

# **Documentação Técnica da Aplicação *Web* Homework**

**Diego Martins Alves de Souza  
Werlerson Andrade Frois  
Renata Pontin de Mattos Fortes**

([conchas@grad.icmc.usp.br](mailto:conchas@grad.icmc.usp.br))  
([wandrade@grad.icmc.usp.br](mailto:wandrade@grad.icmc.usp.br))  
([renata@icmc.usp.br](mailto:renata@icmc.usp.br))

RELATÓRIOS TÉCNICOS DO ICMC

São Carlos  
Setembro/2006

## Índice

1. Introdução.....	3
2. Visão Geral de Homework.....	3
3. Requisitos de Homework.....	3
4. Projeto Lógico e Implementação de Homework .....	7
4.1 Estrutura de diretórios e subdiretórios.....	8
4.2 Descrição da Implementação.....	8
4.2.1. Diretório homework.....	8
4.2.2. Diretório css.....	10
4.2.3. Diretório emails.....	10
4.2.3.1. Subdiretório comments.....	10
4.2.3.2. Subdiretório homeworks.....	10
4.2.3.3. Subdiretório users .....	10
4.2.4. Diretório filter.....	10
4.2.5. Diretório includes.....	11
4.2.6. Diretório library.....	11
4.2.7. Diretório xml.....	17
4.2.7.1. Subdiretório skel.....	19
4.2.8. Diretório xsl.....	19
4.2.8.1. Subdiretório errors .....	20
4.2.8.2. Subdiretório messages.....	21
4.3 Diagrama Entidade Relacionamento .....	21
4.4 Esquema Relacional do Banco de Dados de Homework .....	22
5. Instalação de Homework.....	22
6. Manual do usuário.....	24
6.1 Manual do administrador.....	24
6.2 Manual do estudante.....	28
7. Considerações Finais.....	32

# 1. Introdução

Este relatório apresenta a documentação básica de Homework, uma aplicação *web* que foi desenvolvida no ICMC-USP. Neste relatório estão os principais documentos para o entendimento técnico da solução implementada na aplicação Homework, contendo: uma descrição dos seus requisitos, seu projeto lógico funcional e de dados, bem como detalhes de sua implementação.

O objetivo dessa documentação é propiciar uma visão abrangente dos recursos implementados em Homework e incentivar que a ferramenta seja mais rapidamente compreendida por desenvolvedores interessados em evoluir suas características já disponibilizadas.

## 2. Visão Geral de Homework

O software **Homework** é uma aplicação *web* que foi desenvolvida no ICMC-USP, por Diego Martins Alves de Souza, em seu projeto de Iniciação Científica – PIC-SCC, visando oferecer um apoio automatizado para o gerenciamento das atividades relacionadas com a comunicação e entrega de trabalhos práticos das disciplinas de computação. Por ser concebida como um apoio, ela é denominada ferramenta neste relatório, a qual deve ser utilizada sob demanda dos usuários envolvidos com as atividades de realização dos trabalhos práticos, especialmente professores, alunos e monitores.

## 3. Requisitos de Homework

Os requisitos prioritários, *não-funcionais*, que foram considerados para o projeto de desenvolvimento da aplicação *web* Homework, foram: a **interoperabilidade**, a **manutenibilidade** e **acessibilidade**.

Esses requisitos, embora não estabeleçam os serviços a serem disponibilizados pela ferramenta, orientaram a arquitetura e tecnologias adotadas durante o seu desenvolvimento.

Assim, para garantir **manutenibilidade**, foi utilizada a abordagem de Separação de Interesses (*Separation of Concerns* ou *SoC*) durante o desenvolvimento desta aplicação *web* (<http://safe.icmc.usp.br/homework>). Esta abordagem consiste em separar os elementos de lógica, conteúdo e apresentação, que são essenciais em aplicações *web*.

Dessa forma, o desenvolvimento da implementação da ferramenta, com base nesta abordagem, foi realizado com utilização das seguintes tecnologias:

- a lógica em programas codificados na linguagem de programação PHP,
- o conteúdo das informações tratadas na aplicação em XML e,
- a apresentação das informações processadas por XSLT.

Para garantir **interoperabilidade**, como já se trata de aplicação *web*, foi considerada a configuração de desenvolvimento com LAMP -- Linux, Apache, MySQL e PHP.

Visando atender o requisito de **acessibilidade**, o projeto da interface foi desenvolvido sem necessidade de código em JavaScript e procurando sempre apresentar de forma harmoniosa as informações tratadas.

A ferramenta foi desenvolvida inicialmente na forma de protótipo, seguindo o modelo de processo de prototipação, para o estabelecimento mais preciso dos *requisitos funcionais*. Assim, foi possível a realização do primeiro estudo sobre o processo de suporte ao acompanhamento dos trabalhos práticos das disciplinas de Computação. A partir deste estudo inicial, foram obtidas informações suficientes para o desenvolvimento da primeira versão da ferramenta, que foi implementada (Homework v1.0) e utilizada, durante o segundo semestre de 2004, com os alunos das disciplinas sce602 e sce181

(Introdução a Ciência da Computação II). Além de estabelecer os requisitos funcionais, foi também objetivo dessa primeira versão da ferramenta avaliar as funcionalidades mais importantes e coletar outros requisitos que não haviam sido reportados durante a fase de projeto. Vale mencionar que a ferramenta Homework v1.0, na forma de um protótipo, não foi construída utilizando a abordagem de Separação de Interesses (SoC), pois seu propósito era principalmente auxiliar o estudo detalhado dos requisitos.

Assim, a Homework v1.0 possuía como requisitos funcionais, os seguintes:

- suporte à identificação dos alunos e controle de acesso,
- suporte ao envio de arquivos (referentes à entrega dos trabalhos práticos) dentro de prazo determinado,
- suporte à edição e divulgação de notas aos alunos,
- e suporte à definição de grupos para realização de trabalhos.

Essas funcionalidades da ferramenta Homework v1.0 se aplicavam somente no contexto de *uma* determinada disciplina ou curso.

Durante a utilização da ferramenta Homework v1.0, foi possível identificar que o **processo** para acompanhamento do desenvolvimento dos trabalhos práticos nas disciplinas pode ser descrito resumidamente da seguinte forma:

(1) Um responsável pela disciplina divulga a tarefa a ser realizada.

Eventualmente, é necessário que algum arquivo contendo instruções especiais ou um material de apoio ao desenvolvimento seja anexado ao enunciado do trabalho. Neste momento, são estabelecidos: (a) uma data para o encerramento da atividade de realização da tarefa por partes dos alunos e, (b) o número máximo de alunos que podem compor os grupos (quando for o caso de trabalhos a serem desenvolvidos em grupos);

- (2) Os alunos acessam o enunciado do trabalho para se inteirar do trabalho a ser realizado. Eventualmente, os trabalhos devem ser realizados em grupo, cuja composição deve possuir um número limitado de participantes, definido previamente pelo responsável da disciplina. Assim, os alunos podem se organizar em grupos, convidando os colegas para a composição de grupos;
- (3) Antes do encerramento do prazo para entrega do trabalho, os alunos podem enviar os arquivos contendo os resultados do trabalho realizado, para o professor ou monitor da disciplina;
- (4) O monitor ou professor recebe todos os trabalhos, avalia cada um deles e divulga os resultados da avaliação dos mesmos;
- (5) Eventualmente os trabalhos são disponibilizados para *download* pelos responsáveis da disciplina visando exibir, para todos os estudantes, outras formas de realização da mesma tarefa.

É possível observar que, neste processo, as atividades em grupo são essenciais. Para dar suporte às atividades realizadas em grupos, notou-se a necessidade também de que algumas características de *groupware* fossem incluídas na ferramenta. Entre elas, podemos citar:

- Existência de um meio de comunicação entre as pessoas envolvidas no desenvolvimento de um mesmo trabalho, seja para realizar uma discussão sobre as decisões a serem tomadas, questionamentos, mudanças de planos, *deadlines* ou mesmo o foco do trabalho;
- Possibilidade de arquivamento dessas discussões para que possam servir de memória das discussões, para possível análise futura das razões sobre as decisões tomadas;
- Envio imediato de e-mail para os devidos usuários do sistema, quando for feita uma modificação no enunciado dos trabalhos;

- Possibilidade que os grupos sejam formados de maneira flexível, de modo que os próprios estudantes sejam responsáveis pela sua formação e rearranjo durante a fase de desenvolvimento.

A partir desse novo conjunto de requisitos funcionais, a ferramenta Homework foi re-implementada, conforme descrito nas próximas seções.

## 4. Projeto Lógico e Implementação de Homework

Para possibilitar que o desenvolvimento da implementação de Homework adotasse o conceito de Separação de Interesses (SoC), foi utilizada a seguinte arquitetura/ estrutura de organização dos interesses:

**Lógica:** *Scripts* PHP são utilizados para realizar a lógica da aplicação, processamento numérico, consultas ao banco de dados, controle de seção e etc.

**Dados:** Os dados utilizados para exibição de todas as páginas são representados com documentos XML que são gerados dinamicamente utilizando o processador de templates *Smarty*. Associada a cada página existe um template de documento XML, que é processado todas as vezes que a página é acessada para a obtenção dos dados necessários para a exibição desta página. Desta forma, temos isoladamente o conceito *dado* representado por um documento XML.

**Apresentação:** Uma vez que temos todos os dados que comporão a página representados por um documento XML, uma transformação XSLT é aplicada a este documento para obtermos um documento XHTML que pode ser exibido com a devida formatação na grande maioria dos

*browsers*. A forma como esta transformação é realizada é parametrizada e pode ser modificada, possibilitando que seja realizada no servidor ou no cliente, por padrão esta transformação é realizada no servidor, pois não são todos os *browsers* que possuem a funcionalidade de realizar transformações XSLT.

Nas seguintes subseções, são detalhados os elementos que compõem a implementação desenvolvida em Homework.

## **4.1 Estrutura de diretórios e subdiretórios**

A estrutura de diretórios e subdiretórios contendo os arquivos da ferramenta Homework, é apresentada na Figura 1. Os arquivos, contendo o código-fonte da ferramenta Homework, são descritos nas subseções a seguir.

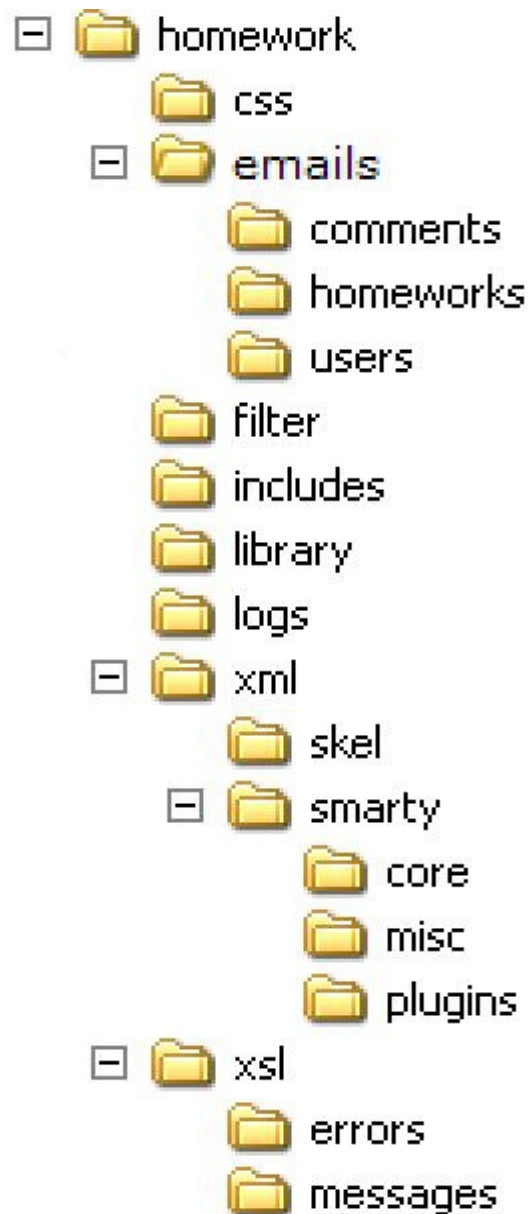
## **4.2 Descrição da Implementação**

### **4.2.1. Diretório homework**

1. **comments.php**: Adiciona novo comentário a um determinado trabalho e envia os e-mails aos usuários que serão capazes de ver o comentário criado.
2. **download.php**: Realiza o *download* do arquivo anexado.
3. **edit\_classes.php**: Exibe as turmas existentes e realiza a inserção ou remoção de turmas.
4. **edit\_grades.php**: Realiza a edição de notas.
5. **edit\_groups.php**: Manipula os grupos dos trabalhos: criação, remoção e abandono de um grupo, adição e remoção de integrantes.
6. **edit\_profile.php**: Realiza a alteração de senha do usuário.
7. **edit\_users.php**: Realiza a adição e remoção do usuário, alteração dos dados do usuário e sua inclusão em determinada turma.
8. **homeworks\_list.php**: Lista os trabalhos do usuário.
9. **index.php**: Página inicial da ferramenta Homework. Realiza a operação de login na ferramenta.
10. **invitations.php**: Manipula os convites para ingresso em grupo.
11. **new\_homework.php**: Cria novos trabalhos.
12. **new\_pass\_request.php**: Envia e-mails para requisição de nova senha de acesso à ferramenta.
13. **setup.php**: Cria as tabelas no banco de dados e o diretório para a compilação dos templates.



14. **show\_homework.php**: Gerencia a exibição das informações do trabalho e funcionalidades disponíveis para sua manipulação, de acordo com o usuário logado e status do trabalho.
15. **upload.php**: Realiza *upload* de arquivos.



**Figura 1.** Estrutura de diretórios e subdiretórios de Homework.

#### 4.2.2. Diretório css

1. **homework.css**: Contém a formatação da utilizada na ferramenta Homework.
2. **menu.css**: Contém a formatação do menu e submenu.
3. **menu-ie.css**: Contém a formatação do menu e submenu para o Internet Explorer.

#### 4.2.3. Diretório emails

1. **footer.tpl**: Arquivo que contém a assinatura do e-mail.

##### 4.2.3.1. Subdiretório comments

1. **new\_comment.tpl**: Arquivo que contém as informações de e-mail referentes ao novo comentário adicionado.
2. **new\_comment\_sub.tpl**: Arquivo que contém o assunto do e-mail referente ao novo comentário adicionado.

##### 4.2.3.2. Subdiretório homeworks

1. **new\_homework.tpl**: Arquivo que contém as informações de e-mail referentes ao novo trabalho adicionado.
2. **new\_homework\_sub.tpl**: Arquivo que contém o assunto de e-mail referente ao novo trabalho adicionado.

##### 4.2.3.3. Subdiretório users

1. **new\_password.tpl**: Arquivo que contém as informações de e-mail referentes a alteração de senha.
2. **new\_password\_sub.tpl**: Arquivo que contém o assunto de e-mail referente a alteração de senha.
3. **new\_user.tpl**: Arquivo que contém as informações de e-mail referentes ao cadastro no Homework.
4. **new\_user\_sub.tpl**: Arquivo que contém o assunto de e-mail referente ao cadastro no Homework.
5. **password\_request.tpl**: Arquivo que contém as informações de e-mail referente a confirmação de alteração de senha.
6. **password\_request\_sub.tpl**: Arquivo que contém o assunto de e-mail referente a confirmação de alteração de senha.

#### 4.2.4. Diretório filter

1. **kses.php**: Arquivo que contém filtro pra tags html.

### 4.2.5. Diretório includes

1. **config\_hw.php**: Arquivo de configuração da ferramenta Homework.
2. **global\_functions.php**: Arquivo que contém as funções globais utilizadas na ferramenta.
3. **pear.php**: Arquivo que cria variável global de conexão com o banco de dados
4. **script\_inicialization.php**: Arquivo que inclui os arquivos necessários para inicialização dos scripts. Deve ser incluído em todo script php da aplicação.
5. **validation.php**: Arquivo que contém as funções que verificam se os campos de e-mail e senha foram preenchidos corretamente.

### 4.2.6. Diretório library

1. **class.class.php**: Arquivo que contém as seguintes funções para manipulação de notas:

```
HWclass()
load($class_id)
loadWithStudents($class_id)
setClassId($class_id)
setClassName($class_name)
setDisciplineName($discipline_name)
setDisciplineCode($discipline_code)
setYear($year)
setCreator($user)
setStudents($students)
getId()
getStudents()
getUsers($class_id, $type)
getAllClasses()
create()
remove()
removeUsersFromThisClass()
removeUser($user)
addUser($user)
```

2. **comment.class.php**: Arquivo que contém as seguintes funções para manipulação de comentários.

```
Comment()
load($comment_id)
store()
getHomeworkId()
getFrom()
getTo()
getType()
setId($comment_id)
setHomework($homework)
setHomeworkId($homework_id)
```

```

setFrom($user)
setFromId($user_id)
setComment($text)
setUpload($upload)
setUploadId($upload_id)
setType($type)
setTo($to)
setToId($to_id)
setCommentDate($date)
canBeDone()
getHomeworkComments($homework, $user)
getStudentsQuestions($homework, $user)

```

3. **email.class.php**: Arquivo que contém as seguintes funções para manipulação de e-mail.

```

Email()
setTo($to)
setFrom($from)
setSubject($subject)
setBody($body)
getRecipients()
setRecipients($object, $homework)
addRecipient($recipient)
getEmailsToSend()
sendGroupInvitation($users, $group)
NotifyNewComment($homework, $comment)
NotifyNewHomework($homework)
NotifyNewUser($user)
newPasswordRequestConfirmation($user, $request_code)
notifyPasswordChange($user, $new_password)
send()

```

4. **error.class.php**: Arquivo que contém as seguintes funções para manipulação dos erros exibidos na ferramenta.

```

Error()
setError($error)
isSeted()
setXslFile($file)
setXmlFile($file)
logError()
getLogMessage()
throw()
throwUserError($user, $error_message)
display()

```

5. **filter.class.php**: Arquivo que encapsula as funcionalidades da aplicação Kses<sup>1</sup>, distribuída como software livre, e contém as seguintes funções.

```
Filter ()  
htmlFilter ()  
filterHash ()
```

6. **grade.class.php**: Arquivo que contém as seguintes funções para edição de notas dos usuários.

```
Grade()  
setStudent($student)  
setHomework($homework)  
setHomeworkId($homework_id)  
getStudent()  
getHomework()  
getValue()  
setValue($value)  
store()  
removeGrade()  
load($student, $homework)
```

7. **group.class.php**: Arquivo que contém as seguintes funções para manipulação de grupos.

```
Group ()  
load ($group_id)  
loadMembersUpload($homework)  
getRecipients($homework)  
getId ()  
getName ()  
getHomeworkId ()  
getOwner ()  
getMembers()  
SetId ($group_id)  
SetName ($name)  
SetHomeworkId ($homework_id)  
SetOwner ($owner_id)  
GetGroupName($group_id)  
getHomeworkIdFromGroup ($group_id)  
create()  
addOneMember ($user)  
removeOneMember ($user)  
deleteGroup()  
getGroupIdFromHomework ($user, $homework)  
haveUser($user_id)  
sendInvitationsTo($recivers, $sender)
```

---

<sup>1</sup> <http://sourceforge.net/projects/kses>

```
getAllGroups($homework)
updateGrade($homework, $grade)
```

8. **homework.class.php**: Arquivo que contém as seguintes funções para criação, listagem e exibição de trabalhos.

```
HWhomework()
load($homework_id)
loadWithStudents($homework_id)
getRecipients($homework)
getId()
getClassId()
getStudents()
isFinished()
getUploads()
setClassStudents()
setClassId($class_id)
setTitle($title)
setSumary($sumary)
setGroupsLimit($gl)
setPublicFiles($pf)
setPublicGrades($pg)
setEnd($end)
create ()
getAdmins()
getAllUsers($homework_id)
getUserInvitations($user)
getGroups()
getComments()
```

9. **invitation.class.php**: Arquivo que contém as seguintes funções para manipulação de convites.

```
Hwinvitation()
setGroupIdFrom($group_id)
setUserIdTo($user_id)
setWhen($when)
send()
verifyInvitation($from, $user)
```

10. **menu.class.php**: Arquivo que contém a classe Menu para criação de menu.

```
menu ()
extractMenu($user_type, $generic_menu)
getMenu ($user)
```

11. **message.class.php**: Arquivo que contém as seguintes funções para manipulação das mensagens exibidas na ferramenta.

```

Message()
setMessage($message)
isSeted()
setXslFile($file)
setXmlFile($file)
throw()
throwMessage($user, $message)
display()

```

12. **session.class.php**: Arquivo que contém as seguintes funções para gerenciamento de sessão.

```

HWsession($session_id)
createSession($user_id, $exibition, $ip_address, $master_session)
terminateSession($sess_id, $end_cause)
getUserId()
getUser()
getExibition($sess_id)
validateSession($complete_sess_id, $ip)
getSessionId(){
checkSession ()

```

13. **template.class.php**: Arquivo que contém as seguintes funções para manipulação de templates.

```

Template()
doTransformation($xml, $xsl)

```

14. **upload.class.php**: Arquivo que contém as seguintes funções para realização de uploads.

```

Upload()
store()
getId()
load($upload_id)
downloadFile()
getFile()
getName()
getSize()
getOwner()
getFileType()
getFileDate()
getHomeworkId()
setHomeworkId($homework_id)
setOwner($owner)
getOwnerId()
getGroupId()
setFile($file)
setFileType($type)

```

setGroup(\$group)

15. **user.class.php**: Arquivo que contém as seguintes funções para manipulação e gerenciamento de usuários.

Hwuser()  
load(\$user\_id)  
loadUpload(\$homework)  
loadClassesId()  
changePassword(\$new\_password)  
loadGrades()  
getRecipients(\$homework)  
getId()  
getName()  
getType()  
getEmail()  
getNumber()  
setId (\$user\_id)  
setName (\$name)  
setType (\$type)  
setEmail (\$email)  
setClasses (\$class\_id)  
setNumber (\$number)  
emailToId (\$email)  
checkKeyFields ()  
validateEmail(\$email)  
validateNumber(\$number)  
getStudentsWithoutGroup(\$homework)  
create()  
includeInClasses()  
includeInClass(\$user, \$class\_id)  
canBeCreated()  
getRandomicPassword()  
getAllUsers()  
getUsersInNoneClass()  
getAllMyClasses()  
removeGrades()  
removeGroups()  
canAccessHomework (\$homework)  
canDoUpload(\$homework)  
canDoDownload(\$upload\_id)  
checkLogin(\$login, \$password)  
getGroup()  
removeFromClass(\$class\_id)  
isInClass(\$class\_id)  
updateEmail(\$new\_email)  
updateNumber(\$new\_number)



```
updateName($new_name)
```

**class admin extends Hwuser**

```
create($user)
remove($user)
leaveClass($user, $class)
joinClass($user, $class)
blessAdmin($user)
canEditUsers()
canEditGrades($homework)
removeFromClass($user, $class_vet)
getRecipients($homework)
```

**class student extends HWuser**

```
changeType()
canCreateGroup($homework)
inviteUsersToGroup($group, $users)
createGroup($name, $homework_id)
isInGroup($group)
isInOneGroup($homework)
rejectInvitation($group_id)
getGroup($homework)
leaveGroup($group)
getUpload($homework)
updateGrade($homework, $grade)
canEditGroup($homework)
```

16. **util.class.php**: Arquivo que contém as seguintes funções para manipulação de senha.

```
Util()
filterPostedList($name)
newPasswordRequest($user)
changePassword($user, $request_code)
```

#### 4.2.7. Diretório xml

1. **class\_added.xml**: Template para geração do arquivo xml que contém os dados para exibição da turma recentemente criada.
2. **class\_removed.xml**: Template para geração do arquivo xml que contém os dados para exibição das informações da turma recentemente removida.
3. **comment\_added.xml**: Template para geração do arquivo xml que contém os dados para exibição das informações do comentário após sua criação.
4. **confirm\_remove\_class.xml**: Template para geração do arquivo xml que contém os dados para exibição das informações sobre exclusão de turma.
5. **confirm\_remove\_user.xml**: Template para geração do arquivo xml que

- contém os dados para exibição da página para exclusão de um usuário.
6. **doubtful\_user.xml**: Template para geração do arquivo xml que contém os dados para exibição das informações para um usuário com o e-mail já cadastrado.
  7. **edit\_grades.xml**: Template para geração do arquivo xml que contém os dados para exibição da página contendo informações sobre status do trabalho ou edição de notas.
  8. **edit\_profile.xml**: Template para geração do arquivo xml que contém os dados para exibição do perfil do usuário.
  9. **edit\_user.xml**: Template para geração do arquivo xml que contém os dados para exibição da página de edição de um usuário.
  10. **error.xml**: Template para geração do arquivo xml que contém os dados para exibição da mensagem de erro.
  11. **group\_added.xml**: Template para geração do arquivo xml que contém os dados para exibição das informações do grupo recentemente criado.
  12. **header.xml**: Template para geração do arquivo xml que contém os dados para exibição do cabeçalho da página contendo o menu, nome e e-mail do usuário.
  13. **homeworks\_list.xml**: Template para geração do arquivo xml que contém os dados para exibição da lista de trabalho.
  14. **index.xml**: Template para geração do arquivo xml que contém os dados para exibição da página inicial da ferramenta Homework.
  15. **invite\_list.xml**: Template para geração do arquivo xml que contém os dados para exibição da lista de usuários a serem convidados.
  16. **message.xml**: Template para geração do arquivo xml que contém os dados para exibição das páginas de mensagens.
  17. **new\_class.xml**: Template para geração do arquivo xml que contém os dados para exibição da página para criação de turma.
  18. **new\_group.xml**: Template para geração do arquivo xml que contém os dados para exibição da página para criação do nome do grupo.
  19. **new\_homework.xml**: Template para geração do arquivo xml que contém os dados para exibição da página para criação de novo trabalho.
  20. **new\_user.xml**: Template para geração do arquivo xml que contém os dados para exibição da página para adição de um novo usuário.
  21. **sent\_invitations.xml**: Template para geração do arquivo xml que contém os dados para exibição das informações do grupo recentemente criado.
  22. **show\_classes.xml**: Template para geração do arquivo xml que contém os dados para exibição da lista de turmas existentes.
  23. **show\_homework.xml**: Template para geração do arquivo xml que contém os dados para exibição das opções de envio de arquivo, manipulação de grupo e adição de comentários.
  24. **show\_users.xml**: Template para geração do arquivo xml que contém os dados para exibição da lista de usuários cadastrados.
  25. **uploaded.xml**: Template para geração do arquivo xml que contém os dados para exibição das informações de um arquivo recentemente submetido pelo aluno.

26. **user\_removed.xml**: Template para geração do arquivo xml que contém os dados para exibição das informações do usuário excluído.

#### 4.2.7.1. Subdiretório skel

1. **class.xml**: Template para geração do arquivo xml que contém os dados para exibição das informações de turmas existentes.
2. **comment.xml**: Template para geração do arquivo xml que contém os dados para exibição das informações dos comentários adicionados.
3. **error.xml**: Template para geração do arquivo xml que contém os dados de uma mensagem de erro.
4. **grade.xml**: Template para geração do arquivo xml que contém os dados para edição das notas.
5. **group.xml**: Template para geração do arquivo xml que contém os dados para exibição das informações de um grupo e de seus membros.
6. **header.xml**: Template para geração do arquivo xml que contém os dados para exibição do cabeçalho da página.
7. **homework.xml**: Template para geração do arquivo xml que contém os dados para exibição informações sobre um trabalho de acordo com o usuário logado e status do trabalho.
8. **invitation.xml**: Template para geração do arquivo xml que contém os dados para exibição de um convite.
9. **menu.xml**: Template para geração do arquivo xml que contém os dados para exibição do menu principal.
10. **message.xml**: Template para geração do arquivo xml que contém os dados de uma mensagem.
11. **submenu.xml**: Template para geração do arquivo xml que contém os dados para exibição do submenu.
12. **upload.xml**: Template para geração do arquivo xml que contém os dados para exibição das informações de um arquivo submetido pelo aluno.
13. **user.xml**: Template para geração do arquivo xml que contém os dados para exibição das informações dos usuários.

#### 4.2.8. Diretório xsl

1. **adm\_groups\_view.xsl**: Contém informações dos grupos visualizadas pelo administrador.
2. **adm\_students\_view.xsl**: Contém informações dos alunos sem grupo visualizadas pelo administrador.
3. **class\_added.xsl**: Exibe informações da turma recentemente criada.
4. **class\_removed.xsl**: Exibe as informações da turma recentemente removida.
5. **comment\_added.xsl**: Exibe as informações do comentário após sua criação.
6. **comment\_form.xsl**: Exibe o formulário para adição de um comentário.
7. **comment\_view.xsl**: Exibe as informações do comentário adicionado.

8. **confirm\_remove\_class.xml**: Exibe informações sobre a exclusão de turma.
9. **confirm\_remove\_user.xml**: Exibe a página para exclusão de um usuário.
10. **doubtful\_user.xml**: Exibe as informações para um usuário com o e-mail já cadastrado.
11. **edit\_grades.xml**: Exibe a página contendo informações sobre status do trabalho ou edição de notas.
12. **edit\_profile.xml**: Exibe o perfil do usuário.
13. **edit\_user.xml**: Exibe a página de edição de um usuário.
14. **error.xml**: Exibe uma mensagem de erro.
15. **group\_added.xml**: Exibe as informações do grupo recentemente criado.
16. **header.xml**: Exibe o cabeçalho da página contendo o menu, nome e e-mail do usuário.
17. **homework.xml**: Exibe informações sobre o trabalho de acordo com o usuário logado e status do trabalho.
18. **homeworks\_list.xml**: Exibe a lista de trabalhos.
19. **index.xml**: Exibe a página inicial da ferramenta Homework.
20. **invite\_list.xml**: Exibe a lista de usuários a serem convidados.
21. **menu.xml**: Exibe o menu principal
22. **message.xml**: Exibe uma mensagem informando o resultado de determinada ação.
23. **new\_class.xml**: Exibe a página de criação de turma.
24. **new\_group.xml**: Exibe a página para criação do nome do grupo.
25. **new\_homework.xml**: Exibe a página para criação de novo trabalho.
26. **new\_test.xml**: Exibe a página para criação de nova prova.
27. **new\_user.xml**: Exibe a página para adição de um novo usuário.
28. **password\_request.xml**: Exibe a página de confirmação de alteração de senha.
29. **sent\_invitations.xml**: Exibe as informações do grupo recentemente criado.
30. **show\_classes.xml**: Exibe a lista de turmas existentes.
31. **show\_homework.xml**: Exibe as opções para envio de arquivo, manipulação de grupo e adição de comentários, de acordo com o usuário logado e status do trabalho.
32. **show\_users.xml**: Exibe a lista de usuários cadastrados.
33. **submenu.xml**: Exibe o submenu.
34. **upload.xml**: Exibe informações de um arquivo submetido pelo aluno.
35. **uploaded.xml**: Exibe a página contendo informações de um arquivo recentemente submetido pelo aluno.
36. **user\_removed.xml**: Exibe informações do usuário excluído.
37. **users\_added.xml**: Exibe informações do usuário recentemente criado.

#### 4.2.8.1. Subdiretório errors

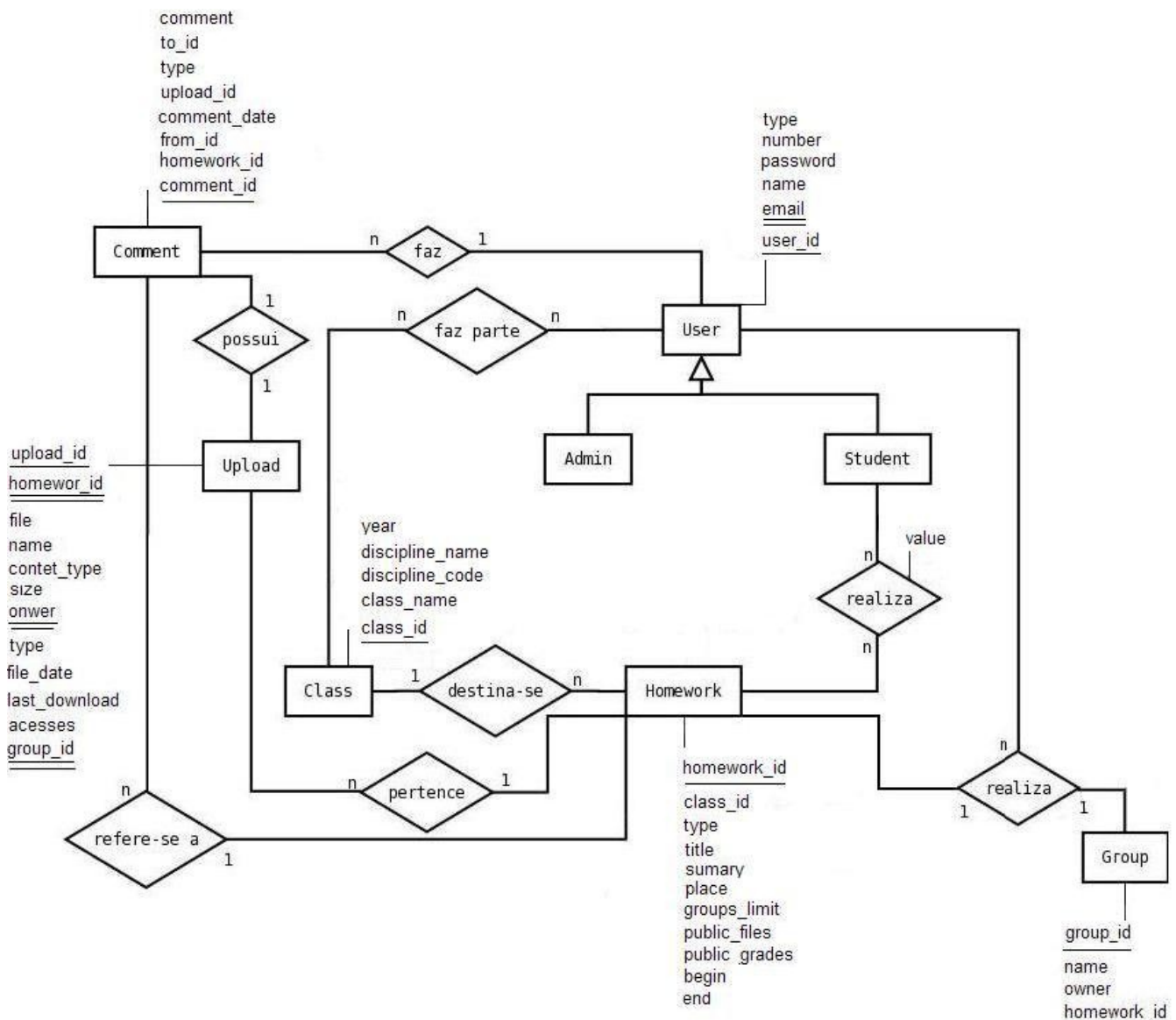
1. **errors.php**: contém todas as mensagens de erros exibidas na ferramenta.

#### 4.2.8.2. Subdiretório messages

1. **messages.php**: contém todas as mensagens exibidas na ferramenta.

### 4.3 Diagrama Entidade Relacionamento

A modelagem do Banco de Dados de Homework foi realizada conforme mostra o Diagrama Entidade-Relacionamento na Figura 2.



**Figura 2.** Modelo de Dados de Homework.

#### 4.4 Esquema Relacional do Banco de Dados de Homework

**Classes** = {class\_id, class\_name, discipline\_code, discipline\_name, year}

**Comments** = {comment\_id, homework\_id, from\_id, comment\_date, upload\_id, type, to\_id, comment}

**Groups** = {group\_id, name, owner, homework\_id}

**Homeworks** = {homework\_id, class\_id, type, title, summary, place, groups\_limit, public\_files, public\_grades, begin, end}

**Uploads** = {upload\_id, homework\_id, file, name, content\_type, size, owner, type, file\_date, last\_download, accesses, group\_id}

**Users** = {user\_id, email, name, password, number, type}

**Grades** = {user\_id, homework\_id, value}

**Invitations** = {group\_id, user\_id\_to, invite\_when}

**Password\_requests** = {request\_id, user\_id, request\_code, request\_date}

**Sessions** = {session\_id, master\_session\_id, user\_id, begin, end, active, end\_cause, ip\_address, exhibition}

**User\_class** = {user\_id, class\_id}

**User\_group\_homework** = {user\_id, homework\_id, group\_id}

### 5. Instalação de Homework

A ferramenta Homework está disponível para *download* no repositório Subversion de ferramentas em desenvolvimento do projeto SAFE (*Software Engineering Available For Everyone*). Esta seção mostra passo-a-passo como instalar e configurar uma instância desta ferramenta.

## Requisitos:

- PHP 4 (o desenvolvimento foi realizado com a versão 4.3.10 e nenhum teste foi realizado com outra versão)
- PEAR:DB (pode ser obtido em <http://pear.php.net>)
- MYSQL (o desenvolvimento foi realizado com a versão 4.1.10 e nenhum teste foi realizado com outra versão)

## Passo a passo:

### (1. ) Fazer checkout do svn.

Para realizar o checkout use o comando:

```
$ svn co http://safe.icmc.usp.br/svn/homework
```

```
usuário: anonymous
```

```
senha: anonymous
```

### (2. ) Criar uma base de dados no mysql.

É necessário criar uma base de dados para as tabelas do Homework, para isso deve-se executar os seguintes comandos (como usuário root do Mysql):

```
mysql> create database homework;
```

```
mysql> grant all privileges on homework.* to homework@localhost  
identified by 'uma_senha_bem_dificil';
```

```
mysql> flush privileges;
```

Desta forma será criada uma base de dados chamada *homework*, o usuário *homework* terá todos os privilégios sobre ela e sua senha para acessar a base será *uma\_senha\_bem\_dificil*.

### (3. ) Configurar os parâmetros do homework para acessar a base de dados.

Editar o arquivo `includes/config_hw.php` para inserir as informações da base de dados:

```
username = homework  
host = localhost  
database = homework  
senha = uma_senha_bem_dificil
```

#### (4. ) Criar tabelas no banco de dados.

Executar o arquivo chamado setup.php com o comando:

```
$ php setup.php
```

Seguindo estes passos a instalação estará completa. Será criado um primeiro usuário cujo login é `admin@admin` e a senha é `admin`.

## 2. 6. Manual do usuário

Dentro do contexto do desenvolvimento de trabalhos práticos foi possível identificar dois tipos de usuários:

- Administradores: Estes usuários (professor e/ou monitor) são responsáveis por:
  - criar e remover outros usuários,
  - criar e remover turmas de estudantes,
  - adicionar comentários,
  - criar e editar trabalhos.
  - avaliar os trabalhos e editar as notas na ferramenta.
- Estudantes: Estes usuários são os que efetivamente realizam os trabalhos práticos, podem formar sua equipe (grupo) de desenvolvimento e editá-la a qualquer momento durante a execução do trabalho proposto. Além disso, podem enviar o trabalho proposto ao professor/monitor e adicionar comentários.

### 6.1 Manual do administrador

O administrador, que na maior parte das vezes é um professor ou monitor de disciplina, deve iniciar o uso da ferramenta criando a **turma** para a qual deseja



gerenciar a entrega dos trabalhos práticos. A seguir, deve proceder o **cadastramento dos alunos** que irão ser incluídos à turma. Caso o aluno já esteja cadastrado, basta o professor alocá-lo à turma em questão. No decorrer do curso, o professor pode então **divulgar os trabalhos** a serem realizados, estipulando seu enunciado, datas e eventuais acertos para que todos os alunos tenham condições e esclarecimentos sobre a tarefa. Após o prazo de entrega dos trabalhos, o professor/monitor pode avaliar os trabalhos entregues e lhes atribuir notas.

### 6.1.2. Criar turma

Para criar uma turma, devem ser preenchidos os campos “Código da disciplina”, “Nome da disciplina”, “Turma” e o “Ano”. Na Figura 3 é apresentada a página de criação de turmas.

A imagem mostra a interface de usuário do sistema "Homework". No topo, há uma barra verde com o texto "Sistema de gerenciamento das tarefas de disciplinas". Abaixo, no canto superior esquerdo, estão as credenciais de login: "Usuário: Administrador" e "Email: administrador@gmail.com". No canto superior direito, o nome do sistema "Homework" é exibido em uma fonte grande e estilizada. Uma barra de navegação contém os links: "Trabalhos", "Editar usuários", "Turmas", "Ajuda", "Meu Perfil" e "Logout". O formulário principal, intitulado "Nova turma", está centralizado e contém os seguintes campos de entrada: "Codigo da disciplina", "Nome da disciplina", "Turma" e "Ano". Abaixo do campo "Ano" há um botão "Criar".

**Figura 3.** Página de Homework que apresenta o formulário para criação de turma

### 6.1.3. Adicionar usuário

Para adicionar um usuário, devem ser preenchidos os campos “Nome”, “E-mail”, “Número USP”. Além disso, deve ser escolhido o tipo de usuário (Estudante ou Administrador) e a turma em que o usuário será adicionado. Na Figura 4 é possível visualizar a página de adição de um usuário.

Sistema de gerenciamento das tarefas de disciplinas

Usuário: Administrador  
Email: administrador@gmail.com

Homework

Trabalhos Editar usuários Turmas Ajuda Meu Perfil Logout

Novo usuário

Nome

E-mail

Número USP

Tipo

Turma

**Figura 4.** Página de Homework que apresenta o formulário para adição de usuário

#### 6.1.4. Editar usuário

Para alterar os dados de um usuário já existente, basta preencher os campos “Nome”, “E-mail”, “Numero USP” que se deseja modificar e/ou selecionar o tipo de turma em que o usuário será incluído. Na Figura 5 é possível visualizar a página de edição de um usuário.

Sistema de gerenciamento das tarefas de disciplinas

Usuário: Administrador  
Email: administrador@gmail.com

Homework

Trabalhos Editar usuários Turmas Ajuda Meu Perfil Logout

Editar Usuário

Aluno teste

Nome:

aluno\_teste@gmail.com

Email:

01

NUSP:

Turmas do usuario:

◆ sce012 | Test-Disciplina | 01

Incluir usuario na disciplina:

Selecione uma disciplina

**Figura 5.** Página de Homework que apresenta o formulário para edição de usuário

### 6.1.5. Criar trabalho

Para criar um trabalho, deve ser escolhida a turma em que o trabalho será realizado. Além disso, cada trabalho deve possuir um título, uma descrição das atividades que devem ser cumpridas, um limite para o número de integrantes dos grupos, se este for realizado em grupo, a data limite para a entrega dos arquivos contendo o trabalho, e as opções de tornar público ou não o acesso as notas e aos arquivos enviados depois de encerradas as atividades do trabalho. Na Figura 6 é possível visualizar a página de criação de trabalhos.

A imagem mostra a interface de usuário do sistema "Homework" para a criação de um novo trabalho. No topo, há uma barra de navegação com o texto "Sistema de gerenciamento das tarefas de disciplinas" e o nome do sistema "Homework". Abaixo, uma barra de usuário indica "Usuário: Administrador" e "Email: administrador@gmail.com". Um menu de navegação contém os itens "Trabalhos", "Editar usuários", "Turmas", "Ajuda", "Meu Perfil" e "Logout". O formulário principal, intitulado "Novo Trabalho", contém os seguintes campos: "Turma:" com uma lista suspensa selecionando "sce558 | Hipemidia | 1", "Título:" com um campo de texto, "Descrição:" com uma área de texto grande, "Encerramento:" com um campo de data/hora no formato (DD/MM/AAAA hh:mm), duas opções de caixa de seleção para "Tornar pública as notas dos trabalhos encerrados" e "Tornar público os arquivos dos trabalhos encerrados", e um campo para "Numero máximo de integrantes de um grupo:". Um botão "Enviar" está localizado na base do formulário.

**Figura 6.** Página de Homework que apresenta o formulário para criação de trabalhos

### 6.1.6. Status do trabalho e edição de nota

Na opção "Status do trabalho" é possível verificar os alunos sem grupo, os alunos com grupo e os arquivos já enviados.

As notas só poderão ser editadas após o encerramento do prazo de entrega de trabalhos. Com isso, a opção "Editar Notas" estará disponível no lugar de "Status do Trabalho". Assim que as notas forem editadas, elas já estarão disponíveis para visualização dos estudantes.

## **6.2 Manual do estudante**

O estudante, ao efetuar login na ferramenta, pode visualizar a descrição dos trabalhos já finalizados ou a serem realizados. Para saber detalhes de um trabalho, o estudante deve clicar sobre o **Id** do trabalho.

Na página de cada trabalho estão presentes informações relevantes como o título do trabalho, data de criação e encerramento, nome da turma, dos professores, dos monitores e uma pequena descrição sobre o trabalho. Além disso, estão presentes opções para manipulação do trabalho, como: envio de arquivos de trabalho, manipulação de grupos e adição de comentários. Na Figura 7 é possível visualizar a página de trabalhos.

Sistema de gerenciamento das tarefas de disciplinas

Usuário: Carlos Alberto  
Email: alunoteste@grad.icmc.usp.br

Homework

Trabalhos Ajuda Meu Perfil Logout

## XML

(criado em 05/09/2006 21:27)

**Trabalho Prático**

**Turma: sce012 | Test-Disciplina | 01**

Professores:  
Andre P. Freire <professor@icmc.usp.br>

**Descrição do Trabalho:**

Elaborar um documento XML. Máximo 2 alunos por grupo.

**Encerramento: 05/09/2006 00:00**

Envio

**Meu grupo**

Obs: Os grupo podem conter no máximo 2 elementos

**Comentários**

Inserir novo comentário

**Selecione o Destinatário**

☐ Professores

☐ Aluno: Carlos Alberto

Anexo

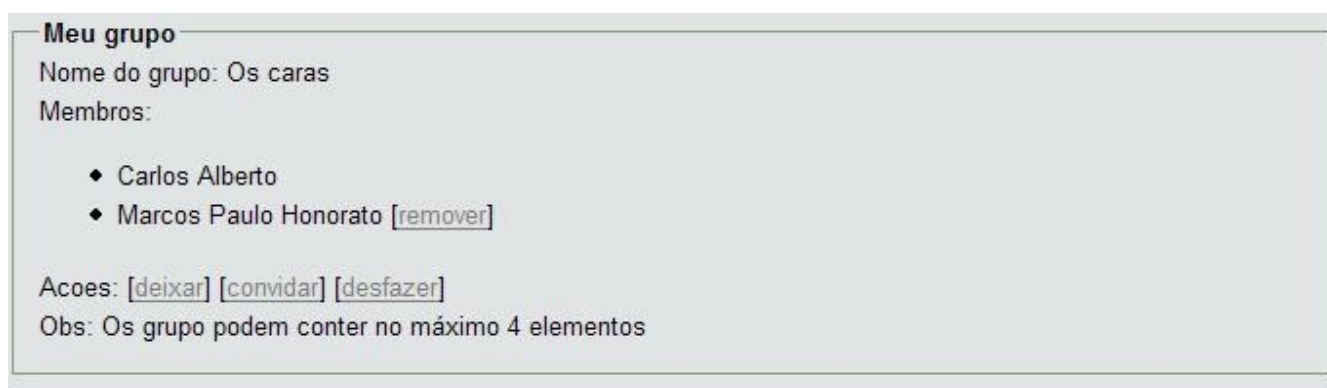
**Figura 7.** Página de Homework que apresenta os trabalhos a serem realizados

### 6.2.1. Criar grupo

O trabalhos podem ser realizados individualmente ou em grupo. Os grupos são criados e manipulados pelos estudantes, sendo que cada estudante só poderá fazer parte de apenas um grupo. Além disso, cada grupo está sempre relacionado a um

trabalho prático, havendo a necessidade de reconfigurar os grupos para os próximos trabalhos a serem executados.

Para criar um grupo, basta clicar na opção “Criar Grupo”, em seguida deve-se preencher o campo “Nome do Grupo” e finalmente clicar na opção “Criar”. Na Figura 8 é possível visualizar as ações para manipulação de grupo.



**Figura 8.** Página de Homework que apresenta as opções de ação que os usuários podem escolher para composição dos grupos

A ação “deixar” não poderá ser executada pelo criador do grupo. Ela se refere a opção de um dos membros do grupo de não participar mais do grupo.

A ação “convidar” poderá ser executada tanto pelo criador do grupo como pelos membros. Após clicar nessa opção, será exibida uma lista de estudantes a serem convidados. Os estudantes poderão ser convidados respeitando o limite máximo de integrantes por grupo.

A ação “desfazer” exclui o grupo e os arquivos enviados pelos seus membros, logo, deve ser tomado muito cuidado ao executar essa ação. Somente o criador do grupo poderá executar essa ação.

A opção “remover” pode ser realizada apenas pelo criador do grupo. Ela se refere à opção do criador do grupo, de excluir um dos membros do grupo.

### 6.2.2. Enviar trabalho

Durante o prazo de entrega de trabalhos, tanto individuais como em grupo, os estudantes podem enviar seus arquivos contendo seu trabalho. Somente a última versão do trabalho enviado é armazenada no banco de dados, as demais são excluídas.

Após a edição da nota pelo professor, ela já estará disponível para visualização junto com as informações do arquivo submetido. Na Figura 9 é possível visualizar a página contendo as informações de trabalho enviado e nota.




The screenshot shows a web interface titled "Envio" (Submission). Inside, there is a box labeled "Versao Atual" containing the following information:

- Arquivo: XML
- Tamanho: 480 b
- Data de Envio: 05/09/2006 21:34
- Nota: 7.5

**Figura 9.** Página de Homework que apresenta informações de trabalho enviado

### 6.2.3. Adicionar comentário e arquivo

Sempre que desejado, os estudantes podem adicionar comentários para um estudante específico, para um grupo, ou para os professores responsáveis pela aplicação do trabalho, assim como anexar um arquivo ao comentário. Assim que um comentário é adicionado, ele fica disponível na página de visualização do trabalho e um e-mail é enviado para o destinatário, contendo o comentário e o arquivo anexado. Na Figura 10 é possível visualizar a página para adicionar comentário e anexar arquivo.



The screenshot shows a web interface titled "Comentários" (Comments). It includes a text area for "Inserir novo comentário" (Insert new comment). To the right, under "Selecione o Destinatário" (Select the Recipient), there are three radio buttons: "Professores", "Grupo:" (with a dropdown menu showing "Os caras"), and "Aluno:" (with a dropdown menu showing "Carlos Alberto"). Below the text area, there is a label "Anexo" followed by a text input field and a button labeled "Arquivo...". At the bottom left, there is a button labeled "Enviar" (Send).

**Figura 10.** Página de Homework que apresenta opções para adicionar comentário e anexar arquivo

### 6.2.4 Envio de e-mail

Esta funcionalidade possibilita os usuários, mesmo não tendo contato freqüente com a ferramenta, de receberem as informações de todas as movimentações realizadas no trabalho. As ações para as quais foi identificada a necessidade de envio de e-mails foram:

- Criação e alteração de trabalho: Todos os usuários envolvidos com o trabalho devem ser notificados sobre as atualizações de sua tarefa.
- Envio de trabalho: Toda vez que um arquivo contendo o trabalho dos estudantes é enviado, uma confirmação chega por e-mail a todos os membros do grupo.
- Convites para compor um grupo: Sempre que um estudante é convidado para participar de um grupo, uma notificação desse convite é enviada por e-mail.
- Comentários: Toda inserção de comentário é notificada a todos os usuários que poderão acessá-lo.

## 7. Considerações Finais

Esse relatório apresentou os principais documentos que descrevem o projeto e implementação do aplicativo Web Homework, que tem sido desenvolvido no ICMC-USP e tem sido utilizado como apoio ao gerenciamento das práticas dos alunos em diversas disciplinas. Trata-se de um projeto de software livre, e como tal, os códigos fonte de Homework estão disponíveis para que seja utilizados, instalados e evoluídos. Como próximos estudos de pesquisa, o projeto de Homework será alvo de uma evolução no sentido de proporcionar suas funcionalidades na forma de Web Services.