

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO
MESTRADO PROFISSIONAL EM ENSINO DE CIÊNCIA E TECNOLOGIA

SHEILA CRISTIANA DE FREITAS

**A UTILIZAÇÃO DE MAPAS CONCEITUAIS COMO APOIO AO
ENSINO DE ANÁLISE DE REQUISITOS DE *SOFTWARE***

DISSERTAÇÃO

PONTA GROSSA

2012

SHEILA CRISTIANA DE FREITAS

**A UTILIZAÇÃO DE MAPAS CONCEITUAIS COMO APOIO AO
ENSINO DE ANÁLISE DE REQUISITOS DE *SOFTWARE***

Dissertação apresentada como requisito parcial à obtenção do título de Mestre em Ensino de Ciência e Tecnologia, do Programa de Pós-Graduação da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Antonio Carlos de Francisco

PONTA GROSSA

2012

Ficha catalográfica elaborada pelo Departamento de Biblioteca
da Universidade Tecnológica Federal do Paraná, Campus Ponta Grossa
n.07 /13

P348 Freitas, Sheila Cristiana de

A utilização de mapas conceituais como apoio ao ensino de análise de requisitos de software. / Sheila Cristiana de Freitas. -- Ponta Grossa, 2013.
105 f : il. ; 30 cm.

Orientador: Prof. Dr. Antonio Carlos de Francisco.

Dissertação (Mestrado em Ensino de Ciência e Tecnologia) - Universidade Tecnológica Federal do Paraná, Campus Ponta Grossa. Curso de Pós-Graduação em Ensino de Ciência e Tecnologia. Ponta Grossa, 2013.

1. Engenharia de software. 2. Software - Projeto. 3. Análise de sistemas. 4. Aprendizagem. 5. Mapas cognitivos (Psicologia). I. Stadler, Rita de Cássia da Luz. II. Carletto, Marciah Regina. III. Universidade Tecnológica Federal do Paraná, Campus Ponta Grossa. IV. Título.

CDD 507



Universidade Tecnológica Federal do Paraná
Campus de Ponta Grossa
Diretoria de Pesquisa e Pós-Graduação
PROGRAMA DE PÓS-GRADUAÇÃO EM ENSINO
DE CIÊNCIA E TECNOLOGIA



FOLHA DE APROVAÇÃO

Título de Dissertação Nº **56/2012**

A UTILIZAÇÃO DE MAPAS CONCEITUAIS COMO APOIO AO ENSINO DE ANÁLISE DE REQUISITOS DE SOFTWARE

por

Sheila Cristiana de Freitas

Esta dissertação foi apresentada às **13 horas** de **19 de dezembro de 2012** como requisito parcial para a obtenção do título de MESTRE EM ENSINO DE CIÊNCIA E TECNOLOGIA, com área de concentração em Ciência, Tecnologia e Ensino, do Programa de Pós-Graduação em Ensino de Ciência e Tecnologia. O candidato foi argüido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Prof. Dr. Wilson Massashiro Yonezawa
(UNESP-Bauru)

Prof. Dr. André Koscianski (UTFPR)

Prof. Dr. Antonio Carlos de Francisco
(UTFPR) - Orientador

Profª. Drª. Nilcéia Aparecida Maciel
Pinheiro (UTFPR)

Visto do Coordenador:

Profª Drª Sani de Carvalho Rutz da Silva
Coordenadora do PPGECT

A FOLHA DE APROVAÇÃO ASSINADA ENCONTRA-SE NO DEPARTAMENTO DE REGISTROS ACADÊMICOS DA UTFPR – CÂMPUS PONTA GROSSA

Dedico este trabalho à minha família, aos
meus professores e aos meus amigos.

AGRADECIMENTOS

À Deus e a Nossa Senhora das Graças pela vida e pela força de vontade que me acompanha todos os dias.

Ao meu orientador Prof. Antonio Carlos de Francisco, pela amizade, pelo auxílio e pela confiança.

Em especial ao meu marido Henrique Natal da Silveira, pelo apoio e companheirismo.

Aos professores do Programa de Pós-Graduação em Ensino de Ciência e Tecnologia pela ajuda e ensinamentos durante o curso.

Aos alunos do Curso Técnico em Informática do IFPR – Campus Irati.

Por fim, agradeço a todas as pessoas que de uma forma ou de outra fazem parte da minha história.

“Você não consegue ligar os pontos olhando pra frente; você só consegue ligá-los olhando pra trás. Então você tem que confiar que os pontos se ligarão algum dia no futuro. Você tem que confiar em algo – seu instinto, destino, vida, carma, o que for. Esta abordagem nunca me desapontou, e fez toda diferença na minha vida”. (JOBS, Steve)

RESUMO

FREITAS, Sheila Cristiana. **A Utilização de Mapas Conceituais como Apoio ao ensino de Análise de Requisitos de Software**. 2012. 105 f. Dissertação (Mestrado em Ensino de Ciência e Tecnologia) - Programa de Pós-Graduação em ensino de Ciência e Tecnologia, Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2012

O objetivo deste trabalho foi avaliar a contribuição da aprendizagem significativa e mapas conceituais no ensino de análise de requisitos de *software*. As temáticas perquiridas abordam a teoria de David Ausubel sobre aprendizagem significativa e a construção de mapas conceituais como ferramentas de apoio ao ensino de análise de requisitos de software nos cursos técnicos em informática. A pesquisa classificada como exploratória de natureza qualitativa, possui origem interpretativa e foi realizada em sala de aula, em uma turma do curso técnico em informática na modalidade subsequente contendo quatorze alunos. As aulas contemplam a utilização de organizadores prévios na apresentação da teoria de análise de requisitos, exercícios que sugerem a busca por conhecimentos já existentes na estrutura cognitiva e a construção de mapas conceituais realizados pelo professor como apoio didático, e pelos alunos, na relação dos conceitos dos requisitos de software relativos ao trabalho de conclusão de curso (TCC). Os resultados foram analisados por aulas do ponto de vista do professor onde se percebeu que a estratégia utilizada neste trabalho, mas precisamente na aula 2, contribuiu para que o aluno faça a ancoragem dos subsunçores e assim a assimilação dos novos conceitos e na aula 7 são capazes de construir mapas conceituais para estabelecer a ligação entre os conceitos. Por fim, na visão dos alunos através de entrevista coletiva quanto aos benefícios de utilização de mapas conceituais nas aulas de análise de requisitos de software e os quais foram classificados como satisfatórios. Foi produzido um manual didático contendo os passos utilizados neste trabalho o qual poderá ser utilizado por outros professores.

Palavras-chave: Engenharia de *Software*, Análise de Requisitos, subsunçores, Projeto de *Software*, Aprendizagem Significativa, Mapas Conceituais.

ABSTRACT

FREITAS, Sheila Cristiana. **The Use of Concept Maps to Support Teaching Software Requirements Analysis**. 2012. 105 f. Dissertation (Master of Teaching Science and Technology) - Graduate Program in Teaching Science and Technology, Federal Technology University - Paraná. Ponta Grossa, 2012.

The aim of this study was to evaluate the contribution of meaningful learning and concept maps in teaching analysis of software requirements. The thematic approach researched David Ausubel's theory of meaningful learning and the construction of concept maps as tools to support teaching analysis of software requirements in technical courses in computer science. The survey classified as exploratory qualitative, interpretive origin and has been held in the classroom, in a class of computer technician course in subsequent modality containing fourteen students. divided into stages. The classes include the use of previous organizers in presenting the theory of requirements analysis, exercises suggest that the search for knowledge existing in the cognitive structure and the construction of concept maps made by the teacher to support teaching and by students, about the concepts of software requirements for the completion of course work (TCC). The results were analyzed by the class teacher's point of view where it was realized that the strategy used in this work, but precisely in class 2, helped the student make the anchoring of subsumers and thus the assimilation of new concepts in class and 7 are able to construct conceptual maps to establish the connection between the concepts. Finally, in view of the students through a press conference about the benefits of using concept maps in class requirements analysis and software which were rated as satisfactory. It produced a teaching manual containing the steps used in this work which can be used by other teachers.

Keywords: Software Engineering, Requirements Analysis, Software Design, Meaningful Learning, Concept Maps.

LISTA DE ILUSTRAÇÕES

Figura 1 - Engenharia de <i>Software</i> em Camadas	23
Figura 2 - Fases da Engenharia de Software – Representação gráfica adaptada	24
Figura 3 - Tipos de requisitos não funcionais.....	32
Figura 4 - Representação esquemática do modelo ausubeliano.....	41
Figura 5 - Requisitos Funcionais	56
Figura 6 - Requisitos Não Funcionais	56
Figura 7 - Requisitos	57
Figura 8 - Cenário	57
Figura 9 - <i>Stakeholders</i>	57
Figura 10 - Cliente.....	57
Figura 11 - Escopo	58
Figura 12 - Ambiguidade	58
Figura 13- Questionário.....	58
Figura 14 - Entrevista	58
Figura 15 - Documento de requisitos	59
Figura 16 - Administrador	59
Figura 17- Problema.....	59
Figura 18 - Análise	59
Figura 19 - Projeto.....	60
Figura 20 - Mapa Conceitual - Análise de Requisitos.....	70
Figura 21 - Mapa conceitual construído por aluno	77
Figura 22 - Mapa conceitual construído por aluno	78
Figura 23 - Árvore de Associações das respostas da entrevista coletiva.....	80
Figura 24 - Mapa conceitual - Trabalho 1.....	93
Figura 25 - Mapa conceitual - Trabalho 2.....	94
Figura 26 - Mapa conceitual - Trabalho 3.....	96

Figura 27 - Mapa conceitual - Trabalho 4.....	97
Figura 28 - Mapa conceitual - Trabalho 5.....	99
Figura 29 - Mapa conceitual - Trabalho 6.....	101
Figura 30 - Mapa conceitual - Trabalho 7.....	102
Figura 31 - Mapa conceitual - Trabalho 8.....	104
Figura 32 - Entrevista coletiva com os alunos	105
Quadro 1 - Unidade curricular do curso técnico em informática	47
Quadro 2 - Resultado de atividade realizada por aluno.....	75
Quadro 3 - Resultado de atividade realizada por aluno.....	76

LISTA DE SIGLAS

TCC Trabalho de Conclusão de Curso

SUMÁRIO

1 INTRODUÇÃO	13
1.1 DEFINIÇÃO DE PROBLEMA	15
1.2 OBJETIVOS	15
1.2.1 Geral.....	15
1.2.2 Específicos	16
1.3 JUSTIFICATIVA	16
1.4 ESTRUTURA DO TRABALHO	19
2 REFERENCIAL TEÓRICO	21
2.1 ENGENHARIA DE SOFTWARE	22
2.2 ANÁLISE DE REQUISITOS	27
2.2.1 Requisitos Funcionais.....	29
2.2.2 Requisitos não Funcionais.....	30
2.2.3 Técnicas de Elicitação de Requisitos	33
2.3 APRENDIZAGEM SIGNIFICATIVA – TEORIA DE DAVID AUSUBEL.....	35
2.3.1 Subsunçores.....	37
2.4 MAPAS CONCEITUAIS	39
2.4.1 O que são mapas conceituais?.....	42
2.4.2 Construção de Mapas Conceituais como técnica de aprendizagem significativa que são mapas conceituais.....	43
3 DESCRIÇÃO DA METODOLOGIA.....	46
3.1 CARACTERIZAÇÃO DA PESQUISA	46
3.1.1 Abordagem da pesquisa.....	46
3.1.2 Local e Universo da Pesquisa	47
3.1.3 Instrumento de coleta de dados	48
3.1.4 Métodos de análise dos dados	49
4 DESENVOLVIMENTO DA METODOLOGIA E ANÁLISE DOS RESULTADOS... 51	
4.1 AULA 1 – AULA EXPOSITIVA E ORGANIZADORES PRÉVIOS	52
4.1.1 Desenvolvimento das atividades da aula 1	52
4.1.2 Análise dos resultados da aula 1	60
4.2 AULA 2 – CRIAÇÃO DE SUBSUNÇORES	61
4.2.1 Desenvolvimento das atividades da aula 2.....	61
4.2.2 Análise dos resultados da aula 2	65
4.3 AULA 3 – ANÁLISE DE REQUISITOS COM ENFOQUE AOS CONCEITOS TRABALHOS	66
4.3.1 Desenvolvimento das atividades da aula 3.....	66

4.3.2 Análise dos resultados da aula 3	68
4.4 AULA 4 – APRESENTANDO MAPAS CONCEITUAIS	68
4.4.1 Desenvolvimento das atividades da aula 4.....	68
4.4.2 Análise dos resultados da aula 4	71
4.5 AULA 5 – ANÁLISE DE REQUISITOS NA PRÁTICA	71
4.5.1 Desenvolvimento das atividades da aula 5.....	71
4.5.2 Análise dos resultados da aula 5	74
4.6 AULA 6 – BUSCANDO PRINCIPAIS CONCEITOS NA ANÁLISE DE REQUISITOS	74
4.6.1 Desenvolvimento das atividades da aula 6.....	74
4.6.2 Análise dos resultados da aula 6	76
4.7 AULA 7 – ALUNOS CONSTROEM SEUS MAPAS CONCEITUAIS	77
4.7.1 Desenvolvimento das atividades da aula 7.....	77
4.7.2 Análise dos resultados da aula 7	78
4.8 ANÁLISE DOS RESULTADOS DO PONTO DE VISTA DOS ALUNOS	79
4.8.1 Questão da entrevista e análise de resultados	79
5 CONCLUSÃO	83
5.1 DIFICULDADES	85
5.2 SUGESTÕES PARA TRABALHOS FUTUROS	85
REFERÊNCIAS.....	87
APÊNDICE A – Respostas da Aula	89
APÊNDICE B – Trabalhos realizados pelos alunos nas aulas 5, 6 e 7	92
APÊNDICE C – Respostas completas da entrevista coletiva	105

1 INTRODUÇÃO

A busca por mecanismos e ferramentas de apoio ao ensino são necessárias em todas as áreas, principalmente quando se trata de assunto totalmente novo, o que é muito comum nas disciplinas técnicas as quais irão refletir diretamente no desenvolvimento profissional do aluno.

Muitos autores conhecedores do ensino como Piaget, Vygotsky e outros contribuíram com suas teorias que auxiliam professores a refletir e sair em busca de criar mecanismos, ferramentas ou fórmulas que, de algum modo, possam contribuir para um único fim, que é facilitar e promover a aprendizagem.

No momento em que professores entram em sala de aula com o objetivo de ensinar e ir além de aulas expositivas e sugestões de bibliografia inicia-se a busca por ferramentas que lhes deem segurança de que a aula foi compreendida, não apenas para a prova, mas para a vida profissional do aluno. Muito empenho e dedicação é preciso para alcançar o objetivo esperado.

Vivemos em um mundo cada vez mais tecnológico e dependente de mão de obra especializada para atender a grande demanda na área de informática. Com isto, cresce também a responsabilidade dos cursos técnicos, tecnológicos e superiores desta área em estabelecer diretrizes e formas de facilitar o aprendizado para que os alunos consigam desenvolver suas habilidades mesclando técnicas de projetos da engenharia de *software* e processos de aprendizagem.

Os referidos cursos são divididos em disciplinas que atendem a todas as etapas da Engenharia de *Software* que vão desde a obtenção de requisitos do *software* até a sua implantação e manutenção, ou seja, todo o projeto de *software*.

A criação de um *software* inicia a partir de um problema, uma necessidade ou um desejo de alguém que esteja disposto a investir nisto, e assim, o profissional de informática dará início ao projeto do qual originará o produto de *software*.

O projeto de *software* é dividido em fases, sendo que a primeira delas é a Análise de Requisitos de *software*. É nesta fase que o profissional responsável pelo projeto deverá extrair as informações dos interessados, a fim de entender as necessidades e compreender o problema. Os interessados pelo produto *software* são de clientes, os quais irão investir no projeto para que o produto *software* seja construído. O processo de extração de requisitos implica obter as informações

necessárias do cliente para que o *software* idealizado pelo mesmo seja projetado. Porém, na maioria das vezes, o diálogo é a parte mais difícil, pois nem sempre é claro o suficiente para que os requisitos do *software* sejam estabelecidos de forma correta.

De acordo com com (SWEBOOK, 2004) a análise de requisitos relaciona os processos:

- Detectar e resolver os conflitos entre os requisitos.
- Descobrir os limites do software e como ele deve interagir com seu meio ambiente.
- Requisitos de sistema elaborado para derivar requisitos de software.

Uma vez que a análise de requisitos não seja feita de maneira correta poderá comprometer todas as outras fases do desenvolvimento de software.

O aluno está formatado em um padrão tecnicista que recebe aulas teóricas com atividades que irão repetir modelos e decorar conceitos a serem respondidos em avaliações escritas, que darão uma nota no final do período, na tentativa de treiná-lo para o mercado de trabalho. Este modelo de aprendizagem é chamado por Ausubel de *aprendizagem mecânica* a qual a nova informação é armazenada de maneira arbitrária. Não há interação entre a nova informação e aquela armazenada (MOREIRA E MASINI, 2011, p. 18).

A teoria da aprendizagem significativa de Ausubel estabelece aprendizagem por conceitos de maneira organizada e bem definida. Para Moreira (1999, p. 113),

É importante não sobrecarregar o aluno de informações desnecessárias, dificultando a organização cognitiva. É preciso buscar a melhor maneira de relacionar, explicitamente, os aspectos mais importantes do conteúdo da matéria de ensino aos aspectos especificamente relevantes na estrutura cognitiva do aprendiz.

Os procedimentos propostos neste trabalho foram aplicados em sala de aula, com uma sequência de aulas que mostram o ensino de análise de requisitos de forma significativa, o qual inclui organizadores prévios, busca por subsunçores e relação dos conceitos através da criação de mapas conceituais.

David Ausubel na sua teoria da Aprendizagem Significativa, parte do princípio que a aprendizagem só é significativa quando o aluno consegue buscar conhecimentos existentes em sua estrutura cognitiva para ancorar com o novo

conhecimento. O conhecimento já existente é chamado por Ausubel de subsunçor e este trabalho procura instigar os alunos a buscarem seus subsunçores e criarem seus conceitos para que possam, posteriormente, relacioná-los utilizando mapas conceituais como ferramenta no ensino de análise de requisitos de *software*.

Desta forma este trabalho irá investigar se a utilização de mapas conceituais poderá contribuir como ferramenta de apoio no ensino de análise de requisitos.

1.1 DEFINIÇÃO DE PROBLEMA

Quando se formula um problema de pesquisa é necessário defini-lo apoiado em regras, e de acordo com Marconi & Lakatos (2010, p. 111):

O problema, assim, consiste em um enunciado explicitado de forma clara, compreensível e operacional, cujo melhor modo de solução ou é uma pesquisa ou pode ser resolvido por meio de processo científico.

Outros autores esclarecem regras para definição do problema. Nas palavras de Gil (2002, p.26) são:

(a) o problema deve ser formulado como pergunta; (b) o problema deve ser claro e preciso; (c) o problema deve ser empírico; (d) o problema deve ser suscetível de solução; (e) o problema deve ser delimitado a uma dimensão viável.

Em conformidade com as regras estabelecidas o problema deste trabalho é: Como a utilização de aprendizagem significativa e mapas conceituais podem contribuir para o ensino de análise de requisitos de *software*?

1.2 OBJETIVOS

1.2.1 Geral

Avaliar a contribuição da aprendizagem significativa e mapas conceituais no ensino de análise de requisitos de *software*.

1.2.2 Específicos

- Instigar a criação de subsunçores na análise de requisitos de *software*.
- Utilizar como apoio no ensino de análise de requisitos a aprendizagem significativa.
- Inserir a criação de mapas conceituais como ferramenta a ser utilizada nas aulas de análise de requisitos.
- Possibilitar que os alunos construam seus mapas conceituais com base nos conceitos da análise de requisitos da disciplina de TCC.
- Investigar, através de entrevista, a contribuição da estratégia proposta.
- Elaborar um material didático com etapas a serem seguidas utilizando a estratégia proposta neste trabalho.

1.3 JUSTIFICATIVA

Os cursos técnicos na área de informática exigem do aluno dedicação e disciplina para desenvolver suas atividades. Muitas vezes, os professores se deparam com alunos dedicados, porém com muita dificuldade de compreender o conteúdo e de desenvolver as atividades práticas.

Este trabalho tomou por base um curso técnico em informática na modalidade subsequente, com duração de dois anos divididos em quatro períodos (semestres de dois bimestres). Nesta modalidade os alunos devem ter o ensino médio completo. As turmas são bastante heterogêneas no que diz respeito à idade e formação, porém todos chegam à sala de aula, motivados por algum interesse pela informática e pela tecnologia.

Já no primeiro dia de aula percebe-se que alguns alunos não sabem ao certo o que irão ver no curso, outros chegam a pensar que irão trabalhar na atividade de manutenção de computadores. Isto preocupa os professores, pois será desafiador trabalhar assuntos totalmente novos, para muitos daqueles alunos.

No primeiro semestre, estes alunos começam a entender o que irão aprender e, assim, iniciam a construção dos seus primeiros algoritmos básicos, fazendo com que seu raciocínio se concentre em entender a lógica das condições, dos laços de repetições, das variáveis que irão armazenar informações e outros dados. Neste momento sentem-se empolgados pelo desafio de resolver pequenos problemas que lhe são propostos. Na sequência surge a preocupação em entender a sintaxe das linguagens de programação e conhecer algumas de suas bibliotecas. Após entender um pouco da linguagem que lhe é apresentada, vem mais sintaxe de comandos com a disciplina de banco de dados e a preocupação aumenta ainda mais.

A partir deste momento os alunos compreendem que as disciplinas não são deixadas de lado a cada fim de semestre, pelo contrário, uma depende da outra para poder atender a um único objetivo, que é ser capaz de desenvolver um projeto de *software* do início ao fim.

Já no segundo semestre, o importante papel das disciplinas de análise de sistemas se apresenta, ou seja, para que se possa desenvolver um sistema do mundo real se faz necessário entender o quê irá desenvolver, para quem irá desenvolver e para quem irá desenvolver.

Na disciplina de Análise de Sistemas o primeiro conteúdo a ser trabalhado é Análise de Requisitos de *software*, e é neste momento, que se encontram os grandes desafios, pois irão se deparar com situações distintas quando iniciam a elicitação de requisitos na prática, não possuem respostas exatas e prontas.

Defrontar-se-ão com situações e cenários que podem até possuir alguma semelhança com algo já visto, mas sempre terão seu percentual inédito. Este percentual trará o desafio do novo.

Quando se fala em criação de *software* logo se pensa em computadores, na linguagem de programação, no ambiente integrado de desenvolvimento, no banco de dados, nas ferramentas que irão auxiliar, e outras ferramentas, porém de nada irá servir todo esse aparato sem “o quê irá desenvolver” e “para quem irá desenvolver” o *software*. Surgem aqui duas perguntas básicas: *O que desenvolver? Para quem desenvolver?*

A responsabilidade do ensino de Análise de Requisitos é fazer com que o aluno compreenda que, primeiramente, ele precisa de informações e dados para depois, utilizando o computador e iniciar a o uso de linguagem de programação.

Pode-se considerar que o ensino de análise de requisitos é dividido em duas partes: a primeira é descobrir o que é análise de requisitos, onde se inicia o desafio de entender o que são requisitos funcionais e não funcionais, *stakeholders*, projeto, cliente, cenário, escopo, ambiguidade, questionário, entrevista, e outros. A segunda parte é entender e ligar os novos conceitos que surgem a cada novo projeto.

O professor tenta encontrar mecanismos para ensinar conceitos ao aluno e levá-lo a identificar as necessidades e desejos dos interessados pelo projeto e transformar isto em requisitos que serão analisados e, depois, implementados em linguagem de programação que dará origem ao produto propriamente dito.

O aluno terá que criar os seus próprios conceitos a partir de um novo cenário. Para definir conceito, serão utilizadas as palavras de Santos (2009, p.20).

Um conceito é qualquer ideia que inclua a descrição das propriedades essenciais de uma categoria. Alguém que formou um conceito pode facilmente incluir novas ideias na categoria a que esse conceito se refere, adquirindo assim, um instrumento efetivo para o seu pensamento e, conseqüentemente, para o seu arsenal de aprendizagem.

No ensino de análise de requisitos talvez o mais importante seja fazer com que o aluno, além de entender, seja capaz de criar seus próprios conceitos e relacioná-los.

Na grade curricular do curso técnico em informática que serviu de base para o desenvolvimento deste trabalho, os alunos possuem uma disciplina de TCC, e nela desenvolverão um *software* do início ao fim, ou seja, passando por todas as etapas de projeto de engenharia de *software*. Neste trabalho de TCC é que os alunos irão utilizar a ferramenta sugerida nesta pesquisa, com objetivo de auxiliar a fase de análise de requisitos.

A estratégia proposta neste trabalho foi realizada com a turma que iniciou em 2011 com conclusão em dezembro de 2012. Acredita-se que com o apoio da teoria da aprendizagem significativa de David Ausubel aliada à ferramenta de Mapas Conceituais, o ensino de análise de requisitos possa ter maior resultado e que a fase

de análise de requisitos não levará um tempo maior do que o estimado a ponto de comprometer todo o projeto.

No campus em que este trabalho foi aplicado, as turmas do curso técnico em informática na modalidade subsequente possuem vagas para quarenta alunos, os quais passam por um processo seletivo através de prova que contempla questões de redação, língua portuguesa, língua inglesa, língua espanhola, matemática, ciências naturais, história, geografia, física, química e biologia.

Este trabalho propõe uma estratégia apoiada na aprendizagem significativa para análise de requisitos de *software*, pois acredita que se o aluno conseguir obter uma aprendizagem significativa e utilizar os recursos de mapas conceituais nesta primeira etapa do trabalho, a mesma será realizada com maior precisão e menos tempo e, conseqüentemente, deixando o maior tempo para a realização das demais etapas e assim, concluir o seu trabalho dentro dos dois semestres como propõe o curso.

1.4 ESTRUTURA DO TRABALHO

Este trabalho encontra-se estruturado num total de seis capítulos divididos da seguinte forma:

- **Introdução:** Contém a definição do problema, os objetivos gerais e específicos, justificativa e a estrutura do trabalho.

- **Referencial teórico:** Trabalha a parte técnica com conceitos relacionados à Engenharia de software, Análise de requisitos de software (requisitos funcionais e não funcionais, técnicas de elicitação de requisitos) e a área ligada à aprendizagem significativa juntamente com os subsunçores e a definição e construção de mapas conceituais.

- **Descrição da metodologia:** Descreve a caracterização da pesquisa quanto à abordagem, local e universo, métodos de análise de dados e o produto.

- **Desenvolvimento da metodologia e análise dos resultados:** Apresenta o desenvolvimento das aulas e as análises dos resultados de cada aula. Também traz uma análise da opinião dos alunos através de árvore de associações e, em

seguida, faz uma pequena observação quanto às desistências ocorridas nas turmas de 2010 e 2011.

- **Conclusão:** Descreva as conclusões obtidas no trabalho.

- **Sugestões de trabalhos Futuros:** Sugere a adaptação da estratégia para outras áreas.

2 REFERENCIAL TEÓRICO

Inicialmente apresenta uma descrição da parte técnica no que diz respeito ao conteúdo do curso de informática de nível técnico, irá descrever a grande área trabalhada que é Engenharia de *software*.

Pressman (2010, p. 17) diz que a Engenharia de software é a criação e a utilização de sólidos princípios de engenharia a fim de obter softwares econômicos que sejam confiáveis e que trabalhem eficientemente em máquinas reais.

A engenharia de software abrange todas as etapas do projeto de software e é a partir destas etapas que os conhecimentos técnicos são transmitidos aos alunos. Assim poderão aprender e se apropriar destes conhecimentos que serão utilizados em sua vida profissional. Como será descrito no próximo item o projeto de software é composto por fases, e em cada fase encontra-se um novo desafio.

O professor necessita de ferramentas para auxiliar no ensino de disciplinas técnicas e ao aluno, que precisa conhecer e construir o seu conhecimento em um novo mundo de conteúdos.

Nesta fase o aluno deve aprender a extrair e analisar os requisitos de software. Nas palavras de Sommerville (2011, p. 20) “análise e definição de requisitos são os serviços, restrições e metas do sistema que são estabelecidas por meio de consulta aos usuários. Em seguida, são definidos em detalhes e funcionam como uma especificação do sistema”.

Trata-se de uma etapa de grandes dificuldades, pois o professor deverá levar o aluno a entender o que é requisito de *software* e desenvolver técnicas para analisar os requisitos e transformar em projeto de software.

Este trabalho se apoiará na teoria de David Ausubel para auxiliar o ensino de análise de requisitos de *software* com a busca de subsunçores e criação de mapas conceituais no curso técnico em informática, na área de formação tecnológica com eixo em Engenharia de *Software*. Aprender de forma significativa os conceitos da análise de requisitos é o que se espera a partir da estratégia proposta. Sobre aprendizagem significativa, citamos as palavras de Ausubel, Novak e Hanesian (1980, p. 34).

A aprendizagem significativa envolve a aquisição de novos significados e os novos significados, por sua vez, são produtos da aprendizagem significativa. Ou seja, a emergência de novos significados no aluno reflete o complemento de um processo de aprendizagem significativa.

Com a aprendizagem significativa, os alunos são capazes de relacionar conceitos e, assim, criar mecanismos de autoaprendizagem em qualquer área de conhecimento sobre qualquer conteúdo.

Como ferramenta para relacionar os conceitos será utilizada a criação de mapas conceituais, que são diagramas utilizados como um instrumento capaz de evidenciar significados atribuídos aos conceitos e às relações entre conceitos no contexto de um corpo de conhecimentos, de uma disciplina, de uma matéria de ensino (MOREIRA, 2010, p. 15).

A estratégia proposta terá como finalidade analisar a contribuição da aprendizagem significativa e a construção de mapas conceituais no ensino de análise de requisitos de software.

2.1 ENGENHARIA DE SOFTWARE

Com a constante evolução da tecnologia dos *hardwares*, os profissionais de informática responsáveis pela criação de *softwares*, recebem desafios cada vez mais importantes para a utilização dos recursos disponíveis, e assim, suprir as necessidades de diversas áreas. Estes desafios dizem respeito à qualidade da mão de obra e preparação dos profissionais responsáveis para atender a crescente demanda de *softwares*. Os cursos da área de informática destinados a educar e formar profissionais que serão responsáveis por esta área, organizam seus planos curriculares de modo que possam dar subsídios em todas as etapas que envolvem a Engenharia de *Software*. Mas o que é Engenharia de *Software*? Segundo Engholm (2010, p. 34) “a engenharia de *software* surge utilizando princípios da engenharia no desenvolvimento de *software* para aumentar a qualidade”, visto que o autor foca sua definição visando à qualidade.

Outra definição de engenharia de software nas palavras de Sommerville (2011, p. 5): “é uma disciplina de engenharia cujo foco está em todos os aspectos da

produção de *software*, desde os estágios da especificação até a sua manutenção, quando o sistema já está sendo usado”. A Engenharia de *software* parte dos princípios da engenharia, com objetivo de construir o *software* de forma ordenada, organizada com o objetivo de trazer qualidade ao produto a ser desenvolvido. Observa-se que Pressman (2010, p.17) também traz a sua definição com foco na qualidade e diz que “a Engenharia de *Software* é uma tecnologia em camadas, onde as camadas se integram com objetivo de organização e qualidade”, como representado na figura 2.



Figura 1 - Engenharia de *Software* em Camadas

Fonte: Adaptado Pressman (2010, p. 17).

A Engenharia de *Software* inclui as rotinas e etapas do projeto de desenvolvimento de *software* capaz de projetá-lo de forma que resolva os problemas do cliente e atenda suas necessidades. Em face do crescente aumento da demanda de *software* surgiu a necessidade de produzir, reduzindo custos e sem correr riscos. A tarefa da engenharia de *software* é projetar, administrar e produzir *software* com qualidade e confiabilidade através de profissionais que são denominados Engenheiros de *Software*, Engenheiros de Sistemas, Analistas de Sistemas ou Analistas de TI, dentre outras nomenclaturas utilizadas no mercado de trabalho.

Nos cursos de informática as etapas da engenharia de *software* estão divididas em disciplinas que interagem formando um só processo de produção de *software*. Do ponto de vista técnico a Engenharia de *Software* segue um roteiro formado por fases do desenvolvimento e cada uma delas possui sua importância e dependência das demais, além dos inúmeros desafios, dificuldades e complexidades. Segundo Engholm (2010, p. 66) “tradicionalmente o desenvolvimento de *software* inclui os seguintes processos: Elicitação de requisitos,

Análise de requisitos, Arquitetura, Design, Implementação, Teste e Implantação” e, apoiando-se neste modelo, segue uma breve explanação destes processos considerando como as principais fases de um projeto de *software*, e pode ser visualizado na figura 3 que trata-se de um modelo de ciclo de vida. Vale salientar que existem outros modelos que não serão citados neste trabalho.

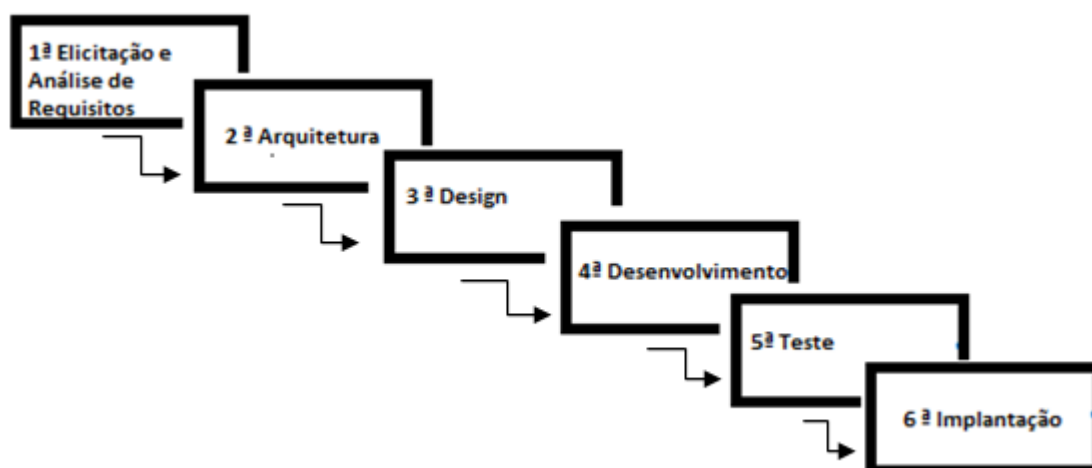


Figura 2 - Fases da Engenharia de Software – Representação gráfica adaptada
Fonte: O autor – Adaptado da teoria de Engholm (2010)

1ª – Elicitação e Análise de Requisitos: Serão unidas as atividades de elicitar e analisar em uma única fase, e é no ensino desta fase que será aplicada a estratégia sugerida neste trabalho, a qual será descrita nos próximos itens. Pode-se dizer que esta é a principal fase no desenvolvimento de um projeto de *software*, é ela que irá definir os objetivos do *software* e suas funcionalidades e, conseqüentemente, as demais fases são dependentes da primeira. Se esta não for definida de forma correta as conseqüências negativas refletirão até a última fase. O primeiro trabalho dos profissionais de informática, também conhecidos como engenheiros de *software*, consiste em definir junto aos interessados pelo projeto quais serão os objetivos e as restrições do *software*, e estes serão chamados de requisitos do *software*. Para definir “requisitos de *software*” as palavras de Machado (2011, p. 24):

Os requisitos expressam as características e restrições do produto de *software* do ponto de vista de satisfação das necessidades do usuário e, em geral, independem da tecnologia empregada na construção da solução,

sendo a parte mais crítica e propensa a erros no desenvolvimento de *software*.

Após elicitar os requisitos, eles serão analisados e, por fim, modelados, de acordo com Pressman (2010, p. 116), “entender os requisitos do problema está entre as tarefas mais difíceis enfrentadas por um engenheiro de *software*”, pois se construir um *software* que não resolva o problema do cliente, não atendendo suas reais necessidades, ele não servirá para nada.

Para Sommerville (2007, p.77), “talvez o maior problema que enfrentamos no desenvolvimento de sistemas de *software* grandes e complexos, seja o da Engenharia que Requisitos”. Os diálogos entre os *stakeholders* são passíveis de sofrer ambiguidades e conflitos, o que torna esta fase complexa, pois todos os demais processos dependerão dos requisitos.

2ª – Arquitetura: Neste momento haverá a preocupação com os requisitos não funcionais do sistema, que são aqueles que não fazem parte direta do sistema, porém são relevantes para o funcionamento do mesmo. De acordo com Engholm (2010, p. 94), as decisões neste momento tomarão por base:

- Onde a empresa pretende hospedar o sistema.
- Quais Plataformas serão utilizadas pela empresa e quais os tipos de plataforma.
- Quais serão os sistemas legados onde o novo sistema deverá ser integrado.
- Quais os tipos de interface que se deseja utilizar.
- Caso necessário, tipo de distribuição do sistema.

Nesta fase é que serão calculados os riscos e analisadas as formas de minimizar o custo e aproveitar o tempo de execução do projeto. Para definir o termo arquitetura, Pressman (2010, p. 208) refere-se comparativamente à arquitetura de um edifício, dizendo que “é a maneira pela qual os vários componentes de um edifício são integrados para formar um todo coeso”. É a arquitetura que irá analisar, estabelecer a comunicação entre os diversos componentes que fazem parte do sistema.

3ª Design: É nesta fase que os detalhes para atender os objetivos do projeto devem ser modelados. Os engenheiros de *software*, que trabalham no *design* são muitas vezes conhecidos como analistas de sistemas e sua

responsabilidade é trabalhar com a modelagem de análise. Neste momento, uma ferramenta de modelar irá auxiliar e será bastante utilizada. Cita-se como exemplo a linguagem de modelagem UML (*Unified Modeling Language*). Pressman (2010, p. 144) diz que “a modelagem de análise usa uma combinação de formas textuais e diagramáticas para mostrar os requisitos de dados, função e comportamento, de modo que seja relativamente fácil de entender”. As representações textuais e diagramáticas do sistema permitem que pessoas envolvidas no projeto, e que não sejam a área de engenharia de *software*, possam visualizar as funcionalidades, os relacionamentos entre os processos e assim, prevenir eventuais falhas no momento da implementação.

Para Engholm (2010, p. 98) a fase de design deve atender aos objetivos: “Resolver o problema de negócio, identificar componentes de suporte que farão o sistema funcionar, produzir especificações detalhadas do sistema, simular casos de testes, estabelecer cenário para implantação.” Esta fase deve retratar o sistema e suas funcionalidades antes de iniciar a fase de implementação, sendo possível prever situações, alterar funções, se for o caso.

4ª Desenvolvimento: A fase de desenvolvimento é onde nascerá o produto propriamente dito, nesta fase os engenheiros de *software* fazem o papel de programadores que são profissionais que irão transportar tudo que foi idealizado na fase anterior para as linguagens de programação.

Nesta fase do projeto os objetivos elencados por Engholm (2010, p. 108), devem ser atingidos, são eles:

- Desenvolver código com qualidade, no menor custo e no menor prazo.
- Códigos devem ser desenvolvidos de maneira eficaz, com rapidez e menor quantidade de erros.
- As especificações contidas nos diagramas devem ser atendidas em sua totalidade.
- O desenvolvimento deve ser de modo interativo e incremental.

Na implementação do *software* vale salientar a importância da escolha da linguagem mais adequada e criação de planos de desenvolvimento.

5ª Teste: A fase de teste tem como objetivo certificar-se de que o sistema realizará as atividades que lhe foram propostas na análise de requisitos e, também,

encontrar possíveis falhas. Cabe lembrar que testes exaustivos de todos os campos e funções tornam-se uma tarefa inviável. Faz parte da garantia da qualidade do *software* e, chamados também de processo de validação do *software*. Segundo Sommerville (2011, p. 145):

A validação de *software*, mas genericamente, verificação e validação (V&V) objetivam verificar se o *software* em desenvolvimento satisfaz suas especificações e oferece a funcionalidade esperada pelas pessoas que estão pagando pelo *software*.

Primeiramente a fase de teste busca a validação dos dados, pois sem a real certeza de que o *software* está de acordo com o que foi solicitado no documento de requisitos, não há conclusão do mesmo como produto. Além da validação, os testes servem também como verificação da integridade, de acordo com Pressman (2010, p. 288), “irão descobrir erros que foram feitos inadvertidamente no momento em que foi projetado e construído”. Esta etapa tem como finalidade principal detectar eventuais falhas que serão ajustadas antes da apresentação do produto ao cliente.

6ª Implantação: Após passar pelas fases anteriores com sucesso, chega o momento de o produto *software* sair do ambiente de projeto e desenvolvimento, ser colocado no ambiente do cliente e, finalmente, utilizado. A implantação deve ser planejada, pois a instalação no ambiente do cliente é apenas uma das tarefas da implantação. De acordo com Engholm (2011, p. 299), “envolve também uma série de outras atividades, tais como treinamento, comunicação, verificação e validação do seu sistema no ambiente de produção da empresa”. Todo processo de implantação deve ser planejado e acompanhado respeitando as características de cada empresa/cliente. Após esta fase ainda teríamos que descrever as fases de ajustes e manutenções que seguem continuamente durante a vida útil do *software*.

2.2 ANÁLISE DE REQUISITOS

Como descrito no item anterior a Engenharia de *Software* diz respeito a todas as etapas de um projeto de *software* e que a primeira fase do projeto é a Elicitação e Análise de Requisitos. Este trabalho irá operar toda a estratégia metodologia proposta no ensino desta primeira fase.

Um *software* nasce a partir de uma necessidade ou desejo, o que dará origem ao problema que o produto *software* deverá resolver, e é neste momento que os profissionais de informática irão analisar o problema proposto para que possam iniciar o projeto e o planejamento para sua execução.

O início do projeto se dará a partir de um diálogo entre os interessados e os profissionais que irão desenvolver o projeto, a fim de estabelecer os requisitos do *software*. Todos os envolvidos no projeto são chamados de *stakeholders*, conforme define Machado (2011, p. 30) “é qualquer pessoa materialmente afetada pelo resultado do projeto, usuários diretos e indiretos, investidores, acionistas, fornecedores, supervisores, gerentes, compradores, pessoal de suporte e manutenção, redatores técnicos”.

O diálogo entre os *stakeholders* nem sempre é uma tarefa fácil, até mesmo para profissionais experientes, o que se torna ainda mais difícil quando estamos falando de alunos de um curso de informática. É a partir do diálogo que se inicia uma investigação das necessidades, objetivos e particularidades que farão parte do projeto de *software*. O projeto inicia com o levantamento de requisitos do sistema.

Nas palavras de Machado (2011, p. 24), “os requisitos expressam as características e restrições do produto de *software* do ponto de vista de satisfação das necessidades do usuário”. Pode-se dizer que os requisitos do sistema incluem tudo aquilo que o sistema deve fazer e, também, suas restrições. Os requisitos são determinados pelo cliente de acordo com a sua necessidade e desejos. De acordo com Sommerville (2011, p. 57) “são as descrições do que o sistema deve fazer, os serviços que oferece e as restrições ao seu funcionamento”.

Tudo que se refere a requisitos de *software* ou sistema faz parte da engenharia de requisitos. Pressman (2010, p. 116), diz que “a engenharia de requisitos ajuda os engenheiros de *software* a compreender melhor o problema que eles vão trabalhar para resolver”. As atividades a serem realizadas buscam entender as necessidades e desejos trazidos pelos clientes, as restrições e definir como o sistema irá operar.

A engenharia de requisitos não é somente um processo técnico, pois nasce de uma relação pessoal em que conversas darão origem a negociações que são influenciadas por vários aspectos como organizacionais, legais além de preferenciais.

Este momento de diálogo e conversas com o usuário passa ser um grande desafio, pois se ouve de alguns alunos em sala de aula que a escolha pela área de informática tem a ver com um perfil profissional mais isolado. O profissional de informática possui várias funções no mercado de trabalho, porém as relações pessoais e compreensão das necessidades de outras áreas são indispensáveis para o desenvolvimento das tarefas.

Nas aulas de análise de requisitos é que o aluno descobre que necessita desenvolver diálogos com o usuário e entender o que ele quer e o que ele precisa. É necessário lembrar que muitas vezes o cliente não sabe ao certo o que quer e nem o que precisa. E caberá ao profissional de informática compreender os desejos, descobrir as necessidades e propor as soluções, mas para que possa propor soluções terá que extrair os requisitos de software.

Os requisitos são usualmente classificados como Requisitos Funcionais e requisitos Não funcionais.

2.2.1 Requisitos Funcionais

Nos primeiros contatos existe a preocupação de esclarecer as funcionalidades do sistema, os *Stakeholders* se preocupam em defini-las. Segundo Sommerville (2011, p. 59) “os requisitos funcionais de um sistema descrevem o que o sistema deve fazer”, e isto depende do tipo de *software* que será desenvolvido. Cabe salientar que para Machado (2011, p. 34), “a especificação de um requisito funcional deve determinar o que se espera que o *software* faça, sem a preocupação de como ele faz”. Um exemplo didático bastante utilizado nas aulas de análise de requisitos é considerar um sistema de biblioteca de uma universidade, e a partir deste elencar alguns requisitos funcionais essenciais, tais como:

1. O sistema deve permitir o cadastro, alteração, consulta e exclusão de usuários da biblioteca, com os seguintes atributos: tipo do usuário, matrícula ou registro (código do usuário), curso, nome, endereço, cidade, estado, telefone, email, CPF, documento de identificação, data de nascimento. Observação: Os usuários podem ser os alunos, os professores ou funcionários administrativos.

2. O sistema deve permitir a cadastro, alteração, consulta e exclusão de itens (livros, revistas, periódicos, e outros). Cada item possui os seguintes atributos: código, tipo de Item, título do item, autores, data da publicação, número de edição, editora, número de cópias existentes, código da cópia.

3. O sistema deve permitir o empréstimo de no máximo três itens por aluno ou funcionário administrativo, com tempo limite de sete dias para devolução e no máximo cinco itens para professores, com tempo limite de quinze dias para devolução. São atributos do empréstimo: identificação do usuário (código do usuário), data do empréstimo, data da devolução, nome do funcionário que efetuou o empréstimo. Observação: O usuário deve estar previamente cadastrado e o sistema deve gerar a data de entrega automaticamente de acordo com as regras de tipo de usuário.

4. As informações serão exportadas para o Sistema de Gerenciamento Acadêmico para que todos os documentos solicitados por alunos só sejam emitidos após a verificação de pendências com a biblioteca.

Os requisitos acima são apenas alguns exemplos, pois os requisitos funcionais variam de representações gerais ou específicas das funcionalidades do *software*. Os requisitos funcionais determinam o que o sistema deve fazer, ou seja, descrevem o comportamento e as ações, definem as funcionalidades (MACHADO, 2011, p. 34). São os requisitos funcionais que determinarão o que o *software* deve fazer, e em princípio, a especificação dos requisitos funcionais deve ser completa e consistente. Completude significa que todos os serviços requeridos pelo usuário devem ser definidos (SOMMERVILLE, 2011, p. 60).

Os profissionais responsáveis pelo projeto iniciam seus planejamentos após conhecer o definir os requisitos funcionais.

2.2.2 Requisitos Não Funcionais

Os requisitos não funcionais não dizem respeito às funcionalidades do sistema, porém são relevantes para o projeto e merecem atenção especial tal qual os requisitos funcionais. Segundo Machado (2011, p. 35), “os requisitos não

funcionais não estão ligados diretamente com as funções fornecidas pelo sistema. Em geral se preocupam com padrões de qualidade.” Mesmo não estando relacionados diretamente ao sistema fazem parte do processo.

Os alunos também descobrem que, além de se preocupar com as funcionalidades do sistema, terão ainda que se preocupar com tudo aquilo que envolve o ambiente onde será implantado e a delimitação das funcionalidades para que não se torne inviável financeiramente.

Alguns alunos constatam que, até aquele momento, achavam que o mais difícil era aprender lógica e linguagem de programação, porém descobrem que possuem um desafio maior ainda na primeira fase do projeto.

Os requisitos não funcionais descrevem atributos do sistema ou do ambiente do sistema, tais como: Extensibilidade, Usabilidade, Confiabilidade, Desempenho, Escalabilidade, Reusabilidade. (ENGHOLM, 2010, p.69)

Os autores descrevem suas definições deixando claro que os requisitos não funcionais não estão diretamente ligados às funcionalidades do sistema, porém estão relacionados a aspectos extremamente relevantes e que, os responsáveis pelo projeto deverão elencá-los, a partir de uma minuciosa análise de cenário de recursos disponibilizados pelos interessados pelo sistema, esta ideia reforça-se nas palavras de Sommerville (2011, p. 60):

Os Requisitos não funcionais como, desempenho, proteção ou disponibilidade, normalmente especificam ou restringem as características do sistema como um todo. Requisitos não funcionais são frequentemente mais críticos que requisitos funcionais individuais.

Para assegurar que o *software* depois de pronto irá realmente atender as necessidades e expectativas daqueles que necessitam dele, é necessário que a atenção dada aos requisitos não funcionais seja tão importante quanto aquelas dadas aos requisitos funcionais. Segundo Machado (2011, p. 35):

São muito importantes, pois definem se o sistema será eficiente para a tarefa que se propõe a fazer. Um sistema ineficiente certamente não será usado.

São exemplos de requisitos não funcionais:

– A base de dados deve ser protegida para acesso apenas de usuários autorizados.

- O tempo de resposta do sistema não deve ultrapassar 30 segundos.
- O *software* deve ser operacionalizado no sistema Linux.
- O tempo de desenvolvimento não deve ultrapassar seis meses.

Os alunos aqui também observam que não há uma fórmula pronta para todos os projetos, e que cada um possui suas particularidades e especificidades.

Cada projeto estabelece os requisitos não funcionais de acordo com a necessidade dos usuários, devido a vários aspectos como: restrições de orçamento, políticas organizacionais, regulamentos de segurança, legislações e outros (SOMMERVILLE, 2011, p. 61).

O diagrama abaixo mostra que os requisitos funcionais e não funcionais são requeridos de acordo com as características de cada sistema:

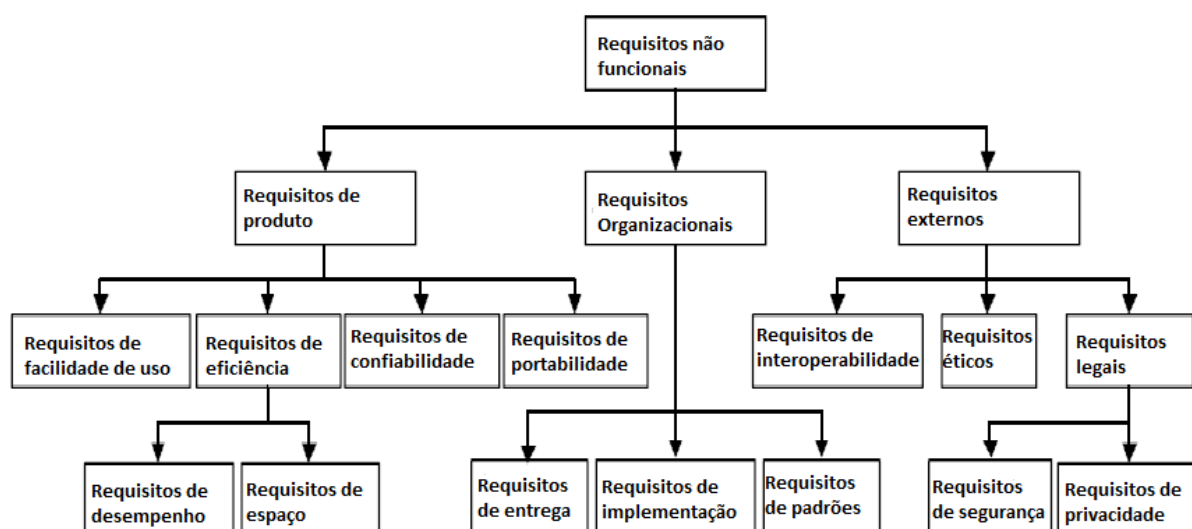


Figura 3 - Tipos de requisitos não funcionais
 Fonte: Sommerville (2011, p. 61).

Conforme a figura 3 vale esclarecer que os três tipos que surgem no primeiro nível são:

- Requisitos de produto: Dizem respeito ao comportamento do *software*, especificam ou restringem. Por exemplo: estabelecer taxas aceitáveis de falha para medir o nível de confiabilidade.

- Requisitos organizacionais: São originados das políticas e procedimentos organizacionais do cliente e do desenvolvedor. Exemplo: a escolha da linguagem de programação que será utilizada.

- Requisitos externos: São todos os requisitos que são originados de fatores externos ao sistema. Exemplificando: os regulamentos legais. (SOMMERVILLE, 2011, p. 61).

Os tipos de requisitos não funcionais serão especificados de acordo com a necessidade, vai depender da área em que o *software* irá operar e da estrutura organizacional do cliente.

2.2.3 Técnicas de Elicitação de Requisitos

Existem algumas técnicas que auxiliam o profissional de informática a extrair os requisitos de software. A palavra extrair é bem pertinente neste caso, pois muitas vezes o sentido é arrancar uma informação certa da pessoa certa. Em sala de aula todos perguntam a mesma coisa: Como fazer isto?

Após uma análise de viabilidade do projeto o primeiro passo é elicitar os requisitos e analisá-los. Segundo Machado (2011, p. 31) “elicitar significa extrair, obter, produzir os requisitos do sistema”, e é neste momento, que os engenheiros de requisitos irão trabalhar diretamente com os clientes externos e usuários finais, para obter as informações necessárias utilizando técnicas que facilitem o diálogo.

- Entrevista: é uma das técnicas mais utilizadas, onde os engenheiros de requisitos abordam todos os envolvidos no projeto e iniciam uma série de perguntas sobre o sistema que usam no momento e sobre o sistema que será desenvolvido, e é a partir dessas perguntas que surgem os requisitos (SOMMERVILLE, 2011, p. 72). Esta técnica tradicional deve ser bem conduzida dando direito aos entrevistados de expor opiniões e ideias. Afirma-se a atenção que deve ser dada para a elaboração da entrevista com as palavras de Machado (2011, p. 142): “é necessário ter um plano de entrevista, para que não haja dispersão do assunto e a entrevista fique longa, deixando o entrevistado cansado e não produzindo bons resultados”.

As entrevistas são de grande utilidade para buscar opiniões e conhecer o envolvimento dos usuários, porém algumas diretrizes são importantes e bem-vindas para que a entrevista seja bem sucedida, tais como: desenvolver um plano geral de entrevistas, verificar as autorizações para realizá-la, planejar o tempo que será utilizado, utilizar ferramentas que sejam adequadas, descobrir qual é o maior interesse do usuário, entre outras. (MACHADO, 2011, p. 142).

A entrevista em alguns casos será a única fonte de requisitos, utilizada pelos engenheiros de requisitos, mas poderá não ser suficiente para coletar todos os dados essenciais.

– Questionário: geralmente o questionário é utilizado quando a entrevista se torna inviável devido à distribuição geográfica ou quando há grande número de usuários. Da mesma forma que a entrevista, o questionário requer uma elaboração criteriosa e uma seleção dos usuários envolvidos. De acordo com Machado (2011, p. 148), “existem vários tipos de questionários que podem ser utilizados. Entre eles pode-se citar: múltipla escolha, lista de verificação e questões com espaços em branco”. Vale salientar que é necessário um conhecimento abrangente do negócio, da estrutura organizacional e das políticas da empresa, antes de formular as questões.

– *Brainstorming*: É uma técnica utilizada para trabalhar com interações bem objetivas entre os seus participantes, segundo Machado (2011, p. 149), “consiste em uma ou várias reuniões que permitem que as pessoas sugiram e explorem ideias”. Esta técnica requer planejamento e também, a inclusão de uma pessoa para conduzir esse processo, além de uma seleção criteriosa dos participantes, e uma explicação clara das regras que serão utilizadas. Durante a reunião uma pessoa é designada para registrar as ideias de forma que todos possam visualizar e, ao final, o grupo trabalha com a revisão de todas as ideias propostas (MACHADO 2011, p. 150). Esta técnica trabalha com a colocação de ideias particulares, juntamente com a colaboração nas ideias dos demais participantes.

– Prototipagem: é utilizada para mostrar, como forma de modelo, alguns subconjuntos do sistema, dando uma ideia do resultado esperado. Para Machado (2011, p. 141), “as técnicas utilizadas na elaboração do protótipo são várias, a saber: interface de usuário, relatórios textuais, relatórios gráficos, entre outras”. Esta

técnica pode ser utilizada como complemento das demais técnicas, sendo necessário ter domínio do processo de negócio.

As técnicas citadas são algumas das formas de elicitação de requisitos e podem ser utilizadas isoladas ou combinadas, de acordo com a necessidade de cada projeto. Vale lembrar que em caso de projetos mais extensos, o uso de recursos mais elaborados se faz necessário, porém este trabalho irá focar apenas nestas técnicas que basicamente fazem parte das ementas dos cursos de informática.

2.3 APRENDIZAGEM SIGNIFICATIVA – TEORIA DE DAVID AUSUBEL

Os alunos do curso de informática possuem um grande número de tarefas a serem executadas, além de muito conteúdo a ser aprendido, para que consigam passar apenas pela primeira fase do projeto de software. Não há como o professor utilizar apenas uma aula expositiva passando pelas teorias que foram citadas neste trabalho, sem procurar ferramentas de apoio que o auxiliem na transmissão desse conteúdo e, conseqüentemente, na aprendizagem desse aluno. O aluno vive em constante aprendizado, sendo necessário compreender conceitos e situações, para que possa administrar as decisões. As ideias são formadas a partir das relações entre conceitos já compreendidos e construídos.

Para que a aprendizagem seja significativa o professor deve refletir sobre a maneira pela qual seus alunos estão construindo conceitos e como os professores estão auxiliando os seus alunos nesta tarefa de criação de conceitos, assimilação e, conseqüentemente, a aprendizagem.

Segundo Moreira e Masini (2011, p. 12): “é significativa uma situação do ponto de vista fenomenológico, quando o indivíduo decide de forma ativa, por meio de uma ampliação e aprofundamento da consciência, por sua própria elaboração e compreensão”.

Este trabalho se apoia na teoria de David Ausubel sobre aprendizagem significativa, como ferramenta no ensino de análise de requisitos. Para isto, a estratégia das aulas foi organizada de modo que possibilite aos alunos criar os seus

conceitos a partir dos conceitos já existentes em sua estrutura cognitiva e fazer a ligação entre eles.

Considera-se que existe aprendizagem significativa a partir do momento que os indivíduos estabelecem significado, organizando, transformando e armazenando informações. Porém, esta situação é muito dependente do aprendiz, pois requer esforço em conectar de maneira não arbitrária e não literal, o novo conhecimento com o conhecimento já existente na estrutura cognitiva (TAVARES, 2010).

Para Ausubel a aprendizagem se concretiza a partir de conhecimentos prévios já existentes na estrutura cognitiva do indivíduo. Quando existe aprendizagem as informações serão armazenadas e organizadas, e futuramente poderão ser utilizadas.

Para dar suporte a esta ideia, vale mencionar as palavras de Moreira e Masini (2011, p. 14):

A aprendizagem significativa processa-se quando o material novo, ideias e informações que apresentam uma estrutura lógica, interagem com conceitos relevantes e inclusivos, claros e disponíveis na estrutura cognitiva, sendo por eles assimilados, contribuindo para a sua diferenciação, elaboração e estabilidade.

Os autores citados esclarecem quando a aprendizagem significativa ocorre. Dessa forma, um desafio se apresenta ao professor no sentido de que a aprendizagem significativa seja inserida em sala de aula, e neste trabalho, nas aulas de análise de requisitos de *software*.

Cada aluno possui particularidades na forma de adquirir conhecimentos e interesses próprios em tornar a sua aprendizagem significativa de modo que, ao final do conteúdo, possa expressar e representar o conhecimento adquirido.

Ausubel coloca como ideia central da sua teoria, a importância sobre o que o aluno já sabe, pois a aprendizagem significativa nasce a partir de conceitos que são formados e elencados a outros que já existem na estrutura cognitiva do indivíduo. O aluno irá associar os novos conceitos a conceitos antigos, ou seja, conceitos que já fazem parte da sua estrutura cognitiva, e estes conhecimentos são chamados por Ausubel de subsunçores.

Pode-se dizer que a aprendizagem possui características individuais, pois cada aluno possui sua base de dados particular e o desenvolvimento da aprendizagem deve ser observado e orientado de maneira distinta pelo professor.

Segundo Ausubel se a aprendizagem não é significativa então é chamada de aprendizagem mecânica, ou seja, uma aprendizagem de nova informação com pouca ou nenhuma interação com os conceitos anteriores existentes na estrutura cognitiva, logo o novo conceito é armazenado de forma arbitrária. Neste caso, podem ser citadas como exemplo, as fórmulas, as definições formatadas, ou seja, tudo aquilo que é decorado e arbitrariamente armazenado (MOREIRA E MASINI, 2011, p. 19).

A partir do momento que o aluno comece a buscar os conceitos existentes, para que possa, a partir destes, criar novos e até mesmo modificá-los e armazená-los, o mesmo aprenderá por descoberta. Entretanto, após a descoberta em si, a aprendizagem só é significativa se o conteúdo descoberto relacionar-se a conceitos subsunçores relevantes e já existentes na estrutura cognitiva (MOREIRA E MASINI, 2011, p. 19).

Ausubel propõe que a partir de conteúdos e conhecimentos que o indivíduo já possui na sua estrutura cognitiva é que a nova aprendizagem irá ocorrer, podendo até mesmo modificar os conceitos já existentes. A aprendizagem significativa é um processo em que uma informação nova é conectada a uma estrutura prévia, ou seja, busca trabalhar a partir do que o aluno já sabe e estes conhecimentos existentes chamados por Ausubel de subsunçores são base de toda sua teoria.

2.3.1 Subsunçores

Os conceitos já existentes na estrutura cognitiva do aprendiz são denominados por Ausubel de conceito subsunçor ou subsunçores. Os conceitos novos serão ancorados aos subsunçores para que haja uma ligação, modificação ou associação e que, a partir daí, exista uma aprendizagem significativa.

Para Ausubel “a aprendizagem significativa ocorre quando a nova informação se relaciona com um aspecto relevante na estrutura do conhecimento do indivíduo” (MOREIRA E MASINI, 2011, p.17). Porém, salienta-se que na formação

de conceitos existem diferenças de acordo com a idade do aluno, e este trabalho irá se referir à formação de conceitos na idade adulta. Crianças não serão levadas em consideração, pois no caso de uma criança na idade pré-escolar existe aquisição espontânea de ideias genéricas por meio de experiência empírico-concreta (MOREIRA E MASINI, 2011, p. 20).

Na infância aprende-se por descoberta, porém nas crianças mais velhas bem como nos adultos existe a assimilação de conceitos, ou seja, fazendo a relação dos novos conceitos com os já existentes na estrutura cognitiva. O processo da assimilação de conceitos envolve de acordo com Moreira e Masini (2011, p. 20), “ideias relevantes estabelecidas na estrutura cognitiva do aprendiz com o conteúdo potencialmente significativo, implícito na definição dos termos ou das pistas contextuais”. Quando se inicia o trabalho baseado na aprendizagem significativa, o primeiro passo é a busca pelos subsunçores, ou seja, é necessário que haja conhecimento prévio. Porém, existem perguntas que são inevitáveis: De onde vêm os subsunçores? E se eles não existirem?

Trabalhando com a não existência dos subsunçores, Ausubel coloca que quando um indivíduo aprender algo extremamente novo, um conceito que não consiga comparar com nada que tenha visto, isto é aprendizagem mecânica, portanto, não há aprendizagem significativa sem subsunçores. Entretanto no momento em que se está disposto a investir e instigar a aprendizagem significativa o primeiro passo será ir à busca dos subsunçores que darão a base para o novo conhecimento e validação da aprendizagem significativa. O aluno deve buscar subsunçores, podendo estes não estar tão elaborados, porém “à medida que a aprendizagem começa ser significativa, esses subsunçores vão ficando cada vez mais elaborados e mais capazes de ancorar novas informações” (MOREIRA E MASINI, 2011, p. 19).

Mas não basta tentar inserir conceitos novos baseados a partir de conceitos já existentes de forma desordenada ou sem planejamento do conteúdo que será ministrado em sala de aula. Para que isto possa ser feito Ausubel “recomenda o uso de organizadores prévios que sirvam de âncora para a nova aprendizagem e levem ao desenvolvimento de conceitos subsunçores” (MOREIRA E MASINI, 2011, p. 21).

A principal função dos organizadores prévios é estabelecer uma ligação entre os conceitos que o aluno sabe e o que precisa saber para assim ter uma aprendizagem significativa. Cabe lembrar que existem situações em que o aluno não

possui subsunçores, neste caso há a necessidade da ajuda do professor para que os mesmo sejam criados.

O professor deverá utilizar materiais que possibilitem uma introdução ao conteúdo através de exemplos, comparações e simulações. De acordo com Moreira e Masini (2011, p. 22) “no caso do material totalmente não-familiar, um organizador explicativo é usado para promover subsunçores relevantes aproximados”. Nesta estratégia não deve colocar os organizadores como sumários e forçar uma aprendizagem e sim, fazer uma conexão entre os novos conceitos e o os conhecimentos já existentes.

Os organizadores construídos de cada unidade podem ser simples comparações introdutórias entre material novo e o material conhecido, permitindo que o aprendiz perceba as características e crie o seu subsunçor, segundo Moreira e Masini (2011, p. 22) capaz de:

- a) identificar o conteúdo relevante na estrutura cognitiva e explicar a relevância desse conteúdo para a aprendizagem do novo material.
- b) dar uma visão geral do material em um nível mais alto de abstração, salientando as relações importantes.
- c) prover elementos organizacionais inclusivos, que levem em consideração mais eficientemente e, ponham em melhor destaque, o conteúdo específico do novo material.

Para que os organizadores sejam úteis devem ser formulados com termos familiares, mas para que haja aprendizagem significativa se faz necessária a relação dos novos conceitos com os conceitos prévios existentes na estrutura cognitiva e o aluno deve estar disposto a aprender e fazer a relação entre os conceitos.

2.4 MAPAS CONCEITUAIS

Ausubel acredita que os conceitos das disciplinas quando hierarquizados de forma organizada permitem que o aluno seja capaz de identificar um sistema de processamento intelectual, ou um mapa intelectual onde os conceitos são gerais e subordinados (MOREIRA E MASINI, 2011, p.55).

O professor traz o seu material organizado, porém deve levar em consideração o contexto de educação que se vive hoje, onde os alunos trazem

informações de todos os lugares, com a internet e a televisão o acesso à informação é muito rápido e com uma grande diversidade. *Como considerar a evolução do comportamento humano, onde as pessoas, tanto crianças, quanto jovens e adultos se comunicam muito, possuem em geral uma facilidade de troca de informações entre si?* Conversam e trocam experiências e tudo isto faz com que o aluno, independente da sua idade, chegue à sala aula com várias informações e conceitos que já elaborou e armazenou em sua estrutura cognitiva.

Mesmo sendo os mapas conceituais ferramentas que já ultrapassam algumas décadas, são muito úteis e compatíveis com as necessidades pedagógicas de hoje. Os alunos se apropriam de vários conceitos e necessitam de uma ferramenta que os auxilie nas ligações, a fim de transformar conceitos em aprendizagem.

Os Mapas Conceituais são uma técnica criada pelo cientista norte-americano Joseph Novak e seus colaboradores na década de setenta na Universidade de Cornell, nos Estados Unidos (MOREIRA, 2010, p. 17). Novak em sua criação baseou-se na teoria da aprendizagem significativa de David Ausubel, especificamente no que diz abaixo Ausubel, Novak e Hanesian (1980, p. 34).

A essência do processo de aprendizagem significativa é que as ideias expressas simbolicamente são relacionadas às informações previamente adquiridas pelo aluno através de uma relação não arbitrária e substantiva (não literal). Uma relação não arbitrária e substantiva significa que as ideias são relacionadas a algum aspecto relevante existente na estrutura cognitiva do aluno.

No ensino a construção de mapas conceituais torna-se uma ferramenta que permite demonstrar de que forma determinado conhecimento de um indivíduo está organizado na sua estrutura cognitiva e permite explicitar a relação entre os conceitos.

A proposta de construção de mapas conceituais de acordo com Novak e Gowin (1999) se faz de forma hierárquica dos conceitos que são apresentados tanto, em diferenciação progressiva quanto de uma reconciliação integrativa.

À medida que a aprendizagem significativa ocorre, os conceitos que são criados interagem. Quando inserir os conceitos mais gerais para os específicos trabalha-se a diferenciação progressiva. Quando se aprofunda até o conceito mais

específico e, em seguida, remete-se ao sentido contrário, ou seja, do conceito específico para o geral, então surge reconciliação integrativa, conforme mostra a figura a seguir (MOREIRA E MASINI, 2011, p. 32).

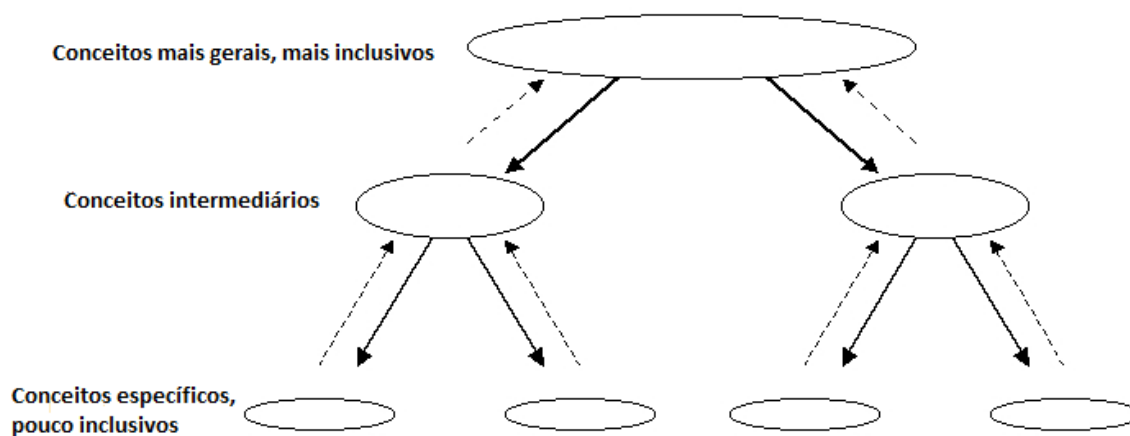


Figura 4 - Representação esquemática do modelo ausubeliano
Fonte: Moreira e Masini (2011, p. 33)

Na figura 4 as linhas contínuas sugerem a direção recomendada para a diferenciação progressiva de conceitos e as linhas traçadas a reconciliação integrativa. A figura mostra que para se atingir a reconciliação integrativa é necessário descer dos conceitos gerais para os específicos e depois, subir dos específicos para os gerais. Segundo Moreira e Masini (2011, p. 50) “do ponto de vista ausubeliano, o desenvolvimento de conceitos é facilitado quando os elementos mais gerais, mais inclusivos de um conceito são introduzidos em primeiro lugar”, posteriormente é que surgem as especificidades, ou seja, as ideias mais gerais das disciplinas devem ser apresentadas logo no início e depois, progressivamente, sejam atribuídos os conceitos diferenciados. Para atingir o que Ausubel chama de princípio de “reconciliação integrativa” é necessário explorar as relações entre proposições e conceitos, chamar a atenção para as diferenças e similaridades importantes. Mapas conceituais são sugeridos como instrumentos úteis na implementação desses princípios.

Para Novak e Gowin (1999), na diferenciação progressiva um determinado conceito é desmembrado em outros conceitos que estão contidos e ligados entre si integralmente ou em partes. Porém, na reconciliação integrativa um determinado conceito é relacionado com outro diferente. Na construção do mapa na reconciliação

integrativa, um conceito de um nó da raiz é relacionado ao outro conceito de outro nó da raiz, proporcionando uma reconciliação, ou seja, a ligação entre dois conceitos que não haviam sido percebidos.

O mapa conceitual hierárquico sendo construído pelo aluno se torna uma ferramenta adequada para estruturar o conhecimento que está sendo construído por ele mesmo. Para Novak e Gowin (1999) as ligações cruzadas podem indicar capacidade criativa proporcionando ao aluno a oportunidade de aprender a aprender.

2.4.1 O que são mapas conceituais

De acordo com Moreira (2010, p.11), “mapas conceituais, ou mapas de conceitos, são diagramas indicando relações entre conceitos, ou entre palavras que usamos para representar conceitos”. Os mapas conceituais são formados por figuras e setas, mas não devem ser confundidos com organogramas ou outros diagramas de fluxo, pois não seguem hierarquias organizacionais ou direcionamentos, e sim possuem relação de significados e relações significativas. Buscam fazer as relações entre conceitos e, também, não devem ser comparados ou confundidos com o que são mapas mentais que são associacionistas. São utilizados como ferramenta de aprendizagem em inúmeras áreas, e terão como objetivo neste trabalho, auxiliar a disciplina de Análise de Sistemas nas aulas de Análise de Requisitos. De acordo com Moreira e Masini (2011, p. 51), “deve-se entender por mapas conceituais, diagramas bidimensionais mostrando reações hierárquicas entre conceitos de uma disciplina e que derivam sua existência da própria estrutura da disciplina”.

A utilização de mapas conceituais em sala de aula, não dispensa as explicações do professor e sim, devem ser utilizados para auxiliar tanto aluno quanto professor, pois podem ser usados entre conceitos que estão sendo utilizados em uma aula, numa unidade de estudo ou em curso inteiro (MOREIRA E MASINI, 2011, p. 55). O objetivo da construção de mapas conceituais é representar a relação entre os conceitos, um instrumento ou técnica para utilização da aprendizagem significativa de Ausubel.

A utilização de mapas conceituais, pelo aluno no seu processo de aprendizagem poderá contribuir em determinados assuntos, mesmo que ainda não possua muitos subsídios, poderá identificar lacunas e poderá buscar fontes de conhecimento para preenchê-las.

O aluno que aprende a trabalhar com mapas conceituais poderá até mesmo ganhar conhecimentos dos conteúdos ministrados em um determinado momento e o mais importante é que poderá desenvolver uma autonomia para construir sua própria aprendizagem em qualquer momento que desejar.

2.4.2 Construção de Mapas Conceituais como técnica de aprendizagem significativa

Quando se decide construir mapas conceituais, buscam-se modelos ou regras, porém existem vários modelos de mapas conceituais, porém serão citados apenas dois: tipo teia de aranha ou do tipo fluxograma. No modelo aranha, o conceito principal é colocado no centro e os demais conceitos serão ramificados a partir do conceito central (TAVARES, 2007, p.4). Já no conceito de mapa do tipo fluxograma mostra os conceitos com uma organização linear, geralmente demonstra uma sequência de procedimentos que encontra um ponto final ou alguns pontos finais. Ainda dentro do tipo fluxograma, encontramos um sistema que trabalha de forma hierárquica e, também, possui dados de entrada, saída e de possibilidades (TAVARES, 2007, p.5).

Para a utilização de mapas conceituais em sala de aula, podem ser definidas algumas regras ou normas, como a escolha da figura a ser utilizada, ou da organização piramidal que serve apenas para determinar a situação do momento, mas o que realmente deve existir é a representação entre os conceitos que estão sendo utilizados. Para Moreira (2010, p. 15), “o importante é que o mapa seja instrumento capaz de evidenciar significados atribuídos a conceitos e relações entre conceitos no contexto de um corpo de conhecimentos”.

Mesmo que existam alguns acordos pré-estabelecidos em sala de aula para criação dos mapas, pode-se considerar que o autor poderá escolher o modelo que melhor lhe convier, ou até mesmo criar o seu modelo. De acordo com Moreira (2010,

p. 33), “*cabe lembrar que os mapas conceituais não são autoexplicativos. Devem sempre ser explicados pelo autor que ao fazê-lo explicita seu entendimento sobre o assunto*”.

Cada mapa possui sua particularidade, é construído através de conceitos de um indivíduo, mas o que importa é que o autor seja capaz de explicar o significado e a relação que ele vê entre os conceitos. Segundo Moreira (2010, p. 16), “é uma técnica muito flexível e, em razão disso, pode ser usado em diversas situações, para diferentes finalidades” e é muito relevante a relação com a teoria da aprendizagem significativa. Lembrando que a aprendizagem significativa trabalha com a ancoragem de conceitos existentes aos novos conceitos, os mapas irão auxiliar para que estes conceitos novos interajam, ou até mesmo haja a explicitação entre o novo conceito e os já existentes. Porém, uma abordagem ausubeliana, em um determinado currículo, determina antes da criação do mapa, que sejam seguidos os seguintes parâmetros (MOREIRA, 2010, p. 22):

1. Identificar a estrutura de significados aceita no contexto da matéria de ensino;
2. Identificar os subsunçores (significados) necessários para a aprendizagem significativa da matéria de ensino;
3. Identificar os significados pré-existentes na estrutura cognitiva do aprendiz;
4. Organizar sequencialmente o conteúdo e selecionar materiais curriculares, usando as ideias de integrativa como princípios programáticos;
5. Ensinar usando organizadores prévios, para fazer pontes entre os significados que o aluno já tem e os que ele precisaria ter para aprender significativamente a matéria de ensino.

Podem-se utilizar mapas conceituais em qualquer uma das etapas acima citadas, pois mapas conceituais podem ser traçados por professores como recursos didáticos e, também, por alunos para demonstração de resultados a serem avaliados, ambos possuem componentes idiossincráticos, que significa que não existe mapa conceitual correto (MOREIRA, 2010, p.22).

O aluno quando solicitado a construir um mapa não poderá ser avaliado como certo ou errado e sim, se deve avaliar se houve aprendizagem significativa ou não.

Mapas conceituais podem ser construídos para quaisquer assuntos de qualquer área de conhecimento. Uma vez que o aluno sinta necessidade e consiga expressar sua aprendizagem de forma significativa, ele deve ser incentivado.

3 DESCRIÇÃO DA METODOLOGIA

3.1 CARACTERIZAÇÃO DA PESQUISA

3.1.1 Abordagem da pesquisa

Esta pesquisa é classificada como **exploratória**, utilizando os princípios descritos por Gil (2002, p. 42), “pois se pode dizer que estas pesquisas têm como objetivo principal o aprimoramento de ideias ou a descoberta de intuições”.

Quanto à natureza caracteriza-se como **qualitativa**, pois não irá trabalhar com ferramentas e questões estatísticas no seu processo de análise do problema e, portanto, não haverá generalização do problema (RICHARDSON, 1999, p. 70).

Esta pesquisa também possui origem **interpretativa**, trabalhou com cenário de sala de aula, e suas interpretações serão realizadas no contexto geral com uma visão holística das etapas e resultados alcançados. A análise dos dados realizados envolveu a contextualização, onde os resultados são analisados com referência ao grupo, cenário ou evento em particular que estiver sendo observado (MOREIRA E CALEFFE, 2008, p.86).

Esta pesquisa é classificada como aplicada, por se tratar de aplicação prática, pois possui o objetivo de utilizar os conhecimentos adquiridos como ferramenta de apoio ao ensino. Está relacionada a interesses e verdades locais e dirigida para solução de problema específico (SILVA E MENEZES, 2001, p.20).

Esta pesquisa foi aplicada em sala de aula com avaliações por etapas cumpridas e buscou contribuir com o ensino de análise de requisitos de *software*. Desenvolveu resultados que possam servir de reflexões para docentes que estão preocupados com o processo de ensino-aprendizagem de diversas áreas, além de contribuir com os saberes e demandas regionais dos cursos técnicos.

3.1.2 Local e Universo da Pesquisa

É necessário identificar a população para a pesquisa. Para Richardson (1999, p. 157) a população da pesquisa refere-se ao “conjunto de elementos que possuem determinadas características em comum”.

Nesta pesquisa, o universo são os alunos devidamente matriculados no Curso Técnico em Informática, na modalidade subsequente do Instituto Federal do Paraná do Campus Irati, da cidade de Irati-PR.

O quadro 1 mostra a organização curricular do Curso Técnico em Informática.

Este trabalho poderá contribuir com o ensino de duas disciplinas: Análise de Sistemas onde o aluno aprende os conceitos da análise de requisitos, que é foco principal da pesquisa e a disciplina de Trabalho de Conclusão de Curso (TCC) que é responsável por acompanhar a elaboração e execução completa de um projeto de *software*. É na disciplina TCC que o aluno irá colocar em prática o que aprendeu sobre análise de requisitos na disciplina de Análise de Sistemas. As disciplinas ocorrem no terceiro e quatro semestres do curso respectivamente.

1º Semestre	2º Semestre
Informática Instrumental Lógica de Programação Matemática Inglês Instrumental Organização e Arquitetura de Computadores Algoritmos	Programação e Sistemas Computacionais Modelagem de Sistemas Banco de Dados Português Instrumental Sistemas Operacionais Empreendedorismo Estatística Básica
3º Semestre	4º Semestre
Programação Orientada a Objetos Análise de Sistemas Técnicas de Banco de Dados Legislação e Ética Web Design Metodologia Científica Aplicada a Projetos	Técnicas Avançadas de Programação Programação WEB Trabalho de Conclusão de Curso Redes Tópicos Especiais em Tecnologia da Informação

Quadro 1 – Unidade curricular do curso técnico em informática

Fonte: Departamento de ensino – IFPR – Campus Irati

A ementa da disciplina de **Análise de Sistemas** prevê os conteúdos: Princípios de engenharia de software; Análise de requisitos, Análise

Orientada a objeto; Especificação, verificação e validação de software; Ferramentas CASE. Sugere como autores da bibliografia básica PRESSMAN, R. S e SOMMERVILLE, I.

A ementa da disciplina de **Trabalho de Conclusão de Curso** prevê: Elaboração de um projeto que contribua para a melhoria da automação, do desempenho, da eficiência e da racionalização dos serviços administrativos das organizações. Caracterização da natureza e objetivos do projeto. Revisão bibliográfica, análise e projeto, execução de acordo com a escolha realizada. A bibliografia a ser utilizada varia de acordo com cada projeto, a critério do orientador.

O curso possui 1360 h/a, divididas em quatro semestres letivos, os quais possuem disciplinas técnicas voltadas à análise e desenvolvimento de *software*.

O TCC torna-se o maior desafio para os alunos do curso, e a primeira etapa deste trabalho é a análise de requisitos. Como o trabalho envolve todas as etapas de um projeto de software, se não houver aprendizagem dos conceitos, os mesmos levam um tempo maior para o desenvolvimento da análise de requisitos de *software*, e, conseqüentemente, comprometem o tempo de todo o projeto.

3.1.3 Instrumento de coleta de dados

Para atingir os resultados esperados é necessário analisar os dados que são extraídos com o auxílio de instrumentos de coleta de dados. De acordo com Marconi & Lakatos (2010, p. 149), esta é “etapa da pesquisa em que se inicia a aplicação dos instrumentos elaborados e das técnicas selecionadas, a fim de se efetuar coleta de dados previstos”.

Os instrumentos utilizados foram: *pesquisa de campo e entrevista*. A pesquisa de campo envolve um experimento, em que se deve selecionar e enunciar o problema, levando em consideração a metodologia apropriada; apresentar os objetivos, sem perder de vista as metas práticas; estabelecer a amostra correlacionada com a área de pesquisa; o universo dos seus componentes; introduzir os estímulos; e, controle e medida dos efeitos (MARCONI & LAKATOS, 2010, p. 169).

Neste trabalho a pesquisa de campo foi realizada através dos exercícios solicitados em fases que foram divididos para medir os efeitos no que diz respeito à aprendizagem significativa no ensino de análise de requisitos de software.

Segundo Severino (2012, p. 124), “entrevista é técnica de coleta de informações sobre um determinado assunto, diretamente solicitadas aos sujeitos pesquisados. Trata-se, portanto, de uma interação entre pesquisador e pesquisado”.

Após a aplicação das atividades em sala de aula através dos exercícios desenvolvidos pelos alunos, foi realizada a entrevista do tipo *não estruturada e não dirigida*.

Para Marconi & Lakatos (2010, p. 180), a entrevista *não estruturada* “é uma forma de poder explorar mais amplamente uma questão. Em geral, as perguntas são abertas e podem ser respondidas dentro de uma conversação informal”.

Nesta entrevista os alunos expressam sua opinião a respeito da estratégia aplicada com a finalidade de externar um parecer, no caso se favorável ou não e sua justificativa e observações.

A entrevista *não dirigida* está dentro do grupo da entrevista não estruturada e de acordo com Marconi & Lakatos (2010, p. 180), “há liberdade total por parte do entrevistado, que poderá expressar suas opiniões e sentimentos, a função do entrevistador é incentivo, levando o informante a falar sobre determinado assunto, sem, entretanto, forçá-lo a responder”.

Neste trabalho os alunos responderam uma pergunta com respostas de livre discurso, da opinião a respeito da contribuição da utilização de mapas conceituais no ensino de análise de requisitos de software.

3.1.4 Métodos de análise dos dados

A pesquisa de cunho qualitativo será avaliada através de análise interpretativa, de acordo com Marconi & Lakatos (2010, p. 152):

Interpretação é a atividade intelectual que procura dar um significado mais amplo às respostas, vinculando-as a outros conhecimentos. Em geral, a interpretação significa a exposição do verdadeiro significado do material apresentado em relação aos objetivos propostos e ao tema.

A cada tarefa realizada em aula foi feita uma análise da validação da ferramenta e sua contribuição no desenvolvimento das atividades do aluno.

Após a análise, os dados serão apresentados através de árvore de associações que darão visibilidade ao encadeamento de repertórios. Árvore de associações, segundo Spink (1999, 114), “permite visualizar o fluxo das associações de ideias inaugurado pela pergunta do entrevistador e encerrado com suas sínteses, com as afirmações conclusivas do entrevistado”.

Os referidos dados serão apresentados por árvore de associações de algumas aulas e da entrevista final realizada com os alunos que participaram do projeto.

4 DESENVOLVIMENTO DAS ATIVIDADES E ANÁLISE DOS RESULTADOS

Para realização deste trabalho as atividades foram divididas em etapas. Por se tratar de atividade prática aplicada em sala de aula cada etapa será denominada como Aula.

Cada aula foi aplicada em datas distintas com intervalos de uma a duas semanas cada uma. As aulas trabalharam o ensino de análise de requisitos de software com aprendizagem significativa e construção de mapas conceituais. Cada aula abordou um tema e um objetivo a ser alcançado.

As aulas foram compostas de momentos teóricos e expositivos e aplicações práticas que utilizam o TCC como material de aplicabilidade e validação das teorias estudadas.

A aprendizagem significativa vem sendo aplicada em várias áreas do ensino e com sucesso nos resultados, porém pouco se encontra sobre os ensinamentos técnicos que, por sua natureza, possuem uma visão operacional tirando o papel do ensino e colocando aulas como treinamentos repetitivos e focados apenas em cumprir uma ementa pré-formatada para o mercado de trabalho. Devido a este estereótipo os cursos técnicos acabam sendo resumidos em aprendizagem mecânica, o que os tornam exaustivos e, com isto, ocasionam um grande número de desistências e evasões, devido às dificuldades encontradas para acompanhar as disciplinas.

Esta sequência de aulas deste trabalho tem como objetivos:

- Sugerir uma estratégia para ensino de análise de requisitos, saindo da tradicional aula expositiva onde conceitos são impostos de forma mecânica e possibilitando uma aprendizagem significativa dos conceitos onde os alunos irão compreendê-los antes de colocar em prática as técnicas de elicitação de requisitos.
- Instigar o aluno a buscar e ancorar subsunções de forma simples e através desta simplicidade perceber o novo significado por conta própria.

- Colocar em prática a busca pelos significados e as ligações entre eles através de um exercício realizado com material de análise de requisitos do TCC.
- Analisar a contribuição da utilização dos mapas conceituais no ensino de análise de requisitos.

.

As atividades foram realizadas em sequência de aulas que serão detalhadas nos subitens a seguir como:

- Aula 1 – Aula expositiva e organizadores prévios.
- Aula 2 – Criação de subsunções.
- Aula 3 – Análise de requisitos com enfoque aos conceitos trabalhados.
- Aula 4 – Apresentando mapas conceituais.
- Aula 5 – Análise de requisitos na prática.
- Aula 6 – Buscando principais conceitos na análise de requisitos.
- Aula 7 – Alunos constroem mapas conceituais.

Os resultados serão apresentados de forma fracionada, observando aula por aula, obedecendo à ordem temporal dos acontecimentos.

4.1 AULA 1 – AULA EXPOSITIVA E ORGANIZADORES PRÉVIOS

4.1.1 Desenvolvimento das atividades da aula 1

A Aula 1 é uma síntese do conteúdo geral ministrado na aula de análise de requisitos e vale salientar que a aula completa, de forma tradicional, foi ministrada para estes alunos no semestre anterior (segundo semestre do curso). Pode-se considerar, neste caso, que estes alunos já possuem algum conhecimento sobre o assunto.

Foram apresentados os conteúdos de forma expositiva, fazendo uma espécie de revisão ainda utilizando a aprendizagem mecânica. Segundo Santos (2009, p.55):

A aprendizagem mecânica é necessária e inevitável no caso de conceitos inteiramente novos para o aprendiz, mas posteriormente, ela se transformará em significativa. Para acelerar esse processo, Ausubel propõe os organizadores prévios, âncoras criadas a fim de manipular a estrutura cognitiva, interligando conceitos aparentemente não relacionáveis por meio de abstração.

Nesta aula os alunos retomam o conteúdo já visto, porém mesmo não sendo o primeiro contato que os alunos tiveram com análise de requisitos, percebe-se que o todo material teórico é algo novo para a maioria dos alunos. Na turma onde este trabalho foi realizado apenas um aluno já trabalhava na área e possuía algum conhecimento técnico a respeito. Pode-se perceber que pouco restou das aulas teóricas do semestre passado.

- Conteúdo da aula:

Aula de revisão sobre análise de requisitos de *software* (Método Tradicional):

Esta aula teve com objetivo trazer os conteúdos já vistos em sala de aula, na disciplina de modelagem. Foram discutidos os conceitos e métodos que podem ser utilizados.

- Síntese do conteúdo da aula:

a) O que são Requisitos?

Para responder utilizamos a definição de Machado (2011, p. 24), “os **requisitos** são objetivos ou restrições estabelecidas por clientes e usuários do sistema que definem as diversas propriedades do sistema”. Esta é uma definição clássica de requisitos, define que é tudo aquilo que o sistema irá fazer ou suas restrições, a que chamamos de requisitos.

b) O que são Requisitos Funcionais?

Nos primeiros contatos existe a preocupação de esclarecer as funcionalidades do sistema, os *stakeholders* (todos os interessados e envolvidos no projeto) se preocupam em definir o que o sistema irá fazer para atender seus objetivos. Os requisitos funcionais determinam o que o sistema deve fazer ainda

sem se preocupar como irá fazer. “são declarações de serviços que o sistema deve fornecer, de como o sistema deve reagir a entradas específicas e de como o sistema deve se comportar em determinadas situações” (SOMMERVILLE, 2011, p. 59).

c) O que são Requisitos não Funcionais?

São restrições aos serviços ou funções oferecidas pelo sistema. Incluem restrições no processo de desenvolvimento e restrições impostas pelas normas. Os requisitos não funcionais, não dizem respeito às funcionalidades do sistema, porém são relevantes para o projeto e descrevem atributos do sistema ou do ambiente. Referem-se à usabilidade, confiabilidade, desempenho, suportabilidade, restrições do projeto, requisitos de implementação, requisitos de interface e requisitos físicos, desenvolvimento e restrições impostas pelas normas (SOMMERVILLE, 2011, p. 59).

d) Quais os objetivos da análise de requisitos?

- Definir as prioridades no conjunto de requisitos.
- Modelar os requisitos elicitados a fim de obter os conceitos relevantes para resolver problemas propostos.
- Analisar cada requisito para resolver possíveis ambiguidades e conflitos entre os requisitos.

e) Como fazer a coleta de requisitos?

As técnicas utilizadas para Levantamento de Requisitos:

- **Entrevistas:** é uma técnica tradicional, simples e que produz bons resultados, além de aproximar o cliente do desenvolvedor, porém com os seguintes cuidados:

- Para evitar dispersão do assunto é necessário que seja feito antecipadamente um roteiro dos assuntos essenciais.
- Solicitar autorização prévia para realizar a entrevista e agendar dia e hora.
- Planejar o tempo da entrevista para não dispersar o assunto.

- Registrar a entrevista em anotações em tempo real ou até mesmo com utilização de gravador para que as informações não se percam.
- Documentar tudo que foi tratado na entrevista.
- Tomar cuidado com o vocabulário técnico, para que o usuário não tenha dúvidas no assunto.
- No planejamento da entrevista, fazer um breve estudo da empresa. (MACHADO, 2011, p. 146).

- **Questionários:** é bem sugerido quando existem vários usuários e estes encontram-se geograficamente em lugares distintos.

- Os questionários podem ser elaborados com questões de múltipla escolha, lista de verificações ou com espaços em branco.
- Segundo Sommerville (2011, p. 59), todas as pessoas envolvidas no questionário devem ser identificadas para que seja analisado o grau de envolvimento com o processo.
- Sempre que houver uma pesquisa deve ser acompanhada por um documento explicativo e esclarecedor (MACHADO, 2011, p. 148).
- Existem outras técnicas de realizar coletas de requisitos, porém trabalharemos no momento com estas duas.

Para iniciar a aprendizagem significativa Ausubel sugere a utilização de organizadores prévios para auxiliar na delimitação do conteúdo a ser trabalhado.

Após a retomada de conteúdo sobre análise de requisitos a aula segue com a apresentação dos principais conceitos como ideias centrais e sugerindo alguns significados familiares que servirão de organizadores prévios, serão apresentados de forma gráfica. Utilizaremos as palavras de Moreira e Masini (2011, p. 107) para esclarecimentos sobre o que é um organizador prévio:

Material introdutório apresentado antes do material a ser aprendido, porém em nível mais alto de generalidade, inclusividade e abstração do que o material em si e, explicitamente relacionado às ideias relevantes existentes na estrutura cognitiva e à tarefa de aprendizagem. Destina-se a facilitar a aprendizagem significativa, servindo de ponte entre o que o aprendiz já sabe e o que ele precisa saber para que possa aprender o novo material.

Esta etapa irá trabalhar como uma preparação para a próxima aula que se refere aos subsunçores, “Ausubel recomenda o uso de organizadores prévios que sirvam de âncora para a nova aprendizagem e levem ao desenvolvimento de conceitos de subsunçores que facilitem a aprendizagem subsequente” (MOREIRA E MASINI, 2011, p. 21).

Os principais conceitos foram colocados e algumas definições sugeridas, para que os alunos iniciem o exercício de buscar conceitos existentes em estrutura cognitiva.

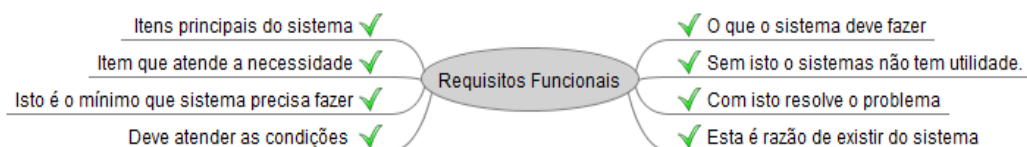


Figura 5 - Requisitos Funcionais

Fonte: O autor

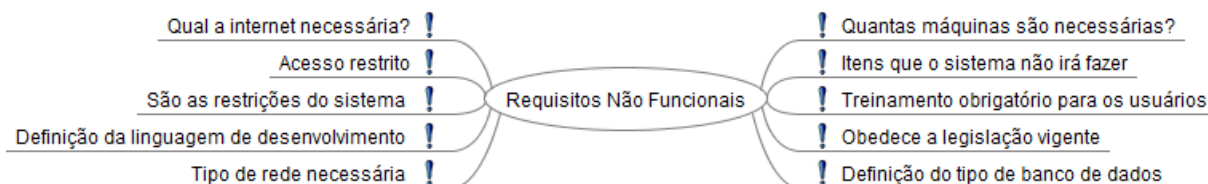


Figura 6 - Requisitos Não Funcionais

Fonte: O autor

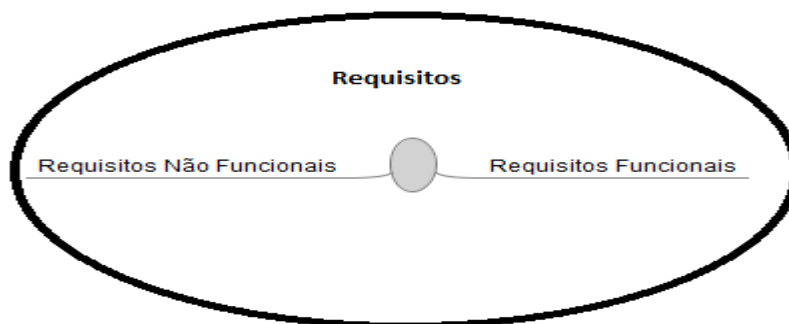


Figura 7 - Requisitos
Fonte: O autor

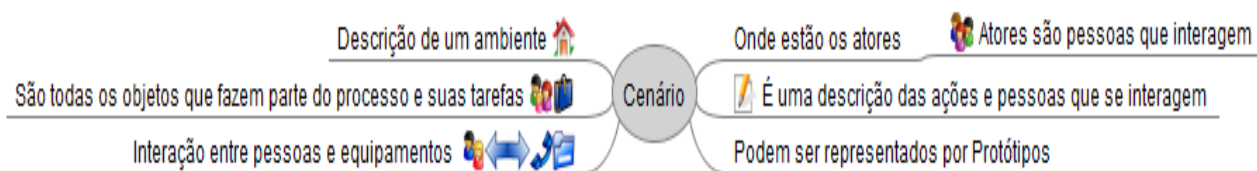


Figura 8 - Cenário
Fonte: O autor

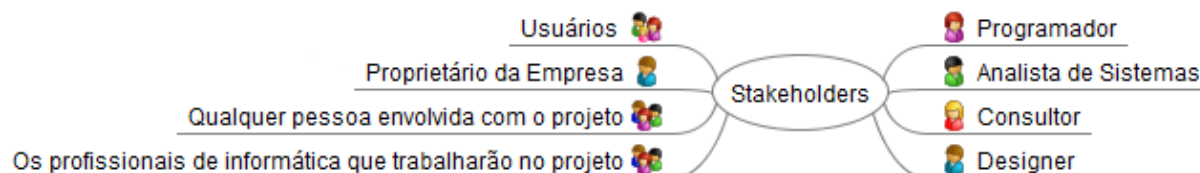


Figura 9 - Stakeholders
Fonte: O autor



Figura 10 - Cliente
Fonte: O autor



Figura 11 - Escopo
Fonte: O autor

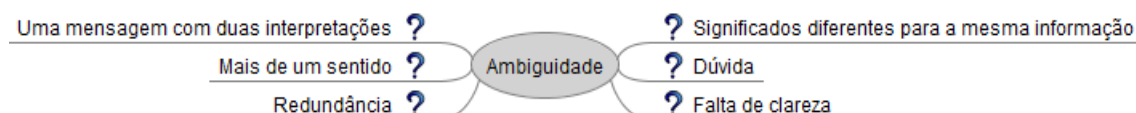


Figura 12 - Ambiguidade
Fonte: O autor

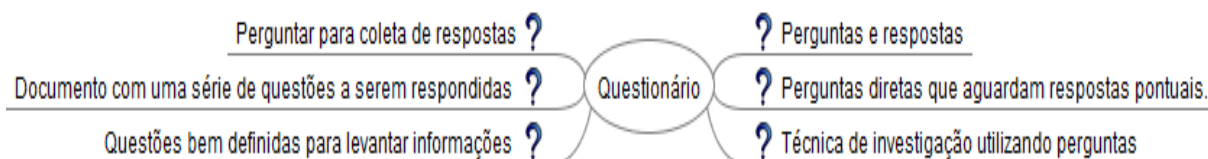


Figura 13 - Questionário
Fonte: O autor

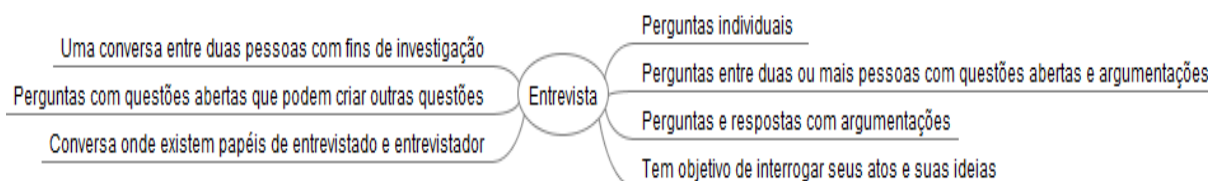


Figura 14 - Entrevista
Fonte: O autor



Figura 15 - Documento de requisitos

Fonte: O autor



Figura 16 - Administrador

Fonte: O autor

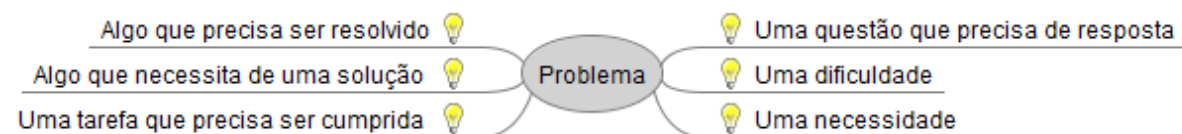


Figura 17 - Problema

Fonte: O autor

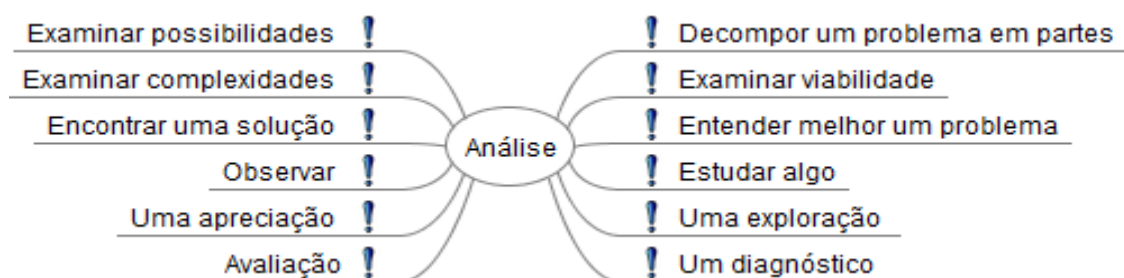


Figura 18 - Análise

Fonte: O autor



Figura 19 - Projeto

Fonte: O autor

Nesta aula foram vistos conceitos isolados para que os alunos busquem conceitos já existentes em sua estrutura cognitiva e consigam fazer a aquisição de significados, com o objetivo de prepará-los para receber o conteúdo da aula de forma significativa. Com a aula 1 cumpriu-se a etapa da criação de organizadores prévios. Este momento de aprender o significado é importante, segundo Ausubel, Novak e Hanesian (1980, p. 43)

A aprendizagem do significado de palavras isoladas ou a aprendizagem do que as palavras representam envolve a aprendizagem significativa de proposições particulares de equivalência representacional – aprendizagem do que as palavras representam e, portanto, significam psicologicamente as mesmas coisas que seus referentes.

A ideia central com conceitos sugeridos fará uma ponte para os conceitos que o aluno já possui.

4.1.2 Análise dos resultados da aula 1

Na visão do professor: o professor realizou uma aula de resgate de um conteúdo já visto e ministrado no semestre anterior e a maioria dos alunos não se recordava do conteúdo da aula. Quando os conceitos foram colocados como ideias centrais e associações foram feitas, os alunos começaram a compreender com maior facilidade.

Nesta aula ainda observou-se a importância e contribuição dos materiais utilizados como organizadores prévios, neste caso as figuras, mostrando a ideia central conceitos e exemplos associados que vão de acordo com as palavras de

Moreira e Masini (2011, p. 107) para esclarecimentos sobre o que é um organizador prévio:

Material introdutório apresentado antes do material a ser aprendido, porém em nível mais alto de generalidade, inclusividade e abstração do que o material em si e, explicitamente relacionado às ideias relevantes existentes na estrutura cognitiva e à tarefa de aprendizagem. Destina-se a facilitar a aprendizagem significativa, servindo de ponte entre o que o aprendiz já sabe e o que ele precisa saber para que possa aprender o novo material.

Quanto ao tempo: As atividades da aula1 foram desenvolvidas em um tempo total de cento e cinquenta minutos, sendo três horas aula de cinquenta minutos cada. Divididos em duas horas aula em dia e uma hora aula em outro dia que teve um intervalo de sete dias.

4.2 AULA 2 – CRIAÇÃO DE SUBSUNÇORES

4.2.1 Desenvolvimento das atividades da aula 2

Esta aula tem como objetivo a busca por subsunçores. Foram utilizadas as mesmas palavras trabalhadas na Aula 1 em um trabalho onde se solicita que o aluno faça, de forma descontraída as suas definições, sendo que podem ser apenas um exemplo, esquecendo as definições técnicas e formatas.

O trabalho foi desenvolvido com a equipe de trabalho de conclusão de curso (dupla ou individual). A ideia de trabalharem com a dupla do TCC possibilita a troca de informações.

Os professores trabalham em sala de aula com exemplos para expor suas ideias e facilitar o aprendizado, e neste trabalho, foi proposto o inverso, ou seja, os exemplos e definições partem dos alunos, que foram instigados a buscar as informações existentes em sua estrutura cognitiva.

Foi solicitado para que os alunos relacionassem com algum conceito que conseguissem buscar em sua estrutura cognitiva, sendo através de exemplos, descrições, associações, comparações ou alguma outra forma que o aluno encontrasse para criar o seu próprio conceito a respeito da palavra, pois “à medida

que a aprendizagem começa a ser significativa, esses subsunçores vão ficando cada vez mais elaborados e mais capazes de ancorar novas informações” (MOREIRA E MASINI, 2011, p. 19).

- Trabalho proposto:

Descreva um exemplo para cada palavra abaixo, sem se preocupar em relacioná-la com a Engenharia de Software.

- 01 – Requisitos
- 02 – Requisito Funcional
- 03 – Requisito não funcional
- 04 – Cenário
- 05 – *Stakeholders*
- 06 – Cliente
- 07 – Escopo
- 08 – Ambiguidade
- 09 – Questionário
- 10 – Entrevista
- 11 – Documentos de Requisitos
- 12 – Administrador
- 13 – Problema
- 14 – Análise
- 15 – Projeto

São apresentadas apenas as respostas de quatro palavras como exemplo, sendo que as respostas das quinze palavras propostas estarão nos apêndices deste trabalho.

01 – Requisitos - Definição: Para Machado (2011, p. 24), “os **requisitos** são objetivos ou restrições estabelecidas por clientes e usuários do sistema que definem suas diversas propriedades”.

Respostas:

1. Para o computador ligar é necessário que tenha energia elétrica.
2. Para jogar truco é necessário conhecer as regras.
3. Um sistema para playground precisa cadastrar crianças, cadastrar responsáveis e emitir pulseiras.
4. Em uma nota fiscal para preencher o destinatário é preciso ter CPF ou CNPJ.
5. É necessário possuir um cadastro de clientes.
6. É necessário informar o CPF do cliente.
7. São Informações absorvidas de um cliente, todas as operações que um sistema deverá executar.
8. Para ser eleitor é necessário ter no mínimo 16 anos.
9. O sistema de venda precisa ter cadastro.

02 – Requisitos Funcionais: Segundo Sommerville (2011, p. 59), “são declarações de serviços que o sistema deve fornecer, de como o sistema deve reagir a entradas específicas e de como o sistema deve se comportar em determinadas situações”.

Respostas:

1. Para o carro funcionar é necessário ter combustível.
2. Possuir um baralho com 4 cartas sendo 4 cartas de Ás ao 7 e quatro de cada uma.
3. O cadastro de cliente (criança) será vinculado com o cadastro do responsável.
4. O sistema deve permitir a inclusão, exclusão e alteração dos clientes.
5. Precisa inserir RG e CPF.
6. Cadastro de cliente.
7. Para cadastrar o tecido que será utilizado na fabricação de uma camiseta é necessário calcular as medidas.
8. Para fazer título de eleitor tem que ter RG.
9. O cliente precisa ter o programa Java instalado.

03 – Requisito não funcional: Segundo Sommerville (2011, p. 59), “são restrições aos serviços ou funções oferecidas pelo sistema. Incluem restrições no processo de desenvolvimento e restrições impostas pelas normas”.

Respostas:

1. Tipo de monitor.
2. Possuir um marcador caracterizado.
3. O sistema do playground não terá interação com o sistema da loja.
4. O sistema não fará a baixa automática dos cheques pendentes.
5. Declarar renda.
6. Windows XP.
7. A cor será um parâmetro para distinguir um tecido de outro.
8. Possuir um endereço de permanência.
9. O sistema deve conter cor azul.

04 – Stakeholder: De acordo com Machado (2011, p. 30), “é qualquer pessoa materialmente afetada pelo resultado do projeto: clientes, usuários diretos e indiretos, investidores, acionistas, fornecedores, supervisores, gerentes, compradores”.

Respostas:

1. Gerente.
2. São as pessoas envolvidas direta e indiretamente ao sistema. Ex: administrador, colaboradores e clientes.
3. Analista, programador, cliente e funcionários.
4. Quem usa o sistema, funcionário ou gerente.
5. Proprietário e funcionários que utilizam o sistema.
6. Funcionários.
7. Clientes, funcionários, fornecedores.
8. Mesário, presidente, candidato, eleitor.
9. Cliente, chefe, programador.

Pelas respostas da atividade da aula 2, pôde-se observar que alunos conseguiram, mesmo de uma forma arbitrária, buscar subsunçores, e estes estão de certa forma, relacionados com o conteúdo das aulas.

O objetivo é que os alunos desenvolvam a partir do trabalho proposto na aula 2, meios próprios de fazer buscas por subsunçores, em sua estrutura cognitiva a fim de ancorar os novos conceitos, facilitando a aprendizagem e, por fim, obtendo a aprendizagem significativa na aula de análise de requisitos e, futuramente, nas demais aulas.

As demais respostas constam no apêndice A.

4.2.2 Análise dos resultados da aula 2

Na visão do professor: a base da aprendizagem significativa é busca de ideia âncora que funciona como subordinador, denominado por Ausubel como subsunçor. “a aprendizagem significativa ocorre quando a nova informação se relaciona com um aspecto relevante na estrutura do conhecimento do indivíduo” (MOREIRA E MASINI, 2011, p.17).

Quando o aluno buscou em sua estrutura cognitiva conceitos isolados, teve a liberdade de contribuir para a criação de conceitos, mesmo que não sejam tão rebuscados. Entretanto, havendo aprendizagem, os conceitos podem ser alterados e, por fim, terá o conceito esperado.

Pode-se comprovar que ao instigar a busca por subsunçores os alunos passaram a analisar conceitos não mais como algo novo e assustador e, sim, como tarefas simples.

Um exemplo de uma resposta:

Requisitos

Definição do autor: Para Machado (2011, p. 24), “os **requisitos** são objetivos ou restrições estabelecidas por clientes e usuários do sistema que definem as diversas propriedades do sistema”.

Exemplos de repostas dos alunos:

“Para o computador ligar é necessário que tenha energia elétrica”.

“Para jogar truco é necessário conhecer as regras’.

Observa-se que houve aprendizagem, mesmo com respostas indiretas e primárias, pois neste caso colocam objetivos através da palavra “necessário”. As respostas podem ser observadas no apêndice A.

Quanto ao tempo: As atividades da aula 2 foram desenvolvidas em um tempo total de cem minutos, sendo duas horas aula de cinquenta minutos cada. Realizadas em um mesmo dia.

4.3 AULA 3 – ANÁLISE DE REQUISITOS COM ENFOQUE AOS CONCEITOS TRABALHOS

4.3.1 Desenvolvimento das atividades da aula 3

Esta aula trouxe o conteúdo teórico da aula de análise de requisitos, porém os conceitos colocados já não são mais novidade para os alunos.

Conteúdo da aula na sequência que foi explanado:

- Resumo da aula de Análise de Requisitos:

A **Análise de Requisitos** é a primeira etapa da criação de um software. É necessário traçar e analisar o **cenário** e descobrir quais as necessidades e desejos dos **clientes**. A partir das necessidades e desejos é que extraímos os **requisitos** de software. “Requisitos são objetivos ou restrições estabelecidas por clientes e usuários que definem as diversas propriedades de um sistema”. Os requisitos são divididos em dois tipos: **Requisitos Funcionais** e **Requisitos Não Funcionais**. Nos primeiros contatos existe a preocupação em esclarecer as funcionalidades do sistema, os **Stakeholders** (todos os interessados e envolvidos no projeto) se preocupam em definir o que o sistema irá fazer para atender seus objetivos. Os **requisitos funcionais** determinam o que o sistema deve fazer ainda sem se preocupar como irá fazer.

Os **requisitos não funcionais** não dizem respeito às funcionalidades do sistema, porém são relevantes para o projeto e descrevem atributos do sistema ou do ambiente. Referem-se à usabilidade, confiabilidade, desempenho,

suportabilidade, restrições do projeto, requisitos de implementação, requisitos de interface e requisitos físicos.

As técnicas mais utilizadas para extrair os requisitos são **Questionários** e **entrevistas**. A **entrevista** é uma das técnicas tradicionais mais simples de utilizar e que produz bons resultados na fase inicial de obtenção de dados (Machado, 2011, p. 142). Para fazer uma entrevista é necessário que a mesma seja planejada, ou seja, com questões e roteiros previamente elaborados.

A entrevista deve ser registrada, com todas as questões e considerações dos entrevistados. O uso de **questionários** é indicado, por exemplo, quando há diversos grupos de usuários que podem estar em diversos locais geograficamente diferentes (Machado, 2011, p. 148). O **questionário** pode ser elaborado de várias formas: múltipla escolha, lista de verificações e questões com espaços em branco. Independente do tipo elaborado deve ter por objetivo minimizar o tempo gasto nas suas respostas.

Após a elicitación de requisitos os desenvolvedores irão analisar o **problema**, e gerar o **Documento de Requisitos**, que será um registro dos requisitos e que será assinado pelas partes envolvidas com o objetivo de acordo para início do **Projeto** de software.

No Documento de requisitos é que se define o **escopo** do projeto, ou seja, é descrita a finalidade daquilo que se pretende atingir. No escopo do projeto não pode haver **ambiguidades** (significados diversos para uma mesma mensagem).

Todo projeto deve ter um **administrador**, ou seja, uma pessoa que irá controlar e gerenciar as atividades.

Pode-se observar que a temática principal da Aula 3 foi trabalhada nas aulas anteriores e, no momento da aula, sobre o conteúdo geral, neste caso foi a análise de requisitos, em que os alunos já possuíam conhecimentos sobre os conceitos que foram trabalhados. A partir deste momento o aluno irá fazer a relação entre os conceitos já conhecidos.

4.3.2 Análise dos resultados da aula 3

Na aula 3 os conceitos de análise de requisitos foram trabalhados com familiaridade pois os conceitos de certa forma já faziam parte da estrutura cognitiva do aluno.

Observou-se que na aula 3 houve aprendizagem significativa e que todos acompanharam com certa facilidade, constatou-se isto a partir de observação individual de todas as respostas sobre o conteúdo ministrado, pois “considera-se que existe aprendizagem significativa a partir do momento que os indivíduos estabelecem significado, organizando, transformando e armazenando informações. Porém, esta situação é muito dependente do aprendiz, pois requer esforço em conectar de maneira não arbitrária e não literal, o novo conhecimento com o conhecimento já existente na estrutura cognitiva” (TAVARES, 2010).

Quanto ao tempo: As atividades da aula 3 foram desenvolvidas em um tempo total trinta minutos.

4.4 AULA 4 – APRESENTANDO MAPAS CONCEITUAIS

4.4.1 Desenvolvimento das atividades da aula 4

Na aula 4, os alunos foram apresentados à ferramenta de mapas conceituais como sendo uma ferramenta utilizada em diversas áreas para trabalhar com a aprendizagem significativa.

Nas aulas anteriores foram realizados trabalhos com o objetivo de dar sentido aos conceitos, Os alunos conseguiram entendê-los através da ancoragem de conceitos que já possuíam em sua estrutura cognitiva. A partir daí com os conceitos familiarizados, fez-se necessária a relação entre eles, para que haja aprendizado do conteúdo de análise de requisitos. Para isto utilizaremos como ferramenta os mapas conceituais.

Nenhum aluno que participou desta aula conhecia mapas conceituais ou até mesmo ouviu falar sobre o assunto.

Esta aula mostra o que são os mapas conceituais, para que servem e como utilizá-los.

a) Resumo da aula de Mapas Conceituais:

Mapas conceituais são representações gráficas da relação entre os conceitos, ou seja, “mapas conceituais, ou mapas de conceitos, são diagramas indicando relações entre conceitos, ou entre palavras que usamos para representar conceitos” (MOREIRA 2010, p.11). Os mapas conceituais são formados por figuras e setas, mas não devem ser confundidos com organogramas ou outros diagramas de fluxo, pois não seguem hierarquias organizacionais ou direcionamentos e, sim, possuem relação de significados e relações significativas. Buscam fazer as relações entre conceitos e, também, não devem ser comparados ou confundidos, são mapas mentais associacionistas. São utilizados como ferramenta de aprendizagem em inúmeras áreas. De acordo com Moreira e Masini (2011, p. 51), “deve entender por mapas conceituais, diagramas bidimensionais mostrando reações hierárquicas entre conceitos de uma disciplina e que derivam sua existência da própria estrutura da disciplina”.

O objetivo da construção de mapas conceituais é representar a relação entre os conceitos, um instrumento ou técnica para utilização da aprendizagem significativa de Ausubel. Os mapas são estruturas construídas pelo próprio aluno de forma hierárquica facilitando o aprendizado por meio de construção do conhecimento.

b) Foi apresentada com o modelo esquemático de acordo com a figura 4.

c) Como exemplo de utilização dos Mapas Conceituais, criamos um mapa da Aula 3, com os conceitos da aula de Análise de Requisitos.

Este exemplo foi feito em quadro branco pelo professor com a participação ativa de todos os alunos.

Depois de pronto foi transposto para a ferramenta gráfica *StarUml*.

O software *StarUml* é uma ferramenta de distribuição gratuita que modela vários tipos de diagrama e já estava instalada no laboratório de informática da instituição onde a aula foi realizada. A escolha por esta ferramenta foi apenas pela

facilidade de acesso. No momento da aula foi esclarecido aos alunos que existem várias ferramentas que podem ser utilizadas e que também podem criar seus mapas de forma manual em papel e sem utilizar ferramenta informatizada.

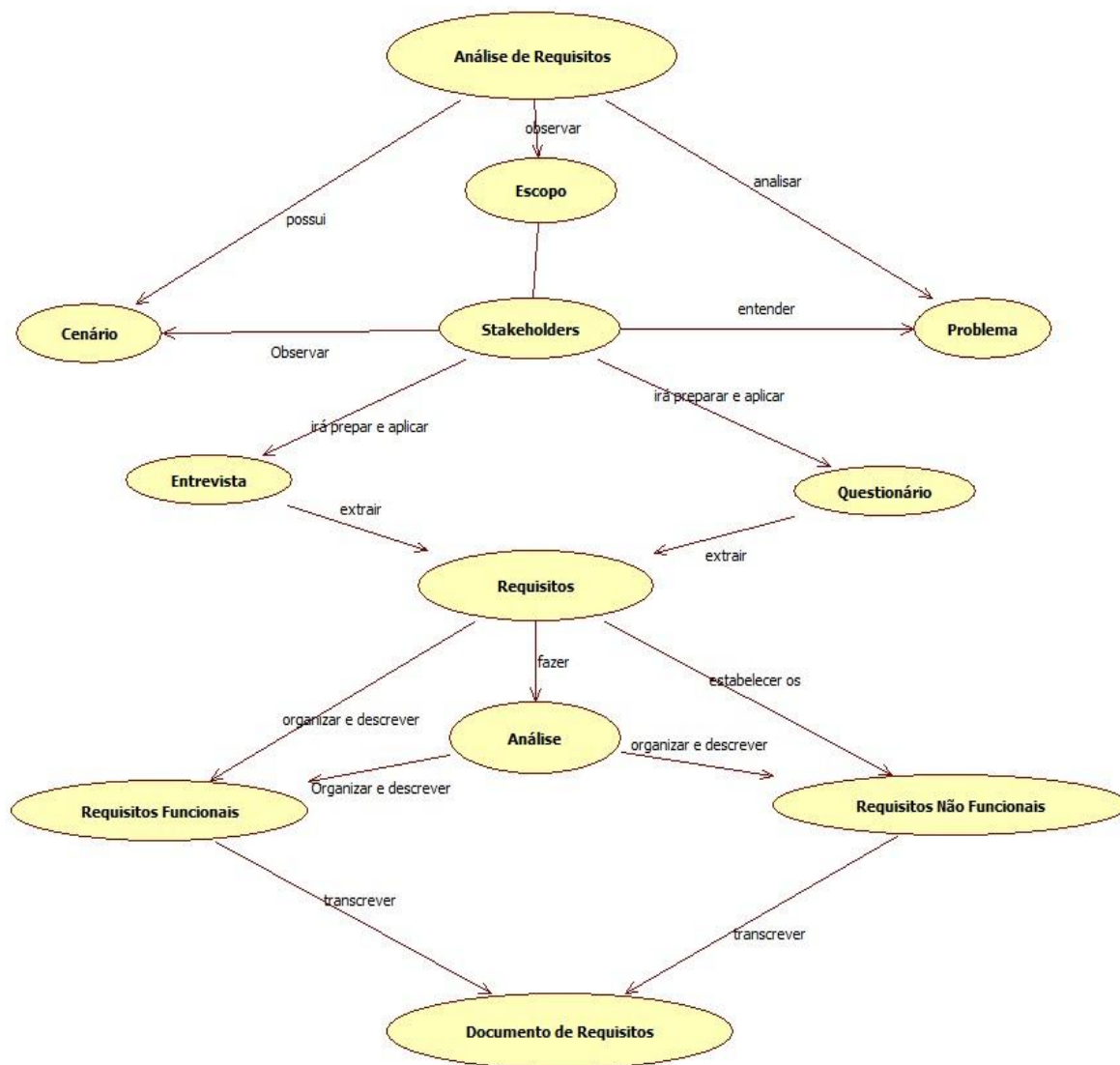


Figura 20 - Mapa Conceitual - Análise de Requisitos
Fonte: o autor

Neste momento da criação do mapa, com o auxílio dos alunos, pode-se observar que os alunos compreenderam os conceitos e as relações entre eles.

4.4.2 Análise dos resultados da aula 4

Na aula 4 os alunos tiveram o primeiro contato com mapas conceituais, pois nenhum dos alunos que participaram desta pesquisa conheciam a ferramenta.

Os alunos auxiliaram e compreenderam a construção do mapa conceitual realizado pelo professor.

De acordo com Moreira e Masini (2011, p. 51), “deve entender por mapas conceituais, diagramas bidimensionais mostrando reações hierárquicas entre conceitos de uma disciplina e que derivam sua existência da própria estrutura da disciplina”.

Conclui-se nesta aula que a partir do momento que os conceitos estão claros para os alunos, os mapas conceituais finalizam a aprendizagem criando as relações que ainda não existiam.

Quanto ao tempo: As atividades da aula 4 foram desenvolvidas em um tempo total de cem minutos, sendo duas horas aula de cinquenta minutos cada. Realizadas em um mesmo dia.

4.5 AULA 5 – ANÁLISE DE REQUISITOS NA PRÁTICA

4.5.1 Desenvolvimento das atividades da aula 5

Até a aula 4 os alunos conheceram o conteúdo da aula de análise de requisitos utilizando aprendizagem significativa e mapas conceituais.

A partir da aula 5 este trabalho caminhará para cumprir outro objetivo que é fazer com que os alunos trabalhem a estratégia utilizada nas aulas anteriores no trabalho de TCC, e tornem a aprendizagem significativa e a construção de mapas conceituais, uma ferramenta para auxiliar de forma autônoma a prática da análise de requisitos.

Irão extrair os requisitos necessários para o projeto do TCC.

Nesta etapa do trabalho de TCC, a proposta para o projeto de software a ser desenvolvido já foi aprovada e possuem uma ligação com uma empresa real que irá

Ihe fornecer informações para que possam desenvolver o seu projeto, a fim de atender uma necessidade proposta pela empresa escolhida pela equipe.

a) Atividade proposta:

Foi estipulado um prazo de duas semanas para que as equipes realizassem o levantamento de requisitos do *software* junto às empresas ligadas ao projeto.

Cada aluno ou equipe irá trazer em forma de redação os principais requisitos funcionais do seu trabalho de TCC. Uma descrição semelhante ao resumo da aula 3.

b) Exemplo1 - Trabalho realizado por uma equipe composta por dois alunos:

Sistema de Controle e Gerenciamento de Farmácias Veterinárias e Pet Shops.

A empresa Farmavet fornece os serviços: vacinas, consultas, exames, banho e tosa. O sistema será operado por um usuário cadastrado no sistema, o qual poderá fazer o cadastro de clientes, animais, funcionários, serviços, compromissos na agenda, gerar relatórios, etc.

O cadastro de clientes abrangerá os campos nome, CPF, RG, telefone, email, data do cadastro e endereço.

Os funcionários também serão cadastrados e abrangerão os seguintes campos: nome, CPF, RG, data de admissão, cargo, CRMV (p/ veterinários), endereço.

Os animais estarão obrigatoriamente ligados a um cliente e serão cadastrados através dos campos: nome, sexo, raça, data de nascimento, código do cliente.

Para o campo raça no cadastro de animais haverá uma tabela na qual serão cadastrados os tipos de raças mais comuns.

Haverá um cadastro de vacinas contendo código da vacina, nome das vacinas, observações.

Serão cadastradas em outra tabela as vacinas realizadas contendo código da vacina, data da aplicação, código do animal, cod. Funcionário.

Em outra tabela serão cadastradas as vacinas futuras contendo cod. Vacina, data da aplicação cod. Animal.

Cada serviço realizado pelo estabelecimento será cadastrado em uma lista de serviços que conterá a descrição do serviço, além de dados como valor e tempo de execução. Esta lista será requisitada pela ordem de serviço a ser aberta quando houver a solicitação de um cliente. Na ordem de serviços será relacionado cliente, animal, funcionário e produtos utilizados para execução do mesmo.

O sistema possuirá uma agenda onde serão marcados os compromissos diários de cada funcionário. Para agendar um serviço é necessária a ligação das tabelas: serviço, cliente, animal, funcionário, além de conter também data e hora.

c) Exemplo 2 - Trabalho realizado por um aluno:

Controle de custo da produção para malharia

O software terá telas que permitam cadastrar, alterar, excluir o lançamento de matéria-prima que suprirá a produção da empresa, também será possível cadastrar os produtos que serão produzidos (produto Acabado).

A função do sistema será levantar os custos da produção, controlando o custo de compra da matéria-prima, desperdícios e os custos fixos para a produção. Realizará também o controle de estoque de produtos acabados que poderão estar no sistema ou serem exportados para outro software já existente.

Receberá informações da entrada de materiais em seu estoque e, para esse controle, será aplicado o sistema de controle de estoque PEPS (Primeiro que Entra Primeiro que Sai), emitirá relatórios de suas rotinas e exportará arquivo para integração com outro software de controle comercial já existente na empresa.

Deverá emitir relatórios para gerenciamento de todos os produtos envolvidos na linha de produção. Deverá existir um controle dos pedidos para que a linha de produção cumpra os prazos e as quantidades solicitadas.

4.5.2 Análise dos resultados da aula 5

Todos os alunos conseguiram realizar a análise de requisitos em um tempo considerado ótimo.

Nesta pesquisa todos os alunos envolvidos conseguiram realizar a análise de requisitos dentro do período proposto. Nas palavras de Machado (2011, p. 24), “os requisitos expressam as características e restrições do produto de *software* do ponto de vista de satisfação das necessidades do usuário”. Este trabalho muitas vezes é difícil até mesmo para profissionais experientes.

Para Sommerville (2007, p.77), “talvez o maior problema que enfrentamos no desenvolvimento de sistemas de *software* grandes e complexos, seja o da Engenharia que Requisitos”. Os diálogos entre os *stakeholders* são passíveis de sofrer ambiguidades e conflitos, o que torna esta fase complexa, pois todos os demais processos dependerão dos requisitos.

Quanto ao tempo: As atividades da aula 5 foram desenvolvidas em um tempo total de cem minutos, sendo duas horas aula de cinquenta minutos cada. Realizadas em um mesmo dia.

4.6 AULA 6 – BUSCANDO PRINCIPAIS CONCEITOS NA ANÁLISE DE REQUISITOS

4.6.1 Desenvolvimento das atividades da aula 6

Nesta aula os alunos terão em mãos o texto que realizaram na Aula 5, contendo os principais requisitos funcionais do projeto a ser realizado no trabalho de TCC.

O objetivo desta aula é que leiam o texto e consigam identificar quais os principais conceitos e consigam defini-los através de conceitos pré-existentes em sua estrutura cognitiva.

- a) O trabalho proposto: Relacionar os principais conceitos do texto construído.

- b) Buscar uma descrição de conceito para cada palavra isolada do texto original. Seguindo os mesmos moldes da aula 2.

Nesta aula os alunos se concentraram apenas nas palavras selecionadas por eles mesmos como sendo as principais, e irão buscar em sua estrutura cognitiva conceitos que poderão ser atribuídos às palavras.

Durante a realização da atividade proposta, percebeu-se que os alunos acharam esta tarefa muito interessante pelo fato de pensarem isoladamente naquilo que escreveram, pois desta forma estavam realmente dando significado ao texto (comentário feito por aluno e outros concordaram com a observação do colega).

- c) Exemplo 1 - Trabalho realizado por uma equipe composta por dois alunos:

Clientes	Quem compra na loja.
Funcionários	Quem trabalha na loja.
Animais	A eles que se destinam os serviços e produtos da loja.
Raça	Cadastro das raças. Ex: cachorro - raça: boxer.
Cadastro vacinas	Medicamento a ser aplicado.
Vacinas realizadas	Histórico de vacinas realizadas em todos os animais cadastrados no sistema.
Vacinas futuras	Cadastro de todas as vacinas que os animais deverão tomar e periodicidade.
Vacinas realizadas	Cadastro de todas as vacinas que o animal já tomou.
Serviços	Atividades realizadas na loja em benefício dos animais e clientes.
Ordem de serviço	Cadastro de um ou vários serviços ligados a um animal de um determinado cliente, que serão executados por um ou vários funcionários.
Agenda	Onde são armazenados compromissos e tarefas da empresa e dos funcionários.

Quadro 2 – Resultado de atividade realizada por aluno

Fonte: O autor

d) Exemplo 2 - Trabalho realizado por um aluno:

Matéria-Prima	Produto ainda não industrializado, pronto para entrar na linha de produção.
Produção	Confecção do produto final estabelecido pela empresa.
Produto Acabado	Produto acabado e pronto para ser comercializado.
Desperdícios	É tudo que a empresa descarta após a produção, os restos, as sobras.
PEPS	“ <i>Primeiro que entra primeiro que sai</i> ” método no qual há uma organização para o controle de produtos guardados em estoque.
Rotinas	Processo de repetição dentro de uma linha de produção.
Gerenciamento	Método para atingir objetivos, otimizando os resultados.
Controle dos pedidos	Controle executado de acordo com a necessidade do cliente e da empresa.

Quadro 3 – Resultado de atividade realizada por aluno

Fonte: O autor

4.6.2 Análise de resultados da aula 6

A tarefa de relacionar os principais conceitos e discriminá-los foi desempenhada com facilidade por ser uma prática conhecida pelos alunos (realizada na aula 2).

O processo da assimilação de conceitos envolve de acordo com Moreira e Masini (2011, p. 20), “ideias relevantes estabelecidas na estrutura cognitiva do aprendiz com o conteúdo potencialmente significativo, implícito na definição dos termos ou das pistas contextuais”. A principal contribuição nesta aula foi tornar claros os requisitos e a interpretação dos conceitos. Todos os alunos realizaram a tarefa com sucesso.

Quanto ao tempo: As atividades da aula 6 foram desenvolvidas em um tempo total de cem minutos, sendo duas horas aula de cinquenta minutos cada. Realizadas em um mesmo dia.

4.7 AULA 7 – ALUNOS CONSTROEM SEUS MAPAS CONCEITUAIS

4.7.1 Desenvolvimento das atividades da aula 7

O objetivo da aula 7 é fazer com os alunos construam seus próprios mapas conceituais, a partir dos conceitos relacionados na análise de requisitos, considerando a atividade realizada na aula 6.

a) Atividade proposta:

Construir um mapa conceitual a partir dos conceitos relacionados e trabalhados na aula 6. Utilizar como apoio as informações da aula 4.

b) Exemplo 1 - Trabalho realizado por uma equipe composta por dois alunos:

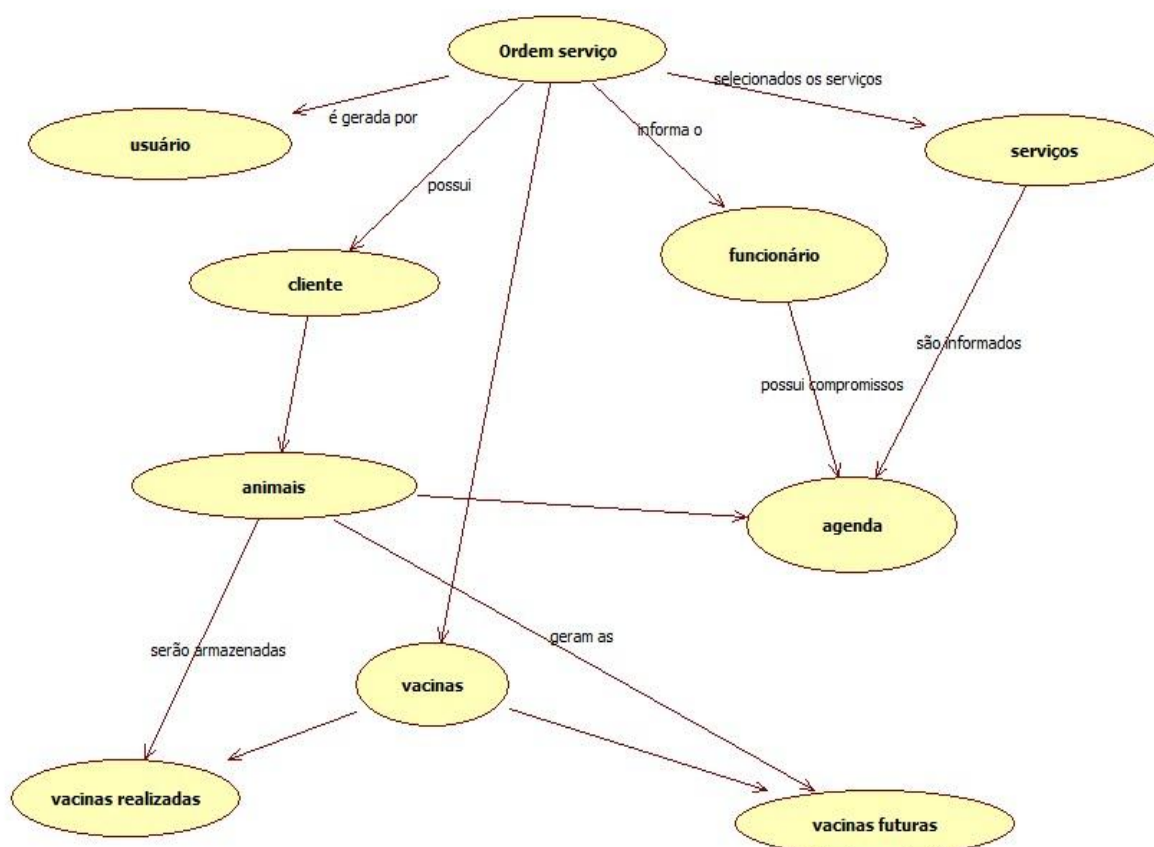


Figura 21 - Mapa conceitual construído por aluno
 Fonte: O autor

c) Exemplo 2 - Trabalho realizado por um aluno:

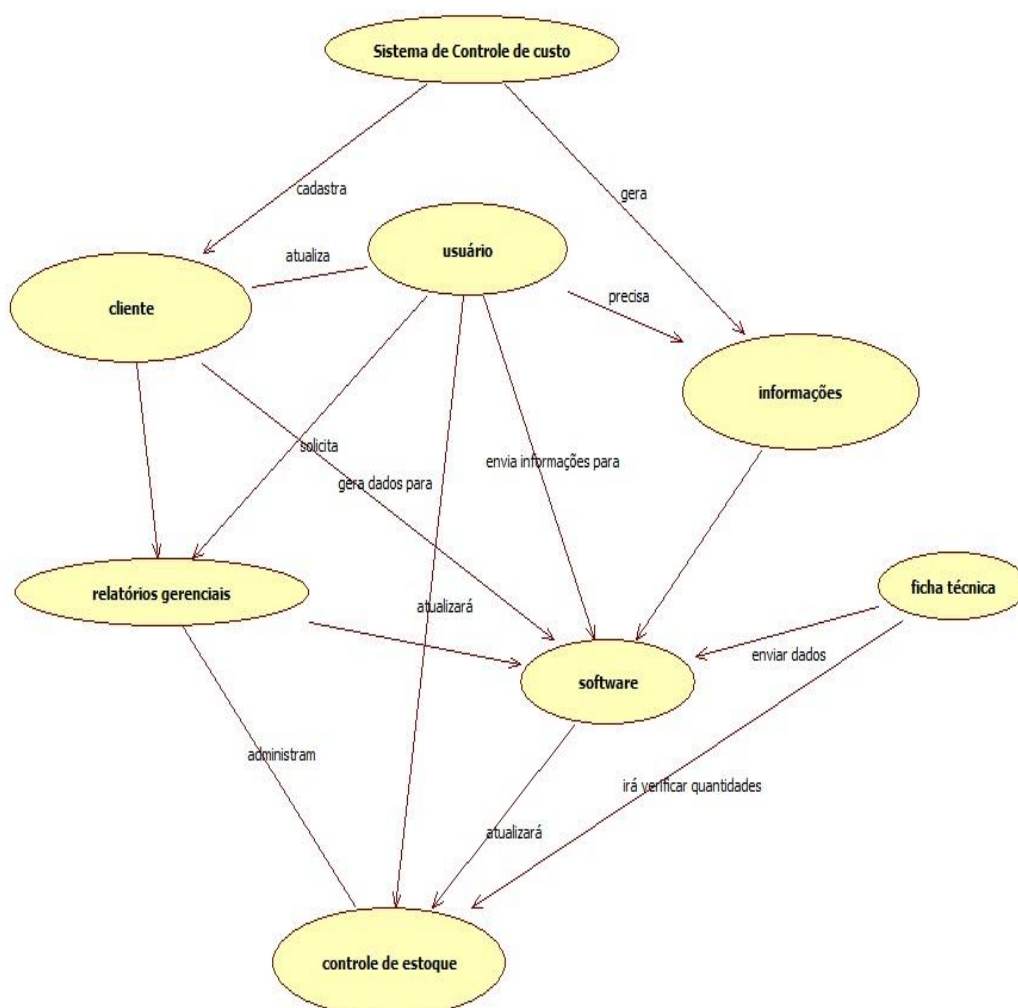


Figura 22 - Mapa conceitual construído por aluno
Fonte: O autor

Todos os alunos que participaram deste trabalho realizaram seus mapas conceituais.

4.7.2 Análise dos resultados da aula 7

Como ferramenta para relacionar os conceitos será utilizada a criação de mapas conceituais, que são diagramas utilizados como um instrumento capaz de evidenciar significados atribuídos aos conceitos e às relações entre conceitos no

contexto de um corpo de conhecimentos, de uma disciplina, de uma matéria de ensino (MOREIRA, 2012, p. 15). Na aula 7 a construção dos mapas conceituais pode comprovar grande influência na aprendizagem da análise de requisitos

Na visão do professor: pode-se que os alunos da turma 2011 tiveram contribuições na aprendizagem e no desenvolvimento do TCC na etapa de análise de requisitos.

Quanto ao tempo: As atividades da aula 7 foram desenvolvidas em um tempo total de cem minutos, sendo duas horas aula de cinquenta minutos cada. Realizadas em um mesmo dia.

4.8 ANÁLISE DOS RESULTADOS DO PONTO DE VISTA DOS ALUNOS

Para averiguação dos resultados na opinião dos alunos, foi solicitado que externassem suas opiniões sobre a contribuição dos mapas conceituais no ensino de análise de requisitos de *software*, através de entrevista coletiva e os resultados foram analisados através de árvore de associações.

4.8.1 Questão da entrevista e análise de resultados

Abordagem feita aos alunos sobre o trabalho de construção de mapa conceitual realizado com os conceitos de requisitos do TCC. *Em sua opinião: o mesmo contribuiu para a sua aprendizagem sobre análise de requisitos de software?*

A partir da entrevista coletiva os dados foram analisados através da construção de árvore de associações para facilitar a observação dos resultados.

Árvore de Associações

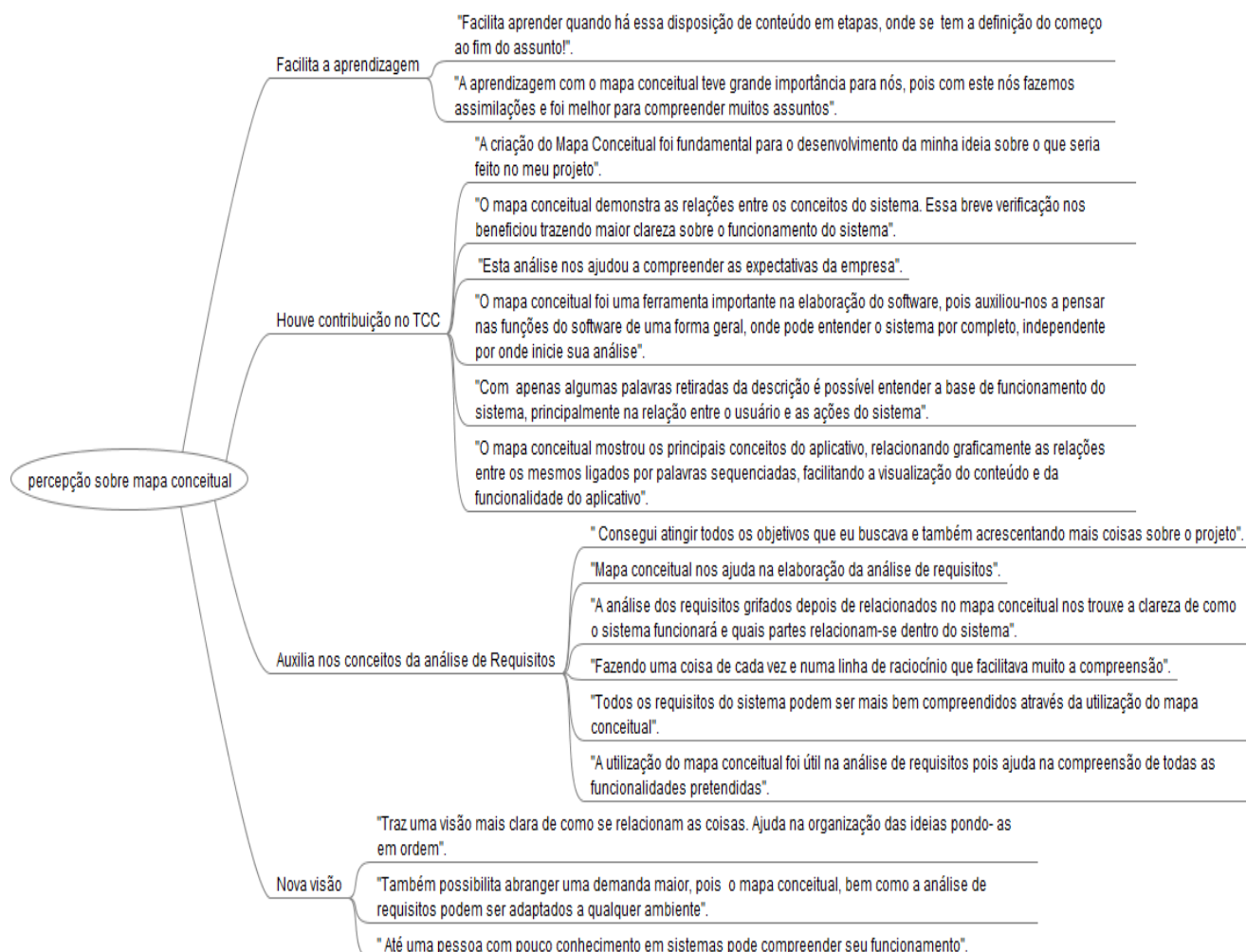


Figura 23 - Árvore de Associações das respostas da entrevista coletiva
Fonte: O autor

Pode-se concluir através da análise da árvore de associações que na opinião dos alunos a construção dos mapas conceituais facilitou a aprendizagem, auxiliou na formação dos conceitos da fase de análise de requisitos de *software* com contribuição no desenvolvimento da análise de requisitos do TCC.

5 CONCLUSÃO

Por tratar-se de trabalho realizado em sala de aula, vale relatar que ao longo das aulas houve observações satisfatórias que foram documentadas e outras que se constituíram apenas em participação dos alunos em aula, porém contribuíram para a realização desta pesquisa.

Uma observação que se considera relevante é o fato de que os alunos mesmo cientes de que estavam participando de um trabalho que, de certa forma era de interesse profissional do professor, durante as aulas manifestaram-se positivamente quanto à participação.

A essência da estratégia proposta é sugerir uma maneira de se trabalhar com aprendizagem significativa e mapas conceituais de modo que os alunos ganhem certa liberdade de ancorar seus subsunçores através dos exemplos familiares. Observou-se que a aprendizagem pode se tornar mais agradável e simples na área de análise de requisitos e acredita-se que poderá também ser viável para outras áreas.

As conclusões desta pesquisa foram consideradas satisfatórias por responder ao objetivo geral e aos objetivos específicos durante as etapas da estratégia.

Quanto aos objetivos específicos:

1º - *“Instigar a criação de subsunçores na análise de requisitos de software”.*

Nesta etapa se observou grande benefício para a aprendizagem dos alunos, com a atividade realizada na aula 2, pois conseguiram mostrar que houve aprendizagem através de liberdade de expressão usando a descontração e a criatividade. Neste momento verificaram-se benefícios em mostrar ao aluno que pode buscar conceitos em sua estrutura cognitiva para que os mesmos sejam ancorados aos novos conceitos.

2º- *“Utilizar como apoio no ensino de análise de requisitos a aprendizagem significativa”.*

A forma de apresentação da aula 1, onde a ideia central ligada a sugestões trazidas pelo professor para que sirvam de organizador prévio, contribuiu para que os alunos pudessem buscar as sugestões que mais lhe fossem familiares. Sugestões simples fizeram com que os alunos se sentissem mais próximos do conteúdo. Nesta etapa pôde-se observar que os organizadores prévios são muito

importantes quando se quer colocar a turma em um nível médio de conhecimento. Todos os alunos conseguiram acompanhar a explicação do professor sem qualquer problema.

Após o trabalho com a criação dos subsunçores realizados na aula 2, os novos conceitos foram facilmente compreendidos pelos alunos na aula 3, onde mostraram que a aprendizagem foi de forma significativa quando auxiliaram o professor na construção do mapa conceitual realizado em conjunto na aula 4.

3º - *“Inserir a criação de mapas conceituais como ferramenta a ser utilizada nas aulas de análise de requisitos”.*

No momento em que os mapas conceituais foram apresentados aos alunos, eles não tiveram dificuldade em entender como iriam fazer a relação entre os conceitos e fizeram observações que nem mesmo o professor havia constatado.

Um aluno observou que depois que o mapa conceitual foi construído viu que os conceitos podem ser analisados de qualquer ângulo, pois partindo de qualquer ponto do mapa, tem-se a visão clara das dependências e relações entre os conceitos.

4º - *“Possibilitar que os alunos construam seus mapas conceituais com base nos conceitos da análise de requisitos da disciplina de TCC”.*

Uma observação feita por um aluno durante a criação dos mapas conceituais chamou a atenção de que é possível, a partir do mapa construído, corrigir a descrição dos requisitos em detalhes que não se percebia antes, ou seja, os mapas contribuíram também na aprendizagem da elaboração das descrições dos requisitos de software.

Pode-se concluir que os mapas conceituais contribuíram para o ensino dos conceitos de análise de requisitos e, também, auxiliaram os alunos a construírem o seu próprio conhecimento, quando utilizaram a ferramenta para fazer a relação entre os conceitos da análise de requisitos do TCC.

5º - *“Investigar através de entrevista a contribuição da estratégia proposta”.*

Na entrevista coletiva as respostas dos alunos foram favoráveis à utilização da estratégia com ênfase na construção dos mapas conceituais e isto pode ser conferido através da árvore de associações construída para análise.

6º - *“Elaborar um material didático com etapas a serem seguidas utilizando a metodologia proposta neste trabalho”.*

O material didático extraído desta pesquisa foi satisfatório para o ensino da análise de requisitos e o mesmo sugere que os professores façam adaptações e utilizem a ideia da metodologia para outros assuntos do Curso Técnico em Informática.

Quanto ao objetivo geral:

- *Avaliar a contribuição da aprendizagem significativa e mapas conceituais no ensino de análise de requisitos de software.*

Observou-se também que quando se faz uma seleção dos conceitos mais importantes e os mesmos são trabalhados de forma significativa, os alunos realmente compreendem o conteúdo e isto ocorre em um tempo relativamente pequeno.

Com a metodologia utilizada neste trabalho, levando em consideração a amostra em questão, pôde-se concluir através da observação do professor, que a aprendizagem significativa e a construção dos mapas conceituais contribuíram para o ensino dos conceitos de análise de requisitos, bem como no apoio à construção do conhecimento no momento da relação entre os conceitos ligados aos requisitos do TCC.

Quanto ao problema: *Como a utilização de aprendizagem significativa e mapas conceituais podem contribuir para o ensino de análise de requisitos de software?*

Os mapas conceituais podem ser utilizados como ferramenta didática de auxílio ao professor na preparação de aula utilizando como mecanismo gráfico para explanar e explorar a relação dos conceitos no ensino da teoria da análise de requisitos e, também, como apoio na aprendizagem dos alunos quando entram na fase prática e necessitam estabelecer relação dos conceitos extraídos da análise de requisitos do TCC.

Pode-se concluir que o resultado da pesquisa foi satisfatório em relação ao problema proposto, podendo-se afirmar que os mapas conceituais podem ser utilizados como apoio ao ensino de análise de requisitos de software.

5.1 DIFICULDADES

Neste trabalho foram encontradas algumas dificuldades como:

- Todos os alunos presentes em todas as aulas é uma situação difícil, porém necessária para o acompanhamento das etapas.

- A elaboração e preparação das aulas são mais detalhadas, tomando um tempo maior do professor o qual terá que dispor de um tempo maior de preparação do que aplicação das aulas.

- O professor deve acompanhar o aluno de forma individual para perceber o desenvolvimento das atividades propostas.

5.2 SUGESTÕES PARA TRABALHOS FUTUROS

Por se tratar de uma ferramenta que irá contribuir para o ensino, dando apoio didático ao professor e auxiliando na construção do conhecimento dos alunos, sugere-se que a metodologia seja aplicada em outros assuntos como as demais etapas da engenharia de software, outras disciplinas do Curso Técnico em Informática, bem como para outros cursos de outras áreas.

Para outras áreas sugere-se uma adequação das aulas seguindo a sequência e sugestões a seguir:

- Aula 1 – **Aula expositiva e organizadores prévios:** O professor elabora o conteúdo da sua aula selecionando os principais conceitos que serão estudados. Para cada um dos principais conceitos irá fazer uma representação gráfica no formato aranha onde o conceito é centro que ganhará braços, os quais irão trazer exemplos e sugestões relacionados ao conceito principal.

- Aula 2 – **Criação de subsunçores:** Nesta aula o professor irá colocar os principais conceitos e solicitar que os alunos se preocupem apenas com as palavras colocadas na lista e assim descrevam exemplos ou definições que surgem, sem se preocupar em relacionar com o conteúdo da aula.

- Aula 3 – **Análise de requisitos com enfoque aos conceitos trabalhados:** Nesta o professor irá ministrar sua aula expositiva que contemplará

todos os conceitos trabalhados na aula 1 e aula 2, e poderá perceber a evolução dos alunos.

- Aula 4 – **Apresentando mapas conceituais:** A aula 4 é necessária para a que a ferramenta de mapas conceituais seja apresentada para os alunos, pois todos deverão ter conhecimento da mesma. Nesta aula o professor deve construir mapas conceituais em conjunto com os alunos para que eles possam opinar e entender.

- Aula 5 – **Análise de requisitos na prática:** Esta aula para outras áreas poderá ser trabalhada com material complementar sugerido pelo professor ou vindos de criação ou pesquisa dos alunos.

- Aula 6 – **Buscando principais conceitos na análise de requisitos.** A partir do material da aula 5 o alunos irá selecionar e retirar do material trabalhado os principais conceitos e irá fazer o trabalho semelhante ao da aula 2: sugerir exemplos ou definições que surgem, sem se preocupar em relacionar com o conteúdo da aula.

- Aula 7 – **Alunos constroem mapas conceituais.** Os alunos poderão construir seus mapas conceituais de forma individual e particular utilizando com base os conceitos trabalhados na aula 6.

Com esta sequência simples de sugestões acredita-se que outras áreas ou assuntos poderão trabalhar com a estratégia proposta.

REFERÊNCIAS

- AUSUBEL, D.; NOVAK, J.; HANESIAN, H. **Psicologia educacional**. 2ª edição, Editora Interamericana, Rio de Janeiro, 1980.
- ENGHOLM, H. JR., **Engenharia de Software na Prática**, 1ª edição, Editora Novatec, São Paulo, 2010.
- GIL, A.C., **Como elaborar projetos de pesquisa**. 4ª edição, Editora Atlas, São Paulo, 2002.
- IEEE, Computer Society. **SWEBOK - Guide to the Software Engineering Body of Knowledge**. California, IEEE, 2004.
- MACHADO, F.N., **Análise e Gestão de Requisitos de Software, onde nascem os sistemas**. 1ª edição, Editora Érica, São Paulo, 2011.
- MARCONI, A. M.; LAKATOS, E. M., **Fundamentos de metodologia científica**, 7ª edição, Editora Atlas, São Paulo, 2010.
- MOREIRA, M. A., **Aprendizagem Significativa: Fórum permanente de professores**. Editora UNB, Brasília, São Paulo, 1999.
- MOREIRA, M. A., MASINI E.F.S., **Aprendizagem Significativa – A Teoria de David Ausubel**. 4ª edição. Editora Centauro, São Paulo, 2011.
- MOREIRA, M. A., **Mapas conceituais e aprendizagem significativa**. 1ª edição, Editora Centauro, São Paulo, 2010.
- MOREIRA, H., CALEFFE, L. G. **Metodologia científica para o professor pesquisador** – 2. ed. – Rio de Janeiro: Lamparina, 2008
- NOVAK, J.D., GOWIN, D. B. **Aprender a aprender**. Plátano Edições Técnicas, Lisboa, 1999.
- PRESSMAN, R. S. **Engenharia de Software**. 6ª edição. Editora Bookman, São Paulo, 2010.
- RICHARDSON, R. J., *et all.* **Pesquisa Social: métodos e técnicas**. 3ª edição, Editora Atlas, São Paulo, 1999.
- SANTOS, J. C. F., **Aprendizagem Significativa – modalidade de aprendizagem e o papel do professor**. 3ª ed., Porto Alegre: Editora Mediação. 2009.
- SEVERINO, A. J., **Metodologia do trabalho científico**. 23ª edição, Editora Cortez, São Paulo, 2012.

SILVA, E.; MENEZES, E. M., **Metodologia da Pesquisa e Elaboração de Dissertação**. 3ª edição, Florianópolis: Laboratório de Ensino à Distância da UFSC, 2001.

SOMMERVILLE, I. **Engenharia de Software**. 9ª edição, Editora Pearson Education, São Paulo, 2011.

SPINK, M. J., **Práticas discursivas e produção de sentidos no cotidiano aproximações teóricas e metodológicas**. Editora Cortez, São Paulo, 1999.

TAVARES, R., **Aprendizagem significativa, codificação dual e objetos de aprendizagem**. Revista Brasileira de Informática na Educação, v.18, n.2, 2010.

TAVARES. R. **Construindo mapas conceituais**. Ciências & Cognição; Ano 04, Vol 12, 2007. Disponível em www.cienciasecognicao.org

APÊNDICE A - Respostas da Aula 2

01 – Requisitos - Definição: Para Machado (2011, p. 24), “os **requisitos** são objetivos ou restrições estabelecidas por clientes e usuários do sistema que definem as diversas propriedades do sistema”.

Respostas:

1. Para o computador ligar é necessário que tenha energia elétrica.
2. Para jogar truco é necessário conhecer as regras.
3. Um sistema para playground precisa cadastrar crianças, cadastrar responsáveis e emitir pulseiras.
4. Em uma nota fiscal para preencher o destinatário é preciso ter CPF ou CNPJ.
5. É necessário possuir um cadastro de clientes.
6. É necessário informar o CPF do cliente.
7. Informações absorvidas de um cliente, todas as operações que um sistema deverá executar.
8. Para ser eleitor é necessário ter no mínimo 16 anos.
9. O sistema de venda precisa ter cadastro.

02 – Requisitos Funcionais: Segundo Sommerville (2011, p. 59), “são declarações de serviços que o sistema deve fornecer, de como o sistema deve reagir a entradas específicas e de como o sistema deve se comportar em determinadas situações”.

Respostas:

1. Para o carro funcionar é necessário ter combustível.
2. Possuir um baralho com 4 cartas sendo 4 cartas de **Ás** ao 7 e quatro de cada uma.
3. O cadastro de cliente (criança) será vinculado com o cadastro do responsável.
4. O sistema deve permitir a inclusão, exclusão e alteração dos clientes.
5. Precisa inserir RG e CPF.
6. Cadastro de cliente.
7. Para cadastrar o tecido que será utilizado na fabricação de uma camiseta é necessário calcular as medidas.
8. Para fazer título de eleitor tem que ter RG.
9. O cliente precisa ter **o programa Java** instalado.

03 – Requisito não funcional: Segundo Sommerville (2011, p. 59), “são restrições aos serviços ou funções oferecidas pelo sistema. Incluem restrições no processo de desenvolvimento e restrições impostas pelas normas”.

Respostas:

1. Tipo de monitor.
2. Possuir um marcador caracterizado.
3. O sistema do playground não terá interação com o sistema da loja.
4. O sistema não fará a baixa automática dos cheques pendentes.
5. Declarar renda.
6. Windows XP.
7. A cor será um parâmetro para distinguir um tecido de outro.
8. Possuir um endereço de permanência.
9. O sistema deve conter cor azul.

04 – Stakeholder: De acordo com Machado (2011, p. 30), “é qualquer pessoa materialmente afetada pelo resultado do projeto: clientes, usuários diretos e indiretos, investidores, acionistas, fornecedores, supervisores, gerentes, compradores”.

Respostas:

1. Gerente.
2. São as pessoas envolvidas direta e indiretamente ao sistema. Ex: administrador, colaboradores e clientes.
3. Analista, programador, cliente e funcionários.
4. Quem usa o sistema, funcionário ou gerente.
5. Proprietário e funcionários que utilizam o sistema.
6. Funcionários.
7. Clientes, funcionários, fornecedores.
8. Mesário, presidente, candidato, eleitor.
9. Cliente, chefe, programador.

05 – Ambiguidade: O Significado da palavra é a possibilidade de uma mensagem ter dois sentidos. De acordo com Sommerville (2011, p. 471), “quando você coleta dados quantitativos sobre o *software* e processos de *software*, precisa analisar este dados para entender seu significado. É fácil interpretar dados erroneamente e fazer inferências **incorretas**”.

Respostas:

1. Garantia do carro são dois ou três anos.
2. O projeto ficará pronto em dois ou três meses.
3. A gata da minha irmã dormiu na minha cama.
4. Proibida a entrada com carros, sem camisa, menores de 18 anos.
5. Proibida a entrada com bebidas e animais. Se eu tiver com bebida e sem animal, posso entrar?
6. (sem resposta).
7. Quando duas informações são idênticas, confusas.
8. Reconsulta em dois ou três dias.
- 9 - Onde está a vaca da sua avó?

06 – Questionário: Segundo Machado (2011, p. 148), “devem ser desenvolvidos de forma a minimizar o tempo gasto de sua resposta”.

Respostas:

1. Respostas do trabalho para a professora.
2. Ficha de Inscrição.
3. Como é o laboratório do IFPR? () Bom () Ruim () Não utilizo.
4. Prova ou as questões que envolvem o sistema: Quais as dificuldades na clínica?
Quanto PCs pretende ter na loja?
5. Quantas pessoas usam simultaneamente o sistema? Qual valor que deseja gastar na implementação?
6. Documentos de requisitos.
7. Questionário socioeconômico.
8. Múltipla escolha, sim ou não, livre resposta.
9. Perguntas feitas pelo IBGE.

07 – Entrevista: Para Machado (2011, p. 142), “é uma das técnicas tradicionais mais simples de utilizar e que produz bons resultados na fase inicial de obtenção dos dados. Convém que o administrador dê margem ao entrevistado para expor suas ideias”.

Respostas:

1. Jô Soares.
2. O delegado interrogando o réu.
3. Como é o laboratório do IPFR? Descreva a resposta.
4. Dia e hora para contato.
5. Marcar data e hora com o cliente para tirar dúvidas.
6. (sem resposta).
7. Entrevista de emprego.
8. Direta, por telefone, particular, coletiva, levante a mão.
9. Conversa para conseguir um emprego.

08 – Documento de Requisitos: Segundo Machado (2011, p.104), “os requisitos aprovados devem ser documentados em um nível apropriado de detalhamento. O documento de requisitos é formal e obrigatório, utilizado para comunicar os requisitos aos clientes, engenheiros e gerentes”.

Respostas:

1. IBGE – Pesquisa demográfica.
2. ABNT.
3. O sistema para playground conterá requisitos funcionais e não funcionais.
4. Diagrama de caso de uso, lista de tabelas.
5. Diagrama de classe, diagrama de objetos e dicionário de dados.
6. Documento de análise de requisitos.
7. Relato, documento de como foram feitos os requisitos.
8. Bula de remédio.
9. Manual de instruções.

09 – Análise: Na fase de Análise de requisitos “ocorre a negociação e análise de viabilidade de execução do que fora solicitado.” (MACHADO, 2011, p. 104).

Respostas:

1. Exame de sangue.
2. Exame de sangue.
3. Estou em dúvida em comprar um sapato ou uma bolsa. Estou analisando qual será mais útil.
4. Análise de ambiguidade de erros.
5. Requisitos funcionais e não funcionais.
6. (sem resposta).
7. Exames médicos.
8. Tirar sangue, medir temperatura.
9. Vendedor vai ao concorrente para verificar seus preços.

10 – Projeto: Segundo Sommerville (2011, p. 25), “um projeto de *software* é uma descrição da estrutura do *software* a ser implementado, dos modelos e estruturas de dados usados pelo sistema, das interfaces entre os componentes do sistema e às vezes, dos algoritmos usados”.

Respostas:

1. Construção de uma casa.
2. Roteiro de férias.
3. Quando terminar o curso, tenho projeto de sair da cidade.
4. Projeto de conclusão de curso.
5. Plano para melhorar o atendimento ao cliente através de um cadastro.
6. Diagrama de caso de uso.
7. Planta de arquitetura.
8. Arquitetônico, ter um filho.
9. Projeto de uma casa.

APÊNDICE B - Trabalhos realizados pelos alunos nas aulas 5, 6 e 7

Trabalho 1

a) **Resumo da Análise de Requisitos:** Software para controle de produção de uniformes.

O desenvolvimento do sistema terá como principal objetivo o controle de produção de uma empresa de confecções de uniformes escolares, industriais. A empresa necessita de um *software* que faça o controle de **cadastro de clientes**: onde deve conter o nome da empresa (Por se tratar apenas de pessoa jurídica); o CNPJ, endereço, telefone, cidade das empresas clientes. Depois de ser feito o cadastro do cliente o software de controle de produção **deverá emitir uma ordem de serviço**, onde **deverá constar** o nome, CNPJ, endereço, telefone, cidade do cliente, tipo do produto, quantidade do produto, data da compra, data da entrega.

O programa deverá conter um **controle de estoque**: onde ele **irá informar** ao vendedor se há ou não **matéria-prima** disponível para confecção dos uniformes, **irá proceder** da seguinte **forma**: data de entrada e saída da **matéria-prima**, cor do tecido, quantidade do tecido, **e, também, acusará** falta de **matéria-prima**. O mesmo controle de estoque fará o **cálculo do tecido** por peça solicitada, onde vai conter a quantidade do produto, tamanho das peças, qual peça de uniforme será confeccionada, estabelecendo para o cliente uma maior precisão na hora da venda. E para a finalização do software ele vai gerar um **controle de prazo de entregas**, gerenciando por data, quantidade, cores, tamanhos e clientes.

b) Conceitos das palavras grifadas:

Palavras	Conceitos atribuídos
Cadastro de clientes	Onde executará uma função para cadastrar todos os dados do cliente. Nesse caso o sistema deverá manter um banco de dados com o CNPJ, nome do cliente, endereço, telefone, cidade.
Emitir uma ordem de serviço	Exibirá uma relação com todos os dados do cliente e todas as informações do produto, como: tipo do produto, quantidade, data da compra, data da entrega.
Controle de estoque	Manterá uma relação entre a ordem de serviço e o cálculo de tecido. Vai exibir a real situação do estoque da empresa com relação à matéria-prima existente no ato da venda; vai mostrar a quantidade de matéria-prima em estoque, também vai haver o cadastro de entrada da matéria-prima por metro linear.
Cálculo do tecido	Essa função estabelecerá a quantidade de tecido que o pedido irá consumir do estoque. Informará precisamente quantas peças podem ser feitas com a determinada quantia de matéria-prima existente.
Controle de prazo de entrega	Vai controlar diariamente a rotina de produção, onde vai mostrar quais pedidos têm que ser entregues no dia ou na semana.

c) Mapa conceitual:

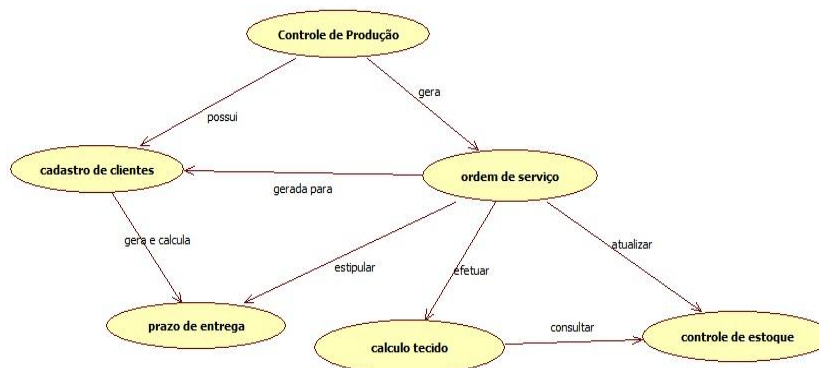


Figura 24 - Mapa conceitual - Trabalho 1
Fonte: autor

Trabalho 2

a) **Resumo da Análise de Requisitos:** Sistema gerenciador de produção de Indústria de Erva Mate.

O sistema terá uma tela para cadastro de **matéria-prima**, permitindo cadastrar, alterar e excluir (caso não haja ligação), **conseguindo** controlar o **estoque**, **as entradas**, **os** valores de entrada, que juntamente ao cadastro de **mão de obra conseguirão obter** valor de lucro real, **valor de custo**, estoque de **matéria-prima** e estoque de **produto acabado**. O cadastro de mão de obra terá as informações referentes a **gastos** para a **produção**, **como:** funcionários, energia elétrica, entre outros.

A parte de cadastro de grupos/subgrupos servirá para identificar os tipos distintos de produtos, **tributações**, conseguindo ter um controle ainda melhor sobre seus produtos.

O cadastro de **fornecedor** permitirá ao usuário ter um índice de melhores vendedores, para poder criar uma **carta de fidelidade**, com melhores preços, **maiores e mais frequentes compras**.

O processo da industrialização é dividido em 3 partes:

Primeiro Processo (1ª secagem) – informar mão de obra (madeira, funcionário, energia elétrica): o funcionário 1 leva a **matéria-prima** até a máquina de **secagem**. **Ao** longo do processo o funcionário 2 adiciona madeiras para **secagem**. **Após** o primeiro processo, teremos mais dois processos. Resultado: gera secagem prévia.

Segundo Processo (2ª secagem) – informar mão de obra (madeira, funcionário, energia elétrica): o funcionário 2 leva ao longo do processo madeiras para secagem. **Após** o primeiro processo teremos mais 1 processo. Resultado: gera secagem total.

Terceiro Processo (Separação) – informar mão de obra (funcionário, energia elétrica): a máquina faz a separação para quatro tipos de produtos, sendo reutilizado apenas 2, sendo eles: **SENE PADRÃO A e PALITO PN2**, e as que não serão utilizadas serão as **PALITO PN1 e SENE CAREJÓ**.

Após os três processos, o funcionário 3 levará as ervas selecionadas com padrão A para o **empacotamento**, enquanto o funcionário 4 leva as ervas não utilizadas.

b) **Conceitos das palavras grifadas:**

Palavras	Conceitos atribuídos
Matéria-prima	Material usado para a confecção do produto final.
Estoque	Local onde ficarão armazenadas as informações sobre quantidades contidas para pronta entrega.
Entradas	Ato de receber algo do seu fornecedor.
Mão de obra	Incluem-se em mão de obra tudo o que for gasto para produção (funcionários, energia, madeira, entre outros).
Valor de custo	Valor pago ao fornecedor somando o valor da mão de obra.
Produto acabado	Produto final, após a saída da industrialização.
Gastos	Valor pago ao fornecedor somando o valor da mão de obra.
Produção	Todo o processo de industrialização.
Tributações	Preenchido pelo usuário a parte tributária da empresa.
Fornecedor	Pessoa física ou jurídica que dá entrada de matéria-prima.
Carta de fidelidade	Parceria adotada junto ao fornecedor, conseguindo visualizar melhores parceiros de negócio.
Secagem	Processo de secagem da matéria-prima (erva).
Sene Padrão A	Erva com maior qualidade mais utilizada para produção.
Palito PN2	Erva com menor qualidade, mas utilizada em pequena proporção.
Palito PN1	É uma parte não utilizável para produção da erva, ou seja, perda.
Sene Carejó	É uma parte não utilizável para produção da erva, ou seja, perda.

c) Mapa conceitual:

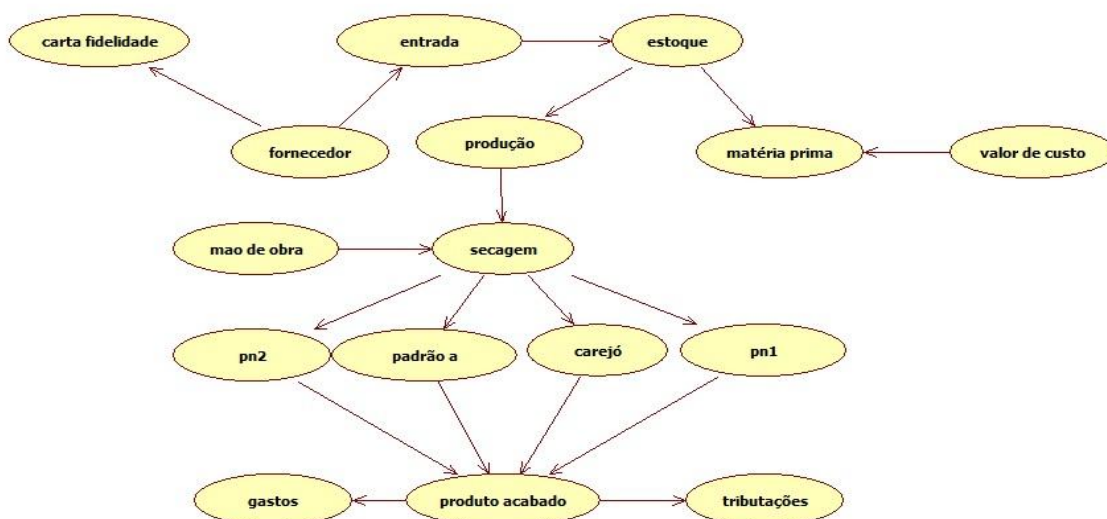


Figura 25 - Mapa conceitual - Trabalho 2

Fonte: autor

Trabalho 3

a) Resumo da Análise de Requisitos: Sistema de Controle e Gerenciamento de Farmácias Veterinárias e Pet Shops.

A empresa Farmavet fornece os serviços: vacinas, consultas, exames, banho e tosa.

O sistema será operado por um **usuário** cadastrado no sistema, o qual poderá fazer o cadastro de clientes, animais, funcionários, serviços, compromissos na agenda, gerar relatórios, etc.

O cadastro de **clientes abrangerá** os campos nome, CPF, RG, telefone, email, data do cadastro e endereço.

Os **funcionários** também serão cadastrados e **abrangerão** os seguintes campos: nome, CPF, RG, data de admissão, cargo, CRMV (p/ veterinários), endereço.

Os **animais estarão** obrigatoriamente ligados a um cliente e **serão cadastrados através dos campos:** nome, sexo, raça, data de nascimento, código do cliente.

Para o campo **raça** no cadastro de animais haverá uma tabela na qual serão cadastrados os tipos de raças mais comuns.

Haverá um **cadastro de vacinas** contendo código da vacina, nome das vacinas, observações. Serão cadastradas em outra tabela as **vacinas realizadas** contendo código da vacina, data da aplicação, código do animal, cod. Funcionário.

Em outra tabela serão cadastradas as **vacinas futuras** contendo cod. Vacina, data da aplicação cod. Animal.

Cada **serviço** realizado pelo estabelecimento será cadastrado em uma lista de serviços que conterá a descrição do serviço, **além de** dados como valor e tempo de execução. Esta lista será requisitada pela ordem de serviço a ser aberta quando houver a solicitação de um cliente. Na **ordem de serviços** será relacionado cliente, animal, funcionário e produtos utilizados para execução do mesmo.

O sistema possuirá uma **agenda** onde serão marcados os compromissos diários de cada funcionário. Para agendar um serviço é necessária a ligação das tabelas: serviço, cliente, animal, funcionário, além de conter também data e hora

b) Conceitos das palavras grifadas:

Palavras	Conceitos atribuídos
Clientes	Quem compra na loja.
Funcionários	Quem trabalha na loja.
Animais	A eles que se destinam os serviços e produtos da loja.
Raça	Cadastro das raças. Ex: cachorro - raça: boxer.
Cadastro vacinas	Medicamento a ser aplicado.
Vacinas realizadas	Histórico de vacinas realizadas em todos os animais cadastrados no sistema.
Vacinas futuras	Cadastro de todas as vacinas que os animais deverão tomar e periodicidade.
Vacinas realizadas	Cadastro de todas as vacinas que o animal já tomou.
Serviços	Atividades realizadas na loja em benefício dos animais e clientes
Ordem de serviço	Cadastro de um ou vários serviços ligados a um animal de um determinado cliente, que serão executados por um ou vários funcionários.
Agenda	Onde são armazenados compromissos e tarefas da empresa e dos funcionários.

c) Mapa conceitual:

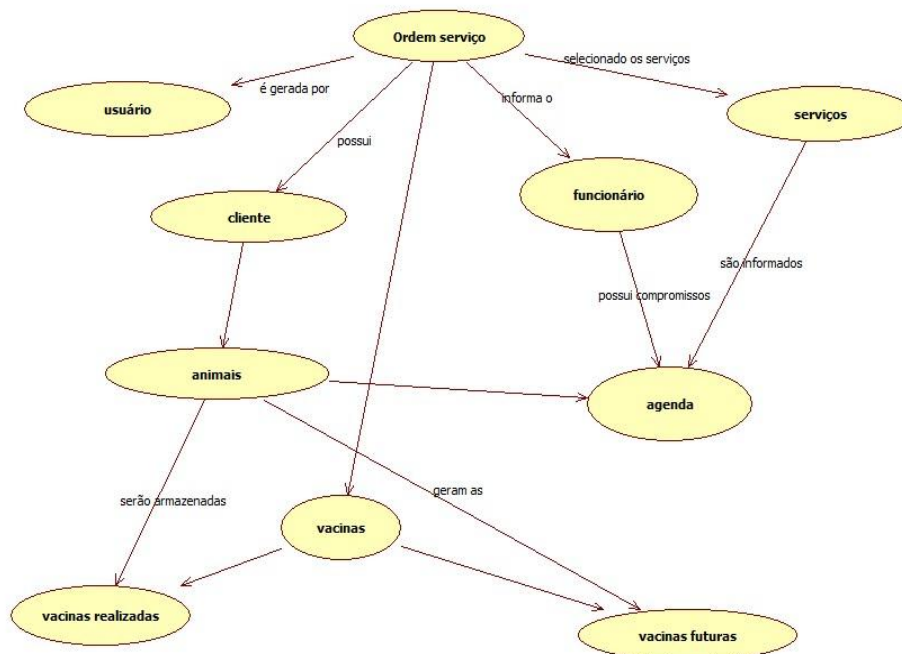


Figura 26 - Mapa conceitual - Trabalho 3
Fonte: autor

Trabalho 4

a) **Resumo da Análise de Requisitos:** Controle de custo da produção para malharia.

O *software* terá telas que permitam cadastrar, alterar, excluir o lançamento de **matéria-prima** que suprirá a **produção** da empresa, também será possível cadastrar os produtos que serão produzidos (**produto acabado**). A função do sistema será levantar os **custos** da produção, controlando o custo de compra da **matéria-prima**, **desperdícios** e os custos fixos para a produção, também irá realizar o controle de estoque de produtos acabados que poderão estar no sistema ou serem exportados para outro *software* já existente.

Receberá informações da entrada de materiais em seu estoque e, **para** esse controle, será aplicado o sistema de controle de estoque **PEPS** (Primeiro que Entra Primeiro que Sai), emitirá relatórios de suas **rotinas** e exportará arquivo para integração com outro *software* de controle comercial já existente na empresa.

Deverá emitir relatórios para **gerenciamento** de todos os produtos envolvidos na linha de produção. Deverá existir um **controle dos pedidos** para que a linha de produção cumpra os prazos e as quantidades solicitadas.

b) Conceitos das palavras grifadas:

Palavras	Conceitos atribuídos
Matéria-Prima	Produto ainda não industrializado, pronto para entrar na linha de produção.
Produção	Fase em que na empresa começa a ser feito o produto final estabelecido.
Produto Acabado	Peça já pronta, após passar por toda a fase de produção, é produto acabado e pronto para ser comercializado.
Desperdícios	É tudo aquilo que a empresa acaba descartando, ou seja, os restos as sobras.
PEPS	“Primeiro que entra primeiro que sai” método no qual há uma organização para o controle de produtos estocados.
Rotinas	Processo de repetição dentro de uma linha de produção.
Gerenciamento	Método para atingir objetivos, otimizando os resultados.
Controle dos pedidos	Todo pedido deve se ter um controle para que possa ser atendido de acordo com a necessidade do cliente e da empresa.

c) Mapa conceitual

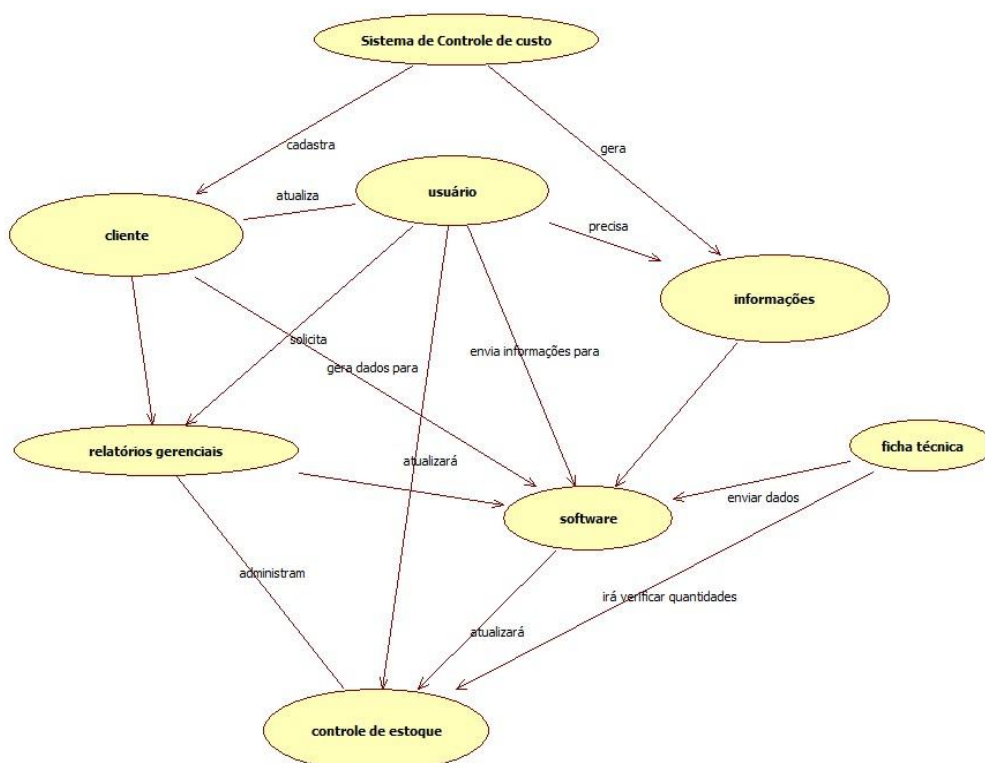


Figura 27 - Mapa conceitual - Trabalho 4
Fonte: Autor

Trabalho 5

a) **Resumo da Análise de Requisitos:** Sistema gerenciador de **clínicas** odontológicas.

O sistema gerenciador de **clínicas** odontológicas possuirá foco maior na parte financeira e na parte de atendimentos.

Quando o **cliente** solicitar **atendimento** o mesmo poderá realizar o **agendamento** por telefone ou pessoalmente, sendo o dia e horários disponíveis verificados no cadastro de **Agendamentos**. **Logo após verificada** a disponibilidade será incluído um novo registro no mesmo cadastro sendo **informada** a data, hora, cliente e o tipo de **atendimento** (cadastro de **tipos de atendimento**, sendo necessário apenas o preenchimento do campo descrição).

Ao chegar **à clínica** o cliente solicitará atendimento, **depois de** verificado o agendamento do mesmo, será lançada uma **consulta**, onde o **cirurgião verificará que** uma nova consulta foi gerada, e quando solicitado o cliente será **atendido**. **Após a consulta**, o cliente **efetuará** o **pagamento**. **Caso seja à vista** será lançado automaticamente para o **caixa**; caso contrário, **a prazo**, será lançado para os **recebimentos**, sendo **informada** a data, o primeiro vencimento, a quantidade de parcelas e o valor. **As parcelas serão** geradas automaticamente, sendo possível o pagamento em **cheque** (a prazo e **à vista**).

Os **cheques** serão lançados separadamente em um cadastro próprio, informando banco, **agência, número da conta**, data de emissão, valor, vencimento, correntista e o cliente. A baixa dos cheques será efetuada manualmente, lançando no caixa automaticamente após a baixa.

Por fim, as despesas da clínica serão lançadas no cadastro de despesas, informando o tipo e o valor, onde serão totalizados de um modo geral. (Ex. Material)

b) **Conceitos das palavras grifadas:**

Palavras	Conceitos atribuídos
Clientes	Toda pessoa capaz de utilizar os recursos disponibilizados pela clínica.
Atendimentos	Sendo responsável desde a entrada até a saída do cliente da clínica.
Tipos de Atendimentos	Tipo de serviço que será realizado no paciente.
Agendamentos	Onde o cliente marcará um horário disponível para utilizar os recursos da clínica.
Consulta	Parte da clínica onde o cliente será acompanhado, juntamente com a secretária ou cirurgião, para realizar o serviço solicitado.
Cirurgião	Profissional de saúde capacitado na área odontológica.
Pagamentos	Parte financeira da clínica.
À vista	Todo dinheiro que entra no caixa na hora do atendimento, ou seja, não tem vencimento.
A Prazo	O pagamento será realizado somente até a data solicitada para o vencimento, logo depois entra no caixa.
Recebimentos	Responsável por contabilizar o que entra e o que sai do caixa da clínica.
Cheque	Ordem de pagamento à vista, não sendo em dinheiro, mas com o mesmo valor.
Despesas	Tudo que será gasto na clínica, para usos internos.
Tipos de Despesa	Tipo de despesa a ser lançada. Ex. Água

c) Mapa conceitual:

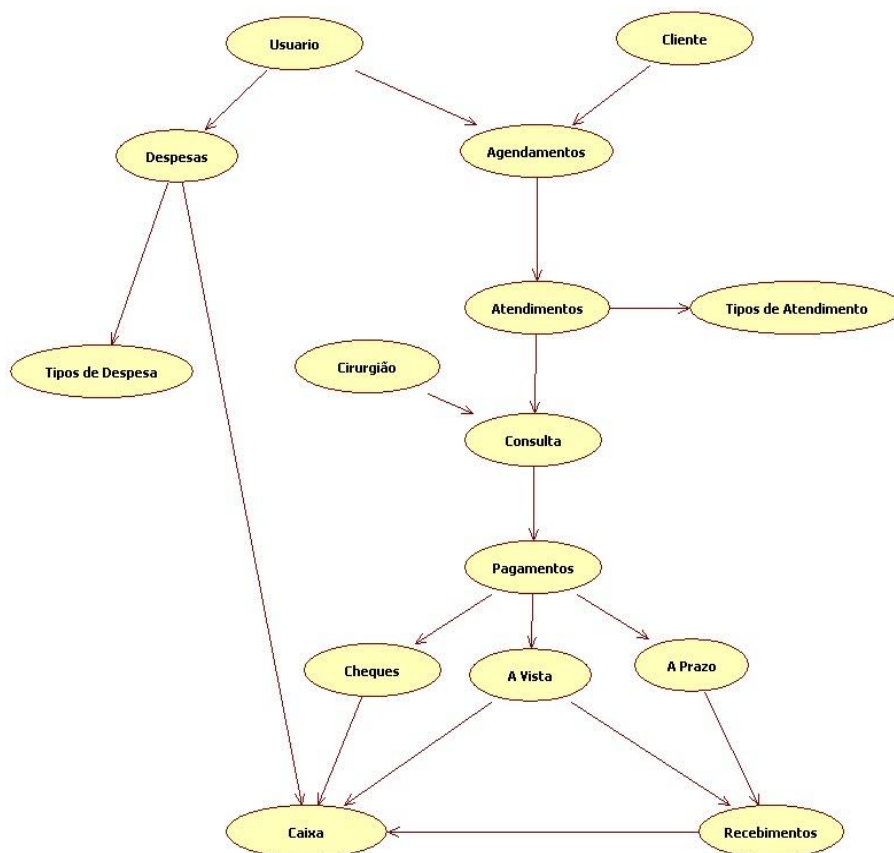


Figura 28 - Mapa conceitual - Trabalho 5
Fonte: Autor

Trabalho 6

a) **Resumo da Análise de Requisitos:** Sistema Controle de Parque Aquático.

O sistema deverá fazer o **cadastro** de todos os **clientes** que adentrarem no recanto. Sendo assim, este cadastro deverá conter o nome completo do cliente, assim como, fornecer um código para este cliente de forma automática. Serão cadastrados: o CPF e RG, a data de nascimento, estado civil, endereço, bairro, telefone1, celular e CEP de forma obrigatória; telefone 2 e e-mail de forma opcional, a cidade e estados que também devem fazer parte do cadastro serão carregados pelo banco de dados, o qual já estará com todos os estados e cidades brasileiras já cadastrados. Estes dados podem ser visualizados, incluídos, alterados ou excluídos sempre que o **usuário máster** do sistema sentir necessidade de modificá-lo. Poderá ser feita a inclusão de **usuários do sistema**, ou seja, cadastro dos funcionários que utilizarão o sistema, sendo assim, cada usuário cadastrado deverá ter um login que pode ser escolhido pela pessoa que vai **utilizar o sistema**. Assim como o login, haverá uma senha que deverá se comportar da mesma forma que o login, podendo assim ser escolhida. Cada usuário do sistema receberá um código aleatório que identifica esse usuário dentro do sistema. Além de fazer todo esse cadastramento de usuários o sistema deve permitir a alteração e exclusão destes usuários através de um usuário máster.

Dentro do parque existem diversas **atrações** sendo que cada uma delas tem um preço diferenciado, e ainda, uma das atrações deve ser paga por vezes utilizada, por isso será criado um cadastro de atrações onde o maior usuário do sistema poderá cadastrar atrações sempre que necessário. **Desta forma as atrações devem conter cada uma delas, além do nome, um código de atração fornecido aleatoriamente pelo sistema em forma ordinal, o valor de cada uma delas,**

no qual o usuário máster poderá fazer inclusão, alteração ou exclusão das atrações sempre que houver necessidade.

No recanto é utilizada uma pulseira para identificação dos clientes. Nela deverá conter um código cadastrado pelo usuário do sistema, onde constará o nome do cliente, para isso o cliente já deverá ter feito o cadastro. Esta pulseira somente serve para controle diário, e por último, a pulseira deve receber uma cor, onde cada dia será uma cor diferenciada. Todas estas opções poderão receber inclusão, alteração ou exclusão, sempre que necessárias.

O sistema deve controlar o **consumo diário dos clientes**, caso desejem pagar somente no fim do dia. Para isso será criada uma tabela de consumo, onde o garçom insere o código da lanchonete que estará cadastrada como uma atração, podendo também ser uma das atrações de entretenimento. O funcionário deverá descrever o produto e valor, além de marcar a situação do consumo (se foi pago ou se ficou em aberto). As atrações, sem exceção, deverão trazer a descrição e o valor do banco de dados. O sistema deverá fazer emissão de relatórios diários contendo nome do cliente, valor gasto e código da pulseira utilizada. O sistema deverá gerar relatório com todo o histórico de um determinado cliente.

b) Conceitos das palavras grifadas:

Palavras	Conceitos atribuídos
Cadastro	É a inserção dos dados de uma pessoa (cliente) dentro do sistema. Serve para controlar os dados dos frequentadores do recanto. O cadastro é feito apenas uma vez para cada pessoa, e depois ele pode ser consultado, alterado ou excluído pelo usuário do sistema.
Clientes	São todas as pessoas que entram no recanto para utilizar-se de algum dos serviços oferecidos.
Usuário Máster	É o usuário que opera o sistema com o poder de alterar, excluir ou incluir qualquer tipo de dado dentro do sistema, ou seja, o gerente do estabelecimento, o qual poderá manipular tudo o que o sistema permitir.
Usuários do sistema	São os usuários do sistema que, como o usuário máster, podem alterar excluir ou incluir dados, entretanto, algumas funcionalidades de manipulação são permitidas tão somente ao funcionário máster.
Atrações	São todos os serviços oferecidos aos clientes, assim como: lanchonete, piscinas, pesque-pague, entre outros.
Pulseira	Todos os clientes recebem uma pulseira na entrada do recanto, esta pulseira serve como forma de comprovação de que o cliente pagou por algum determinado serviço. Esta pulseira tem uma cor e um código, através deste código que é inserido no sistema, o usuário do sistema pode consultar quais serviços o cliente está utilizando.
Consumo	Serve para marcar a situação dos serviços que os clientes utilizam, mesmo para aqueles que desejam pagar no fim do dia, para isso serve o consumo diário.
Relatórios	O gerente pode solicitar ao sistema um controle de seus clientes para saber quais gastam mais, ou quais os clientes que mais frequentam o recanto e, através destes relatórios, o usuário máster do sistema poderá ter esta visualização e controle.
Histórico	É um tipo de relatório onde é exibida a movimentação de um determinado cliente, em um determinado período.

c) Mapa conceitual:

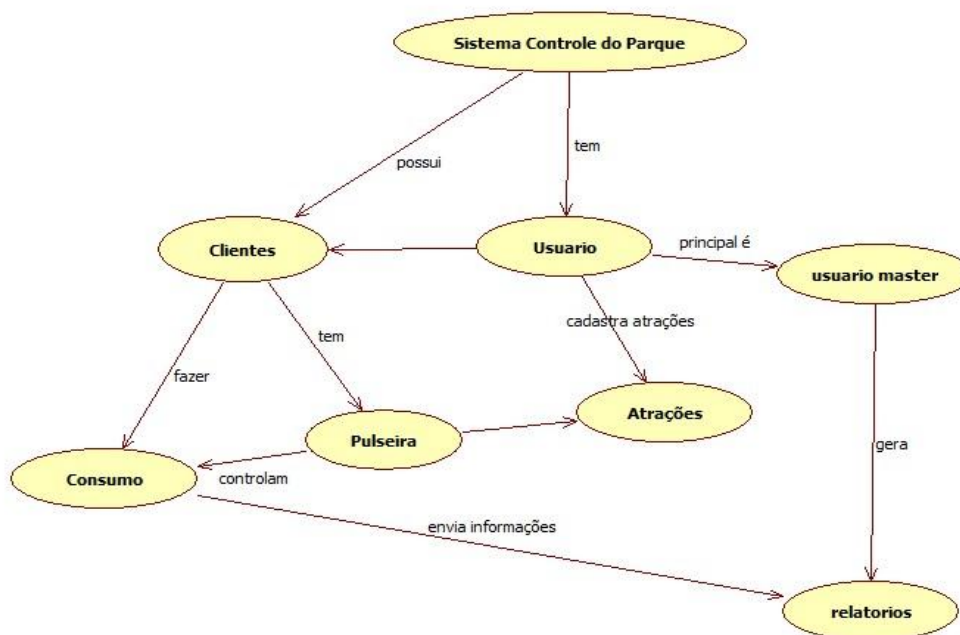


Figura 29 - Mapa conceitual - Trabalho 6
Fonte: Autor

Trabalho 7

a) Resumo da Análise de Requisitos: SASSIS - Sistema de Controle de Estoque.

A empresa abordada se trata de uma Associação de Agricultores Ecologistas sem fins lucrativos, a qual tem como finalidade potencializar desenvolvimento da Agricultura, Combate à Fome, Preservação do Meio Ambiente através da execução de Programas Sociais. A associação compra e vende produtos que beneficiam os agricultores e diversas entidades no seu campo de atuação.

Devido às dificuldades apresentadas pela Associação abordada, em virtude de não possuir nenhuma ferramenta para controle de seus estoques, propomos então, a elaboração de um **Software** que deverá atender as necessidades principais, como **controle de estoque**, cadastro de clientes, cadastro de **produtos**.

O **software** irá possuir um cadastro de **clientes e fornecedores**, com todas as informações necessárias de cada um deles, facilitando seu controle e pesquisas sobre eles ou mercadorias fornecidas ou adquiridas por cada um. Pretendemos desenvolver também um controle de estoque, cadastrando as mercadorias com seu tipo, quantidade e o fornecedor. Conforme as mercadorias forem sendo adquiridas será possível fazer consultas no estoque e verificar as necessidades de reposição. O programa deverá emitir um **cupom não fiscal** que será entregue ao cliente.

Pretendemos com o desenvolvimento deste programa facilite o trabalho desenvolvido por essa associação, que não possui fins lucrativos e não possui recursos para adquirir novas tecnologias que facilitem sua atividade que hoje vem sendo acompanhada através de planilhas de entrada e saída de produtos, sem nenhum controle do estoque.

Haverá uma pessoa responsável pelo sistema denominado gerente.

b) Conceitos das palavras grifadas:

Palavras	Conceitos atribuídos
Vendedor	Quem faz a venda dos produtos.
Fornecedor	Agricultores familiares associados à Assis, os quais produzem os alimentos orgânicos.
Produto	Frutas, verduras e outros.
Cliente	São pessoas físicas ou jurídicas que compram os produtos da Associação Assis.
Estoque	Tudo que se tem disponível para vender.
Cupom não Fiscal	Demonstrativo de venda.
Compra	Aquisição de produtos dos agricultores.
Venda	Distribuição dos produtos para os programas sociais.
Gerente	Responsável pelo sistema.

c) Mapa conceitual:

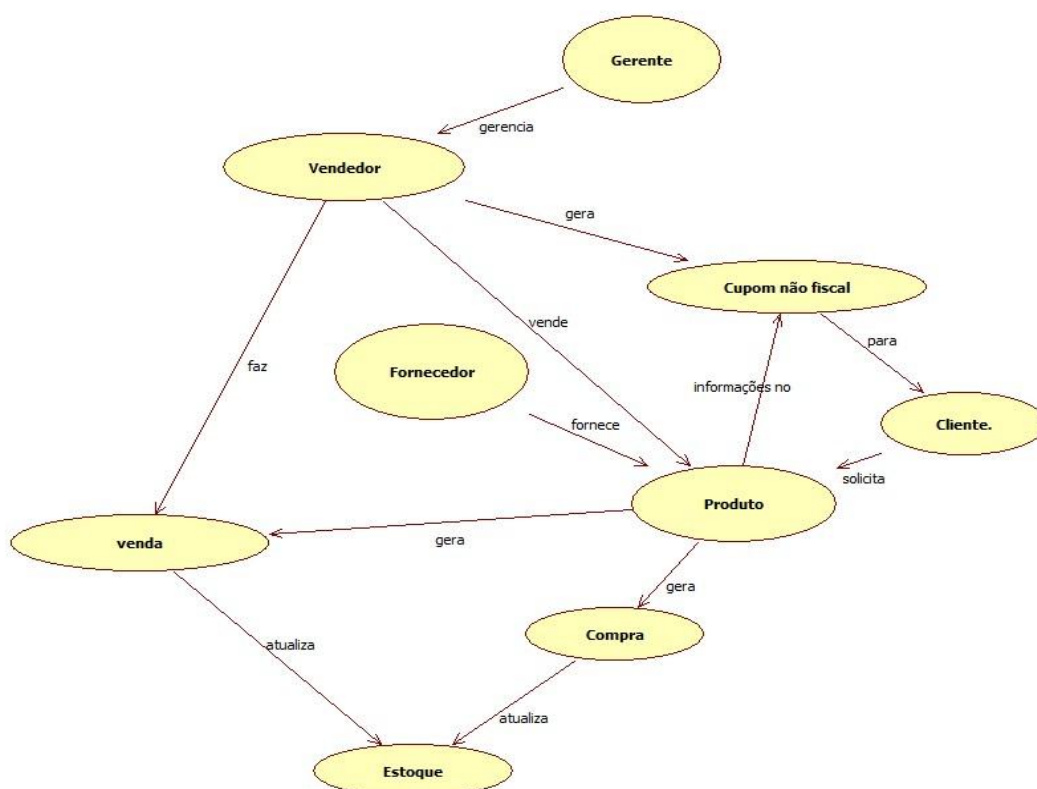


Figura 30 - Mapa conceitual - Trabalho 7

Fonte: Autor

Trabalho 8

a) Resumo da Análise de Requisitos: Controle de Playground

O sistema a ser desenvolvido, servirá para controlar um playground que terá como requisitos: **cadastrar a criança**: nome da criança, a data de nascimento, nome do responsável. Também será realizado o **cadastro dos responsáveis** onde serão coletadas informações como: nome, RG, CPF, telefone para contato e nome da criança. Feito o cadastro, será emitida uma pulseira para uso da criança contendo a sua identificação, bem como a de seus responsáveis. Também será emitido um comprovante para o responsável o qual deverá ser entregue na hora da saída da criança do playground. O **cálculo do valor** a ser pago dependerá do tempo permanecido que será feito a partir da hora de entrada até a hora da saída da criança (O valor a ser cobrado será por hora) .O cadastro da criança e do seu responsável será salvo para que este procedimento não seja refeito várias vezes economizando tempo e servindo como **controle de frequência de clientes**. O Playground está localizado dentro de uma loja, a qual disponibilizará **cupom fiscal** que, **ao ser** apresentado no playground, o cliente poderá receber um **desconto** (Para receber o desconto o cliente deverá gastar no mínimo, R\$ 50,00 para ganhar uma hora grátis no playground). O sistema conta também com o **relatório de faturamento** que tem por função emitir o **faturamento** diário, semanal, mensal e anual. Este relatório apresentará a quantidade de clientes que utilizaram o playground, separando os clientes pagantes dos clientes que utilizaram o cupom de desconto e a arrecadação total.

b) Conceitos das palavras grifadas:

Palavras	Conceitos atribuídos
Playground	Parque de diversões infantil.
Cadastro da criança	Fará a coleta de informações pessoais referentes à criança e à vinculação de seus responsáveis.
Cadastro do responsável	Fará a coleta de informações pessoais referentes aos responsáveis e à vinculação com a criança.
Emitir pulseira	Serve para uso da criança, onde conterà informações pessoais e de seus responsáveis.
Comprovante	Entregue aos responsáveis para comprovação de vínculo com a criança.
Cálculo do valor	Feito através da entrega do comprovante calculando o horário permanecido no local.
Tempo permanecido	Necessário para calcular o valor a ser pago.
Controle de frequência	Para saber quais os clientes que mais utilizam o parque.
Cupom fiscal	Contendo data, valor gasto e número de cupom usado para receber desconto no playground.
Desconto	Cedido para clientes que utilizarem a loja anexa ao parque.
Relatório de faturamento	Utilizado para demonstrar o faturamento do playground podendo ser comparado ao faturamento de diferentes dias/semanas/meses/anos.
Faturamento	Valor líquido durante tais períodos (dias, semanas, meses, anos).
Clientes	Todas as pessoas que usam os serviços do playground.

c) Mapa conceitual

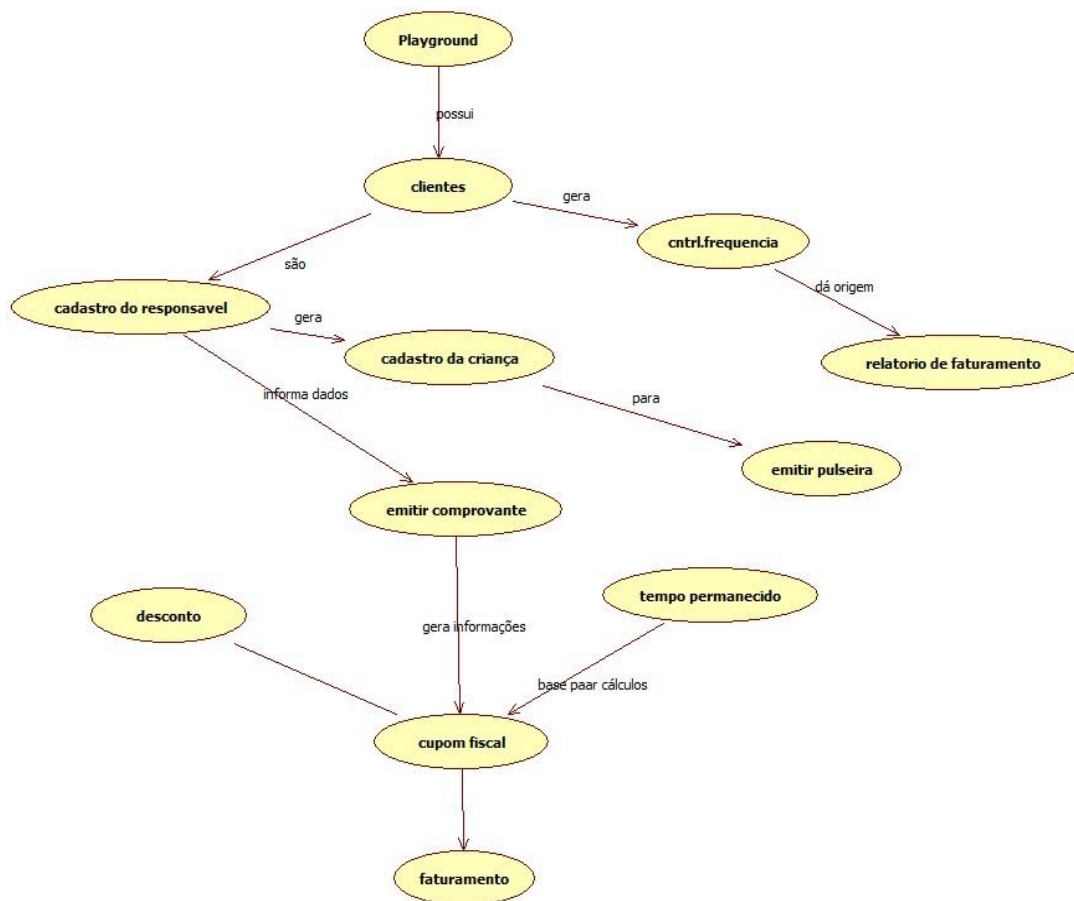


Figura 31 - Mapa conceitual - Trabalho 8
Fonte: Autor

APÊNDICE C - Respostas completas da entrevista coletiva

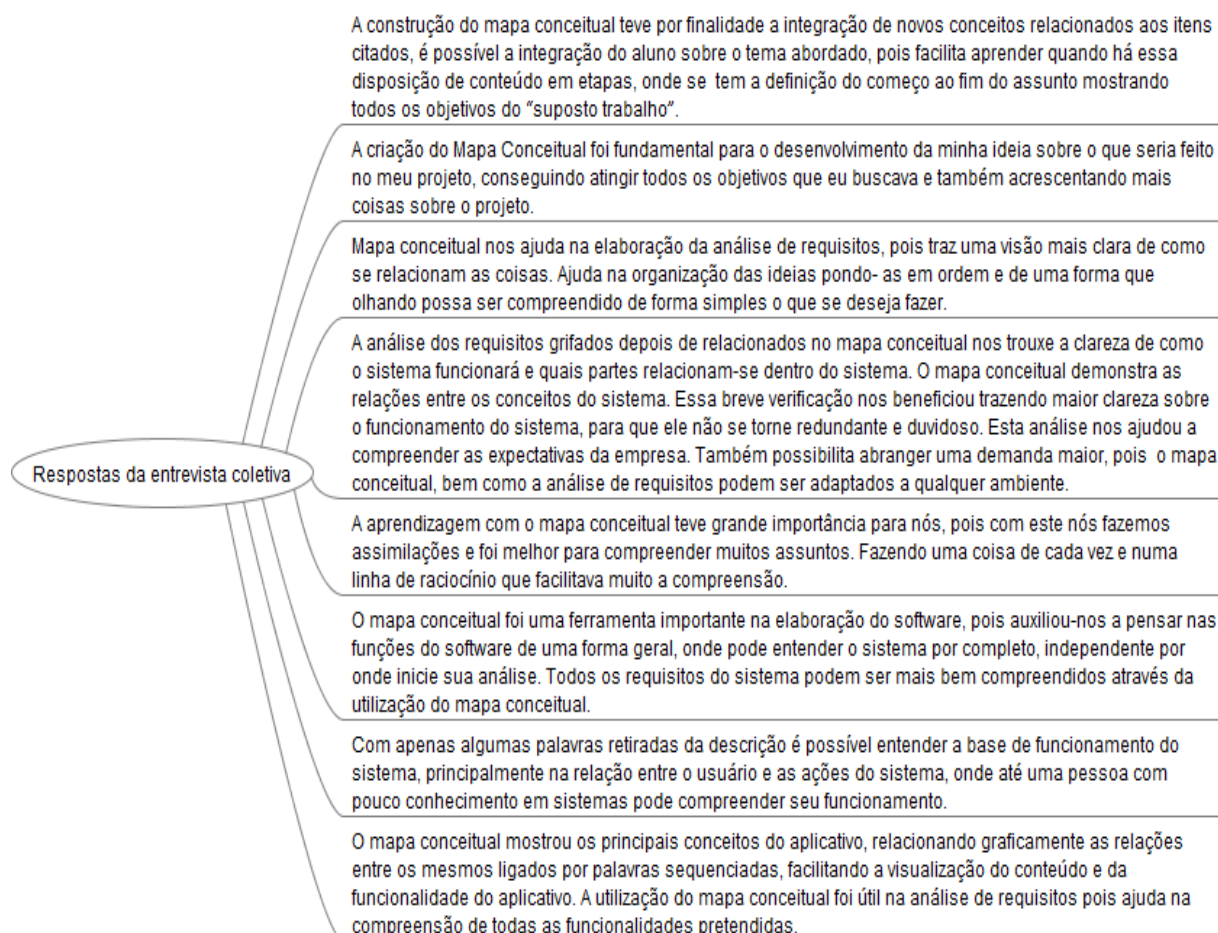


Figura 32 - Entrevista coletiva com os alunos

Fonte: O autor