

How Scrum Solves Project Problems

This is **not** essential reading for my course. However, if you do like reading, and you want to take a break from videos then you might find the below interesting and useful.

Contents

| | |
|--|----------|
| Part 1: An all too common project disaster | 2 |
| How the Scrum Framework might have helped. | 3 |
| Iterative Development: | 3 |
| Flexibility to Changing Requirements: | 3 |
| Regular Inspection and Adaptation: | 3 |
| Cross-functional Teams: | 3 |
| Continuous Communication: | 4 |
| Time-Boxed Development: | 4 |
| Part 2: Real world examples of similar projects that failed and how Scrum might have helped | 4 |
| HealthCare.gov (2013): | 4 |
| Scrum Solution: Iterative Development | 4 |
| Denver International Airport Baggage System (1994): | 5 |
| Scrum Solution: Cross-functional Teams | 5 |
| NHS National Programme for IT (United Kingdom, 2002-2011): | 5 |
| Scrum Solution: Time-Boxed Development (Sprints) | 5 |
| Airbus A380 (mid-2000s): | 6 |
| Scrum Solution: Regular Inspection and Adaptation (Sprint Review) | 6 |
| Google Glass (2013-2015): | 6 |
| Scrum Solution: Flexibility to Changing Requirements | 6 |

Part 1: An all too common project disaster

Here is a fictional story about a project at a fictional company. Think to yourself if this story sounds familiar...



In the bustling city of Techopolis, there was a renowned software development company named ByteCraft Solutions. With a stellar reputation for delivering cutting-edge solutions, they were commissioned to undertake a groundbreaking project by a large multinational

corporation, Global Innovations Inc. The project, codenamed "Project QuantumLeap," aimed to develop a revolutionary enterprise resource planning (ERP) software that would streamline operations and revolutionize the way businesses managed their resources.

Excitement filled the air as ByteCraft assembled a team of brilliant engineers, designers, and project managers. The vision was grand, and expectations were high. The initial project plan outlined an ambitious timeline and budget, promising delivery of a feature-rich and highly scalable ERP software within 18 months.

As the project kicked off, the team encountered unforeseen challenges. The requirements from Global Innovations Inc. were constantly evolving, with new features and functionalities being added regularly. The changes created a ripple effect, causing delays and disruptions in the carefully crafted timeline.

The project managers at ByteCraft struggled to keep up with the changing scope and maintain the delicate balance between delivering a high-quality product and meeting the deadlines. The development team found themselves working long hours, and the pressure began to mount.

Meanwhile, the budget that seemed more than adequate at the project's outset started to dwindle. Unanticipated technical challenges, coupled with the constant scope changes, led to increased development costs. ByteCraft tried to renegotiate the terms of the contract with Global Innovations Inc., but the client was unwilling to compromise on the initially agreed-upon features and timeline.

As the months passed, the project's status meetings became tense. ByteCraft was falling behind schedule, and the once-innovative features promised by Project QuantumLeap were either scaled down or abandoned altogether. The development team was demoralized, and turnover increased as talented engineers sought more stable and less stressful work environments.

Eventually, the project reached its deadline, but the ERP software delivered by ByteCraft fell short of expectations. Critical features were missing, and the software was plagued with bugs and performance issues. Global Innovations Inc. was furious, having invested a significant amount of time and money into a product that didn't meet their needs.

The aftermath was devastating for both parties. ByteCraft faced legal repercussions, and their reputation took a severe hit. Global Innovations Inc. had to go back to the drawing board, investing more resources to salvage what they could from the failed project.

In the end, Project QuantumLeap became a cautionary tale in the tech industry, a reminder of the importance of realistic project planning, effective communication, and managing scope changes. The once-promising partnership between ByteCraft Solutions and Global Innovations Inc. ended in disappointment, leaving behind a trail of lessons for future endeavors.

How the Scrum Framework might have helped.

The Scrum framework, with its iterative and adaptive approach, could have provided several benefits to the Project QuantumLeap situation, potentially mitigating some of the challenges faced by ByteCraft Solutions. Here are ways in which Scrum might have helped:

Iterative Development:

Scrum's iterative development cycles, known as sprints, could have allowed ByteCraft to deliver incremental versions of the software regularly. This would have enabled Global Innovations Inc. to see tangible progress and provide feedback throughout the development process. Early and continuous delivery of valuable software could have helped in managing changing requirements more effectively.

Flexibility to Changing Requirements:

Scrum embraces changes in requirements, with a commitment to delivering the highest priority features first. This would have allowed ByteCraft to adapt to the evolving needs of Global Innovations Inc. The Product Owner, representing the client, would be empowered to reprioritize the backlog at the end of each sprint.

Regular Inspection and Adaptation:

Scrum's regular inspect-and-adapt approach through events like Sprint Review and Sprint Retrospective could have provided a platform for ByteCraft and Global Innovations Inc. to review progress and address issues regularly. The team could have identified challenges early on and adjusted their strategy to prevent the accumulation of problems over time.

Cross-functional Teams:

Scrum encourages the formation of cross-functional teams, where members possess diverse skills. This could have fostered better collaboration and communication within the team, reducing dependencies and bottlenecks. Cross-functional teams are better equipped to handle changes in requirements and address a variety of challenges that may arise during development.

Continuous Communication:

Scrum promotes transparency and open communication. Daily Stand-up meetings would have allowed team members to discuss their progress, impediments, and plans. This increased transparency could have facilitated early identification of issues and better collaboration among team members.

Time-Boxed Development:

Scrum enforces time-boxed development cycles, ensuring that each sprint has a fixed duration. This time-boxing could have prevented an open-ended development timeline and created a sense of urgency to deliver functionality within a specified timeframe, making it easier to manage client expectations.

While Scrum could have helped address some challenges, it's essential to note that the success of any framework depends on its proper implementation and the commitment of the entire team. Scrum is not a silver bullet, but when applied correctly, it can enhance collaboration, flexibility, and the overall success of software development projects.

Part 2: Real world examples of similar projects that failed and how Scrum might have helped

While specific details about failed projects are often kept confidential, there have been instances where high-profile projects experienced significant challenges, including budget overruns, delays, and end products that didn't fully meet expectations. Here are a few examples:



HealthCare.gov (2013):

The launch of the U.S. federal health insurance exchange website, HealthCare.gov, in 2013 is a well-known example. The project faced numerous issues, including technical glitches, inadequate testing, and mismanagement. The initial rollout was marred by crashes, slow response times, and functionality issues, leading to a loss of public trust. The project eventually required extensive rework to address the issues.

Scrum Solution: Iterative Development

Scrum's iterative development approach, with regular releases and feedback loops, could have allowed for incremental improvements and early detection of technical issues. By delivering a functional product incrementally, the development team could have identified and addressed problems sooner, reducing the impact on the overall project.

Denver International Airport Baggage System (1994):

The baggage handling system at Denver International Airport was intended to be an automated marvel, but it faced severe technical and logistical challenges. The system experienced frequent malfunctions, causing delays and lost luggage. The project ended up significantly over budget, and after numerous attempts to fix the issues, portions of the automated system were eventually abandoned.



Scrum Solution: Cross-functional Teams

The use of cross-functional teams in Scrum, where members have diverse skills, might have facilitated better collaboration and communication. In a complex project like the baggage system, having a cross-functional team with expertise in logistics, software development, and automation could have improved problem-solving and decision-making.

NHS National Programme for IT (United Kingdom, 2002-2011):

A screenshot of a newspaper article from The Times. The headline reads "Exclusive NHS computer fiasco". Below the headline, there are two bullet points: "Patient database was axed in autumn but runs up £2bn bill" and "Risk-takers will foot the bill for a further £2 billion on a failed NHS IT project even though the Government has already pulled the plug on it." The article includes several small images, including a keyboard and a computer screen, and is dated Thursday December 8 2011.

The National Programme for IT in the United Kingdom aimed to modernize and integrate information technology systems across the National Health Service (NHS). The project faced challenges such as changing requirements, scope creep, and difficulties in achieving interoperability. It was eventually scrapped in 2011 after significant delays and financial overruns.

Scrum Solution: Time-Boxed Development (Sprints)

Scrum's time-boxed development cycles, or sprints, could have helped in managing the project timeline more effectively. Setting specific durations for development phases and reassessing priorities at the end of each sprint could have provided greater visibility into the project's progress and allowed for quicker adjustments to changing requirements.

Airbus A380 (mid-2000s):

The development of the Airbus A380, the world's largest passenger aircraft, faced considerable challenges. Delays in production and technical issues led to significant cost overruns. Airbus had to revise its delivery schedules multiple times, causing financial strain and damage to its reputation. While the A380 eventually became operational, the project faced substantial difficulties during development.



Scrum Solution: Regular Inspection and Adaptation (Sprint Review)

Scrum's Sprint Review, where stakeholders assess the product at the end of each sprint, might have facilitated continuous inspection and adaptation. Regularly reviewing the status of the project could have allowed Airbus to identify and address challenges early, preventing the accumulation of issues that led to significant delays.

Google Glass (2013-2015):

Google Glass, a wearable augmented reality device, faced challenges during its development and rollout. The product was met with concerns about privacy, usability issues, and a high price point. Google decided to halt consumer sales in 2015 and refocus on enterprise applications. The project demonstrated the importance of understanding user needs and market dynamics in the development of new technologies.

Scrum Solution: Flexibility to Changing Requirements

Scrum's commitment to adapting to changing requirements could have been beneficial for a project like Google Glass. Embracing changes in user feedback and market dynamics through regular refinement of the product backlog might have enabled Google to pivot more effectively and address concerns related to privacy, usability, and pricing.



These examples highlight the complexities and challenges that can arise in various industries, from healthcare and aviation to information technology and consumer electronics. It's important to note that project failures can result from a combination of factors, including technical issues, changing requirements, mismanagement, and external pressures. Learning from these examples can help organizations improve project management practices and increase the likelihood of successful outcomes in the future. While Scrum is not a one-size-fits-all solution, these solutions illustrate how its principles and practices can be applied to address specific challenges encountered in different projects.

It is worth noting that these real world project failures suffered from common problems. The main ones being inability to cope with changing requirements, for which a well managed product Backlog and regular inspection and adaptation in Sprints would have really helped. The other main problem being issues dealing with the complex nature of developing something new, for which cross-functional teams and regular communication at Daily Scrums and inspection and adaption opportunities would have helped.

I hope you enjoyed these stories. Now back to the course 😊