API Endpoints

Base URL: `https://fidus-api.onrender.com/api/viking`

Method	Endpoint	Description
GET	`/accounts`	Get all VKNG accounts (CORE & PRO)
GET	`/deals/{strategy}`	Get trade history
GET	`/analytics/{strategy}`	Get performance analytics
GET	`/summary`	Get combined summary
GET	`/symbols/{strategy}`	Get symbol distribution
GET	`/balance-history/{strategy}`	Get balance chart data
GET	`/monthly-returns/{strategy}`	Get monthly returns

4. ENVIRONMENT VARIABLES

4.1 Frontend (Render - fidus-investment-platform)

```
REACT_APP_BACKEND_URL=https://fidus-api.onrender.com
NODE_ENV=production
CI=false
GENERATE_SOURCEMAP=false
```

4.2 Backend (Render - fidus-api)

```
# Database
MONGO_URL=mongodb+srv://[user]:
        [pass]@fidus.y1p9be2.mongodb.net/fidus_production
DB_NAME=fidus_production

# Security
JWT_SECRET_KEY=[secure-random-string]
```

```
# CORS
FRONTEND_URL=https://fidus-investment-platform.onrender.com
BACKEND_URL=https://fidus-api.onrender.com

# Optional - MT5 Bridge
MT5_BRIDGE_URL=http://92.118.45.135:8000
MT5_BRIDGE_ACTIVE=true
```

5. DATABASE CONFIGURATION

5.1 MongoDB Atlas

Field	Value
Cluster	FIDUS
Database	fidus_production
Region	AWS us-east-1

5.2 Collections Used by VKNG

Collection	Purpose	Records
mt5_accounts	Master account data	~15
mt5_deals	MT5 trade history	~3,000+
viking_accounts	VKNG account metadata	~3
viking_deals_history	MT4 historical trades	~4,700
viking_analytics	Performance metrics	~4
viking_balance_history	Balance snapshots	~1,600

6. TRADING STRATEGIES DISPLAYED

6.1 CORE Strategy

Field	Value
Current Account	885822 (MT5)
Historical Account	33627673 (MT4)
Platform	MT5
Broker	MEXAtlantic
Total Trades	~4,143

6.2 PRO Strategy

Field	Value
Account	1309411
Platform	MT4

Broker	Traders Trust
Total Trades	~1,152

7. DEPLOYMENT WORKFLOW

7.1 Code Changes to GitHub

```
# From local development
git add .
git commit -m "Update VKNG public dashboard"
git push origin main
```

7.2 Render Auto-Deploy

1. Render detects push to `main` branch
2. Triggers build for both services
3. Frontend: `npm install && npm run build`
4. Backend: `pip install -r requirements.txt`
5. Health check runs
6. Traffic routes to new deployment

7.3 Manual Deploy (Render Dashboard)

1. Go to https://dashboard.render.com
2. Select service (fidus-investment-platform or fidus-api)
3. Click "Manual Deploy" → "Deploy latest commit"

8. MONITORING & HEALTH

8.1 Health Check Endpoints

Service	Endpoint	Expected Response
Backend	`GET /api/health`	`{"status": "healthy"}`
Frontend	`GET /`	200 OK

8.2 Logs Access

Render Dashboard: - https://dashboard.render.com → Select Service → Logs

API:

```
curl -H "Authorization: Bearer rnd_zyltqZyReSorlDoVp4hs8LdmJPo3" \
  "https://api.render.com/v1/services/[SERVICE_ID]/logs"
```

9. SECURITY NOTES

9.1 Public Dashboard (/getvkng)

- **No authentication required** - Publicly accessible
- Read-only data display
- No sensitive operations exposed
- CORS configured for Render domain only

9.2 Admin Dashboard (/viking)

- Requires username/password authentication
- Admin credentials stored securely
- Session-based access control

---

10. TROUBLESHOOTING

10.1 Common Issues

Issue	Solution
404 on /getvkng	Ensure frontend is deployed with latest code
API errors	Check backend logs in Render dashboard
MongoDB connection	Verify IP whitelist in MongoDB Atlas
Stale data	Check MT5 bridge sync status

10.2 Support Contacts

- **Render Support:** support@render.com
- **MongoDB Atlas:** support.mongodb.com

---

11. QUICK REFERENCE

Production URLs

Service	URL
VKNG Public Dashboard	https://fidus-investment-platform.onrender.com/getvkng
VKNG Admin Dashboard	https://fidus-investment-platform.onrender.com/viking
FIDUS Platform	https://fidus-investment-platform.onrender.com
Backend API	https://fidus-api.onrender.com

Admin Credentials (VKNG /viking)

Field	Value
Username	admin

Password   Password123

---

12. RENDER API ACCESS

API Key: rnd_zyltqZyReSorlDoVp4hs8LdmJPo3

Get Service Status

```
curl -H "Authorization: Bearer rnd_zyltqZyReSorlDoVp4hs8LdmJPo3" \
  "https://api.render.com/v1/services"
```

Trigger Deploy

```
curl -X POST \
  -H "Authorization: Bearer rnd_zyltqZyReSorlDoVp4hs8LdmJPo3" \
  "https://api.render.com/v1/services/[SERVICE_ID]/deploys"
```

---

13. FILE STRUCTURE

Frontend Components (VKNG)

```
/app/frontend/src/
├── components/
│   ├── GetVKNGPublic.js    # Public dashboard (NO AUTH)
│   ├── VikingApp.js        # Admin dashboard (AUTH)
│   ├── VikingLogin.js      # Login page
│   └── VikingDashboard.js  # Main dashboard UI
├── index.js                # Route detection & loading
└── App.js                  # Main app routes
```

Backend Routes (VKNG)

```
/app/backend/
├── routes/
│   └── viking.py           # All VKNG API endpoints
└── server.py               # FastAPI app setup
```

Document End

APPENDIX: Current System Status

Component	Status	Last Verified
Frontend (Render)	✓ Running	Jan 23, 2026
Backend (Render)	✓ Running	Jan 23, 2026
MongoDB Atlas	✓ Connected	Jan 23, 2026
/getvkng Route	✓ Working	Jan 23, 2026

CORE Strategy	✓ Active	Jan 23, 2026
PRO Strategy	✓ Active	Jan 23, 2026