

Basic Technical Interviewing

Jan Pool

NioCAD – University of Stellenbosch

29 August 2007

Who am I?

- Studied electronic engineering at the University of Stellenbosch.
- Worked as an electronic/software engineer and project manager at Stone Three.
- Project manager at NioCAD.
- Mostly done technically oriented software development.

What makes me qualified to give advise about interviewing?

- Nothing whatsoever ...
 - I have no formal training.
- But,
 - I have performed a number of interviews recently.
 - Have done some research on the subject.
- Disclaimer
 - I'm a mere mortal and does not claim to be an expert on the subject.

Target Audience

- Primary: Persons who needs to interview and hire technical personnel.
- Secondary: Technical employees.
- Organisations of all sizes, especially those organisations with no interview process.
- This presentation is customised for the software development industry.

Sources

- Joel Spolsky (Joel on Software):
 - <http://www.joelonsoftware.com/articles/ThePhoneScreen.html>
 - <http://www.joelonsoftware.com/articles/GuerrillaInterviewing3.html>
- Steve Yegge (Google):
 - <http://steve-yegge.blogspot.com/2006/03/truth-about-interviewing.html>
 - <http://steve.yegge.googlepages.com/five-essential-phone-screen-questions>
- Personal experience.
- Discussions with friends and colleagues.

Brief Overview of the Recruitment Process

- Identify the need.
- Advertise.
 - Be specific to limit noise.
- Screen resumes.
 - Use objective measures.
 - Never hire based on resume alone.
- Interviews.
- Make offers to suitable candidates.
- Appoint best available candidate.
- Determine candidate is suitable during probation period.
- Repeat process if necessary.

Purpose of the Interview

- Yet another filter process.
- Determine if there is a match between the candidate and the organisation.
 - Skills
 - Personality
- Look for people who:
 - are “Smart and can get things done” (Spolsky),
 - can think about a problem at different levels of abstraction.
- Opportunity for the candidate to evaluate the company as well.

Some Interview Problems

- Not enough time is spend on interviewing candidates.
- Candidates are often not required to demonstrate their skills.
- Interviewers do not try to determine if the needs and career plan of the candidate matches the requirements of the company in the long run.

General Tips

- People are the most important part of software project, you should always try to hire the best people you can afford.
- Never hire a maybe. It is much better to miss out on a good candidate than to hire a bad one.
- More than one person should interview the candidate, including prospective peers.
- Technical questions must be asked by competent persons.
- Candidates must demonstrate their skills during the interview.
- However, do not hire simply based on current skill set unless you hire a short term contractor.
- Look for people who are passionate about their field.
- Look for adaptable people.
- Let the candidate do most of the talking. Lean towards open ended questions.

General Tips

- It is important to try to determine if the candidate will fit into the organisation's culture.
- Try to determine if the candidate can grow in your organisation.
- Try to determine if the candidate can productively work independently and as part of a team.
- Ensure you and the candidate are looking for the same kind of development.
 - Web, embedded, rich client, GUI, financial, engineering, specialised, etc.
- Gracefully terminate the interview early if you are sure candidate is not suitable.
- Remember, the interview is often the only change you have to evaluate the candidate, make it count!
- More selective recruiting can minimise staff turnover.

An Interview Process

- NioCAD's Interview Process (adapted from Spolsky and Yegge)
 - Overview of organisation and/or project.
 - Discuss recent project the candidate completed.
 - Technical questions and skill demonstration.
 - Social and personality questions.
 - Wrap-up.

Overview of Organisation

- Briefly describe the interview process to the candidate.
- Provide a short history of the organisation.
- Describe your business.
- Short sales pitch.
 - What makes your organisation such a good place to work for?
- Explain what skills and types of people you are looking for.
- Good time to determine if you and the candidate can meet each other's requirements.

Recent Projects

- Ask questions about recent project the candidate were involved with.
- Does he understand, and can he explain what he actually did.
- Try to determine what role they played in the project (leadership, etc.)
- Get a better idea of the candidate's past experiences.
- How passionate is the candidate about his past projects.
- How does he talk about his ex and current colleagues.
- Does the candidate understand how these projects fit into the organisation's business plan?

Technical Questions

- Want to get a feeling of the candidate's technical capabilities.
- Important to let the candidate demonstrate his skills.
- Does the candidate understand the subject matter on a deep level?
- Good place to determine if the candidate is smart and can get things done.
- Does the candidate use and understand industry standard terminology.
- Start with easy questions.
- Ask open ended or "impossible" questions as well to get an idea of how the candidate go about solving problems.

Technical Question Examples

- Our technical questions are divided into seven categories (adapted from Yegge).
 - Bits and Bytes
 - Data structures
 - Coding
 - Scripting
 - GUI
 - OO and Design
 - Threading and Concurrency
- One should adapt the questions to the type of development the organisation does.
- Only ask questions that you can solve as well!
- See Yegge for many examples.

Technical Question Examples

- Bits and Bytes
 - Must know what bits and bytes are.
 - Count in binary and hex and convert between binary, decimal and hex.
 - Know how the logical operators (and, or, not, etc.) works and how to use them.
 - Difference between logical and bitwise operators.
 - Have an idea of size and limits of basic data types (int, float) on current system architectures.
 - Signed vs. unsigned types.

Technical Question Examples

- Data Structures
 - Demonstrate an understanding of the basic data structures.
 - Understand “Big-O” and can classify operations on common data structures.
 - Understand the difference between abstract and concrete data types.
 - Concrete data structures that the candidate must understand:
 - Arrays, Vectors, Linked Lists, Hash Tables, Trees and Graphs
 - Must be to explain how to use these data structures, when to use them, the operations they provide and their performance trade-offs.

Technical Question Examples

- Coding
 - Candidates should write code in his favourite language and use correct syntax.
 - Good questions has these properties:
 - It is simple and easy to solve (few minutes).
 - Has loops and recursions.
 - Has formatted output.
 - Has text file I/O.
 - Is testable.
 - Ask how the candidate will test his code.
 - Valid inputs, boundary conditions, test cases, etc.

Technical Question Examples

- **Scripting**
 - Does the candidate know how to use scripting languages effectively?
 - Knowledge of basic tools such as regular expressions.
 - You don't want candidates who implement something in C++/Java that can be done quickly with a small script or shell commands.
- **GUI**
 - Demonstrate basic knowledge of user interfaces.
 - Model-View-Control Pattern.
 - How is events handled.
 - Graphic knowledge.

Technical Question Examples

- OO and Design
 - The candidate should be able to define common OO terms:
 - Constructor, class vs. object, encapsulation, class/static methods, abstract/virtual, is-a and has-a relationships.
 - The candidate should be able to design.
 - What classes?
 - What methods and in what classes?
 - What data structures used?
 - Draw pictures.
 - UML knowledge.
 - Give candidate a small design and ask them to implement the design in code.
 - Design pattern knowledge.

Technical Question Examples

- Threading and Concurrency
 - This becomes more and more important.
 - Candidates must understand the issues arising when building a multi-threaded system
 - Shared data.
 - Synchronisation.
 - Mutexes, semaphores.
 - Synchronous vs. Asynchronous.

Social and Personal Questions

- Assess the candidate's personality and work ethic.
- Simply chat with the candidate.
- Ask about candidate's hobbies and what he does to relax.
- Determine if the candidate:
 - will fit into the organisation's culture,
 - has a sense of humour,
 - can handle pressure,
 - is the candidate the kind of person who will commit to goals and deadlines,
 - looks after his own personal and career growth.
- What is the candidate's proudest professional moment. Worst?

Wrap-up

- Ask the candidate if he/she has any more questions.
- Ask when the candidate can start.
- Let the candidate know when he can expect to hear from you.

Recap and Final Advise

- Interviewing is hard and not a perfect method to determine if you should hire someone.
- Only interview if you are sure you know what kind of person and skills you require.
- A process helps.
- Prepare questions beforehand.
- The candidate must demonstrate his technical skills.
- But do not look at specific and current skills only.
- Look for people who are
 - Intelligent,
 - can get things done,
 - passionate about their field,
 - adaptable.
- I also prefer candidates who are committed to personal growth and education.
- Never, ever hire a maybe.

Questions and Suggestions?

Contact Information

- jpool@pshymorphic.com
- <http://research.ee.sun.ac.za/niocad/>