



# **STRUGGLES OF NEW COLLEGE GRADUATES IN THEIR FIRST SOFTWARE DEVELOPMENT JOB**

**Presented By:**  
Gurpinder kaur  
Student ID- 7960628

# INDEX

- Introduction
- Research Questions
- Methodology
- Results
  1. Abilities
  2. Difficulties
  3. Misconceptions which Hinder
- Implications for Educators
- Conclusion



# INTRODUCTION

- Preparing CS graduates for eventual roles in soft. Dev. is a goal for many CS programs.
  - However, employers recognize that fresh graduates do not meet the industry demands.
- Significant literature has documented the costs of bringing software developers up to speed on a project or a new team.eg. Eric Brechner study
- In this, study discovers what occurs during the beginning of transition period from college graduate to experienced software developer.



# RESEARCH QUESTIONS

- RQ1: How do new college graduates experience their first software development jobs?
- RQ2: In what ways are they prepared by their educational experience?
- RQ3: In what ways do they struggle in order to be productive in their new positions?



# METHODOLOGY

- Ethnographic study
- Studied recent graduates hired by Microsoft b/w 1 and 7 months.
- Total=8 (7 men, 1 women)
- 2 were educated in US, 2 China, 1 Mexico, Pakistan, Kuwait and Aus.
- W,X,Y,Z had BS degrees in CS
- V had MS
- U, R, T had PhDs



# METHODOLOGY

- Observed the struggles of NSDs using fly on wall observations for 85 hours plus pre study and post study interviews for 2 months.
- Subjects were compensated \$50 weekly.
- Each subject was observed 8-13 hrs over 2 weeks with a break of month in b/w.
- Observation was conducted in silence with the observer sitting behind the participant watching screen.



# RESULTS

## ○ ABILITIES

1. Coding
2. Reading and Writing Specifications
3. Persisting/Generating Hypothesis



# ABILITIES-CODING

- NSDs demonstrated many programming strengths.
- Capable of dealing with complex issues, using critical coding tools.
- Evidenced excellent debugging strategies , debated various test cases.
- Using online documentation to explore and utilize APIs.





# ABILITIES-READING & WRITING SPECIFICATIONS

- Showed excellent ability in reading the documents, engaging in discussion with the lead which further lead to the clarification of design and in outlining specific use cases .
- NSDs seemed confident in developing a structured, lengthy feature planning document.



# PERSISTING/GENERATING HYPOTHESIS

- Persistence was commonly observed of NSDs.
  1. **Eg.** In dealing with new and large databases
  2. In struggling to utilize new tools.
  3. In seeking to understand institutional norms.
  
- Specific interest in SD experience is in generating hypothesis for unexpected behavior.



# RESULTS

## ○ DIFFICULTIES

1. Communication
2. Collaboration
3. Technical
4. Cognition
5. Orientation



# COMMUNICATION

- NSDs do not ask questions soon enough and often struggle to ask questions at an appropriate level.
- Everyone was very careful in crafting work related emails.
- English skills were a problem for some non-native speakers.



# COLLABORATION

- Several NSDs recognized that their team interaction skills were something they needed to focus on.
  1. Eg. Subject X was finding time to prepare for team meetings and critically read prior for discussions.
- But with time all NSDs were prepared to describe and detail their work efficiently in meeting and even learn to negotiate with colleagues about task mgmt.



# TECHNICAL

- NSDs find it difficult to work on tools that support large scale development. Eg. seem to flounder with revision control system.
- Testing robustly was also an issue.
- The technical difficulties often coupled with collaboration and orientation issues.



# COGNITION

- NSDs struggled to collect, organize and document the wide range of information that they needed to absorb.
- They struggle to know ‘when they don’t know something’.
- Sometimes the info Session would occur in the middle of general request for inf. and thus not organized and stated terms with which NSDs are not familiar.
- **Example:**
- Even after asking for help on some code and getting a very specific ans., Subject V continued attempting to reason thru it.



# ORIENTATION

- Some NSDs were woefully isolated from their teams.
- Sometimes not even knowing all the members of their team.
- Thus rarely knowing who to talk about certain issues.
- This impacted both NSD productivity and frustration greatly.
- Similarly subject V requested a mentor from his mgr after 4 weeks only to find the mentor was very busy.





## MISCONCEPTIONS WHICH HINDER

- *I must do everything myself so that I look good to my manager.*
- *I must be the one to fix any bug I see- and I should fix it “right” way, even if I don’t have any time for it.*
- *If there was only more documentation.*
- *I know when I am stuck when solving a problem.*



## IMPLICATIONS FOR EDUCATORS

- Instead of a SE Greenfield project, a more valuable experience could be to fix bugs from the large pre-existing codebase and write additional features.
- Adding Management component.
  - Where mentors can be the students who have previously taken the class.
  - And PM can be the TA.

The issue of teaching students techniques to recognize when they are stuck.



## CONCLUSION

- This paper reports on the most in-depth studies of new developer exp.
- Findings show that many of the problems are not due to the lack of experience in programming, design or debugging but their communication, collaboration and orientation skills are not as well addressed.
- Suggestions for curricular reform are a preface for renewed dialogue b/w the needs of industry and goals of CS education.



QUESTIONS??

