

1.0 Business Practices		Yes	No	Comments
1.1	Understand what business scenarios a vendor is or is not willing to enter into a contract.			
1.2	Have we requested 6 to 10 user account references (this should include 3 to 5 accounts that "kicked them out" so the "great" references comments are put into perspective)?			
1.3	Have we contacted industry analysts or associates who know the potential vendor to determine what type of reputation they have and what impact this could have on our project?			
1.4	Have we obtained names of companies with whom they have worked with before?			
1.5	Have we requested a review of the Financials (either Annual Report or their most recent private audit) so we can determine the financial stability of the company?			
1.6	If the company is private, have we done credit checks on the principals of the company?			
2.0 Project Management Practices		Yes	No	Comments
2.1	Do they have someone who will be fulfilling a Project Manager type function?			
2.2	Are they willing to work with our Project Manager (basically do they expect to have total control of the project or are they willing to take direction from us)? And how will this interaction specifically occur?			
2.3	Who is to be the focal point for communication on their end? Does this vary by functional area?			
2.4	Have we told them and do they understand that we own the product specification?			
2.5	Have we told them and do they understand that NOTHING goes into the product without our knowledge?			
2.6	Have we told them and do they understand that we own the Project Change Request sign-off, and that they can make no changes to			

requirements without our approval?			
2.7 Have we made clear to them what the Project Change Request process is?			
2.8 Where does their software development, risk management, quality assurance, and configuration management processes differ from ours and how do we merge the two, with the least pain to them and still meet our quality bar?			
2.9 Have we provided them access to our intranet, bug database, build process?			
2.10 Have we reviewed their processes, such as:			
• Technical reviews			
• Customer Focus Groups			
• Source code control			
• Configuration management			
• Builds			
• Status reporting			
• Change management			
• Estimation			
• Design documentation and notations			
• Project Retrospectives			
• Project planning			
• Project tracking and performance measurement			
• Development metrics			
• Coding standards			
• Unit testing			
• System Integration and Integration testing			
• Prototypes			

<ul style="list-style-type: none"> <li>Risk management in plans</li> </ul>			
<b>3.0 Software Development Practices</b>	<b>Yes</b>	<b>No</b>	<b>Comments</b>
3.1 Have we requested and received an org chart of the proposed team?			
3.2 Have we reviewed resumes for every developer?			
3.3 Do we have a clear indication of percentage of time each developer will spend on project?			
3.4 Have they worked together before?			
3.5 Have we asked and have they explained their technical approach to the project?			
3.6 Have we worked together to define milestones and approximate target dates (there should never be more than six weeks between one independently measurable event and the next)?			
3.7 Are they using existing components as part of the product? If so, have we requested and received the following info:			
<ul style="list-style-type: none"> <li>A list of existing components that will be part of the product</li> <li>Where did the component come from (i.e.: in-house, contract, published product)</li> <li>Has it used before? — in what? how long has it been in use?</li> <li>How big is it (in KLOC)</li> <li>Licensing/support arrangements for out-of-house components</li> </ul>			
3.8 Have we reviewed the track record of the team with regards to:			
<ul style="list-style-type: none"> <li>On time delivery</li> <li>Previous localized products</li> <li>Frequency of emergency fixes and maintenance releases</li> <li>Did they face tough performance or working set goals?</li> <li>Have they developed using alpha or beta OS and tools?</li> <li>Have they worked with NT? Windows? C++?</li> <li>Have they had penalty clauses or very tough deadlines?</li> </ul>			

<ul style="list-style-type: none"> <li>• What is their experience with multitasking/multithreaded applications?</li> </ul>			
3.9 Have we reviewed the tools and special equipment they plan to use, such as: <ul style="list-style-type: none"> <li>• Compilers</li> <li>• Linkers</li> <li>• Source code control</li> <li>• Test harness</li> <li>• Analyzers</li> <li>• Debugging tools</li> <li>• Amount of hardware per developer</li> <li>• Bug tracking tools/methods</li> </ul>			
<b>4.0 Quality Assurance Practices</b>	<b>Yes</b>	<b>No</b>	<b>Comments</b>
4.1 Do they have a test team in place?			
4.2 Are testers dedicated exclusively to testing tasks?			
4.3 What is average duration that each tester has been testing?			
4.4 What is the highest level of tester expertise?			
4.5 What is developer/tester ratio (recommend 2:1 max.)?			
4.6 Do they have a formal method of classifying bugs?			
4.7 Do they have a bug tracking database?			
4.8 What equipment is available to each tester (recommend 1 high end and 1 low end machine)?			
4.9 Are they equipped to do configuration testing?			
4.10 Has the test process been institutionalized?			
4.11 Have they written a test plan that defines the test process?			
4.12 Do they have formal measures of quality?			
4.13 Do they measure the test process relative to goals?			

4.14 Is the test process managed based on previous experiences?			
4.15 Are the effectiveness of techniques, tools, and methods measured for improvement and optimization?			
4.16 Have we discussed and documented who will be responsible for and under what circumstances Quick Fix Engineering will take place?			
<b>5.0 Product Documentation Practices</b>	<b>Yes</b>	<b>No</b>	<b>Comments</b>
5.1 Have we reviewed how they do documentation with regards to:			
• Who writes?			
• What is their process?			
• How does the writing cycle fit in with the development cycle?			
How is the documentation tested?			
5.2 Do we know what is their experience with on-line documentation?			
5.3 Have we reviewed samples of their work?			
5.4 Have we asked for them to provide any or all of the following:			
• Any technical documentation they might already have or could produce that would help understand their product.			
• Any end-user documentation the vendor might have already produced, even if we're not planning on using it. Can this be supplied in electronic form?			
• The vendor must be prepared to participate in the technical review cycles of the documentation.			