QnGenius JavaFX Enhancement Roadmap

Immediate Fixes & Improvements

1. Critical Issues

- **Role-based Authentication**: Currently missing proper role checking after login
- **Database Connection Management**: No connection pooling - can lead to connection leaks
- **Error Handling**: Inconsistent error handling across controllers
- **User Management**: Incomplete User and Role model classes

2. Architecture Enhancements

A. Service Layer Implementation

```java
// Create service classes to separate business logic from controllers
com.qngenius.service/
├── UserService.java
├── CourseService.java
├── QuestionService.java
├── BlueprintService.java
└── ExamService.java
```

B. Enhanced Security

- JWT token-based authentication
- Role-based authorization annotations
- Session management
- Password policies

C. Configuration Management

- Move from hardcoded file paths to classpath resources
- Environment-specific configurations
- Database connection pooling (HikariCP)

Feature Enhancements

3. Question Management System

- **Bulk Question Import**: Excel/CSV import functionality

- **Question Categories**: Tag-based categorization
- **Question History**: Version tracking and audit logs
- **Advanced Search**: Multi-criteria question filtering
- **Question Validation**: Duplicate detection and quality checks

4. Enhanced Blueprint System

- **Visual Blueprint Designer**: Drag-and-drop interface
- **Template Management**: Pre-defined blueprint templates
- **Intelligent Question Selection**: ML-based question recommendation
- **Blueprint Analytics**: Statistics and insights

5. User Experience Improvements

- **Modern UI/UX**: Material Design or modern theme
- **Dashboard Analytics**: Charts and statistics
- **Notification System**: In-app notifications
- **Keyboard Shortcuts**: Power user features
- **Responsive Design**: Better window resizing handling

6. Advanced Features

- **Question Paper Variants**: Multiple versions of same paper
- **Collaborative Features**: Multi-user question creation
- **Integration APIs**: External system integration
- **Mobile Companion**: JavaFX mobile support or web interface
- **Cloud Backup**: Automated backup solutions

Technical Improvements

7. Performance & Scalability

- **Lazy Loading**: For large datasets
- **Caching Strategy**: Redis/In-memory caching
- **Async Operations**: Non-blocking UI operations
- **Database Optimization**: Query optimization and indexing

8. Testing Strategy

- **Unit Tests**: JUnit 5 with Mockito
- **Integration Tests**: TestFX for UI testing

- **Database Tests**: H2 in-memory testing
- **Performance Tests**: Load testing for database operations

9. DevOps & Deployment

- **Build Automation**: Maven/Gradle with proper profiles
- **Packaging**: Native installers with jpackage
- **Docker Support**: Containerized deployment
- **CI/CD Pipeline**: Automated testing and deployment

Implementation Priority

Phase 1 (Immediate - 2-3 weeks)

1. Fix role-based authentication
2. Implement proper error handling
3. Add connection pooling
4. Complete User and Role models
5. Improve UI consistency

Phase 2 (Short-term - 1-2 months)

1. Service layer implementation
2. Enhanced question management
3. Bulk import functionality
4. Advanced search and filtering
5. Blueprint improvements

Phase 3 (Medium-term - 2-3 months)

1. Modern UI redesign
2. Analytics dashboard
3. Advanced blueprint designer
4. Collaboration features
5. Mobile/web integration

Phase 4 (Long-term - 3-6 months)

1. Machine learning integration
2. Advanced analytics
3. Cloud features

4. API development
5. Enterprise features

Development Guidelines

Code Quality Standards

- Follow Java naming conventions
- Implement proper logging (SLF4J + Logback)
- Use design patterns appropriately
- Maintain comprehensive documentation
- Code coverage > 80%

UI/UX Guidelines

- Consistent color scheme and typography
- Responsive design principles
- Accessibility compliance
- User feedback for all actions
- Loading indicators for long operations

Security Best Practices

- Input validation and sanitization
- SQL injection prevention
- Secure password storage
- Session timeout handling
- Audit logging