

Wordle Game - Complete File Structure

```
wordle-game/
|
+-- src/
|   +-- main/
|   |   +-- java/
|   |   |   +-- com/
|   |   |   |   +-- game/
|   |   |   |   |   +-- wordle/
|   |   |   |   |   |   +-- domain/
|   |   |   |   |   |   |   +-- model/
|   |   |   |   |   |   |   |   +-- Word.java
|   |   |   |   |   |   |   |   +-- Guess.java
|   |   |   |   |   |   |   |   +-- GuessResult.java
|   |   |   |   |   |   |   |   +-- LetterResult.java
|   |   |   |   |   |   |   |   +-- GameState.java
|   |   |   |   |   |   |   |   +-- Game.java
|   |   |   |   |   +-- exception/
|   |   |   |   |   |   +-- InvalidWordException.java
|   |   |   |   |   |   +-- GameOverException.java
|   |   |   |   |   |   +-- WordNotFoundException.java
|   |   |   |   +-- factory/
|   |   |   |   |   +-- GameFactory.java
|
|   +-- service/
|       +-- repository/
|           +-- WordRepository.java (interface)
|           +-- WordListProvider.java (interface)
|
|       +-- validation/
|           +-- WordValidator.java (interface)
|           +-- WordValidationRule.java (interface)
|
|       +-- engine/
|           +-- GameEngine.java (interface)
|           +-- FeedbackCalculator.java (interface)
|
|       +-- session/
|           +-- GameSession.java (interface)
|           +-- GameSessionListener.java (interface)
|
|       +-- infrastructure/
|           +-- repository/
|               +-- FileWordRepository.java
|               +-- InMemoryWordRepository.java
|               +-- CachedWordRepository.java
|
|           +-- validation/
|               +-- DictionaryWordValidator.java
|               +-- LengthValidationRule.java
|               +-- AlphabeticValidationRule.java
|               +-- CompositeWordValidator.java
|
|           +-- engine/
|               +-- StandardGameEngine.java
|               +-- StandardFeedbackCalculator.java
|
|           +-- session/
|               +-- SinglePlayerGameSession.java
|               +-- StatisticsTrackingSession.java
|
|           +-- strategy/
|               +-- WordSelectionStrategy.java (interface)
|               +-- RandomWordSelectionStrategy.java
|               +-- SequentialWordSelectionStrategy.java
|               +-- DifficultyBasedSelectionStrategy.java
|
|           +-- presentation/
|               +-- ui/
```




Key Package Responsibilities

domain/ - Core Business Logic (Framework Independent)

- model/**: Pure domain entities and value objects
- exception/**: Domain-specific exceptions
- factory/**: Object creation with business rules

service/ - Business Logic Interfaces

- repository/**: Data access abstractions
- validation/**: Validation strategy interfaces
- engine/**: Core game logic interfaces
- session/**: Game session management interfaces

infrastructure/ - Technical Implementations

- repository/**: Concrete data access (files, memory, cache)
- validation/**: Concrete validation implementations
- engine/**: Game logic implementations
- session/**: Session management implementations
- strategy/**: Word selection algorithms

presentation/ - User Interface Layer

- ui/**: UI abstractions and console implementation
- controller/**: Application flow control
- input/**: User input handling
- formatter/**: Output formatting strategies

config/ - Application Configuration

- Configuration classes and builders
- Dependency injection setup (if using DI framework)

Design Benefits

- Clear separation of concerns** - Each package has a single responsibility

2. **Testability** - Easy to test each layer independently
3. **Extensibility** - New implementations can be added without modifying existing code
4. **Maintainability** - Easy to locate and modify specific functionality
5. **Framework independence** - Domain layer has no external dependencies
6. **GUI-ready** - Easy to add Swing/JavaFX UI by implementing GameUI interface