

**A Gamified, Scientifically-Backed Learning Platform  
Built with Next.js, Prisma, and SM-2**



# QuizView

# **MOTIVATION: DEFEATING THE FORGETTING CURVE**

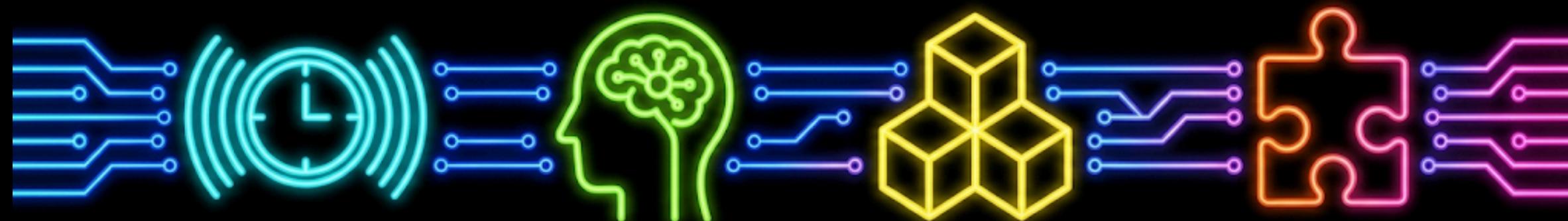
- **The Problem:** IT interviews require recalling massive amounts of highly technical data under pressure.
  - **The Flaw:** Traditional studying leads to the 'Forgetting Curve,' where 70% of information is lost within 24 hours.
  - **The Solution:** QuizView shifts focus from 'binge reading' to high-focus, gamified drill sessions.



**QuizView**

# SCIENTIFIC SUSTENTATION

- **Spaced Repetition (SM-2 Algorithm):** Dynamically predicts when you will forget a topic and schedules reviews (Overdue > New > Scheduled).
- **Active Recall:** The brain builds stronger pathways by retrieving answers before seeing explanations.
- **Interleaving Practice:** 'Remix Mode' mixes Backend, Frontend, and SQL questions to prevent mental mechanization.
- **Microlearning:** Sessions capped at 5-10 questions to eliminate cognitive overload.

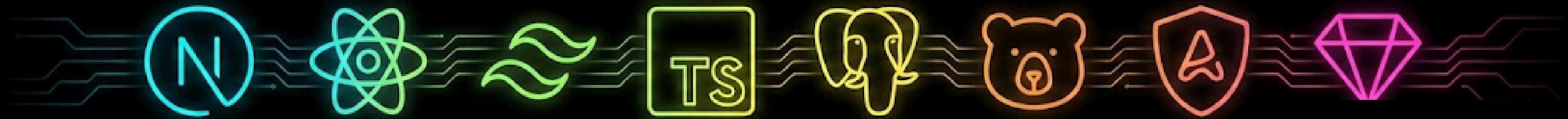


# **INTERACTIVE PRACTICE MODES**

- **Quiz Simple:** Traditional 4-option multiple choice.
- **True/False:** Rapid-fire technical assertions.
- **Fill the Blank:** Complete the missing logic in code snippets.
- **Remix Mode:** A cross-topic virtual session built only from questions you have previously studied.
- **Gamification:** Progress tracking and 'Streaks' requiring 24-hour engagement.

# **MODERN TECH ARCHITECTURE**

- **Frontend:** Next.js 14 (App Router), React 18, Tailwind CSS
- **Language:** Strict Mode TypeScript (Zero 'any' policy).
- **Backend/Data:** PostgreSQL via Prisma ORM.
- **State:** Zustand (Local Session) & Hookstate.
- **Auth:** Auth.js credentials gate with real-time Zod OWASP-compliant validation.



# **SECURITY & QUALITY ASSURANCE**

- **Brute-Force Protection:** Per-IP rate limiting (10 attempts/15 min).
- **QA Gates:** Husky pre-commit hooks block bad code.
- **Coverage Strategy (100/80/0):** 100% on Core SM-2 logic, 80% on UI components.
- **Testing Stack:** **Vitest** (Unit), **React Testing Lib** (Component),  
**Playwright** (E2E).
- **Vulnerability Scanning:** Automated **Snyk** CI/CD pipeline.

# **DEPLOYMENT & CI/CD ON VERCCEL**

- **Package Manager Constraints:** Strict use of pnpm v9+ to ensure CI parity.
- 
- **Environment Variables:** Split between Development, Preview, and Production scopes (**Database URLs, Auth Secrets**).
- 
- **Database Migrations:** Prisma schema pushes and seed operations automated during build.
- 
- **Versioning:** In-app versions perfectly synced with package.json and Semantic Versioning CHANGELOGs.

