Tasklist — Adaptabrawl (Assignment #5)

Course: Senior Design (Fall 2025)
Repository: Adaptabrawl — Senior Design

Team

- Kartavya Singh — Netcode & Infrastructure
- Saarthak Sinha — Combat & Systems
- Kanav Shetty — UX/UI & Content
- Yash Ballabh — Tools, CI/CD & QA

Tasks (20 total; ~5 per teammate)

1. **Identify** an online model (host/client) and **map** session flow (host, join, ready, reconnect) consistent with our design diagrams — **Kartavya Singh**.
2. **Design** state updates (20–30 Hz) and **set** a simple client-prediction + server-correction flow with a fixed RNG seed — **Kartavya Singh**.
3. **Implement** Lobby/Relay integration with secure room codes and resilient error handling — **Kartavya Singh**.
4. **Develop** movement prediction so controls feel smooth and **correct** drift from the server within a small resim budget — **Kartavya Singh**.
5. **Stress-test** online play (≈120 ms RTT, ~2% loss, ~20 ms jitter) and **note** the limits we can accept — **Kartavya Singh**.

6. **Outline** the combat state flow (startup → active → recovery), **define** cancel windows and input buffer rules — **Saarthak Sinha**.
7. **Implement** hit/hurtbox resolution and a damage system (hitstop, knockback vectors, armor break) — **Saarthak Sinha**.
8. **Create** two starter fighters with simple, balanced move sets (via ScriptableObjects) — **Saarthak Sinha**.
9. **Develop** the status/condition system (e.g., Poison, Heavy-Attack, Low-HP) with stacking and disclosure events — **Saarthak Sinha**.
10. **Test** and tune frame data for fairness/readability across both fighters using structured playtests — **Saarthak Sinha**.

11. **Design** the HUD (HP bars, status icons with timers) and **wire** it to gameplay events — **Kanav Shetty**.
12. **Build** lobby and post-match UI flows (create/join, ready, rematch/exit) with clear error and success states — **Kanav Shetty**.
13. **Implement** Input System maps for keyboard/controller with in-game rebind UI and persistence — **Kanav Shetty**.
14. **Produce** readable VFX/SFX palettes for hits, counters, and condition changes with performance budgets — **Kanav Shetty**.
15. **Conduct** usability tests for readability/accessibility (font scale, color-safe modes) and **iterate** based on findings — **Kanav Shetty**.

16. **Add** GitHub Actions to **build** a Win64 player and **run** smoke tests on every PR — **Yash Ballabh**.
17. **Define** what we log (match length, move use, win rate) and **ship** an opt-in exporter for playtests — **Yash Ballabh**.
18. **Make** a simple training mode with a hitbox viewer and dummy record/playback for QA — **Yash Ballabh**.
19. **Profile** CPU/GPU/GC to **hit** 60 FPS on mid-range PCs and **file** clear performance budgets — **Yash Ballabh**.
20. **Document** architecture/controls/troubleshooting and **package** the submissions — **Yash Ballabh**.