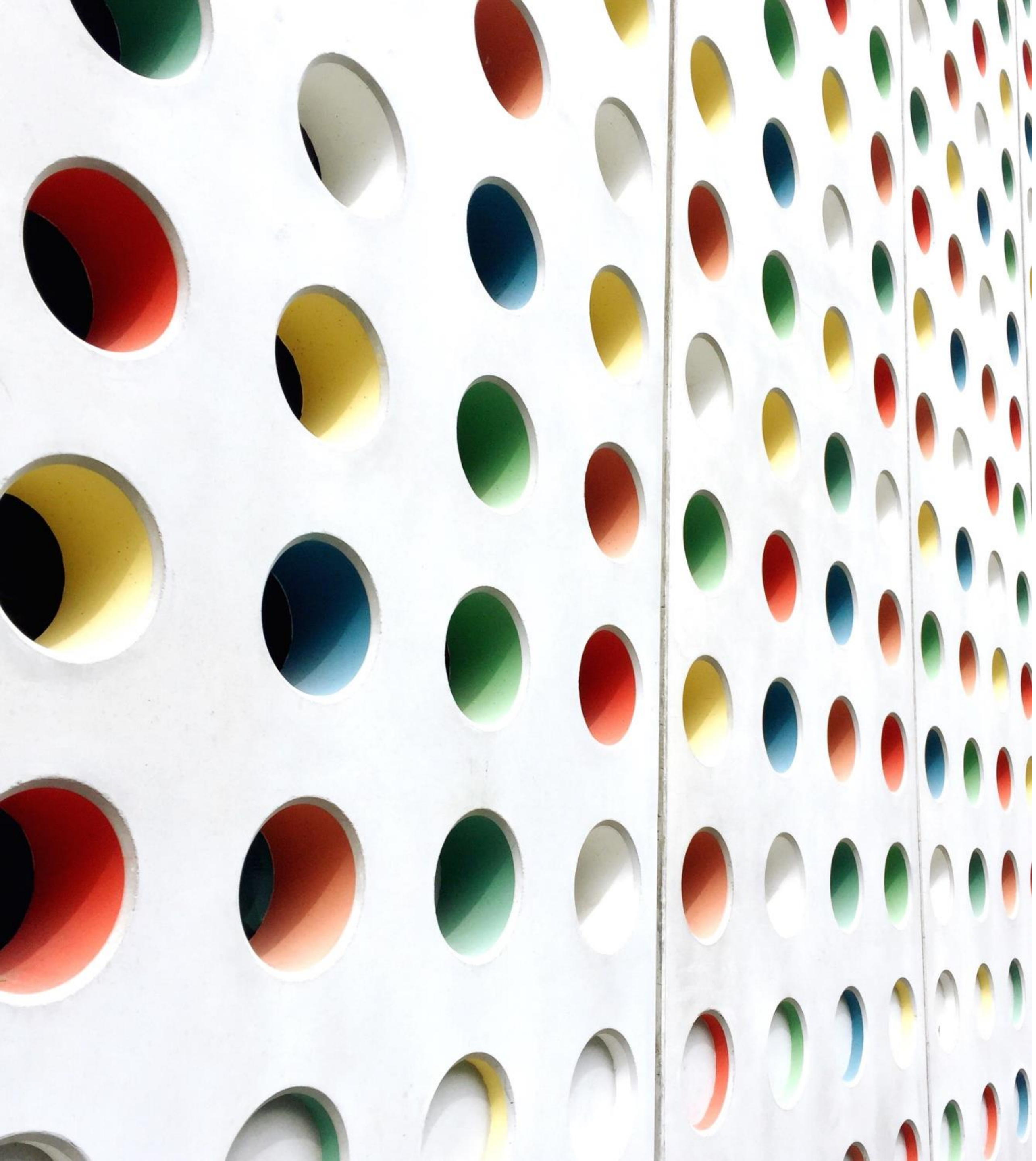


Prototyping Model

MSE2

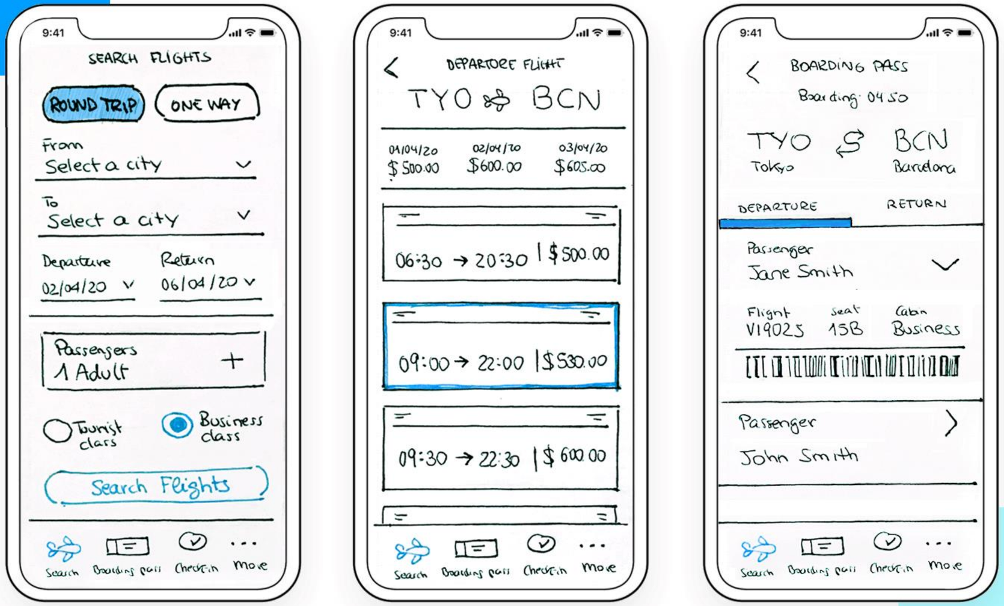
SOFTWARE ENGINEERING I

Prototyping Mode

- A prototyping model suggest that before carrying out the development of the actual software, a working prototype of the system should be built.
- A prototype is a toy implementation of the system.
- Prototype is a working model of software with some limited functionality.
- Prototyping is used to allow the users evaluate the developer proposals and try them out before implementation.
- By using this prototype, customer can understand the requirements of desired system and also the customer can get an “actual feel” of the system. It is an attractive idea for complex and bigger systems.

EXAMPLES

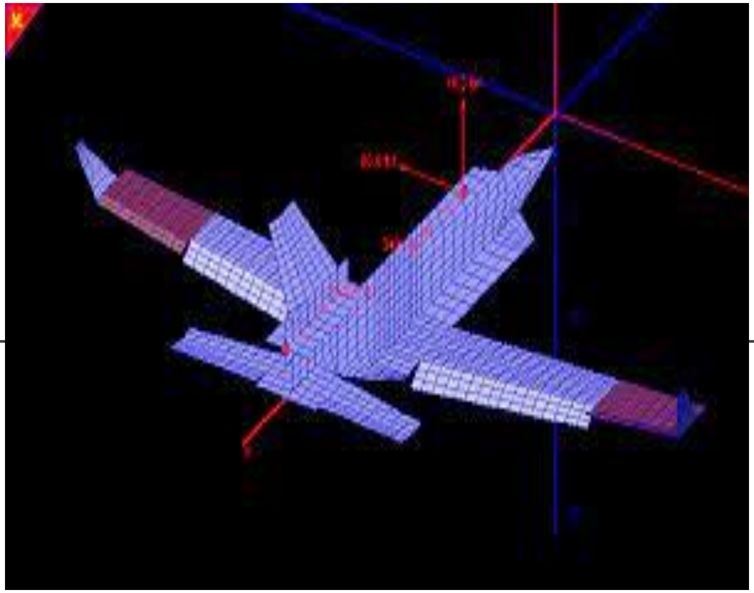
- **Paper Prototype-** Paper prototype is an example of throwaway prototype created in the form of rough or hand sketched drawings of the product’s interface, front-end design, and sometimes the back end work.



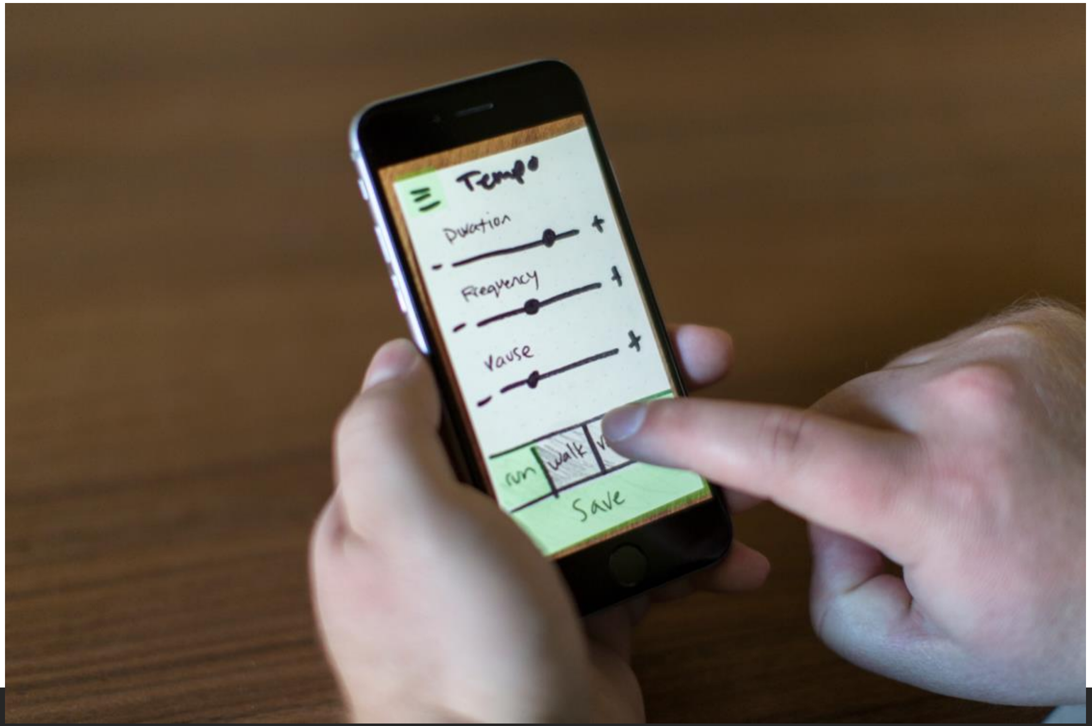
EXAMPLES

3D print- The latest technology has made it possible to print a 3d version of your product which can actually work. But this technique is not feasible for mass production.

EXAMPLES



- **Digital Prototype -**A digital prototype allows product developers to create a virtual model of the product which enables them to see how the individual components will work together and how the product will look once it’s completed. That is, it lets the developers virtually explore the complete product before it’s actually built.



Prototyping Process

- A preliminary project plan is developed
- A partial high-level paper model is created
- The model is source for a partial requirements specification
- A prototype is built with basic and critical attributes
- The designer builds – the DB, UI, Algorithmic functions
- The designer demonstrates the prototype, the user evaluates for problems and suggests improvements
- This loop continues until the user is satisfied

Prototyping Process

❑ the s/w engineers build a prototype during the requirements phase

❑The prototype is evaluated by the end users

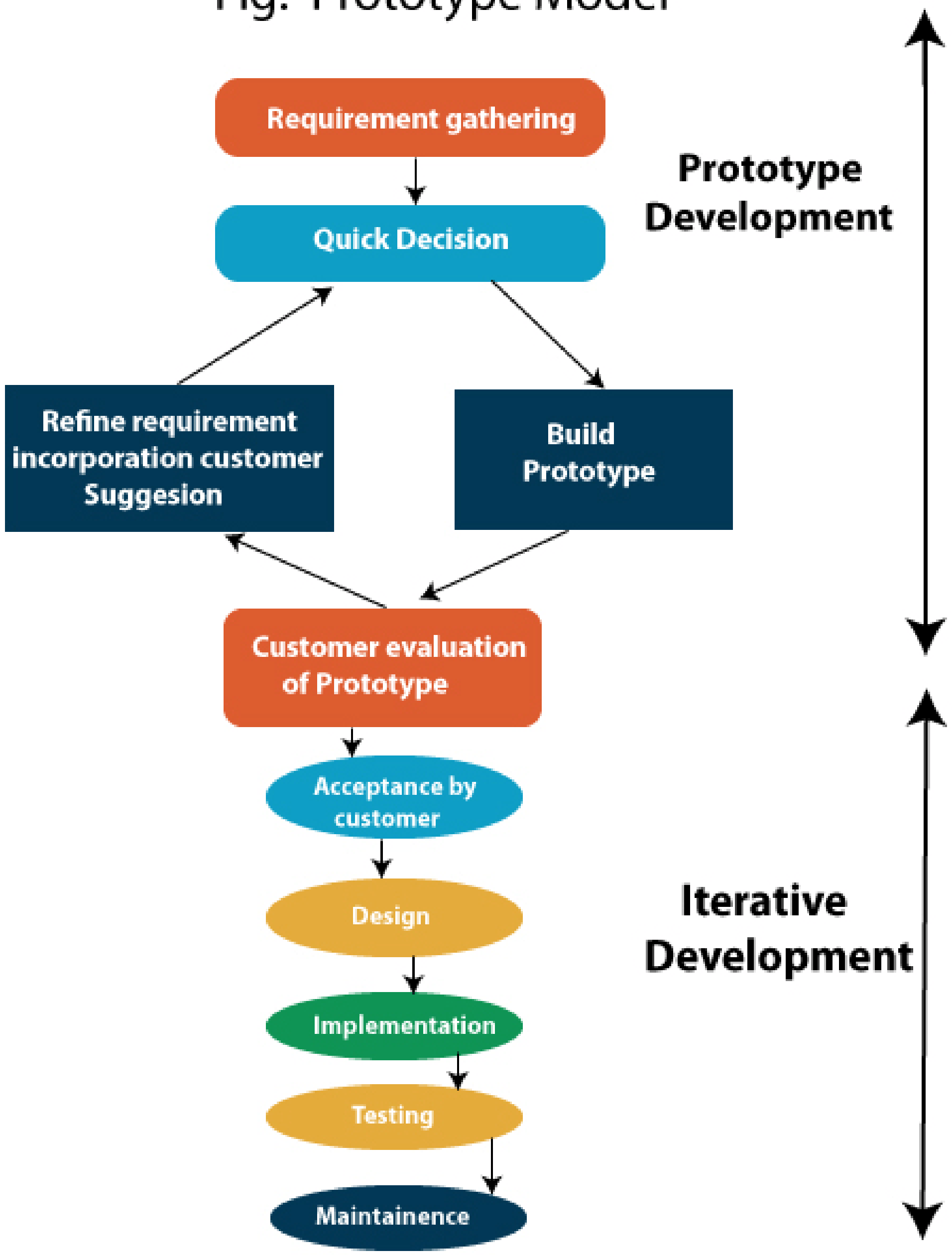
❑Users give corrective feedback

❑The developing team further refines the prototype

❑When the user is satisfied, the prototype code is brought up to the standards needed for a final product

Prototyping Model Diagram

Fig: Prototype Model



Types of Prototypes

Evolutionary prototyping

- An approach to system development where an initial prototype is produced and refined through a number of stages to the final system

Throw-away prototyping

- A prototype which is usually a practical implementation of the system is produced to help discover requirements problems and then discarded. The system is then developed using some other development process.

Prototyping objectives

The objective of ***evolutionary prototyping*** is to deliver a working system to end-users. The development starts with those requirements that are best understood.

The objective of ***throw-away prototyping*** is to validate or derive the system requirements. The prototyping process starts with those requirements that are poorly understood.

Uses of system prototypes

The principal use is to help customers and developers understand the requirements for the system

- **Requirements elicitation**. Users can experiment with a prototype to see how the system supports their work
- **Requirements validation**. The prototype can reveal errors and omissions in the requirements

Prototyping can be a **risk reduction** activity that reduces requirements risks.

Prototyping benefits

- Misunderstandings between software users and developers are exposed.
- Missing services may be detected and confusing services may be identified.
- A working system is available early in the process.
- The prototype may serve as a basis for deriving a system specification.
- The system can support user training and system testing.

Prototyping benefits

- Improved system usability
- Closer match to the system needed
- Improved design quality
- Improved maintainability
- Reduced overall development effort (?)

Advantages

Reduced time and costs:

Changes cost exponentially more to implement as they are detected later in development, the early determination of what the user really wants can result in faster and less expensive software.

Rapid delivery and deployment are sometimes more important than functionality or long-term software maintainability

Improved and increased user involvement

Prototyping requires user involvement and allows them to see and interact with a prototype allowing them to provide better and more complete feedback and specifications.

Not only is the system more likely to meet user requirements, they are more likely to commit to the use of the system

Disadvantages

Management problems

- Existing management processes assume a waterfall model of development
- Specialist skills are required which may not be available in all development teams

Maintenance problems

- Continual change tends to corrupt system structure so long-term maintenance is expensive

Contractual problems

- Excessive development time of the prototype
- Expense of implementing prototyping

User confusion of prototype and finished system

When to use Prototyping

○Projects for which either the user requirements or the underlying technical aspects are not well understood.

○Especially popular for development of the user-interface part of the project.

NEXT