

# Project Proposal: Vesania

## 1. Project Title

Vesania (*Latin for mania or madness*) we picked this title because many pvp games involve the word *mania* so taking inspiration on how *Balatro* was named we picked a Latin name for our game.

## 2. Brief Overview

Vesania is a "toys-to-life" digital card game that bridges physical collectibles with virtual gameplay. Players own physical cards embedded with NFC chips which, when scanned via a mobile device, unlock unique characters and items within a web-based game. The application features a turn-based combat system including both PvP, PvE, and tournament modes. Users manage their character and item collections on the web and use their mobile devices as the primary bridge for physical integration.

## 3. Motivation

Our team is inspired by the "toys-to-life" nostalgia of franchises like *Skylanders*. We aim to implement a modern take on this by using NFC technology and web frameworks to prove that physical assets can still create immersive, tangible connections to digital gaming experiences.

## 4. Features and User Roles

- **Player/User Role:**
  - **NFC Scanning:** Use a mobile interface to scan physical card tags to "claim" and bind assets to a personal account.
  - **Collection Management:** View a personal roster of unlocked characters, items, and weapons on a web dashboard.
  - **Gameplay (PvE/PvP/Tournaments):** Engage in turn-based combat sessions (Slay the Spire-style) using unlocked digital assets.
  - **Run Persistence:** Save and resume game states (runs) via a cloud-based backend.
- **Admin/Developer Role (Internal):**
  - **Content Management:** Tools to create, edit, and balance characters, spells, and weapon stats.
  - **NFC Provisioning:** Generate unique identifiers for new physical cards.

## 5. Risks / Challenges

- **Mechanical Balance:** Ensuring 6+ characters and dozens of items are balanced for fair PvP and engaging PvE.
- **Hardware Integration:** Managing the "handshake" between mobile NFC scans and real-time web app updates.
- **Asset Creation:** Developing a cohesive art style and UI for a complex card-game interface.

## 6. Existing Related Projects

- **Skylanders / Pokémon:** Primary inspirations for physical-to-digital collection loops.
- **Slay the Spire:** Influence for the turn-based roguelike combat mechanics.
- **Difference:** Vesania offers a platform-agnostic (web-based) approach to toys-to-life, removing the need for proprietary "portal" hardware by utilizing the NFC readers already present in modern smartphones. Furthermore the cards can be used to play the game physically in person.

## 7. Intended Platform / Programming Language

- **Frontend:** Next.js (Web)
- **Backend:** Next.js, could move to node in the future
- **Database:** PostgreSQL (Primary), MongoDB (Metadata/Logs), Firebase (Authentication).
- **Hardware:** Physical NFC tags.

## 8. Third-Party Libraries / APIs

- **Internal API:** Custom-built REST API to manage character/item metadata and game state.
- **NFC Tools App:** To read/write nfc cards

## 9. Team Organization and Responsibilities

- **Team Organization:** Chase Blancher will serve as Project Manager, Logan Harmon will help Chase in delegating tasks and overseeing the timeline. The team communicates via an iMessage group chat and meets in person every Wednesday for about 2 hours.

Member	FSU ID	Expertise	Project Responsibility
<b>Chase Blancher</b>	cgb22	C#, SQL, TS, React	<b>Lead/PM:</b> Core Platform, DB Schema, NFC flow, Ticketing.
<b>Logan Harmon</b>	lmh22c	TS, SQL, REST, C#, Rust	<b>Core Platform:</b> Account systems, API plumbing, NFC binding, CI/CD, Ticketing
<b>Greg ElDeiry</b>	gce22	SQL, REST, React	<b>Gameplay Loop:</b> UI/UX for combat, hand/deck mechanics.

<b>Zach Sandifer</b>	zts22a	Mobile, SQL, JS	<b>Gameplay Loop:</b> Backend "Run State" snapshots and combat logic.
<b>Gio Espinal</b>	ge22	React, SQL, Python	<b>Content Pipeline:</b> Admin tools, character/item seed scripts.
<b>Laura Saravia</b>	LS20FA	C#, SQL, Java	<b>Content Pipeline:</b> Character identity design and balance tooling.

- **Expanded Labor Division Details:** We decided to divide the project into three functional "squads" to ensure that the core technology, the game mechanics, and the actual content are developed in parallel.
- **Squad 1: Infrastructure & The "Toys-to-Life" Loop**
  - **Members:** Chase Blancher & Logan Harmon
  - **Focus:** The end-to-end flow of turning a physical card into a digital asset.
    - **Database & Security:** Designing the PostgreSQL schema to handle User Accounts, Card Ownership, and Character Stats.
    - **NFC Handshake:** Developing the mobile-optimized `/scan` route that parses URL parameters from physical tags (e.g., `vesania.com/scan?id=xyz123`) and binds them to the logged-in user.
    - **The Roster System:** Building the "My Collection" interface where players can view and manage their claimed cards.
    - **Deliverables:** Database migrations, user authentication, functional NFC-to-DB binding, and a card-claiming API.
- **Squad 2: Gameplay Loop (The Combat Engine)**
  - **Members:** Greg ElDeiry & Zach Sandifer
  - **Focus:** Creating a playable "Slay the Spire-style" combat slice.
    - **Battle Interface:** Building the frontend "Run" screen, including hand/deck UI animations, turn-order indicators, and enemy health bars.
    - **State Management:** Implementing backend "Run State" snapshots to ensure that if a user refreshes their browser, the game resumes exactly where they left off.
    - **Logic Engine:** Defining the turn-based rules—energy costs, card draw/discard piles, and damage calculation.
    - **Deliverables:** A playable combat demo where users can play placeholder cards against an AI opponent and a persistent "run" session in the database.
- **Squad 3: Content Pipeline & Balance Tooling**
  - **Members:** Gio Espinal & Laura Saravia

- **Focus:** Populating the world with unique characters and maintaining game balance.
  - **Admin Dashboard:** Creating a developer-only interface to quickly add or edit character names, health values, and card abilities without writing new code.
  - **Seeding & Content:** Writing scripts to populate the database with the initial 6 characters and 40+ unique items/spells.
  - **Game Design:** Defining the "identity" for each character (e.g., a "Tank" character vs. a "Mage") and balancing their numerical stats to prevent any one card from being overpowered.
  - **Deliverables:** A content browser, complete character/item datasets, and the logic that attaches a specific digital character identity to an NFC-claimed card.
- **Project Timeline Overview**
  - **Month 1:** Database setup, basic NFC scanning, and combat UI wireframes.
  - **Month 2:** Integrating scanned cards into the combat engine and building the content admin tools.
  - **Month 3:** PvP implementation, game balancing, and final UI polishing for the FSU Lab presentation.