**Anonymous Builders Client/Job Data Application: Risk Assessment Document**

**Risk Identification:**

The identified risks have been organized into four broad categories.

*Development Risks:*

* Withdrawal of Team Member – Should a member of the development team withdraw from the project, our team would be left shorthanded according to the forecasting of our initial project scope and scheduling.
* Technical Limitations of Development Team – The demands of the project might exceed the knowledge, experience, and capabilities of the team.
* Failure to Meet Deadline – The team may fail to complete the application to requirements and/or client satisfaction within the designated deadline for product delivery.
* Failure to Meet Requirements – The complete application may fail to meet one or more of the requirements outlined in previous documents.
* Miscommunication Among Team – Insufficient communication among the members of the remote-working team might result in any number of setbacks to the project, including double-working, overwriting, and confusion of objectives.
* Misalignment of Software Components – The various major modules of the project software, developed with varying degrees of independence, might fail to interact properly. For example, the server-side script might be coded to accept parameters that do not properly correspond to values passed from the client-side script.

*User Risks:*

* Users Cannot Adequately Operate Application – Upon delivery, the application’s intended users are not able to utilize it in the desired manner.
* Malicious Use of Application by Authorized Users – Once logged into the application, users, for whatever motive, may utilize their access to the database’s information for malicious purposes, such as data theft, data corruption, or data deletion.
* Unwanted Software Alteration Post-Production – After product delivery, administrative-level users may willingly or inadvertently alter the structure of the software, rendering it inoperable or faulty.

*External Risks:*

* Security Penetration of Application – A malicious agent may gain unauthorized access to the application and data contained therein.
* Software/Hardware Unavailability/Incompatibility – Upon delivery, it may be discovered that certain hardware and/or software employed by the client is no longer compatible with the final product or is otherwise unavailable.
* Infrastructure Disruption – Upon delivery, unforeseen complications, such as inclement weather, accidents, etc. might make the electricity supply or internet access at AB’s office unavailable, thus rendering them temporarily incapable of performing vital business functions.

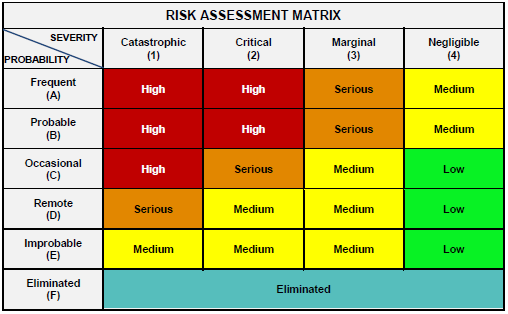
*Data Risks:*

* Database loss/corruption – Customer and/or job data stored within the database could become corrupted or deleted, impacting AB’s ability to perform necessary duties.
* Data Exposure – Customer and/or job data stored within the database could be accessed by parties not authorized, which may or may not then be further disseminated outside of AB’s personnel.
* Loss of Cloud Hosting – Through equipment casualty, service cancellation, etc. from AB’s cloud hosting provider, Amazon Web Services, AB might find themselves in a situation where they are unable to access their cloud database and application data.

**Risk Analysis:**

Our team has assessed the above risks and prioritized them according to probability of occurrence and severity of consequences. NOTE: Given that our application, at least in its incarnation for this project, involves no physical safety, public infrastructure, environmental hazards, financial data, or the like, we did not deem the severity of any risk to qualify as “Catastrophic.”

*The following Risk Assessment Matrix was used as a guide to categorize the listed risks.*



|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Probability** | **Severity** | **Priority#/Risk Rank** |
| Miscommunication Among Team | **Probable**  These occurrences are likely to happen to some degree. | **Marginal**  While often easily rectified, some instances may lead to project delays, missed deadlines, and incompatibilities. | **1-Serious** |
| Misalignment of Software Components | **Probable**  Fairly likely to occur to some degree in a team project, possibly exacerbated by remote work. | **Marginal**  Though quite correctable, the degree of the misalignment could threaten deadlines. | **2-Serious** |
| Failure to Meet Deadline | **Probable**  Any number of factors might contribute to a situation where a deadline might prove difficult to meet. | **Marginal**  A product that does not meet client deadline will delay their desired switchover to a new system and may result in renegotiation of contract and/or damage to our team’s reputation. | **3-Serious** |
| Failure to Meet Requirements | **Occasional**  It is quite possible, due to skill or time constraints, that some requirements may remain unfulfilled. | **Marginal**  A product that does not meet client requirements could delay their desired switchover to a new system and may result in renegotiation of contract and/or damage to our team’s reputation. | **4-Medium** |
| Technical Limitations of Development Team | **Occasional**  With a relatively inexperienced team, even a project of this modest scope may test our abilities at points. | **Marginal**  While research and teamwork can often overcome these obstacles, some instances may lead to project delays, missed deadlines, and missing requirements. | **5-Medium** |
| Database Loss/Corruption | **Remote**  Though data loss through malicious action or technical malfunction is possible, it is not particularly likely to occur. | **Critical**  Such a loss could threaten our client’s data pool and their ability to conduct business in the short-term and even beyond. | **6-Medium** |
| Malicious Use of Application by Authorized Users | **Remote**  We cannot rule out the possibility of sabotage by someone like a disgruntled current/former employee. | **Critical**  This access, particularly if at an administrative level, could threaten our client’s data pool and their ability to conduct business in the short-term and even beyond. | **7-Medium** |
| Security Penetration of Application | **Remote**  Though possible, especially if client’s employees exhibit lax security practices, our client’s application and data does not represent a particularly tempting target. | **Critical**  This access, particularly if at an administrative level, could threaten our client’s data pool and their ability to conduct business in the short-term and even beyond, as well as damage the reputation of our team. | **8-Medium** |
| Software/Hardware Unavailability/Incompatibility | **Occasional**  Incompatibility with the existing technical environment, even in relatively simple software systems, is not uncommon. | **Marginal**  Could impair functionality, necessitating that client purchase new equipment, keep backups on standby, or limit personnel able to access the system temporarily. | **9-Medium** |
| Unwanted Software Alteration Post-Production | **Remote**  In normal circumstances, this is not likely to occur unintentionally. | **Critical**  Depending on the nature of the alteration, the application could be rendered temporarily unavailable or faulty, impacting business functions. | **10-Medium** |
| Loss of Cloud Hosting | **Remote**  The host profits from this relationship with our client and seeks to provide a reliable service, making such a loss possible, but unlikely. | **Critical**  Until able to migrate data to new host, the client would be unable to access records and perform vital business functions, possibly resulting in lost revenue, customer dissatisfaction, and tarnished reputation. | **11-Medium** |
| Withdrawal of Team Member | **Remote**  While a team loss due to unforeseen circumstances is always possible, it is unlikely at this stage of our project. | **Marginal**  The project could proceed largely as planned with the loss of a team member, but workloads would increase, leading to possible missed deadlines. May need adjustment of contract, deadlines, or requirements. | **12-Medium** |
| Data exposure | **Remote**  Though access to data is limited to authorized users, the possibility of malicious exposure cannot be ruled out. | **Marginal**  While privacy is preferred, the data currently used by the project is not particularly sensitive in nature. | **13-Medium** |
| Infrastructure Disruption | **Remote**  While not likely to happen often nor for long durations, outages are common enough that they must be taken into consideration. | **Critical**  Though not involving permanent losses, our client’s business functions may be severely inhibited for an indefinite period. | **14-Medium** |
| Users Cannot Adequately Operate Application | **Occasional**  While straightforward in nature, it is not unlikely that some employees may not immediately grasp the application’s UI. | **Negligible**  This setback would be limited in scope, likely temporary, and easily remediated. | **15-Low** |

**Risk Planning:**

*Summary:*

In rationalizing our priorities, it was clear that our most serious risks were assessed as such based on their high probability of occurrence. While more critical risks existed, most were deemed relatively unlikely to occur. Fortunately, most of the high-probability risks fell within the development process itself, meaning mitigation could be applied early in the project’s overall progress. Also to our favor is the fact that all risks identified can be mitigated in some way, reducing their likelihood, severity, or both. It should be noted that no identified risks were able to be completely eliminated. What follows is an itemized risk mitigation plan, followed by a revised risk assessment table that reflects the expected risk levels after mitigation is applied.

*Mitigation Actions:*

* Miscommunication Among Team – All members will utilize periodic meetings, Discord chats, GitHub comments, and code comments to reduce the likelihood of miscommunication. Josh will act as team coordinator and strive to ensure team members understand their objectives and deadlines. He will also assess progress from all project areas to ensure everyone has a common understanding of target goals for each phase.
* Misalignment of Software Components – Adherence to Design Document will be expected, which will help to ensure the disparate components, their output, and parameters all align correctly for proper application function. Members will promptly communicate any lack of clarity involving the interactions of their respective modules.
* Failure to Meet Deadline – To reduce the chances of our team failing to deliver the final product by the deadline, members are expected to adhere to the Project Schedule and the associated goals and milestones. Any delays will be reported and discussed as early as possible to allow shifting of resources or reassessment of feasible requirements. For this project, meeting the final deadline and providing a functional product will be prioritized over completion of every client requirement, should this trade-off become unavoidable.
* Failure to Meet Requirements – To reduce the chances of our team failing to meet all client requirements, members are expected to maintain the objectives of the Requirements Document. Any obstacles will be reported and discussed as early as possible to allow for shifting of resources or assistance researching solutions. Meeting all requirements shall not be prioritized over delivering a functional product by the project deadline; the team will work together to decide what requirements shall be omitted, if necessary.
* Technical Limitations of Development Team – This risk has been mitigated by designing the project to fall within the capabilities and experience of our team members as much as was feasible. Novel technologies will still require some degree of research and learning, which will be frontloaded as much as possible before actual coding begins. Technologies selected were partially chosen for their ubiquity in the field, reliable documentation, and abundant tutorials.
* Database Loss/Corruption – Database access will be designed to prevent record duplication or broken associations. Loss will also be mitigated by utilizing the database backup features available to AWS clients, allowing for data recovery or rollback of changes. Direct access to overall database scheme and configured options will be restricted to the AWS account administrator.
* Malicious Use of Application by Authorized Users – User Documentation provided to the client upon product delivery will include guidance regarding the prompt revocation of application access authorization for terminated employees or others that the client’s management believes may misuse their access. Restricting administrative level access to minimal necessary users will also mitigate potential harm.
* Security Penetration of Application – The risk of a hacking attempt will be mitigated by several controls. First, access to the application’s main functions will be controlled by username/password login, with a server-side API will providing an intermediate layer between client-side interaction and the actual database contents. Second, the program will require a minimum level of password complexity. Third, administrative-level access will only be achievable through the AWS account with two-factor authentication. Fourth, AWS database backup will ensure that malicious alterations through a hacked application can be reverted at the administrative level. Fifth, severity of a successful penetration is mitigated by the fact that the database information for this application is not sensitive in nature.
* Software/Hardware Unavailability/Incompatibility – This risk has been mitigated by deliberately targeting and testing our application for only PCs and Android mobile devices utilizing Chrome and Firefox browsers. The ubiquity of these platforms should ensure that compatibility is easily attained or, at the very least, suitable replacements can be readily procured.
* Unwanted Software Alteration Post-Production – The primary mitigation here is that only the AWS administrators will have access to the application code and database. User Documentation will clearly outline for the administrators which files/settings should not be altered. AWS backup features will allow for recovery from alterations if needed.
* Loss of Cloud Hosting – We hope to mitigate this risk by choosing Amazon Web Services as a provider; their considerable resources and widespread use by many enterprises provides some level of reassurance that the frequency and duration of hosting outages will be minimal. The easy generation of new virtual resources in a cloud environment also makes unlikely the possibility of a complete outage of service. Upon delivery of the completed product, our team will also ensure that our client, as requested, is set up with automatic payments to AWS, thereby minimizing the chance of service interruption due to non-payment. In a worst-case scenario, the database and application files could be installed on another provider quite easily, due to the cloud nature of the deployment.
* Withdrawal of Team Member – While the probability of this risk cannot be mitigated directly, the impact can be minimized by ensuring that all members’ progress is carefully documented and accessible. If all members have a fundamental awareness of all project areas, a vacated position can have its duties more easily assumed by another.
* Data Exposure – The same factors that mitigate unauthorized, unwanted, or otherwise malicious apply to this risk as well. Furthermore, User Documentation will advise the client to train users not to input sensitive data (SSNs, Credit Card Numbers, etc.) into the application in ways not intended (such as insertion into text area inputs like the Job Description field). The risk of data exposure is thereby mitigated by simply not including data that would be a tempting target or damaging if obtained.
* Infrastructure Disruption – While interruptions to electricity or internet utility service to our client is obviously not within the development team’s purview, the fact that we have designed our product to operate within the cloud and on web browsers not tied to a specific enterprise ensures that our client’s personnel could relocate elsewhere (perhaps working remotely from home) and still perform their job functions using the application. Furthermore, a cellular connection on a laptop or Android device could be employed to access all database and administrative functions if needed.
* Users Cannot Adequately Operate Application – Our application will be designed to be as simple and user-friendly as possible, per the client’s requirements. User Documentation will also include instructions on how to properly utilize the application, helping to remediate any usage difficulties our client’s personnel may encounter.

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Probability** | **Severity** | **Risk Rank** |
| Miscommunication Among Team | **Occasional** | **Marginal** | **Medium** |
| Misalignment of Software Components | **Occasional** | **Marginal** | **Medium** |
| Failure to Meet Deadline | **Occasional** | **Marginal** | **Medium** |
| Failure to Meet Requirements | **Occasional** | **Marginal** | **Medium** |
| Technical Limitations of Development Team | **Remote** | **Marginal** | **Medium** |
| Malicious Use of Application by Authorized Users | **Improbable** | **Critical** | **Medium** |
| Security Penetration of Application | **Improbable** | **Critical** | **Medium** |
| Unwanted Software Alteration Post-Production | **Remote** | **Marginal** | **Medium** |
| Loss of Cloud Hosting | **Improbable** | **Marginal** | **Medium** |
| Withdrawal of Team Member | **Remote** | **Marginal** | **Medium** |
| Infrastructure Disruption | **Remote** | **Marginal** | **Medium** |
| Database Loss/Corruption | **Remote** | **Negligible** | **Low** |
| Software/Hardware Unavailability/Incompatibility | **Remote** | **Negligible** | **Low** |
| Data exposure | **Remote** | **Negligible** | **Low** |
| Users Cannot Adequately Operate Application | **Remote** | **Negligible** | **Low** |