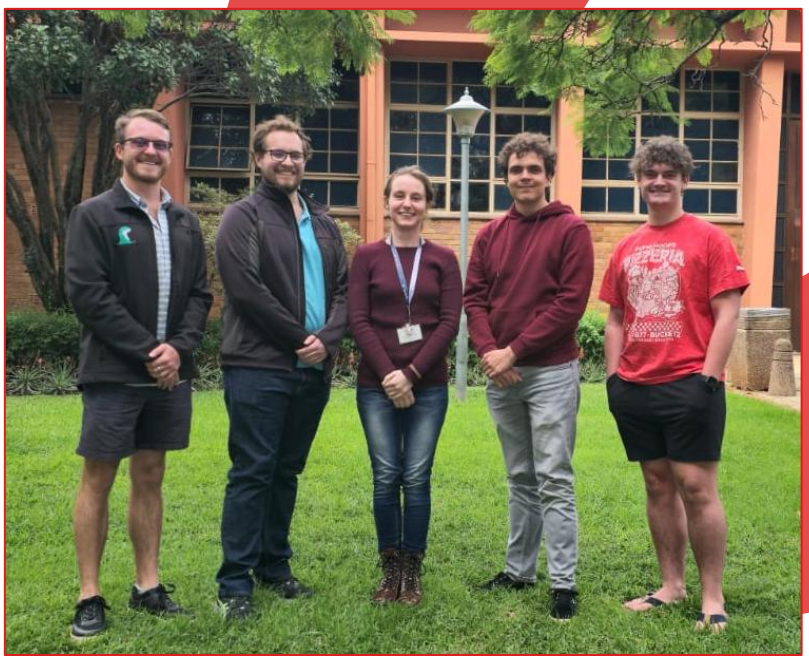
SuperLap Racing Line Optimization System

EPI-USE



Quintessential

Amber Ann Werner [u21457752]

Milan Kruger [u04948123]

Qwinton Knocklein [u21669849]

Sean van der Merwe [u22583387]

Simon van der Merwe [u04576617]

Testing Policy

Testing Scope & Levels

Level	*Focus*	*Tools/Methods*
Unit Testing	Individual functions (e.g: track image processing, RL reward function).	Pytest (Python), JUnit (Java).
Integration Testing	Interaction between services (e.g: track processor → RL engine).	Postman, Jest (API tests), Selenium (UI flows).
System Testing	End-to-end workflows (e.g: upload image → simulate → visualize).	Cypress, Robot Framework.
Performance Testing	Scalability (e.g: 50 concurrent users), RL training speed.	Locust (load testing), NVIDIA Nsight (GPU profiling).
Security Testing	Data encryption, auth vulnerabilities.	OWASP ZAP, SonarQube.
User Acceptance (UAT)	Real-world usability (by target users).	Beta releases, A/B testing.

Testing Types & Frequency

Test Type	*Description*	*Frequency*
Automated Regression	Validate existing features after updates.	On every Git commit (CI/CD).
Manual Exploratory	Unscripted UX/edge-case testing.	Before major releases.
Physics Validation	Compare AI racing lines against known heuristics (e.g: apex accuracy).	Per RL model update.
Hardware Compatibility	GPU/CPU performance benchmarks.	Quarterly.

Entry & Exit Criteria

Entry Criteria (Tests Start When):

- Requirements are documented (e.g: FR/NFRs).
- Code is merged to the test branch.
- Test environment mirrors production (GPU-enabled).

Exit Criteria (Tests Pass When):

- **Unit/Integration:** ≥90% code coverage (measured via Coveralls).
- **Performance:** <2s response time for track processing; RL training FPS ≥30.
- **Security:** Zero critical OWASP vulnerabilities.
- **UAT:** ≥80% positive feedback from beta testers.

Defect Management

- **Severity Levels:**
    - **Critical** (Crash/data loss): Fixed within 24h.
    - **Major** (Feature failure): Fixed in next sprint.
    - **Minor** (UI glitch): Backlogged for prioritization.
- **Tracking:** Jira/Linear with labels (bug, reproducible, blocker).

Environments

Environment	*Purpose*	*Access*
Development	Feature development.	Engineers only.
Staging	Pre-production (mirrors prod).	QA/Product Team.
Production	Live user-facing system.	Automated deployments only.

Test Data Management

- **Realistic Datasets:**
    - 10+ sample tracks (F1, MotoGP circuits).

    - Synthetic data from racing sims (Assetto Corsa).
- **Anonymization:** User-uploaded tracks scrubbed of metadata.

Compliance & Reporting

- **Audits:** Monthly test coverage/review meetings.
- **Reports:** Dashboards for:
    - Test pass/fail rates.
    - Performance trends (e.g: lap time prediction accuracy).

Policy Exceptions

- **Emergency Fixes:** Hotfixes may bypass some tests but require:
    - Post-deployment regression testing.
    - Retrospective review.