Risk Assessment and Management Plan (RMP)

1. Purpose:

This Risk Management Plan aims to systematically identify, analyze, and mitigate potential risks associated with the development and implementation of the Decagon condo management app and its companion website to ensure a secure and seamless experience. Identifying Potential risk concerns and documenting the mitigation approach to those risks early in the development and deployment processes allows for proactive planning and implementation ensuring the achievement of the project requirements. Moreover, by prioritizing risks based on their likelihood and impact, resources are allocated efficiently, and team members will focus on high-priority risks first.

Furthermore, the risk assessment will be continuously monitored and updated throughout the project lifecycle with open communication with the stakeholders. Thus, this plan aims to minimize the negative impacts of risks while maximizing opportunities for success.

2. Process:

The project manager will serve as the risk manager within this project. The main responsibilities of the risk manager will be formulating and revising the project scope and schedule, engaging with stakeholders, and overseeing the project to ensure the delivery of a high-quality product within the designated timeframe.

3. Risk Identification:

In the early stages of the Decagon condo management app development, our team conducted brainstorming sessions to identify potential risks and areas of concern through open discussion and collaboration. This approach gave us a strong base for risk analysis and mitigation plans.

Table 1: Risk Assessment Overview

Risk ID	Description	Impact	Probability
R1	Unauthorized access to user profiles and data leaks of users' sensitive information	High **Definition:** Sensitive data breaches leading to identity theft and legal consequences. **Explanation:** due to the sensitivity of user information. Identity	High Security risks are very common and no one can achieve 100% security.

		theft and privacy violations are substantial concerns, impacting both users and the reputation of the Decagon project **Quantitative impact:**.could lead to 3-4 weeks delay as breaches require extensive examination	
R2	Users may encounter challenges when using the software if the user interface is complex	Medium **Definition:** Decreases user satisfaction and leads to loss of potential users. **Explanation:** A complex user interface may hinder user satisfaction and discourage potential users. It affects the overall user experience, potentially leading to user dissatisfaction and a decrease in user adoption rates. **Quantitative impact:**.could lead to 2 weeks delay as it takes some time to ensure the UI meets the user needs.	Medium it's common to have complex interfaces in new software
R3	Expertise and knowledge gaps between members are required for software, frameworks, and languages	Medium **Definition:** Can affect code quality and lead to potential delays.	Medium Skill gaps are common in rapidly evolving technology environments

		Explanation: Inadequate expertise and knowledge gaps may result in coding challenges and potential delays. Quality issues in the codebase can emerge, impacting the overall efficiency of the development process **Quantitative impact:**.could lead to 5 days to one-week delay as addressing skill gaps can slow down the project's progress	
R4	Potential miscommunication between stakeholders	High **Definition:** This leads to work overload and project delays. **Explanation:** Miscommunication among stakeholders can create significant work overload and delays, affecting project timelines and potentially leading to a cascading effect on various aspects of the project. **Quantitative impact:**.could lead to 2 weeks delay as it needs additional meetings and communication efforts	High This is a very common issue in projects especially those with multiple stakeholders.

R5	Adequacy of test cases for functional validation	Low **Definition:** Might allow minor bugs in the system. **Explanation:** Inadequate test cases may result in the oversight of minor bugs. While these bugs might be minor, they can accumulate and impact the overall robustness of the system. **Quantitative impact:**.could lead to 5 days to one-week delay as it only affects a few aspects of the software	Medium Depending on the thoroughness of the testing process
R6	Database scalability issues	High **Definition:** Can cause system crashes and performance issues. **Explanation:** Scalability issues in the database can lead to severe consequences such as system crashes and performance degradation, affecting the reliability of the Decagon system. **Quantitative impact:**.could lead to 3 weeks delay as it requires some additional resources	High A common issue in systems with growing data

R7	Miscommunication within the team	Medium **Definition:** This can lead to conflicts and software delays. **Explanation:** Team miscommunication may result in conflicts and delays, hindering the smooth progress of the project. It's crucial to maintain effective communication channels within the team. **Quantitative impact:**.could lead to 3-5 days delay but internal miscommunication can be solved easily	Medium It is common to have miscommunication between members at the beginning Depending on team dynamics and communication processes
R8	Accuracy of condo availabilities and real-time update info	High **Definition:** It affects user satisfaction **Explanation:** Real-time data accuracy is crucial for user satisfaction. Inaccuracies in condo availabilities may lead to dissatisfaction among users, impacting the perceived reliability of the system. **Quantitative impact:**.could lead to 2 weeks delay as it might need extensive testing and revising the data sources	High Real-time data is vulnerable to synchronization issues

R9	Incomplete or inconsistent documentation	High **Definition:** This leads to implementation errors and misunderstandings consequently leading to delays **Explanation:** Incomplete or inconsistent documentation can result in implementation errors and misunderstandings, leading to delays in the development process and potentially compromise the project's success. **Quantitative impact:**.could lead to 2 weeks delay as documentation is an important aspect and could delay other tasks.	High Usually happens in rapidly evolving technology environments
R10	Inaccuracies in displaying condominium fees	Medium **Definition:** Lead to financial discrepancies and user distrust, impacting user satisfaction and financial operations. **Explanation:** Inaccuracies in displaying condominium fees can create financial discrepancies, eroding user trust and	Medium Depending on the accuracy of the data sources and calculation methods

		satisfaction **Quantitative impact:**.could lead to 5 days to one-week delay as it requires additional testing to find the bug	
R11	Inaccurate or incomplete implementation of property dashboard functionalities	Medium **Definition:** Affects the utility and effectiveness of the dashboard **Explanation:** Inaccuracies or incompleteness in the implementation of property dashboard functionalities impact the overall utility and effectiveness of the dashboard, affecting user experience **Quantitative impact:**.could lead to 2 weeks delay to ensure that it meets the user expectations and needs.	Medium Depending on the developer industry experience
R12	Incomplete or inconsistent implementation of personalized notification features	Medium **Definition:** Affects user engagement and satisfaction **Explanation:** Incomplete or inconsistent implementation of personalized notification features can impact user engagement and satisfaction,	Medium Relies on thorough testing and understanding of user preferences

		hindering the effectiveness of communication within the system. **Quantitative impact:**.could lead to 2 week delay as it requires re-working on the functionalities and requires more user acceptance testing	
R13	The risk involves interdependence between backend tasks assigned to different team members.	Medium **Definition:** Can lead to delays as one team member may need to wait for another to complete their task. **Explanation:** This occurs when tasks are interconnected, requiring one team member to finish their part before another can proceed. Such interdependencies can lead to delays in the overall project timeline and bottlenecks. **Quantitative impact:** could lead to 1-2 weeks delay as the progression of one task could be delayed by another.	Medium It is a common issue in software development projects

Table 2: Risk Management Strategies

Risk ID	Risk Type	Risk Score	Resolved in Sprint	Strategy and effectiveness
R1	Security	High	Sprints 3,4	**Mitigation strategy** encourages users to use strong passwords with a mix of uppercase, lower, and special characters and prompts them to update their passwords regularly. Enable multi-factor authentication by receiving codes on their phones.
R2	Technical	Medium	Sprint 2	**Mitigation strategy** Employ user-centred design principles to create a more user-friendly interface. Incorporating feedback to enhance usability and user satisfaction.
R3	Technical	Medium	Sprint 2	**Avoidance Strategy** Provide continuous professional development opportunities for team members. Encourage knowledge transfer between team members.
R4	Management	High	Sprints 2,3,4 as this will require regular communication throughout the sprints	**Mitigation strategy** Implement agile methodologies for continuous feedback. Conduct regular stakeholder and team

				meetings to ensure ongoing clarity of requirements.
R5	Technical	Low	Sprints 2, 3	**Mitigation strategy** Implement automated testing to ensure coverage.
R6	Technical	High	Sprints 3, 4	**Avoidance/mitigation strategy** Implement database scaling and optimization strategies early, ensuring system stability and performance under high loads.
R7	Management	Medium	Similar to R4, Sprints 2,3,4	**Mitigation strategy** Ensures an open communication culture. Conduct regular meetings to discuss any issues.
R8	External	High	Sprints 3,4	**Mitigation strategy** Implementation of real-time data validation and synchronization mechanisms within the software.
R9	Technical and Management	High	Similar to R4, Sprints 2,3,4	**Mitigation strategy** Conduct regular reviews of architecture documentation. Use tools that automate documentation consistency.
R10	Technical	Medium	Sprints 3,4	**Mitigation strategy**

				Implement precise calculation mechanisms, conduct thorough testing, and implement validation checks to ensure an accurate display of condominium fees.
R11	Technical	Medium	Sprints 2,3	**Mitigation Strategy** Conduct comprehensive testing of the property dashboard features while involving stakeholders in validation and verification to identify and rectify any inaccuracies or incomplete implementations. This strategy aims to catch and rectify issues during the testing phase, reducing the likelihood of inaccurate or incomplete property dashboard functionalities affecting user experience.
R12	Technical	Medium	Sprints 3,4	**Mitigation Strategy** Conduct thorough user acceptance testing for notification personalization to identify any inconsistencies or incomplete implementations. User acceptance testing will help ensure that personalized notification features are implemented as intended, minimizing the risk of user dissatisfaction due to overlooked notifications or decreased engagement.
R13	Technical	Medium	Sprints 4,5	**Mitigation Strategy**

				The team plans to closely coordinate the allocation of interdependent tasks during Sprints 4 and 5, ensuring efficient collaboration and minimizing delays.

The risk manager will analyze and identify the impact and probability of each identified risk.

1. **Impact:** In terms of risk assessment, the impact is the potential consequences and effects of each risk, it is categorized as low, medium, and high. It considers many factors such as project delays, disruptions, etc…
    - High: Events or consequences that may cause major delays and significant impact or severe effect on project objectives. Examples for the Decagon Project: Major security breaches leading to data leaks, critical miscommunication causing substantial delays and rework, and major technical challenges leading to extensive delays.
    - Medium: Events or consequences that may cause moderate delays, disruptions, or have a noticeable effect on project objectives. Examples for the Decagon Project: User interface complexities causing moderate dissatisfaction. Some delays in communication led to adjustments.
    - Low: Events or consequences that may cause minor delays, minor disruptions, or have minimal effect on project objectives. Examples for the Decagon Project: Temporary UI glitches, minor miscommunications resolved quickly, minimal deviation from the timeline.

2. **Probability (Likelihood):** It is the possibility of a risk occurring, it is also categorized as low, medium, and high.
    - High: If it has a chance of occurring between ($70\ \% \leq x \leq 100\%$). Examples for the Decagon Project: Frequent security vulnerabilities, and regular communication breakdowns.

- Medium: If it has a chance of occurring between (30% ≤ x < 70%). Examples for the Decagon Project: Usual technical challenges during development, and occasional miscommunication.
- Low: If it has a chance of occurring between (0 % < x < 30%). Examples for the Decagon Project: Minor software glitches, and infrequent communication issues.

3. **Risk level:** This is the combination of the likelihood of risk and its impact.

Table 3: Risk Probability-Impact Matrix

Impact / Probability	Low	Medium	High
Low			
Medium	R5	R2, R3, R7, R10, R11, R12, R13	R4
High			R1, R6, R8,R9

Based on the tables above, risks like R5 have low impact and low probability meaning that they are less critical than others. Risks like R1, R4, R6, R8, and R9 have high/medium impact and probability making them critical risks and team members should focus on them. Other risks with medium impact and medium probability are significant however, they can be solved later and might not occur.

4. Risk Assessment Planning:

Risks will be allocated to the team members for monitoring purposes, and each member will be responsible for monitoring the risks they are assigned to. There are a variety of approaches to handling risks:

1. Avoidance: Changing the project plan to eliminate/avoid the risk caused.
2. Mitigation: Reducing the probability and/or consequences of an adverse risk to an acceptable level by taking early actions.
3. Acceptance: Accepting the risk and not taking any actions to deal with it.
4. Transference: Shifting the consequences of a risk to a third party (ex: use of insurance, warranties)

We'll apply quantitative risk analysis techniques to estimate the potential impact of identified risks on project objectives. This will involve quantifying the range of possible outcomes and their probabilities. Example: Using historical data and simulations, we estimate that database scalability issues (R6) could result in a 15% probability of a 3-week delay and a 5% probability of a 4-week delay.

5. Risk Monitoring and Reporting:

As mentioned earlier, the amount of risks will be continuously measured, managed and updated throughout the project lifecycle. This involves a dynamic process where the project team consistently evaluates risk factors, tracks their status, and adapts strategies accordingly. This proactive approach to risk monitoring and reporting significantly contributes to the overall resilience and success of the project.

We'll implement a quantitative risk monitoring and review process involving regular quantitative assessments of risk status and performance. This will include using quantitative metrics to track changes in risk levels and evaluate the effectiveness of risk mitigation measures. Example: Weekly risk review meetings will include discussions on risk metrics such as the number of open risks, the average time to mitigate risks, and the trend in risk scores over time.

References:

[1] N. Lavanya and T. Malarvizhi, “Risk analysis and management,” Project Management Institute, Mar. 03, 2018.
https://www.pmi.org/learning/library/risk-analysis-project-management-7070