### **TDD vs BDD vs FDD – A Comparative Infographic**

#### **🔴 TDD – Test-Driven Development**

**Approach:** 🧪 Write tests *before* writing the actual code.  
 🔁 Cycle: **Red (fail) → Green (pass) → Refactor**

**Benefits:** ✅ High code quality  
 ✅ Fewer bugs  
 ✅ Better test coverage  
 ✅ Easy to maintain and refactor

**Best For:** 💻 Back-end logic  
 🔒 Critical systems  
 🧪 Developers focused on quality and precision

#### **🔵 BDD – Behavior-Driven Development**

**Approach:** 💬 Write tests in plain language based on **user behavior**.  
 🧑‍🤝‍🧑 Collaboration between devs, testers, and business teams

**Example Format:** **Given-When-Then**

*Given a user is logged in, when they click 'Logout', then they should be logged out.*

**Benefits:** ✅ Aligns with business goals  
 ✅ Improves communication  
 ✅ User-focused functionality  
 ✅ Tests act as documentation

**Best For:** 📱 User-facing apps  
 👨‍👩‍👧‍👦 Cross-functional teams  
 🌐 Agile environments

#### **🟢 FDD – Feature-Driven Development**

**Approach:** 🧱 Develop software **feature-by-feature**, each with clear value  
 🔧 Each feature: **Plan → Design → Build → Test**

**Benefits:** ✅ Clear progress tracking  
 ✅ Scalable for large teams  
 ✅ Well-structured documentation  
 ✅ Client-focused delivery

**Best For:** 🏢 Enterprise-level applications  
 📅 Long-term projects  
 🧩 Large development teams

v