



UNO Flip Remix: Revision 1 Demo

An exciting walkthrough of our UNO Flip Remix game.

This presentation will demonstrate our implementation of the game, the challenges we faced, and the solutions we developed.

Presented by: Team 24

Meet the team



Kevin Ishak

Project Manager



Zain Garada

UI/UX Expert



Mingyang Xu

Back End Developer



Jianhao Wei

Full Stack Developer

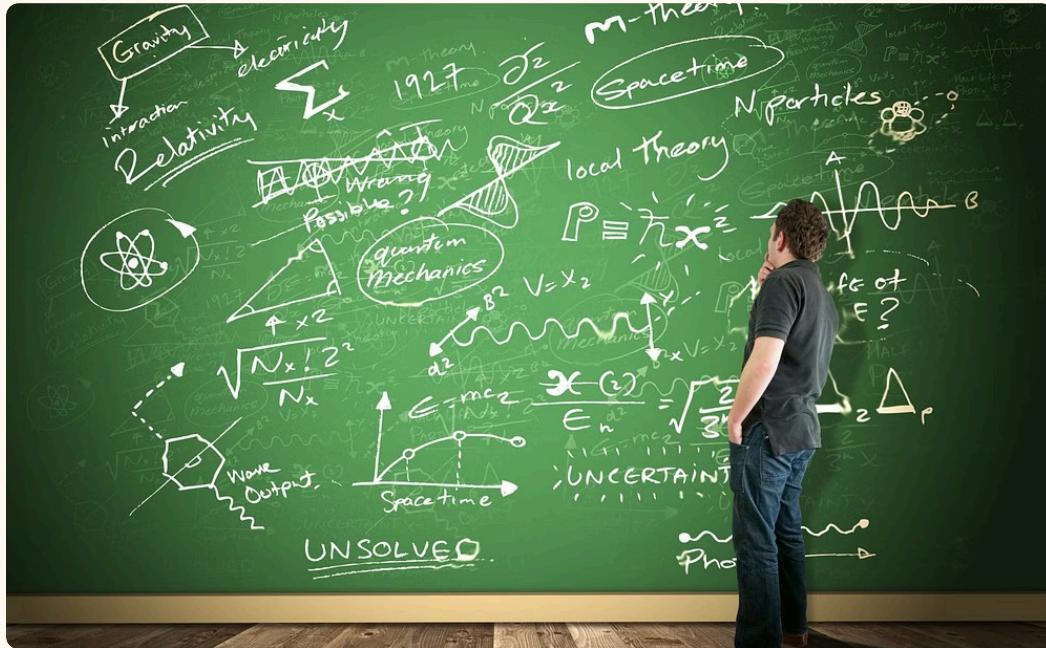


Made with Gamma

Roadmap



Project Goals



Problem Statement

The current version of UNO Flip **fails** to offer an enjoyable online experience.

Existing implementations lack polish and accessibility.



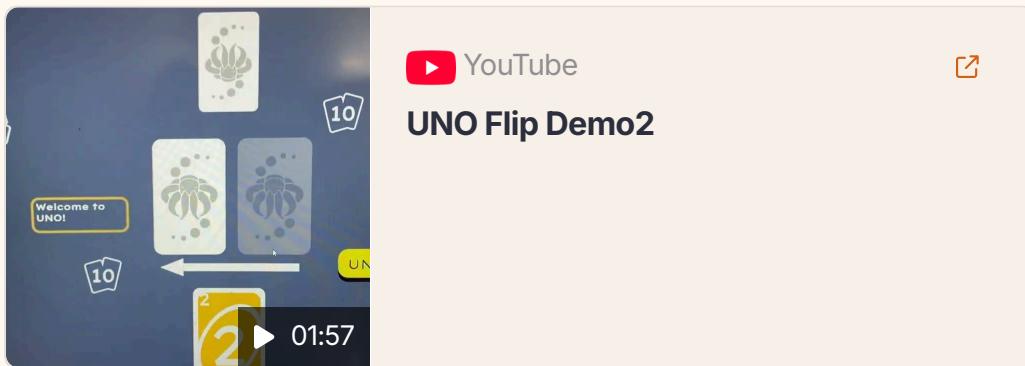
Our Solution

A complete digital UNO Flip Remix with **multiplayer** and **rule-based AI**.

Built for fun, strategy, and smooth play — whether **solo** or with **friends**.

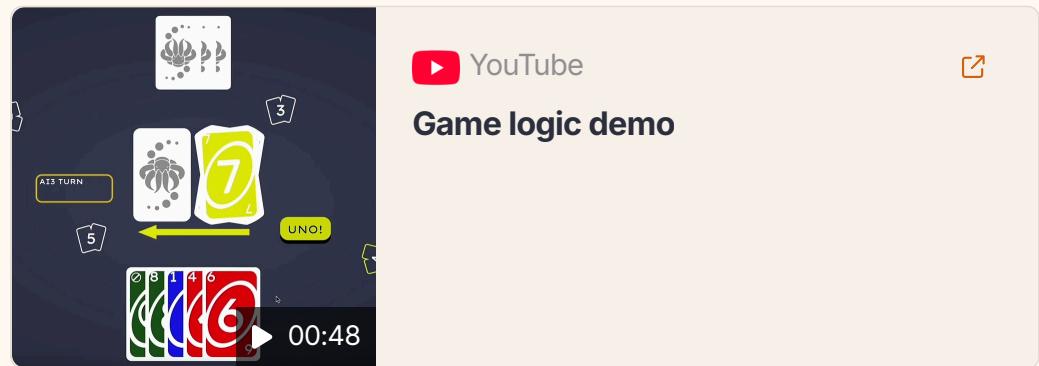
Live Demo

Multiplayer Synchronization



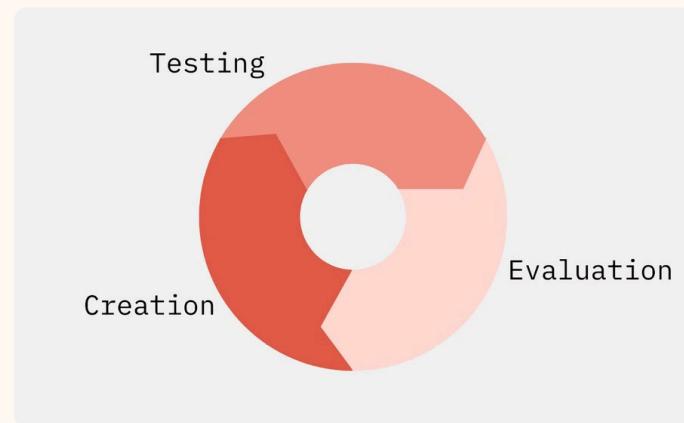
Demonstrates the initial setup of the game, including player registration and game logic.

End Game



Shows a player successfully emptying their hand and winning the round.

Design Process



Iterative Design Process

Built → Tested → Refined



Feedback

Refinements based on direct feedback



Documentation

Traceable and Transparent

Comparison to Competitors

UNO Flip Remix (Team 24)

Our remix enhances the classic UNO Flip with improved multiplayer synchronization, rule-based AI, and a polished user experience.

- Multiplayer support
- Rule-based AI
- Smooth user experience



UNO Flip (Official App)

The official UNO Flip app provides a basic digital version of the game but often lacks the strategic depth and engagement of a dedicated remix.

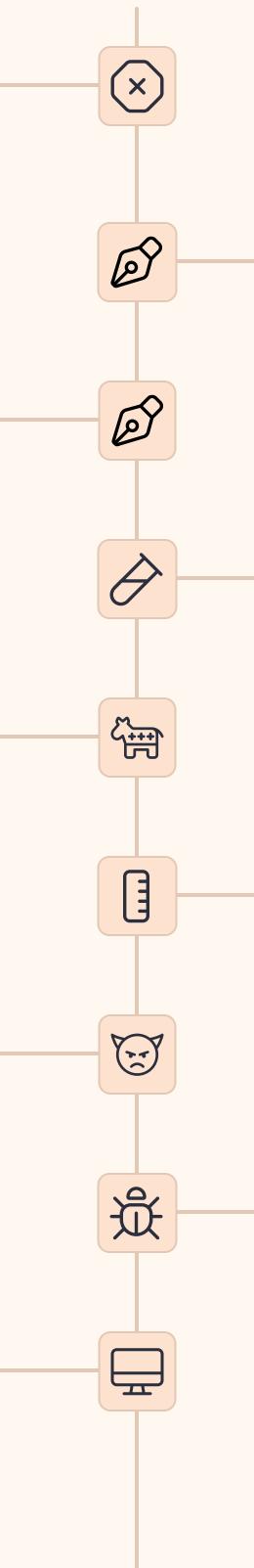
- Basic digital version
- Poor UI/UX
- Can be buggy



Story - From Zero to Flip

Problem Statement & Development Goals

Defining the challenges and objectives for UNO Flip Remix.



Software Requirements Specification

Documenting requirements.

Hazard Analysis

Identifying potential hazards.

Verification and Validation Plan

Planning the testing process to ensure quality and reliability.

Proof of Concept Demo

Presenting a preliminary demonstration of the game's functionality.

Design Documents

Creating detailed design documents for the game.

Rev0 Demo

First iteration of the game demonstration.

Verification and Validation Report

Reporting the results of the verification and validation process.

Final Demo Rev1

The final demonstration of UNO Flip Remix.

SRS Alignment and Core Goals



1

Multiplayer Synchronization

- **Real-time turn-based play** with minimal latency
- Reliable **TCP** communication
- Synchronizes **game state** across **all** players
 - *Linked to Requirements: MSR1, MSR2, TMR1*

2

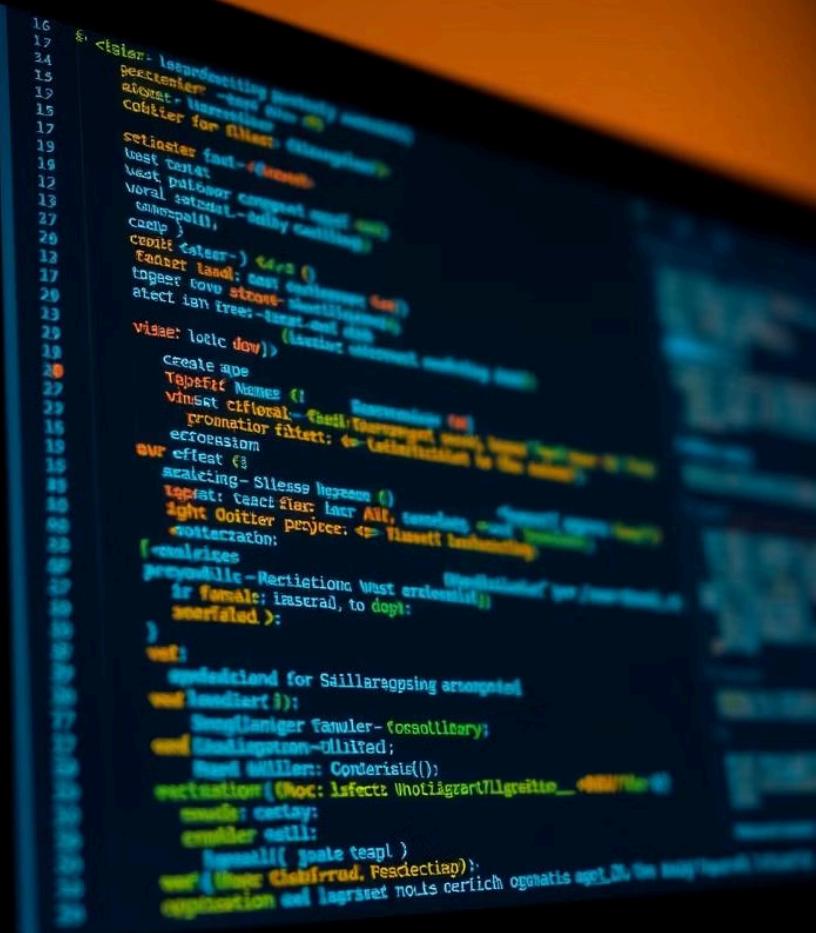
Accurate Rule Enforcement

- **Enforces** all UNO Flip card **rules**, including flip-side dynamics
- Prevents **illegal** plays and guides users through UI feedback
- Flip card swaps deck and hands in real time
 - *Linked to Requirements: GRE1, GRE2, CP4, CP5*

3

Strategic AI Gameplay

- Rule-based AI mimics player behavior
- Smart card decisions, reacts to wild cards and declares UNO
- No machine learning, but challenging enough for solo play
 - *Linked to Requirements: AIB2, UC-03, AI1*



Technical Specifications



Unity/C#

Cross-platform development

Optimized for **smooth**
UI and card
animations



Real-time Networking

Custom TCP server

Ensures **fast**,
synchronized
multiplayer turns



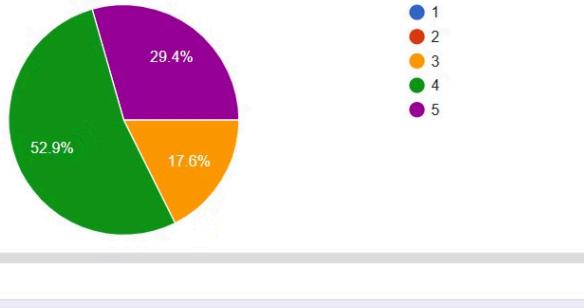
AI Algorithms

Rule-based AI

Mimics player behavior

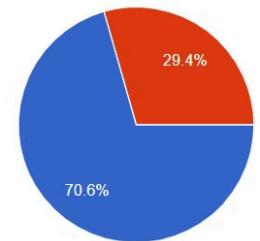
Usability Testing Feedback

How would you rate the game interface design? (1 = Very Poor, 5 = Excellent)
(17 条回复)

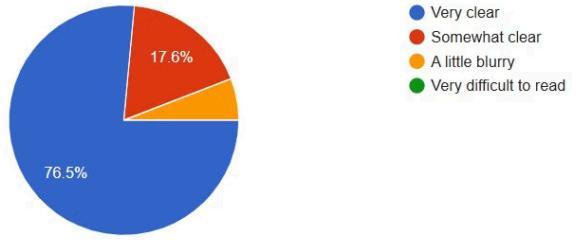


Do you find the game controls intuitive and easy to understand?
(17 条回复)

复制图表

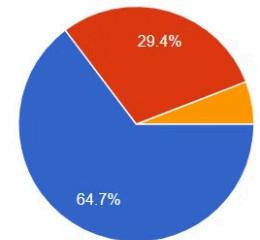


Are the text and icons clear and easy to read?
(17 条回复)



Are the drag/click actions for playing cards smooth and responsive?
(17 条回复)

复制图表



GenderMag Analysis



Friendly to all genders

Accommodating diverse personalities and preferences from all genders.



Intuitive Interface

Easy-to-follow buttons and simple text guide users.



Clear Instructions

No ambiguous instructions or redundant text.



Adaptable Usability

Easy to use for both tinkerers and those who prefer instructions.



Made with Gamma



Project Qualities

Portability

Runs seamlessly on desktop — built with Unity for maximum reach.

Low-Cost

Built entirely with free and open-source tools. No licensing.

Installability

Easy setup process

Performance

Optimized for smooth gameplay



Testing and Validation



Unit Tests

Over 50 unit tests implemented, achieving 71.8% code coverage.



Component Validation

Ensured individual components function correctly through rigorous testing.

3/10/25, 10:54 PM

Summary - Coverage Report

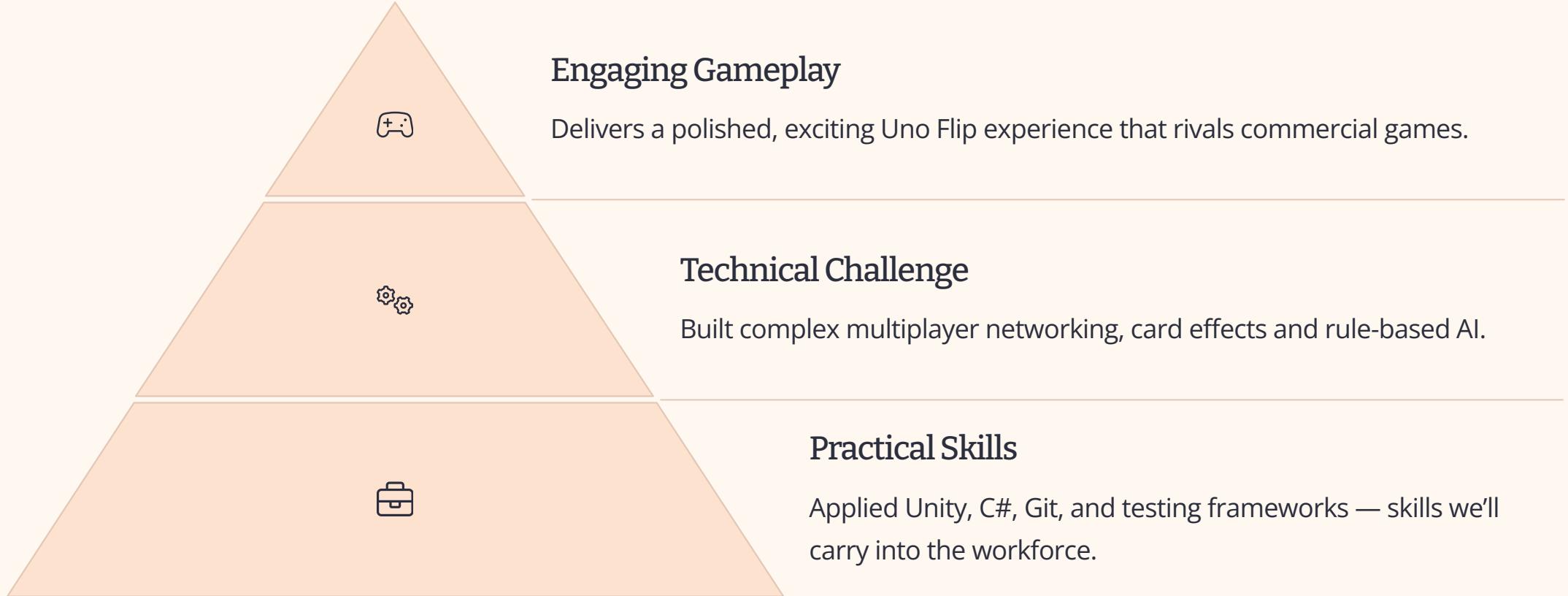
Summary

Generated on:	2025-03-10 - 9:21:15 PM
Parser:	MultiReportParser (59x OpenCoverParser)
Assemblies:	2
Classes:	17
Files:	17
Covered lines:	1486
Uncovered lines:	583
Coverable lines:	2069
Total lines:	3518
Line coverage:	71.8% (1486 of 2069)
Covered branches:	0
Total branches:	0
Covered methods:	128
Total methods:	171
Method coverage:	74.8% (128 of 171)



Made with Gamma

Impact and Takeaways



This wasn't just coursework — we built, tested, and delivered a complete game experience from the ground up.

Disadvantages & Future Work



Limited Scope

Focused on core features, some advanced functionalities were omitted.



Resource Constraints

Development time and team size limited the project's complexity.



Future Development/Documentation

Explore advanced features, improve AI, and expand platform compatibility. Update any previous documentation.



Conclusion and Questions

Thank you for joining us today. We hope you enjoyed the presentation about our project.

Now, we'd be happy to answer any questions you may have.