

Phase	Activity ID #	Activity
<b>Strategy and Planning</b>		Project Definition
		Initial Project Plan
		Initial Data Analysis
		User Interface Architecture
		Initial Requirements
<b>Planning and Analysis</b>		Refine the Project Plan
		Select UI Platform and Tools
		Conceptual Design
		Detailed Design for each UI Component
		Analyze and refine the low-fidelity UI prototype
		UI Design Specification & Project Style Guide
		Preliminary Requirements
		Analyze Requirements
		Construct a Navigational Prototype
		Commit and Baseline Preliminary Requirements
<b>Construction and Implementation</b>		Construct UI components
		Implement Change Management Process
		Testing

---

	Project Implementation and Maintenance

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	This design process focuses on UI design, requirements definition, and UI specification. We assume that a business case has been approved and that a Project Manager is assigned.	Assume that the following are available: <ul style="list-style-type: none"> <li>• Business Objectives</li> <li>• Business Case/Approval document</li> <li>• High-level user needs</li> <li>• Work requests</li> <li>• User Expectations</li> <li>• Project Constraints</li> </ul>				
P	<b>Project Planning Phase</b>					
A.1	UI Design Team - Roles and Responsibilities	Identify UI Design Team roles and responsibilities required for each phase of the product lifecycle. ** UI Team resources will vary based on the SDLC phase. <ol style="list-style-type: none"> <li>1. Define UI Design Team roles and responsibilities</li> <li>2. Identify which UI Team roles are required for each SDLC phase.</li> <li>3. Assign resources to each design team role.</li> </ol>	UI design team roles & responsibilities matrix (UMI)	UMI Roadmap workshop	<ul style="list-style-type: none"> <li>• Skill rqmts.</li> <li>• Available resources</li> <li>• Budget \$'s</li> </ul>	<ul style="list-style-type: none"> <li>• UI Design Team Roles and Responsibilities</li> <li>• Resource Assignments</li> <li>• Org Chart</li> <li>•</li> </ul>
	Build an initial project schedule and project plan					•
	<b>Perform On-Site Data Gathering and Analysis</b>					

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	Perform an On-site Analysis (Job, Work, User, Environment)	<ol style="list-style-type: none"> <li>1. Identify the objectives of the site visit, as well as issues and questions that need to be answered. What must be learned?</li> <li>2. Identify who will participate in site visits (design team, end-users, subject matter experts). Who will be interviewed and who will observe? How many people will participate?</li> <li>3. Identify sites to visit to ensure adequate representation</li> <li>4. Schedule the site visits</li> <li>5. Recruit users and participants for the site visit.</li> <li>6. Select appropriate methods for on-site data gathering</li> <li>7. Determine how data will be reported after site visits are complete.</li> <li>8. Build a plan to performing the on-site visit and analysis</li> <li>9. Build artifacts to be used during analysis:                             <ul style="list-style-type: none"> <li>• interview forms</li> <li>• questionnaires</li> </ul> </li> <li>10. Obtain video and audio equipment and lap-top computers required for site-visit</li> <li>11. Perform the on-site analysis using</li> </ol>	<ul style="list-style-type: none"> <li>• Contextual Inquiry (observe, listen, talk with users)</li> <li>• Interview users</li> <li>• Process Analysis (understand work flow)</li> <li>• Questionnaires</li> <li>• User surveys</li> <li>• Video tape</li> <li>• Photograph work environment</li> <li>• Lap-top computer to record events, observations, and notes while on-site.</li> </ul>		<ul style="list-style-type: none"> <li>• Initial knowledge of user groups</li> <li>• Business Objectives</li> <li>• High-level user needs</li> <li>• User Expectations</li> <li>• Project Constraints</li> </ul>	<ul style="list-style-type: none"> <li>• Affinity diagrams (summary of notes and observations)</li> <li>• List of Users and % of total user population observed</li> <li>• Process and workflow diagrams</li> <li>• Task lists</li> <li>• Video of the user site and users performing the tasks</li> <li>• Still photographs taken</li> <li>• List of Environments observed</li> <li>• Summary of interviews and QA's</li> <li>• Summary and analysis of user artifacts collected</li> </ul>

<b>User Interface Requirements, Design, and Specification Process</b>						
<b>Activity ID</b>	<b>UI Design Activity</b>	<b>Design Sub-activities</b>	<b>UCD Techniques &amp; Tools</b>	<b>UMI Deliverables</b>	<b>Input</b>	<b>Output</b>
		the selected techniques 12. Obtain user artifacts 13. Document and distribute findings				
	Identify and document High-Level Business Functions that must be supported by the system	1. Review available user/client artifacts (e.g., business objectives, user needs documents, constraints) to assemble high-level functions. 2. Define each high-level function (1-2 sentences) 3. Store High-level Functions document in the Requirements Management repository.	<ul style="list-style-type: none"> <li>• MS Word</li> <li>• RMT (Rqmts Mgmt Tool)</li> </ul>		<ul style="list-style-type: none"> <li>• Existing project artifacts</li> <li>• Business Objectives</li> <li>• High-level user needs</li> <li>• Work requests</li> <li>• User Expectations</li> <li>• Project Constraints</li> </ul>	<ul style="list-style-type: none"> <li>• List of High-level Business Functions with a brief description of each function.</li> </ul>
	Develop User Profiles	1. Identify user categories 2. Identify background information for each category 3. Provide a narrative description of each user category 4. Build a list of key attributes and findings for each user category. 5. Build a “character matrix” (sampling of users and key characteristics)	<ul style="list-style-type: none"> <li>• UMI User Profile data collection template</li> <li>• Photograph users in their work environment</li> </ul>	UMI User Profile deliverable	<ul style="list-style-type: none"> <li>• Results of on-site analysis</li> </ul>	<ul style="list-style-type: none"> <li>• Profile Description and picture for each User Category</li> <li>• Poster summary of all User Categories</li> <li>• Character Matrix</li> <li>• User/Task Matrix</li> </ul>
	Identify User Goals	1. Identify user goals 2. Contrast user goals with business objectives 3. Discover how users select tasks and	<ul style="list-style-type: none"> <li>• Interview</li> <li>• Questionnaires and surveys</li> </ul>		<ul style="list-style-type: none"> <li>• Results of on-site analysis</li> </ul>	<ul style="list-style-type: none"> <li>• List of User Goals</li> </ul>

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		<p>tools (within their available tool-set) to accomplish the <i>user</i> and <i>business</i> goal.</p> <p>4. Discover how users handle errors and how they impact goals.</p>	<ul style="list-style-type: none"> <li>• Focus Group Sessions</li> <li>• Direct observation</li> </ul>			
	Identify User Environment and Constraints	<p>1. Document a profile for each unique user environment.</p>	<ul style="list-style-type: none"> <li>• Photograph user environments</li> <li>• Narrative description</li> </ul>		<ul style="list-style-type: none"> <li>• Results of on-site analysis</li> </ul>	<ul style="list-style-type: none"> <li>• List of Environmental Constraints</li> </ul>
	<p>Develop a Context Diagram (high-level graphical representation of the proposed system)</p> <p>Identify boundaries, external and internal entities.</p> <p>Identify interactions; system initiated event, response to an external event.</p> <p>Interactions evolve into user and system tasks.</p>	<p>1. Identify organizations and users that will interface with the UI</p> <p>2. Identify external databases &amp; systems</p> <p>3. Identify external hardware devices</p> <p>4. Use a Context Diagram (CD) to represent all external entities. The CD should show the following:</p> <ul style="list-style-type: none"> <li>• single bubble representing the process, organization</li> <li>• terminators (users, organizational entity) that communicate directly with the system via data or control flows or data stores</li> <li>• data-flows (input and output of data)</li> <li>• control flows (signals received or generated by the system)</li> <li>• data stores (databases, external sources of data)</li> </ul>	<ul style="list-style-type: none"> <li>• Context Diagramming techniques</li> <li>• Diagramming Tool</li> <li>• MS PowerPoint</li> <li>• MS Word</li> </ul>		<ul style="list-style-type: none"> <li>• High level business functions</li> <li>• Description and Context Diagram of external systems (databases)</li> </ul>	<ul style="list-style-type: none"> <li>• Context Diagram</li> <li>• Internal and External Interactions</li> </ul>

<b>User Interface Requirements, Design, and Specification Process</b>						
<b>Activity ID</b>	<b>UI Design Activity</b>	<b>Design Sub-activities</b>	<b>UCD Techniques &amp; Tools</b>	<b>UMI Deliverables</b>	<b>Input</b>	<b>Output</b>
	<b>Create a Concepts Catalog</b>					
	Develop the initial Concepts Catalog	<ol style="list-style-type: none"> <li>1. Develop a repository of terms, concepts, and user artifacts</li> <li>2. Provide a description and/or definition in the context of the task and project.</li> </ol> <p>The Concepts Catalog is refined throughout the project lifecycle.</p> <p>The Concepts Catalog should be written such that any project team member can obtain a context specific description of a term or concept employed in the design or that is significant within the user's domain.</p>	<ul style="list-style-type: none"> <li>• MS Word</li> </ul>		<ul style="list-style-type: none"> <li>• Existing project artifacts</li> <li>• High-level user needs</li> <li>• Results of on-site analysis</li> </ul>	<ul style="list-style-type: none"> <li>• Concepts Catalog</li> </ul>
	<b>Identify Initial User Needs and User Acceptance Criteria</b>					
	Perform a high-level Task Analysis to Identify User Needs  Produce initial set of Task Descriptions	<ol style="list-style-type: none"> <li>1. Select representative users and subject matter experts to participate in the Needs gathering sessions.</li> <li>2. Identify initial set of tasks to be performed (current and proposed). Document each task separately. These tasks will evolve throughout the Planning, Analysis, and Design</li> </ol>	<ul style="list-style-type: none"> <li>• Facilitated sessions with selected user representatives to identify "as is" and proposed user needs</li> <li>• Interview and/or</li> </ul>		<ul style="list-style-type: none"> <li>• Business Objectives</li> <li>• High-level user needs</li> <li>• User Expectations</li> <li>• High-level</li> </ul>	<ul style="list-style-type: none"> <li>• List of User Needs with description</li> <li>• Set of Initial Tasks with description and task characteristics</li> <li>•</li> </ul>

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		<p>phases.</p> <ul style="list-style-type: none"> <li>• brief task description (overview of what happens, information contained within the task, choices that can be taken within the task, changes in task state)</li> <li>• when (under what conditions) does the task occur</li> <li>• who initiates the task (user or system)</li> <li>• what is the expected output or result from the task</li> <li>• criticality of task</li> <li>• frequency task is performed</li> <li>• complexity and difficulty of task</li> <li>• requirements for task performance</li> <li>• tolerance of the task to error (impact, ability to recover, typical errors)</li> <li>• information/data needed to perform the task</li> </ul> <p>3. Identify known issues with functionality, process flow, internal or external interfaces, performance, usability, capacity, maintainability, etc.</p> <p>4. Identify known process and workflow exceptions that must be handled by the design.</p>	<p>direct observation of users, domain experts, management not covered in the on-site analysis</p> <ul style="list-style-type: none"> <li>• Questionnaires</li> <li>• Surveys</li> <li>• User artifact review</li> <li>• MS Word</li> <li>• Requirements Management Tool</li> </ul>		<p>business functions w/brief description of each function.</p> <ul style="list-style-type: none"> <li>• Results of all on-site analysis (process and workflow diagrams, task lists, affinity diagrams)</li> <li>• User Profile</li> <li>•</li> </ul>	



User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		5. Identify known constraints and assumptions for each of these tasks 6. Ensure that all internal & external entities described within the Context Diagram are supported by one or more User Needs. 7. Review the following for completeness of requirements: <ul style="list-style-type: none"> <li>• ensure that all components in the Context Diagram are supported</li> <li>• ensure that the needs of each user category (user profile) are supported</li> <li>• ensure that all on-line and batch processing is described</li> <li>• ensure that all data-processing (inputs, outputs, reference) is described</li> <li>• ensure that report (on-line, paper) information, format, frequency, and expected volume are described.</li> <li>• describe expectations for error handling, include formats and messages for typical errors</li> <li>• ensure that requirements for data retention, and access of historical data is described.</li> </ul>				
	Define Acceptance	1. Identify the criteria against which	• Facilitated sessions		• List of User	• Acceptance

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	Criteria for User Needs	<p>the User Need can be verified. These criteria are used to ensure that the User Needs are met by the system design</p> <p>2. Identify measurable and accurate quantities for each criteria. What measurements are acceptable?</p>	<p>with selected user representatives to identify acceptance criteria for each user need</p> <ul style="list-style-type: none"> <li>• Questionnaires</li> <li>• Surveys</li> </ul>		<p>Needs with description</p> <ul style="list-style-type: none"> <li>• Set of Initial Tasks with description and task characteristics</li> </ul>	Criteria for each User Need
	<p>Document User Needs and Acceptance Criteria</p> <p>Specify <u>usability</u> attributes for each User Need. Quantify and measure attribute in Acceptance Criteria.</p>	<p>1. Document User Needs, provide the following attributes:</p> <ul style="list-style-type: none"> <li>• ID (define a formal scheme for numbering/identifying each user need)</li> <li>• name</li> <li>• exact statement of user need</li> <li>• description of user need</li> <li>• component (system, functional)</li> <li>• dependencies</li> <li>• constraints</li> <li>• user acceptance criteria and metrics</li> <li>• source of user need</li> <li>• date user need identified</li> <li>• priority of user need</li> </ul> <p>2. Capture a usability attribute for each user need where possible. Specify acceptable values for the usability attributes in the Acceptance Criteria.</p>	<ul style="list-style-type: none"> <li>• Requirements Management Tool (RMT)</li> <li>• MS Excel</li> <li>• MS Word</li> </ul>		<ul style="list-style-type: none"> <li>• List of User Needs with description</li> <li>• Acceptance Criteria for each User Need</li> </ul>	<ul style="list-style-type: none"> <li>• Documented set of concise User Needs and measurable User Acceptance Criteria for each user need</li> </ul>

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		<ol style="list-style-type: none"> <li>Document an Acceptance Criteria for each requirement. How will you verify that the requirement has been met by the system/product?                             <ul style="list-style-type: none"> <li>define the range of acceptable values (minimum, maximum, expected)</li> </ul> </li> <li>Record user needs and acceptance criteria in the Requirements Management tool repository.</li> </ol>				
	Build a Traceability Matrix (User Needs to Business Objectives and High-level Functions)	<ol style="list-style-type: none"> <li>Build a User Needs to High-Level Functions Traceability Matrix:                             <ul style="list-style-type: none"> <li>ensure that all High-level functions are supported by one or more User Needs</li> <li>ensure that all User Needs are supported by one or more High-level functions</li> </ul> </li> <li>Build a User Needs to Business Objectives Traceability Matrix:                             <ul style="list-style-type: none"> <li>ensure that all Business Objectives are supported by one or more User Needs</li> <li>ensure that all User Needs are supported by one or more Business Objectives</li> </ul> </li> </ol>	<ul style="list-style-type: none"> <li>User Need to Business Objectives Matrix template</li> <li>User Need to High Level function Matrix template</li> <li>MS Excel</li> <li>Requirements Management Tool (RMT)</li> </ul>		Documented set of concise User Needs and measurable User Acceptance Criteria for each user need	<ul style="list-style-type: none"> <li>User Need → High-level function Traceability Matrix</li> <li>User Need → Business Objective Traceability Matrix</li> </ul>
	Refine User Needs	<ol style="list-style-type: none"> <li>Review User Needs for ambiguous terminology and phrasing. User</li> </ol>	<ul style="list-style-type: none"> <li>Team Review</li> </ul>		User Needs document, User	Refined User Needs document, User

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		<p>Needs should be consistent and exact.</p> <p>2. Review User Acceptance Criteria for each user need.</p> <ul style="list-style-type: none"> <li>• ensure that acceptance criteria has been defined for each User Need</li> <li>• ensure that acceptance criteria is measurable and sufficient</li> </ul>			Acceptance Criteria document, and Traceability Matrixes	Acceptance Criteria document, and Traceability Matrixes
	<p>Perform User/Client Review of User Needs and User Acceptance Criteria</p> <p>Modify User Needs and Acceptance Criteria based on Feedback</p>	<p>1. Distribute requirements documents to the user/client for review</p> <p>2. Review User Needs and related documents (User Acceptance Criteria, business objectives, high-level functions, external entities diagram, project constraints, project assumptions).</p> <p>3. Modify User Needs documents based on user/client feedback</p>	<ul style="list-style-type: none"> <li>• User Walkthrough</li> <li>• Facilitated review</li> </ul>		Refined User Needs document, User Acceptance Criteria document, and Traceability Matrixes	<p>Documents that have passed user/project team review:</p> <ul style="list-style-type: none"> <li>• User Needs</li> <li>• User Acceptance Criteria</li> <li>• Business objectives</li> <li>• High-level functions</li> <li>• Context Diagram.</li> </ul>
	<p>Initiate an Issues Log; Maintain issues log throughout project life</p>	<p>1. Maintain an ongoing log of Issues</p> <ul style="list-style-type: none"> <li>• Issue ID, name</li> <li>• Issue description</li> <li>• Project documents affected (e.g., business objectives, user needs)</li> <li>• Date identified</li> <li>• Status of issue (open, closed)</li> <li>• Responsible team member to</li> </ul>	<ul style="list-style-type: none"> <li>• Issues Repository</li> <li>• Requirements Management Tool</li> <li>• MS Excel</li> <li>• MS Word</li> <li>• Group facilitation</li> <li>• Issue reporting</li> </ul>			Issues Log

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		resolve <ul style="list-style-type: none"> <li>• Date issue resolved</li> <li>• Comments</li> </ul> 2. Review, update, and resolve open issues 3. Log additional issues	process <ul style="list-style-type: none"> <li>• Issue tracking form</li> <li>• Project Management procedures</li> </ul>			
	<b>Identify “Non-Functional” and “Non-Technical” User Needs</b>					
	Identify initial “Non-functional” and “Non-technical” User Needs  General product Usability Goals/Objectives are defined as part of the Non-functional User Needs. Specific usability attributes are also associated with each User Need and Functional Requirement.	1. Select representative users and subject matter experts to participate in facilitated sessions to identify non-functional and non-technical user needs (note that the project team and client representatives are key participants in this activity). 2. Identify Non-functional user needs (usability objectives, performance, availability, maintainability) <ul style="list-style-type: none"> <li>• Identify <u>usability objectives</u></li> <li>• Identify measurable and verifiable metrics for each usability objective(values for current, planned, optimistic)</li> </ul> 3. Identify Non-technical user needs (documentation products, training products, project dates/milestones)	<ul style="list-style-type: none"> <li>• Facilitated sessions</li> <li>• Interviews</li> <li>• Questionnaires</li> <li>• Surveys</li> <li>• Direct observation</li> <li>• Artifact review</li> </ul>	UMI Usability Objectives deliverable	<ul style="list-style-type: none"> <li>• User Profile</li> <li>• User Expectations</li> <li>• User Needs</li> <li>• Business Objectives</li> <li>• Environmental Profile</li> </ul>	<ul style="list-style-type: none"> <li>• Preliminary Non-functional User Needs</li> <li>• Preliminary Non-technical User Needs</li> <li>• List of Usability Objectives</li> <li>• Measurement values for each Usability Objective</li> </ul>

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		4. Ensure that the following systems characteristics are described and supported sufficiently in the non-functional and non-technical user needs: <ul style="list-style-type: none"> <li>• performance, response time, bandwidth</li> <li>• capacity (maximum number of users, transactions, storage)</li> <li>• availability (24x7, time zones supported)</li> <li>• reliability (tolerance to defects, downtime, data loss, communications failure)</li> <li>• accuracy</li> <li>• security</li> <li>• disaster recovery</li> <li>• flexibility (ability to respond to changes in the business)</li> <li>• expected lifetime of system</li> <li>• reusability of existing components</li> <li>• integration of third-party components</li> <li>• backup and recovery expectations</li> <li>• portability, upward compatibility</li> <li>• internationalization</li> </ul>				

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	<p>Define Acceptance Criteria for Non-functional and Non-technical User Needs</p> <p>Usability Measurements for each usability objective and attribute.</p>	<ol style="list-style-type: none"> <li>1. Identify the criteria against which Non-functional and Non-technical User Needs can be verified. These criteria are used to ensure that the User Needs are met by the system design</li> <li>2. Identify measurable and accurate quantities for each criteria. What measurements are acceptable?</li> <li>3. Note: Some Non-technical and Non-functional user needs may not have measurable acceptance criteria.</li> </ol>	<ul style="list-style-type: none"> <li>• Facilitated sessions with selected user representatives to identify acceptance criteria for each non-functional and non-technical user need</li> <li>• Questionnaires</li> <li>• Surveys</li> </ul>		<ul style="list-style-type: none"> <li>• Preliminary Non-functional User Needs</li> <li>• Preliminary Non-technical User Needs</li> <li>• List of Usability Objectives</li> <li>• Measurement values for each Usability Objective</li> </ul>	<ul style="list-style-type: none"> <li>• Acceptance Criteria for appropriate Non-functional and Non-technical User Needs.</li> </ul>
	Refine Non-functional and Non-technical User Needs and User Acceptance Criteria	<ol style="list-style-type: none"> <li>1. Review Non-functional and non-technical requirements for ambiguous wording</li> <li>2. Review User Acceptance Criteria (measurements) for each Non-functional and non-technical requirement.</li> </ol>				
	Build a Traceability Matrix (Non-functional and Non-technical User Needs → Business objectives)	<ol style="list-style-type: none"> <li>1. Ensure that all Nonfunctional and nontechnical requirements are within scope and map back to a valid business objective.</li> <li>2. Build a traceability matrix</li> </ol>	<ul style="list-style-type: none"> <li>• Traceability matrix template for Non-functional and Non-technical User Needs → Business Objectives</li> </ul>			<ul style="list-style-type: none"> <li>• Non-functional User Needs → Business Objectives Traceability Matrix</li> <li>• Non-technical User Needs →</li> </ul>

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
						Business Objectives Traceability Matrix
	<b>Develop Initial “Cross-Component” Usage Scenarios</b>					
	Build high-level usage scenarios (description of how user goals will be performed using the new UI design)	<ol style="list-style-type: none"> <li>1. Review the user actions and decisions documented in the task scenarios (description of how the existing product is used to perform a task)</li> <li>2. Consider user goals and how users currently perform their work.</li> <li>3. Consider changes to the constraints and user environment as a result of the new UI design.</li> <li>4. Identify the steps needed to complete the each task using the new UI design.</li> <li>5. Document a narrative of the user performing the task using the new UI design.</li> </ol>			• Task Flows	• Usage Scenarios
	<b>Identify Production Platform(s) for UI Deployment</b>					
	Identify the production UI/Presentation Layer platform(s)	<ol style="list-style-type: none"> <li>1. Identify the operating system and hardware for the target UI/Presentation layer platform</li> </ol>	• Results from site visits			UI deployment platform (OS and hardware)



User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		2. If multiple platforms are selected (e.g., Windows, Web, Motif, Macintosh), identify known cross platform issues.	<ul style="list-style-type: none"> <li>• User Profile analysis</li> <li>• Environment analysis</li> <li>• User goals analysis</li> <li>• Business objectives</li> <li>• User Needs</li> </ul>			
	Identify the UI development and prototyping tools	<ol style="list-style-type: none"> <li>1. Identify the principle development platform to build the UI for the target platform(s).</li> <li>2. Identify other development tools and libraries needed.</li> <li>3. Identify unique skill sets required for development and advise project manager.</li> </ol>				UI development tools and platform
	<b>Refine the Project Schedule and Develop the Usability Test Plan</b>					
	Develop an Initial Usability Test Plan	<ol style="list-style-type: none"> <li>1. Identify a standard format to be used for the Usability Test Plan Document (test purpose, objectives, test method, task list, user profile, test environment, data collection) and results Report.</li> <li>2. Identify what will be tested, the number of testing iterations, and when each test will be performed.</li> </ol>	UMI Usability Test Plan Template	UMI Roadmap Workshop		Initial Usability Test Plan

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		<ol style="list-style-type: none"> <li>3. Identify the project team's expectations and goals for each usability test.</li> <li>4. Develop a preliminary schedule each planned usability test and include in the Project Schedule.</li> <li>5. Identify the following resource and schedule constraints in the Test Plan. Obtain commitment and buy-in where needed to obtain resource and time commitments:                             <ul style="list-style-type: none"> <li>• budget</li> <li>• lab facilities</li> <li>• usability team resources</li> <li>• project schedule impact</li> </ul> </li> <li>6. Document an initial test plan that includes each planned test. Plan this document changing as your design becomes better defined.</li> </ol>				
	Integrate UI design and usability testing activities into project schedule.	<ol style="list-style-type: none"> <li>1. Include UI design activities in the Project Plan and schedule</li> <li>2. Plan for an iterative design cycle. <u>You must decide upfront how many iterations will be performed and schedule for each iteration.</u> Weigh cost/schedule against improved design and efficiency<sup>1</sup></li> </ol>	MS Project			Project Schedule

<sup>1</sup> *Website Design from the Trenches*, Tom Brinck, Darren Gergle, Scott Wood, ACM SIGCHI 97 Tutorial Notes.

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		<ol style="list-style-type: none"> <li>3. Assign UI design activities included in this process as project milestones</li> <li>4. Identify well understood deliverables</li> <li>5. Establish exit criteria for each milestone deliverable.</li> </ol>				
	<b>Perform High-Level Conceptual UI Design</b>					
	Perform a "Current" Workflow Analysis - Scope is the job and/or set of tasks for which the product is being designed	<ol style="list-style-type: none"> <li>1. Identify how the work is currently performed (moderate amount of detail).                             <ul style="list-style-type: none"> <li>• Identify each person involved in performing the work.</li> <li>• Identify the Triggers (inputs), Process, and Results of the workflow.</li> </ul> </li> <li>2. Develop a task flow that represents the "Current Work" performed. Task steps and branches are recorded on separate index cards. Index cards are taped to a large blank sheet of paper. "Sticky arrows" are used to connect the cards together and represent flow.</li> <li>3. Ensure that workflow remains within scope.</li> </ol>	<ul style="list-style-type: none"> <li>• Participatory design session where tasks are modeled using index cards. Indicate flow between task cards.</li> </ul>	UMI Framework Design Session - Part 1	<ul style="list-style-type: none"> <li>• Set of Initial Tasks with description and task characteristics</li> <li>• Documented set of concise User Needs and measurable User Acceptance Criteria for each user need</li> <li>• User Profile</li> <li>• List of Usability Objectives</li> <li>• Environmental Profile</li> <li>• Project Constraints</li> </ul>	<ul style="list-style-type: none"> <li>• High-level Task flow that represents the Current Workflow</li> </ul>

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	Perform a Job Analysis - considers all jobs/tasks performed by the user (optional)					
	Identify Problems and Bottlenecks in the Current Workflow - User Perspective (Scope is the job/task for which the product is being designed)	<ol style="list-style-type: none"> <li>Briefly describe each problem &amp; bottleneck</li> <li>Attach stickies with Problems &amp; Bottlenecks to the appropriate index card in the "Current Workflow"</li> <li>Prioritize problems &amp; bottlenecks</li> </ol>		UMI Framework Design Session - Part 1		Current workflow with attached problem stickies.
	Perform a "Blue Sky" Redesign of the Current Workflow to Produce the "Ideal" Workflow	<ol style="list-style-type: none"> <li>Based on the "Blue Sky" redesign, build the Ideal task flow (moderate detail) that resolves the problems &amp; bottlenecks identified. This is a "think-out-of-the-box" brainstorming session.</li> <li>Record issues or concerns on a sticky and attach to the back of the index card.</li> <li>For each task index card in the Blue Sky/Ideal flow, identify the following: <ul style="list-style-type: none"> <li>User Priority (high, medium, low)</li> <li>Difficulty to Implement (easy, moderate, hard)</li> </ul> </li> </ol>	Index cards to represent individual tasks Arrows to show task flow.	UMI Framework Design Session - Part 1		Ideal Task Flow with User and Developer priorities and Issues recorded.
	Create the Realistic Task Flow	<ol style="list-style-type: none"> <li>Redesign the Ideal task flow (moderate level of detail) to be</li> </ol>	Index cards to represent individual	UMI Frame-	Ideal task flow	Realistic Task Flow that will become the

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	(Scope is the job/task for which the product is being designed)	<p>attainable within given technology, constraints, budget, and schedule.</p> <ol style="list-style-type: none"> <li>2. Ensure that each task card has a Noun and a Verb</li> <li>3. Use arrows to designate task flow.</li> <li>4. Show triggers and results.</li> </ol>	<p>tasks</p> <p>Arrows to show task flow.</p>	<p>work</p> <p>Design Session - Part 1</p>		<p>basis for UI design</p>
	Identify Task Objects (nouns, verbs, attributes)	<ol style="list-style-type: none"> <li>1. Copy each Noun from the Realistic Task Flow onto a separate index card. Provide a concise description of the Noun. Each index card is a Task Object "candidate".</li> <li>2. Identify the attributes for each Task Object candidate. If the object has no attributes, it is most likely an attribute itself and "not" a Task Object.</li> <li>3. Identify the actions for each Task Object. Ensure that the Verbs from the Realistic Task Flow are captured. Also consider generic actions such as Print, Create, Edit, Delete, Save, etc.</li> <li>4. Task Object attributes can be either Properties or Child objects. Identify attributes that are actually child objects (have their own index card).</li> </ol>	<p>Index cards to represent a Task Object.</p> <p>Stickies are attached to the index cards to record attributes, and actions.</p>	<p>Framework Design Session - Part 2</p>		<p>Task Objects</p>
	Identify Hierarchy and	<ol style="list-style-type: none"> <li>1. Build an initial object hierarchy by</li> </ol>		<p>Framework</p>		<p>Task Object</p>

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	Relationships between Task Objects (represents the user's existing mental model of the relationships between these objects)	defining the parent/child containment relationships between the Task Objects. 2. Refine the task object containment model to ensure that object hierarchy supports the user tasks in the most efficient manner.		k Design Session - Part 2		Hierarchy with parent / child relationships
	Perform task walkthroughs and informal usability testing on Task Objects and the Hierarchy.	1. Verify that the Task Objects are sufficient to perform the user tasks and are optimized to perform these tasks efficiently. 2. Perform the walkthroughs without knowledge of the UI itself. 3. Refine the task object hierarchy, add container objects to better organize the task objects, and modify the task objects themselves based on the user walkthroughs and testing.		Framework Design Session - Part 2	Task Objects and Object hierarchy	Refined Task Objects and hierarchy
	Identify the principle components of the UI. These components should map directly to areas of product functionality and to the Task Objects.	1. Using the Task Objects as input, identify the principle components of the UI. These areas represent the major areas of functionality that can be designed in parallel.		Framework Design Session - Part 2	Task objects	
	If the UI is "non-Object Oriented" – Build the	1. Develop a UI Metaphor (optional) 2. Using the task object hierarchy as a	<ul style="list-style-type: none"> <li>Task Hierarchy</li> </ul>		<ul style="list-style-type: none"> <li>Task Objects</li> <li>User Profile</li> </ul>	<ul style="list-style-type: none"> <li>UI Conceptual Model</li> </ul>

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	initial Conceptual Model of UI	basis, define a initial menu structure and navigational model 3. Identify basic Interaction model 4. Select UI controls that best represent the data, task, interaction model, and navigational model			<ul style="list-style-type: none"> <li>• Usability Goals</li> <li>• Environment Profile</li> </ul>	<ul style="list-style-type: none"> <li>• Metaphor</li> </ul>
	If the UI is “Object Oriented”, use the OO UI Design Framework (menu structure, windows, control buttons)		<ul style="list-style-type: none"> <li>• OO UI Design Framework<sup>2</sup></li> </ul>	Framework Design Session - Part 3		
	Build a high-level, low-fidelity prototype of the UI Conceptual Model that supports the task objects and principle UI components.  (Scope is the job/task for which the product is being designed)	1. Build a high-level, low fidelity prototype (refer to PICTIVE <sup>3</sup> , paper and pencil techniques) of the Conceptual UI design that includes the following: <ul style="list-style-type: none"> <li>• Metaphor design</li> <li>• Navigational model to support task flow</li> <li>• Interaction model</li> <li>• Menu structure</li> <li>• Principle UI controls to represent data and control navigation/interaction</li> <li>• High-level primary and</li> </ul>	<ul style="list-style-type: none"> <li>• Paper and pencil, stickies, tape, markers</li> </ul>	Framework Design Session - Part 3	Task Flows	<ul style="list-style-type: none"> <li>• Primary and secondary windows</li> <li>• Task object views</li> </ul>

<sup>2</sup> From *User Needs to Graphical User Interface Design: A Participatory Approach*, Version 6/7/96, Tom Dayton, Al McFarland, Joseph Kramer

<sup>3</sup> *PICTIVE - An Exploration in Participatory Design*, Muller, M.J., 1991, *CHI'91 Conference Proceedings*, New Orleans, LA: ACM, 225-231

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		secondary window designs • High-level dialog designs				
	Perform walkthroughs and informal usability testing on the low-fidelity window prototypes. Modify design based on user feedback.			Framework Design Session - Part 3		
	Build a Storyboard that contains the low-fidelity window prototypes of the UI Conceptual Model.	<ol style="list-style-type: none"> <li>1. Build a Storyboard using the low-fidelity window prototypes of the UI Conceptual Model.</li> <li>2. Demonstrate the navigation and flow between windows.</li> <li>3. Post Storyboard in a public location and establish an informal feedback channel. Post subsequent Storyboards in this same location.</li> </ol>				
	<b>Develop Initial Documentation of the User Interface</b>					
	Initiate the project UI Style Guide - Phase 1 (document lives throughout the entire project lifecycle)  Style guide is essential to ensure design	<ol style="list-style-type: none"> <li>1. Build a table of contents based on the UMI Style Guide template.</li> <li>2. If project is a part of a product suite, integrate look-and-feel conventions of the product suite into project level style guide.</li> <li>3. If there is a Corporate Style Guide, integrate look-and-feel conventions</li> </ol>	<ul style="list-style-type: none"> <li>• UMI Style Guide template</li> <li>• MS Word</li> <li>• Visio</li> <li>• MS PowerPoint</li> </ul>	UMI Style Guide deliverable		Project UI Style Guide - Phase 1



User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	consistency and that the original design principles are implemented throughout the <i>maintenance</i> phase of a project. Typically the original design team does not maintain the product. Usability engineers are typically not funded during maintenance.	<p>as appropriate into the project level style guide.</p> <ol style="list-style-type: none"> <li>4. Document the following: <ul style="list-style-type: none"> <li>• Overview of the Metaphor, if implemented</li> <li>• Window/page design templates</li> <li>• Overview of the Navigational model</li> <li>• Overview of Interaction model</li> <li>• Menu structure - use a drawing tool to show the pull-down menu titles and selections.</li> <li>• Common command buttons</li> <li>• Preferred UI controls used to represent data and control navigation/interaction</li> <li>• Terminology specific to the project</li> <li>• Project and corporate signatures, logos, images</li> </ul> </li> <li>5. Circulate the draft style guide for review by project team members</li> <li>6. Refine the style guide based on team feedback</li> <li>7. Assign a team member to manage updates/revisions and distribution of the style guide throughout the project life cycle.</li> </ol>				
	Initiate the UI	1. Start with the UI Specification	• UMI UI	UMI UI		• UI Specification -

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	<p>Specification document. Include high-level conceptual design, metaphor design, and any other system and usability specifications.</p> <p>Essential for subsequent product upgrades and as a basis for product redesign.</p>	<p>template</p> <ol style="list-style-type: none"> <li>2. Describe UI/Presentation layer production and development platforms.</li> <li>3. Define requirements for cross-platform compatibility (if applicable)</li> <li>4. Document the UI metaphor, if implemented</li> <li>5. Document the Task Flows (Current, Ideal, Realistic) from the Framework Design Session Part 1</li> <li>6. Document the Task objects from the Framework Design Session Part 2. Include the object name, description, attributes (properties, and child objects), and actions.</li> <li>7. Document the menu structure (commands performed by each menu selection, accelerators, mnemonics)</li> <li>8. Use a drawing tool to build the primary and secondary windows and dialogs from the Framework Design Session Part 3. Include all known design detail.</li> </ol>	<p>Specification template</p> <ul style="list-style-type: none"> <li>• MS Word</li> <li>• Visio</li> <li>• MS Visual Basic</li> <li>• MS PowerPoint</li> </ul>	<p>Spec deliverable</p>		<p>Phase 1</p>
	<p><b>Refine “Cross-Component” Usage Scenarios across UI</b></p>					

<b>User Interface Requirements, Design, and Specification Process</b>						
<b>Activity ID</b>	<b>UI Design Activity</b>	<b>Design Sub-activities</b>	<b>UCD Techniques &amp; Tools</b>	<b>UMI Deliverables</b>	<b>Input</b>	<b>Output</b>
	<b>Components</b>					
	Refine Usage Scenarios based on Conceptual UI Design	<ol style="list-style-type: none"> <li>1. Review the Initial Usage Scenarios</li> <li>2. Identify gaps in the Usage Scenarios based on the “to be” Realistic Task Flow and Conceptual UI Design prototype</li> <li>3. Identify additional task, user, and environmental detail that should be included in the Usage Scenarios.</li> <li>4. Update the Usage Scenarios as appropriate</li> </ol>			<ul style="list-style-type: none"> <li>• Initial User Scenarios</li> <li>• Realistic “to be” Task Flows</li> <li>• Conceptual UI Design Prototype</li> </ul>	<ul style="list-style-type: none"> <li>• Refined Usage Scenarios</li> </ul>
	<b>Refine Functional, Non-functional, and Non-technical User Needs, User Acceptance Criteria, and Traceability Matrix</b>					
	Refine all User Needs (Functional, Non-functional and Non-technical) and User Acceptance Criteria based on task analysis results and the UI Conceptual Design Prototype	<ol style="list-style-type: none"> <li>1. Update all User Needs (Functional, Non-functional, and Non-technical) based on the task analysis and UI design prototypes.</li> <li>2. Review all User Needs (Functional, Non-functional, and Non-technical) for ambiguous wording</li> <li>3. Review User Acceptance Criteria (measurements) for each User Need (Functional, Non-functional, and Non-technical). Note: some Non-functional and Non-technical user</li> </ol>		Usability Objectives		<ul style="list-style-type: none"> <li>• Preliminary Non-functional and Nontechnical requirements</li> <li>• Usability Objectives</li> </ul>

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		needs may not have an acceptance criteria.				
	Refine the Traceability Matrix (Nonfunctional and Non-technical requirements - Business objectives)	<ol style="list-style-type: none"> <li>1. Ensure that all Nonfunctional and nontechnical requirements are within scope and map back to a valid business objective.</li> <li>2. Use a Requirements Management Tool for traceability if available.</li> </ol>	<ul style="list-style-type: none"> <li>• Traceability matrix - Non-Functional and Non-Technical Requirement to Business Objectives</li> <li>• Requirements Management Tool</li> </ul>			
	<b>Document User Needs, User Acceptance Criteria, and Traceability Matrices in a Formal Repository</b>					
	Document Refined Functional User Needs and User Acceptance Criteria in a Formal Repository	<ol style="list-style-type: none"> <li>1. Enter Functional User Needs and Acceptance Criteria into a Requirements database (repository)</li> <li>2. Identify dependent Functional User Needs and record in the Requirements database.</li> <li>3. Record notes or additional information associated with each Functional User Need.</li> </ol>	<ul style="list-style-type: none"> <li>• Requirements Management Tool</li> <li>• MS Excel</li> </ul>			
	Document Refined Non-Functional and Non-Technical User Needs and User Acceptance	<ol style="list-style-type: none"> <li>1. Enter Nonfunctional and non-technical User Needs into a Requirements database (repository)</li> <li>2. Identify dependent Nonfunctional</li> </ol>	<ul style="list-style-type: none"> <li>• Requirements Management Tool</li> <li>• MS Excel</li> </ul>			

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	Criteria in a Formal Repository.	and non-technical User Needs and record in the Requirements database. 3. Record notes or additional information associated with each Non-functional and non-technical User Need.				
	Document the Refined Traceability Matrix (User Needs → Business Objectives) in a Formal Repository	1. Ensure that each User Need supports a business objective 2. Ensure that Business Objectives are supported by the appropriate User Need. 3. Use a Requirements Management Tool for traceability if available.	• Requirements Management Tool			
	<b>Perform a User/Client Review of Refined User Needs and Acceptance Criteria</b>					
	Perform a User/Client review of Functional, Non-Functional, and Non-Technical User Needs and User Acceptance Criteria	1. Distribute User Needs and Acceptance Criteria to User/Client representatives for review 2. Facilitate a group session with User/Client to identify any issues and obtain buy-in.	• Facilitated session			
	Review Issues	1. Identify, review, and resolve issues 2. Update the Issue Log				

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	Modify User Needs and User Acceptance Criteria as a result of user review.	<ol style="list-style-type: none"> <li>1. Update User Needs to reflect agreed upon changes as a result of the user/client review.</li> <li>2. Update task diagrams, UI prototypes, and all other analysis documents to reflect changes in User Needs.</li> </ol> <p><i>Every project must implement a document management process, provide a common repository for project documents, implement a consistent revision/release mechanism, and control who has access and when documents are updated.</i></p>				
	<b>Elevate User Needs and Acceptance Criteria to Preliminary Requirements once Approved by Client/User</b>	<ol style="list-style-type: none"> <li>1. If User Needs are approved by the Client/User, elevate the status of User Needs to Preliminary Functional, Preliminary Non-Functional, and Preliminary Non-Technical Requirements</li> <li>2. Change the status within the Requirements Management Tool</li> </ol> <p><i>Note that Preliminary Requirements are not Final or Committed and thus are not under revision control.</i></p>				
	<b>Analyze Preliminary</b>					

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	<b>Requirements</b>					
	Perform Process, Data, and System Analysis of the Preliminary Requirements	<ol style="list-style-type: none"> <li>Analyze and model the preliminary requirements using the techniques listed to better understand the requirements and identify the gaps.</li> <li>For each of the diagrams and flowcharts produced, provide a brief description of the data flows, data stores, data attributes, and processing.</li> </ol>	<ul style="list-style-type: none"> <li>Process Analysis</li> <li>Data Flow Diagramming</li> <li>Flow Charting</li> <li>I/O Diagramming</li> <li>Data Modeling</li> </ul>			
	Evaluate Quality of Preliminary Requirements	<ol style="list-style-type: none"> <li>Evaluate the following to ensure that requirements are:                             <ul style="list-style-type: none"> <li>complete</li> <li>accurate</li> <li>consistent</li> <li>feasible from a technical, economics, marketing and business perspective</li> <li>verifiable</li> <li>traceable</li> </ul> </li> </ol>				
	Evaluate Feasibility of Preliminary Requirements and User Acceptance Criteria	<ol style="list-style-type: none"> <li>Review all requirements documents</li> <li>Evaluate Economic feasibility</li> <li>Evaluate Technical feasibility</li> <li>Evaluate Operational feasibility</li> </ol>				
	Refine and Update Preliminary Requirements and User					Updated Preliminary Requirements Document

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	Acceptance Criteria based on Analysis, Feasibility Study, and Quality check					
	Update the Traceability Matrix (Preliminary Requirements → Business Objectives)	<ol style="list-style-type: none"> <li>1. Ensure that each Preliminary Requirement supports a business objective</li> <li>2. Ensure that Business Objectives are supported by the appropriate Preliminary Requirements.</li> <li>3. Use a Requirements Management Tool for traceability if available.</li> </ol>	<ul style="list-style-type: none"> <li>• Requirements Management Tool</li> <li>• Excel</li> </ul>			Updated Traceability Matrix
	Perform a User/Client review of Preliminary Requirements	<ol style="list-style-type: none"> <li>4. Distribute Preliminary Requirements to User/Client representatives for review</li> <li>5. Facilitate a group session with User/Client to identify any issues and obtain buy-in.</li> </ol>				
	Identify, review, and resolve issues; Update the Issue Log	<ol style="list-style-type: none"> <li>1. Log any issues that require further investigation or that must be resolved.</li> <li>2. Review, update, and resolve open issues</li> <li>3. Update Issues Log</li> </ol>				
	Modify Preliminary Requirements and User Acceptance Criteria as a	<ol style="list-style-type: none"> <li>1. Update Preliminary Requirements to reflect agreed upon changes as a result of the user/client review.</li> </ol>				



User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	result of user review.	2. Update task diagrams, UI prototypes, and all other analysis documents to reflect changes in Preliminary Requirements.				
	<b>Perform Detailed Design for each UI Component</b>					
	<p>For each UI component perform detailed task analysis (level of detail needed for UI design).</p> <p>Focus on “what” is performed, not “how” it’s performed</p> <p>Detailed task analysis should be performed prior to detailed UI design. Detailed task analysis for multiple UI components may occur in parallel.</p>	<ol style="list-style-type: none"> <li>1. Perform task analysis using the techniques listed to identify the business rules, data, processing, data/control flow needed for detailed UI design. <i>Typically, the UI does not require as much detail as the back-end system processes.</i></li> <li>2. Refine Task Descriptions. Ensure the following is documented for each task: <ul style="list-style-type: none"> <li>• task name</li> <li>• task description</li> <li>• dependent and associated tasks</li> <li>• business rules required for task</li> <li>• data required for task (information to be displayed/entered, selections and choices)</li> <li>• describe all actions and commands that can be invoked by the task</li> <li>• describe all paths of execution</li> </ul> </li> </ol>	<ul style="list-style-type: none"> <li>• UMI Task Description Template</li> </ul>	<ul style="list-style-type: none"> <li>• Task Lists</li> <li>• Task descriptions</li> </ul>	<ul style="list-style-type: none"> <li>• Detailed Task Description for each task.</li> </ul>	

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		for the task				
	Refine Usage Scenarios to reflect detailed task analysis for the UI component.	<ol style="list-style-type: none"> <li>1. Review the Usage Scenarios</li> <li>2. Identify gaps in the Usage Scenarios based on the detailed UI Component design and refined "to be" Realistic Task Flow.</li> <li>3. Identify additional tasks, user, and environmental detail that should be included in the Usage Scenarios.</li> <li>4. Update the Usage Scenarios as appropriate</li> </ol>				
	For each UI component refine the Current, Ideal, and Realistic Task flows	<ol style="list-style-type: none"> <li>1. Review the task flows developed during the initial Conceptual UI Design Framework session.</li> <li>2. Refine (rebuild if necessary) these task flows to reflect changes in requirements and additional detail identified during detailed design.</li> </ol>		Framework Design Session - Part 1		
	For each UI component refine the Task Objects. Further define attribute details.	<ol style="list-style-type: none"> <li>1. Review the task objects developed during the initial Conceptual UI Design Framework session.</li> <li>2. Refine (rebuild if necessary) these task objects to reflect changes in requirements and additional detail (data attributes, actions) identified during detailed design.</li> </ol>		Framework Design Session - Part 2		

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		<ol style="list-style-type: none"> <li>Review and refine the task object hierarchy.</li> <li>Add task objects (data, container) or modify task objects as necessary.</li> </ol>				
	<p>For each UI component, build a detailed, low-fidelity prototype (primary and secondary windows, task object views)</p> <p>Use metaphor and conceptual model already defined.</p>	<ol style="list-style-type: none"> <li>Review the window prototypes developed during the initial Conceptual UI Design session.</li> <li>Refine the metaphor (if one), template design (if any), menu structure, navigational model, interaction model, and key UI components as needed to reflect changes in the requirements.</li> <li>Using the results of detailed design, iteratively refine the original window prototypes. The window prototypes should be reasonably complete and detailed at this point - sufficient detail to communicate design and perform user walkthroughs using real data.</li> </ol>		Framework Design Session - Part 3		
	<p>For each UI component, perform user/client walkthroughs and informal usability testing on the low-fidelity window prototypes.</p>	<ol style="list-style-type: none"> <li>Perform user walkthroughs and informal usability testing of the detailed window design using the use scenarios</li> <li>Iteratively refine and retest the prototype until complete and</li> </ol>				

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		sufficient.				
	For each UI component, integrate the component window low-fidelity prototypes into the Storyboard.	<ol style="list-style-type: none"> <li>1. Integrate the UI component's low fidelity window prototypes into the Storyboard. Storyboard should demonstrate the UI design to-date, including the high-level conceptual design and each completed UI component design.</li> <li>2. Demonstrate the navigation and flow between windows.</li> <li>3. Post Storyboard in the establish location for informal feedback.</li> </ol>				
	Update the Project Style Guide to reflect the design of each UI component	<ol style="list-style-type: none"> <li>1. Update the style guide to include changes in requirements and UI design (metaphor, conceptual model, menu structure, UI elements, command buttons, interaction model, navigational model)</li> <li>2. Document "new" design decisions that impact common look-and-feel, style, consistency, terminology, and structure.</li> </ol>				
	Update the UI Specification for each UI component.	<ol style="list-style-type: none"> <li>1. Update the UI Spec to include changes in requirements and detailed UI design. <ul style="list-style-type: none"> <li>• refined task flows</li> <li>• refined task objects</li> </ul> </li> </ol>				

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		<ul style="list-style-type: none"> <li>changes to the metaphor or conceptual model</li> <li>detailed UI design</li> </ul>				
	<b>Refine low-fidelity UI Design Prototype</b>					
	Review and refine the Storyboard, ensure consistency between the UI components, review for completeness. Refine Storyboard and low-fidelity prototypes based on review results.	<ol style="list-style-type: none"> <li>Conduct a review of the UI Storyboard. All project stakeholders should participate.</li> <li>Verify completeness, consistency, flow between windows, etc.</li> <li>Iteratively refine and review the Storyboard and window prototypes until sufficient to proceed.</li> <li>Keep Storyboard posted for continued feedback.</li> </ol>				
	Perform usability testing of low-fidelity UI prototype.	<ol style="list-style-type: none"> <li>Perform a formal usability test of the low-fidelity UI prototype with actual users and real-world tasks.</li> <li>Analyze findings and identify necessary design changes.</li> <li>Incorporate design changes into the Storyboard and window prototypes.</li> </ol>				
	Update the project style guide to reflect the design of each UI component	<ol style="list-style-type: none"> <li>Update the style guide to reflect design changes in requirements and UI design as a result of the usability test.</li> <li>Document "new" design decisions</li> </ol>				

<b>User Interface Requirements, Design, and Specification Process</b>						
<b>Activity ID</b>	<b>UI Design Activity</b>	<b>Design Sub-activities</b>	<b>UCD Techniques &amp; Tools</b>	<b>UMI Deliverables</b>	<b>Input</b>	<b>Output</b>
		that impact common look-and-feel, style, consistency, terminology, and structure. 3. Distribute updated Style Guide				
	Update the UI Specification for each UI component.  Update the Style Guide as needed.	1. Update the UI Spec and Style Guide to include changes in requirements and detailed UI design as a result of the usability test. <ul style="list-style-type: none"> <li>• refined task objects</li> <li>• changes to the metaphor or conceptual model</li> <li>• detailed UI design</li> </ul> 2. Redistribute updated UI Spec and Style Guide.				
	<b>Refine Preliminary Requirements and User Acceptance Criteria</b>					
	Refine and Update Preliminary Requirements and User Acceptance Criteria as needed based on usability testing results and review of the Storyboard and UI prototype.	1. Review Detailed UI design and Storyboard against the Preliminary Requirements and User Acceptance Criteria 2. Refine and update the Preliminary Requirements and User Acceptance document based on usability testing results and review of the Storyboard and UI prototype.				

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		<ol style="list-style-type: none"> <li>3. Review Preliminary Requirements for ambiguous wording</li> <li>4. Ensure that all Preliminary Requirements are in scope.</li> <li>5. Eliminate redundant Preliminary Requirements; Resolve conflicts between Preliminary Requirements.</li> <li>6. Identify relationships between Preliminary Requirements for appropriate groupings and to establish dependencies between requirements.</li> </ol>				
	Update the Traceability Matrix as needed (Preliminary Requirements - Business Objectives)	<ol style="list-style-type: none"> <li>1. Ensure that each Preliminary Requirement supports a business objective</li> <li>2. Ensure that Business Objectives are supported by the appropriate Preliminary Requirements.</li> <li>3. Use a Requirements Management Tool for traceability if available.</li> </ol>				
	<b>Build a Navigational UI Prototype</b>					
	Build a operational, navigational prototype to reflect the target UI platform.	<ol style="list-style-type: none"> <li>1. Build a navigational prototype of the UI prototype:                             <ul style="list-style-type: none"> <li>• use a development tool in the target or similar environment.</li> <li>• prototype must be operational</li> </ul> </li> </ol>	<ul style="list-style-type: none"> <li>• MS Visual Basic</li> <li>• UIM/X (CDE/Motif)</li> </ul>			

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		<p>and contain simulations of real data.</p> <ul style="list-style-type: none"> <li>• prototype must allow the user to interact with data and navigate through the tasks.</li> <li>• prototype must allow the user to complete a task.</li> </ul>				
	<p>Perform usability testing on the Navigational prototype.</p> <p>Analyze usability testing results. Determine appropriate design modifications to prototype.</p>	<ol style="list-style-type: none"> <li>1. Perform a formal usability test of the navigational UI prototype with actual users and real-world tasks.</li> <li>2. Analyze findings and identify necessary design changes.</li> <li>3. Incorporate design changes into the Navigational prototype.</li> </ol>				
	<p>Refine the Navigation Prototype based on usability testing results.</p>	<ol style="list-style-type: none"> <li>1. Refine the Navigational Prototype based on analysis from usability testing.</li> <li>2. Release the refined prototype for UI design team review</li> </ol>				
	<p>Update the project style guide to reflect the prototype design changes and any changes to the conceptual model or</p>	<ol style="list-style-type: none"> <li>1. Update the style guide to reflect design changes in requirements and UI design as a result of the usability test.</li> <li>2. Document “new” design decisions that impact common look-and-feel,</li> </ol>				



User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	metaphor.	style, consistency, terminology, and structure. 3. Re-distribute updated Style Guide				
	Update the UI Specification to reflect changes and enhancements to the Navigational Prototype.	1. Update the UI Spec to include changes in requirements and detailed UI design as a result of the usability test. <ul style="list-style-type: none"> <li>• refined task objects</li> <li>• changes to the metaphor or conceptual model</li> <li>• detailed UI design</li> </ul> 2. re-distribute updated UI Spec.				
	<b>Refine Preliminary Requirements and User Acceptance Criteria</b>					
	Refine all Preliminary Requirements (Functional, Non-functional and Non-technical) and User Acceptance Criteria based on development and usability testing of the Navigational Prototype	1. Update all Preliminary Requirements and Acceptance Criteria (Functional, Non-functional, and Non-technical) based on the development and usability testing of the Navigational Prototype. 2. Review all Preliminary Requirements (Functional, Non-functional, and Non-technical) for ambiguous wording 3. Review User Acceptance Criteria				

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		(measurements) for each Preliminary Requirement (Functional, Non-functional, and Non-technical). Note: some Non-functional and Non-technical Preliminary Requirements may not have an acceptance criteria.				
	Refine the Traceability Matrix (Functional, Non-Functional and Non-Technical Preliminary Requirements → Business objectives)	<ol style="list-style-type: none"> <li>1. Ensure that all Non-functional and Non-technical requirements are within scope and map back to a valid business objective.</li> <li>2. Use a Requirements Management Tool for traceability if available.</li> </ol>				
	<b>Conduct User Review of Preliminary Requirements and Acceptance Criteria</b>					
	Perform a User/Client review of Preliminary Requirements and User Acceptance Criteria	<ol style="list-style-type: none"> <li>1. Distribute Preliminary Requirements to User/Client representatives for review</li> <li>2. Facilitate a group session with User/Client to identify any issues and obtain buy-in.</li> </ol>				
	Review Issues	<ol style="list-style-type: none"> <li>1. Identify, review, and resolve issues</li> <li>2. Update the Issue Log</li> </ol>				
	Modify Preliminary	<ol style="list-style-type: none"> <li>1. Update Preliminary Requirements</li> </ol>				

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
	Requirements and User Acceptance Criteria as a result of user review.	to reflect agreed upon changes as a result of the user/client review. 2. Update task diagrams, UI prototypes, and all other analysis documents to reflect changes in Preliminary Requirements.				
	Obtain user/client acceptance of Preliminary Requirements and User Acceptance Criteria	1. Distribute updated Preliminary Requirements to all user/client stakeholders. 2. Obtain acceptance of Preliminary Requirements. <i>Acceptance indicates that the Preliminary Requirements may be submitted for formal review and approval.</i>				
	Reclassify the category of Preliminary Requirements	1. Modify the category of Preliminary Requirements based on user/client review and acceptance. Categories may include: <ul style="list-style-type: none"> <li>• Preliminary - not accepted</li> <li>• Accepted</li> <li>• Accepted Pending Issue resolution</li> <li>• Deferred</li> <li>• Canceled</li> </ul>	Requirements Management Tool			
	Identify, review, and resolve issues; Update the Issue Log	1. Log any issues that require further investigation or that must be resolved.	<ul style="list-style-type: none"> <li>• Issues Repository</li> <li>• Requirements Management Tool</li> </ul>			

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		<ol style="list-style-type: none"> <li>2. Review, update, and resolve open issues</li> <li>3. Update Issues Log</li> </ol>	<ul style="list-style-type: none"> <li>• MS Excel</li> <li>• MS Word</li>   <li>• Group facilitation</li> <li>• Issue reporting process</li>   <li>• Issue tracking form</li> <li>• Project Management procedures</li> </ul>			
	<b>Commit Preliminary Requirements - Baseline</b>					
	Conduct Formal User/Client Review of Accepted Preliminary Requirements and User Acceptance Criteria	<ol style="list-style-type: none"> <li>1. Distribute <i>accepted</i> Preliminary Requirements to all user/client stakeholders and project team members.</li> <li>2. Review requirements with all user/client stakeholders and project team members until everyone shares a common understanding and feels comfortable moving forward with the design and project.</li> <li>3. Consider distributing video summaries of formal usability testing, participatory design</li> </ol>	<ul style="list-style-type: none"> <li>• Formal review</li> <li>• Walkthrough</li> <li>• Facilitated session</li> </ul>			

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		sessions, and any user feedback session.				
	Review and resolve Open Issues; Update Issues Log	<ol style="list-style-type: none"> <li>1. Log any issues that require further investigation or that must be resolved as a result of the formal review.</li> <li>2. Review, update, and resolve open issues</li> <li>3. Update Issues Log</li> </ol>	<ul style="list-style-type: none"> <li>• Issues Repository</li> <li>• Requirements Management Tool</li> <li>• MS Excel</li> <li>• MS Word</li> <li>• Group facilitation</li> <li>• Issue reporting process</li> <li>• Issue tracking form</li> <li>• Project Management procedures</li> </ul>			
	Obtain formal sign-off from User/Client Approving Requirements.	<ol style="list-style-type: none"> <li>1. Obtain signatures on a formal Sign-off form from representatives of each user/client and project team group. All major stakeholders should sign this form.</li> <li>2. Signature signifies acceptance of the Preliminary Requirements as Approved.</li> </ol>	<ul style="list-style-type: none"> <li>• Sign-off form</li> </ul>			
	Baseline Requirements	<ol style="list-style-type: none"> <li>1. Change the status of the Requirements to “baselined” in the Requirements Management Tool.</li> </ol>	<ul style="list-style-type: none"> <li>• Requirements Management Tool</li> </ul>			

User Interface Requirements, Design, and Specification Process						
Activity ID	UI Design Activity	Design Sub-activities	UCD Techniques & Tools	UMI Deliverables	Input	Output
		Baselined items drive project estimates, project management, code development, and product deployment.	<ul style="list-style-type: none"> <li>• MS Excel</li> <li>• MS Word</li> </ul>			
	Place Baselined Requirements under Revision Control.	1. Once requirements are baselined and placed under revision control, they may only be modified using a formal SCM process.	<ul style="list-style-type: none"> <li>• Software Configuration Management Tool</li> </ul>			
	Implement a formal Software Configuration Management Control process.					
	Identify, review, and resolve issues; Update the Issue Log	<ol style="list-style-type: none"> <li>1. Log any issues that require further investigation or that must be resolved as a result of committing and baselining the requirements.</li> <li>2. Review, update, and resolve open issues</li> <li>3. Update Issues Log</li> </ol>	<ul style="list-style-type: none"> <li>• Issues Repository</li> <li>• Requirements Management Tool</li> <li>• MS Excel</li> <li>• MS Word</li> <li>• Group facilitation</li> <li>• Issue reporting process</li> <li>• Issue tracking form</li> <li>• Project Management procedures</li> </ul>			

