Case study of the Collections Application: Identification of flaws in development lifecycle including Requirements that Led to Flawed Architecture and Development

Presented By: Faisal Iqbal

Advanced Topics in Requirements Engineering (CS 846)

Spring 2019

UNIVERSITY OF WATERLOO

UNIVERSITY OF WATERLOO

Outline

- Introduction
- Overview of the Application
- Critical Events including Requirements gathering
- Financial Loss
- Recommendations
- Conclusion

Introduction

- This project is about a collection department's application
- Awarded to a team with lowest cost bid
- Software team with limited senior developers
- Bad requirements gathering process
- Bad development
- Increased cost and possible failure

Application Overview

- Required by Collections department of an institution
- Supposed to wrap around multiple software
  - PSYS – for credit accounts
  - NAS – recovery management system
  - NICO – big data system
- Supposed to present client data on one screen for the collectors

Application Overview – cont.

- Application performs multiple tasks:
  - Take promise to make payment
  - Payment schedule
  - Display record
  - Accounts maintenance
  - Calculators to perform maintenance
  - Reporting

Critical Events including Requirements gathering

- Requirements gathering/RE process
  - No BSA assigned to the RE
  - No project manager assigned
  - Multiple people gathering requirements from stakeholders including agents, managers, higher management
    - No collaboration was done between them
  - They tried to put functional and non functional requirements in words and put stress that those be fulfilled as described

Critical Events including Requirements gathering – Cont.

- "What" and "how" of requirements were stressed as described by stakeholders without consulting any consultant if those are possible or not
- Incomplete scenarios
- Missed links between chain processes assuming the process workflow in a specific way

Critical Events including Requirements gathering – Cont.

- Project award bid
    - Software Development teams participated
    - Awarded to a team with lowest bid
    - Did not assess the background and successful projects of the team
- The team
    - Pretty new
    - Still following waterfall approach
    - Few senior developers and those were assigned their own projects
    - New hire – junior developer assigned the project

Critical Events including Requirements gathering – Cont.

- Software architecture:
  - Never planned software architecture and technology stack
  - Software scalability was not considered
  - Database structure was not considered
- Change requests:
  - Due to ambiguous words, changes were requested very often
  - RE was not performed well so the missing requirements were added later

Critical Events including Requirements gathering – Cont.

- Changes that happened later broke the application
  - Requirement of building the UI in the strict "what" and "how" manner, led to broken application when changes were requested
  - Some screens were completely re-written

Financial burden

- 1 developer and 2 DBAs were assigned fulltime and many employees from collections were assigned partially to the project at the start
- A contractor was hired later and assigned to this project.
- Initial estimation of work was 12 months but it has been 2.5 years the project is still in development

Financial Burden – cont.

- Later realised the burden of the on going project
  - Assigned a fulltime project manager to over look the project
  - Contractor is being extended

Recommendations

- Need to stop the development to stop financial loss
- Revise the requirements and assign BSA for proper RE
- Submit the application for technical audit to identify potential security issues
- Senior DBA should analyse the database structure
- Assign the Agile development team to the project

Conclusion

- RE is very important for:
  - Project completion time estimation
  - cost estimation
  - Less change in requirements
- Technical skills and background of the team is very important
- Project management techniques are very important
  - Project manager, BSA, Kanban board, GitHub
- Software development techniques are very important
  - Agile vs waterfall

UNIVERSITY OF WATERLOO

THANK YOU

QUESTIONS