

UMI Roadmap Workshop Workbook

"Precision instruments are designed to achieve an idea, dimensional precision, where perfection is impossible. There is no perfectly shaped part of the motorcycle and never will be, but when you come as close as these instruments take you, you go flying across the countryside under a power that would be called magic if it were not so completely rational in every way."

Robert Pirsig, "Zen and the Art of Motorcycle Maintenance"

"Straw Man Agenda":

- 9:00 Introduction to the Roadmap
 - 10:30 Set strategic and detailed usability objectives.
 - 12:00 Lunch
 - 1:00 Define the UI design process
 - 2:00 Identify a design team and clarify roles.
 - 3:00 Develop and incorporate user profiles
 - 4:00 Prioritize critical product components and task flow.
-
- 9:00 Develop a usability evaluation strategy.
 - 10:30 Incorporate the user assistance strategy into the usability plan
 - 12:00 Lunch
 - 1:00 Document and manage UI specifications and requirements
 - 2:00 Establish and maintain product design guidelines
 - 3:00 Maintain a feedback channel from users
 - 4:00 Summarize Cyborg Usability Roadmap

INTRODUCTION

What is Usability?

What is User Centered Design?

How did UMI arrive at the Roadmap?

What is a Software Development Methodology?

Discuss the high-level UI Process flow.

Discuss each component of the UMI Roadmap.

Overview of UMI Roadmap

The UMI Roadmap: The Steps Towards Product Usability

1. User and Task Analysis

- Identify and prioritize task flows and critical product components
- Set strategic and detailed usability objectives
- Develop and incorporate user profiles

2. UI Design Process and UI Team

- Identify a design team and clarify roles
- Define and implement a UI design process

3. Style Guide and UI Specification

- Establish product design guidelines
- Develop a procedure to document and manage Requirements and the UI Specification

4. Design, Evaluation, and User Assistance Strategy

- Develop and implement a usability evaluation strategy
- Develop and implement a user assistance strategy
- Maintain a feedback channel from users

1. Set Detailed and Strategic Usability Objectives

- Identify if usability objectives are already defined by client
- If objectives exist
 - facilitate a discussion to verify that usability and product objectives are in sync
 - are the usability objectives measurable?
 - are the usability objectives documented as non-functional requirements?
- If objectives do not exist or are not fully defined
 - provide an overview of usability objectives
 - facilitate a group brainstorming session to identify usability objectives
 - summarize and prioritize objectives from brainstorming session
- Discuss how usability objectives impact UI design
- Discuss UMI Usability Objectives deliverable

In Requirements Management, Usability Objectives are considered Non-Functional Requirements. Usability measurements may be considered as User Acceptance Criteria for the Non-Functional Requirement.

Definition of a Non-Functional Requirement:

Non-functional requirements are the expectations of the overall qualities, characteristics, or conditions to be exhibited by the product. They address characteristics such as performance, availability, and maintainability.

Characteristics of Usability Objectives
Must be measurable
Must be well-defined
Must be documented as a non-functional system requirement
Must be have an associated set of criteria for user acceptance (quantity acceptability)
Must support business objectives, user goals, and product goals
Must support and represent the user profile

DIMENSIONS OF USABILITY	USABILITY OBJECTIVE MEASUREMENTS
Learnability	Ability to learn and perform the task; Length of time to learn the task; If training is provided, can the user complete the task after training; Length of time for a new user to become proficient with the task (ability to perform useful work vs. mastery);
Efficiency	Performance (time to perform the task)
Memorability	Ability of a casual user to perform a task based on previous learning; Is the UI easy to remember?; Can the user remember where to find tasks and commands?
Errors	Number of user errors (catastrophic and non-catastrophic); Length of time to recover from an error.
Satisfaction	Subjective satisfaction rating; User's confidence level with the application; Is the application complete?; For non-business applications, is it entertaining to use?;

Usability Objectives

List the Top 5 Product Objectives:

Priority	Product Objectives
1	
2	
3	
4	
5	

List the Top 5 Usability Objectives¹:

Priority	Usability Objectives
1	
2	
3	
4	
5	

Usability Objective Measurements

Each usability objective has 1 or more attributes that objectively quantify the objective.

- **Attributes must be measurable**
- **Values are assigned to each attribute**
 1. Current Value - The “as is” value for performing the task with the existing system, a competitor’s system, or non-automated/manual process.
 2. Planned Value - The “to be” value for performing the task
 3. Best Value - The “state-of-the-art” value for performing the task. A better than expected value. May be a goal for a future release or enhancement of the system obtainable through improved design or technology.
 4. Current Assessment - The current value of the usability attribute

Usability Objective: _____ _____		Usability Measurement Values				
	Usability Attribute	Measurement Technique	Current Value (Unacceptable)	Planned Value (Acceptable)	Best Value (State-of-the-Art)	Current Assessment of Objective
1						
2						
3						
4						
5						

	Usability Objective	Impact of objective on Product Design	Product Components Affected
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

2. Develop User Profiles

- Identify if user profiles are already defined by client
- Identify differences between the customer and the user
 - If profiles exist facilitate discussion to verify that profiles are accurate and complete.
- If profiles do not exist or are not fully defined
 - provide an overview of user profiles
 - facilitate a discussion to identify the major user categories and distinguishing characteristics for each category
 - facilitate a discussion to identify the major tasks performed by each category (focus on task, not how or implementation)
 - identify skill level and frequency of use for each category
 - write a short summary of each user category
 - build a Characteristic Matrix - (X dimension - user categories; Y dimension - major characteristics)
 - build a User/Task Matrix - (X dimension - user categories; Y dimension - high level goals/objectives, user tasks, approaches to using the product)
- Discuss UMI User Profile Deliverable

Know Thy User! Constructing a User Profile

Brainstorm User Categories - Refine as you learn more about your users.

Gather User Profile information

- Individual characteristics
- Unique domain knowledge
- Familiarity with operating environment (Windows, Web browser) and hardware (PC, mouse)
- Motivation to use your product; User attitude toward the product
- Physical impairments (handicapped users, age/vision impairments)
- Languages
- Cultural differences

Gather data about the user's environment

- Workplace description and location
- Workplace constraints
- Relationship between users in the workplace

Brainstorm User Categories for your product.

-

These categories will be refined as you learn more about your user. This is a starting point.

Initial User Categories
1.
2.
3.
4.
5.
6.
7.
8.
9.
10.

Phases of Use

Below are the phases of use from which your users *start* using your product. As the user learns more about using your product and the subject domain, the user migrates through the 5 phases. Very few users make it to the Expert Phase, and most users who move beyond Beginner only make it to Advanced Beginner.

1. Phase 1 - Beginner - Getting starting phase
 - Initial user experience is critical (“out-of-the-box experience”, installation, menu design, help, tutorial, splash screen)
 - User’s goal is to get started, get comfortable, and to become productive quickly
 - Users draw on existing knowledge or perceptions about their area of domain knowledge or other products they’ve used.
2. Phase 2 - Advanced Beginner - Users begin to learn and perform tasks beyond the basic set of tasks; begin to understand context-specific uses of the product.
 - Users need procedural, step-by-step guidance (help, affordances within the UI)
 - Users will “learn by exploring” (cancel, undo, random access to tasks/commands)
 - Consistency and familiarity is important (industry standards, conceptual design, terminology)
 - Wizards allow the user to produce tangible results quickly
 - Tendency to cancel dialog boxes that are too complex or contain too many fields/settings
 - Infrequent or casual users may never advance beyond Phase 2
3. Phase 3 - Competent - Users start to solve problems using the product
 - Support problem solving
 - Avoid creating obstacles to problem solving (error messages should advise how to fix problems)
 - Users begin to formulate and execute plans for using the product to solve problems
 - Must be a frequent user to move to Phase 3
4. Phase 4 - Proficient - Users seek to understand, work more efficiently, and learn product or domain concepts
 - Provide usage examples of tasks that are important to the user
 - Users will rely on consistency between tasks
 - Users begin to perform more advanced tasks
 - Users must correctly understand product concepts in order to move onto Phase 5. Any misunderstood concepts must be learned.
5. Phase 5 - Expert - Users seek to further enhance efficiency, better understand and analyze results, and understand how the product works and product internals.
 - Users demand quick access to information and tasks
 - Users want complete reference information with examples - expert users learn by experimenting
 - User’s work is well defined, detailed, rapid, and fluid.

What are the Stages of Use for your product?

Stage of Product Use	Description	Tasks Performed	Comments
Phase 1 - Beginner		1.	
Phase 2 - Advanced Beginner			
Phase 3 - Competent			
Phase 4 - Proficient			
Phase 5 - Expert			

Categorize your user population based on the stages of use.

Build a Character Matrix

- Select the most relevant user characteristics
- Group your users into categories

Build a Situational Matrix

Build a Character Profile for each User Category

User Profile Information for a User Category

User Category:		
Characteristics of User Category	Range	Frequency Distribution (%)
General User Information		
Education (number of years)	<12, 12, 16, >16	
Age	15 - 65	
Sex	Male, female	
Native language	English, Spanish, German	
Is the user color blind? (red insensitive, green insensitive)		
Is the user handicapped? (hearing, motor, vision)		
Typing skill	none, hunt and peck, good, skilled	
Computer Experience		
General experience with computers	None - 5 years	
Computer literacy	None, low, intermediate, advanced	
General experience with hand held devices		
Experience with interaction devices (mouse, touch screen, pen)		
Experience using a GUI vs. a character based interface	0 - 10 years	
Operating system experience	DOS, Windows, UNIX, VM/CMS	
Experience using Windows 95	0 - 3 years	
Experience using Web Browsers	0 - 3 years	
Experience using email	0 - 5 years	
Experience using the Web to order products/services and to locate information		
Types of applications used frequently	word processing, spreadsheet, email,	

User Category:		
Characteristics of User Category	Range	Frequency Distribution (%)
	money management, graphics, overheads	
Job/Domain Experience		
Job Category (sales, clerical, professional, manager, executive)		
Job experience	None, beginner, intermediate, expert	
What tasks or activities are performed in this job?		
Applications used on the job	Spreadsheet, word processing, graphics, database, email	
How frequently are these applications used?	None, low, moderate, high	
Computer expertise required to perform job	None, beginner, intermediate, expert	
Turnover rate of employees in your position on the job	None, <10%, 10-20%, >20%	
Type of training or technical support provided to you on the job	None, formal training, hot-line, CBT training, manual	
How are you rewarded?	Salary, bonus	
Is use of this product optional or mandatory?		
Do you or have you used competitive products?		
Task Specific Characteristics		
How important is the task performed by this product in performing your job?	Low, moderate, high	
Is this task core or supportive in performing your job?		
How frequently do you perform this task	Never, daily, weekly, bi-monthly, monthly, quarterly, yearly	
Are you measured based on transactions,		

User Category:		
Characteristics of User Category	Range	Frequency Distribution (%)
accuracy, customer satisfaction?		

User Environment Profile

Environmental Characteristic	Snap-shots of Representative Users				
	User #1	User #2	User #3	User #4	User #5
Location where product is used					
Telecom bandwidth available to user					
Modem speed					
Monitor size and resolution					
CPU speed and capacity					
Desktop or laptop PC?					
Browser make and release					
Is user's workspace shared or private?					
Lighting conditions?					
Noise level?					
Interruption frequency?					
Type of telephone available to user?					
Desktop space available?					
Is application used away from the office?					

User Characteristic Matrix

1. Identify each user category/type. Record these categories along the x-axis
2. Identify key user characteristics along the y-axis.
3. Complete the matrix by assigning values to each User Characteristic in each User Category.
4. Assess the differences in characteristics between user categories and the impact on your design.

User Characteristic	Snap-shots of Representative Users				
	User #1	User #2	User #3	User #4	User #5
Number of years experience using a Windows 95					
Number of years experience using a Web browser					
Used the Web to order products or services?					
Used the Web to look up information?					
Number of hours/week spent on the Internet?					
Number of times email is accessed per week?					

User / Task Matrix

1. Record the User Categories/types previously identified along the x-axis (refer to the Character matrix)
2. Identify the high level goals/objectives and approaches to using the product along the y-axis.
3. Complete the matrix by assigning values to each User Task in each User Category.
4. Assess the differences in user situations between user categories and the impact on your design.

	Snap-shots of Representative Users				
User Task (Web-based Bookstore)	User #1	User #2	User #3	User #4	User #5
Access site					
Search site for a specific title?					
Search site for titles by author?					
Search site for titles about a specific subject?					
Want to “thumb through” the book before buying?					
Order 1 or more books?					
Provide credit card information on-line?					

Build a Character Profile for each User Category/Type

The goals of the Character Profile are to:

1. Personalize each user category
2. Describe each user category in the context of use
3. Describe how each user category will approach a situation and complete the task.
4. Describe any constraints

What each Character Profile should contain:

1. Brief 1 or 2 paragraph description that addresses the goals listed above
2. Photograph of a real user that meets the profile category. Photograph should be taken in the user's environment.
3. Use the first name of the user to "name" the profile category

Build posters that contain all the profile sheets.

1. Display the posters in the same area as the design team
2. Continuously show any on-site videos that capture the user actually performing the task.

3. Identify and Prioritize Task Flows and Critical Product Components

Discuss each usability method to perform task analysis

Usability Method	Description of Method	Results of Method	How Results are Mapped into the Bridge	Cost

-
- Identify if task analysis has been performed and is documented by the client
 - If task analysis is complete...
 - board major tasks of product and/or tasks associated with new features
 - board major product components and/or new product features
 - facilitate a discussion to ensure that existing task analysis documentation is complete and accurate.
 - briefly discuss task analysis techniques to identify opportunities
 - If task analysis has not been performed, facilitate the following activities for the defined scope (new product, product component, new release, maintenance release)...
 - build the *big picture* task flow that represents the job or reason for using the product (task flows constructed using notecards, Post-It™ notes, etc.)
 - build the *current* task flow that represents how the job is performed today
 - identify the *issues and bottlenecks* associated with the current task flow
 - build a *blue sky*, re-engineered task flow to perform the job
 - build the *realistic*, re-engineered task flow that will be constructed
 - facilitate a discussion to rate the major task characteristics (use a 1 - 10 scale).
Below are examples of task characteristics:
 - probability of performing this task for each user category (T)
 - frequency task is performed for each user category (F)
 - task importance, impact of error by user category (C)
 - performance objectives for each user category (P)
 - motivation to use the product to perform the task by user category (M)
 - impact of physical work environment (1 - no impact; 10 - major impact) (E)
 - impact of physical constraints in performing the task (1 - no impact; 10 - major impact)

- build a *Task Matrix* for the realistic task flow - (X dimension - user categories; Y dimension - high level tasks) - Sort this list based on task priority (e.g., importance and frequency of use).

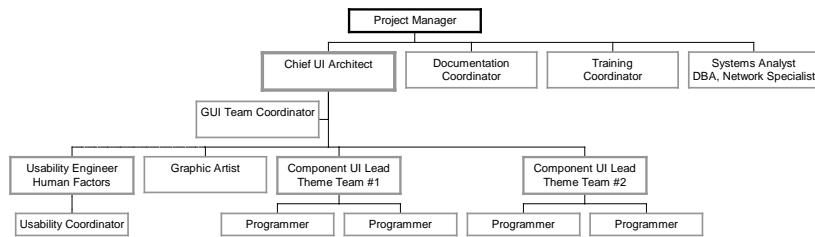
Tasks	User Categories																	
	User #1						User #2						User #3					
	T	F	C	P	M	E	T	F	C	P	M	E	T	F	C	P	M	E
Task 1																		
Task 2																		
Task 3																		
Task 4																		
Task 5																		
Task 6																		
Task 7																		
Task 8																		
Task 9																		
Task 10																		

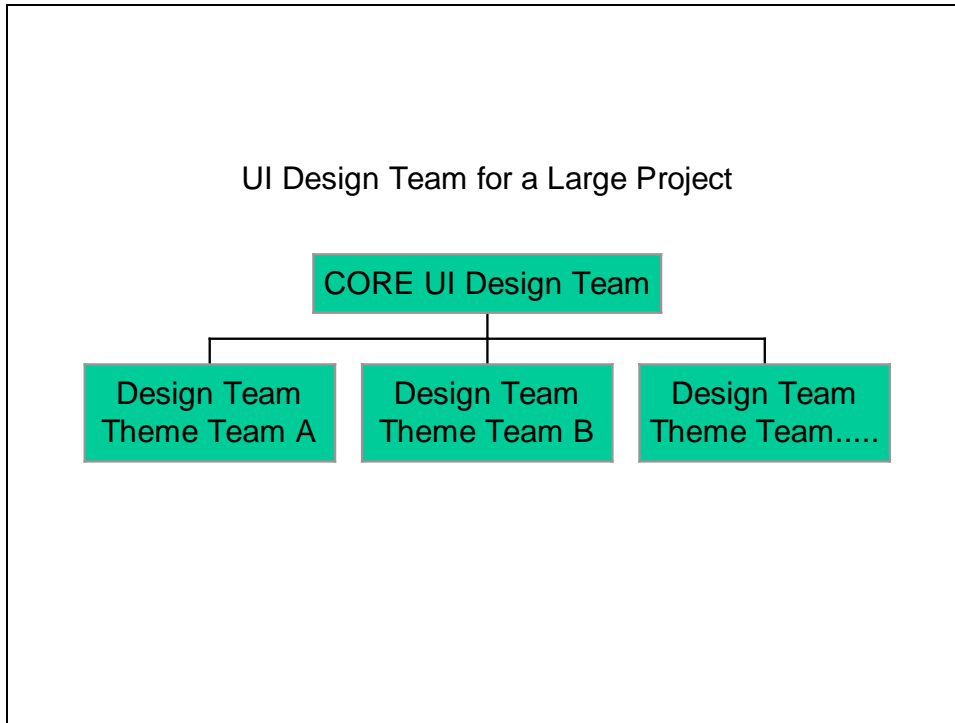
- Facilitate construction of high-level, typical usage scenarios
- Discuss UMI User Profile Deliverable

4. Identify a design team and clarify roles

- Identify if the client already has a design team in place with well-defined roles
- Assess the project size, project needs, available team resources, budget constraints, and schedule constraints to build a team model.
- If a design team is already in place...
 - review the team roles against the UMI recommended roles and identify gaps
- If a design team is not in place or is loosely defined...
 - provide an overview of the UI design team roles
 - facilitate a discussion to assign client personnel to the design team
 - clarify roles, responsibilities, and expected deliverables of each team member.
- Discuss the UMI UI Design Team deliverable

UI Design & Development Project Team





CORE UI Design Team - The CORE GUI Team has complete responsibility for GUI design, development, and maintenance throughout the life of the application. This team is composed of the Chief GUI Architect, GUI Design and Development Team leads, Usability Engineer(s), key user domain experts, and management responsible for user interface design. The CORE Team must be empowered to make all GUI design decisions. The turn-over rate within this team should be very low throughout the project.

Design Team (Theme Team) - Responsible for building the application GUI. Each major component of the application is represented by an individual Design Team. This is an effort to break down the development process into smaller manageable components. Each Design Team is responsible for design and development of a single UI component. The lead GUI analyst from each Design Team should be a member of the CORE GUI design team. Each Design Team must ensure that all GUI components are integrated together so that the navigation and hierarchy of the GUI is natural to the user. Each Design Team must adhere to application style standards and must integrate reusable GUI dialogs and components where appropriate.

CORE GUI Team should meet on a regular basis to:

- Review product design (prototypes and completed design) - review for style guide compliance, consistency with other windows, and adherence to design heuristics for usability.
- Review product and usability goals
- Review usability data - ensure that the GUI design meets usability goals and is acceptable to the user
- Ensure Corporate, Project, and Industry Style Guide compliance by all design teams
- Ensure that each GUI component is smoothly integrated into the overall GUI design and plan for navigation.
- Make routine GUI design decisions
- Update the Application User Interface Handbook

During initial project phases, the CORE GUI team should meet frequently to establish standards, direction, and develop early prototype design. As GUI design moves into implementation, the CORE team will meet less frequently.

A large project may have multiple GUI design teams (Theme Teams) that each design a component of the GUI.

- A GUI Design team member should participate in all UCD workshops that pertain to their GUI component. It is recommended that the same team member participate in all UCD workshops.

- As the GUI Design Team builds and modifies the interface, they should verify their design with the Task Flows and original window prototypes developed in the UCD workshops. Informal task walk-throughs will help identify design gaps, navigational problems, and missing information/functionality needed to complete the task.

Identify the UI design roles currently within your organization:

Title / Role	Description	Required Skill Set	Person Assigned

Map current roles into “to be” UI design team:

Proposed Role	Description of Role	Skills Required	Existing Role	Person Assigned
Chief GUI Design Architect	<p>Responsible for overall GUI design and all coordination between the GUI design teams (Theme Teams).</p> <p>The Chief GUI Architect is responsible for the following:</p> <ul style="list-style-type: none"> Identify and design any application <i>Metaphor</i> Ensure that the overall <i>GUI navigation model</i> meets the user’s task flow requirements and that this model is used consistently across design and GUI design teams. As GUI Objects and GUI Object Views are added or modified, the GUI/Task Object descriptions must be reviewed and updated accordingly. You must identify and document the impact of these changes on 	<p>The Chief GUI Architect acts as a mentor for each GUI design team and should have extensive experience in designing and building GUI applications.</p>		

Proposed Role	Description of Role	Skills Required	Existing Role	Person Assigned
	<p>other GUI Objects and Views in the design. You must also ensure that the new or modified GUI objects are consistent with documented Task/GUI Object descriptions and that the established heirarchy and relationships between these Objects is maintained.</p> <ul style="list-style-type: none"> • Ensure that all changes to Primary and Secondary Window are effectively communicated to all GUI design teams. • Ensure that all window designs are reviewed to ensure style guide consistency. • Work with Graphic Artists to effectively integrate graphic art into the design 			
CORE GUI Team Coordinator	<p>Responsible for overall coordination of the GUI Design team, scheduling of design and review meetings, creating and updating the Application User Interface Handbook and communicating all design changes/standards/conventions to all design teams and participants. The CORE team leader is an active participant in each design team and solicits problems and coordinates resolution to all design and style guide conformance issues. <u>The CORE Team Coordinator should be the librarian for the GUI Requirements document and all UCD workshop artifacts (task flows, task object descriptions, window prototypes).</u> The CORE Team coordinator should work with</p>			

Proposed Role	Description of Role	Skills Required	Existing Role	Person Assigned
	<p>the Chief GUI Architect and each GUI Design Team Lead to update and maintain the GUI Requirements document. As this document is updated, it should be re-distributed to all GUI design team members.</p> <p>As <u>librarian</u>, the CORE Team Leader should:</p> <ul style="list-style-type: none"> • Maintain the <i>GUI Requirements document</i> that contains a complete specification for GUI design • Maintain <i>Task Flow documentation</i> (index cards, task flow-charts) developed in the UCD workshops. Document task flow changes resulting from follow-up discussions or subsequent UCD workshops. Modify and/or create new Task Flows as driven by iterative design • Maintain <i>GUI/Task Object descriptions</i> (index card, stickies) developed in the UCD workshops. As requirements mature and as Object Views are added or modified, the GUI/Task Object descriptions must be reviewed and updated accordingly. • Maintain <i>Primary and Secondary Window designs</i> developed in the UCD Workshop and refined within each GUI Design Team. Throughout design and construction, it is critical that the GUI Requirements document and Window prototypes be 			

Proposed Role	Description of Role	Skills Required	Existing Role	Person Assigned
	<p>updated with design enhancements so that the changes can be effectively communicated to all design teams.</p> <ul style="list-style-type: none"> Maintain any supplemental documentation from UCD workshops 			
GUI Design Team Lead	<p>Each GUI Design Team has a lead analyst that is the spokesperson and design lead for that team. The Team lead should be the most experienced member of the design team and should have expertise in the development/production environment. The GUI Design Team Lead has the following responsibilities:</p> <ul style="list-style-type: none"> GUI Team Lead is the representative on the CORE GUI Team. In larger projects where there are multiple GUI design teams, each team lead should be on the CORE GUI Design team. Track and champion the team's GUI design throughout development Update the GUI Requirements document Argue for features critical to meeting usability/product objective and future enhancements Proactively exploit easily supported functionality requested by the user Ensure GUI design consistency between all GUI design teams <p>Ensure BST and application style guide consistency</p>			
GUI	Responsible for implementing			

Proposed Role	Description of Role	Skills Required	Existing Role	Person Assigned
Design/Coding Analysts	and constructing the GUI design prototype and production product. Each GUI Design Team is composed of one or more GUI analysts. These individuals must have expertise in the development platform and production environment (e.g., MS Windows, OSF Motif).			
Usability Engineer	<p>The Usability Engineer uses Task Flows identified in the UCD Workshops and Usage Scenarios to test usability and verify product and usability objectives of the prototype and fully functional GUI design. Usage scenarios are developed to test specific aspects of the interface in actual use by a typical user.</p> <p>The Usability Engineer should notify the CORE Team Leader, GUI Team Design Lead, and Requirements Representative of all usability problems identified within a Task Flow.</p> <p>The GUI Design Team will need to work with the Requirements Group to assess and resolve the usability problem. Refer to the original UCD workshop documentation to ensure that no information was lost or misinterpreted by the GUI Design Team.</p>			
Usability Coordinator	<p>Representative from the User Organization responsible for the following:</p> <ul style="list-style-type: none"> Schedule routine <u>informal</u> usability evaluations (refer to the Usability Planning Guide 			

Proposed Role	Description of Role	Skills Required	Existing Role	Person Assigned
	<p>for detailed information of planning and executing informal usability evaluations</p> <ul style="list-style-type: none"> • Schedule and coordinate formal usability evaluations after key deliverable (refer to the Usability Planning Guide for detailed information of planning and executing formal usability evaluations.) • Select users for the evaluation and coordinate transportation • Help analyse usability results • Work with the Usability Engineer to summarize and communicate usability results to the GUI Design Teams 			
Documentation	<p>Representatives from the organizations responsible for providing training and documentation to the end-user should be a member of the CORE GUI team. Not only may these individuals provide valuable input into your design, they should be involved early so that they can begin planning to implement appropriate packages and delivery mechanisms for training and documentation. In fact, these representatives should plan parallel sessions to gather requirements and specifications for training and documentation/help.</p> <p>On-line help and training packages may be tested for usability in parallel to the application. Training and documentation should complement and help achieve the usability goals of your application.</p>			

Proposed Role	Description of Role	Skills Required	Existing Role	Person Assigned
Training				

CORE Team Representative from the Requirements Group

Iterative design and testing will always introduce changes and insertions into requirements that must be carefully coordinated and documented. Large projects have formal processes for modifying requirements once “baselined” and changes to requirements are often frozen to meet product delivery dates.

The team responsible for building, documenting, and maintaining formal application requirements should have a representative on the CORE GUI Design Team. This representative is responsible for initiating change requests to the formal requirements baseline as a result of GUI design enhancements/changes.

Whereas the UCD workshops do not identify every detailed Task Flow of the application, the workshops do identify the principle flows that are the basis for GUI design. These flows are also critical to ensure design completeness and for GUI testing and should supplement scenarios typically developed for usability testing. Thus, the Requirements Representative and the Usability Engineer should communicate frequently.

Work with each GUI Design group to ensure that they understand and concur with the Task Flows that pertain to their group. Note that some Task Flows may pertain to multiple GUI Design groups. Throughout the design, the Requirements Representative should work closely with each GUI Design Team to identify changes and extensions to the Task Flows. Any Task Flow update must be communicated to all affected Design Teams. The Task Flow documentation should also be updated routinely to communicate the latest design baseline.

Management Representative

The manager responsible for GUI development and the management representative from the owner/user organization responsible for the project should be permanent members of the GUI design team. Whereas they may not participate in every activity and meeting, they should have the following roles:

- maintain a current knowledge of the issues, the GUI design, and all project schedules
- provide direction to the team
- obtain resources as needed
- provide access to users

Define the UI design process

- Identify if client already has a well defined UI Design process
- If a UI Design Process is already in place...
 - review the process against the UMI recommended UI Design Process
 - identify gaps and recommend where the client's process could be supplemented or enhanced based on the specific project or situation.
- If a UI Design Process is not in place...
 - assess where the project is positioned within the development lifecycle
 - provide an overview of the UMI's UI Design Process
 - facilitate a discussion to enhance the client's UI Design process
 - build a flow diagram of the recommended UI Design process
- Discuss the UMI UI Design Team deliverable

UMI's UI Design Process

1. Assemble UI design team and clarify roles
2. Develop usability test plan & integrate usability activities into the project schedule
3. Perform Data Collection activities (user profile, goals, needs ...)
4. Identify "high level" task flow, user activities & product components (collect user artifacts, business rules)
5. Select UI platform, GUI development tools
6. Produce a "detailed" task flow - reinvent & re-engineer
7. Develop usage scenarios & provide recommendations for any problems identified.
8. Develop a low fidelity prototype (conceptual model, metaphor, windows/dialogs)
9. Iterative design: test usability --> refine --> test (average usability test identifies 70-100 design issues)
10. Develop an application UI style guide (consistency, design templates, terminology)
11. Document the UI specification & Usability Requirements
12. Develop a navigational prototype
13. Iterative design: test usability --> refine --> test
14. Baseline UI specifications and prototype
15. Build
16. Evaluate user experience using the product
17. Document and analyze process for improvements

Day 2

Review Day 1

Provide an overview of Day 2

Develop a usability evaluation strategy

- Identify if the client already has a documented plan for usability
- Assess the project objectives, available team resources, budget constraints, and schedule constraints that may impact the usability plan.
- If a usability evaluation strategy is already documented...
 - review the usability evaluation plan against the UMI recommended strategy and identify gaps
 - determine if the usability plan is integrated into the project schedule
 - usability tests should be milestones for project management
 - identify how usability test results are integrated back into the UI design and project requirements
- If a usability evaluation strategy is not documented or is loosely defined...
 - provide an overview of the UMI Usability Evaluation strategy
 - facilitate a discussion to draft a Usability Evaluation strategy document
 - identify what will be tested
 - identify when it will be tested
 - identify how it will be tested
 - identify the objectives of each test
 - facilitate a discussion to integrate the usability plan into the project schedule
 - establish usability tests as milestones for project management
 - discuss how usability test results should be integrated back into the UI design and project requirements
 - discuss the process for changing the Usability Evaluation Strategy
- Discuss the UMI Usability Evaluation Strategy deliverable

Scenario ID:
Scenario Name:

Scenario Narrative:

Task #	Task Description	Task Detail
1		Rqmts: Success Criteria: Max Time to Complete:
2		Rqmts: Success Criteria: Max Time to Complete:
3		Rqmts: Success Criteria: Max Time to Complete:
4		Rqmts: Success Criteria: Max Time to Complete:
5		Rqmts: Success Criteria: Max Time to Complete:
6		Rqmts: Success Criteria: Max Time to Complete:
7		Rqmts: Success Criteria: Max Time to Complete:
8		Rqmts: Success Criteria: Max Time to Complete:
9		Rqmts: Success Criteria: Max Time to Complete:
10		Rqmts: Success Criteria: Max Time to Complete:

For each User Category, identify the target number of users participating in the study that will meet each key user characteristic. For example,

Number of Users that meet Category Characteristics		
User Category	Key User Characteristics	
	#1	#2

Establish Product design guidelines

- Identify if the client already has a documented project Style Guide
- Assess the project objectives, positioning within a product family, usability test results, available resources, etc. that may impact the Style Guide content.
- If a Style Guide is already documented ...
 - review the Style Guide against the UMI recommended style guide outline and identify gaps
 - assess the relationship between the UI Spec and Style Guide (should be a quick access reference)
 - ensure that the team has a process for updating and managing the style guide
- If a Style Guide is not documented or is loosely defined...
 - provide an overview of the UMI Style Guide template
 - discuss the relationship between the UI Spec and the Style Guide
 - facilitate a discussion to draft a table of contents for the Style Guide
 - facilitate a discussion to determine who will author and maintain the Style Guide
 - facilitate a discussion to identify the process to be used to update and manage the Style Guide as a living document
- Discuss the UMI Application Style Guide deliverable

UMI Yardstick Elements
1. Clarify the core concepts.
2. Plan and maintain consistency.
3. Fit content to customers who use the product.
4. Provide reassuring feedback.
5. Clarify interaction rules.
6. Structure navigation clearly.
7. Use plain terminology.
8. Optimize user assistance.
9. Optimize visual design.
10. Design for the context of use.

-
1. Introduction
 2. Hardware and Operating System Support
 3. GUI Window Specification Template
 4. Web Page Template
 5. Clarify the Core Concepts (Yardstick element #1)
 6. Plan and Maintain Consistency (Yardstick element #2)
 - 6.1 Window Title and Management
 - 6.1.1 Window Titles
 - 6.1.2 Window Parenting
 - 6.1.3 Window Menu and Controls
 - 6.1.4 Window and Dialog Placement and Size Policies
 - 6.2 Page Titles (Web/HTML)
 - 6.3 Common Dialogs
 - 6.4 Input Fields
 - 6.4.1 Field Labels
 - 6.4.2 Input Field Formats
 - 6.4.3 Image Labels (Web)
 - 6.5 GUI Controls
 - 6.5.1 Control Area Command Buttons (Secondary window, dialog box)
 - 6.5.2 Client Area Push Buttons (Primary window, Secondary window, Dialog box)
 - 6.5.3 Web Page Command buttons
 - 6.5.4 Window Panes
 - 6.6 Menus
 - 6.6.1 Pull-down Menu Bar Categories
 - 6.6.2 Pop-up Menus
 - 6.7 Non-Standard GUI Elements (Internally developed and project licensed)
 - 6.7.1 Notebook (tab control)
 - 6.8 Tables (Web/HTML)
 - 6.9 Printing (Web/HTML)
 - 6.10 Undo
 - 6.11 Capitalization
 7. Fit Content to User Needs (Yardstick element #3)
 8. Provide Reassuring Feedback (Yardstick element #4)
-

- 8.1 Error Handling
- 8.2 Handling Input Errors in Web Design
- 8.3 Error Messages
- 8.4 Warning Messages
- 8.5 Informational Messages
- 8.6 Mouse Pointer Shapes
- 8.7 Download Feedback in Web design

- 9. Clarify Interaction Rules (Yardstick element #5)
 - 9.1 Window Resize Policy
 - 9.2 Mouse Button Definitions

- 10. Structure Navigation Clearly (Yardstick element #6)
 - 10.1 Keyboard Equivalence
 - 10.1.1 Accelerator Keys
 - 10.1.2 Mnemonic Keys
 - 10.1.3 Function Keys
 - 10.1.4 Tabbing between UI elements
 - 10.1.5 Initial Window Focus
 - 10.1.6 Web/HTML Guidelines for Keyboard Equivalence
 - 10.2 Hyperlinks (Web)
 - 10.3 HTML Web Page Navigation
 - 10.4 Help Access

- 11. Use Plain Terminology (Yardstick element #7)
 - 11.1 Terminology
 - 11.2 Abbreviations & Acronyms

- 12. Optimize User Assistance (Yardstick element #8)
 - 12.1 Methods of User Assistance for the Web
 - 12.2 Help Page Template

- 13. Optimize Visual Design (Yardstick element #9)
 - 13.1 Color Coding
 - 13.2 Fonts

13.3 Icons

13.4 Logos

13.5 Graphic Images

13.6 Background Images

13.7 Thumbnail Images

14. Design for the Context of Use (Yardstick element #10)

Develop a UI specifications documentation procedure

- Identify if the client already has a well-defined process and style template for documenting a UI Specification
- Assess the project size, project needs, available team resources, budget constraints, and schedule constraints that may impact documenting the UI Specification
- If a UI Specification is already documented and a process established to create and maintain ...
 - review the documented spec and process against UMI recommendations and identify gaps
- If a UI Specification and process is not in place or is loosely defined...
 - provide an overview of the UMI UI Specification process and template
 - facilitate a discussion to draft a table of contents for a UI Spec
 - facilitate a discussion to draft a process to author, baseline, change, and maintain the UI Spec over the lifecycle of the project
 - facilitate a discussion to assign client personnel to author and maintain the UI Spec - assign responsibilities and expected deliverables
- Discuss the UMI UI Specification deliverable

1. Introduction

1.1 Purpose and Scope of UI Spec

1.2 Revision Summary to UI Spec

1.3 Target Audience (of this document)

1.4 Review and Concurrence Information

This section contains sign-offs from the project sponsors (client) and Development/Design.

This section should also list the major stakeholders in the design and development.

1.5 Structure and Use of the UI Spec

1.6 Requirements Traceability and Numbering

Specify company methods and processes for documenting requirements (what information should be specified in each requirement)

Specify how requirements should be numbered.

Discuss how traceability is implemented.

2. User Requirements

2.1 Overview

Provide an overview of the user requirements.

2.1.1 Product Objectives and Benefits

2.1.2 Assumptions, Dependencies, Constraints

2.1.3 User Profile Description

2.1.4 User Tasks

List each "high level" user task from Task Analysis. Describe the tasks that comprise the user's job or function.

2.1.5 Work Environment

Describe the physical environment in which the tasks will be performed. Include constraints, interruptions, stress, etc.

2.1.6 Related Products

Describe the other products used by the user to accomplish their job and how they interact with the product described in this spec.

2.1.7 Additional Information Users Need

2.2 Task Flows

Graphical representations of the user's task flow.

2.2.1 Realistic Task Flows

The re-engineered and optimal task flow that represents how the user's job will be performed. This task flow is the basis for the UI design.

2.2.2 Novice GUI-Object Flow Examples

If applicable, specifically represent how a novice user will complete a task using the Realistic Task Flow. This is a specific use example for a category of users documented in the User Profile.

2.2.3 Expert GUI-Object Flow Examples

If applicable, specifically represent how an expert user will complete a task using the Realistic Task Flow. This is a specific use example for a category of users documented in the User Profile.

3. GUI Platform and Environment Assumptions

3.1 Platform Constraints

-
- 3.1.1 System Configuration Requirements
 - Hardware, CPU, Monitor, Network, Modem, etc.
 - Performance requirements
 - 3.1.2 GUI Style Guide Compliance
 - List the corporate and industry style guides that must be implemented.
 - 3.1.3 Development Toolkits
 - List the tools that will be used to build this product (name, vendor, version)
 - 3.1.4 Operating System
 - 3.1.5 Desktop Manager

4. Common GUI Requirements

- Describe any component of GUI design that will be consistent across this product.
 - Dialog design
 - Menu design
 - Icon design
 - Command Buttons
 - Short-cut keys (mnemonics, accelerators, function keys)
 - Font usage
 - Color usage

5. Requirements Specific Window (representing a GUI Object)

- 5.1 Launching the Application
 - 5.1.1 Desktop Representation
 - How will this application be represented on the desktop? Icon design? Icon Location?
 - How will drag and drop to implemented on the desktop?
 - What are the pop-up menus associated with the application icon?
 - 5.1.2 Login and Security
 - Document the steps the user will take to log onto this application.
 - What are the security standards that must be implemented?
 - What are the dialogs that will be presented during login?
 - What other information must be presented to the user (e.g., copyright, proprietary markings)?
 - 5.1.3 Splash Window
 - Define the graphics and presentation of the Splash window
 - Define the objective of the Splash.
- 5.2 GUI Objects
 - Provide a visual representation of the GUI object structure that represents the Realistic Task Flow.
 - 5.2.1 GUI Object Name
 - 5.2.1.1 Parent/Children
 - Describe the parent/child GUI object relationships
 - 5.2.1.2 Closed GUI Object Representation and Actions
 - How will the closed GUI object be represented?
 - Describe what actions can be applied to the closed Object representation.
 - Describe the direct and indirect manipulation actions that can be applied to this closed GUI object.
 - 5.2.1.3 Pop-up Menu
 - Describe the pop-up menu and selections for the specified GUI object.
 - 5.2.1.4 Drag and Drop (Valid Drop Zones)

If drag and drop is implemented, describe the valid drop sites for the GUI object when dragged.

What is the default drop action?

Describe the overall drag and drop behavior for the object.

5.2.1.5 Window Definition

Show a picture of the window design

Provide the window type (primary, secondary, dialog)

Describe the GUI elements used with the client area and window border.

Describe any other window properties (e.g., modality, size, placement, title, colors, window menu)

5.2.1.6 Tasks Supported by Window

List the tasks supported by this window. Provide traceability back to the Realistic Task Flow.

5.2.1.7 Pull-down Menu-bar

Graphically show the pull-down menus and selections for the window. This is a job-aid that shows all the menus posted. Include accelerators, mnemonics, and separators.

5.2.1.8 Tool Bar

Show and describe each tool bar icon.

Provide the equivalent menu selection that represents the tool bar action.

5.2.1.9 Summary of GUI Object Views

List and describe all Views defined for the GUI Object represented by the window

Provide the <View> Name

Describe the View design (show all GUI elements in the design)

Describe miscellaneous View Properties

5.3 Common Dialog Requirements

5.3.1 Name of Dialog

5.3.1.1 Dialog Design

6. REFERENCES

7. BIBLIOGRAPHY

8. GLOSSARY

Develop a User Assistance strategy

- Identify if the client already has a design team in place with well-defined roles
- Assess the project size, project needs, available team resources, budget constraints, and schedule constraints to build a team model.
- If a design team is already in place...
 - review the team roles against the UMI recommended roles and identify gaps
- If a design team is not in place or is loosely defined...
 - provide an overview of the UI design team roles
 - facilitate a discussion to assign client personnel to the design team
 - clarify roles, responsibilities, and expected deliverables of each team member.
- Discuss the UMI UI Design Team deliverable

Techniques and Tools for Providing User Assistance

UI Elements

1. Tool tips - role over help
2. Status line
3. On-screen prompts
4. Labels
5. Icons

UI Feedback

1. Feedback to user actions
2. Messages (error, warning, information, status)
3. Navigational Mapping - where am I?

Graphics

Context Sensitive “What’s This?” Help

Help

1. Table of Contents of Help Topics
2. Index
3. Find

Hypertext links

Tutorial

Splash Screen

Maintain a feedback channel for users

- Identify if the client already has a well-defined feedback channel in place
- If a feedback channel is already in place...
 - review the feedback mechanism against the UMI recommended mechanism
- If a feedback channel is not in place or is loosely defined...
 - provide an overview of the feedback channel recommended by UMI
 - facilitate a brainstorming session to identify the potential feedback channels for this project/product
 - from the brainstorm list, identify the top 3 feedback mechanisms and discuss a strategy for implementation.
 - facilitate a discussion to identify who is responsible and the deliverables
- Discuss the UMI Feedback Channel deliverable

ISO 13407 Requires a feedback mechanism for the HCI design activities and associated methods.