General good backend coding practices

- Code should be safe
  - Sensitive information should be private
  - Use gitignore to avoid committing sensitive files
  - Things like passwords and keys should never be hard coded
- Database normalization
  - Database should be structured in a way that reduces redundancy
  - Reduces the risk of errors when reading/writing data
  - Database is easier to query
- Code should be readable
  - Use relevant variable names
  - Use clear formatting
  - Code should be easy to understand, even if it is less compact
  - No magic numbers
- Have modular code
  - Makes code easier to modify
  - Makes building new projects faster, as modules can be reusable
  - Allows a large project to be broken down into small manageable pieces
- Error Handling
  - Return useful error messages for debugging
  - Gracefully handle non-critical errors without crashing program
- Validate inputs
  - Check that input is correct datatype/length/structure etc.
  - Sanitize user input to prevent code injection
- Document code
  - Allows code to be understood
  - Makes code easier to maintain
  - Improves team collaboration
- Code should be easy to expand upon
  - Allows software to be modified as the design vision changes

  - Allows fast response to feedback
- Test everything
  - Reduces risk
  - Ensures design requirements are met
  - Extremely important when using LLM code
- Don't overuse dependencies
  - Can cause licensing issues
  - Dependencies must be maintained
  - Can cause unnecessary overhead and security risks