

group-project

Sprint 3 Guides

These guides support Sprint 3. Read them as you work through the requirements.

Design Patterns & Architecture

Guide	When to Read
The Observer Pattern	Requirements 2, 4, 6 – reacting to GameState.OVER, scoring from PropertyChangeEvents, and publishing new states in back-end features
Model-View-Controller (MVC)	All requirements – maintaining clean separation as you add scoring logic and extra features

GUI & Events

Guide	When to Read
Handling Key Events	Requirement 3 – key binding display; Requirement 5 – optional user-changeable key bindings
Animation with javax.swing.Timer	Requirements 2 & 4 – stopping the timer on game over, adjusting delay for leveling
Custom Painting with Java 2D	Requirements 2 & 6 – visual effects for end game and Panic/Hyper modes

Back-End Extension

Guide	When to Read
Java Enums	Requirement 6 – adding new GameState enum constants for Panic Mode or Hyper Piece Mode

Code Quality

Guide	When to Read
Linters and Code Quality	Requirement 7 – resolving all Checkstyle and IntelliJ inspection warnings
Checkstyle Rules Reference	Requirement 7 – understanding specific Checkstyle rules that may be flagged
IntelliJ Inspections Reference	Requirement 7 – understanding specific IntelliJ inspection warnings

References

Primary Texts:

- Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley. – Observer pattern (pp. 293-303).
- Freeman, E., & Robson, E. (2020). *Head First Design Patterns* (2nd ed.). O'Reilly. – Chapter 2: Observer Pattern.
- Horstmann, C. S. (2022). *Core Java, Volume I: Fundamentals* (12th ed.). Oracle Press. – Swing event handling, Timer, custom painting, file I/O.
- Bloch, J. (2018). *Effective Java* (3rd ed.). Addison-Wesley. – Item 34: Use enums instead of int constants; Item 42: Prefer lambdas to anonymous classes.

Language Documentation:

- [Oracle JDK 25: PropertyChangeListener](#) – Observer interface for property changes
- [Oracle JDK 25: javax.swing.Timer](#) – EDT-safe timer, setDelay for leveling
- [Oracle JDK 25: KeyListener](#) – Keyboard event interface

- [Oracle JDK 25: javax.sound.sampled](#) – Audio playback API (for Sound/Music feature)
- [Oracle JDK 25: java.util.prefs.Preferences](#) – Persistent user preferences (for themes, high scores)

Tutorials:

- [Tetris Back-End API Documentation](#) – Game model Javadoc
- [Oracle: Playing Back Audio](#) – Official audio playback tutorial