

workflow引擎 API 函数的需求说明

没有得到南威软件的书面许可，禁止任何方式的全部或部分复制。

**Reproduction in whole or in part by any means without written permission of
Linewell software is strictly forbidden.**

文件更改记录

版本号	更改内容提要	编制人/更改人	生效日期	审核人	批准人
V1.0	新建文件	研发中心	2011-6-15		

目录

1. 发送功能	4
2. 意见填写功能的 api	4
3. 退回功能的 api	5
4. 退回上一节点的 api	5
5. 获取当前节点对象的 api	5
6. 获取节点所属流程对象的 api	6
7. 获取当前办理人 id 的 api	6
8. 获取流程已办理节点列表的 api	6
9. 获取当前节点的出口节点列表的 api	6
10. 获取各出口节点审批方式的 api	7
11. 获取各出口节点的办理人/办理部门/办理角色/办理职位的 api	7
12. 获取当前节点按钮列表的 api	7
13. 实现流程直接结束功能的 api	7
14. 获取意见列表的 api	8
15. 获取日志列表的 api	8
16. 获取流程实例列表	8
17. 获取当前节点出口子流程列表	8
18. 发送子流程功能	9
19. 直接进入流程功能	9
20. 获取上一节点 id	9
21. 子流程收回	9
22. 流程办理	10
23. 初始化流程（文档与流程绑定）	10
23. 根据文档 id 获取流程实例列表	11
23. 判断是进行发送还是进行多人办理结束	11

1. 发送功能

```
Map map = new HashMap();
map.put("request", request);
UcapWorkFlow ucWorkFlow = new UcapWorkFlow(map);
ucWorkFlow.sendFlowApi(String docUnid, String instanceUnid, String
auditType, String nextNodeUnid, String transactor); // return
true || false
```

参数说明:

名称	含义
docUnid	文档 ID
instanceUnid	流程实例的标识
auditType	下一节点审批方式 0--单人, 1--单个多人, 2--多人顺序, 3--多人并行, 4--主办阅批
nextNodeUnid	下一节点 id
transactor	接收人员 id, 多值 逗号分隔

2. 意见填写功能的 api

```
包 com.linewell.ucap.workflow.tbl.opinion
OpinionWrapper ow=new OpinionWrapper();
Opinion opinion=new Opinion();
设置 opinion 的属性 (属性见表格)
ow.getOperateSql(opinion); //获得 sql 语句
```

意见的相关属性:

属性名称	属性含义
unid	意见标识
instanceId	流程实例的标识
docId	文档标识
flowId	流程标识
nodeId	节点标识
opinionName	意见名称
opinionContent	意见内容
opinionType	意见类型
transactorId	办理者标识
transactorName	办理者名称
transactorBelongId	办理者所属部门的标识
transactorBelongName	办理者所属部门名称
transactTime	办理时间
state	办理状态 0 : 为未办理完毕的意见, 1: 为

	已办理完毕的意见
isNew	判断是否为新意见 isNew 为 true 的时候，则添加，否则为更新

3. 退回功能的 api

```
Map map = new HashMap();
map.put("request", request);
UcapWorkflow ucWorkflow = new UcapWorkflow(map);
ucWorkflow.returnFlowApi(String docUnid, String instanceUnid, String
nodeUnid, String nodeTransactorids) // return true||false
```

参数说明：

名称	含义
docUnid	文档 ID
instanceUnid	流程实例的标识
nodeUnid	当前节点 id
nodeTransactorids	接收人员 id，多值 逗号分隔

4. 退回上一节点的 api

```
Map map = new HashMap();
map.put("request", request);
UcapWorkflow ucWorkflow = new UcapWorkflow(map);
ucWorkflow.completeFlowApi(docUnid, instanceUnid) // return true||false
```

参数说明：

名称	含义
docUnid	文档 ID
instanceUnid	流程实例的标识

5. 获取当前节点对象的 api

```
Map map = new HashMap();
map.put("request", request);
UcapWorkflow ucWorkflow = new UcapWorkflow(map);
FlowNode node = ucWorkflow.getCurNodeApi(docUnid, instanceUnid);
//返回 com.linewell.ucap.workflow.bean.flow.FlowNode 对象
```

属性名称	属性含义
id	节点 id
name	节点名称
formid	表单 id

auditType	审批方式:0--单人, 1--单个多人, 2--多人顺序, 3--多人并行, 4--主办阅批
-----------	--

6. 获取节点所属流程对象的 api

```
Map map = new HashMap();
map.put("request", request);
UcapWorkFlow ucWorkFlow = new UcapWorkFlow(map);
Flow flow=ucWorkFlow.getFlowConfigApi (docUnid, instanceUnid);
//返回 com.linewell.ucap.workflow.bean.flow.Flow 对象
```

属性名称	属性含义
id	流程 id
name	流程名称
formid	表单 id
belongDept	流程所属部门

7. 获取当前办理人 id 的 api

```
Map map = new HashMap();
map.put("request", request);
UcapWorkFlow ucWorkFlow = new UcapWorkFlow(map);
String value=ucWorkFlow.getFuiHmByName (docUnid, instanceUnid
, com.linewell.ucap.workflow.WorkFlowConstant.UCAP_NODE_TRANSACTOR);
```

8. 获取流程已办理节点列表的 api

```
Map map = new HashMap();
map.put("request", request);
UcapWorkFlow ucWorkFlow = new UcapWorkFlow(map);
String value=ucWorkFlow.getFuiHmByName (docUnid, instanceUnid,
"instance_all_tran_node_unid ");
```

9. 获取当前节点的出口节点列表的 api

通过获取当前节点对象的 api 可以得到 com.linewell.ucap.workflow.bean.flow.FlowNode

```
List transitionsList=flowNode.getTransitions();
```

transitionsList 中每个都是 com.linewell.ucap.workflow.bean.flow.Transition 节点出口对象

Transition 对象属性

属性名称	属性含义
id	出口节点 id

name	出口节点名称
transitionType	节点出口类型:0--普通节点, 1--子流程

10. 获取各出口节点审批方式的 api

获取节点所属流程对象的 api 获取 flow 对象

```
FlowNode node=flow. getNodeById( id)
```

```
String auditType=node.getAuditType();//0--单人,1--单个多人,2--多人顺序,3--多人并行,4--主办  
阅批
```

参数说明:

名称	含义
id	节点 ID

11. 获取各出口节点的办理人/办理部门/办理角色/办理职位的 api

获取节点所属流程对象的 api 获取 flow 对象

```
FlowNode node=flow. getNodeById( id)
```

```
List participantList=flowNode. getParticipants ();
```

participantList 中每个都是 com.linewell.ucap.workflow.bean.flow. Participant 节点参与者对象

Participant 对象属性

属性名称	属性含义
id	参与者 id, 多值逗号分隔
participantType	0--人员, 1--部门, 2--角色, 3--职位, 4--流程角色, 5--动态角色

12. 获取当前节点按钮列表的 api

此接口已提供, 参考现在获取按钮的方法

13. 实现流程直接结束功能的 api

```
Map map = new HashMap();  
map.put("request", request);  
UcapWorkFlow ucWorkFlow = new UcapWorkFlow(map);  
ucWorkFlow. endFlowApi (docUnid, instanceUnid); // return true||false
```

14. 获取意见列表的 api

```
包 com.linewell.ucap.workflow.tbl.opinion
/**
 * @return 意见列表, list 中的元素为 Opinion 对象 java.util.List
 */
OpinionWrapper ow=new OpinionWrapper ();
ow.getOpinionListByDocId(String docUnid, IUcapWorkflowDBUtil db)
```

参数说明:

名称	含义
docUnid	文档标识
db	数据库访问对象, IUcapWorkflowDBUtil 的在应用系统中的实现对象

15. 获取日志列表的 api

可以直接查询 ucap_fw_log 表
log_doc_id 字段为文档 id

16. 获取流程实例列表

```
Map map = new HashMap();
map.put("request", request);
UcapWorkFlow ucWorkFlow = new UcapWorkFlow(map);
List instanceList=ucWorkFlow. getInstanceList(docUnid, " ");
instanceList中每个都是com.linewell.ucap.workflow.bean.InstanceBean对象
```

InstanceBean 对象属性

属性名称	属性含义
instanceUnid	流程实例UNID
flowUnid	流程配置UNID
flowName	流程名称

17. 获取当前节点出口子流程列表

获取当前节点的出口节点列表的 api 可以获取到出口节点列表

节点出口类型为 1 表示子流程

获取到子流程节点的 id 后

通过获取节点所属流程对象的 api 获取 flow 对象

```
FlowNode node=flow. getNodeById( id)// node对象为子流程节点对象
```

```
com.linewell.ucap.workflow.bean.flow.SubFlow subflow= node. getSubFlow();
```

```
String subflowid=subflow.getId();//子流程的id
```


18. 发送子流程功能

```
Map map = new HashMap();
map.put("request", request);
UcapWorkFlow ucWorkFlow = new UcapWorkFlow(map);
ucWorkFlow.sendSubFlowApi(String docUnid, String instanceUnid, String
subFlowUnid, String subNodeUnid, String auditType, String userIds);
// return true||false
```

参数说明:

名称	含义
docUnid	文档 ID
instanceUnid	主流程实例的标识
auditType	下一节点审批方式 0--单人, 1--单个多人, 2--多人顺序, 3--多人并行, 4--主办阅批
subFlowUnid	子流程 id
subNodeUnid	接收的子流程节点 UNID
userIds	节点办理人, 逗号分隔

19. 直接进入流程功能

```
String instanceUnid =ucWorkFlow.saveInstanceFlowDoc(flowUnid, curUnid ,docUnid,
docTitle);// return流程实例id
```

参数说明:

名称	含义
docUnid	文档ID
flowUnid	流程id
curUnid	节点id
docTitle	文档标题, 可以为空

20. 获取上一节点 id

```
String preNodeUnid=ucWorkFlow.getFuiHmByName(docUnid, instanceUnid,
com.linewell.ucap.workflow. WorkflowConstant.UCAP_FLOW_PREVIOUS_NODE_UNID);
```

21. 子流程收回

```
ucWorkFlow.completeFlow(docUnid, instanceUnid, docTitle, "5", subFlowUnid);
```

参数说明:

名称	含义
docUnid	文档ID
instanceUnid	流程实例id

subFlowUnid	<p>子流程实例id</p> <p>子流程实例id获取方法如下：</p> <pre>List subFlowList =ucWorkFlow.getSubFlowCallBackList(docUnid, instanceUnid);</pre> <p>subFlowList中每个都是 com.linewell.ucap.workflow.bean.SubFlowBackInstanceId 对象</p> <p>SubFlowBackInstanceId对象属性</p> <table> <tr> <th>属性名称</th><th>属性含义</th></tr> <tr> <td>subFlowInstanceId</td><td>子流程实例id</td></tr> <tr> <td>subFlowUnid</td><td>子流程配置id</td></tr> <tr> <td>subFlowName</td><td>子流程名称</td></tr> </table>	属性名称	属性含义	subFlowInstanceId	子流程实例id	subFlowUnid	子流程配置id	subFlowName	子流程名称
属性名称	属性含义								
subFlowInstanceId	子流程实例id								
subFlowUnid	子流程配置id								
subFlowName	子流程名称								
docTitle	文档标题，可以为空								

22. 流程办理

```
ucWorkFlow.completeFlow(docUnid, instanceUnid, docTitle, type, "");
```

参数说明：

名称	含义
docUnid	文档ID
instanceUnid	流程实例id
docTitle	文档标题，可以为空
type	<p>为流程办理的不同类型，其中为“1”时为上一节点收回；</p> <pre>type =0 多人办理结束 =1 撤办 =2 本节点收回 =3 阅批办理结束 read_over =5 子流程收回 =6 退回上一节点 =8 子流程退回 =9 收回转办 =12 转办返回办理人</pre>

23. 初始化流程（文档与流程绑定）

```
String instanceUnid =ucWorkFlow.saveInstanceFlowDoc(flowUnid, curUnid, docUnid,
docTitle) //返回流程实例 id
```

获取初始节点标识

```
Map map = new HashMap();
```

```
map.put("request", request);
```

```
UcapWorkFlow ucWorkFlow = new UcapWorkFlow(map);
```

```
List nodeList = ucWorkflow. getFlowBeginNodeList(flowUnid);
Node beginNode = (FlowNode)nodeList.get(0);
curUnid = beginNode. getId();
```

参数说明:

名称	含义
docUnid	文档ID
flowUnid	流程id
curUnid	初始节点id
docTitle	文档标题, 可以为空

23. 根据文档 id 获取流程实例列表

```
HashMap map = new HashMap();
map.put("request", request);
// 先构造对象
UcapWorkFlow ucapWorkFlow = new UcapWorkFlow(map);
List list = ucapWorkFlow.getInstanceList(docUnid, null); //流程实例列表
```

```
if(null!=list && list.size()>0){
    InstanceBean ib = (InstanceBean)list.get(0);
    instanceId = ib.getInstanceUnid(); //取到相应的流程实例id
}
```

23. 判断是进行发送还是进行多人办理结束

```
Map map = new HashMap();
map.put("request", request);
UcapWorkFlow ucWorkflow = new UcapWorkFlow(map);
Boolean result = ucWorkflow. auditTypeSend();
result 为 true 或 false, 为 true 的时候则可进行发送, 为 false 的时候则为多人办理结束
```