WANDERERS
CONTINGENCY PLAN

Version *<1.0>*

<12/03/2025>

VERSION HISTORY

Version #	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	*Isaac Chun Jun Heng*	*12/3/2025*	*Isaac Chun Jun Heng*	*12/3/2025*	

UP Template Version: 12/03/25

Detailed Risk Mitigation and Contingency Plan for Wanderlog Project

Risk Analysis Overview

- Insufficient Testing (Red - Highest priority, Rank #1)
- Real-time synchronization failures (Red - Rank #2)
- Data breaches or leaks (Red - Rank #3)
- Scope Creep (Red - Rank #4)
- Integration issues with external APIs (Google Places) (Yellow - Rank #5)
- Security Misconfiguration (Red - Rank #8)
- Unsatisfactory Performance (Yellow - Rank #10)

1. Insufficient Testing

Element	Description
Risk	Insufficient Testing
Probability/Impact	70% (High) / High
Mitigation Plan	- Implement mandatory code reviews with test coverage requirements - Set up automated testing pipeline in CI/CD - Establish a minimum test coverage threshold (80%) - Create comprehensive test documentation - Conduct regular QA training sessions
Contingency Plan	- Deploy emergency hotfix team for critical bugs - Implement feature toggles for quick disabling of problematic features - Establish rollback procedures for all deployments - Maintain a separate staging environment that mirrors production - Have on-call support team during major releases

Responsible Parties	- QA Lead - Development Team - DevOps Engineer
Monitoring Indicators	- Test coverage percentage - Number of bugs reported post-release - Failed test metrics in CI pipeline
Timeline	Immediate implementation; weekly review

2. Real-time Synchronization Failures

Element	Description
Risk	Real-time Synchronization Failures
Probability/Impact	50% (Medium) / High
Mitigation Plan	- Enhance backend retry mechanisms - Implement event queuing system - Develop conflict resolution strategies - Optimize database locking procedures - Add network latency monitoring
Contingency Plan	- Implement offline mode capabilities - Create manual sync trigger option for users - Design graceful degradation to non-real-time mode - Set up automated recovery procedures - Establish emergency communication protocol for widespread failures
Responsible Parties	- Backend Engineers - DevOps Team - Database Administrator
Monitoring Indicators	- Sync failure rate - Average sync latency - Number of conflict resolutions required
Timeline	2 weeks for implementation; continuous monitoring

3. Data Breaches or Leaks

Element	Description
Risk	Data Breaches or Leaks
Probability/Impact	50% (Medium) / High
Mitigation Plan	- Conduct regular penetration testing - Review and enhance Supabase security configuration - Implement comprehensive data encryption - Establish secure access controls - Regular security training for team
Contingency Plan	- Execute incident response plan - Notify affected users according to regulatory requirements - Engage security forensics team - Isolate affected systems - Implement emergency patches - Prepare public communications strategy
Responsible Parties	- Security Officer - Legal Team - Communications Team - Development Lead
Monitoring Indicators	- Unusual access patterns - Failed authentication attempts - Security scan results
Timeline	Monthly security audits; immediate response to incidents

4. Scope Creep

Element	Description
Risk	Scope Creep
Probability/Impact	45% (Medium) / High

Mitigation Plan	- Conduct weekly scope review meetings - Implement strict change request process - Create detailed documentation of requirements - Enforce prioritization methodology - Set clear project boundaries with stakeholders
Contingency Plan	- Freeze non-essential feature development - Re-evaluate project timeline and resources - Negotiate contract adjustments if necessary - Implement phased delivery approach - Temporarily reassign resources to priority items
Responsible Parties	- Project Manager - Product Owner - Stakeholder Representatives
Monitoring Indicators	- Number of change requests - Deviation from original scope - Project timeline extensions
Timeline	Weekly scope reviews; quarterly major assessments

5. Integration Issues with External APIs (Google Places)

Element	Description
Risk	Integration Issues with External APIs (Google Places)
Probability/Impact	20% (Low) / High
Mitigation Plan	- Add secondary API fallback options - Increase Places API quotas - Implement request caching - Develop redundancy mechanisms - Create comprehensive API documentation
Contingency Plan	- Switch to alternative map provider - Implement cached data fallback - Deploy simplified offline location functionality - Reduce API call frequency temporarily - Notify users of limited functionality

Responsible Parties	- Integration Lead - Backend Engineers
Monitoring Indicators	- API failure rate - Quota usage - Response time metrics
Timeline	Review API usage weekly; test fallbacks monthly

6. Security Misconfiguration

Element	Description
Risk	Security Misconfiguration
Probability/Impact	30% (Medium) / High
Mitigation Plan	- Implement stricter Supabase rule checks - Schedule weekly security reviews - Enable notifications for configuration changes - Create security configuration templates - Adopt infrastructure as code for consistent setup
Contingency Plan	- Apply emergency security patches - Temporarily restrict access to affected components - Roll back to last known secure configuration - Conduct immediate security assessment - Monitor for exploitation attempts
Responsible Parties	- Security Engineer - DevOps Team - Database Administrator
Monitoring Indicators	- Failed security checks - Configuration drift metrics - Unauthorized access attempts
Timeline	Weekly security configuration reviews

7. Unsatisfactory Performance

Element	Description
Risk	Unsatisfactory Performance
Probability/Impact	30% (Medium) / Medium
Mitigation Plan	- Conduct regular caching optimization - Implement performance benchmarking - Optimize database queries - Set up performance monitoring tools - Create performance budgets for key features
Contingency Plan	- Scale infrastructure resources - Temporarily disable resource-intensive features - Implement emergency caching strategies - Optimize critical user paths - Deploy CDN for static content
Responsible Parties	- Performance Engineer - Full-stack Developers - DevOps Team
Monitoring Indicators	- Page load times - Database query performance - User-reported performance issues - Server response times
Timeline	Monthly performance reviews; immediate action on critical issues

Implementation Recommendations

1. **Prioritize actions based on risk ranking**: Focus immediate resources on the top 3 risks (Insufficient Testing, Real-time synchronization failures, and Data breaches)
2. **Establish regular risk reviews**: Schedule bi-weekly risk assessment meetings to evaluate the effectiveness of mitigation strategies and update contingency plans as needed
3. **Create dedicated response teams**: For each high-impact risk, assign specific team members responsible for executing the contingency plan
4. **Document all incidents**: Maintain a detailed log of all risk events that occur, including response effectiveness and lessons learned
5. **Conduct simulations**: Regularly test contingency plans through controlled simulations to ensure team readiness

This comprehensive plan addresses both preventive measures to reduce the likelihood of risks occurring and responsive strategies to minimize impact when they do occur. The plan should be treated as a living document and updated as the project evolves.