Ranking	Threats (1)	Vulnerabilities (2)
2 - Very low	Validation failure	SQL Injection Cross-Site Scripting (XSS) Cross-Site Request Forgery (CSRF) Command Injection Buffer Overflow Directory Traversal Insecure Deserialization Open Redirects
2 - Very low	Insecure file upload	Arbitrary File Execution Server-Side Request Forgery (SSRF) Stored Cross-Site Scripting (XSS) File Inclusion Vulnerabilities Directory Traversal Denial of Service (DoS) Malware Upload

2 - Very low	Role assignment vulnerability	Privilege Escalation Unauthorized Access Data Leakage Role Manipulation Access Control Bypass
2 - Very low	JWT security	Signature Bypass Token Hijacking Weak Key Vulnerability None Algorithm Attack Replay Attacks
2 - Very low	API vulnerability	Insecure Direct Object References (IDOR) Broken Authentication Excessive Data Exposure Lack of Rate Limiting Injection Flaws
2 - Very low	Front-end vulnerability	Cross-Site Scripting (XSS) Cross-Site Request Forgery (CSRF) Clickjacking Insecure Direct Object References (IDOR) Client-Side Logic Manipulation

2 - Very low	Validation failure	SQL Injection Cross-Site Scripting (XSS) Command Injection Buffer Overflow Directory Traversal
2 - Very low	Registration key security	Key Forgery Unauthorized Key Distribution Key Generation Algorithm Weakness Replay Attacks Key Storage Compromise
3 - Very low	Insufficient testing	Unidentified Functional Bugs Security Flaws Performance Bottlenecks Compatibility Issues Data Integrity Problems
4 - Low	UX impact	Poor Usability Accessibility Barriers Negative User Perception Decreased User Satisfaction Lower Engagement and Retention Rates

3 - Very low	Data security	Data Breaches Unauthorized Data Access Data Leakage Data Tampering Loss of Data Integrity
6 - Low	Password security	Brute Force Attacks Phishing Credential Stuffing Password Spraying Social Engineering
9 - Medium	Authentication vulnerabilities	Weak Authentication Mechanisms Insufficient Session Management Credential Stuffing Attacks Man-in-the-Middle (MITM) Attacks Account Enumeration and Guessable User Credentials

RISK ANALYSIS TABLE for Condo M

Last Update: 2/7/24

Contextual factors	Risks (3)	Likelihood from 1 to 5
User Input Application Complexity Security Requirements Development Practices Regulatory and Compliance Standards	To be subjected to Validation Failure	1
File Upload Functionality User Base Data Sensitivity Regulatory and Compliance Requirements Infrastructure Security Posture	To be subjected to Insecure file upload	1

Complexity of Role Definitions **Dynamic User Base** **Integration with External Systems** **Level of Access Control Granularity** **Administrative Control Mechanisms**	**To be subjected toRole assignment vulnerability**	1
Token Handling and Storage **Encryption and Signing Algorithms Used** **Token Lifetime Management** **Cross-Origin Resource Sharing (CORS) Policy** **Statelessness of JWT**	**To be subjected to JWT security issues**	2
API Architecture and Design **User Authentication and Authorization Mechanisms** **Data Sensitivity and Privacy Requirements** **API Client Types and Integrations** **Deployment Environment and Infrastructure**	**To be subjected to API vulnerability**	1
Web Application Architecture **User Interaction and Data Handling** **Third-party Libraries and Frameworks Usage** **Browser Security Policies and Features** **Content Security Policy (CSP) Implementation**	**To be subjected to Front-end vulnerability**	1

Input Data Sources and Types **Application Complexity and Technology Stack** **User Privilege Levels and Access Controls** **Development and Testing Practices** **Regulatory and Compliance Requirements**	**To be subjected to Validation failure**	1
Distribution Mechanism for Registration Keys **Key Generation and Validation Processes** **Application Licensing Model** **End-User Environment and Security Awareness** **Software Distribution Channels**	**To be subjected to Registration key security issues**	1
Development Lifecycle and Timeline Pressures **Testing Coverage and Methodologies Employed** **Complexity and Size of the Application** **Availability of Testing Resources and Tools** **Skill and Experience Level of the Development and Testing Teams**	**To be subjected to Insufficient testing**	3
User Expectations and Needs **Application Complexity and Feature Set** **Design and Navigation Consistency** **Device and Platform Compatibility** **Content Readability and Clarity**	**To be subjected to negative UX impact**	2

Data Sensitivity and Classification **Data Storage and Transmission Methods** **Regulatory and Compliance Obligations** **User Access Levels and Privileges** **Third-party Services and Integrations**	**To be subjected to Data Security breaches**	1
Password Policy Strength and Enforcement **User Education and Awareness** **Authentication System Design** **Use of Multi-Factor Authentication (MFA)** **System and Application Access Requirements**	**To be subjected to Password Security breaches**	2
Complexity of Authentication Processes **User Behavior and Password Management Practices** **Deployment Environment Security** **Integration with Third-party Authentication Services** **Regulatory and Compliance Requirements Regarding User Data Protection**	**To be subjected to Authentication vulnerabilities**	3

Management System

Impact from 1 to 5	Reduction of Threats (Acceptance Strategy)
2	Risk Assessment Monitoring and Alerting Incident Response Plan
2	Risk Assessment Monitoring and Alerting Incident Response Plan

2	**Regular Security Audits** **Role-Based Access Control (RBAC) Reviews** **Incident Response Preparedness**
1	**Risk Assessment for Token-Based Authentication** **Implementing Monitoring for Suspicious Activities** **Developing a Token Revocation Strategy**
2	**Regular API Security Audits and Assessments** **Continuous Monitoring and Anomaly Detection** **Preparedness for Incident Response and Data Breach Management**
2	**Security Awareness and Training for Developers** **Regular Security Code Reviews** **Implementation of a Robust Error and Exception Handling Mechanism**

2	**Prioritization of High-Risk Validation Areas** **Implementation of Monitoring and Logging for Anomalies** **Regular Security Training for Developers**
2	**Acceptance of Minimal Key Misuse Risk** **Regular Review and Update of Key Management Policies** **Education and Communication with Users on Key Security**
1	**Prioritization of Critical Functionality and Security Testing** **Incremental Deployment with Rollback Strategies** **Use of Beta Testing to Identify Issues in Real-World Use**
2	**Identification of Minimum Viable Product (MVP) Features for Initial Release** **Gathering User Feedback for Iterative Improvements** **Balancing Functional Requirements with Aesthetic Design**

3	**Risk Assessment to Identify Acceptable Levels of Data Security Risk** **Implementing Data Access and Use Policies** **Planning for Data Breach Incident Response**
3	**Educating Users on the Importance of Strong Password Practices** **Accepting Residual Risk with Enhanced Monitoring for Anomalous Access Attempts** **Regular Review and Update of Password Policies**
3	**Prioritizing Critical Systems for Stronger Authentication Measures** **Implementing Adaptive Authentication Measures Based on Risk Assessment** **Continuous Monitoring for Authentication Failures and Anomalies**

Reduction of Vulnerabilities (Protection Strategy)	Level of residual Risk
Input Validation and Sanitization Secure Coding Practices Access Control and Authentication Mechanisms	2 - Very low
File Type Restriction File Size Limit Virus Scanning Content Validation User Authentication and Authorization Secure File Storage Use of Secure Upload Libraries	3 - Very low

Principle of Least Privilege **Strong Authentication Mechanisms** **Regular Access Reviews and Re-certifications** **Segregation of Duties** **Audit Logging and Monitoring**	**3 - Very low**
Use Strong Signing Algorithms **Secure Token Storage** **Implement Token Expiry and Refresh Mechanisms** **Validate Token Signature Rigorously** **Use HTTPS to Prevent MITM Attacks**	**2 - Very low**
Implement Strong Authentication and Authorization **Data Minimization in API Responses** **Use Secure Communication Protocols (e.g., HTTPS)** **Apply Rate Limiting and Throttling** **Conduct Input Validation and Sanitization**	**3 - Very low**
Implement Content Security Policy (CSP) **Use Anti-CSRF Tokens** **Enable X-Frame-Options Header** **Validate and Sanitize User Input** **Securely Handle User Session and Authentication**	**2 - Very low**

Implement Comprehensive Input Validation **Use Secure Coding Practices** **Employ Automated Security Scanning Tools** **Conduct Regular Code Reviews** **Apply Least Privilege Principle in Access Controls**	**3 - Very low**
Implement Strong Key Generation Algorithms **Secure Key Distribution Methods** **Regularly Update and Rotate Keys** **Apply Hardware or Software-Based Key Protection Mechanisms** **Monitor and Audit Key Usage**	**3 - Very low**
Implement Comprehensive Testing Plans Covering All Aspects of the Application **Utilize Automated Testing Tools for Efficiency and Coverage** **Conduct Security, Performance, and Usability Testing** **Regularly Review and Update Testing Strategies Based on Previous Findings** **Engage in Continuous Integration and Continuous Deployment (CI/CD)** **Practices for Ongoing Testing**	**2 - Very low**
Conducting User Research and Usability Testing **Implementing Responsive and Adaptive Design Principles** **Ensuring Accessibility Compliance** **Regularly Updating UI/UX Based on User Feedback** **Utilizing Design Systems for Consistency**	**5 - Low**

Encrypting Data at Rest and in Transit **Implementing Strong Access Control and Authentication Mechanisms** **Regular Data Security Audits and Compliance Checks** **Data Minimization and Retention Policies** **Continuous Monitoring for Suspicious Activities**	**4 - Low**
Enforcing Strong Password Policies (Complexity, Length, Expiry) **Implementing Multi-Factor Authentication** **Regularly Auditing User Accounts for Weak Passwords** **Encouraging or Requiring Password Managers** **Training Users on Recognizing Phishing Attempts and Secure Password Practices**	**5 - Low**
Enforcing Multi-Factor Authentication (MFA) **Implementing Strong Password Policies and Regular Password Changes** **Secure Session Management with Timeouts and Token Invalidations** **Using HTTPS to Encrypt Data in Transit** **Regular Security Awareness Training for Users on Safe Authentication Practices**	**9 - Medium**