

## APPENDIX C

### EVALUATION RESULTS OF ORGANIZATIONS "A" AND "B"

TABLE 1: EVALUATION SCORES OF ORGANIZATION "A".

Level-2 (Basic) assessment							
NO	Practices	Approach 0,2,4,6,8,10	Deployment 0,2,4,6,8,10	Result 0,2,4,6,8,10	Average	Total of average scores	Final score
1	After SW completed: Conducting a summative evaluation by an external expert using some basic usability practices such as heuristic evaluation, cognitive walkthroughs, user testing, or questionnaires (such as the System Usability Scale (SUS)) with the end users after the software is completed	10	10	8	9	9	9
Level-3 (Implemented) assessment							
1	Requirements Elicitation: elicit the usability requirements using the basic elicitation methods such as (interviews with stakeholders, brainstorming, observations, surveys, or personas)	2	2	2	2	2	2
2	Design Practice: developing UI prototypes (Low fidelity prototyping)	8	8	6	7	15	7.5
3	Design Practice: developing the software following usability guidelines\heuristics (user interface design guidelines) proposed by researchers such as Nielsen	8	8	8	8		
4	Testing Practice: conducting a formative evaluation by an internal usability engineer using user testing, heuristic evaluation, cognitive walkthroughs, usability metrics, or questionnaires such as the System Usability Scale (SUS) with the end users before releasing the software	4	4	4	4	4	4
Level-4 (Integrated) assessment							
Requirement Phase Practices							
1	Requirements Elicitation: Elicit the usability requirements and identify the user needs through (user Analysis, task analysis, competitive analysis, brainstorming, site visits with (shadowing\observation, ethnography, contextual inquiry), designing the product vision which is called (user environment design model or roadmap, or contextual model), studying available documentations, building a tree structure using interface design guidelines and usability guidelines that help the analyst to capture usability requirements, focus group, surveys, or personas)	4	4	2	3	10	2.5
2	Requirements Analysis: analyze the usability requirements through prototyping, user behavioral modeling processes (UBMP) such as (Usability use case metamodel and task model), card sorting, analyzing the user workflow, walking through of use case, user profile analysis, the definition of usability goals, hierarchical task analysis (cognitive and physical), scenarios of use, users research, functional analysis, or impact analysis	4	2	2	3		
3	Requirement Specification: create usability specifications, identify standards and guidelines for HCI design	0	0	0	0		
4	Requirements Validation: validate the usability requirements by evaluating the prototypes using cognitive walkthroughs, pluralistic walkthroughs, heuristic evaluation, focus groups, or think Aloud\reflective think-Aloud (RTA)	4	4	4	4		
Design Phase Practices							
1	Create conceptual UI model to establish the following: how the users would understand the way the product worked, how the product was organized around the tasks, and how to use the product to accomplish the tasks. After that the UI is designed	0	0	0	0	32	2
2	Iterative usability testing of the designed UI components (UI reviews and walk-throughs that can be done while the UI is being developed and that provide the opportunity for immediate feedback)	6	6	6	6		
3	Identify a detailed interaction design	6	6	6	6		
4	Design a convenient help component within the system	2	0	0	1		
5	Conduct an exploratory test to compare design alternatives or possible solutions	4	2	0	2		
6	Using interaction patterns which represent design solutions for known usability problems	0	0	0	0		
7	Evaluate the designed UI using cognitive walkthroughs, heuristic evaluation, focus groups, or think Aloud\reflective think-Aloud (RTA)	4	4	4	4		

8	Establish, maintain, and prioritize the needs for user quality attributes while designing the user interface using usability standards and guidelines	0	0	0	0		
9	Use the HCI interface patterns to capture best practices for solving particular user-interface design scenarios in order to improve system interface appearance and navigation; improve access to system functionalities (such as menus)	2	2	2	2		
10	Evaluate design against user requirements	0	0	0	0		
11	Focus closely on users by designing prototypes (low-fidelity prototypes or high-fidelity prototypes, detailed mock-ups simulating the functionality) and having users participate to get user feedback promptly. Then, according to feedback, the design is improved continuously, until it meets the usability requirements	0	0	0	0		
12	Involve the user in the design process (parallel design): As the design reaches a substantive stage, it is important to continue contact with actual users over several iterations to make sure that the final product meets their needs	0	0	0	0		
13	Using navigation maps when designing the UI	4	4	4	4		
14	Measure the usability attributes (such as understandability of a user interface) earlier using the designed conceptual model	0	0	0	0		
15	Share design documents and artifacts by maintaining communication among team members to make informed decisions about design tradeoffs	8	8	6	7		
16	Analyze the cognitive behaviors of users' perception, memory, and thinking in software interface design to make the UI have a greater usability level	0	0	0	0		
Coding Phase Practices							
1	Review the code concerning the usability	0	0	0	0	0	0
2	Verification of design requirements while coding	0	0	0	0		
3	Ensure the code usability by using highly usable APIs (Application Programming Interface) make the integration of usability to UI easier and faster and provides a stable means of extending and integrating usability across organizational processes' infrastructure	0	0	0	0		
4	Use the adaptation techniques (Adaptive Presentation, Adaptive Navigation Support) when possible to make the content of software interfaces presented according to the user's class and knowledge state	0	0	0	0		
Testing Phase Practices							
1	Conduct usability tests after the system integration using cognitive walkthroughs\pluralistic walkthroughs, heuristic evaluation, usability metrics, or questionnaires	6	4	4	5	29	3.22
2	Conduct user testing with system end-users (Remote through online tools /in-person in the lab) to assess the developed system's usability	0	0	0	0		
3	Assess the usability of the developed system using interview and opinion mining with end-users	0	0	0	0		
4	Assess the developed system via usability automated tools	0	0	0	0		
5	Verify the usability of the developed system using a usability checklist	6	6	6	6		
6	Conduct focus groups, and peer group review sessions with experts to evaluate the developed system's usability	6	6	6	6		
7	Conduct a component-based usability testing before the system integration	6	6	6	6		
8	Perform usability studies at user sites	0	0	0	0		
9	Perform scenario-based testing (usage scenario)	6	6	6	6		
Maintenance Phase Practices							
1	Maintain communication with real users	4	4	4	4	4	4
Installation / Transition into Use							
1	Collect user feedback after the software product(system) is installed to identify (1) user suggestions (2) new requirements, and (3) continuous improvement concerning the usability	0	0	0	0	3	1
2	Preparation of user manual to increase the system usability	4	4	2	3		
3	Support of user training	0	0	0	0		
Level-5 (Optimized) assessment							
1	General at any SDLC phase: Develop a standardized usability policy.	0	0	0	0	5	2.5
2	General at any SDLC phase: Implement some usability practices that could be used as a UX investigation method such as think-aloud, focus groups, user testing... etc.	6	4	4	5		

Adopt UX practices during some SDLC phases:							
3	Requirements Phase: conduct user interviews to understand the cognitive needs of users during the requirements elicitation	2	0	0	1	1	1
4	Design phase: produce artifacts or prototypes to guide UX	0	0	0	0	5	1.66
5	Design phase: investigate the user experience (UX) using user experience questionnaires (UEQ), or interviews through the developed prototype with end users	0	0	0	0		
6	Design phase: Use the user experience design guidelines	6	4	4	5		
7	Testing phase: investigate the user experience (UX) using user experience questionnaires (UEQ), or interviews through the developed system with end users	8	6	6	7	7	3.5
8	Testing phase: conduct a User Examine Method (User testing) with the help of observation and recording techniques to capture the UX such as (video recording, Eye-tracking technology, measuring stress, and emotional metric for example ( GSR (galvanic skin response) to measure skin conductivity, BVP (Blood Volume Pulse) to measure blood pulse, HRV (Heart Rate Variability) to measure heartbeat interval, electromyography (EMG) to measure muscle activity, and EEG to measure brain activity,... etc.) based on the software context	0	0	0	0		

TABLE 2: EVALUATION SCORES OF ORGANIZATION "B".

Level-2 (Basic) assessment							
NO	Practices	Approach 0,2,4,6,8,10	Deployment 0,2,4,6,8,10	Result 0,2,4,6,8,10	Average	Total of average scores	Final score
1	After SW completed: Conducting a summative evaluation by an external expert using some basic usability practices such as heuristic evaluation, cognitive walkthroughs, user testing, or questionnaires (such as the System Usability Scale (SUS)) with the end users after the software is completed	2	2	3	2	2	2
Level-3 (Implemented) assessment							
1	Requirements Elicitation: elicit the usability requirements using the basic elicitation methods such as (interviews with stakeholders, brainstorming, observations, surveys, or personas)	4	4	4	4	8	4
2	Design Practice: developing UI prototypes (Low fidelity prototyping)	4	4	4	4		
3	Design Practice: developing the software following usability guidelines/heuristics (user interface design guidelines) proposed by researchers such as Nielsen	4	4	4	4		
4	Testing Practice: conducting a formative evaluation by an internal usability engineer using user testing, heuristic evaluation, cognitive walkthroughs, usability metrics, or questionnaires such as the System Usability Scale (SUS) with the end users before releasing the software	0	2	4	2	2	2
Level-4 (Integrated) assessment							
Requirement Phase Practices							
1	Requirements Elicitation: Elicit the usability requirements and identify the user needs through (user Analysis, task analysis, competitive analysis, brainstorming, site visits with (shadowing)observation, ethnography, contextual inquiry), designing the product vision which is called (user environment design model or roadmap, or contextual model), studying available documentations, building a tree structure using interface design guidelines and usability guidelines that help the analyst to capture usability requirements, focus group, surveys, or personas)	6	4	4	5	18	4.5

2	Requirements Analysis: analyze the usability requirements through prototyping, user behavioral modeling processes (UBMP) such as (Usability use case metamodel and task model), card sorting, analyzing the user workflow, walking through of use case, user profile analysis, the definition of usability goals, hierarchical task analysis (cognitive and physical), scenarios of use, users research, functional analysis, or impact analysis	4	4	4	4		
3	Requirement Specification: create usability specifications, identify standards and guidelines for HCI design	6	4	4	5		
4	Requirements Validation: validate the usability requirements by evaluating the prototypes using cognitive walkthroughs, pluralistic walkthroughs, heuristic evaluation, focus groups, or think Aloud\reflective think-Aloud (RTA)	4	4	4	4		
Design Phase Practices							
1	Create conceptual UI model to establish the following: how the users would understand the way the product worked, how the product was organized around the tasks, and how to use the product to accomplish the tasks. After that the UI is designed	6	6	6	6	62	3.87
2	Iterative usability testing of the designed UI components (UI reviews and walk-throughs that can be done while the UI is being developed and that provide the opportunity for immediate feedback)	2	2	2	2		
3	Identify a detailed interaction design	6	4	4	5		
4	Design a convenient help component within the system	6	6	6	6		
5	Conduct an exploratory test to compare design alternatives or possible solutions	2	2	2	2		
6	Using interaction patterns which represent design solutions for known usability problems	4	4	4	4		
7	Evaluate the designed UI using cognitive walkthroughs, heuristic evaluation, focus groups, or think Aloud\reflective think-Aloud (RTA)	4	4	4	4		
8	Establish, maintain, and prioritize the needs for user quality attributes while designing the user interface using usability standards and guidelines	6	4	4	5		
9	Use the HCI interface patterns to capture best practices for solving particular user-interface design scenarios in order to improve system interface appearance and navigation; improve access to system functionalities (such as menus)	6	4	4	5		
10	Evaluate design against user requirements	6	4	4	5		
11	Focus closely on users by designing prototypes (low-fidelity prototypes or high-fidelity prototypes, detailed mock-ups simulating the functionality) and having users participate to get user feedback promptly. Then, according to feedback, the design is improved continuously, until it meets the usability requirements	4	4	4	4		
12	Involve the user in the design process (parallel design): As the design reaches a substantive stage, it is important to continue contact with actual users over several iterations to make sure that the final product meets their needs	2	2	2	2		
13	Using navigation maps when designing the UI	2	2	2	2		
14	Measure the usability attributes (such as understandability of a user interface) earlier using the designed conceptual model	4	2	4	3		
15	Share design documents and artifacts by maintaining communication among team members to make informed decisions about design tradeoffs	4	2	2	3		
16	Analyze the cognitive behaviors of users' perception, memory, and thinking in software interface design to make the UI have a greater usability level	4	4	4	4		
Coding Phase Practices							
1	Review the code concerning the usability	2	2	2	2	8	2
2	Verification of design requirements while coding	4	4	4	4		
3	Ensure the code usability by using highly usable APIs (Application Programming Interface) make the integration of usability to UI easier and faster and provides a stable means of extending and integrating usability across organizational processes' infrastructure	2	2	2	2		
4	Use the adaptation techniques (Adaptive Presentation, Adaptive Navigation Support) when possible to make the content of software interfaces presented according to the user's class and knowledge state	0	0	0	0		
Testing Phase Practices							

1	Conduct usability tests after the system integration using cognitive walkthroughs/pluralistic walkthroughs, heuristic evaluation, usability metrics, or questionnaires	4	4	4	4	16	1.77
2	Conduct user testing with system end-users (Remote through online tools /in-person in the lab) to assess the developed system's usability	0	0	0	0		
3	Assess the usability of the developed system using interview and opinion mining with end-users	2	2	2	2		
4	Assess the developed system via usability automated tools	2	2	2	2		
5	Verify the usability of the developed system using a usability checklist	2	2	2	2		
6	Conduct focus groups, and peer group review sessions with experts to evaluate the developed system's usability	0	0	0	0		
7	Conduct a component-based usability testing before the system integration	2	2	2	2		
8	Perform usability studies at user sites	0	0	0	0		
9	Perform scenario-based testing (usage scenario)	4	4	4	4		
Maintenance Phase Practices							
1	Maintain communication with real users	6	4	4	5	5	5
Installation / Transition into Use							
1	Collect user feedback after the software product(system) is installed to identify (1) user suggestions (2) new requirements, and (3) continuous improvement concerning the usability	6	6	6	6	14	4.66
2	Preparation of user manual to increase the system usability	6	6	6	6		
3	Support of user training	2	2	2	2		
Level-5 (Optimized) assessment							
1	General at any SDLC phase: Develop a standardized usability policy.	0	0	0	0	2	1
2	General at any SDLC phase: Implement some usability practices that could be used as a UX investigation method such as think-aloud, focus groups, user testing... etc.	2	2	2	2		
Adopt UX practices during some SDLC phases:							
3	Requirements Phase: conduct user interviews to understand the cognitive needs of users during the requirements elicitation	6	6	6	6	6	6
4	Design phase: produce artifacts or prototypes to guide UX	8	6	6	7	13	4.33
5	Design phase: investigate the user experience (UX) using user experience questionnaires (UEQ), or interviews through the developed prototype with end users	2	2	2	2		
6	Design phase: Use the user experience design guidelines	4	4	4	4		
7	Testing phase: investigate the user experience (UX) using user experience questionnaires (UEQ), or interviews through the developed system with end users	2	2	2	2	2	1
8	Testing phase: conduct a User Examine Method (User testing) with the help of observation and recording techniques to capture the UX such as (video recording, Eye-tracking technology, measuring stress, and emotional metric for example ( GSR (galvanic skin response) to measure skin conductivity, BVP (Blood Volume Pulse) to measure blood pulse, HRV (Heart Rate Variability) to measure heartbeat interval, electromyography (EMG) to measure muscle activity, and EEG to measure brain activity... etc.) based on the software context	0	0	0	0		