

Interview (screening) questions

X-factor

Potential

0	1	2	3	4	5
---	---	---	---	---	---

What potential does this candidate have to be a standout? Can they move up a level? When?

Flexibility

0	1	2	3	4	5
---	---	---	---	---	---

Could this candidate fill a another role? A tester or business analyst role? A sys. admin., DBA, PM role, as needed? Can they work on a variety of applications/technolgies.

Leadership

0	1	2	3	4	5
---	---	---	---	---	---

In a group setting, does this candidate find a way forward? Will others look to them for guidance?

Discipline

0	1	2	3	4	5
---	---	---	---	---	---

Does this candidate demonstrate a commitment to disciplined development practices? Is the candidate familiar with different ways of working and the tradeoffs involved? Has the candidate been in a place that used more than just an "ad hoc" development methodology?

Passion and learning

0	1	2	3	4	5
---	---	---	---	---	---

Is this candidate likely to go the extra mile in developing and sharing his or her skills and knowledge? Would they inspire others to improve?

Talent

0	1	2	3	4	5
---	---	---	---	---	---

What unique strengths, talent, knowledge, and experience does this candidate bring that might add to our team?

General Programming

1. Object-oriented design and development

0	1	2	3	4	5
---	---	---	---	---	---

Possible questions :

- What makes code object oriented?
- What principles do you consider when designing classes?
- How long should a method be? A class?
- What kind of code do you put in a service layer, persistence layer, application/controller layer, etc.?
- What's a correctly named method?
-

What we're looking for: Mention of state and behavior belonging together. An understanding of at least the ideas behind some OO [principles](#). Some awareness of the importance of short, meaningfully named methods and classes. An idea of why and how we layer an application.

2. Testing

0	1	2	3	4	5
---	---	---	---	---	---

Possible questions:

- How do you know when the code is done?
- What's the difference between unit testing and integration testing?
- How do you write tests?
- How much code coverage is the right amount?
- What kinds of tests do you do/write first?
- What makes code hard to test?
-

What we're looking for: Knowledge of the role of regression testing (checking) vs. exploratory testing/acceptance testing vs. TDD as a design activity. An idea of why you would use a mock object (dummy object, test stub, test spy, etc.). The idea that some kinds of tests are more valuable than others for different purposes. An idea of where to start testing **and** where to stop!

3. Other tools and practices - SCM, Maven, continuous integration, change and release management, developer tools

0	1	2	3	4	5
---	---	---	---	---	---

Possible questions:

- Where do you keep code?
- How does your team demo the software?
- Where do you keep track of what you've changed?
- How do you determine version numbers?
- How do you compile/build a project?
- How often do you commit code?
- What do you typically do before a release?
- What kind of branching policies do you or your team use?

What we're looking for: Familiarity with source control and versioning. An awareness of the software development lifecycle beyond coding. Some level of discipline about identifying and tracking changes. Experience with common continuous integration tools and patterns.

4. Agile and other development practices

0	1	2	3	4	5
---	---	---	---	---	---

Possible questions:

- How do you know what to do?
- What makes a good story/requirement?
- How often do you/your team release software?
- Which part of an application do you develop first?
- When you're stuck on a problem what do you do?
- How often do you work on somebody else's code?
- How long does it take for you/your team to develop an average size feature.
- How do you know how long something will take? What do you typically do before a release?

What we're looking for: Someone who knows what a prioritized backlog and iterative/incremental development is. Stories that talk about who will use a feature w/ have clear acceptance criteria. Short releases, with opportunities for feedback. Working in vertical slices. Working with a pair. The value of small chunks of work. Not working in isolation. An idea of the whole software development lifecycle.

5. Frameworks and languages

0	1	2	3	4	5
---	---	---	---	---	---

Possible questions:

- What are some of the parts of the Spring framework?

- What are some of the advantages of using a framework?
- What other frameworks have you used (MVC, DI, persistence, front-end) and why did you choose those?
- How does Java make class design easier? Harder?
- What other languages have you used?

What we're looking for: More knowledge about the strengths and weaknesses of different frameworks. The breadth of different technologies used.

6. Persistence

0	1	2	3	4	5
---	---	---	---	---	---

Suggested questions;

- When would you use Hibernate, iBatis (or another mapping framework), JDBC?
- What's the advantage of Spring's JDBC template?
- How can you represent common object relationships in the physical data model (the database)? What's the advantage of NoSQL approaches?
- Where would you typically place the transaction boundaries in a layered web application? Why?
- How do you evolve a database schema?

What we're looking for: Knowledge of common database and persistence patterns. A candidate can demonstrate strength with DBA skills or deep knowledge of one or more persistence frameworks.

7. Applications experience

0	1	2	3	4	5
---	---	---	---	---	---

Possible questions:

- What kinds of workflow applications have you worked on?
- How do those compare with reporting applications or data entry applications?
- What other kinds of business processes have you worked with?
- Any consumer/retail or game experience?
- How about ETL/backend processing? Vendor applications?

What we're looking for at the senior/lead level: Someone who has worked on different types of applications (not just different technologies, architectures or frameworks). Someone who has spent time learning different generic domains. Someone who has an understanding of more than one specific domain. At the intermediate level, they may only have significant experience with a single

domain, but they should still be able to demonstrate interests outside of that domain.

8. General application/architecture knowledge

0	1	2	3	4	5
---	---	---	---	---	---

Suggested questions:

- Tell us a little about your experience with front end development, responsive design, single page applications. MVC and request/response web applications, web services, messaging and event driven components. Desktop, batch and ETL processing?

What we are looking for: An awareness of different patterns of enterprise application architecture. Experience with a variety of design solutions.

9. Communication

0	1	2	3	4	5
---	---	---	---	---	---

Suggested questions:

- How long do you usually work on a problem before asking for help?
- How often do you talk to an end user?
- When you are working on a feature, and the requirements are unclear, what do you do?
- How often do you write up problem descriptions, complex business rules, engineering notes (say, in email or a wiki or issue tracking system)?

What we're looking for: Someone who knows when they need help, doesn't hesitate to get it. A preference for face to face communication. An ability to write clear prose for others.

10. Learning

0	1	2	3	4	5
---	---	---	---	---	---

Suggested questions:

- What kinds of reading do you do (online, in books/magazines) about software?
- What's something you ran across recently that was interesting?
- Did you share it with your team?
- Do you do any coding at home for fun.

- Can you give an example?
- Do you contribute to online coding/technical forums?
- Do you take/follow courses on Coursera, iTunes U, etc? If so, what is a course you enjoyed recently?
- **What did you learn yesterday/last week?**

What we're looking for: Someone who genuinely enjoys learning and talking about their craft.

JavaScript Questions

1. Object Oriented JavaScript

0	1	2	3	4	5
---	---	---	---	---	---

Suggested questions:

- What is namespace? Why do we need it?
- How object attributes (property) and methods can be added to custom objects?
- What is prototype in JS?

2. Asynchronous programming

0	1	2	3	4	5
---	---	---	---	---	---

Suggested questions:

- When do we use callback instead of return?
- Can you compare callback to promise/deferred?

3. JavaScript Practice

0	1	2	3	4	5
---	---	---	---	---	---

Suggested questions:

- What is 'use strict'? Why/when do we need it?
- Why do we need to avoid global variables?
- How '==' differs from '==='?
- How do you organize your code? (module pattern, classical inheritance?)

GIS Experience

1. GIS experience

0	1	2	3	4	5
---	---	---	---	---	---

Suggested questions:

- Tell us little about your experience with GIS packages.
- What is your favorite Geoprocessing libraries? And why?
- What is georeferencing?
- What are expected challenges when we are going to map building floor plans?
- What are the differences between label vs annotation? What would you use for room numbers or building names when you map building floor plans? why?

What we're looking for: Knowledge of common GIS concepts.

2. Web mapping

0	1	2	3	4	5
---	---	---	---	---	---

Suggested questions:

- Tell us little about your experience with web mapping .
- What is your favorite web mapping toolkits? And why?
- What are the differences between tiled layer vs vector layer? How would you use them for mapping building floor plans? why?
- What kind of web services would you want to provide to users with detailed building floor plans?

What we're looking for: Someone who have experience in web mapping with visions

Miscellaneous

1. Database experience

0	1	2	3	4	5
---	---	---	---	---	---

Suggested questions:

- What is the basic structure of SQL?
- What is the difference between inner join and outer join?
- What is the difference between WHERE clause and HAVING clause?

2. HTML/CSS experience

0	1	2	3	4	5
---	---	---	---	---	---

Suggested questions:

- What is the difference between class selector and id selector?
- What is the box model?
- What is the difference between absolute positioning and fixed positioning?
- Can you explain the difference between GET and POST?