

Sprint Retrospective Meeting

Project: Cybersecurity: Lock and Key: Encrypted Password Management and Collaboration System

Attendees: Mauricio Ballart Alvarez (Product Owner), Alexis Henkel, Christian Dominguez (Team Lead), Isaac Quintero, Alberto Espinoza

Date: 09/13/25

Start time: 4:00 PM

End time: 5:00 PM

Sprint Overview

Sprint Duration: 2 weeks

Sprint Goal: Expand backend microservices, organize the repository and database structure, and begin frontend integration for authentication and login

What Went Well

Team Collaboration

- Strong peer support: Alberto actively assisted Isaac in troubleshooting encryption service issues and setup
- Consistent engagement: all members contributed ~2 hours and maintained regular progress updates
- Hands-on support: team members reviewed each other's work to ensure consistency across services

Technical Progress

- Database planning underway: Alexis began structuring the database diagram and identifying key relationships

- Authentication groundwork: Christian created and began developing the GitHub repository
- Major backend progress: Isaac completed a full Encryption Service with 24+ endpoints, Docker fixes, Redis caching, and quantum algorithms
- Frontend/backend alignment: Mauricio verified backend service structure against data models and prepared React login/authentication integration

Process Adherence

- Daily Scrum meetings highlighted blockers quickly
 - Progress updates were honest and tracked realistically
-

What Didn't Go Well

Technical Decision-Making

- Some uncertainty remains in choosing authentication platforms for implementation
- Database organization still feels unclear, creating minor planning delays
- Integration steps between backend microservices and planned frontend need clearer mapping

Project Organization

- General structure for repository and documentation still developing
- Lack of formal prioritization sometimes slowed progress

Resource Allocation

- Heavy backend progress contrasted with slower movement in project organization and frontend planning

What We Learned

Technical Insights

- Encryption service development requires careful integration testing with other services
- Repository organization impacts collaboration and efficiency
- Authentication and database planning should be aligned earlier with backend progress

Team Dynamics

- Peer-to-peer troubleshooting accelerates problem solving
- Collaborative repository setup prevents isolated work
- Reviewing each other's work improves quality and consistency

Process Improvements

- Need clearer task prioritization to balance backend and frontend progress
- More structured repository guidelines will reduce confusion
- Mid-sprint alignment meetings help keep frontend/backend efforts connected

Action Items for Next Sprint

Immediate Actions (Sprint 2)

- Finalize authentication framework decision (Owner: Christian)
- Complete database structure and confirm with backend needs (Owner: Alexis)

- Begin React frontend login/authentication integration (Owner: Mauricio)

Process Improvements

- Define repository standards for documentation, services, and frontend (Owner: Alberto)
- Implement integration checkpoints to confirm services communicate properly (Owner: All)

Sprint 2 Focus Areas

- Prioritize implementation tasks alongside organization
- Document technical setups for encryption, authentication, and database for future reference

Team Feedback & Commitments

Alexis Henkel

Commitment: Will focus on completing database diagram and ensuring repository integration

Request: Needs more clarity on service-to-database connections

Mauricio Ballart Alvarez

Commitment: Will move forward with React frontend for login/authentication and verify integration with backend services

Observation: Backend progress is strong; need to bring frontend up to speed

Christian Dominguez

Commitment: Will finalize authentication framework and continue expanding the GitHub repository

Action: Will facilitate alignment checkpoint for backend/frontend

Isaac Quintero

Commitment: Will expand backend by adding Notification and Audit services

Appreciation: Thanks Alberto for technical support during encryption setup

Alberto Espinoza

Commitment: Will lead repository organization and documentation standards

Suggestion: Pair programming could help accelerate database and frontend integration work

Sprint Metrics

- **Velocity:** 18 story points completed (target was 20)
 - **Burndown:** Strong backend progress, with slight delays in organization and frontend setup
 - **Team satisfaction:** 7/10 (happy with backend progress, some concern about organization and integration pace)
-

Next Sprint Planning Notes

- Shift balance from backend-heavy tasks to frontend and integration
 - Ensure repository is fully standardized before adding new services
 - Break technical decision stories into smaller, time-boxed tasks with clear acceptance criteria
-

Meeting adjourned at: 5:00 PM

Sprint Retrospective Meeting

Project: Cybersecurity: Lock and Key: Encrypted Password Management and Collaboration System

Attendees: Mauricio Ballart Alvarez (Product Owner), Alexis Henkel, Christian Dominguez (Team Lead), Isaac Quintero, Alberto Espinoza

Date: 09/28/25

Start time: 7:00 PM

End time: 8:00 PM

Sprint Overview

Sprint Duration: 4 weeks (09/02/25 - 09/26/25)

Sprint Goal: Establish a complete microservices architecture, integrate frontend-backend systems, and achieve a production-ready MVP with advanced security features

What Went Well

Technical Breakthroughs

- **Microservices Architecture Excellence:** Isaac's implementation of 7 complete microservices with Kong API Gateway represents exceptional technical achievement
 - Authentication Service (8001) with MFA and device trust
 - User Service (8002) progressing toward completion
 - Vault Service (8003) with 15 API endpoints and zero-knowledge encryption
 - Encryption Service (8005) with 24+ quantum-resistant algorithms
 - Notification Service (8006) with Celery background processing
 - Audit Service (8007) implementation in progress
 - Kong API Gateway (8888) unifying all services
- **Advanced Security Implementation:** Team achieved industry-leading security standards
 - Quantum-resistant cryptography (Dilithium, Kyber, SPHINCS+)

- Zero-knowledge encryption with user-specific key derivation
 - Comprehensive RLS policies and MFA implementation
 - Docker containerization with proper service isolation
- **Frontend-Backend Integration Success:** Mauricio and Alexis achieved seamless React integration
 - Complete frontend authentication flow
 - Real-time communication with all backend services
 - Responsive design with master password forms and Q-Shield indicators
 - TDD implementation with 24/24 tests passing

Infrastructure and DevOps Excellence

- **Production-Ready Infrastructure:** Alberto's comprehensive CI/CD and deployment pipeline
 - Docker containerization across all services
 - Automated testing and deployment workflows
 - Performance optimization, achieving sub-150ms response times
 - Comprehensive monitoring and logging systems
- **Database Architecture Maturity:** Alexis advanced the database structure to production readiness
 - Comprehensive schema with 4 main tables supporting all features
 - Integration with microservices architecture
 - Performance optimization for concurrent users

Process and Team Dynamics

- **Exceptional Collaboration:** Cross-functional teamwork resulted in breakthrough solutions
 - Isaac's infrastructure work supporting all team workstreams
 - Christian's GitHub repository setup enables smooth collaborative development
 - Strong peer assistance culture (Alberto helping Isaac, etc.)
- **Consistent Delivery:** Team maintained 2-hour daily commitments while achieving major technical milestones
 - Regular sprint planning and backlog grooming meetings
 - Effective daily scrum communication
 - Proper risk identification and mitigation

What Didn't Go Well

Integration Complexity Management

- **Service Integration Coordination:** Multiple microservices created coordination overhead
 - Need for better integration testing early in the development cycle
 - API versioning and service dependency management could be improved
 - Cross-service error handling and rollback procedures need refinement

Documentation and Knowledge Transfer

- **Technical Documentation Lag:** Rapid development outpaced documentation efforts
 - Need for better API documentation maintenance
 - Architecture decision logs could be more comprehensive
 - Knowledge sharing across a complex microservices architecture

Timeline and Scope Management

- **Feature Scope Expansion:** Advanced features (quantum cryptography, etc.) expanded beyond the initial scope
 - Need for better scope management and feature prioritization
 - Timeline estimation for complex security features proved challenging
-

What We Learned

Technical Insights

- **Microservices Architecture Value:** Properly implemented microservices provide exceptional scalability and maintainability
- **Security-First Development:** Early integration of security consultation prevents technical debt
- **API Gateway Benefits:** Centralized routing dramatically simplifies service management
- **Docker Containerization ROI:** Containerization effort pays significant dividends in deployment and scaling

Team Development

- **Skill Specialization Benefits:** Team members develop deep expertise in their domains while maintaining collaboration
- **Cross-Functional Knowledge:** Understanding each other's work improves integration quality
- **Technical Leadership:** Isaac's infrastructure leadership enabled rapid team velocity

Process Improvements

- **Sprint Planning Effectiveness:** Structured sprint planning with user stories improved delivery predictability
 - **Daily Communication Value:** Regular check-ins prevented major integration issues
 - **Risk Management Success:** Early identification of technical risks prevented major setbacks
-

Action Items for Next Sprint

Integration and Testing Enhancement

Comprehensive Integration Testing Framework *(Owner: All)*

- Implement automated integration testing across all microservices
- Create service contract testing to prevent breaking changes
- Establish performance testing for the integrated system under load

API Documentation and Versioning *(Owner: Isaac, Christian)*

- Complete API documentation for all services with examples
- Implement an API versioning strategy for backward compatibility
- Create integration guides for each service

Production Readiness

Security Audit and Compliance *(Owner: Christian, Isaac)*

- Complete a comprehensive security audit of all services
- Document security architecture for enterprise validation
- Implement additional monitoring and alerting for security events

Performance Optimization *(Owner: Alberto, Isaac)*

- Complete load testing for 100+ concurrent users

- Implement caching strategies for frequently accessed data
- Optimize database queries and service communication

User Experience and Feature Completion

Frontend Feature Completion *(Owner: Mauricio, Alexis)*

- Complete all remaining user interface components
- Implement comprehensive error handling and user feedback
- Finalize mobile responsiveness across all features

Advanced Feature Implementation *(Owner: All)*

- Complete User Service implementation for team collaboration
 - Finalize Audit Service for comprehensive logging
 - Implement advanced password sharing and permission systems
-

Team Feedback & Commitments

Mauricio Ballart Alvarez (Product Owner)

- **Achievement:** Successfully transitioned from planning to implementation with frontend integration
- **Learning:** The Product owner role requires a balance between technical contribution and stakeholder management
- **Commitment:** Focus on user experience validation and feature prioritization for production readiness
- **Goal:** Complete comprehensive user testing and feedback integration

Alexis Henkel

- **Achievement:** Advanced database architecture and supported frontend development effectively
- **Learning:** Database design complexity requires early integration with microservices architecture
- **Commitment:** Complete database optimization and continue frontend development support
- **Focus:** Ensure database performance meets production requirements

Christian Dominguez (Team Lead)

- **Achievement:** Established effective team leadership and GitHub collaboration framework
- **Learning:** Team leadership requires a balance between technical contribution and coordination
- **Commitment:** Lead security audit completion and maintain team coordination excellence
- **Leadership:** Facilitate production readiness across all team workstreams

Isaac Quintero

- **Achievement:** Delivered exceptional microservices architecture with advanced security features
- **Learning:** Complex infrastructure work benefits the entire team and enables rapid feature development
- **Commitment:** Complete remaining services and maintain system integration excellence
- **Innovation:** Continue pushing boundaries of security implementation and performance

Alberto Espinoza

- **Achievement:** Created a comprehensive DevOps infrastructure supporting all development activities
 - **Learning:** Infrastructure investment early in the project pays significant dividends throughout development
 - **Commitment:** Complete production deployment pipeline and performance optimization
 - **Excellence:** Maintain high standards for testing, monitoring, and deployment automation
-

Sprint Metrics

- **Velocity:** 20 story points completed (target was 20) - 100% completion rate
 - **Technical Achievement:** 7 microservices architecture with advanced security features
 - **Code Quality:** Maintained 90%+ test coverage across all services
 - **Team Satisfaction:** 9/10 (highest rating to date, reflecting major technical achievements)
 - **Innovation Index:** Exceptional - quantum-resistant cryptography and zero-knowledge architecture
-

Risk Assessment for Next Sprint

Managed Risks

- **Technical Infrastructure:** Microservices architecture is now stable and production-ready
- **Team Coordination:** Established effective collaboration patterns and communication
- **Security Implementation:** Advanced security features implemented and tested

Emerging Opportunities

- **Performance Optimization:** System is ready for advanced performance tuning
 - **User Experience Enhancement:** Stable backend enables focus on user experience refinement
 - **Enterprise Readiness:** Architecture supports enterprise-scale deployment and security requirements
-

Innovation Achievements

Technical Innovation

- **Quantum-Resistant Cryptography:** Industry-leading implementation of post-quantum cryptographic algorithms
- **Zero-Knowledge Architecture:** User privacy-focused encryption with user-specific key derivation
- **Microservices Excellence:** Comprehensive 7-service architecture with API gateway integration
- **Advanced Frontend Integration:** Seamless React integration with complex backend services

Academic and Professional Impact

- **Cutting-Edge Security Research:** Implementation of algorithms that represent the future of cybersecurity
 - **Scalable Architecture Design:** Framework supporting enterprise-scale password management
 - **Open Source Contribution Potential:** Architecture and implementation suitable for broader community impact
-

Next Sprint Planning Notes

- **Focus on Production Readiness:** Complete remaining services and comprehensive testing
 - **User Experience Optimization:** Finalize all user-facing features and interactions
 - **Performance and Security Validation:** Complete enterprise-grade validation and documentation
 - **Market Preparation:** Prepare for potential beta testing and enterprise demonstrations
-

Meeting Conclusion:

Exceptional sprint demonstrating the team's ability to deliver complex, production-ready systems with advanced security features. The microservices architecture represents a significant technical achievement that positions the project for enterprise success.

Confidence Level: VERY HIGH - Team has proven ability to deliver complex technical requirements. **Innovation Achievement:** EXCEPTIONAL - Quantum-resistant cryptography and comprehensive security implementation. **Production Readiness:** ON TRACK - Strong foundation for final sprint toward market-ready product

Meeting adjourned at 8:30 PM

Sprint Retrospective Meeting

Project: Cybersecurity: Lock and Key: Encrypted Password Management and Collaboration System

Attendees: Mauricio Ballart Alvarez (Product Owner), Alexis Henkel, Christian Dominguez (Scrum Master), Isaac Quintero, Alberto Espinoza

Date: 07/12/25

Start time: 8:45 PM

End time: 9:45 PM

Sprint Overview

Sprint Duration: 2 weeks

Sprint Goal: Complete frontend foundation implementation, establish collaboration system architecture, develop import/export functionality, and prepare alpha testing infrastructure

What Went Well

Technical Achievements

- **Frontend Integration Success:** Alexis and Isaac's collaboration on API integration resulted in seamless frontend-backend connectivity with robust error handling
- **Performance Excellence:** Alberto's search optimization exceeded expectations, achieving sub-200ms response times for large datasets
- **Design System Maturity:** Mauricio's UI component specifications and style guide created a cohesive, professional user interface
- **Infrastructure Reliability:** Staging environment and deployment pipeline proved stable and secure for alpha testing preparation
- **Security-First Approach:** Christian's collaboration features architecture incorporating security consultation from the start prevented technical debt

Process Improvements

- **Cross-functional Collaboration:** Team successfully coordinated multiple parallel workstreams without integration conflicts
- **Realistic Hour Tracking:** Team maintained consistent 1-2 hour daily commitments while delivering high-quality results

- **Proactive Problem Solving:** Early identification of complexity in permission systems and import/export allowed for proper planning adjustments
- **Communication Excellence:** Daily scrum meetings effectively surfaced dependencies and coordination needs

Team Dynamics

- **Supportive Culture:** Isaac's willingness to assist with encryption requirements and Mauricio's UI translation support demonstrated strong teamwork
- **Ownership Mentality:** Each team member took clear ownership of their domains while remaining collaborative
- **Learning Mindset:** Team adapted well to increased complexity in collaboration features and data validation requirements

What Didn't Go Well

Estimation and Planning Challenges

- **Complexity Underestimation:** Permission system and import/export data validation proved more complex than initially estimated, resulting in incomplete stories
- **External Dependencies:** Security consultant availability and alpha testing user recruitment timelines were not fully accounted for in sprint planning
- **Story Point Accuracy:** Some 3-point stories required 4-5 points of effort, indicating need for better complexity assessment

Resource and Timeline Issues

- **Alpha Testing Recruitment:** User recruitment for alpha testing slower than expected, requiring strategy adjustments
- **Security Review Bottleneck:** Limited security consultant availability created potential timeline risks for collaboration features
- **Technical Debt Accumulation:** Focus on feature completion may have created shortcuts in error handling and edge case coverage

Process Gaps

- **Integration Testing Timing:** Some integration testing happened later in sprint, creating last-minute discovery of complexity
- **Documentation Lag:** Technical documentation didn't keep pace with rapid development, creating knowledge gaps

- **User Experience Testing:** Limited early user feedback on implemented features may have missed usability issues

What We Learned

Technical Insights

- **Security Review Value:** Early security consultation prevented major rework and technical debt in collaboration features
- **API Integration Benefits:** Dedicated frontend-backend integration sessions significantly improved development velocity and code quality
- **Performance Optimization Impact:** Investing in search performance early provides strong foundation for scaling and user satisfaction
- **Design System ROI:** Comprehensive UI component library and style guide dramatically improved frontend development consistency

Process Discoveries

- **Complexity Assessment:** Need for more detailed technical analysis during story pointing, especially for security-related features
- **External Dependency Management:** Earlier engagement with external resources (security consultants, alpha testers) needed for timeline predictability
- **Cross-team Communication:** Regular coordination sessions between related workstreams prevented integration issues

Team Development

- **Skill Complementarity:** Team members' different strengths (Alexis-frontend, Isaac-integration, Christian-backend, Mauricio-design, Alberto-infrastructure) created effective specialization
- **Collaborative Problem Solving:** Complex technical challenges addressed more effectively through pair programming and consultation

Action Items for Next Sprint (Sprint 3)

Estimation and Planning Improvements

Story Complexity Deep Dive (Owner: Christian as Scrum Master)

- Implement "complexity spikes" for technically uncertain stories before sprint planning
- Add security review checkpoint requirements to story acceptance criteria

- Create estimation guidelines for stories involving external dependencies or data validation

Cross-functional Coordination Enhancement

Integration Testing Early and Often (Owner: All)

- Schedule frontend-backend integration sessions at sprint start, not end
- Implement daily integration smoke tests for API endpoints
- Create shared testing scenarios for collaborative feature development

External Dependency Management

Stakeholder Engagement Strategy (Owner: Mauricio)

- Establish regular security consultant check-ins rather than one-time reviews
- Create alpha testing user recruitment pipeline with multiple backup strategies
- Document all external dependencies in sprint planning for timeline buffer calculation

Technical Documentation

Living Documentation Practice (Owner: Alberto)

- Implement documentation-as-code approach with automated updates
- Create technical decision logs for architecture choices and their rationales
- Establish weekly documentation review sessions during daily scrums

Quality Assurance Enhancement

User Experience Validation (Owner: Alexis & Mauricio)

- Implement weekly internal user testing sessions with team members rotating as users
- Create user experience checklists for story completion criteria
- Establish error handling and edge case testing protocols before integration

Sprint 3 Focus Areas

Priority Adjustments

1. **Complete Incomplete Stories:** Finish collaboration features and import/export functionality with proper complexity allocation
2. **Alpha Testing Launch:** Execute refined recruitment strategy and complete staging environment security review

3. **Integration Hardening:** Comprehensive testing of all system components working together
4. **User Experience Polish:** Address any usability issues discovered through internal testing

Process Improvements Implementation

1. **Enhanced Story Planning:** Use complexity spikes for uncertain technical work
2. **Proactive External Coordination:** Front-load security reviews and alpha testing preparation
3. **Documentation Integration:** Make technical documentation part of definition-of-done
4. **Quality Gates:** Implement integration testing checkpoints throughout sprint

Team Feedback & Commitments

Alexis Henkel

- **Achievement:** Successfully established robust React foundation with excellent API integration
- **Learning:** Frontend-backend collaboration sessions dramatically improved development quality
- **Commitment:** Lead user experience validation sessions and continue strong API coordination with Isaac
- **Goal:** Complete frontend implementation for collaboration features with focus on error handling

Mauricio Ballart Alvarez

- **Achievement:** Created comprehensive design system that accelerated frontend development
- **Learning:** Alpha testing recruitment requires more diverse outreach strategies and earlier timeline buffers
- **Commitment:** Execute enhanced alpha testing recruitment and support Isaac with user experience design for import/export
- **Focus:** Balance product owner responsibilities with hands-on frontend work more effectively

Christian Dominguez

- **Achievement:** Successfully designed collaboration features architecture with security-first approach

- **Learning:** Permission system complexity requires more detailed upfront analysis and security review integration
- **Commitment:** Complete collaboration features backend with enhanced complexity management and facilitate integration planning
- **Leadership:** Implement improved story complexity assessment and sprint planning processes

Isaac Quintero

- **Achievement:** Delivered excellent API integration and encryption handling with strong cross-team collaboration
- **Learning:** Import/export data validation complexity benefits from early format analysis and custom parsing prototyping
- **Commitment:** Complete import/export functionality with comprehensive error handling and continue security support for collaboration features
- **Collaboration:** Maintain strong frontend coordination while ensuring encryption and security standards

Alberto Espinoza

- **Achievement:** Exceeded performance expectations for search functionality and created robust staging infrastructure
- **Learning:** Performance testing with larger datasets earlier in development provides better optimization insights
- **Commitment:** Complete staging environment security review and implement enhanced automated testing framework
- **Innovation:** Lead technical documentation-as-code implementation and quality assurance automation

Sprint Metrics

- **Velocity:** 20 story points completed (target was 22) - 91% completion rate
- **Story Completion:** 3 of 5 stories fully completed, 2 stories 60-75% complete (higher quality than Sprint 1's rushed completions)
- **Code Quality:** Maintained automated testing coverage above 90% throughout sprint
- **Team Satisfaction:** 8.5/10 (improved from 8/10 in previous sprint, reflecting better collaboration and technical achievement)
- **Technical Debt:** Minimized through security reviews and integration testing, but documentation debt identified

Risk Assessment for Sprint 3

High Priority Risks

1. **Alpha Testing Delay:** User recruitment challenges could delay testing feedback loop
2. **Security Review Bottleneck:** Limited consultant availability for collaboration features final review
3. **Integration Complexity:** Multiple incomplete stories creating integration complexity

Mitigation Strategies

1. **Backup Recruitment:** Expand alpha testing recruitment to internal networks and professional contacts
2. **Security Review Scheduling:** Book security consultant time in advance with specific deliverable requirements
3. **Integration Planning:** Schedule dedicated integration session at sprint start rather than end

Next Sprint Planning Notes

- **Carry-over Stories:** 8 story points from incomplete collaboration and import/export work
- **New Stories:** Focus on integration, alpha testing launch, and user experience refinement
- **Capacity Planning:** Account for alpha testing support and user feedback integration time
- **External Dependencies:** Pre-schedule security reviews and alpha testing coordination time

Sprint Retrospective Meeting

Project: Cybersecurity: Lock and Key: Encrypted Password Management and Collaboration System

Attendees: Mauricio Ballart Alvarez (Product Owner), Alexis Henkel, Christian Dominguez (Scrum Master), Isaac Quintero, Alberto Espinoza

Date: 10/26/25

Start time: 9:15 PM

End time: 10:00 PM

Sprint Overview

Sprint Duration: 2 weeks

Sprint Goal: Complete production deployment, finalize documentation, validate security and performance for enterprise readiness

What Went Well

Team Collaboration

- Highly synchronized teamwork across security, DevOps, backend, and frontend
- Strong cross-functional support, especially during deployment and security audits
- Effective meetings that ensured alignment on final deliverables

Technical Progress

- Successful production deployment with monitoring, CI/CD, and scale testing
- Full completion of security validation with zero critical vulnerabilities
- Frontend optimizations exceeded performance requirements
- Documentation finalized to enterprise and academic standards

Process Adherence

- Consistent daily scrums and focused coordination meetings
 - All acceptance criteria met on time with zero backlog rollover
 - Quality assurance standards maintained: >90% code coverage
-

What Didn't Go Well

Time Compression Challenges

- High workload during documentation finalization led to late evenings for several team members
- Security audit timing created short testing windows near the end of the sprint

Beta Feedback Integration

- Limited time to incorporate deeper UX feedback from beta testers
- Prioritization was focused on stability, leaving some usability features deferred

Dependency Pressure

- Coordination with external security consultant increased schedule rigidity
 - Production scaling required additional troubleshooting during load testing
-

What We Learned

Technical Insights

- Early setup of monitoring and deployment automation would reduce end-of-sprint pressure
- Third-party validation should be scheduled earlier in the release pipeline

- Smaller documentation increments earlier in the project avoid final rush

Team Dynamics

- Division of responsibilities aligned well with strengths and accelerated delivery
- Collaboration during final security and deployment stages prevented roadblocks
- Clear ownership of tasks improved accountability and efficiency

Process Improvements

- Future sprints should limit “everything must finish at the end” compression
 - Introduce rolling documentation updates every sprint
 - Earlier performance and stress testing improves predictability
-

Action Items for Future Development Cycles

Documentation Strategy (Owner: Mauricio)

- Adopt incremental documentation updates tied to feature completion
- Maintain templates for faster updates during iteration

User Feedback Prioritization (Owner: Alexis)

- Structured usability backlog from beta input
- UX improvements tracked on continuous release basis

Performance & Security Pipeline Enhancements (Owner: Christian & Alberto)

- Shift load testing earlier into sprints

- Add automated security scanning to CI/CD pipeline

Deployment Optimization (Owner: Alberto & Isaac)

- Expand infrastructure redundancy for increased resilience
- Further automate rollback and versioning for safer updates

Team Feedback & Commitments

Alexis Henkel

Commitment: Lead ongoing UX improvements and mobile enhancements

Reflection: Appreciates team responsiveness during performance tuning

Mauricio Ballart Alvarez

Commitment: Maintain clear backlog prioritization for post-launch roadmap

Observation: Product now well-positioned for academic and market success

Christian Dominguez

Commitment: Strengthen proactive security and compliance automation

Action: Continue as governance lead in operational rollout

Isaac Quintero

Commitment: Expand cryptographic validation and backend optimization

Appreciation: Values strong DevOps support during deployment milestones

Alberto Espinoza

Commitment: Maintain and enhance production infrastructure and monitoring

Suggestion: More automated alerts to reduce manual involvement

Next Development Cycle Notes

- Plan roadmap for post-launch enhancements
- Expand beta user pool and incorporate usability insights
- Explore enterprise integration options (SSO, LDAP, admin dashboards)

- Prepare academic publication and conference showcase submissions

Meeting adjourned at 10:00 PM

Sprint Retrospective Meeting

Project: Cybersecurity: Lock and Key – Encrypted Password Management and Collaboration System

Attendees: Mauricio Ballart Alvarez (Product Owner), Alexis Henkel, Christian Dominguez (Scrum Master), Isaac Quintero, Alberto Espinoza

Date: 11/09/2025

Start Time: 6:00 PM

End Time: 7:00 PM

Sprint Overview

Sprint Duration: 2 weeks

Sprint Goal: Stabilize post-production environment, address user-reported issues, enhance usability, and prepare for enterprise pilot rollout

What Went Well

Team Collaboration

- Coordination between backend, DevOps, and frontend teams was seamless during hotfix deployments.
- Quick response cycles for user-reported bugs and feature feedback.
- Strong communication during daily scrums reduced redundant work and improved delivery speed.

Technical Progress

- Successfully implemented production monitoring with error tracking and analytics dashboards.

- Achieved 99.97% uptime during continuous runtime testing.
- Load-balancing and rate-limiting validated in real user scenarios.
- Resolved all medium and high-priority bugs identified during post-launch testing.
- Accessibility score improved to 98% (Lighthouse audit).

Process Adherence

- Maintained consistent sprint cadence and clear task ownership.
 - All production patches documented with rollback notes and version tags.
 - QA maintained >95% test pass rate during patch cycles.
-

What Didn't Go Well

Incident Response Load

- Multiple rapid hotfixes compressed development time for secondary features.
- Occasional late-night deployments caused minor scheduling fatigue.

Documentation Timing

- Deployment pipeline documentation updates lagged behind configuration changes.
- Need clearer version tracking for staging vs. production environment guides.

Feature Trade-offs

- Focus on stability delayed minor UX improvements such as advanced filtering and keyboard shortcuts.

- Localization (multi-language support) postponed to the next sprint.
-

What We Learned

Technical Insights

- Implementing automated post-deployment validation can prevent manual regression checks.
- Standardizing health check endpoints improved error visibility and debugging efficiency.
- Early incorporation of analytics data helps prioritize impactful UX fixes.

Team Dynamics

- Delegating post-deployment monitoring across team members improved coverage and reduced burnout.
- Maintaining a shared sprint board for live status tracking increased transparency.
- Quick internal review cycles boosted turnaround time for small bug fixes.

Process Improvements

- Document infrastructure changes immediately after implementation.
 - Introduce automated rollback validation into CI/CD pipelines.
 - Increase sprint buffer time for user feedback integration.
-

Action Items for Future Development Cycles

Monitoring & Analytics (Owner: Isaac & Alberto)

- Automate post-deployment system validation.
- Implement custom Grafana dashboards for real-time error visualization.

Documentation Workflow (Owner: Mauricio & Christian)

- Update and version environment documentation alongside commits.
- Include “change summary” notes for each CI/CD configuration update.

User Experience Enhancements (Owner: Alexis)

- Revisit deferred UX feedback from beta and production users.
- Implement advanced filtering and accessibility shortcuts.

Security & Compliance (Owner: Christian & Isaac)

- Expand automated security tests in CI/CD.
- Conduct monthly compliance audits for encryption standards.

Team Feedback & Commitments

Alexis Henkel

- **Commitment:** Lead refinement of user dashboard and interactive accessibility tools.
- **Reflection:** Notes that faster bug-triage meetings reduced turnaround time significantly.

Mauricio Ballart Alvarez

- **Commitment:** Strengthen backlog organization and roadmap prioritization.
- **Observation:** Product stability now allows for focus on UX and enterprise features.

Christian Dominguez

- **Commitment:** Continue improving CI/CD governance and compliance oversight.
- **Action:** Introduce automated rollback and security patch scheduling.

Isaac Quintero

- **Commitment:** Optimize backend performance and maintain encryption audit compliance.
- **Appreciation:** Highlights collaborative success during live production testing.

Alberto Espinoza

- **Commitment:** Ensure infrastructure scalability and enhance monitoring alerts.
- **Suggestion:** Proposes a nightly automated build report to improve visibility.

Next Development Cycle Notes

- Begin enterprise pilot testing with controlled user groups.
- Develop extended audit log visualization and admin dashboards.
- Prepare for multi-language (English/Spanish) support phase.
- Plan security compliance review for ISO/IEC 27001 alignment.
- Coordinate upcoming presentation for FIU Capstone Showcase Week.

Meeting adjourned at 7:00 PM

Sprint Retrospective Meeting

Project: Cybersecurity: Lock and Key – Encrypted Password Management and Collaboration System

Attendees: Mauricio Ballart Alvarez (Product Owner), Alexis Henkel, Christian Dominguez (Scrum Master), Isaac Quintero, Alberto Espinoza

Date: 11/23/2025

Start Time: 6:00 PM

End Time: 7:00 PM

Sprint Overview

Sprint Duration: 2 weeks

Sprint Goal: Finalize enterprise pilot deployment, complete remaining usability refinements, ensure system-wide stability, finalize documentation, and prepare the full presentation package for Capstone Showcase.

Sprint Status:

✓ Sprint fully completed — project at ~95% completion.

What Went Well

Team Collaboration

- Extremely strong coordination across backend, frontend, and DevOps during final deployment tasks.
- Smooth handoff between team members for document reviews, UI polish, and compliance checks.
- Communication stayed consistent and productive, especially as deadlines approached.

Technical Progress

- Enterprise pilot environment deployed successfully with stable telemetry and zero downtime.
- Final performance tuning reduced API latency below 100ms across all services.
- Multi-language (English/Spanish) interface fully completed and validated.
- All remaining usability updates implemented based on pilot user feedback.
- Encryption and audit systems passed the final security review with no issues.

Process Adherence

- Daily scrums remained focused and efficient.
 - All tasks were completed inside their assigned story points.
 - Documentation updates were streamlined toward the end of the sprint.
-

What Didn't Go Well

Time Compression

- The final sprint required high concentration and speed, leading to increased workload toward the last few days.
- Some documentation updates were made later than planned due to shifting priorities.

Presentation Preparation Constraints

- Limited buffer time remained for slide polish and demo rehearsals.

- Some last-minute requests for design changes caused minor delays.

External Dependencies

- Pilot user feedback arrived later than expected, forcing quick turnaround on UI adjustments.
 - Coordination with external test users required additional scheduling effort.
-

What We Learned

Technical Insights

- Early preparation of pilot testing environments can reduce final-week pressure.
- Maintaining consistent performance benchmarks made last-week optimization predictable.
- Separating enterprise and non-enterprise logs improves monitoring clarity.

Team Dynamics

- The team functions extremely well during high-pressure stages when responsibilities are clearly divided.
- Pair programming sessions accelerated UI and performance fixes.
- The shared sprint board provided excellent visibility for final deliverables.

Process Improvements

- Future long-term projects should incorporate a dedicated “presentation prep sprint.”

- Documentation should evolve with each feature rather than being heavily updated at the end.
 - A mid-sprint UX review would have spread feedback processing more evenly.
-

Action Items for Future Development Cycles

(Final polish before project completion)

Showcase Preparation (Owner: Mauricio & Alexis)

- Finalize slides, diagrams, and demo route.
- Validate UI flow for live presentation.

Performance & Monitoring (Owner: Isaac & Alberto)

- Run final nightly performance cycle before project freeze.
- Refresh monitoring dashboards for showcase demonstration.

Security & Compliance (Owner: Christian & Isaac)

- Package final compliance reports.
- Prepare security explanation for Q&A segment.

Documentation (Owner: Christian & Mauricio)

- Export final documentation bundle.

- Create repository navigation guide for professors and reviewers.
-

Team Feedback & Commitments

Alexis Henkel

Commitment: Finalize UI polish for showcase and ensure multi-language consistency.

Reflection: This sprint demonstrated strong UI/UX collaboration across the team.

Mauricio Ballart Alvarez

Commitment: Lead final presentation preparation and ensure polished demo experience.

Observation: The system is stable and ready for academic showcasing.

Christian Dominguez

Commitment: Finalize compliance documentation and support Q&A preparation.

Action: Prepare concise explanations for architecture and CI/CD.

Isaac Quintero

Commitment: Finish last performance sweep and prepare talking points for encryption architecture.

Appreciation: Praised the team's collaboration during stress-testing phases.

Alberto Espinoza

Commitment: Present infrastructure overview and ensure demo stability.

Suggestion: Run a final full-system monitoring drill before the showcase.

Next Development Cycle Notes

- Conduct final full-system rehearsal for the showcase.
- Freeze production environment for the presentation week.
- Prepare project submission package and upload final repository.
- Conduct final team review of presentation flow and timing.

Meeting adjourned at 7:00 PM