بهبود تعامل کاربر و ERP با استفاده از تئوری همکاری

دکتر علیرضا خراسانی

عضو هیئت علمی گروه مهندسی فناوری اطلاعات ، دانشگاه آزاد اسلامی- واحد تهران جنوب

نشانی پست الکترونیکی :

alireza1@mac.com

سارا عرب یارمحمدی

دانشجوی کارشناسی مهندسی فناوری اطلاعات، دانشگاه آزاد اسلامی- واحد تهران جنوب

نشانی پست الکترونیکی :

Sara.arabyarmohammadi@gmail.com

Archive of SID

چکیده :

یادگیری و استفاده از سیستمهای برنامه ریزی منابع سازمان (ERP) علی رغم صرف منابع وسیع جهت آموزش کارمندان، همچنان به عنوان یک معضل پیش روی مدیران اطلاعات سازمان (CIO`s[1]) خودنمایی می کند. بیشتر اوقات حتی کارمندانی هم که به خوبی آموزش دیده اند ، برای گرفتن راهنمایی و کمک ، به کاربران مطلع تر رجوع می کنند یا مجبورند برای تقویت و تکمیل استفاده از سیستم ، نرم افزار های دیگری نظیر برنامه های کاربردی صفحه گسترده یا پایگاه داده را بکار ببرند. چنین وضعیتی بر روی عملکرد کارمندان تاثیر منفی داشته و مانع کسب تمامی منافعی است ، که از ERP برای یک شرکت به ارمغان می آید. یکی از عوامل مؤثر در بروز این مسئله ، نحوه ی طراحی سیستم ERP است که این عامل می تواند در قابلیت استفاده از این سیستم نقش بسزایی داشته باشد . این مقاله سعی می کند تا با استفاده از تئوری همکاری راه های جدیدی را برای حل این مشکل مطرح کرده و طراحی منعطف تری از ERP را معرفی نماید تا کاربر با دغدغه خاطر کمتری بتواند از سیستم ERP استفاده کند.

کلمات کلیدی : سیستم برنامه ریزی منابع سازمان (ERP system) ، تئوری همکاری (Collaboration Theory) ، واسط کاربر (Interface) ، قابلیت استفاده (Usability)

[1] Chief Information Officer :

www.SID.ir

مقدمه :

گزارش مشکلاتی که کاربران حین کار با سیستم برنامه ریزی منابع سازمان (ERP) با آن ها رو برو می شوند و همچنین مطالعات اخیر نشان داده است که این سیستم ها در زمینه ی استفاده دارای ضعف هایی هستند.

مطالعه ای از کاربران ERP در ۵۰۰ شرکت موفق ، ۶ سطح از مشکلات استفاده را از جمله :

۱- دشواری در شناسایی و دسترسی به عملکرد صحیح

۲- فقدان پشتیبانی از یک تراکنش

۳- محدودیت های خروجی سیستم

۴- پشتیبانی ناکافی در شرایط پدید آمدن Error

۵- عدم تطابق بین مجموعه اصطلاحات کاربر و سیستم

و مشکلاتی که در رابطه با استفاده از سیستم پدید آمده، مشخص کرده است.[۵]

شکل ۱ : مشکلات استفاده از سیستم ERP

گرچه کارمندان دوره های آموزشی را گذرانده اند ، باز به طور گسترده متکی به یادداشت های غیر رسمی همکاران خود هستند. گاهی در پی کاربران رده بالاتر[1] جهت پرسیدن سوال هستند و گاهی جهت رسیدگی به نیازهای برآورده نشده ، به استفاده از نرم افزار های خارجی مانند Microsoft Excel روی می آورند.

مباحث مربوط به قابلیت استفاده ، می توانند تاثیر بسزایی بر روی کارایی کسب و کار و بخصوص بازدهی کاربر نهایی ، داشته باشند .

مطالعات اخیر نشان داده که طراحی ضعیف واسط کاربر، تاثیرات منفی زیادی از جمله : افزایش هزینه ی آموزش کاربر ،کاهش بهره وری و درجه ی پایین سازگاری کاربر دارد.

عجیب است که علی رغم صرف زمان ، هزینه و تلاش فراوان جهت پیاده سازی و آموزش، توجه بسیار کمی معطوف به درک راه های تعامل کاربر با نرم افزار ERP و درجه ی پشتیبانی مدل تعاملی از وظایفی که انجام می گیرد ، می شود.

به وسیله ی تئوری همکاری بر همکاری بین کاربر وسیستم تاکید می شود . این مفهوم با همکاری بین افراد که به وسیله ی تکنولوژی کامپیوتری پشتیبانی می شود و "تشریک مساعی با حمایت کامپیوتر " (CSCW) نامیده می شود، متفاوت است. هر

[1] Power User :

Archive of SID

www.SID.ir

چقدر فناوری با اهدف کاربران بیشتر مطابقت داشته باشد ، ضمن افزایش کارایی کاربر و رضایتمندی از سیستم ، بهتر می تواند با استفاده از روش همکاری به نیاز های کاربران پاسخ دهد.

هدف این مقاله یافتن راهی برای بهبود استفاده از سیستم های سازمانی ، از طریق افزایش قابلیت همکاری در واسط های کاربر این سیستم هاست.[۱]

تحقیقات مرتبط :

مطالعات اندکی بر طراحی واسط کاربر و قابلیت استفاده سیستم های سازمانی ، متمرکز شده اند. اما بیشو [1] و همکارانش در سال ۱۹۹۹ یک سری عوامل انسانی را در ارتباط با سیستم های برنامه ریزی منابع سازمان ، مطرح کردند. برخی از این عوامل عبارتند از : عدم توجه به آموزش کاربر ، عدم توجه به هزارتویی از صفحات که یک کاربر باید آنها را به درستی طی کند. کالیسیر و کالیسیر [2] در سال ۲۰۰۴ ، تاثیرات قابل استفاده بودن یک واسط ، میزانی سودمندی و قابلیت کاربرد آسان آن را بر میزان رضایتمندی کابر نهایی از سیستم ERP ، روی ۵۱ کاربر در ۲۴ شرکت ، بررسی کرده اند.

ویژگی های قابل استفاده بودن در بررسی عبارتند از: توانایی ، سازگاری ، انعطاف پذیری ، راهنمایی کاربر ، قابلیت یادگیری و حداقل ذخیره گذاری در حافظه. این محققان دریافتند که قابل استفاده بودن ، تاثیر بسیار چشمگیری بر رضایتمندی کاربر دارد در حالی که قابلیت یادگیری ، به نسبت تاثیر کمتری دارد اما همچنان اثر آن قابل توجه است. قابلیت کاربرد آسان ، به طور غیر مستقیم از طریق احساس سودمندی توسط کاربر ، بر رضایتمندی ، موثر است ، به این معنا که کاربران در صورتی سیستم ERP را برای استفاده مفید می دانند که استفاده از آن آسان باشد .

به عنوان مثال کارمندی را در نظر بگیرید که برای انجام وظایف روزانه ی خود از طریق سیستم ERP نصب شده در کارخانه ، دچار مشکل است .فرض کنید او موظف است هر روز لیستی از مواد اولیه ی مورد نیاز بخشی از کارخانه را تهیه کرده و آنها را سفارش بدهد اما او به روش سنتی عادت دارد و برای استفاده از سیستم مجبور است مدام از همکارانش سوال کند. این وضعیت ، انجام عملیات معمول و همیشگی اش را ، سخت و زمانبر می کند .بعلاوه او پس از پرسش های فراوان و طی قدم ها با شک وتردید ، مطمئن نمی شود که آیا سفارش مواد اولیه را نهایتا درست تهیه و ارسال کرده یا خیر. در نظر چنین کارمندی استفاده از روش سفارش و پیگیری سابق ، به مراتب راحت تر ،مطمئن تر ، سریع تر و سودمند تر است.

همچنین ، توانایی سیستم اثر مهمی بر احساس سودمندی توسط کاربر دارد و محققان پیشنهاد می کنند که طراحان سیستم های ERP باید بیشتر به تحلیل نیاز های کاربر توجه کنند ، تا اصول و وظایف مناسب و مرتبط را در سیستم خود در نظر بگیرند . این مطالعه همچنین دریافت که یک نقشه ی راهنمای خوب برای کاربر ، قابلیت یادگیری سیستم را بالا می برد و حجم کار ذهنی را کاهش می دهد. این طرح راهنما موارد زیر را پیشنهاد می دهد :

- پیغام های خطا (error message) ی قابل فهم
- قابلیت استفاده از سیستم بدون نیاز به یادگیری تمامی آن
- فراهم بودن امکان خنثی کردن (Undo)
- عملیات کنترلی بازگشت و وجود سوالات تایید کننده پیش از انجام دستورات پرمخاطره

---

[1] : Bishu

[2] : Calisir and Calisir

Archive of SID

www.SID.ir

موارد فوق می تواند احساس سودمندی توسط کاربر و قابلیت یادگیری را افزایش دهد.

شکل ۲ : موارد موثّر در افزایش احساس سودمندی توسط کاربر و قابلیت یادگیری

در حالی که در ارتباط با قابلیت استفاده در طراحی سیستم های ERP ،تحقیقات بسیار اندکی انجام شده ، تحقیق در مورد تعامل انسان با کامپیوتر به طور کلی ، به دستاورد های قابل توجهی رسیده است که نتیجه ی آن تعداد زیادی از انواع واسط های نوآورانه است.گرچه بسیاری از واسط های آزمایشی وارد عرصه های کاربردی شده اند ، تا آنجا که ما می دانیم هیچکدام به طور کامل در زمینه ی سیستم های ERP به کار نرفته اند . در حال حاضر هیچ محصولی وجود ندارد که تمامی این ویژگی های پیشنهادی را به تمامی در نظر گرفته باشد. استفاده از پیشرفت های علمی و تکنولوژیکی که در عرصه ی واسط های کاربر ، صورت گرفته است ، در این سیستم ها ، می تواند تضمین کننده ی افزایش و بهبود قابلیت استفاده ی آنها باشد.

در حقیقت ، ساختن سیستمی که قابلیت استفاده ی ان در سطح بالایی قرار دارد ،بدون در نظر گرفتن همه ی مباحث در این زمینه از ابتدا ، غیر ممکن خواهد بود :

اولین مطلب ، تحلیل نیازها و در نظر گرفتن سطوح طراحی است (مگوآیر [1]، ۲۰۰۱) . این رویکرد تحت عنوان "طراحی متمرکز بر کاربر" [2] شناخته می شود. گرچه روش هایی که برای این طراحی به کار برده می شود با تئوری های بنیادی خود ، تفاوت زیادی دارد ، امّا کلید همه ی طراحی های نوین قابلیت استفاده و ارزیابی تکنیک ها، درک درست و روشنی از اهداف کاربر و وظایف اوست.

قلمرو وسیع ERP و سایر سیستم های بزرگ سازمانی ، نیازهای خاص خود را جهت تحلیل قابلیت استفاده ، دارد.علاوه بر ارزیابی یک عمل به طور جداگانه ، استفاده از چندین سیستم به صورت یکپارچه و متحد برای رسیدن به یک هدف فراگیر نیز ،

[1] Maguire :
[2] User-centered design :

Archive of SID
www.SID.ir

باید تحلیل شود. در حال حاضر تعداد کمی از متدولوژی های قابلیت استفاده، وجود دارند ، که برای این گونه از تحلیل مناسبند. برای مثال روش های بر پایه ی مدل ، مانند GOMS (کارد [1] و همکاران ، ۱۹۸۳) نمی توانند استفاده شوند چراکه به طور طبیعی ، مشخص کردن وظایف پیچیده و رفتار های کاربر به صورت دقیق و جزئی ، کار دشواری است. در حالی که "تحلیل وظیفه"[2] (ردیش [3] و ویکسون [4] ، ۲۰۰۳) برای مدل کردن سلسله ای از اهداف و وظایف در سطح بالا ، قابل استفاده است ، این روش تعامل بین سیستم و کاربرانش را کنترل نمی کند. بنابراین "تحلیل وظیفه" به طور موثر نمی تواند در طراحی واسط کاربر ، به کار برده شود.

گرچه از نظر تئوری ، ارزیابی تعامل های مورد نیاز برای انجام یک وظیفه ی جامع، با استفاده از روش های نظارت و بازرسی ، امکان پذیر است ، (مثل نیلسِن [5] ۱۹۹۳ ؛ وارتُن [6] و همکاران ۱۹۹۴) کوکتُن [7] و همکارانش در سال ۲۰۰۳ ، گزارش می دهند که در ارزیابی قابلیت استفاده به کلی از این روش ممیزی ، صرف نظر شده است. این مسئله می تواند نشان دهنده ی این واقعیت باشد که راهبردی که روش ها بر پایه ی آن گذاشته شده اند ، پویایی تعامل بین سیستم و کاربرانش را کنترل نمی کند ، بلکه بیشتر بر جنبه های ایستای سیستم متمرکز است . روش های بر مبنای کاربر[8] (دوماس [9] ، ۲۰۰۳) برای آشکار کردن مشکلات قابلیت استفاده ، خوب عمل می کنند اما بیشتر بر ویژگی های مشخص پیاده سازی موجود ، تمرکز می کنند . بنابراین آنها می خواهند اطلاعاتی را که در چارچوب یک ابزار مشخص در حال ارزیابی ، محصور شده ، استخراج کنند که منجر به اصلاحات محدود و موضعی می شود و نه دگرگونی طراحی سیستم به طور وسیع .

کنستانتین [10] (۲۰۰۶) در مورد رویکرد جدیدتری که "مدلسازی فعالیت"[11] نامیده می شود ،بحث می کند تا با این رویکرد ، سیستم و نقش کاربران در ارتباط با سیستم ، را به کاربران بیاموزد. این رویکرد بر مبنای " تئوری فعالیت"[12] و"طراحی متمرکز بر کاربرد"[13] است . پس مدلسازی فعالیت به جای توجه به کاربران (که هدف "طراحی متمرکز بر کاربر" است) ، بیشتر به فعالیت هایی اهمیت می دهد که کاربران با آنها درگیر هستند و وظایفی که می خواهند از طریق سیستم انجام دهند.

از میان رویکردهای فوق ، "مدلسازی فعالیت" بیشتر از سایرین با "تئوری همکاری" منطبق است ، به این خاطر که این مدلسازی بر تعامل کاربر با سیستم و فعالیت هایی که کاربر و سیستم در آنها شرکت دارند ، متمرکز است.

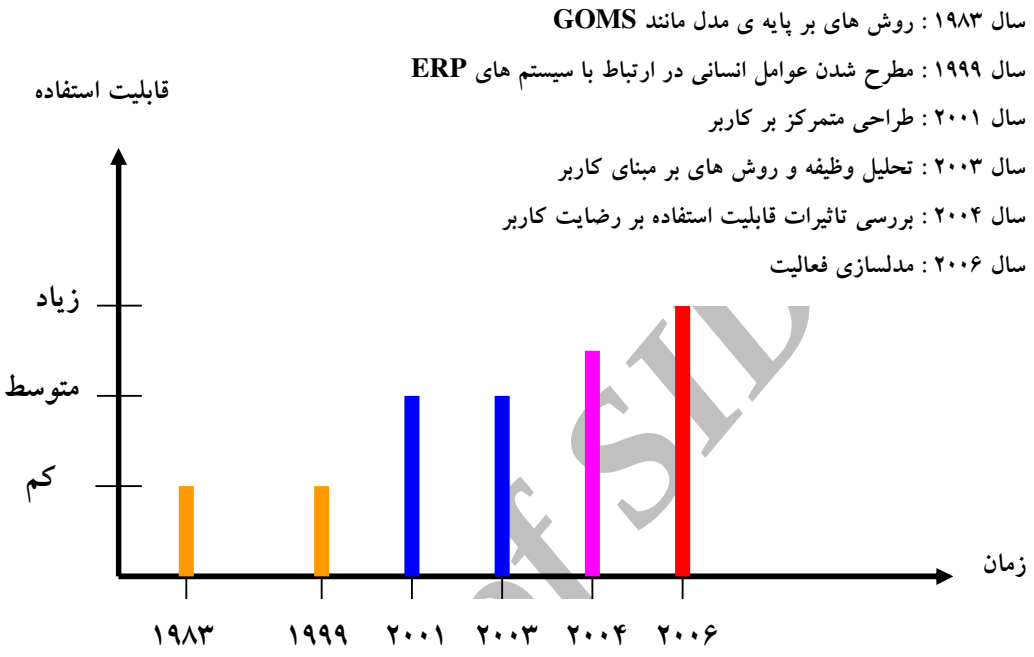
نگاه به تعامل کاربر و سیستم از دید همکاری ، به طور صریح و آشکار ، کاربر و سیستم را عضوی از مدلی در نظر می گیرد که در آن مجموعه ای از تعاملات برای تکمیل یک وظیفه ، انجام می شود. در این دیدگاه نقش سیستم به عنوان یاوری است که کاربر را در رسیدن به اهدافش یاری می کند. این تئوری ، کاربر را از فردی با مسئولیت ها و دانش در مورد فرآیند ، به فردی که به عنوان شریک با سیستم در تکمیل یک فرآیند ، مشارکت می کند، تغییر می دهد که این به طور طبیعی نیازمند دانش و رفتار متناسب سیستم است که در ادامه بررسی می شود.[۱]

---

[1] : Card
[2] : Task analysis
[3] : Redish
[4] : Wixon
[5] : Nielsen
[6] : Wharton
[7] : Cockton
[8] : User based Methods
[9] : Dumas
[10] : Constantine
[11] : Activity Modeling
[12] : Activity theory
[13] : usage-centered design

Archive of SID

www.SID.ir



شکل ۳ : سیر پیشرفت روش های ارائه شده در زمینه ی طراحی واسط کاربر

ویژگی های مهم واسط کاربر

به طور کلی می توان ویژگی های یک واسط کاربر خوب را به صورت زیر برشمرد :

- با استفاده از یک واسط کاربر خوب ، کاربر می تواند به آسانی به سیستم بگوید که چه می خواهد بکند یا سیستم به سادگی می تواند از کاربر درخواست اطلاعات کند و اطلاعات قابل درک را به عنوان خروجی ، نمایش بدهد . درک ارتباط واضح و روشن بین کاربر و سیستم ، پیش نیاز طراحی یک واسط کاربر خوب است.
- یک واسط خوب از اشتباهات کاربر پیشگیری می کند و اطلاعات مهم را به خوبی آشکار می کند.
- یک واسط خوب دارای ویژگی سازگاری است ، یعنی اینکه به کاربران اجازه می دهد ، آموخته های خود را در انجام وظایف جدید به کار بگیرند.
- یک واسط خوب ، پیچیده نیست.طراحی های ساده ، به آسانی یادگرفته و استفاده می شوند و به سیستم قابلیت انطباق می دهند. در حقیقت طراحی خوب توازنی بین افزایش قدرت کارکرد و حفظ سادگی طی انجام فرآیند است [۴] همانگونه که آلن کوپر می گوید : " اینکه واسط شما چه جذابیت های خیره کننده ای داشته باشد ، مهم نیست ، بلکه هرچه ساده تر باشد بهتر است." [۲]

Archive of SID

www.SID.ir

- کاربران باید رابطه ی عمل و عکس العمل را بین کاری که می خواهند انجام دهند و نتیجه ی آن در صفحه ، ببینند. این باعث می شود که کاربران احساس کنند که در عملکردهای سیستم ، موثر هستند.
- واسط خوب باید کاربران را ببخشد : کاربران همیشه ممکن است مرتکب خطایی شوند ، بنابراین واسط باید همیشه امکان بازگشت را داشته باشد و انجام دوباره ی یک عمل ، آزمودن و یادگرفتن از خطاهای قبلی را برای کاربر امکان پذیر کند.
- یک واسط خوب باید ظاهر مطلوب و زیبایی داشته باشد. هر عنصری که در صفحه پدیدار می شود ، سعی در جلب نظر کاربر دارد اما نکته ی مهم در این رابطه این است که واسط باید محیطی را فراهم کند که کاربر آن را بپسندد و به درک بهتر کاربر از اطلاعات نمایش داده شده ، کمک کند. [۴]
- یک واسط خوب ، در زمان مناسب ، از متن و همچنین در جای مناسب ، از تصویر هم باید استفاده می کند.
- بهتر است هریک از موارد درون لیست ، یک نماد گرافیکی به عنوان شناسه ، در کنار خود داشته باشد.
- در تمام پنجره های محاوره (dialog box) ، ok و Cancle باید وجود داشته باشد.
- همیشه در سیستم قسمتی وجود داشته باشد که در آن فرآیند ها ی مورد نیاز به همراه مجموعه ای از راه حل های خوب برای انجام آن ها ، فراهم باشد. [۲]

استفاده از "تئوری همکاری" برای طراحی واسط و ارزیابی سیستم های ERP

تِروین [1] (۱۹۹۵) همکاری را فرآیندی معرفی می کند که طی آن ۲ عامل یا بیشتر با هم کار می کنند تا به اهداف مشترک دست یابند ، به گفته ی او همکاری انسان با کامپیوتر ، کاری است که در آن لااقل یک انسان و یک عامل کامپیوتری شرکت داشته باشند.

گِراوش [2] (۱۹۹۶) و شیبِر [3] (۱۹۹۶) ،پیشنهاد می کنند که سیستم کامپیوتری باید طوری طراحی شود که همچون یاور کاربر او را در جهت رسیدن به اهدافش ، کمک کند . این دیدگاه با چارچوب های ریاضی رسمی ، قبلا در طراحی و پیاده سازی نمونه های اولیه از واسط ها در اجتماع عامل های هوشمند ، استفاده شده است. که هیچکدام از آنها واسط هایی برای سیستم های گسترده ی اجرایی سازمان ،نبوده اند.

نکته ی قابل ذکر در رابطه با اصطلاح"یاور سیستمی" [4] اینست که فناوری کامپیوتر هنوز توانایی پیاده سازی یک همکار ، با توانمندی های یک انسان را ، ندارد بلکه همکاری سیستم با کاربر از طریق طراحی دقیق بر مبنای اصول همکاری ، میسر است. همانگونه که توسط براتمن [5] در ۱۹۹۲ و بعدها توسط گِراوش و کِراس [6] در ۱۹۹۶ به دقت توضیح داده شد ، این اصول عبارتند از :

---

[1] : Terveen
[2] : Grosz
[3] : Shieber
[4] : system-partner
[5] : Bratman
[6] : Kraus

Archive of SID

www.SID.ir

- **تعهّد به فعالیت مشترک :** هر کدام از دو طرف این فعالیت مشترک را به رسمیت بشناسند و متعهّد به انجام آن باشند. به عنوان بخشی از این تعهّد ، طرفین نیاز دارند که مفهومی را که همکاری آنها را احاطه کرده ، درک کنند چراکه در مشخص کردن جزئیات فعالیت ، مهم است.
- **پاسخدهی متقابل :** هر کدام از شرکا می خواهد ، رفتار خود را بر مبنای رفتار دیگری و مبتنی بر تعهد خود به فعالیت مشترک ، سازگار کند. پاسخدهی متقابل ، در رابطه با این تعهّد ، به این معناست که طرفین باید عملکردهای خود را برای رسیدن به نتیجه ی مشترکی که بهینه تر باشد ، با هم تطبیق دهند .
- **تعهد به پشتیبانی متقابل :** هر کدام از دو طرف باید متعهد به حمایت از تلاش های دیگری باشد. اگر یک عامل بفهمد که طرف مقابل در اجرای یک زیروظیفه نیازمند کمک است و این زیروظیفه مربوط به فعالیت مشترکشان است و او می تواند این کمک را فراهم کند ،آنگاه آن عامل آماده ی کمک و دستگیری است و طرف مقابل هم این همیاری را به رسمیت شناخته و از آن پشتیبانی می کند. تعهد به پشتیبانی مشترک همچنین بر ارتباطی دلالت دارد که هدف آن به اشتراک گذاشتن اطلاعاتی است که برای تکمیل یک فعالیت مشترک ، لازم است.
- **غربال زیربرنامه ها :** طرفین باید در پی تقسیم یک وظیفه به حلقه های در هم تنیده باشند به طوریکه هر زیر وظیفه ی مستقل در یک شبکه از زیر وظایف ، همانند حلقه ای است.پس طرفین باید درگیر رابطه ای شوند که طی آن این زیروظیفه های مستقل را در زمان های مشخص که مورد نیاز است ، با هم هماهنگ کنند.

شکل ۴ : اصول همکاری

برای نشان دادن اینکه این اصول چگونه رویکرد طراحی و ارزیابی سیستم ها را تغییر می دهند ، ما ابتدا یک روش ارزیابی قابلیت استفاده را که شناخته شده است ، توصیف می کنیم . این روش که "عبور شناختی" [1] نامیده می شود توسط وارتن [1] و

[1] : cognitive walkthrough

Archive of SID

www.SID.ir

همکارانش در ۱۹۹۴ ابداع شد. سپس نشان خواهیم داد که چگونه تئوری همکاری ، در نوع نگاه به تعامل بین کاربر و سیستم ، تاثیرخواهد داشت. مبنای روش عبور شناختی ، تئوری یادگیری اکتشافی است که CE+ نامیده می شود (پالسون و لوییس[۲] ، ۱۹۹۰) .این تئوری برای راهنمایی جهت طراحی واسط هایی که به راحتی قابل یادگیری باشند ، ایجاد شده بود که شباهت هایی هم با هدف ما در استفاده از تئوری همکاری دارد.

روش عبور شناختی شامل تحلیل گری است که با ایجاد یک سناریو از کاربرد یک وظیفه ی بخصوص ، یک واسط را ارزیابی می کند. در طول این روش ، سوالات زیر باید پاسخ داده شود تا موفقیت کاربری که آموزش ندیده در انجام عملکردی ویژه در سیستم، در هر مرحله ، تشخیص داده شود :

- آیا کاربرها سعی خواهند کرد که نتیجه ی درست را بدست آورند ؟
- آیا کاربرها توجه دارند که عمل صحیح ،امکان پذیر است؟
- آیا کاربرها می توانند عمل صحیح را با نتیجه ای که می خواهند به آن دست یابند ، پیوند بزنند؟
- اگر عمل صحیح اجرا شود ، آیا کاربر می تواند پیشرفت آن را در جهت انجام وظیفه و رسیدن به هدف ، نظاره کند؟ (وارتن و همکاران ، ۱۹۹۴)

پاسخ به این سوالات مبنایی خواهد بود برای پیش بینی اینکه آیا یک کاربر در انجام یک وظیفه ، موفق خواهد بود یا خیر .

امّا استفاده از تئوری همکاری ، فرمول این سوالات را تغییر خواهد داد .دقت داشته باشید که از دید تئوری همکاری ، سیستم به منزله ی یک یاور هم سطح کاربر ، در یک فرآیند عمل می کند.

برای مثال به اولین سوال روش عبور شناختی دقت کنید : " آیا کاربرها سعی خواهند کرد که نتیجه ی درست را بدست آورند ؟ " . این سوال برای این طراحی شده که توانایی کاربر را در تشخیص رابطه ی بین ساختار یک وظیفه با عملکردهای شناخته شده برای سیستم ، ارزیابی کند. مثالی که وارتن برای نشان دادن مفهوم این سوال به کار برده ، کاربری است که وظیفه ی او چاپ یک سند است ، اما او به این هدف نمی رسد مگر اینکه ابتدا یک چاپگر را بر گزیند. پس سوالی که ایجاد می شود اینست : " آیا کاربر می داند که قدم بعدی او باید انتخاب یک چاپگر باشد ؟"

ارزیابی این شرایط از دیدگاه همکاری ما را به نتیجه ی متفاوتی در زمینه ی سوال مرتبطی که باید بپرسیم ، می رساند . موضوع این نیست که آیا کاربر می داند که باید چاپگر را برگزیند یا خیر ، بلکه موضوع اینست که آیا سیستم می داند که قدم بعدی در فرآیند چاپ ، انتخاب چاپگر است و باید مطابق با این دانش عمل کند. پس سیستم باید از دستورالعمل کلی چاپ آگاه باشد، که ۲ عمل انتخاب چاپگر و فرستادن سند به آن را ، به هم متصل می کند. زمانی که کاربر هدف چاپ یک سند را شناخت، سیستم باید این امکان را به کاربر بدهد که چاپگر را انتخاب کند ، یا از طریق برگزیدن یک چاپگر از میان لیستی که خود سیستم قبلا ایجاد کرده ، یا با مشخص کردن یک چاپگر جدید . این همان تعهد به پشتیبانی متقابل است که باعث می شود سیستم با فرآیند انتخاب چاپگر ، به کاربر کمک کند. اینکه سیستم ، بدون مشورت با کاربر ، سند را به هر چاپگری بفرستد ، مطابق با اصول همکاری نیست ، بنا به دلایلی از جمله اینکه کاربر باید بداند که سند چاپ شده را از کجا بردارد و اینکه شاید گزینه ی پیش فرض ،در این نمونه ی خاص ، بهترین انتخاب ممکن نباشد. بنابراین اولین سوال با توجه به اصول همکاری به صورت زیر اصلاح می شود :

---

[1] Wharton :
[2] Polson & Lewis :

Archive of SID

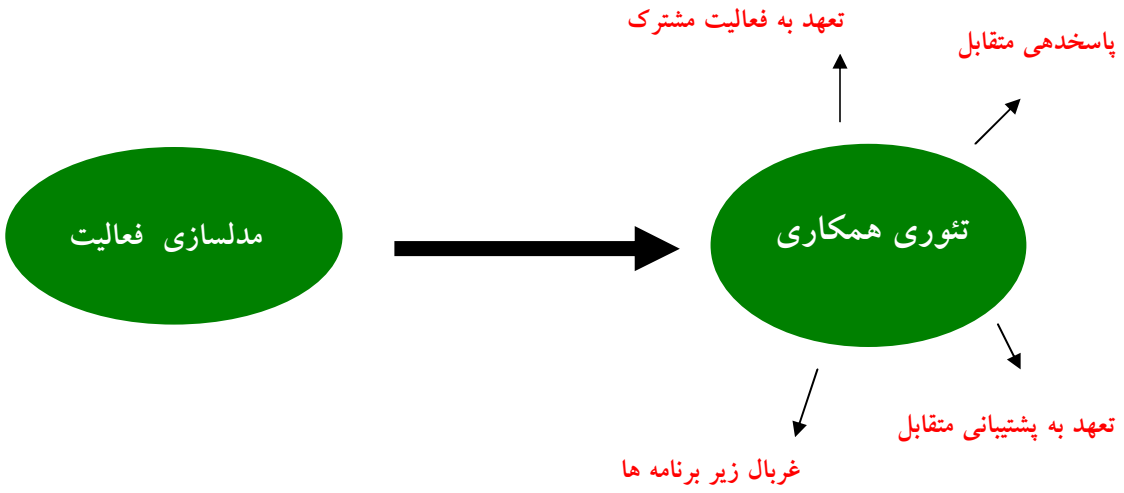
www.SID.ir

- با توجه به هدف کلی کاربر ، آیا سیستم قدم بعدی در فرآیند را خواهد شناخت ؟ و یا برای اجـرای آن قـدم ، عمـل خواهد کرد؟ ، یا اگر ورودی کاربر لازم است ، مجموعه ای از اعمال جایگزین را که کاربر می تواند از میان آنها انتخاب کند را نشان خواهد داد ؟

نگاهی مشابه از دیدگاه تئوری همکاری به سایر سوال ها ،ما را به پرسش های زیر می رساند :

- آیا سیستم به کاربر کمک خواهد کرد که عمل بعدی را بشناسد و آن را به طریقه ای بسیار آشکار ارائه بدهد ؟
- آیا سیستم یک مجموعه ی معنادار از عملکردهای جایگزین را بر اساس هدف کلی کاربر ، ارائه خواهد داد؟
- آیا سیستم ، کاربر را از عملکردهایی که پس از دیگری ،برای پیشرفت در جهت موفقیـت در یـک وظیفـه ، توسـط کاربر یا سیستم ،صورت می گیرد ، مطلع خواهد کرد ؟

در تبدیل سوالات روش "عبور شناختی" به صورتی که بازتاب دهنده ی "تئوری همکاری" باشند ، ما به جای اینکـه تنهـا بـر عملکرد، دانش و توانایی های یک کاربر تمرکز کنیم ، سیستم را به عنوان یک همکار در فرآیند ، دخیل کـردیم. در حـالی کـه مثال ما در اینجا برای ارزیابی (انتخاب چاپگر برای چاپ) در مقایسه با وظایفی که در محیط سازمانی مطرح است ، وظیفه ی بسیار کوچکی بود ، اما به خوبی نشان دهنده ی این مطلب است که حتی ساده ترین وظایف ، می توانند از به کاربردن اصـول همکاری سود ببرند. مزایای همکاری کاربر با سیستم ، در وظایف پیچیده تر سازمانی ، خیلی بیشتر خواهد بود ، بخـصوص در مواردی که ترتیب صحیح وقایع حتی برای تحصیل کرده ترین کاربران ، به آسانی قابل تشخیص نیست ، سیستم می تواند دانش و کمک فراهم کند. [۱]



شکل ۵ : مدلسازی فعالیت ، شبیه ترین رویکرد به تئوری همکاری

www.SID.ir

مثالی از یک تراکنش

برای اینکه کاربرد اصول همکاری را در حوزه ی ERP نشان دهیم ، مثالی از یک تراکنش کاری را در نظر می گیریم. در این سناریو نشان می دهیم که چگونه یکی از سیستم های ERP که اکنون به صورت گسترده استفاده می شود ، در برآوردن نیازهای یک کاربر ، ناموفق می ماند و اینکه چگونه می توان آن را ، برای رسیدن به اهداف کاربر ، به یک همکار مفید ، تبدیل کرد.

سناریو

اِستیو ،یک مهندس و کاربر جدید یک سیستم سازمانی بزرگ است. به عنوان بخشی از وظیفه اش، او باید یک قطعه ی سخت افزاری را سفارش بدهد.پس هدف او ایجاد یک درخواست خرید (Purchase requisition) در سیستم است ،اما او برای مشخص کردن قطعه ی مورد نظرش، دچار مشکل می شود زیرا آن قطعه در لیست اصلی قطعات (Material Master) ثبت نشده.

اگرچه افزودن قطعه ، در جایی از سیستم امکان پذیر است ،گزینه ی افزودن یک قطعه ی جدید به لیست اصلی، در بخش درخواست خرید ، وجود ندارد. در طراحی واسط کاربر بر اساس تئوری همکاری ، به نیازهایی که ممکن است در جریان ایجاد یک درخواست خرید ، پیش بیاید ، توجه می شود. بنا بر اصل پشتیبانی متقابل ، سیستم باید دسترسی آسان به هر فرآیند مرتبط را که پیش نیاز فرآیند دیگر است ، فراهم کند . همانند اضافه کردن یک قطعه ی جدید به لیست قطعات در فرآیند ایجاد یک درخواست خرید :

اِستیو درخواست خرید را ناتمام رها کرده ، قطعه ی مورد نظر را در لیست اصلی وارد می کند و دوباره اقدام به ایجاد یک درخواست خرید در سیستم می کند. برای ایجاد یک درخواست جدید در سیستم ، اِستیو مسیر زیر را دنبال می کند :

Logestics - Material Master - Purchasing – Purchase Requisition –Creat

او اطلاعات مربوط به تاریخ تحویل ، واحد صنعتی که باید قطعه به آنها تحویل داده شود ، مکان نگهداری و گروه خرید ، را وارد می کند.

هنگامی که اِستیو گزینه ی ok را می زند تا وارد پنجره ی بعدی شود ، سیستم پیغام می دهد که " فیلد روز (D) در تاریخ ، صحیح وارد نشده " . اِستیو به فیلد تاریخ بر می گردد و تلاش می کند که مشخصه ی تاریخ را اصلاح کند.خواندن فایل های Help در مورد فرمت های مختلف ، در توضیح اینکه چگونه گزینه های D ،T ،W یا M بر روی فرمت تاریخ ، اثر می گذارد ، ناکام می ماند (به خصوص از آن جهت که او نمی داند که استفاده از حرف T برای تاریخ ، بر مبنای زبان انگلیسی نیست) . او همچنان سردرگم است ، تا آنکه به طور اتفاقی به گزینه ورودی های ممکن برای فیلد تاریخ ، برخورد می کند. انتخاب این گزینه منجر به پدیدار شدن یک تقویم می شود که اِستیو می تواند یک تاریخ را در آن برگزیند که به صورت خودکار و صحیح وارد فیلد تاریخ می شود.

Archive of SID

www.SID.ir

واسط کاربر گزینه ی بسیار مفید انتخاب تاریخ از درون یک تقویم را دارد اما این گزینه عرضه نمی شود . این امکان حتی زمانی که سیستم خطای کاربر را تشخیص می دهد و اعلام می کند ، غیرفعّال می ماند .
دو اصل "پاسخدهی متقابل" و "تعهد به پشتیبانی متقابل" ، از سیستم می خواهند ، تا به جای اینکه صرفا کاربر را از خطایش مطّلع کند ، این امکان را هر زمان که می تواند ، عرضه کند:

پس از آن ، یک همکار به اِستیو پیشنهاد می کند که گزینه ی Model service specifications را انتخاب کند تا اسامی فعلی همه ی قطعاتی را که به همراه شناسه های عددی خود ، لیست شده اند ، مشاهده کند.اِستیو متوجه می شود که این گزینه برای بازبینی ، نظم و ترتیب و تایید ، بسیار مفید است و تصمیم می گیرد از آن استفاده کند.

اصل "تعهّد به پشتیبانی متقابل" ، از طرفین همکار می خواهد تا دانشی را که به موفقیت فعالیت مشترکشان مربوط است ، به اشتراک بگذارند. در مثال قبل با وجود اینکه نمایش اسامی به همراه شناسه هایشان ، در اطلاع رسانی به یک کاربر بسیار مفید است و برای سیستم نیز کار ساده ای است ، اما واسط کاربر بدون صدور یک درخواست مشخص ، آن را فراهم نمی کند. طبیعتا کاربرهای جدید از چنین امکاناتی در سیستم مطلع نیستند و از بهره بردن از آنها بی نصیب می مانند :

اِستیو پس از بازبینی، در سیستم تایید می کند که اطلاعاتی که وارد کرده از جمله واحد صنعتی مقصد ، صحیح است و به صفحه ای وارد می شود که از او می خواهد که قطعات مورد سفارش را، لیست کند .متاسفانه اِستیو شماره ی دقیق ID آن قطعه را که وارد لیست اصلی کرده بود ، فراموش کرده است.

اگر سیستم ، اِستیو را قدم به قدم دنبال می کرد که در هر قدم چه کرده است ، حالا می توانست با استفاده از این اطلاعات به تعامل فعلی ، رسیدگی کند ؛ بعد از آن، می توانست تشخیص دهد که با وارد کردن یک قطعه ی جدید ، مهندس احتمالا در ادامه ی فرآیند درخواست خرید خود ، به اطلاعات قطعه ، نیاز پیدا خواهد کرد.

اِستیو سعی می کند با مرور توصیفات قطعه ، با استفاده از گزینه ی مقادیر ممکن برای فیلد قطعه ، شماره ی ID را پیدا کند. پس از مدتی ، طی مرور فرآیند ،اطلاعات روی صفحه تغییر می کند. اِستیو نمی داند ، که کدام کار او منجر به این تغییر شده ، او حتی نمی داند که آیا اطلاعات مربوط به درخواست خرید در حال حاضر موجود است یا نه!

تغییر سریع محتوای صفحه ، این گمان را ایجاد می کند که فرآیند درخواست خرید ، نیمه تمام رها شده است. اِستیو در حال حاضر مطمئن نیست که آیا سیستم همچنان متعهد به فعالیت مشترک ایجاد درخواست خرید هست یا نه. این شرایط نشان دهنده ی نیاز به سیستم جهت هدایت قدم های آینده برای انجام یک وظیفه است . به همان اندازه مراحل طی شده و تعاملات فعلی اهمیت دارد. دو همکار باید اطمینان حاصل کنند که برنامه های متقابلشان برای رسیدن به اهداف مشترک ، با هم هماهنگ است :

بعد از ترس و اضطراب ناگهانی اولیه ، اِستیو می فهمد که می تواند با دکمه ی GO Back ، به لیست قطعات، در درخواست خرید ، دست یابد .

Archive of SID

www.SID.ir

اگر اِستیو می دانست که دقیقا در کدام مرحله از فرآیند است ، هیچ دلیلی نداشت که نگران شود. او می توانست بـه وسـیله ی سیستم از نقشه آگاه باشد و به راحتی بتواند به مرحله ی قبلی برگردد :

بیش از ۱۲ گزینه ی ممکن برای نمایش لیست قطعات وجود دارد ؛ تعدادی آنچنان زیاد ، که استفاده از تمام آنها برای اِسـتیو ، دشوار است.

اِستیو به تازگی اطلاعات مربوط به واحد صنعتی که قطعه برای آن سفارش داده شده را تکمیل کرده. سیستم باید بتواند استنباط کند که لیست قسمت های این واحد صنعتی برای جستجو ، سودمندترین است وشاید آن گزینـه را بـالاتر از سـایر جـستجوها برای قسمت ها ، بسنجد :

اِستیو در حالی که در گزینه های مختلف سردرگم است ،ناگهان به گزینه ای توجه می کند کـه قطعـات را بـه همـراه واحـد صنعتی شان نمایش می دهد و زمانی که لیست قطعات را برای واحد صنعتی مقصدشان مرور مـی کنـد ، مشخـصات قطعـه را تعیین می کند. به محض مشخص کردن تعداد ، اِستیو ایجاد این سند را به پایان می رساند. اما او مطمئن نیست که آیا اطلاعاتی که وارد کرده کامل است یا نه ، چراکه هیچ تاییدیه ای دالّ بر کامل شدن فرآیند دریافت نکرده است. بعد از مـشورت بـا چنـد همکار با تجربه ، او نتیجه می گیرد که درخواست خرید تکمیل شده و آن را ذخیره می کند.

سیستم می داند که فرآیند چه زمانی کامل شده و باید به وضوح این دانش را به کاربر منتقل کند.

مباحثه

همانطور که شواهد نشان می دهد ، زیادی حجم داده و وابستگی فرآیندها که در سیستم ERP وجود دارد ، عمدتاً از دید کاربر پنهان است. علاوه بر این ، گستردگی این سیستم ها باعث می شود که هیچ کاربری نتواند همه ی این وابستگی ها را بـشناسد. این در حالی است که کاربرها باید لااقل با تعدادی از این وابستگی ها آشنایی داشته باشند تا بتوانند نتیجه ی اعمال خود را به طور کامل ببینند ، تعیین کنند که مرحله ی بعدی چه کاری انجام دهند تا فرآیند کاری آنها انجام شود ، یـک خطـا را تـشخیص داده و برطرف کنند. اصل "پاسخگویی متقابل" از سیستم می خواهد اطلاعاتی را که برای رسیدن به هدف لازم است ، به طریقی واضح وموثر ، به اشتراک بگذارد.

در زمینه ی خطاهای انسانی و عدم اطمینان در انجام اعمال (همانطور که اِستیو در وارد کردن تاریخ دچار مشکل شـده بـود) خیلی مهم است که سیستم ERP ، به جای اینکه به سادگی فقط خطا را گزارش دهد ، روش های جایگزین یـک عمـل را بـا کاربر، در میان بگذارد. ارتباط موثر ، مشخصه ی یک همکاری موفق است و باید مکانیزم هایی در نظر گرفته شود تـا ارتبـاطی روشن را از طرف سیستم به سمت کاربر برقرار کنیم. صادقانه باید بگوییم که این رابطه بر مبنای فرهنگ کسب و کـاری اسـت که بر سازمان حاکم است.

اصول همکاری باید برای هدایت رفتار سیستم در هر دو زمینه ی ایستا و پویا ، استفاده شود. [۱]

Archive of SID

www.SID.ir

بنابر آنچه که گفته شد ، از دیگر ویژگی هایی که می توان بر اساس تئوری همکاری در طراحی واسط موثر به شمار آورد ، به شرح زیر است :

- سیستم باید مستقیما اطلاعات کافی را به کاربر بدهد تا از اشتباهات او جلوگیری کند.
- هرجا که سیستم از کاربر سوالی را می پرسد ، ارزش آن را دارد که پاسخ کاربر را در حافظه اش ثبت کند.
- سیستم باید در نظر داشته باشد که کاربر دانا اما پر مشغله است.به عبارتی کاربر می خواهد کارش را به سرعت اما با دقت هرچه تمام تر به پایان برساند .
- در همه ی مسیر ها باید امکان بازگشت به مرحله ی قبل ، وجود داشته باشد.
- سیستم باید به گونه ای باشد که سوالات مناسب و مورد نیازش را در زمان مناسب بپرسد یعنی نباید وضعیتی پیش بیاید که کاربر وجود اطلاعاتی را برای تکمیل تراکنش لازم بداند اما نداند که این اطلاعات مهم را کجا باید وارد سیستم کند.
- مهم است بدانیم ، زمانی که سیستم به کاربر می گوید که در انجام سلسله عملیات با شکست مواجه شده ، او احساس حقارت می کند و سیستم باید با در نظر گرفتن تمهیداتی ، حتی الامکان از این وضعیت پیشگیری کند.
- قسمتی در سیستم در نظر گرفته شود که کاربر بتواند مجموعه عملیات پرکاربرد و مهم برای خود را در قالبی مانند Favourites ذخیره کند تا مجبور نباشد هر بار پس از ورود به سیستم ، صفحات مورد نیاز خود را پس از باز کردن لیست های تو در تو بیابد .
- سیستم همیشه باید برای اتفاقات و پیشامد های احتمالی که ممکن است در شرایط مختلف بروز کند، آماده باشد و در زمان مناسب کاربر را برای رسیدن به هدفش راهنمایی کند.[۲]
- نکته ی دیگری که باید به آن توجه شود ، قرار دادن دستورات و عملیات مربوط به هم ، در یک دسته و گروه (مِنوی یکسان) است. به این معنی که از قبل در نظر بگیریم که در جریان یک فرآیند ممکن است لازم شود که عملیات خاصی صورت بگیرد که گرچه به طور معمول کمتر اتفاق می افتد اما زیر مجموعه ای از سلسله عملیات مورد نظر است . لذا باید از قبل در کنار بخش های مختلف تراکنش ما به عنوان یک گزینه ، قرار بگیرد. مهمترین حسن چنین دسته بندی اینست که کاربر در صورت نیاز به انجام آن عمل خاص ، در میان گزینه های فراوان منو های مختلف ، سردرگم نمی شود.بلکه در اسرع وقت و به راحتی می تواند گزینه ی مربوط به عملیات مورد نظر را یافته و مراحل آن را بپیماید.
- تشخیص یک چیز ، زمانی که آن را می بینیم به مراتب آسان تر از به خاطر آوردن آن است. به خصوص در زمان انجام یک فرآیند ، به خاطر آوردن هر قدم و اینکه چه اطلاعاتی در هر قدم مورد نیاز است ، ممکن است دشوار باشد. بنابراین دسترسی داشتن به اطلاعات مرتبط ، در همان زمانی که کاربر کاری را انجام می دهد ، ایده ی بسیار خوبی است. به طور کلی اطلاعاتی که با هم استفاده می شوند ، بهتر است در کنار هم به نمایش گذاشته شوند ، به جای اینکه کاربر مجبور باشد آنها را به خاطر بسپارد. بنابراین حتی می توان هدف اصلی هر صفحه را در یک یا دو جمله ، در بالای صفحه شرح داد تا در صورتی که کاربر سردرگم شد که دقیقا در چه مرحله ای قرار دارد و چه بخشی از کار را انجام داده است ، به خوبی از موقعیت فعلی و وضعیت خود در یک تراکنش اطلاع پیدا می کند .[۳]

Archive of SID

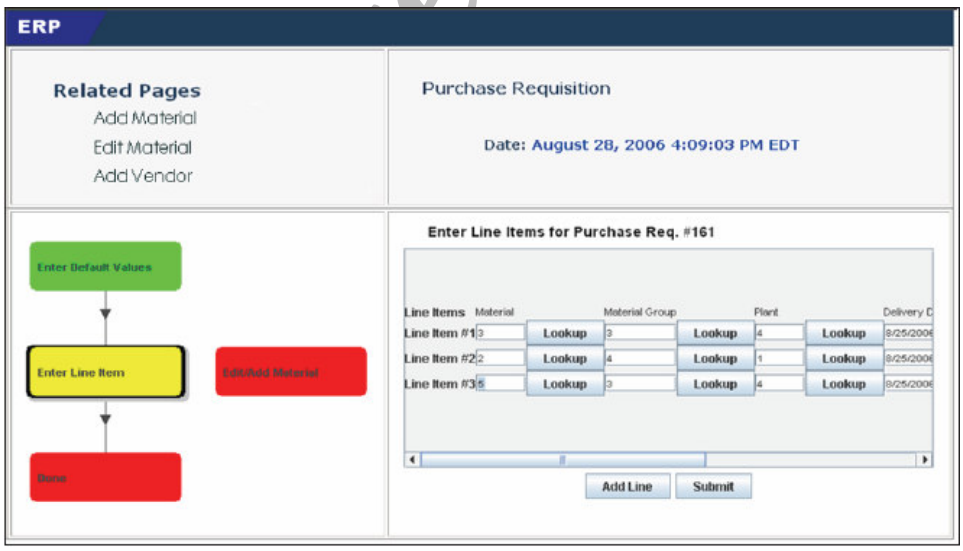
www.SID.ir

نمونه ی آزمایشی مصور

ما می خواهیم روی طراحی و پیاده سازی یک نمونه ی آزمایشی که شامل چندین گروه از وظایف ERP است ،کار کنیم تا نشان دهیم که چگونه اصول همکاری می تواند برای ساختن واسطی قابل استفاده و کارا جهت انجام وظایف سازمانی، به کار برده شود. تصویری که در شکل ۶ نشان داده شده است چندی از مهمترین ویژگی های این واسط را نشان می دهد.

این شکل لحظه ای از فرآیند ایجاد درخواست خرید ، توسط کاربر را نمایش می دهد. در سمت راست صفحه ، کاربر باید داده ها را وارد کند که در این مرحله از فرآیند لازم است. لذا مقادیر پیش فرض که توسط کاربر در مرحله ی قبل مشخص شده برای فیلد هایی مانند Material Group و Plant ، به صورت اتوماتیک توسط سیستم وارد شده اند اما می توانند پاک شوند. دکمه ی Lookup دسترسی به لیستی از مقادیر ممکن ،برای هر فیلد را فراهم می کند. اگر کاربر یک تاریخ نامعتبر را وارد کند ، مانند فیلد Delivery Date ، یک تقویم به طور اتوماتیک بر روی صفحه ظاهر می شود که کاربر می تواند از میان آن یک تاریخ معتبر را برگزیند.

سمت چپ صفحه، شامل ۲ قسمت است : صفحات مرتبط و گراف فرآیند. صفحات مرتبط شامل لینک ها به وظایفی است ، که بیشترین ارتباط را با فعالیت فعلی دارند و ممکن است برای تکمیل موفقیت آمیز فرآیند ، لازم شوند. فرض کنید در مثال قبلی ، اِستیو لینک Add Material را انتخاب کند که ممکن است صفحه ی قطعات جدید را به صورت جداگانه از پنجره ی درخواست خرید ، باز کند. این وضعیت اجازه می دهد که کاربر ، بدون اینکه رشته ی فعالیت کنونی را از دست بدهد ، امکان انجام وظایف مرتبط را داشته باشد.



شکل ۶ : پیاده سازی نمونه ی آزمایشی

گراف فرآیند ، مراحلی را که برای تکمیل بخشی از فرآیند کاری ، با استفاده از سیستم لازم است ، به تصویر می کشد. در این مورد ، گراف ، وظایف مورد نیاز برای ایجاد یک درخواست خرید را نشان می دهد. وضعیت هر یک از وظیفه ها به وسیله ی

www.SID.ir

مقیاسی از رنگ نشان داده شده است . رنگ سبز نشان دهنده ی " کامل بودن" است ، زرد ، " در جریان پیشرفت بودن " را نمایان می سازد و قرمز می گوید که "هنوز شروع نشده است". در این شکل کاربر قبلا مقادیر پیش فرض را وارد کرده و اکنون مشغول وارد کردن Line Item است. خط مشکی دور گره Enter Line Item نشان می دهد که این مرحله در حال حاضر فعال است.کاربر ممکن است به مرحله ی قبل برود که چیزی را مرور کند و یا تغییر دهد ، او می تواند مراحل بعدی را ببیند (اما تا زمانی که همه ی مراحل قبلی کامل نشده باشند ، نمی تواند داده وارد کند) .

سایر وظایف در صورت نیاز در گراف فرآیند ، ظاهر می شوند.

هنگامی که فرآیند درخواست خرید تکمیل شد ، همه ی گره های گراف ، سبز خواهند شد ، بجز گره Done که فعال و به رنگ زرد است.کاربر ، این تذکر را دریافت می کند که فرآیند کامل شده است و او می تواند پنجره ی درخواست خرید را ببندد یا مراحل قبل را بازبینی کند و تغییر بدهد. این وضعیت سردرگمی اِستیو را درباره ی عدم اطمینان از تکمیل فرآیند ، از بین خواهد برد و به او اجازه می دهد که با اعتماد به نفس ، سایر کارهایش را پیش ببرد.

نمونه ی آزمایشی که در اینجا نشان داده شده است ، چند مرحله ی ساده را نشان می دهد که می توانند قابلیت استفاده ی واسط ERP را برای یک وظیفه ی خاص بر مبنای اصول تئوری همکاری ، افزایش دهد.خیلی مهم است به این موضوع توجه کنیم که باارزش ترین دستاورد به کاربردن این اصول در واسط های ERP ، توانایی سیستم در پشتیبانی از کاربران ، برای انجام دامنه ی گسترده ای از وظایف است[۱]

نتیجه گیری و کار آینده

بهبود بخشیدن به قابلیت استفاده ی سیستم های ERP ، منافعی را فراهم می کند که با رسیدن کاربران به هدف هایشان ، توسعه پیدا می کند. این بهبود نهایتاً ، از طریق کاهش زمان آموزش ، افزایش رضایت کارمندان و ایجاد اطلاعات ارزشمند به کل سازمان ، منفعت می رساند. این مقاله ، استفاده از اصول همکاری را در توسعه ی سیستم ها پیشنهاد می کند، که موجب می شود ابزاری ، برای پر کردن خلاء بین قابلیت های سیستم ERP و مهار کردن آن قابلیت ها برای برآوردن اهداف هر کاربر ، فراهم شود . همانگونه که دیدید تئوری همکاری ،چارچوب مفهومی بسیار مناسبی است که می تواند به طور مؤثر برای هدایت طراحی و ارزیابی تعامل کاربر با سیستم ، در زمینه ی سیستم های سازمانی بزرگ ، استفاده شود.

طراحی برای بالا بردن قابلیت استفاده با تغییر بخش محدودی از واسط کاربر ، به دست نمی آید. همکاری را نمی توان وصله زد! بلکه باید از ابتدا تا به انتهای کار ، به دقت طراحی وپیاده شود. پیاده سازی فرهنگ همکاری در تعامل ، نیازمند پشتیبانی مناسب در Data model و Algorithmic module های یک سیستم است.

ما معتقدیم که تئوری همکاری یک مبنای بسیار خوب برای طراحی قابلیت استفاده و ارزیابی است .

تحقیقات آینده بر روی ERP و قابلیت کاربرد سیستم سازمانی باید علاوه بر بحث در مورد تاثیر واسط های ERP بر تصمیم سازی سازمانی ، در مورد عوامل تکنیکی طراحی واسط کاربر هم سخن بگوید.

مراجع :

Archive of SID

www.SID.ir

[1]: Angappa Gunasekaran ; "Techniques and Tools for the Design and Implementation of Enterprise Information Systems "; University of Massachusetts-Dartmout ;USA , New York ; IGI Publishing ; 2008.

[2]: Alan Cooper ; " About Face :The Essentials of User Interface Design" ;USA , California ; IDG Books Worldwide, Inc ;1995.

[3]: Clayton Lewis , John Rieman ;"Task-Centered User Interface Design A Practical Introduction "; University of Colorado, Boulder ; 1993,1994.

[4]: "What makes a good user interface?"; http://ou800doc.caldera.com/en/SDK_vtcl/vtclgN.style_goodui.html; Retrieved November 10, 2008;

[5]: Topi, H., Lucas, W., & Babaian, T; "Using informal notes for sharing corporate technology know-how"; Europian journal of information systems ; 15(5); 2006; 486-499.

Archive of SID

www.SID.ir