The Art Dealer Game : Agile Project Documentation

Submitted by

Group-6

Durga Vijaya Ramaraju Pamidi

SU25-CPSC-60500-001

**Instructor:** Dr. Fadi Wedyan

Lewis University

Table of Contents

[1.0 Introduction 2](#_Toc27933133)

[1.1 Purpose 2](#_Toc27933134)

[1.2 Game Concept 2](#_Toc27933135)

[2.0 Process Model 3](#_Toc27933136)

[3.0 Use Cases 3](#_Toc27933137)

[3.1 Use Case 1: Start game 3](#_Toc27933138)

[3.2 Use Case 2: Offer cards 3](#_Toc27933139)

[3.3 Use Case 3: Buyers Response 4](#_Toc27933140)

[3.4 Use Case 4: Make a Guess 4](#_Toc27933141)

[3.5 Use Case 5: Win condition 4](#_Toc27933142)

[4.0 UML Model 5](#_Toc27933144)

[4.1 Use Case Diagram 5](#_Toc27933145)

[4.2](#_Toc27933147) Class Diagram…………………………………………………………………………………………………………………………..6

[4.3 Activity Diagram 6](#_Toc27933149)

[5.0 Customer Journey Map 6](#_Toc27933150)

5.1 Entry: Starting the experience………………………………………………………………………………………………………..6

5.2 Exploration: Payful Experimentation………………………………………………………………………………………………7

5.3 Discovery : Learning Through Process……………………………………………………………………………………………7

5.4 Mastery : Solving the Puzzle………………………………………………………………………………………………… ………7

5.5 Replay: Motivation to continue …………………………………………………………………………………………………….7

[6.0 UI Mock-up 8](#_Toc27933152)

6.1 User Interface………………………. ……………………………………………………………………………………………………..8

6.2 Guess Input Field………………………. ……………………………………….……………………………………………………….8

6.3 Dealer Response Model for correct guess…………………………………………………………………………………….9

[7.0 Testing Strategy 9](#_Toc27933153)

[7.1 Unit Testing 9](#_Toc27933154)

[7.2 Integration Testing 10](#_Toc27933155)

[7.3 Usability Testing 10](#_Toc27933156)

[7.4 Validation Testing 11](#_Toc27933157)

1. Introduction
   1. Purpose

This document outlines the design of *The Art Dealer Game*, a pattern-based educational simulation designed for classroom use by students in grades K–8. It bridges playful interaction with structured logic, helping students build essential cognitive skills in a highly engaging environment. The agile documentation approach ensures adaptability—allowing educators and developers to expand, iterate, and refine the game over time without overwhelming complexity. Ultimately, this project aims to provide a modular, scalable framework that meets both pedagogical goals and software best practices.

## Game Concept

In *The Art Dealer Game*, each student takes on the role of a gallery owner curating artwork from a shuffled deck of traditional playing cards. Their goal is to successfully sell selections to a virtual art buyer whose preferences follow a hidden but consistent pattern. As cards are accepted or rejected, players must rely on observation, trial-and-error, and logical reasoning to uncover the rule behind the buyer’s choices.

Every round invites a fresh opportunity to experiment with different card combinations while mentally tracking the buyer’s responses. Once students believe they’ve deduced the rule, they select a pattern from a guided list and test their hypothesis. A win condition is met when the student presents four perfectly matching cards and correctly identifies the buying pattern—prompting visual feedback such as balloons or celebration animations to reinforce success.

To match students’ developmental stages, the game’s difficulty scales progressively by grade level. Early learners explore basic attributes like color and suits, middle grades navigate numerical logic and arithmetic constraints, while older students delve into abstract rules and optional multiplayer modes that demand strategic thinking. This tiered approach ensures the game remains accessible, challenging, and rewarding across diverse age groups.

The development of *The Art Dealer Game* follows an **evolutionary process model**, which blends the flexibility of **agile methods** with the structured risk control of the **spiral model**. This hybrid approach is ideal for educational software where requirements evolve based on classroom feedback and cognitive progression.

# Process Model

The development of *The Art Dealer Game* follows an **evolutionary process model**, which blends the flexibility of **agile methods** with the structured risk control of the **spiral model**. This hybrid approach is ideal for educational software where requirements evolve based on classroom feedback and cognitive progression.

**Key Characteristics:**

* **Rapid Prototyping:** Each version of the game introduces new patterns and mechanics, allowing educators to test and refine learning outcomes in real time.
* **Incremental Complexity:** Game logic and pattern difficulty scale with grade levels, ensuring age-appropriate challenges and sustained engagement.
* **Feedback Integration:** Student performance and teacher observations guide the next iteration, helping improve usability, clarity, and educational value.
* **Risk Management:** Spiral principles help identify potential design pitfalls early—such as overly complex patterns or inaccessible UI—and address them before full deployment.
* **Multiplayer Expansion:** Optional peer-to-peer gameplay for older students is introduced in later iterations, once core mechanics are validated.

This model ensures that the game remains adaptable, scalable, and aligned with both pedagogical goals and technical best practices.

# Use Cases

The following six use cases were identified by the development team and educational stakeholders as the foundational requirements for delivering a fully functional prototype of *The Art Dealer Game*. Each use case represents a key interaction between the student and the game system, enabling gameplay progression and learning outcomes. As the game evolves, these baseline functions can be extended with additional use cases, particularly for expanded pattern sets.

Each use case includes the name of the use case, the primary actor, preconditions required, a description of the interaction, and specific acceptance criteria to verify successful implementation.

* 1. Use Case 1: Start game

**Primary Actor:** Student

**Preconditions:** The game application is installed and launched; deck is ready.

**Description:** As a student, I want to begin a new session with a fresh deck of 52 playing cards so I can start exploring the virtual art buyer’s preferences. The system should randomly shuffle the deck and initialize the game environment for pattern discovery.

**Acceptance Criteria:** I can start the game and see a complete, shuffled deck of 52 cards

on screen.

* 1. Use Case 2: Offer cards

**Primary Actor:** Student

**Preconditions:** Game session has started; deck is available and visible.

**Description:** As a student, I want to select and display four cards from my deck to offer to the art buyer so I can observe which cards are purchased. The cards should be arranged clearly for review.

**Acceptance Criteria:** The four selected cards are shown on the interface and passed to the buyer for evaluation.

* 1. Use Case 3: Buyer Response

**Primary Actor:** Virtual Art Buyer

**Preconditions:** Four cards have been selected and offered by the student..

**Description:** As a virtual art buyer, I evaluate the offered cards against an internally assigned pattern to determine which cards I wish to "purchase." My selections are highlighted for the student to observe and analyze.

**Acceptance Criteria:** The buyer selects and indicates matching cards according to the hidden pattern.

* 1. Use Case 4: Make a Guess

**Primary Actor:** Student

**Preconditions:** One or more rounds of card offering have occurred.

**Description:** As a student, I want to guess the buyer's hidden pattern using a dropdown list of available patterns so that I can test my hypothesis and potentially win the game.

**Acceptance Criteria:** The system verifies my selected pattern and provides feedback—correct, incorrect, or remaining guesses.

* 1. Use Case 5: Win Condition

**Primary Actor:** Student

**Preconditions:** A valid guess has been submitted and four matching cards were previously offered.

**Description:** As a student, I want confirmation that I’ve successfully identified the pattern and offered a perfect set so I can win the game and receive positive feedback.

**Acceptance Criteria:** If all offered cards fit the pattern and the guess is correct, a celebratory animation is triggered.

# UML Model

* 1. Use Case Diagram

In The Art Dealer Game, the main actor is the **student**, who interacts with the game in four major ways: starting a new session, offering cards, making guesses about the buyer’s pattern, and accessing multiplayer mode (for older students). Additionally, a hidden use case is triggered when the student correctly identifies the pattern while offering matching cards—this is the **win condition**, which results in a celebratory moment and ends the round.

## Class Diagram

### Below figure shows *The Art Dealer Game*, class diagram showing components like card management, pattern validation

### class diagram for the art dealer game project in software documentation

Figure 1. Class diagram for the Art dealer game

## Activity Diagram

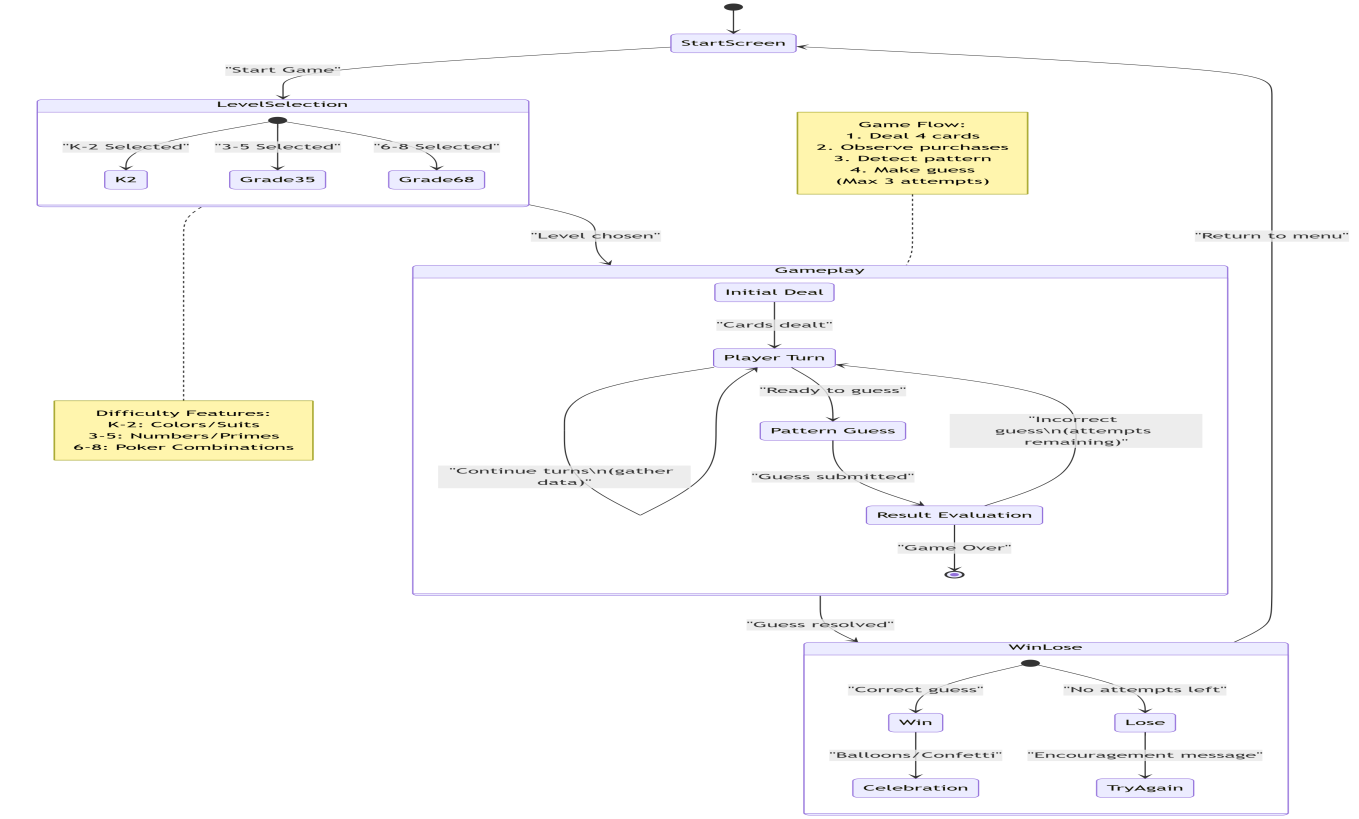


Figure 2. Activity diagram for the Art dealer game

The activity diagram shown in above figure 2 presents a more detailed description of the high-level behavior of the Art dealer game.

## Customer Journey Map – How Students Engage with the Game

## 5.1 Entry: Starting the Experience

Students begin by launching the Art Dealer Game and selecting their grade level. The interface is warm and inviting, using age-friendly visuals, audio cues, and simple navigation. This helps young learners feel safe and excited, reducing any fear of complexity. At this stage, students begin with curiosity, eager to explore what the game is about.

## 5.2 Exploration: Playful Experimentation

Once in the game, students select four cards from the virtual deck and observe how the software (the art dealer) reacts. They try different combinations to test what the dealer might be "buying." No penalties are imposed; the game encourages fearless play and hands-on learning. This trial-and-error approach invites discovery while keeping things fun.

## 5.3 Discovery: Learning Through Patterns

Over time, patterns start to emerge. Maybe the dealer prefers red cards, or only selects prime numbers. Students begin to understand the logic behind the dealer’s decisions and develop strategies to narrow down the pattern. The experience is empowering—it teaches computational thinking and analytical reasoning organically.

## 5.4 Mastery: Solving the Puzzle

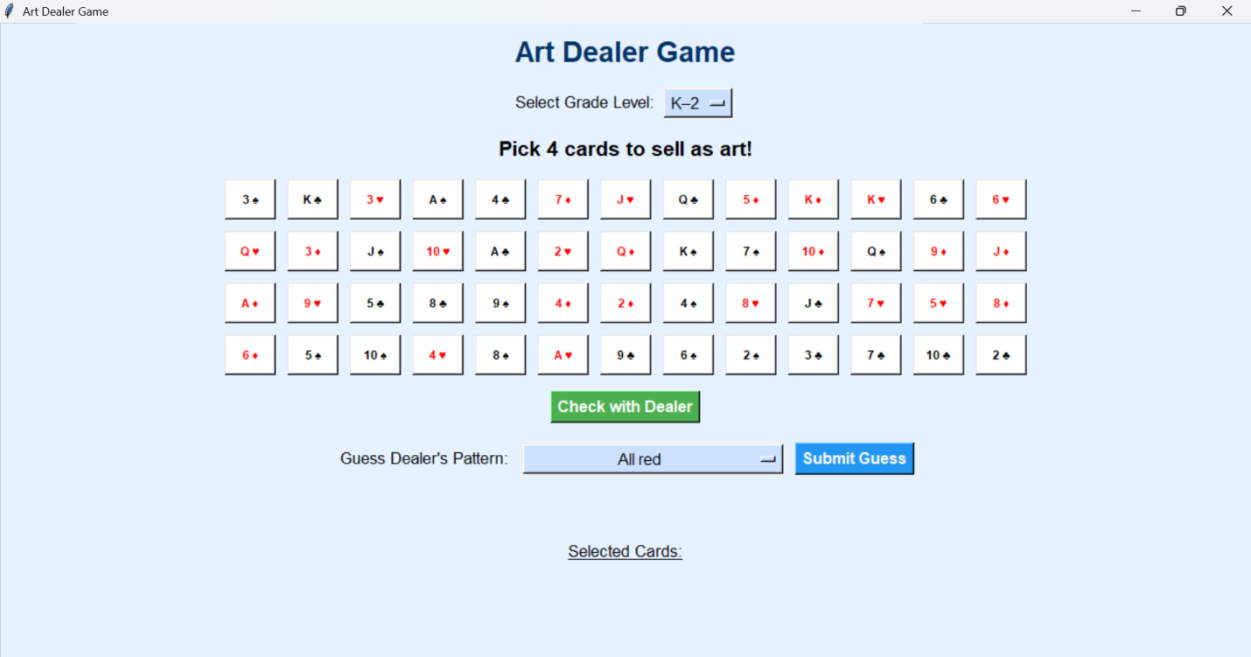
Eventually, a student successfully selects four cards that match the dealer’s hidden pattern and accurately guesses the rule from the provided options. The game rewards this moment with visual celebration—balloons, animations, or applause sounds. Students feel validated, proud of their reasoning, and encouraged to keep playing.

## 5.5 Replay: Motivation to Continue

After solving a pattern, students can jump into a new round with tougher challenges or even switch to multiplayer mode. Their engagement deepens with confidence, and they’re now motivated not just to play—but to learn better and perform more cleverly in the next round.

## 6.0 UI Mock-up

## 6.1 **User Interface**: Game User Interface



## 6.2 Guess Input Field: Text or multiple-choice pattern guesses



## 6.3 Dealer Response Panel for correct Guess: Shows matches visually (balloons)

## 

## 7.0 Testing Strategy – Ensuring Quality, Usability, and Educational Value

## 7.1 Unit Testing – Validating the Building Blocks

Unit testing focuses on individual components to ensure that each logical module functions correctly in isolation.

* **Card Selection Logic**: Verify that the user can select exactly four cards per turn. Check for constraints preventing over-selection and ensure cards are reset between rounds.
* **Pattern Matching Algorithms**: Test the pattern engine for various grade levels. Confirm it correctly identifies matches based on suit, rank, color, or mathematical relationships.
* **Feedback Generation**: Ensure that when cards are evaluated, accurate and context-sensitive feedback is displayed (e.g., “3 out of 4 cards matched”). Validate visual cues like balloons or hints.

Tools recommended: unittest, pytest for Python language.

## 7.2 Integration Testing – Verifying System Cohesion

Integration testing ensures that separate modules interact smoothly as part of the full gameplay experience.

* **End-to-End Game Flow**: Confirm the user journey from selection → evaluation → guessing → feedback → replay. Every transition should be smooth and maintain game state accurately.
* **Role Transitions**: Validate that students can switch between dealer and seller modes without breaking the game logic or UI flow.

## 7.3 Usability Testing – Testing for Real-World Interaction

Usability testing ensures that students with varying cognitive and sensory abilities can enjoy and learn from the game.

* **Cognitive Walkthroughs**: Conduct walkthroughs using the three personas—Aarav, Zara, and Leo. Observe how each interacts with the interface, interprets feedback, and navigates challenges.
* **Accessibility Checks**: Evaluate compatibility with screen readers, colorblind-friendly palettes, and simplified text settings for early learners. Ensure that instructions and feedback don’t assume prior technical knowledge.
* **User Error Recovery**: Simulate typical classroom mis-clicks (rapid key presses, accidental submissions) and confirm that the system responds gracefully.

Usability feedback from pilot classrooms can be used to iterate UI improvements.

## 7.4 Validation Testing – Ensuring Educational Effectiveness

Validation testing confirms that the game meets its intended learning outcomes and aligns with curricular standards.

* **Educator Reviews**: Share prototype builds with teachers to assess whether the game supports logic, reasoning, and pattern recognition objectives. Gather qualitative feedback.
* **Curriculum Alignment**: Match patterns and game tasks against grade-level benchmarks (e.g., recognizing numerical patterns in Grade 5, working with combinations in Grade 7).
* **Pattern Calibration**: Test the perceived difficulty of patterns across age groups and refine them based on actual student performance data to avoid frustration or boredom.