Contents

[Prototyping 2](#_Toc249159047)

[Abstract 2](#_Toc249159048)

[Prototyping Introduction 2](#_Toc249159049)

[Definition 2](#_Toc249159050)

[A software development approach: 4](#_Toc249159051)

[An IS project management approach: 5](#_Toc249159052)

[Variety of Prototypes 8](#_Toc249159053)

[Purpose of prototyping 10](#_Toc249159054)

[Clarify the problem 10](#_Toc249159055)

[Review the technical aspects 11](#_Toc249159056)

[Adjusting and tailoring 11](#_Toc249159057)

[Prototyping Advantages 12](#_Toc249159058)

[Cost Saving 12](#_Toc249159059)

[Find a better solution 12](#_Toc249159060)

[Problems with Prototyping 13](#_Toc249159061)

[Operation fails 13](#_Toc249159062)

[Low success rate in big sized prototyping 13](#_Toc249159063)

[“Unrealistic expectation” [iii] 14](#_Toc249159064)

[Implementation of prototyping results 15](#_Toc249159065)

[New technologies 19](#_Toc249159066)

[Conclusions 19](#_Toc249159067)

[References 19](#_Toc249159068)

# Prototyping

## Abstract

This research paper addresses the role of prototyping and importance of correct implementation of prototyping process and its results. And also will cover some new fashions and tools in prototyping in different fields. Further it will focus on strategies for implementing the prototyping results in corresponding areas. Overall this paper tries to define and portrait the strategies that an analyst can apply to better practice prototyping and implementing the outcomes.

[Go to TOP](#_top)

## Prototyping Introduction

### Definition

In system analysis and design related texts the term prototyping has been used with different definitions[[1]](#endnote-1), but they all have shared the concepts of parties involved (one side called designer, analyst or developer and the other side client or user), and concept of tangible (virtually or physically). Also in different texts the authors has emphasized on big consequences of not correctly implementing the prototyping on final solution or product.

[Click here](http://www.ted.com/talks/lang/eng/tim_brown_on_creativity_and_play.html) to watch a video from IDEO to see a creative look to prototyping

[ted_logo.gif](http://www.ted.com/)

[](http://www.ideo.com/)

[Go to TOP](#_top)

### A software development approach:

“Prototyping is an approach based on an evolutionary view of software development, affecting the development process as a whole. Prototyping involves producing early working versions (“prototypes”) of the future application system and experimenting with them. Prototyping provides a communication basis for discussions among all groups involved in the development process, especially between users and developers. In addition, prototyping enables us to adopt an approach to software construction based on experiment and experience.”[[2]](#endnote-2)

Link: [http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=368126&isnumber=8425](http://ieeexplore.ieee.org.ezproxy.umsl.edu/stamp/stamp.jsp?arnumber=368126&isnumber=8425)

[Go to TOP](#_top)

### An IS project management approach:

“There are at least four different kinds of prototypes;

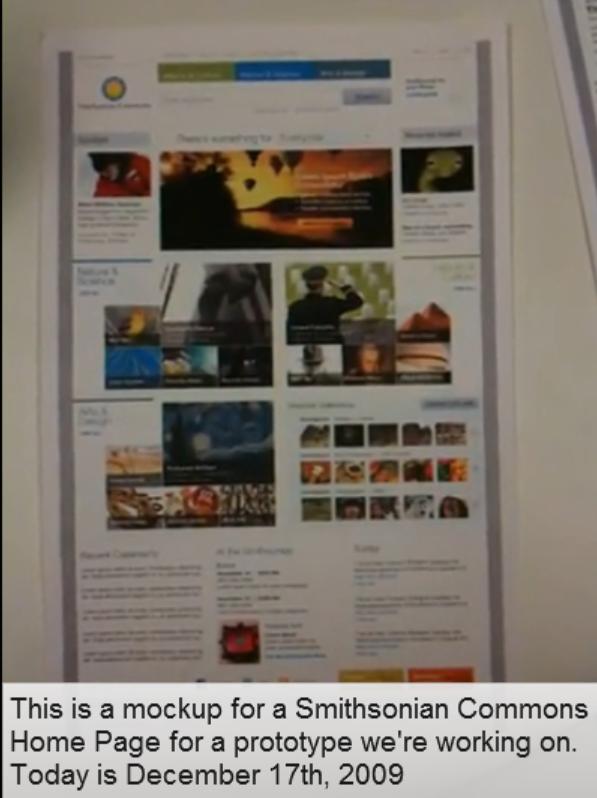
A **mock-up prototype** models physical aspects of the final system. Mock-ups are non-executing versions of systems built with cardboard boxes, wood or plastic, perhaps even with people simulating the computer operations [12, 16].

See how **the Smithsonian** web site Prototype is shared for feedback in this stage;

[](http://www.si.edu/)

[Click](http://www.si.edu/) to see the current Smithsonian website

[Click](http://www.youtube.com/watch?v=mtzJaOSS44g) to watch a video on how the developer in contract exposes the prototype in this stage for feedback

[](http://www.youtube.com/watch?v=mtzJaOSS44g)

[Go to TOP](#_top)

A **throw-away** **prototype** is perhaps the most common category of prototype. It actualizes requirements as specified as an executable system. It is often described as an operational or running requirements specification. Throw-away prototypes include user interface prototypes that have limited functionality but embody the human interfaces.

A **quick and dirty prototype** is an early implementation without prior analysis and design. By revising it until the users are satisfied, it can evolve into the final system.

A **design-driven prototype** used for technical experiments, e.g. with a platform or communication technology. Where design-driven prototypes are for a pre-finalization “test-drive”, they may implement a design that is very close to the complete final system. However, for efficiency purposes, the final system is developed traditionally. Finally an evolutionary prototype is a modifiable, running model of part of a system. It is incrementally developed into a final version, which is then used as the production system. Evolutionary prototypes are a form that underlies many release-oriented software products.”[[3]](#endnote-3)

Link: [http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4755727&isnumber=4755314](http://ieeexplore.ieee.org.ezproxy.umsl.edu/stamp/stamp.jsp?arnumber=4755727&isnumber=4755314)

[Go to TOP](#_top)

### Variety of Prototypes

There are different types of prototypes based on the stages of processes of solution development.

**First type** – when identifying the problem

Starting prototype, in which the analyst or developer is trying to identify the problem and to convince the client of the feasibility. This type of prototype can be a presentation or a meeting between those parties.

**Second type** – when discovering system requirements

When analyzing system requirements and business needs a type of prototype which was called “prototype proper” in [ii] is used. It is a mocked semi-functional system that is to show to the client what would be the solution in hand and to touch some of the possible problems. In case of University of Missouri St. Louis Calendar System this type of prototype was used when having the interviews by using the demo versions of commercial solutions’ specifications.

**Third type** – when designing and constructing

When in designing and construction the developer teams sometimes produce an unfinished version of the solution to have an internal touch on the product in this type of prototype the user is not included in. this type is called “breadboard” in [ii]

**Forth type** – communication with user

“Illustrative purposed” as referred in [ii] is type of prototype of a tangible entity that can be virtual or physical to share the results and get feedback.

**Fifth type** – when a final version of product or solution is ready to test

This time developer steps in application area with a so called “pilot system” [ii], to open the system to real word. The task of user is to use the pilot system which is now hardly a prototype but since the developer is open to feedbacks and ready to use them to change the pilot system it is still a prototype[[4]](#endnote-4)

[Go to TOP](#_top)

## Purpose of prototyping[[5]](#endnote-5)

### Clarify the problem

A prototype can help both developer and user to shed light to the environment and have a better understanding about the problem and help them to communicate better in a shared channel. It also helps them to avoid sticking to their own specific approach and brings variety of options by facilitating conversation.

[Go to TOP](#_top)

### Review the technical aspects

When developer has a system in hand and can test it by implementing it in a real working system he can better check the usability, feasibility and suitability of product with system requirements. On the other hand when user can test drive the prototype he can talk about the possible miscommunications and supports he may need for such a system when completed in future.

[Go to TOP](#_top)

### Adjusting and tailoring

When finalizing the delivery of ultimate product user gives feedback by using this semi-pilot or evaluating prototype to help the developer adapt the solution and bug the possible errors. Also developer role play an early controller to observe what needs to be done to complete the project with less cost and less maintenance tasks. v, ii

[Go to TOP](#_top)

## Prototyping Advantages

### Cost Saving

When a problem or an extra feature which has not requested is found during the communication on the prototype it definitely helps with a cheaper solution than finding the problem on finalized and delivered product. For example when you make changes on a your sketch by erasing a drawn line it is way cheaper and easier than tearing dawn a wall.([web link](http://www.precisionmachiningpro.com/2008/08/01/what-is-prototyping-advantages-and-disadvantages.html)[[6]](#endnote-6))

[Go to TOP](#_top)

### Find a better solution

Prototype affords a concrete media to list and check the requirements and gives a picture of how the final solution will fit or not in the designated area. By cutting the perplexity of communication of abstract specifications it improve the direct communication among stakeholders.[[7]](#endnote-7)

[Go to TOP](#_top)

## Problems with Prototyping

### Operation fails

Based on Boar’s findings[[8]](#endnote-8) typically prototyping function is insufficient[[9]](#endnote-9).

### Low success rate in big sized prototyping

The larger the size of prototype is the higher likelihood of failure. When a prototype is large, first thing to notice is the quantity of resources needed is larger, such as human resources and time resource. For example in a limited time frame when you want to produce a larger prototype you need to use more people so the communication error goes up.[[10]](#endnote-10)

[Go to TOP](#_top)

### “Unrealistic expectation” [iii]

Even by looking to same tangible entity not all of interpretation from it is similar. Number of different pictures is equal to number of people and when a good developer team leader is lacking to integrate the assumption there are colorful pictures of mixture of variety of colors in the system requirement design.

Often users expect more from what they see in a prototype. The general assumption is to see a really different outcome from what a user has seen in a prototype. A professional developer team leader or representative needs to make sure that the requirements limit has been identified based on projects’ resources’ limitation and clarified for user when getting feedback on prototyping to avoid the creation of overwhelmed infeasible expectations.[[11]](#endnote-11)

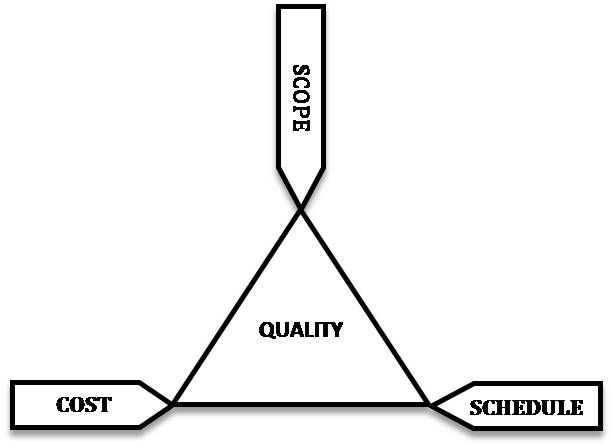
[Go to TOP](#_top)

## Implementation of prototyping results

“At every project's core is the trio of time, money, and [scope (scope: The combination of all project goals and tasks, and the work required to accomplish them.)](javascript:AppendPopup(this,'pjdefScope_1_1'))[[12]](#endnote-12). These are the factors you juggle every day to keep your Project plan on track” ([web link](http://office.microsoft.com/en-us/project/HA010211801033.aspx)[[13]](#endnote-13)).

These factors also apply to any software developing or system analysis project too, it is called Project management triangle.

[Click](http://www.youtube.com/watch?v=JAEjqXGd0lc) here to watch a short video about the project management.



[Go to TOP](#_top)

When a developer uses prototyping in his project process the modification to the projects setting based on feedbacks can be one of the options from the table below. All the content is from [iii] except the last row ‘system specs’, which I observed when studying some prototyping projects and also I experienced in University of Missouri St. Louis Calendar System ;

|  |  |  |  |
| --- | --- | --- | --- |
|  | Accepting | Changing | Ending |
| Time | TA | TC | TE |
| Resource | RA | RC | RE |
| Scoping | SA | SC | SE |
| System Specs | SSA | SSC | SSE |

[Go to TOP](#_top)

The choices are Accepting, Changing or Canceling. The decision maker on analyst side (developers) based on inputs and existing constraints makes the decision as depicted in the following graph;

**Ending (E)**

**Accepting (A)**

**Changing (C)**

**Accepting**; developer accept the users input and add the new setting to the process it can be a new system requirement, or new time frame or new project resources allocation.

**Changing**; developer receives the users input and modifies it then inform the user and got his approval in a communication process final modifications will make to project’s factors.

**Canceling**; developer receive the user’s input and based on feasibility and suitability analysis kills the project to stop further losses.

[Go to TOP](#_top)

## New technologies

[Click here](http://www.youtube.com/watch?v=PDLOmoQj4H0) to watch a video on how new technologies help quick prototyping

[Go to TOP](#_top)

## Conclusions

While prototyping is an essential tool and step for a solving process it can be useless if not implemented correctly. There are some pitfalls that system developer should aware when using the prototyping result. Also when expose the prototype to the user developer need to stay focused on user expectations.

[Go to TOP](#_top)

## References

B. Patton, “Prototyping-A nomenclature problem,” ACM SIGSOFT Software Eng. Notes, vol. 8, no. 2, pp.14-16, 1983.

Lichter, H.; Schneider-Hufschmidt, M.; Zullighoven, H., "Prototyping in industrial software projects-ridging the gap between theory and practice," Software Engineering, IEEE Transactions on, vol.20,no.11,pp.825-832, Nov 1994

Granlien, M.S.; Pries-Heje, J.; Baskerville, R., "Project Management Strategies for Prototyping Breakdowns," System Sciences, 2009. HICSS '09. 42nd Hawaii International Conference on , vol., no., pp.1-10, 5-8 Jan. 2009

1. G. Rzevski, “Prototypes versus pilot systems: Strategies for evolutionary information system development,” in Approaches to Prorofyping, Buddeer al., Eds. Heidelberg, West Germany: Springer-Verlag. 1984, pp. 34 1-356.
2. C. Floyd, “A systematic look at prototyping,” in Approaches to Prototyping,Budde et al., Ed. Heidelberg, West Germany: Springer-Verlag,1984, pp. 105-122**.**

http://www.precisionmachiningpro.com/2008/08/01/what-is-prototyping-advantages-and-disadvantages.html

1. T. Moynihan, "Coping with requirements-uncertainty:The theories-of-action of experienced IS/software project managers," The Journal of Systems and Software, vol. 53, pp.99-109, Aug 31, 2000.
2. B. Boar, Application Prototyping: A Requirements Definition Strategy for the 80's. New York: J. Wiley, 1984.
3. K. Lantz, E., "The prototyping methodology," Prentice-Hall, Inc., 1986, p. 191.
4. M. Alavi, "As Assessment of The Prototyping Approachto Information Systems Development," Communications of The ACM, vol. 27, 1984, pp. 556-563.

J. Iivari and M. Karjalainen, "Impact of Prototyping on User Information Satisfaction During The IS Specification Phase," Information & Management, vol. 17, 1989, pp. 31-45.

Chatfield, Carl. "[A short course in project management](http://office.microsoft.com/en-us/project/HA102354821033.aspx)". Microsoft. <http://office.microsoft.com/en-us/project/HA102354821033.aspx>.

http://office.microsoft.com/en-us/project/HA010211801033.aspx

1. *B. Patton, “Prototyping-A nomenclature problem,” ACM SIGSOFT*

   *Software Eng. Notes, vol. 8, no. 2, pp. 14-16, 1983.* [↑](#endnote-ref-1)
2. *Lichter, H.; Schneider-Hufschmidt, M.; Zullighoven, H., "Prototyping in industrial software projects-bridging the gap between theory and practice," Software Engineering, IEEE Transactions on , vol.20, no.11, pp.825-832, Nov 1994* [↑](#endnote-ref-2)
3. Granlien, M.S.; Pries-Heje, J.; Baskerville, R., "Project Management Strategies for Prototyping Breakdowns," *System Sciences, 2009. HICSS '09. 42nd Hawaii International Conference on* , vol., no., pp.1-10, 5-8 Jan. 2009 [↑](#endnote-ref-3)
4. G. Rzevski, “Prototypes versus pilot systems: Strategies for evolutionary information system development,” in Approaches to Prorofyping, Buddeer al., Eds. Heidelberg, West Germany: Springer-Verlag. 1984, pp. 34 1-356. [↑](#endnote-ref-4)
5. C. Floyd, “A systematic look at prototyping,” in Approaches to Prototyping,

   Budde et al., Ed. Heidelberg, West Germany: Springer-Verlag,

   1984, pp. 105-122**.** [↑](#endnote-ref-5)
6. http://www.precisionmachiningpro.com/2008/08/01/what-is-prototyping-advantages-and-disadvantages.html [↑](#endnote-ref-6)
7. T. Moynihan, "Coping with requirements-uncertainty:The theories-of-action of experienced IS/software project

   managers," The Journal of Systems and Software, vol. 53, pp.99-109, Aug 31, 2000. [↑](#endnote-ref-7)
8. B. Boar, Application Prototyping: A Requirements Definition Strategy for the 80's. New York: J. Wiley, 1984. [↑](#endnote-ref-8)
9. K. Lantz, E., "The prototyping methodology," Prentice-Hall, Inc., 1986, p. 191. [↑](#endnote-ref-9)
10. M. Alavi, "As Assessment of The Prototyping Approachto Information Systems Development," Communications of

    The ACM, vol. 27, 1984, pp. 556-563. [↑](#endnote-ref-10)
11. J. Iivari and M. Karjalainen, "Impact of Prototyping on User Information Satisfaction During The IS Specification Phase," Information & Management, vol. 17, 1989, pp. 31-45. [↑](#endnote-ref-11)
12. Chatfield, Carl. "[A short course in project management](http://office.microsoft.com/en-us/project/HA102354821033.aspx)". Microsoft. <http://office.microsoft.com/en-us/project/HA102354821033.aspx>. [↑](#endnote-ref-12)
13. http://office.microsoft.com/en-us/project/HA010211801033.aspx [↑](#endnote-ref-13)