PROYECTOS DE INGENIERÍA Y GESTIÓN DEL SOFTWARE

REPORT NUMBER	**6**
TEAM NUMBER	**24**
TITLE	**Mitigation, monitoring, and containment**
DATE	**13/04/2025**

Team Member	Risk		Measure
Fran	**Integration issues between the WebSocket-based real-time tracker and the backend API**	Mitigation	Clearly define detailed API contracts at the beginning of development, conduct regular integration testing to ensure both systems communicate effectively.
		Monitoring	Regularly track connection stability, response time, and error rates from the real-time system.
		Containment	Develop an alternative solution using HTTP polling to ensure continuous updates in case of prolonged WebSocket failures.
Fran	**The system may not be able to handle properly a lot of vehicles, drivers, or real time locations due to the database restrictions.**	Mitigation	Optimize the database schema by ensuring efficient indexing, caching commonly accessed data, and regularly reviewing database performance.
		Monitoring	Constantly monitor database load, query performance, response times, and system resources.
		Containment	Prepare to scale database capabilities or migrate to a more robust database infrastructure if performance bottlenecks arise.
Fran	**More rural or isolated zones of the places we are working in may have poor connection, which would lead to wrong real-time data.**	Mitigation	Design mobile and client applications with offline capabilities, enabling data to be stored locally and synchronized when connectivity returns.
		Monitoring	Continuously analyze connectivity issues reported by users, reviewing error logs from these areas.
		Containment	Provide backup methods such as periodic updates or SMS-based location reporting to ensure critical data reaches the system.

Instructions:

Each member must select a risk and determine one measure to mitigate it, one to monitor it, and one the contain it.

Antonio	**Client requirements changing during development (especially related to route visualization and ETA calculations)**	Mitigation	Frequently involve the client through agile sprints and demonstrations, gathering continuous feedback to minimize surprise changes.
		Monitoring	Document client feedback and requested changes clearly and consistently.
		Containment	Structure the application modularly, allowing quick adjustments without needing major overhauls of the system.
Antonio	**Team overload due to simultaneous backend and frontend tasks**	Mitigation	Clearly prioritize tasks based on importance and complexity, use sprint planning and agile methodologies to manage workloads effectively.
		Monitoring	Track individual and team productivity, workload distribution, and overall project progress.
		Containment	Consider temporarily adding external resources or adjusting timelines and deadlines if the team is consistently overloaded.
Antonio	**User interface is not intuitive enough for fleet managers with low tech literacy**	Mitigation	Conduct user testing sessions regularly, involving actual fleet managers to gather practical feedback and improve UI/UX iteratively.
		Monitoring	Review usability metrics and regularly collect feedback from real-world users.
		Containment	Provide clear, accessible tutorials, user manuals, and training sessions to help users understand and effectively use the system.
Shen	**Underestimated development time for implementing traffic-aware route tracking with MapBox/Google Maps API**	Mitigation	Conduct in-depth research into API capabilities and complexity, allocating conservative timelines to anticipate potential delays.
		Monitoring	Set regular project milestones and checkpoints to quickly identify and address delays.
		Containment	Maintain a reserve of buffer time and have simpler fallback routes and tracking methods ready in case of prolonged issues.
Shen	**Changes in data privacy regulations affecting user geolocation tracking**	Mitigation	Stay informed about evolving privacy laws, consulting experts early to ensure compliance from the start of the project.
		Monitoring	Regularly monitor legal developments and privacy guidelines related to data tracking.

Instructions:

Each member must select a risk and determine one measure to mitigate it, one to monitor it, and one the contain it.

		Containment	Have alternative privacy-focused tracking methods ready, which can be quickly implemented to comply with new regulations.
Shen	**Lack of clear API contract between frontend and backend teams, causing delays in parallel development**	Mitigation	Develop comprehensive and clear API documentation at the outset and update it continuously; hold regular alignment meetings between frontend and backend teams.
		Monitoring	Periodically review API documentation and implementation status.
		Containment	Quickly address any discrepancies through clear communication channels, interim documentation, and mock API setups if necessary.
Francisco	**Limited QA/testing resources causing delayed detection of bugs**	Mitigation	Leverage automated testing frameworks like Jest and React Testing Library and integrate developers into the QA process.
		Monitoring	Continuously track bug discovery rates, resolution times, and test coverage
		Containment	Conduct regular team-wide testing sessions ("bug bashes") and utilize community or beta testing groups to catch issues early.

Instructions:

Each member must select a risk and determine one measure to mitigate it, one to monitor it, and one the contain it.