

Project Proposal for GROUP 19 INDICATE PROJECT OPTION 4

Team Number: 19

Team Members: Aditya Tripathi SN65268641, Adrian Kusuma SN83150367, Brenner De Vos SN24226755, Henry Augustiano SN32741829, Leo Henon SN14983282

1 Overview

We want to make a free to play 1 on 1 multiplayer card game that runs in a web browser. We want players to be able to go to our site, immediately play as a guest, or create an account to track their stats and play against friends. The game itself is a custom made trading/collecting card game, brought to the digital space, in the same wheelhouse as Pokémon or Magic: The Gathering. Since it's custom made, it has its own mechanics and cards unique to it. Our MVP will be a bare bones version of this, where players will be able to go to our site and immediately play a quick match successfully, and they can create an account that tracks their stats. Having a friends list and being able to host games with friends will have to be an additional feature, since the scope is already quite large (We have to all learn Unity, figure out multiplayer servers, figure out user accounts, and we have to build/finish building a custom card game).

By making it run on a web browser, it will be way more accessible to the player than other card games (both online and offline versions), since there isn't any software required (apart from a browser they should have by default), and they don't have to pay for any cards at an actual store. On top of that, the whole game will be accessible for free, without having certain features locked behind a paywall (Magic: The Gathering Online is technically free to play, but has some features that only premium accounts have access to).

From a game design perspective it's mechanically unique in the sense that players are playing from a single pile of cards instead of individual piles. We'll start off with having three decks of 15-20 cards, and each deck will be mechanically unique. Each player chooses one, and the chosen decks get shuffled together and they will have to both play from the shuffled pile. There will also be unique interactions between chosen piles and the player that chooses the pile, like additional gameplay effects.

1.1 Envisioned Usage

User Group 1: Players: anyone that navigates to our game's website, and isn't an admin. Since we have Guest and User accounts, and our game has custom rules, everyone will be on the same playing field to start (there isn't a different version of the game for more experienced players). Everyone will have the option to watch the same tutorial, play a quick match, or create an account.

User Scenario for Group 1 (Players): players will navigate to our website that is hosting our game, where they will be allowed to play a quick match as a guest (which will not track their stats), log in to/or create a user account (in order to have their stats tracked), or watch the tutorial (which will briefly cover our game's core mechanics). If they log in, they will be able to join a quick match or watch the tutorial as well, but they will also be able to edit their account's information (email and password etc.), and view their account's statistics (number of wins, losses, games played etc.). Account holders will also be able to log out if they wish.

User Group 2: Admins: all the members of our team and anyone who will be in charge of running/testing our game's matchmaking and user account systems.

User Scenario for Group 2 (Admins): admins will have all the same access to our game as the players, but additionally, they can access our game server's information (who is connected and currently playing), as well as all user account information (passwords will be encrypted, but they can see email addresses or usernames). They can also delete user accounts.

2 Major Milestones

Deadline	Deliverable
Term 1 week 9: MiniPresentation	<p>-Exploration. We have to learn how to use Unity. So we aim to have a launchable unity “game” that runs in a web browser. Maybe it's just the main menu, maybe people can log in? It would also be good to have all the card game’s rules finalized (on paper, not in game)</p>
Term 1 week 13: Design	<p>-The “game” runs in a web browser, and starts at the main menu that has a consistent user interface implemented. From the main menu, users can see how they will join a quick match (maybe not play), login or create an account, view their stats (unless they are a guest), play/watch the tutorial, see their account info (if logged in), log out (if logged in), and quit the game.</p>
submission	
Term 2 week 4: Peer Testing	<p>-A simple version of the game (not full roster of cards yet) is implemented (1st feat.), and users will be able to join and play against each other (just quickmatch, 2nd feat.). The game will need a card database (3rd feat.), card interactions (4th), player interactions (health and resources (i.e. mana—5th feat.), and the players will need to actually play the cards against each other (6th feat.). This version of the game will also need to be play tested (7th feat.). The cards and battlefield will only display placeholder art (8th feat.). Users will be able to create and log in to an account (9th+10th feat.).</p>

Term 2 week 8: Peer Testing	<p>-Account holders can now edit their account's information (from the account info. page), where they can also delete their account (1st+2nd feat.). They will have encrypted passwords and a reset password option as well (3rd+4th feat.). The account will track player stats, and the users will be able to view them in the stats page (5th+6th feat.). More cards are implemented (7th and 8th feats—implement and test to balance). The version of a tutorial will be implemented (9th feat. create, and 10th to implement).</p>
Term 2 week 13: Final project submission	<p>-All playable cards are implemented, game battlefield and each card art is finished (1-3 feat.s – All cards, create and implement assets, plus music). There is a game over screen (4th feat.). The menu works as envisioned, and has a consistent U.I. design. Players can add an avatar to their accounts (5th feat.). Players can send one of four emotes to their opponent while playing a match (smiley face, sad face, thumbs up, or lol are examples of possible emotes) (6th and 7th feat.s—create and implement).</p>

Table 1: Proposed Project Milestones: Provide any explanation necessary to make your milestones understandable. These milestones need to make sense of the number of people in your team and the number of weeks between each milestone.

3 Technology Stack

-Users will interact with it using **Web browsers**: we are using this because of the ease of access they provide. They are free, usually already downloaded on whatever device you are on, and can be made to work on both mobile or pc. Being forced to download and set up another app can be frustrating, and posting the game to Steam or other game launcher platforms will require the user having set up and being familiar with their systems.

-We are going to use **Unity and Unity WebGL** to build our software. Unity is widely used across the gaming industry and across many platforms, so having experience in it would be invaluable to have on a resume. As a bonus since it is so popular, there is a lot of documentation, tutorials, and packages out there to help us utilize it to its full potential. WebGL just allows us to run our game in a web browser.

-We are going to need a **server** hosted by the school for communicating game data and user information.

-We are going to use **Netcode for Game Objects** to handle the multiplayer matches (communicating what each player is doing to each other). It was made for Unity, so there is lots of documentation available. Unfortunately, Netcode doesn't support WebGL (it doesn't allow access to IP Sockets), so we have to use **Photon Realtime** as a **transport** to get around this. Photon has a limit of 20 connections, but for our scope that should be fine (we could host around 9-10 games with 2 users in each, depending if we allow spectators or admins to join in-progress games or not).

-For our Testing Framework, we are going to use **UnityTest** or **UTF**. It uses a Unity integration of NUnit, and will allow us to test our code in both Edit and Play Mode, which could be really useful.

-We will use SQL databases to **host user information** (things like usernames/emails, passwords, stats).

-**Programming Languages**: Web browsers use HTML, CSS, and PHP, while Unity, Netcode, and Photon use

C#, and MySQL Workbench uses SQL.

4 Teamwork Distribution and Anticipated Hurdles

Use the table below to help line up everyone's strengths and areas of improvement together.

Category	Aditya Tripathi	Leo Henon	Brenner De Vos	Henry Augustiano	Adrian Kusuma
Experience	<p>Designed a prototype model of a ride-sharing app in COSC 341</p> <p>Designed an inventory tracking system for a hypothetical ski and snowboarding rental shop in COSC</p> <p>Designed a few websites on my own</p> <p>Have experience with java, SQL and Python</p>	<p>Financial management software project with next.js and react.js</p> <p>AI article generator REST api</p> <p>2+ years professional experience in PHP web development with Symfony</p> <p>Adtracker and SEO bots, python.</p> <p>smart contract programming in Solidity.</p>	<p>Designed a few websites (COSC 304, 360, and a couple outside of school). Mostly did the front end.</p> <p>Designed a very simple point and click python game (only two levels). I created all of its art on my own in photoshop or using pygame's system.</p> <p>Developed several projects as a team using GitHub and its project management systems.</p> <p>Database experience with MySQL most recently, but have used SquirrelL and Docker in the past.</p>	<p>Utilized NextJs, NodeJs, ReactJs as the tech stack,work around with REST API during co-op employment.</p> <p>COSC304 Inventory tracking website using mySQL.</p> <p>Used PHP, XAMPP for discussion forum website</p> <p>Database: -MySQL -MongoDB -Postgres</p>	<p>Designed a website for COSC360 using an XAMPP stack.</p> <p>Game design but for physical games, not digital ones.</p> <p>Worked with a lot of project management programs (Smartsheet, Asana, etc.)</p> <p>Database experience in MySQL, MongoDB, Postgres.</p>
Good At	<p>I am good at front end programming</p> <p>Communication</p> <p>Reviewing</p>	<p>backend programming</p> <p>communication</p>	<p>Art Design, good with photoshop, and premiere pro in case we end up making video presentations</p> <p>Coding,</p>	<p>-Backend -Finding solution when bugs occurs</p>	<p>I'm good at research and planning.</p> <p>Coding wise, I believe my strength lies in my database skills.</p>

	Working as a team and managing the project collectively with my team members		frontend, user interface design, GitHub project management Really good at not giving up on a problem		Backend is my preferred role and project management too.
Expect to Learn	Game development in unity from scratch Unity as a software itself Implementati on of the game in a browser	game development Unity Improve at project management	Learning Unity from scratch will be a journey for sure. Backend things like having the players communicate through a server will also be interesting. Automated Testing.	-testing -game development -CI/CD	Expecting to learn a lot about game development and using unity. Also expecting to learn about implementing unity in browsers.

Table 2: Team Experience, Expertise, and Areas of Learning: Give the reader some context and explanation about your table. It can be short and described in the caption, but it needs to help the reader how to interpret what's in the table.

Distribution

Category of Work/Features	Aditya Tripathi	Leo Henon	Brenner De Vos	Henry Augustiano	Adrian Kusuma
Project Management: Trello Maintenance	✓	✓	✓	✓	✓
Technical Direction: Time Estimation, Making Programming Choices		✓			✓
Technical Help: Finding Technical Solutions	✓	✓		✓	✓
Troubleshooting: The Go-To When Others Are Stuck		✓		✓	
System Architecture Design		✓	✓		✓
User Interface Design	✓		✓		✓

CSS Development	✓		✓		
Main Menu	✓	✓	✓	✓	✓
Login/Create Account/Logout	✓			✓	✓
Password Reset & Encryption		✓		✓	
Account Statistic Tracking & Display (wins, losses etc.)				✓	
Edit Account Information & Delete Account				✓	✓
Placeholder Art (cards and battlefield etc.)	✓		✓		✓
Custom Art (all cards and battlefield, game music etc.), A.I. can be used to lighten the workload if needed.			✓		✓
Decks of Cards (Need at least 3 unique ones by the end, 15-20 cards each)					✓
Quick Match (Can join and play a match that enforces the rules and knows who wins)	✓	✓	✓	✓	✓
Game Mechanics (health system, resource system, etc.)		✓			✓
Creating card database			✓		✓
Developing card interactions		✓	✓		✓
Admin. Account & Features				✓	✓
Game Mechanic Tutorial			✓		✓
Quick Match Emotes	✓		✓		
Database Setup	✓			✓	✓
Photon set up	✓				✓
Photon and Multiplayer testing	✓	✓			
Unity Testing Framework		✓		✓	
Presentation Preparation	✓	✓	✓	✓	✓
Design Video Creation	✓		✓		
Design Video Editing	✓	✓	✓	✓	
Design Report	✓	✓	✓	✓	✓

Final Video Creation			✓	✓	
Final Video Editing		✓	✓		
Final Team Report	✓	✓	✓	✓	✓
Final Individual Report	✓	✓	✓	✓	✓

Table 3: Expected Areas of Contributions: Explain how things are assigned in the caption like this, or put the explanation into a separate paragraph so the reader understands why things are done this way and how to interpret your table.

Everyone is assigned to the project management and any of the presentations/reports as well, since each of us will constantly be part of what is getting done and assigned to each person. For the back end features like system architecture, programming choices, and database setup we assigned Leo and Adrian. For the front end stuff like the U.I., main menu, and CSS, Brenner and Aditya are assigned because they enjoy designing and building front end features. For the actual aspects of the card game (Decks of Cards, Tutorial, game mechanics), we assigned Adrian, since they created the game outside of this class, and Brenner, since he is really passionate about the game. The User Account features like logging in, creating an account, and editing the account are assigned to Adrian, Aditya and Henry because they have experience making user accounts in previous classes. Admin Account features are assigned to Henry and Adrian because they have prior experience in developing admin features particularly admin pages. The Statistic Tracker and the Stats page is assigned to Henry. Any video creation or editing is assigned to Brenner and Leo, because they are familiar with and enjoy doing it. C# testing is assigned to Henry and Leo because they had prior experience in unit testing. Network and Photon testing is going to be tested by Aditya and Leo.

Everyone will be a part of the main menu because all of the main features will interact with it, and everyone will be a part of the Quick Match feature. We need to implement the game's logic and rules, which will be a big task since it only exists on paper currently and all of the cards are not quite created. Additionally, we will need back end people to set up how players will join and interact with each other, and front end people to design how it looks. We will also need everyone to play test it as we go. The Quick Match Feature is the main feature of our MVP after all.

5 Bonus Stretch Goals: (No-one assigned yet, since they are time permitting)

	Aditya Tripathi	Leo Henon	Brenner De Vos	Henry Augustiano	Adrian Kusuma
Friends List					
Join/host friend games					
Spectator Mode					
4 decks					
Custom deck picker.					

These are features we want to add if we have time, but are outside of the scope of our MVP. A friend list for account holders would be great since it would allow people to play with their friends and show off their stats, but the logistics of having the user's host a game themselves seems like it could be alot (do we want a lobby system? Or maybe just a room code?). Watching friends in a spectator mode could also be cool. Additional cards will also make the game more interesting and provide more variety to the game, but every card introduces a lot of

new interactions that need to be tested as well as more assets to be made, so we are aiming for only 3 decks of about 15-20 cards for our MVP. Additionally, account holders should be able to custom pick a deck of cards they want to use from the existing cards that they will use in game. This will need its own page with a whole new interface and new assets, and since the deck gets mixed in with the opponents anyways, we decided this is better as a bonus feature than a core one.