Prototype

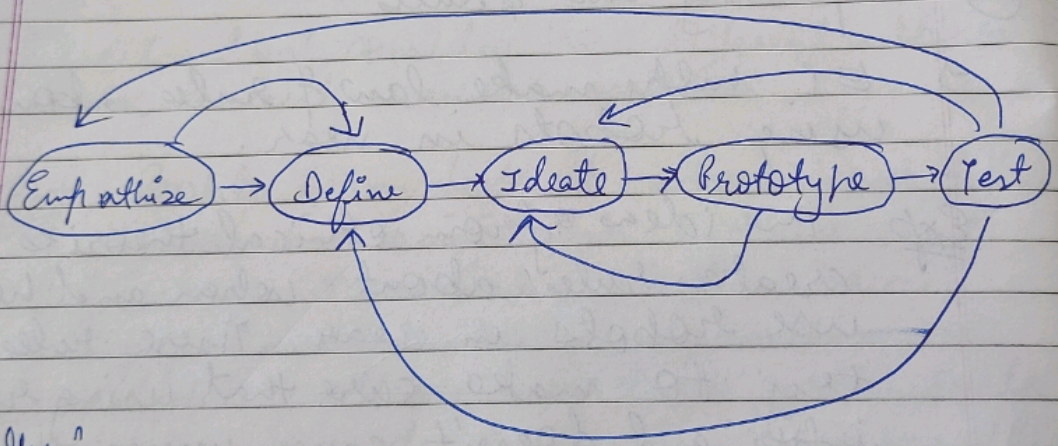
↳ Why? ⇒ Importance in design thinking process
⇒ Try, fail, try Again, Succeed loop

→ An artefact that someone can touch, hold, see or interact with in some way so that they can offer feedback.

↳ provides means for examining design problems and evaluating solutions

↳ Purpose :- what role artefact plays?
How should it look & feel?
How should it be implemented

↳ Important & Essential part of design thinking proc



Why?
↳ Evaluation & Feedback
↳ Stakeholders can interact with it rather than a drawing
↳ Communicate idea more effectively
↳ encourages reflection
↳ Helps answer questions
↳ helps choosing b/w alternatives

Types
↳ Thr
↳ Evol
feat
desi
↳ Exti

Make
⇒ Prot
vari
• End
• Des
↳
↳
↳
↳

Makes design, Human - Centered

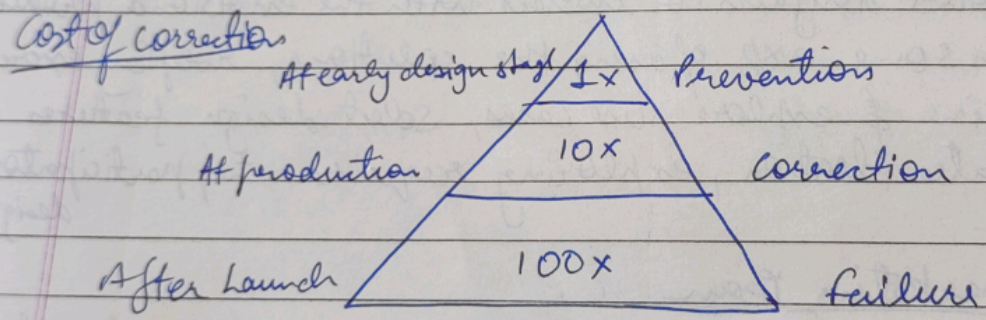
» Prototyping allows interaction of an idea with various stakeholders during the design process

- End Users
  ↳ Allows end users to interact with and explore its ~~sustain~~ • suitability & usability
- Design Team
  ↳ Allows designers to interact with the envisiond product
  ↳ gain some exp of using the solutions, roleplay exercise
  ↳ Imagine & explore use cases, select design features
  ↳ facilitates ideation, exploring requirements, participatory design
- Sales/Marketing Team
  ↳ Allows estimation of prospective sale based on reception of alpha versions; market analysis; product placement.
- Production Team
  ↳ Allows production of intermediary product for required analysis, process streamlining, identify

need for procurement of equipment.
↳ Allows incremental & iterative building of digital solutions

- funders/sponsers
  ↳ Allows product vizualization for investors; helps in go/no-go decisions.

1-10-100 Rule



Key Points

- Trial & Error → Developing > Testing > Improving
- facilitates Design Iteration
- Low cost - low risk way
- Saves time & money for innovators & stakeholders
- Allows for testing ideas before implementing

Types

↳ Throwaway :- cheap, fast, lo-fi to test initial ideas

↳ Evolutionary / Incremental → similar to final product, features are added as per feedback to refine the design

↳ Extreme → high fidelity for analysis driven testing

Forms

⇒ Physical (hard)
→ Physical (Soft)
→ Digital (Different System)
→ Digital (same System)
→ Interactive / Electronic

Fidelity

↳ refers to how close a prototype is to the final product

↳ there is no single right way ~~to~~ of choosing fidelity lvls
↳ It is based on the goal of the project, the current stage of the project, available resources

↳ Lo-fi → sketch / Drawing
↳ Hi-fi → HTML/CSS

Low-fidelity

↳ Provides a more tangible representation of the final Product

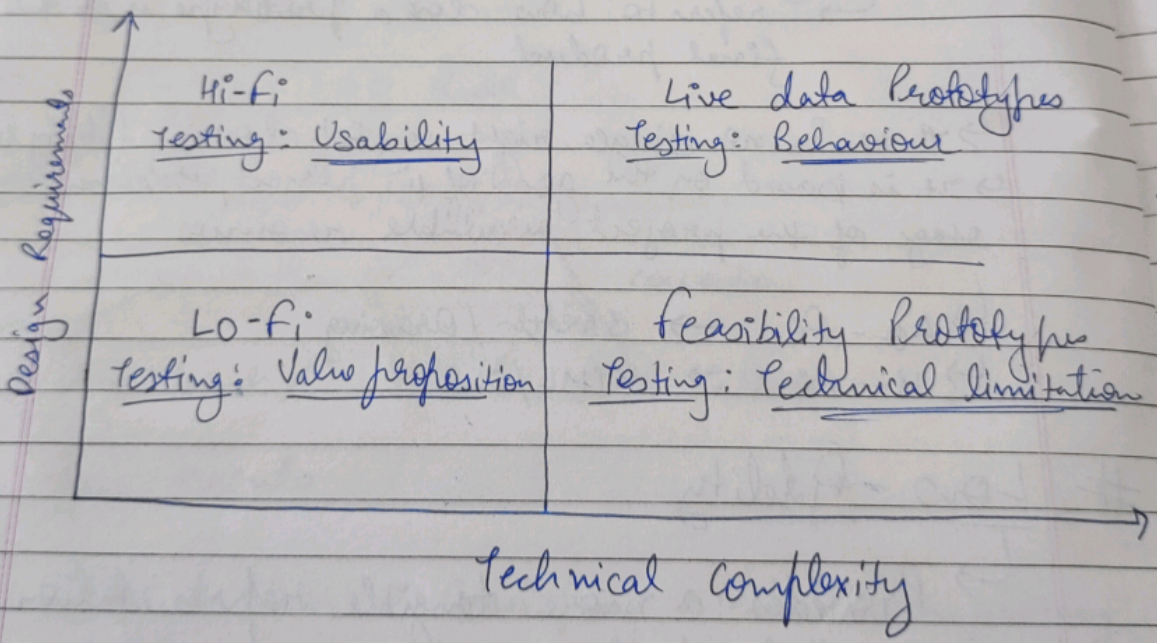
↳ The main goal is to outline the flow of the product and test the usefulness and usability of the functions.

↳ It can be useful in exploring various forms of ideation

↳ Methods :- sketches
Paper / cardboard Prototypes
Click-through Prototypes

High fidelity

↳ More technically complex & aesthetically pleasing
↳ closer to final product
↳ final stages of design for structured usability testing or analysis
↳ Methods : Interactive
Digital
Coded



Prototyping for Testing

Purpose
↳
↳
Co
⇒

Purpose

↳ to involve users in testing design ideas and get their feedback in the early stage of development to reduce time and cost.

↳ provides an efficient & effective way to refine and optimize interactions through discussion, exploration, testing, and iterative revisions

↳ Early evaluation can be based on faster & cheaper prototypes before the start of a full-scale implementation.

Cardboard Prototyping

↳ Most common Lofi prototyping method
↳ Can easily be done using cheap & available materials
↳ gives tangible form to an idea where functionalities can be tested
↳ Exploring tangible form

Software Based

⇒ faster
↳ Several real life combined in one
↳ Saves time required for physical prototypi
↳ Simulations, analysis, modifications can be done quickly

⇒ Cheaper
↳ Saves material costs, process costs
↳ Human resource can be easily trained

→ Iterative
↳ Modifications can be done at any stage easily.

Computer Aided Design (CAD)

→ 2D & 3D visualization & Design

→ Enables exploration, development, modification and optimization of design.

→ Engineering, architecture, design

→ Time and cost of physical prototyping saved

→ Easy to learn

→ Increased Productivity

→ Allows easy sharing & collaborative work

Uses

⇒ Product Design → Exploration & Visualization
⇒ Streamlining working mechanisms & movement
⇒ Engineering design → Analysis & Optimization

Types of CAD modelling

→ Solid Modelling
→ Wireframe modelling
→ Surface modelling

(1) Solid Mod
⇒ Works wit
⇒ Shapes ma
⇒ Some sta
to produc
⇒ Some pr
the so
it in
solid
⇒ Softw
⇒ Ad

(1) Solid Modelling

⇒ Works with 3D primitive shapes

⇒ Shapes may vary but they act like building blocks together

⇒ Some start with 2D sketches that are then extruded to produce a 3D figure

⇒ Some programs can use modifiers, working with the solids as if you were physically milling it in a workshop. Others add solid over solid in order to produce more complex figures

⇒ **Softwares**
- ↳ Entry level : TinkerCAD & freeCAD
- ↳ Advanced : SketchUp, SolidWorks, fusion360

⇒ **Advantages**
- ↳ Tools are easy to understand and work with; the user doesn't require extensive training
- ↳ Computational requirements are lower since the computer isn't working with thousands of triangles.
- ↳ The final pieces are always mathematically correct in the sense that the model is possible in the real world.

⇒ **Disadvantages**
- ↳ High realism in the representation of organic shape is almost impossible to achieve.

(2) Wireframe Modelling

⇒ Used in cases where the surface is complex & curved

⇒ Provides the finesse for more complex forms

⇒ Many W.f M tools use triangles as their basic elements, and the more triangles use, the higher the realism.

⇒ "Polygon Count", the total no of triangles contained within the wireframe of a model

⇒ Software: Blender, Maya, Daz 3D

⇒ Advantages:

⇒ It is possible to achieve more complex surfaces and curves compared to solid modelling

⇒ DisAdvantages:

⇒ Users require more training for developing complex organic forms

⇒ High resolution will require millions of polygons and computational needs will be higher.

(3) Surface Modelling

⇒ Highly professional applications demands smooth surfaces and seamless integration.

⇒ It relies on guiding lines to define the shape and curvature of a part. The software then calculates a smooth surface that connects the guiding lines

⇒ Used in automotive, aerospace

⇒ Since the seamless integration of all elements is necessary, surface modelling is the best way to approach these challenges.

⇒ Most solid modelling tools can handle this type of modelling. Ex:- Catia, freeCAD, SolidWorks

⇒ **Advantages:**

⇒ Possible to produce complex surfaces

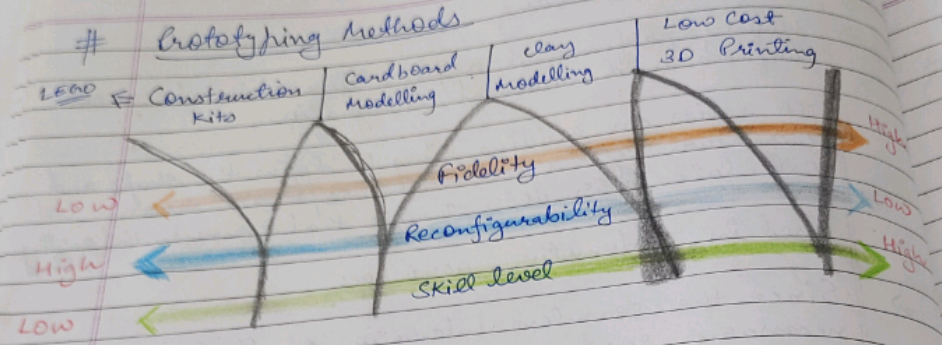
⇒ facilitates surface related analysis such as aerodynamic analysis in cars and aircrafts.

⇒ **Disadvantages:**

⇒ This technique is more complex and require more complex/advanced programs

⇒ More advanced programs will demand far more training & exp from the designer.

Prototyping Methods



Clay Modelling

⇒ Most common Lo fi prototyping method

⇒ It can also be done using easily available and cheap materials

⇒ It gives tangible form to an idea where functionalities can be tested.

⇒ Exploring tangible form

Laser Cutting

⇒ A slitting process with which metallic and non-metallic raw materials of different material thicknesses may be cut

⇒ This is based around a laser beam which is guided, formed and bundled

⇒ When it hits the workpiece, the material heats up to the extent that it melts or vaporises

⇒ The building of the part, model, or assembly is typically accomplished through assembly of cut parts in various ways
- Layered
- 3D Assembly

⇒ Faster, cheaper as compared to 3D printing

⇒ Can be done with various materials i.e. wood, paper, acrylic, metals

⇒ Innovative designs possible

ARDUINO

What?

⇒ An open source platform
⇒ Based on an easy to use hardware & software
→ A programmable microcontroller
→ User Friendly Development Environment (IDE)

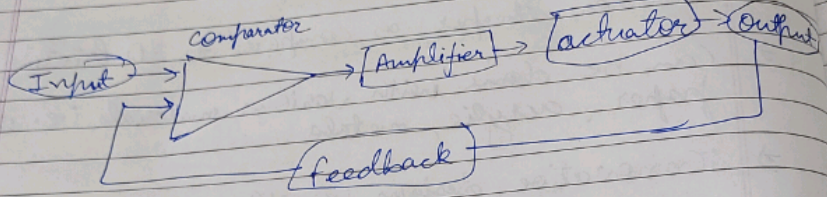
Why?

→ Low cost
→ Easy to learn
→ Comes with variety of compatible sensors
→ Easy to use IDE and uses simple coding in C++
→ Saves designers / students from the high cost of PCB design & printing

Open I/O Loop

Input → Amplifier → actuator → Output

closed I/O Loop



Logic

① AND (∧) $A \wedge B = Q$

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

② OR (||) $(A \| B)' = Q$

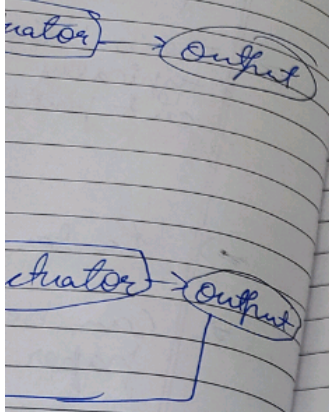
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

③ NOT (!)

A	A!
0	1
1	0

④ NAND

⑤ NOR



③ NOT (1.)

A	A!
0	1
1	0

④ NAND

⑤ NOR