

DA “RESISTÊNCIA AO USO” AO CONTEXTO DE USO: compreendendo a não utilização de sistemas informatizados

Rosângela Maria de Almeida Camarano Leal¹, Francisco de Paula Antunes Lima²

¹ Universidade Federal de São João del-Rei

² Universidade Federal de Minas Gerais

rocam@ufsj.edu.br, fpalima@ufmg.br

Resumo. O presente artigo baseia-se em um estudo de caso sobre o desenvolvimento e uso de um *software*, módulo “Lançamento de Horas”. O estudo de caso mostrou as limitações do método de construção de *software*, empregados pelos projetistas, quando estes não levam em conta a participação dos usuários na concepção do *software*, que terá implicações negativas para a atividade de trabalho desses usuários. Mostrou, também, a pertinência da aplicação da análise da atividade e da entrevista de autoconfrontação na identificação das dificuldades de utilização do *software* pelos usuários. O aspecto inovador do estudo de caso consiste em estudar um *software* desenvolvido não para usuários leigos, mas para usuários especialistas em informática. Assim, o estudo evidenciou que a identidade sociofuncional entre analistas-desenvolvedores e os usuários-especialistas (também desenvolvedores) permitiu identificar as representações sociocognitivas comuns, que remetem aos preconceitos do cotidiano, ao senso comum como justificativa e racionalização dos problemas de usabilidade.

Abstract. The present paper is based on a case study on the development and use of a *software* module “Launching of Hours”. This study had as an objective to demonstrate the inappropriateness of the methodological procedures, used by the designers when they do not take into account the participation of the users on the software conception, which will have negative implications on the working activity of these users. It demonstrates, also, the pertinence of the application of the activity analysis to show the difficulties in using the software by the users. The novelty of this case study is to present a *software* development not for laic users, but for users that are experts on information technology. Thus, the study showed clearly that the sociofunctional identity among analysts-designers and users-experts (also designers themselves) allowed us to identify the common sociocognitive

representations, which send us back to everyday prejudice, to common sense as justification and rationalization of problems related to usability.

Palavras-chave: saber tácito, saber social, cognição situada, análise da atividade, entrevista de autoconfrontação, estudo de usuários.

1. Introdução

Um dos argumentos recorrentes entre os especialistas em informática para a não utilização de sistemas de informação por usuários leigos recai sobre as supostas deficiências e limitações dos próprios usuários, seja para expressar suas necessidades (por exemplo, a tese da insuficiência linguística), seja por falta de conhecimento para se servir das novas tecnologias. Pode-se provar que essa responsabilização do usuário é infundada se se conseguir demonstrar que a não utilização ocorre também entre os próprios analistas quando desenvolvem sistemas para eles próprios.

Um estudo de caso realizado com usuários em uma empresa que desenvolve sistemas de informação, mais especificamente um módulo desenvolvido para o Lançamento de Horas em um sistema interno de gestão de projetos que funciona na intranet, nos permitiu mostrar que a “resistência” ao uso se explica, de fato, por inadequações dos *softwares*, decorrentes de limitações das técnicas convencionais de identificação das reais necessidades dos usuários de sistemas de informação. O recurso a entrevistas, a utilização dos questionários e a observação espontânea não permitem uma imersão no contexto em que são construídos os significados que os sujeitos atribuem a situações cotidianas. São essas significações que possibilitam uma melhor compreensão das causas da suposta “resistência” ao uso e à especificação de sistemas que atendam às suas necessidades, em especial quanto à usabilidade, que envolve não apenas as interfaces imediatas, como também a lógica e as funcionalidades dos sistemas de informação.

A situação analisada revelou-se particularmente interessante por se tratar de um sistema desenvolvido para uso dos próprios especialistas em informática, servindo de apoio à gestão interna dos projetos, portanto, sendo usado apenas pelos analistas e programadores que desenvolvem outros sistemas. No caso da explicitação das necessidades, encontramos os mesmos problemas com os usuários-especialistas, o que se explica não pela insuficiência linguística dos analistas de sistema, mas, sim, pelas limitações dos métodos de definição de requisitos que utilizam apenas entrevistas baseadas no senso comum, insuficientes para explicitar certas dimensões tácitas do conhecimento e do comportamento dos sujeitos. A articulação entre entrevista de autoconfrontação e análise da atividade possibilitou a explicitação do saber tácito do usuário, que permanece escondido a um olhar menos cuidadoso. Sendo assim, em consonância com estudos empíricos desenvolvidos pela ergonomia da atividade, perguntar aos usuários quais são as dificuldades de sua tarefa não é suficiente, pois vários dos problemas já se tornaram “naturais”, ou seja, não são mais percebidos como situações problemáticas (LIMA, 1998; sobre a “naturalização” das dificuldades para se realizar um trabalho, ver a Guérin, 1991).

Observou-se que as explicações para os problemas relacionados com a utilização de um *software* em contexto de trabalho não estão baseadas na atividade profissional, mas se apoiam no senso comum. O comportamento do usuário, ainda que este seja um especialista em informática, é interpretado a partir de categorias do saber cotidiano, às vezes generalizações preconceituosas, revelando uma lacuna metodológica que não permite romper esse ciclo vicioso: como as dificuldades são atribuídas à psicologia do senso comum (“todo mundo é igual”, “brasileiro é assim mesmo” etc.) sobre a qual se constrói um consenso entre desenvolvedores e usuários-desenvolvedores, não surgem conflitos que coloquem o sistema de informação como problema, o que bloqueia o desenvolvimento de alternativas mais adequadas à natureza da atividade. Isso não quer dizer que o saber do cotidiano, do senso comum, seja “arbitrário, subjetivo ou falso,

opinativo, ou tudo mais que se possa contrapor à ‘objetividade’ da ciência”, mas que são conhecimentos diferentes, com alcances práticos também diferentes (LIMA, 1998, p. 6).

O estudo também permitiu explicitar que as dificuldades práticas no uso desse *software* pelos usuários-desenvolvedores foi consequência da metodologia, não estando, portanto, na consciência imediata dos usuários. Ou seja, o desvendamento das dificuldades na utilização do módulo “Lançamento de Horas” foi possível a partir da análise da atividade e da entrevista de autoconfrontação, realizadas em momentos diversos, ao longo de uma pesquisa que se desenrolou durante 12 meses. Em média, foram observadas 2 jornadas de trabalho durante cada mês.

1.1 Sobre a Empresa

Trata-se de uma empresa que, desde 1987, se dedica à concepção de sistemas de informação, equipamentos e serviços para clientes de diversos segmentos empresariais, no Brasil e no exterior (África do Sul, Argentina, Tailândia, Irlanda do Norte, Inglaterra e México, dentre outros), tais como: alimentos, bebidas e fumo, alumínio, cimento e cal, logística (portos e ferrovias), metais, mineração e fertilizantes, óleo e gás.

O gerenciamento e o acompanhamento dos projetos da empresa, tanto internos quanto externos, são feitos por meio do *software* denominado SPAP (Sistema de Planejamento e Acompanhamento de Projetos), objeto do nosso estudo, concebido no Departamento de Tecnologia da Informação. A pesquisa consistiu em analisar as dificuldades na interação entre usuário/*software*, em especial com o módulo “Lançamento de Horas”. A escolha deste se deu em função de ser um módulo, diferentemente dos outros módulos, utilizado por todos os funcionários, exceto os gerentes, e que apresentava problemas atribuídos à “resistência” dos usuários; nesse caso, como dissemos, que são também desenvolvedores (analistas e programadores) de sistemas informáticos. Os usuários deveriam alimentar, todos os dias, o *software* com

informações referentes às horas trabalhadas e às horas gastas nas tarefas de desenvolvimento de projetos. Essas informações, para quem gerencia os projetos na empresa, servem para subsidiar ações de controle, gestão dos custos, negociação com o cliente, gestão do tempo dos projetos e acompanhamento do cronograma de execução dos projetos desenvolvidos pela empresa. No entanto, esses usuários-desenvolvedores se atrasam no lançamento dessas informações, causando, assim, dificuldades para o gerenciamento dos projetos da empresa.

Quanto aos sujeitos do estudo, eles são os usuários do sistema, os próprios funcionários da empresa, que trabalham na área técnica no desenvolvimento de *software*, ou seja, os desenvolvedores ou programadores. Dentre estes, 20, de um total de 273 sujeitos, foram selecionados para participar do estudo. A seleção dos sujeitos ocorreu a partir da análise dos *e-mails*, encaminhados aos responsáveis pelo *software*, que pediam ajuda ou relatavam problemas práticos de uso do módulo “Lançamento de Horas”.

A totalidade dos sujeitos são homens, em sua maioria concentrados na faixa etária entre 24 e 34 anos (90%). O tempo de serviço é de 1 a 5 anos. Com relação à escolaridade, prevaleceu o nível de usuários com o grau superior completo e incompleto nas áreas da Ciência da Computação, da Engenharia de Automação e um caso da Economia.

1.2. Sobre a Metodologia

Em termos metodológicos, procurou-se obter respostas para o problema de pesquisa que vai além das abordagens que têm unicamente a entrevista como procedimento de coleta de dados. As discussões, bem como as análises, foram realizadas ao longo do processo investigativo, recorrendo, sobretudo, ao método de entrevista em autoconfrontação.

Portanto, privilegiou-se como categoria de análise o fazer do usuário, e não unicamente o seu discurso, o que ele fala ou o que ele pensa dos problemas e das dificuldades que tem ao interagir com o *software*. A produção dos dados que constitui o *corpus* de análise se deu pela mediação da observação do fazer dos usuários no trabalho, e não somente pela entrevista, ou seja, estabeleceu-se o ir e vir entre os discursos dos usuários e a realidade concreta de trabalho.

2. Dos resultados

Os resultados da pesquisa, ao estudar o desenvolvimento de um *software* construído não para usuários leigos, mas para especialistas, mostraram que essa convivência no âmbito de uma comunidade de especialistas em informática não é suficiente para o emprego eficaz dessa tecnologia. Assim, a identidade sociofuncional entre analistas e usuários-desenvolvedores permitiu identificar as representações sociocognitivas compartilhadas, que remetem aos preconceitos do cotidiano e ao senso comum como justificativa e racionalização dos problemas de usabilidade.

Pode-se perceber que, quando apresentados aos usuários seus próprios conhecimentos, um processo de tomada de consciência se instaura, estabelecendo uma condição crucial para a desestruturação e a reconstrução das representações do trabalho. Nesse sentido, ao passar do discurso para a ação – o estudo do fazer dos usuários, o que implicou em compreender a atividade de trabalho dos usuários-desenvolvedores –, o que se ganha é o acesso ao que está latente, ao que está implícito.

Assim, colocam-se em xeque os métodos de pesquisa tradicionais ao evidenciar o viés inerente aos procedimentos classificatórios das estatísticas e questionários e a arbitrariedade das perguntas formuladas pelos investigadores, que refletem mais as preocupações destes do que os reais interesses dos entrevistados. Portanto, o emprego de entrevistas dirigidas ou o uso de questionários é questionável ao imporem uma problemática ao sujeito, tendo como consequência a pobreza da resposta, a indução da

resposta pela formulação da pergunta e o desconhecimento de seus quadros de referência (THIOLLENT, 1985).

Nossa proposta é que se adote a perspectiva da atividade de trabalho para uma melhor compreensão do sentido que o usuário atribui ao seu comportamento. Dessa forma, é possível ultrapassar limitações das técnicas convencionais baseadas unicamente no discurso dos usuários, que, assim, são considerados apenas como informantes. Ou seja, o usuário é visto como detentor de uma informação já acumulada em sua memória e transmissível oralmente sempre que lhe for solicitada, independentemente da situação. As entrevistas normalmente são realizadas fora da situação real de trabalho ou quando a atividade se interrompe. No caso da empresa estudada, o usuário é subtraído do fluxo normal de sua atividade para prestar informações ao desenvolvedor ao invés de este procurar se inserir no fluxo da atividade do usuário.

Ao contrário, as verbalizações em autoconfrontação estão diretamente relacionadas com a dinâmica do curso da ação, diferentemente, portanto, das formas tradicionais de entrevista. Quando se entende a cognição de forma situada, como demonstram os estudos empíricos desenvolvidos no âmbito da ergonomia da atividade, pode-se, então, buscar otimizar as características do dispositivo técnico, tendo como referência os processos cognitivos de uma determinada população. Como ressaltam Abrahão *et al.* (2005), “quanto mais se incorpora o usuário na (re)concepção de sistemas informatizados, mais se aproximam da lógica de funcionamento a interface e a linguagem às representações e às estratégias que eles utilizam em sua ação” (p. 170).

A pesquisa de campo revelou que, muitas vezes, os projetistas fundamentam-se em representações do cotidiano, o que impede uma representação baseada na atividade. Não levam em conta a comunidade, as regras e a divisão de trabalho em que a tecnologia será implementada. Nesse tipo de abordagem projetual, todo o direcionamento está voltado para uma ideia preconcebida de um artefato, cujos

significados podem ser definidos muito mais pelos contextos daqueles que projetam do que pelas experiências de vida dos usuários. O contexto de uso como referência e a consideração dos aspectos culturais que envolvem a relação sujeito/artefato podem levar ao desenvolvimento de outros tipos de produtos, impossíveis de serem previstos sem o estudo da atividade de uso. Um artefato se torna um verdadeiro instrumento quando se inscreve numa utilização, quando é um meio para o usuário poder realizar determinado objetivo.

O desenvolvimento desses sistemas tem colocado em evidência que a lógica da tecnologia acaba se sobrepondo à lógica do usuário. Por esse motivo, são constantes as alterações nos *softwares* com o objetivo de resolverem falhas das versões anteriores decorrentes do fato de não se levar em conta as características dos usuários, seus conhecimentos, suas competências e a natureza da sua atividade.

A atividade de “Lançamento de Horas”, tanto no ponto eletrônico quanto na tarefa, é uma atividade secundária, não é atividade-fim dos usuários-desenvolvedores e acaba por roubar recursos cognitivos, como tempo, atenção e memória e praticidade do sistema. Por isso, os próprios usuários também recorrem à explicação em termos psicologizantes: “preguiça”, “esquecimento”, “chato”, “trabalhoso”. Compartilham a representação de quem concebe o *software* sobre a utilização ou não do *software*, que não está baseada na atividade profissional, mas se apoia no cotidiano, no senso comum.

Sendo assim, as explicações dos desenvolvedores de *software* sobre as dificuldades práticas relacionadas ao seu uso, principalmente àquelas que se referem ao atraso no lançamento de horas, são sempre formuladas em termos psicológicos e atribuídas a uma suposta “natureza humana”. Como tal, não podem ser mudadas, apenas controladas do exterior pela força da recompensa ou da punição, o que acaba por naturalizar os problemas e impedir que sejam objeto de um novo projeto de concepção.

Dessa forma, os problemas relacionados ao atraso no lançamento de horas estariam relacionados a conceitos como “resistência à mudança”, “preguiça”, “falta de

interesse” e “esquecimento”, não estando relacionados diretamente à atividade de trabalho dos usuários e tampouco às especificidades do instrumento informatizado, mas, sim, aos traços de personalidade ou atitudes pessoais mesmo que o usuário seja um especialista em informática.

Essa resistência, que se manifesta em comportamento, como “deixar para fazer na última hora” ou mesmo esquecer de lançar as horas, na verdade, não se explica pela “desorganização”, mas decorre, sobretudo, da incompatibilidade entre a atividade de trabalho dos desenvolvedores, que é de natureza cognitiva, e a atividade de lançamento de horas, que é de natureza administrativa, ou seja, não é atividade-fim do sujeito. Ela não está integrada à atividade dos usuários. Trata-se, sim, de uma atividade que é dirigida a um outro, atividade de controle gerencial, não sendo uma atividade integrada na atividade dos usuários-desenvolvedores.

A atividade de desenvolvimento de *software* é uma atividade intelectual, complexa, que exige atenção, concentração, criatividade, sendo, portanto, incompatível com a atividade de lançar horas tanto no ponto eletrônico quanto na tarefa. Algumas das dimensões da complexidade das tarefas dizem respeito à lida com abstrações lógicas, manipulando um número grande de dados, normalmente em ambientes sujeitos a múltiplas interferências. Assim, pela própria natureza da atividade de programação, que exige certo grau de criatividade, o principal meio de trabalho (a despeito das metodologias atualmente existentes) é a capacidade intelectual, entendida aqui como o domínio de certos conhecimentos de informática e de uma prática profissional capaz de fundi-los em modelos novos para novas situações (TAVARES, 1983, p. 223). Como nos disse um *tester*,

o trabalho exige muita atenção. Eu tenho que ser muito crítico. Tenho que ter muita paciência. Às vezes, a gente fica muito irritado, porque um erro, você já mostrou para o desenvolvedor 10 vezes e ele continua acontecendo. Você tem que ter muita paciência.

Pode-se dizer que os usuários se encontram diante de um tempo subjetivo e um tempo objetivo. O primeiro refere-se à atividade dos usuários, o desenvolvimento de *software*, no qual o sujeito se coloca por inteiro, e, no segundo caso, é a gestão do externo, das contrariedades, é a divisão do sujeito. Como nos disse um desenvolvedor, “*desenvolver é um trabalho que está dentro da cabeça da gente, que depende muito da nossa concentração, da nossa criatividade*”.

A criatividade da atividade dos desenvolvedores está relacionada em conseguir construir regras formais a partir das demandas específicas do usuário, considerando as situações singulares em que este atua. [Por esse motivo,] a geração de *software* não é apenas a codificação de um conjunto de dados e regras, trata-se de uma atividade que exige transformar uma linguagem prática em uma linguagem formal, de códigos, o que torna sua atividade complexa e difícil (LEAL e LIMA, 2006, pp. 113-114).

Dada a complexidade que envolve a atividade dos desenvolvedores, “*quando a gente perde a linha de raciocínio, quando, por exemplo, a gente é interrompido, é muito estressante*” (desenvolvedor).

A interrupção na atividade de trabalho do desenvolvedor para o preenchimento do lançamento de horas é entendida como mudanças de orientação cognitiva provocadas pelas paradas na atividade que estava sendo realizada pelo usuário.

Você interromper o que você está fazendo para lançar hora é um saco. Este negócio de, por exemplo, 10 minutos fazendo uma coisa no contrato, tem que lançar, se ficar uma hora em outro contrato tem que lançar. É um saco (Desenvolvedor).

Este negócio de ficar lançando tudo que você fez me dá a maior preguiça (Desenvolvedor).

Às vezes, acontece da gente trabalhar tanto que a última coisa que lembro é de lançar hora (Desenvolvedor).

Para Norman (1991), as interrupções no fluxo da atividade exigem um esforço consciente de atenção sobre a tarefa. A perturbação traz à consciência a atividade perturbada mesmo quando esta não é mais o centro da atenção principal. Esse fenômeno pode ter efeito negativo sobre a tarefa a executar. As perturbações desse tipo podem levar a erros desde que a atividade perturbada interfira sobre a manutenção da memória ativa para a tarefa. Como nos disse um desenvolvedor, “*quando interrompe o que a gente está fazendo, ao retornar, fica difícil, a gente perde o foco*”.

Por esse motivo é que a grande maioria dos usuários observados (95%) faz o lançamento de horas de única vez, seja no final do dia, ou no início do outro dia, ou somente no dia do bloqueio, o que diminui o tempo de interrupção da atividade, diminuindo o custo cognitivo que a interrupção acarreta. Além disso, o lançamento de hora exige atenção e raciocínio.

É no final do dia que eu faço o lançamento. Eu tenho outras coisas para fazer, ao invés de ficar vindo no SPAP toda hora. Eu prefiro lançar tudo de uma única vez (Desenvolvedor)

Esse negócio de lançar um pedaço, pára, lanço outro pedaço, pára, lança outro. entendeu? É chato. Lançar de uma vez, significa que você só está ligado só no lançamento (Desenvolvedor).

Ao contrário do que pensam os analistas de sistemas os motivos do atraso se vinculam à inadequação da concepção do *software*, módulo “Lançamento de Horas”, que desconsidera as características dos usuários-desenvolvedores. Portanto, as razões que levam os usuários ao atraso no lançamento de horas não podem ser atribuídas a fatores unicamente de natureza psicológica, como “resistência”, “preguiça”, “esquecimento”.

É por esse motivo que o usuário só se lembra de lançar as horas quando já está saindo do escritório da empresa; por exemplo, indo para a faculdade, ou já no elevador, ou quando está indo para casa, ou seja, apenas quando o usuário se “desliga” da sua atividade-fim é que a tarefa de lançar horas vem à sua mente.

A inadequação na concepção do *software* exige adequação desse artefato à atividade de trabalho dos usuários, demandando a elaboração de estratégias e de outros instrumentos que assegurem a realização do trabalho. Portanto, a concepção inadequada do *software* induz à criação de instrumentos paralelos. Estes funcionam como uma ajuda para a execução da tarefa (função instrumental). Nesse sentido, eles funcionam como alternativas reais de transformação dos objetos da atividade, como suporte para ações dos usuários para gerir as dificuldades decorrentes da inadequação do *software*.

Durante as observações, verificou-se a utilização de um caderno de anotações ao lado do computador. A escolha do caderno apareceu como estratégia para outros cinco dos 20 desenvolvedores observados. A principal vantagem que o caderno oferece com relação a outros instrumentos está relacionada ao fato de poder, segundo os entrevistados, “ficar perto do computador, ao lado, é só pegar a caneta e anotar, não tem que entrar em nenhuma tela”. Além disso, anotar as horas no caderno funciona como um “lembrete” que auxilia os usuários a recuperarem as informações na memória.

Outros desenvolvedores preferem o uso de uma planilha para registrar as tarefas executadas e o tempo gasto em cada tarefa. Trata-se de um programa, *Excel*, no qual também são lançadas as horas na tarefa. Um outro usuário criou um sistema que tem como função contar horas das tarefas que estão sendo realizadas. Para ele, o sistema tem a função de ser um “lembrete” para ajudá-lo no lançamento de horas. Nesse sentido, “facilita por ser um lembrete porque antes eu teria que colocar num caderno ou numa planilha, mas, na hora de lançar, eu não tenho alternativa, tenho que criar a tarefa e lançar”.

A criação de instrumentos paralelos pode ser entendida como estratégias cognitivas para dar conta das exigências da tarefa de lançamento de horas. Eles funcionam como “recurso informal” para reduzir a carga de trabalho e, além disso, exercem a função mediadora na interação sujeito-meio, ajudando o usuário a construir o seu modo operatório.

Durante as observações, verificou-se, ainda, que é comum a utilização da música como uma estratégia de se isolar das interferências presentes, como: telefone tocando ou um colega que conversa ao lado. O distanciamento proporcionado pela música permite que *“a gente fique isolado e pode concentrar-se no nosso trabalho. Senão, é difícil concentrar, é telefone tocando toda hora, é gente te perguntando as coisas, é gente chamando para tomar café, é gente ao seu lado conversando”* (desenvolvedor). Portanto, ouvir música e concentrar-se na tarefa é decorrente da prática, uma vez que

muitos processos que inicialmente requerem atenção e concentração se tornam automáticos com a prática. Na verdade, o automatismo provavelmente é uma questão de grau – não somos completamente automáticos, mas antes, mais ou menos automáticos em nosso desempenho, dependem de quanta prática tivemos (NORMAN, 1991, p. 224).

A funcionalidade da música serve como uma barreira que os ajuda a evitar a distração que os usuários não controlam.

Concluindo, acredita-se que foi possível mostrar a pertinência da análise da atividade e da entrevista de autoconfrontação, para perceber os sentidos latentes das ações e atitudes dos usuários-desenvolvedores. As conclusões do estudo se referem às particularidades da empresa pesquisada, o que impede que os resultados possam ser generalizados em toda a sua extensão. No entanto, a questão do conhecimento tácito, a relação entre especialistas e a relação entre atividade principal e secundária podem ser analisadas em outras situações e podem servir de modelo geral para explicar a “resistência”. Por outro lado, ainda não foi esgotada a análise de todas as implicações do

fato de se tratar de usuários não-leigos, mas especialistas em informática, e, sobretudo, a questão de como integrar atividades secundárias ao fluxo da atividade-fim, situação que é frequente em várias outras profissões, nas quais se convive com uma atividade altamente absorvente e tarefas secundárias que não têm interesse imediato ou direto.

3. Referências

- Abrahão, J. I.; Silvino, A. M. D.; Sarmet, M. M. (2005). Ergonomia, Cognição e Trabalho Informatizado. *Psicologia: teoria e pesquisa*, Brasília, v. 21, n. 2. pp. 163-171, maio/ago.
- Guérin, F. et al. Compreender o trabalho para transformá-lo. São Paulo, Edgard Blücher. 2001
- Leal, L.; Lima, F.P.A. (2006). O analista de sistema, o artesão e a fábrica: os equívocos do modelo da “fábrica de software”. *14º Congresso Brasileiro de Ergonomia, 4º Fórum Brasileiro de Ergonomia e 2º Congresso Brasileiro de Iniciação em Ergonomia*, Curitiba.
- Lima, F. A. (1998). *Fundamentos teóricos de metodologia e prática de análise ergonômica do trabalho*. Belo Horizonte. Departamento de Engenharia de Produção (mimeo).
- Norman, A. (1991). Les artefacts cognitifs. In: CONEIN, Nicolas Dodier; LAURENT, Thévenot. *Les objets dans l'action*. Paris: Ed. de L'École des Hautes Etudes en Sciences Sociales, p. 15-34.
- Tavares, S. R. S. (1983). Da crise do *software* ao projeto estruturado: a submissão não real do trabalho em programação. In: FLEURY, A. C. C.; VARGAS, N. *Organização do trabalho*. São Paulo, pp. 220-232. (Atlas).
- Thiollent, M. (1985). *Crítica metodológica, investigação social e enquete operária*. 4. ed. São Paulo: Polis.