Dado a especificação mais detalhada do novo SISGEE a ser feito, verificou-se a necessidade de trabalharmos mais com a entidade do *convênio*, onde efetuamos o cadastro da *empresa* ou *pessoa* que são atributos do *convênio*. Além disso, a classe *agente de integração* deve ser eliminada tendo em vista que não será usada para mais nada e agente de integração será um atributo *boolean* da empresa.

O caso de uso que antigamente era Cadastrar Empresa será substituído por **Cadastrar Convênio** que por sua vez, levará a implementação de 3 novos casos de uso, **Buscar Convênio**, **Renovar Convênio** e **Relatório de Convênios a Vencer**; toda parte do convênio será feita do zero, porém será aproveitado grande parte das validações já implementadas no *ValidaUtils*.

A relação convênio ficará com o seguinte esquema:

*Convenio(idConvenio, numeroConvenio, numero, ano, dataAssinatura, id\_empresa, id\_pessoa)*

As classes envolvidas no Cadastramento do convênio e que foram usadas e ou modificadas são: Convenio, ConvenioServices, ConvenioDao, Empresa, EmpresaServices, EmpresaDao, Pessoa, PessoaServices, PessoaDao. Os servlets usados foram: ValidaConvenioServlet, que faz a validação dos dados e IncluiConvenioServlet que efetivamente inclui o convenio. Os jsps utilizados para o cadastro foram: form\_cadastrar\_convenio.jsp para que o usuário inclua os dados do convênio e o cadastro\_convenio\_sucesso.jsp que emite uma mensagem de cadastro concluído e o número do convênio.

Terá 2 construtores, dado que para possuir um convênio, ou é uma empresa, ou é uma pessoa, nunca os 2 podem ser diferentes de null ao mesmo tempo.

*Public Convenio(String ano, String numero, Date dataAssinatura, Empresa empresa* ***OU*** *Pessoa pessoa)*

O atributo *numeroConvenio* é um atributo derivado dos atributos *ano* e *numero* devido ao **Renovar Convênio**, visto que ao renovar deve-se mudar apenas o ano do número convênio.

As classes envolvidas no Renovar Convênio e que foram usadas e ou modificadas são: Convenio, ConvenioServices, ConvenioDao. Os servlets foram: BuscarConvenioServlet, que foi criado para efetuar a busca do convênio já cadastrado e RenovarConvenioServlet que faz a atualização dos dados do convênio renovado. Os jsps utilizados para a renovação foram: form\_renovar\_covenio.jsp que é onde ocorre a busca do convênio para que possa ser realizada a renovação, renovar\_convenio.jsp que é onde o usuário faz as alterações desejadas dos dados para a renovação do convênio e cadastro\_convenio\_sucesso.jsp que emite a mensagem de cadastro concluído e o novo número do convênio.

No **Relatório de Convênios a Vencer**, faz-se um cálculo para verificar-se todos convênios que estão a 2 meses de vencer de acordo com a data atual e à data que o convênio foi cadastrado. A partir disso todos os convênios que estão prestes a vencer irão aparecer na tela mostrando ao usuário a data que o convênio foi cadastrado e a data que ele irá vencer, assim como outras informações dele.

As classes envolvidas no Convênios a vencer, que foram usadas e ou modificadas são: Convênio, ConvenioServices, ConvenioDao, ItemConvenioJoin. E o servlet criado foi: RelatorioConvenioServlet. O jsp utilizado foi form\_relatorio\_convenio\_vencer.jsp

Na parte de **Emitir Relatório Consolidado** não teve muita mudança, apenas uma chamada a mais para um novo jsp que foi criado para imprimir o relatório. Será aproveitado o sistema atual nessa parte, só incrementando essa tela de impressão.

Tendo em vista as modificações da especificação do novo Sisgee, o caso de uso **Registrar aditivo termo estágio** deve ser alterado para incluir os novo campos e atributos. Com a entrada dos tipos de aditivos, Se faz necessário uma mudança no esquema do **Termo Aditivo**, além o acréscimo dos atributos novos, com a entrada do **tipo de aditivo** se faz necessário permitir que todos os campos respectivos a os tipos possam ser inseridos como *null* no banco, tendo assim inserido somente o atributo correspondente ao seu tipo. Neste caso de uso não teve muitas alterações na estrutura do caso de uso.

Foram adicionados três novos atributos :

nomeSupervisor, cargoSupervisor, eobrigatorio .

As classes envolvidas no **Registrar aditivo termo estágio**, que foram usadas e ou modificadas são: form\_termo\_aditivo.jsp ,IncluirTermoAditivoServlet, ValidaUtils, TermoAditivo, TermoEstagio , : ProfessorOrientadorServices, GenericDao ,TermoEstagioServices, TermoEstagioDAO .

O caso de uso **Consultar Termo Estágio/Aditivo** precisará de um atributo que não está contido nem em Aditivos e nem em Termo Estágios que é **status**, esse atributo será calculado a partir da comparação da data atual com a data do Fim e da data da rescisão do Termo Estágio ou Aditivo, mostrando assim se é ativo ou inativo. A base do caso de uso não será muito alterada , porém terá que ser ajustadas para mostrar os atributos que o caso de uso exige . Neste caso de uso foi criado o form\_buscar\_termo\_aditivo.jsp visando uma melhor organização. Os servlets foram refeitos para deixar mais simples e organizado.

As classes envolvidas no **Consultar Termo Estágio/Aditivo**, que foram usadas e ou modificadas são: form\_buscar\_termo\_aditivo.jsp , BuscaAlunoServlet, ValidaUtils, AlunoServices , AlunoDAO, BuscaTermoAditivoServlet, TermoEstagioServices, TermoEstagioDAO , TermoAditivoServices, TermoAditivoDAO.

A relação **TermoAditivo** ficará com o seguinte esquema do banco de dados:

termoaditivo(idtermoaditivo, bairroenderecotermoaditivo, cargahorariatermoaditivo, cargosupervisor, cependerecotermoaditivo, cidadeenderecotermoaditivo, complementoenderecotermoaditivo, datacadastramentotermoaditivo, datafimtermoaditivo, enderecotermoaditivo, eobrigatorio, estadoenderecotermoaditivo, nomesupervisor, numeroenderecotermoaditivo, tipoaditivo, valorbolsatermoaditivo, professororientador\_idprofessororientador, termoestagio\_idtermoestagio).

O **Registro de Termo Estágio** continua semelhante à versão anterior do *SISGEE*, portanto, os *servlets* podem ser parcialmente reaproveitados. O *form* de registro sofreu algumas mudanças, então o *servlet* de validação também teve de ser alterado, além de alguns campos como Nome e Cargo do Supervisor do estágio que foram acrescentados. Este caso de uso também depende dos casos Buscar Aluno e Buscar Convênio, que encontram e preenchem no *form* de registro os campos correspondentes e são utilizados por outros c/asos de uso nessa nova versão.

A entidade do convênio agora é mais significante no sistema, mas sua relação estrutural com Termo Estágio continua a mesma. O que muda na nova versão é o método de incluir termo estágio, em *TermoEstagioServices*, que agora passa como parâmetro um objeto de convênio e não mais empresa.

O Termo de Estágio sofreu modificações na Entidade do TermoEstagio com acréscimo de Métodos e campos. Foram alterados/incluídos: BuscarAlunoServlet, TermoEstagioServices, FormTermoEstagioServlet, IncluirTermoEstagioServlet, a ValidaUtils, a Entidade Aluno, o Botão BuscaConvenioBotaoServlet, form\_termo\_estagio.jsp e o import\_scripts.jspf.

A Entidade TermoEstagio contará com o seguinte novo formato:

TermoEstagio(Integer idTermoEstagio, String estado, Date dataInicioTermoEstagio, Date dataFimTermoEstagio, Date dataRescisaoTermoEstagio, Integer cargaHorariaTermoEstagio, Float valorBolsa, String enderecoTermoEstagio, String numeroEnderecoTermoEstagio, String complementoEnderecoTermoEstagio, String bairroEnderecoTermoEstagio, String cepEnderecoTermoEstagio, String cidadeEnderecoTermoEstagio, String estadoEnderecoTermoEstagio, Boolean eEstagioObrigatorio, Aluno aluno, Convenio convenio, String nomeSupervisor, String cargoSupervisor, String nomeAgenciada, ProfessorOrientador professorOrientador, List<TermoAditivo> termosAditivos)

Seram realizadas modificações nos Construtores para usarem os campos de nomeSupervisor, cargoSupervisor e nomeAgenciada quando os mesmo forem utilizados.

Na versão anterior, durante o Cadastrado do Termo era criado um Convênio para a Empresa selecionada. Isso será eliminado já que o Convênio é inserido previamente.

**Rescisão do Termo de Estágio**

A Rescisão de Termo Estágio do UC08 – Encerrar Termo de Estágio agora se dará pela tela Consulta de Termos/Aditivos no menu Consulta / Aditivos / Rescisão do Termo. Nessa tela, além de realizar a consulta aos Termos de Estágios e Aditivos de um Aluno, existem outras opções que são a de Cadastrar Novo Aditivo para um Termo de Estágio Ativo e Rescindir um Termo de Estágio.

A Rescisão do Termo não tem uma Entidade já que usa do TermoEstagio e TermoAditivo para realizar o encerramento do Termo de Estágio e seus Aditivos, através dos itens modificados do form\_termo\_rescisao.jsp e do FormTermoRescisaoServlet.

**Busca Aluno**

A Busca do Aluno do UC11 – Buscar Aluno será realizada através de um botão que deve acionar um JSON do import\_scripts.jspf e do BuscarAlunoServlet modificados para trazer novas informações sobre o Aluno de acordo com a Matrícula digitada.

Foi criado um diagrama de implantação, de componente, de atividades e um de estado para termo aditivo.

Os casos de uso antigos eram:

UC01 - Cadastrar Empresa

UC02 – Cadastrar Termo de Estágios

UC03 - Cadastrar Aditivo para um Termo de Estagio

UC04 - Consultar Termos/Aditivos de um Aluno

UC05 - Encerrar Termo de Estagio

UC06 - Emitir Relatório Consolidado

Na mudança obtivemos:

UC01 - Cadastrar Convênio

UC02 - Renovar Convênio

UC03 - Emitir Relatório de Convênios a Vencer

UC05 - Registrar Termo de Estágio

UC06 - Consultar Termos de Estágio

UC07 - Registrar Aditivo de Termo Estágio

UC08 - Encerrar Termo Estagio

UC09 - Emitir Relatório Consolidado

UC10 - Buscar convênio

UC11 - Buscar Aluno

UC12 - Visualizar Termo/Aditivo