

# Slippi Stats Project Pitch

University of Wyoming  
Senior Design 2023-2024

Beckham Carver [bcarver4@uwyo.edu](mailto:bcarver4@uwyo.edu), Kyle Lofthus [klofthus@uwyo.edu](mailto:klofthus@uwyo.edu),  
Ben Wilkin [bwilkin@uwyo.edu](mailto:bwilkin@uwyo.edu), Zachary Crimmel [zcrimmel@uwyo.edu](mailto:zcrimmel@uwyo.edu),  
Michael Stoll [mstoll3@uwyo.edu](mailto:mstoll3@uwyo.edu)

*A ragtag team of has-beens, wanna-be's, and do-ers of. Our team brand is one of conventional unconventionalism.*

Pitch [Slideshow](#): (We made this in case we needed to give a more formal 5-minute pitch)

## Project Goals:

- Build a standalone desktop app for computing and displaying statistics from .slp files
  - Desktop App (or integrate all features into fork of the Slippi-Launcher)
    - Electron & React using JS/TS
  - Create and program **cumulative statistics**
    - Compiles many single-game stats into useful cumulative statistics
    - Use a sqlite database to store game stats
  - Create and program **advanced single game statistics**
  - Merge statistics features into Slippi-Launcher

## Resources:

- Slippi-JS (<https://github.com/project-slippi/slippi-js>)
  - Javascript codebase for parsing and computing basic statistics from single game
  - Current items parsed from .slp files:
    - Book-keeping info, characters & stage selected, player names etc.
    - Move count
    - Fighting state vs. neutral state
    - Combo % (semi-advanced)
    - Wavedashes, wavelands (semi-advanced stat)
- Slippi Launcher (<https://github.com/project-slippi/slippi-launcher>)
  - Electron & React Frontend for launching the game
    - Connects players for multiplayer,
    - Communicates with the Dolphin emulator
    - Displays basic single game stats
    - Replays games from .slp files

## Jobs:

- Frontend - Designing layout of application and logical flow
  - Create graphs for cumulative stats
  - Create other visual outputs to display
  - Separate tab/window in app for what Slippi can already do (replays, basic stats, etc.)
- Backend - Working with .slp files directly to compute more advanced statistics
  - Aggression % - What percent of the match were you running towards (aggressing) your opponent
  - How to measure a ledge guard
    - Hit to the ledge vs. Opponent willingly grabbing the ledge
    - Successful ledge guard vs SD's

- Easier match stats
    - Win % vs each matchup
    - Winning moves
- Local database work - Store calculations of .slp files locally in order to improve efficiency

## Rough Roadmap:

- End of Fall Semester - Fork Slippi-Launcher and add a new stats tab.
  - Optional: store single game stats in a basic fashion (non-database), and display a summary on stats page
- Minimal Viable Product:
  - Add a new stats page to the launcher,
  - Generate cumulative statistics from all existing files, and visualize
  - Store results in SQLite database, & update incrementally as games are played
- Realistic additions:
  - Create a new single game advanced statistic (ie. whiff punish vs. counter-hit, aggression %, etc.)
  - Create a game state system used to contextualize sections of a game: ex: opening -> counterhit -> edgeguard -> kill -> opening -> ...
- Stretch goals:
  - Match feedback system: 'you should've done x at moment y', 'use x move more often' etc.
  - Playstyle profile graph, using a spider chart
    - Requires advanced single game stats that inform cumulative playstyle
  - Use new stats as additional data for existing ML models that create AI players