The LUCID Design Framework
(Logical User Centered Interaction Design)

developed by Cognetics Corporation

LUCID – Logical User Centered Interaction Design – began as a way of describing the approach to interface design at Cognetics Corporation. It has evolved into a framework to manage the process of designing an interface in a way which incorporates the best industry practices of user-centered design and usability engineering.

Its goals are:

- To provide UI designers with a framework within which to apply best practices
- To allow for seamless integration of design and usability activities with software development methodologies
- To support a user-centered approach to interface design
- To enhance the usability of the finished software

	LUCID	
Logical	**User Centered**	**Interaction Design**
The design process builds on a strong conceptual model. Iterative review and refinement includes user feedback at all critical stages. Successive prototypes and team reviews allow opportunities for technical review and ensure viability of the design	Software is designed in the context of the overall tasks and work flow (including both manual and computerized activities). Design is based on user activity and employs the user's language and context. The design model fits the user's mental model rather than the technical implementation model.	Interaction design is treated as distinct from technical design. The scope of the design is "everything but code" and includes: □ look and feel □ language □ screen objects & layout □ navigation □ user assistance

Over the past 30 years, several techniques for managing software development projects have been developed and documented. While these techniques have helped large software development projects meet time, budget, and quality goals, they do not directly address usability issues. Because most systems being developed today are interactive, software development methodology must be expanded to include the design of the user interface. The LUCID Framework was developed to fill this need.

Overview

The activities in the LUCID framework are organized into six “stages”:

Envision
Develop a clear, shared and communicable vision of the product. Decide on the usability goals for the interface design. “UI Roadmap" to document the preliminary analysis and concepts developed during these activities

Conduct User and Task Analysis
Perform a comprehensive and systematic analysis of user task requirements through studying users to understand needs, expectations, tasks and work process and determine implications for the interface of this information

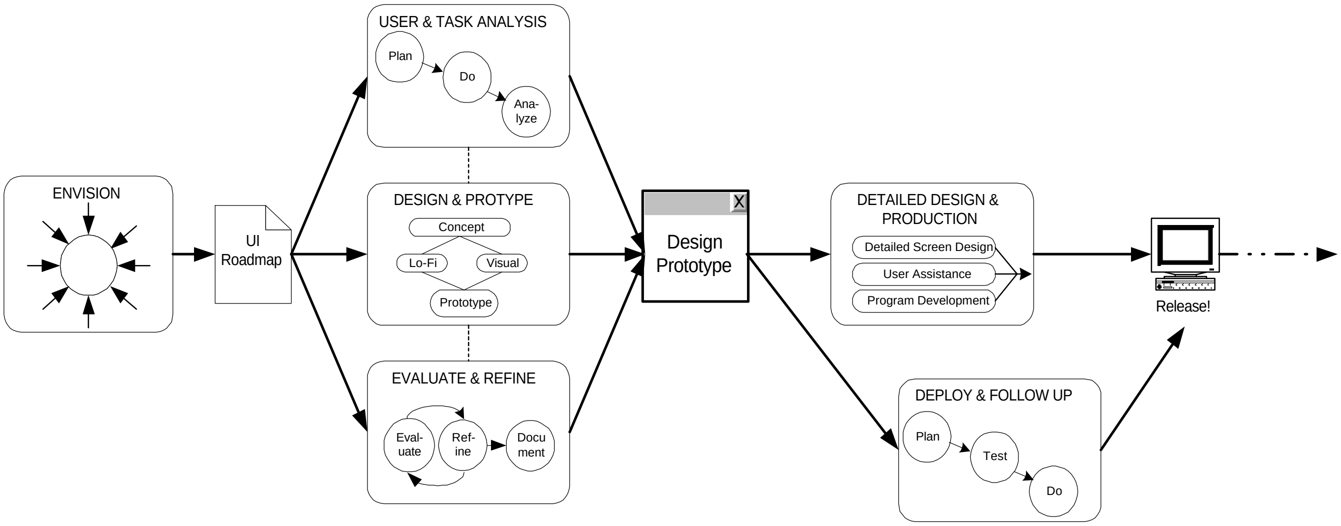
Design and Prototype
Create a design concept and create a key screen prototype to illustrate it

Evaluate and Refine
Evaluate the prototype for usability and iteratively refine expand the design.

Complete Detailed Design and Production
Complete the detailed screen design for the full program. Develop all user-assistance material included in the interface. Manage late-stage change.

Evaluate and Refine
Repeat usability evalution activities with early versions of the program or an enhanced prototype.

Release and Follow Up
Plan and implement the introduction of the product to users, including final usability evaluations to ensure that the has met the goals established at the beginning of the project. Create and monitor feedback mechanisms to gather data for future releases

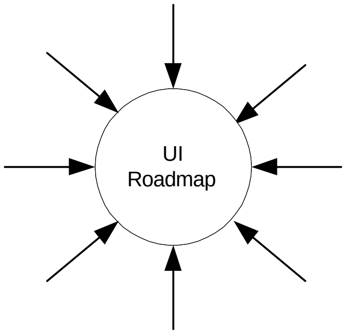


For more information: http://www.cognetics.com or info@cognetics.com

Envision

Goals:

- **Develop a clear, shared and communicable vision of the product.** An understanding of the design environment and requirements is essential to guide subsequent design and development. This vision is described in the UI Roadmap—a comprehensive high-level document that communicates the design vision, manages expectations, and serves as the basis for evaluating progress throughout the project.



- **Decide on usability goals for the interface design.** These usability goals will be used to guide design decisions, especially where compromises are required, and to measure the success of the interface.

Activities:

1 – Conduct Brainstorming Session

All stakeholders in the product (business managers, marketing, product management, user representatives, software developers and interface design team) work to come to consensus on the usability, business and development parameters for the product. Identify areas for further investigation. The results of this session are used in creating the UI Roadmap.

2 – Create UI Roadmap

Create a document outlining the goals and business and technical environment for the project. The UI Roadmap includes:

- A high concept statement – a short “mission statement” for the project
- Business objectives – a summary of the business goals
- Target users – a definition of the expected users of the software
- Technical or environmental constraints – system platform or physical environment
- Key functionality – a short description of how the software will be used
- User assistance – how will users be helped in learning and using the product
- Usability goals – what will be the measures of success
- Initial sketches - to document design ideas raised during the brainstorming

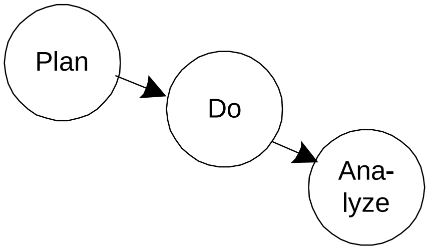
3 – Create Project Plan

Create or refine a project plan for user interface activities, ensuring that they mesh appropriately with other development activities.

Conduct User and Task Analysis

Goal:

Develop a comprehensive and systematic analysis of user and task requirements through studying the users to understand their needs, expectations, tasks and work process, and determine the implications for the interface of this information.



Activities:

1 – Plan

- ❑ Determine the types of user analysis and information gathering necessary for the project (What do we need to know?)
- ❑ Create a plan for data gathering, including dividing the target population into user segments (How will we get the information we need?)
- ❑ Prepare the team with background research in the subject domain as appropriate

2 – Gather Data

Options include: contextual observations, user interviews, participatory explorations such as PICTIVE, focus groups, usability tests, reuse of previous analysis. Satisfaction surveys or benchmark testing of the current solution may also be done to provide data for longitudinal comparison.

3 – Analyze User Requirements

Analyze user observations and other data gathered and create an analysis report documenting the user and task context and the implications for the interface design. This report should be accompanied by work-products from the analysis process:

- ❑ Scenarios documenting current or ideal interactions
- ❑ Workflow or task interaction descriptions
- ❑ Contextual issues, constraints or work-style observations, or other requirements
- ❑ User segment characteristics
- ❑ Task matrix matching user segments to key tasks
- ❑ User assistance required to support performance

4 – Review and Update the UI Roadmap

At the end of each module of work, the UI Roadmap should be reviewed by the entire product team in light of the new information and decisions, and revised if necessary.

The analysis and requirements document should be carefully reviewed with the software developers to ensure that technical implications are discussed early in the process. The interface requirements analysis must also be compared to the formal software requirements document when one exists.

Design and Prototype

Goal:

- **Create a design concept for the product and create a prototype to illustrate it.** As much as possible, the work of interface design should be done as a tangible, visible prototype rather than by describing the design verbally. The design prototype is iterated through user and expert reviews until the team is satisfied that the design concept meets usability goals and is strong enough to be used for complete UI design specifications.

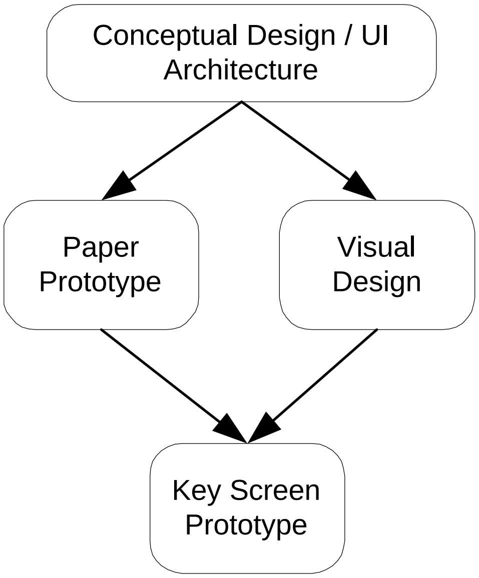
Activities:

1 – Develop Conceptual Model

The conceptual model for the interface establishes the overall architecture of the user interface and the way the users will interact with the software to complete their tasks.

2 – Develop Paper Prototype

The first prototype is a series of sketches illustrating key screens, information organization and task flow. The paper prototype can be used for cognitive task walk-throughs and user review. Because they are easy to create, they can be rapidly iterated.



3 – Develop Visual Design

The visual design includes the graphic look, screen layout, typographical choices and color rules. Sample screens are created to illustrate the design, and can be tested for acceptance and readability.

4 – Create Key Screen Prototype

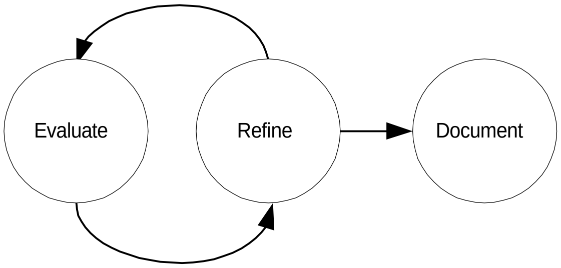
The key screen prototype merges visual design and task or information flow of the paper prototype into a representation of how the software will appear and function. Typically, these prototypes include the entry point, top level or menu screens, key work screens (often one task from entry to completion). Embedded user assistance, including on-screen text, roll-over prompts, procedural help or interactive coaches should also be represented in the prototype. The prototype can be interactive, but can also be a slide show of high-fidelity screen shots.

During all of the design activities, the team is gathering notes for the style guide or interface specifications, documenting design decisions as they are made.

Evaluate and Refine

Goals:

- **Complete the design of all functionality**
  Extend the key screen prototype to include the design for all functionality.
- **Evaluate the design**
  Evaluate the design through formal or informal usability testing and heuristic reviews, iterating and refining the design with the results of each evaluation.
- **Document the design**
  Create a style guide or design specifications to annotate the prototype and document design rules for the interface.



Activities:

1 – Complete the design for all functionality

The key screen prototype often does not include a representation of all functionality and must be extended to provide design solutions for all tasks and functions.

2 – Conduct usability evaluations

Options include: heuristic reviews, task walk-throughs with users, formal usability testing for task success, timed metrics or other scenarios. The methods selected should provide a way to evaluate the success of the design in meeting the usability goals established in the UI Roadmap.

3 – Create the style guide

The interface style guide documents the design decisions, especially visual design rules, and is used to guide the detailed screen design work. Depending on the scope of the project, the guide may be a simple annotation of a screen layout or may provide more detailed rules. If the interface must conform to industry or corporate style guides, the guide for this interface may either refer to those documents or be written as an addendum to them.

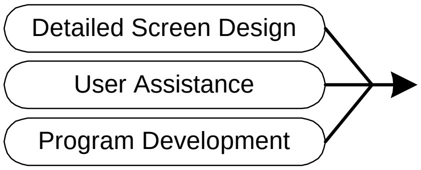
4 – Create design specifications

The design specifications document the interface, especially the structure and navigation, for the developers. The degree of formality of this document depends on scope and style of project, and interaction between the two teams. At a minimum, the specification annotates and explains the key screen prototype. At the other extreme are specifications which describe each screen and the placement and functionality of each object on it as well as menus, shortcuts or hot-keys. Connections to user assistance elements are included in the design specification.

Complete Detailed Design and Production

Goal:

- **Complete the development of the software.** During this phase, the software is completed and quality tested for release. The design activities support this goal by providing detailed screen designs, user assistance material and design decisions needed to support development.


Detailed Screen Design
User Assistance
Program Development

Activities:

1 – Complete Detailed Screen Designs

As the program is developed, each of the screens, dialogs or pages must be laid out in the development environment following the design rules. Icons or other graphics may also need to be developed. This work can be done by a member of the design team, or by the development staff with the work reviewed by the designers.

2 – Create User Assistance Material

Implement all user assistance material including final screen text, tool tips, embedded help, cue cards, reference and related materials such as tutorials, documentation, and quick start guides. Where appropriate, integrate the user assistance source files with the development code.

3 – Support Software Development

During this period, the core software development is completed and the interface design team supports the developers as necessary with reviews and design decisions. If technical limitations arise, the original usability goals and product concept documented in the UI Roadmap are used to guide compromises and adjustments to the design.

4 – Update style guide with late decisions

Update the style guide and design specifications with any information learned during the detailed design and development. These updates might include extensions to the design rules, more specific examples or annotation of specific issues relating to the development environment. If there is a corporate standard, new elements from this project can be added to the master document.

Release and Follow-Up

Goals:

- **Plan and implement the introduction of the product to users**
  The introduction of the product can determine its success or failure. If the introduction is inadequate, even the a product with the best interface may fail.
- **Create and monitor feedback mechanisms to gather data for future releases**
  It is essential that the creators of a program have a means to close this loop and integrate lessons learned from one design into the next effort, rather than simply handing off post-release support to a separate department.

Activities:

1 – Create introduction plan

The introduction plan is created by the same group of stakeholders who gathered for the Envision workshop that produced the UI Roadmap. Elements of the introduction plan include:

- Internal or external marketing materials or presentations
- Tutorials, both online and classroom training
- Help desk training
- Rollout method and sequence (pilot groups, beta tests, press requirements)
- Incentives (especially for internal corporate applications)

2 – Design the installation procedures

Design, prototype and support the development of the installation procedure. Often left to the last minute, the installation process is a separate design task and should be prototyped and tested like any other design.

3 – Test “out of the box” experience

Test the user’s initial experience with the product from “opening the box” through first successful task completion. These tests bring together the packaging design, installation process and initial user assistance materials in an integrated evaluation of the experience.

4 – Plan and implement follow-up evaluations

Create a plan to collect and monitor user feedback. Methods for follow up include:

- Downstream testing (especially longitudinal tests to determine if goals are met)
- QUIS or other satisfaction surveys
- Site visits and repeat observations
- Remote evaluation
- Marketing follow-up and analysis of technical support reports

5 – Planning for the next version

Determine how reported issues will be tracked and prioritized. Plan for monitoring and regular analysis of feedback for design implications.