Talent assessment in software development firms

L Alawneh
Software Engineering Department
Jordan University of Science & Technology, Irbid - Jordan
lmalawneh@just.edu.jo

Abstract. Developing talents in a firm starts by the acquisition of the right candidates. The firm should have a well-defined process to acquire highly talented people. Software development companies constantly look for fresh graduates for strategic and economic reasons. Fresh graduates usually require intensive training to pick up on the pace of the new job requirements. Several software development companies started conducting in-house training for carefully selected candidates who passed a comprehensive placement test followed by a technical interview. At the end of this training, only highly qualified trainees will be hired for full time positions. In this paper, we present a number of methods for talent assessment in software development companies followed by a questionnaire completed by 11 software engineers in a software company in Jordan.

1. Introduction

Though talent acquisition is a big problem to the companies, the development of their talents [1] is at a higher concern to the decision makers. Employees should sustain a competitive level of knowledge and expertise. This opens new horizons to the firm by discovering new markets based on the skill set of its employees. Talent development means that the employees should continue to advance their careers by gaining new skills and knowledge and be up-to-date with the latest technologies in the market. In order to be competitive and capable of handling new challenges, we always need to have the required talents available. The lack of talented resources will have a direct impact on the turnover rates as resources with obsolete skills shall be forced to leave if they cannot keep up with the learning pace [2]. On the other hand, scarce innovative and skilled employees will look for new workplaces especially when their current employer is incapable of sufficing their needs as they look for more challenging opportunities. Therefore, continuous talent development is becoming a necessity for each company in order to stay and grow in the market. The rest of the paper is organized as follows. In Section 2, we present our talent development plan followed by a questionnaire in Section 3. Finally, we conclude our paper in Section 4.

2. Talent Development Plan

A talent development plan should be aligned with the company’s objectives. Therefore, building the right plan should be carefully studied [3]. In the following, we present a number of methods that could
be integrated in software development companies’ processes for developing the talents of their current employees [4]. We present these methods as bullets where each one could be applied separately or in conjunction with others.

- The company should avail all needed sources of information and learning material – Thanks to Google for this, but, we think the company should also have the latest useful material and books available when necessary. This can be achieved by building an online library at the firm. The employees themselves will be part of developing this library by adding the material and books they found useful. Moreover, this library should contain the list of pilot projects, developed by the employees in their learning process, to be set as a preference for other employees.

- The normal culture in the company should be ‘I need to have a mentor’. Communication between mentors and mentees should be available and facilitated. Employees undergoing talent development should be mentored by the senior level employees in order to encourage, motivate, and assess their learning progress.

- At the beginning of each year, the employee should have a one-on-one formal meeting with his manager (or senior as some companies tend not to have a manager title) where they will discuss the weaknesses and areas of improvement (skills required by the company) and the new desired skills by the employee. The mentor should advise the employee where to focus in order to be aligned with the company’s strategies and objectives.

- The company should also conduct a survey periodically (annually, semi-annually or quarterly) where the employees will be asked about their preferences on the skills they would like to gain and the courses they are interested in. The outcome of the survey should not be applied immediately but should be used for future planning (following year). People tend to forget what they were thinking of if it was not documented. Therefore, we think such a survey should be very useful in recording all the potential ideas, needed skills, etc.

- The learning process could be improved by having employees coach each other. Senior technical resources can always guide in the process of talent development and can provide the right feedback on each employee’s technical abilities.

- The best way a person can gain new skills is by ‘getting his hands dirty’. Therefore, asking the employees to come up with new ideas where they can apply and implement the new knowledge is an excellent strategy to make sure that they are ready to drive when new projects rise in the horizon.

- Another way is by applying Stretch Tasks. These kinds of tasks will force the employees to acquire the knowledge they are lacking much faster. Stretch tasks could be part of an ongoing project. However, this should be handled carefully with the severity of the project.

- Employees with best ideas translated into tangible pilot projects will be asked to lead their projects and build up their own teams. This will encourage the employees to innovate and excel as it will reflect directly on their career paths and will make them more engaged and loyal to the company.

- Knowledge sharing. The culture of sharing the knowledge should be the norm at the company. Employees who just delivered a major milestone should share their experience with the rest of the team members and the other teams in the company (daily stand-up meeting may be insufficient). In every project there is something new to be learned. Spreading the new
knowledge among the other teams is substantial in building the talent across the company. Moreover, knowledge gained from reading a new book, training course or any other experience should be presented to the interested teams in the company.

- Cross-team collaboration is also important in developing the talents at the company.
- Competition. Setting up a monthly/quarterly challenge where the best pilot project will be awarded. This will motivate the employees to innovate.
- Learning new skills shall be integrated in the annual review process. Employees who gain more knowledge and translate this knowledge in their work should be rewarded.

Finally, losing some of the company’s most talented resources is inevitable. However, by applying certain measures, the company can reduce this number by providing the right environment to its employees. Talent retention through talent development is currently the trending approach in major software firms and proved its validity. Outstanding employees always look for competitive remuneration packages and a very challenging and highly skilled environment. Thus, companies should take talent development more seriously as it will have its benefits on all the stakeholders.

3. Employee Questionnaire Results

The objective of this survey is to measure the level of knowledge and expertise in the team and to identify opportunities for skills development. The survey was answered by 11 developers (Table 1).

Table 1. Target Employees

| Employee Category                  | Number of Employees |
|------------------------------------|--------------------|
| Junior (0-2 years)                 | 8                  |
| Intermediate (2-5 years)           | 1                  |
| Senior (5-8 years)                 | 0                  |
| Principal (more than 8 years)      | 2                  |

The topics included in the survey are core java skills, requirements engineering skills, configuration management and quality assurance skills. Developers were categorized based on their level of experience in four categories.

Table 2. Java Related Responses (average per developer group)

| Question                                                                 | Junior | Intermediate | Principal |
|--------------------------------------------------------------------------|--------|--------------|-----------|
| Do you apply 'Exception Handling' in your code properly?                 | 3.375  | 4            | 3.5       |
| Rate your understanding of 'Generics'                                   | 3.375  | 4            | 3.5       |
| Rate your skills in writing multithreaded programs                       | 2.75   | 4            | 3         |
| Rate your skills in using Reflection in Java?                           | 2.875  | 3            | 2.5       |
| Rate your skills in using Dynamic Class Loading features?               | 2.5    | 3            | 3         |
| Are you familiar with Java's Dynamic Language Support?                  | 1.875  | 1            | 2.5       |
| Are you familiar with using Lambda Expressions in Java?                 | 2.375  | 4            | 2.5       |
| Rate your skills in using Annotations?                                  | 2.875  | 4            | 3         |
| Are you familiar with Java Native Interfaces?                           | 2.125  | 2            | 2         |
| Rate your understanding of session management                           | 2.625  | 3            | 3         |
| **Average**                                                              | 2.73   | 3.4          | 3         |
Table 2 provides a summary of the responses for the Java related questions. Each value represents the average for each answer based on each developer category (Junior, Intermediate, and Principal). The last row provides the averages of all the responses for each category. The results for junior engineers seem to be normal and aligned with their level of expertise. However, for advanced Java topics, the intermediate and principal engineers show similar knowledge compared to the junior engineers.

When asked about which requirements (functional and non-functional) they find more difficult to develop, the responses were as follows:

1. Two (2) responses were: Neither of them.
2. Four (4) responses were: Non-Functional Requirements.
3. Five (5) responses were: Depends on the task’s size.

Being part of a highly dynamic and agile environment means that requirements may change at any time. Thus, the developers need to be familiar with the best practices for requirements gathering, analysis, and specifications. One important aspect is requirements traceability. The usage of Atlassian Suite at the company facilitates the traceability of requirements especially with the availability of tools such as Jira, Confluence and BitBucket.

Table 3 provides a summary of the responses related to requirements engineering. The responses indicate that most of the developers in the team need to improve on this important aspect. Requirements could come from business or from technical teams. In both ways, the engineer needs to be well acquainted with Requirements Engineering skills.

| Question                                                                 | Junior | Intermediate | Principal |
|--------------------------------------------------------------------------|--------|--------------|-----------|
| Are you able to elicit business/technical requirements properly?          | 2.875  | 3            | 3.5       |
| Are you able to eliminate conflicts from elicited requirements?          | 2.875  | 3            | 3         |
| Are you able to prioritize requirements correctly?                       | 2.875  | 3            | 3         |
| Do you prefer to provide a visual description of the specifications?     | 3.375  | 3            | 3.5       |
| Are you able to specify the requirements in a unified and consistent manner? | 2.875  | 3            | 3         |
| Do your colleagues understand your specifications?                       | 3      | 3            | 3         |
| Are you following the convention at PS in writing the specifications?    | 3.25   | 3            | 3         |
| Are you always capable of writing your test cases right after the requirements specifications are complete? | 3.125  | 4            | 3.5       |
| **Average**                                                              | **3.03** | **3.125**    | **3.19**  |

When asked about which requirements the developers find more difficult to elicit, the responses were as follows:

1. Two (2) responses were: Depends on the stakeholders.
2. Three (3) responses were: Non-Functional Requirements.
3. Six (6) responses were: Depends on the requirements complexity.

Table 4 provides a summary of the responses regarding Configuration Management, DevOps and Software Quality Assurance. The engineers seem to have good knowledge in this area. However, the junior engineers still need to improve on some aspects.
Table 4. CM, DevOps and Quality Assurance Responses (average per developer group)

| Question                                                                 | Junior | Intermediate | Principal |
|--------------------------------------------------------------------------|--------|--------------|-----------|
| Rate your acquaintance level with Continuous Delivery?                   | 2.75   | 3            | 3         |
| Do you apply best practices when using Version Control Systems?           | 3      | 3            | 3         |
| Your checked-in fixes work from the first time                           | 3.5    | 5            | 3         |
| Your test cases cover all scenarios                                       | 3.25   | 4            | 3         |
| Rate your level of comfort when applying Continuous Delivery              | 3      | 5            | 3         |
| Rate your level of satisfaction with applying Continuous Delivery         | 2.75   | 3            | 3.5       |
| Rate your level of experience with Bamboo                                | 2.875  | 3            | 4         |
| Do you always provide a proper description of your changes in Jira?       | 3.625  | 2            | 4         |
| Are you comfortable in taking the role of the scrum master?              | 3.125  | 3            | 3         |

Average: 3.1 3.44 3.28

Table 5 provides a summary of the averages per section per developer. Averages below 3 were highlighted to show that these developers need to improve on the required skill.

Table 5. Average for each section per developer (J: Junior, I: Intermediate, P: Principal)

| Question                                                                 | J | J | J | J | J | J | J | J | P | P |
|--------------------------------------------------------------------------|---|---|---|---|---|---|---|---|---|---|
| How many years have you been at the Company?                             | 1 | 1 | 1 | 2 | 2 | 2 | 1 | 2 | >2 | >2 |
| 1 year, 2 years, more than 2 years (> 2)                                 |   |   |   |   |   |   |   |   |   |   |
| How do you rate yourself in Java?                                        | I | A | I | B | I | B | I | A | A | A |
| B-Beginner, I-Intermediate, A-Advanced                                   |   |   |   |   |   |   |   |   |   |   |
| Rate your acquaintance level with Object Oriented Principles?            | I | I | E | I | I | E | E | E | I |   |
| B-Beginner, I-Intermediate, E-Expert                                     |   |   |   |   |   |   |   |   |   |   |
| Core Java Skills (average)                                               | 2 | 3.5| 3.2| 2.5| 2.8| 2.2| 2.6| 3.1| 3.4| 3.8|
| 1 – Weak, 5 – Strong                                                     |   |   |   |   |   |   |   |   |   |   |
| Requirements Engineering (average)                                       | 3.5| 3.1| 2.1| 2.7| 3  | 3.6| 3  | 3.1| 3.1| 3.3|
| 1 – Weak, 5 – Strong                                                     |   |   |   |   |   |   |   |   |   |   |
| Configuration Management and Quality Assurance (average)                 | 3.3| 2.2| 3.1| 2.8| 3.1| 3.8| 3.2| 3.2| 3.4| 3.3| 3.2|
| 1 – Weak, 5 – Strong                                                     |   |   |   |   |   |   |   |   |   |   |   |

Table 6 summarizes the list of references the developers use to enrich their skills. As we can see, the developers depend mainly on well-known books and available blogs.

Table 6. References used by the team members

| Title                                                                 | Type       |
|-----------------------------------------------------------------------|------------|
| Design Patterns: Elements of Reusable Object-Oriented Software         | Book       |
| Object-oriented and classical software engineering                     | Book       |
| Java in a Nutshell                                                    | Book       |
| Competitive Programming 3rd Edition                                    | Book       |
| OAuth 2 in Action                                                     | Book       |
| Clean Code                                                            | Book       |
| Cucumber Cookbook                                                     | Book       |
| Java Testing with Spock                                               | Book       |
| Head First Java                                                       | Book       |
| Effective java                                                        | Book       |
| Functional programming, reactive programming, design principles       | Blogs      |
4. Conclusion

In this paper we proposed different methods that can be used by the talent development teams in software development companies. Additionally, we presented the results of a survey conducted by 11 developers in a well-known software development company in Jordan. The survey shows that in most cases, the knowledge of the developers is mainly related to the number of years of experience and the number of years spent at the company. In the future, we intend to present an assessment of the developers’ skills after applying the proposed talent development methods and their impact on their productivity.

References

[1] A. H. Church, C. T. Rotolo, N. M. Ginther, R. Levine, “How are top companies designing and managing their high-potential programs? A follow-up talent management benchmark study,” Consulting Psychology Journal: Practice and Research, Vol. 67(1), Mar 2015, 17-47

[2] E. Mone, M. London, “Employee Engagement Through Effective Performance Management,” New York: Routledge, 2018.

[3] J. A. Gruman, A. M. Saks, “Performance management and employee engagement,” Human Resource Management Review, 2012, Vol. 21(2), 123-136.

[4] A. H. Mehdiabadi, J. Li, “Understanding talent development and implications for human resource Development: An integrative literature review. Human Resource Development Review,” 2016, Vol. 15(3), 263–294.