



**Linnéuniversitetet**

Kalmar Växjö

# 2DV608 Assignment 1

## Software Design



*Author:* Rashed\_Qazizada  
*Supervisor:* Mauro Caporuscio  
*Semester:* Autumn 2020  
*Course code:* 2DV608



## Abstract

Place you abstract here

This page was intentionally left unchanged 😊

## Keywords

Place your keywords here

# Table of Contents

<b>1 Task1</b>	<b>3</b>
1.1 No Silver Bullet-Essence and Accident in Software Engineering .....	3
1.2 How could a software project be such a monster? .....	3
1.3 Are these approaches successful? .....	3
1.4 The central question of how to improve the software and art centres? .....	3
<b>2 Task</b>	<b>4</b>
2.1 The Design Question .....	4
2.2 My experience and my mitigation.....	5
<b>References</b>	<b>6</b>

# 1 Task1

## 1.1 No Silver Bullet-Essence and Accident in Software Engineering

Within this paper the author asserts that there is no single development in either technology or management technique, which by itself promises even one order-of-magnitude improvement with a decade in productivity, reliability, simplicity.

Furthermore, in the paper the author introduces the metaphor of a software project as werewolf a person who might otherwise appear perfectly human, can transform into a monster of horror stories and nightmares.

Last, the author purpose that each software organization must determine and proclaim that great designers are as important as a successful manager [1].

### 1.2 How could a software project be such a monster?

The author believes that a monster would be a project that has mis schedules, blown budgets and flawed products. The software industry is traditionally approaching large development projects as if it was building a bridge or a skyscraper. The idea of software design has established best practice in how to analyse the requirements, establish a design, proceed to build an item and then to test it and verify that what they have built is what was desired [1].

### 1.3 Are these approaches successful?

Yes, the author suggested ideas such as maybe writing new computer languages makes us more productive possibly buying third party software or components to so that we do not have to write everything from scratch and we can use stuff that is been tested by other software engineers [1].

### 1.4 The central question of how to improve the software and art centres?

The author purposes that each software organization must determine and proclaim that great designers are as important to its success as great managers are, and that they can be expected to be similarly nurtured and rewarded.

To grow great designers following steps are obvious:

- Systematically identify top designers as early as possible.
- The best is often not the most experienced.
- Assign a career mentor to be responsible for the development of the prospect and keep a careful career file.
- Devise and maintain a career development plan for each prospect, including carefully selected apprenticeships with top designers, episodes of advanced formal education, and short courses, all interspersed with solo design and technical leadership assignments.
- Provide opportunities for growing designers to interact with and stimulate each other [1].

## 2 Task

### 2.1 The Design Question

At First worth mentioning here that effective design is the main core from software development to engineering, to lead towards effective, elegant designs, the challenge is to find a method to work that allows design to be iterative and should avoid new approaches and new solution.

I read material from the first three chapters of The Design of Design.

The fundamental question about design:

- What is Design?
- The Design Concept?
- Kinds of Design?

The Design Question begins with interesting quotes:

1. Francis Bacon in 1605 said that we can learn a lot by taking lessons learned from single discipline and looking to see if we can use it in another.
2. Herbert Simon The Sciences of the Artificial in 1969 suggested we might not be able to talk about the specifics of our fields, However, we can share the experiences of the creative professional design and problem solving.

Diverse designers will have various approaches we can learn a lot about design just by sharing experiences. Furthermore, Brooks briefly mentions a few types of design:

- System Design
- Artistic Design
- Routine Design
- Adaptive Design
- Original Design

The author starts chapter 2 with quoting Herb Simon and the basic idea is that, design can be learnt as a systematic process.

- Goals
- Utility Function
- Constraints
- Resources

Moreover, in chapter 16 the author said that “We can get good designs by following good practices instead of poor ones. Good design practices can be taught” [2].

## 2.2 My experience and my mitigation

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna. Nunc viverra imperdiet enim. Fusce est.

Vivamus a tellus. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Proin pharetra nonummy pede. Mauris et orci. Aenean nec lorem.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna. Nunc viverra imperdiet enim. Fusce est.

Vivamus a tellus. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Proin pharetra nonummy pede. Mauris et orci. Aenean nec lorem.

## References

- [1] F.P. Brooks, No Silver Bullet-Essence and Accident in Software Engineering,ch-16
- [2] F.P. Brooks, The Design Question, ch-1