南京大学教务系统软件体系结构描述文档

南京大学软件学院 2013年10月

韩旭(121250041)

郭天昊(121250039)

陈俊杉(121250011)

施周利 (121250126)

目录

1	引言	ā ····································	
	1.1	编制目的 • • • • • • • • • • • • • • • • • • •	
	1.2	词汇表 • • • • • • • • • • • • • • • • • • •	
	1.3	参考资料 •••••• 2)
2	产品	占概述 • • • • • • • • • • • • • • • • • • •)
3	逻辑	i视角 •••••• · · · · · · · · · · · · · · · ·)
4	组合	· i视角 •••••• · · · · · · · · · · · · · · · ·	j
	4.1	· · · · · · · · · · · · · · · · · · ·	
	4.2	运行时进程 •••••••••• 6	
	4.3	物理部署 • • • • • • • • • • • • • • • • • • •	
5	接口	· 视角 • • • • • • • • • • • • • • • • • • •	
	5.1	· 旋块的职责 • • • • • • • • • • • • • • • • • • •	
	5.2	用户界面层的分解 ••••••• 10	
	0.2	5.2.1 用户界面层模块的职责 •••••• 12	
		5.2.2 用户界面层模块的接口规范 •••••• 12	
		5.2.3 用户界面模块设计原理 •••••• 12	
	5.3	业务逻辑层的分解 ••••••••••••••••••••••••••••••••••••	
	3.3	5.3.1 业务逻辑层模块的职责 ••••••• 12	
		5.3.2 业务逻辑层模块接口规范 •••••• 12	
	5.4	数据层的分解 ••••••••••••••••••••••••••••••••••••	
	J. 4	5.4.1 数据层模块的职责 ••••••••••••••••••••••••••••••••••••	
		5.4.2 数据层模块的接口规范 •••••• 19	
6	∕≐⊨	3.4.2	
6	百元 6.1	3代用 数据持久化对象・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	
	6.2	数据库表····································	
	n.z	级 1 店	•

1. 引言

1.1 编制目的

本报告详细完成对教务系统的概要设计,达到指导详细设计和开发的目的,同时实现 和测试人员及用户的沟通。

本报告面向开发人员,测试人员及最终用户而编写,是了解系统的导航。

1.2 词汇表

1.3 参考资料

2. 产品概述

参考教务系统用例文档和教务系统需求规格说明文档中对产品的概括描述。教务系统 体系结构采用分层风格。

3. 逻辑视角

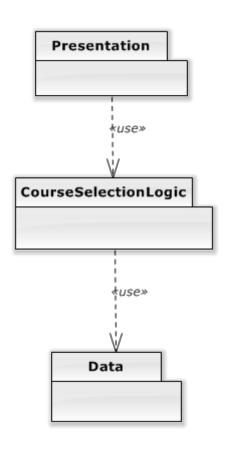


图 1 参考体系结构风格的包图表达逻辑视角

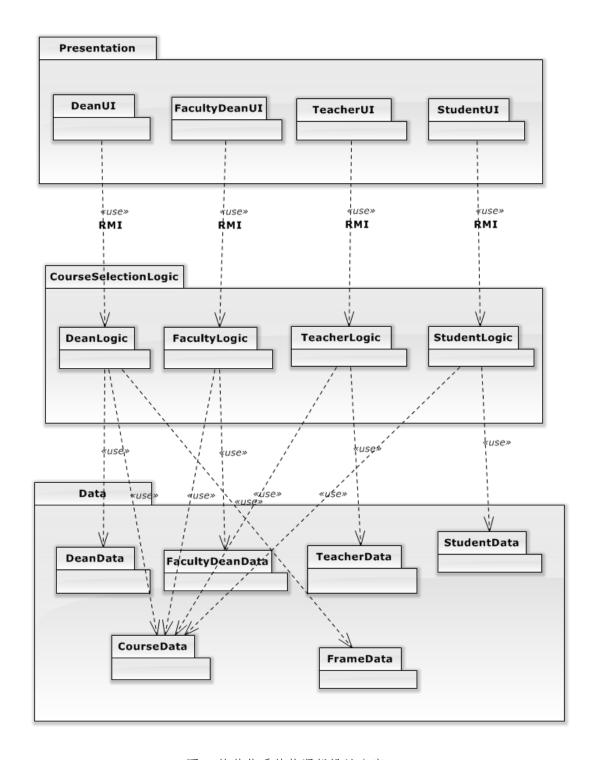


图 2 软件体系结构逻辑设计方案

4. 组合视角

4.1 开发包图

表 1 教务系统的最终开发包设计

开发(物理)包	依赖的其他开发包
com.client.UI.mainUI	loginUI,deanUI,facultyDeanUI,teacherUI,stud entUI,vo
com.client.UI.loginUI	vo

com.client.UI.deanUI	logicService,vo
com.client.UI.facultyDeanUI	logicService,vo
com.client.UI.teacherUI	logicService,vo
com.client.UI.studentUI	logicService,vo
com.server.logic.logicService	Java RMI,po
com.server.logic.deanImpl	com.server.logic.deanImpl.method,po
com.server.logic.facultyDeanImpl	com.server.logic.facultyDeanImpl.method,po
com.server.logic.teacherImpl	com.server.logic.teacherImpl.method,po
com.server.logic.StudentImpl	com.server.logic.StudentImpl.method,po
com.server.logic.deanImpl.method	dataService,po
com.server.logic.facultyDeanImpl.method	dataService,po
com.server.logic.teacherImpl.method	dataService,po
com.server.logic.StudentImpl.method	dataService,po
com.server.data.dataService	po
com.server.data.dataImpl	com.server.data.dataImpl.method,po
com.server.data.dataImpl.method	po,JDBC
com.server.data.dataImpl.method.dean	po,JDBC
com.server.data.dataImpl.method.facultyDean	po,JDBC
com.server.data.dataImpl.method.teacher	po,JDBC
com.server.data.dataImpl.method.student	po,JDBC
com.data.po	
com.data.vo	

教务系统的客户端开发包图如图 3 所示,服务器端的开发包图如图 4 所示

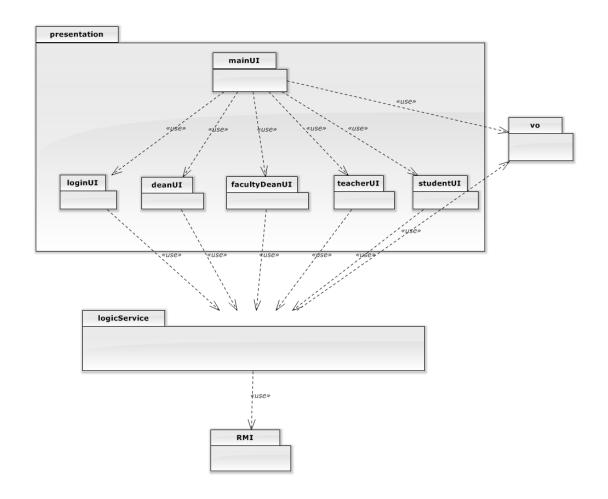


图 3 教务系统客户端开发包图

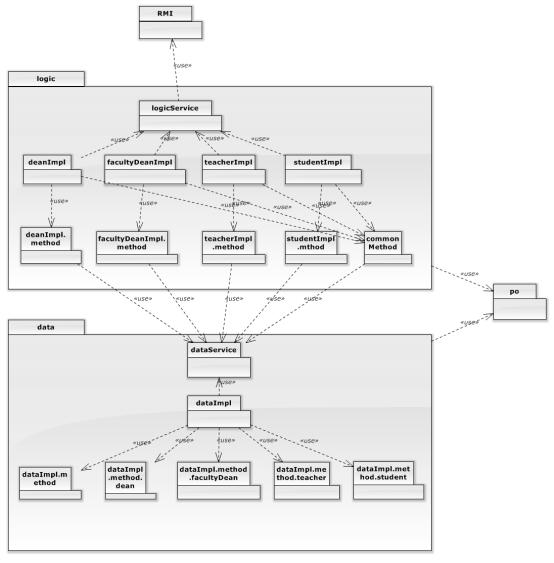


图 4 教务系统服务器端开发包图

4.2 运行时进程

在教务系统中,会有多个客户端进程和一个服务器端进程,其进程图如图 5 所示。结合部署图,客户端进程是在客户端机器上运行,服务器端进程是在服务器机器上运行。

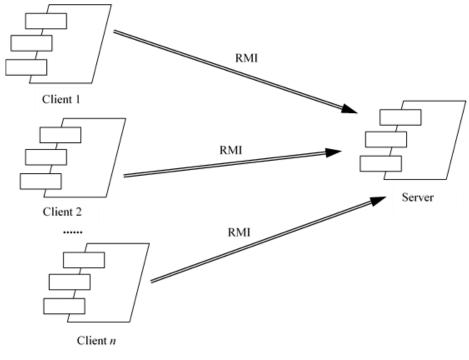


图 5 进程图

4.3 物理部署

教务系统中客户端构件时放在客户端机器上,服务器端构件是放在服务器端机器上。在客户端节点上,还要部署 RMIStub 构件。由于 Java RMI 构件是属于 JDK 6.0 的一部分。所以,在系统 JDK 环境已经设置好的情况下,不需要再独立部署。部署图如图 6 所示。

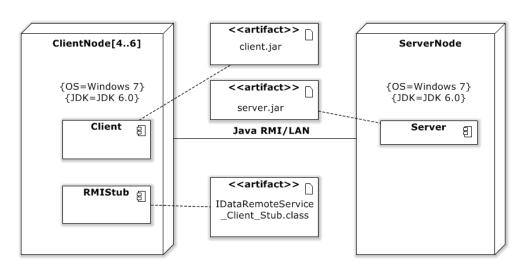


图 6 部署图

5. 接口视角

5.1 模块的职责

客户端模块和服务器模块试图分别如图 7 和图 8 所示。客户端和服务器端各层的职责分别如表 2 和表 3 所示。

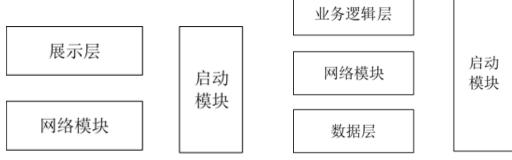


图 7 客户端模块视图

图 8 服务器端模块视图

表 2 客户端各层的职责

	秋 2 谷 /	加甘広的 机贝		
 层			职责	
启动模块		负责初始化网	络通信机制,启动用户界面	
用户界面层		基于窗口的教	务系统客户端用户界面,并	
用厂介围压		对用户的输入	进行响应	
客户端网络模块		利用 Java RMI	I 机制查找 RMI 服务	
	表 3 服务	器端各层的职责		
 层			职责	
启动模块		负责初始化网	络通信机制,启动用户界面	
业务逻辑层		对 RMI 传递过	过来的数据进行业务处理逻辑	
数据层		负责数据的持	久化及数据访问接口	
HI 전 HI 24L CT (전 44 LL		利用 Java RMI	I 机制开启 RMI 服务,注册	
服务器端网络模块		RMI 服务		
表 4 层之间调用的接口				
接口	服务证	周用方	服务提供方	
logicService 客户端展示层			服务器端逻辑层	
dataService 服务器端逻辑			服务器端数据层	

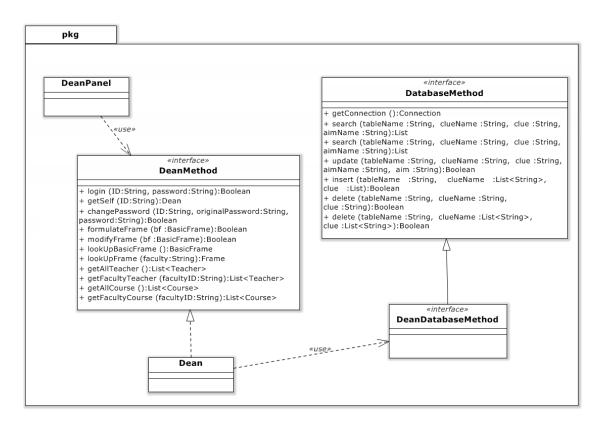


图 9 教务员

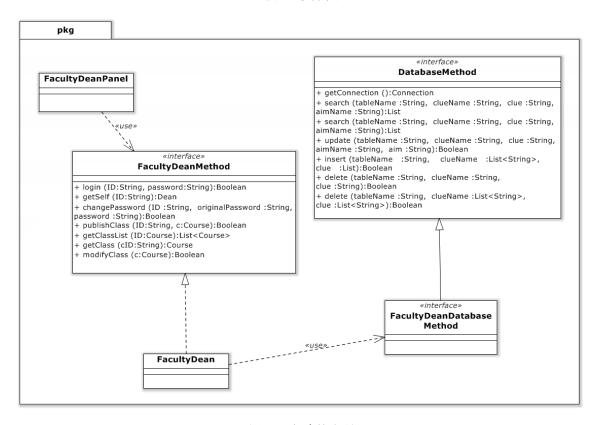


图 10 院系教务员

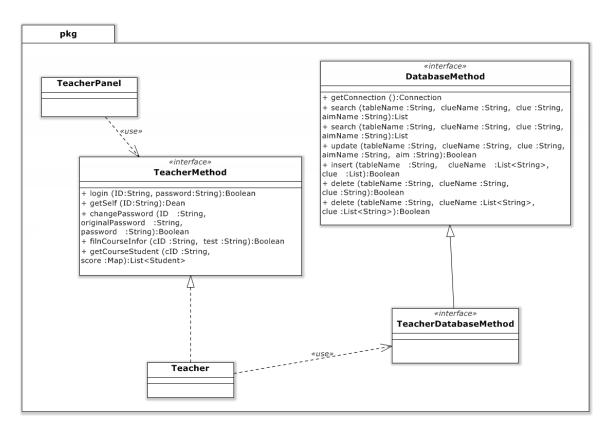


图 11 老师

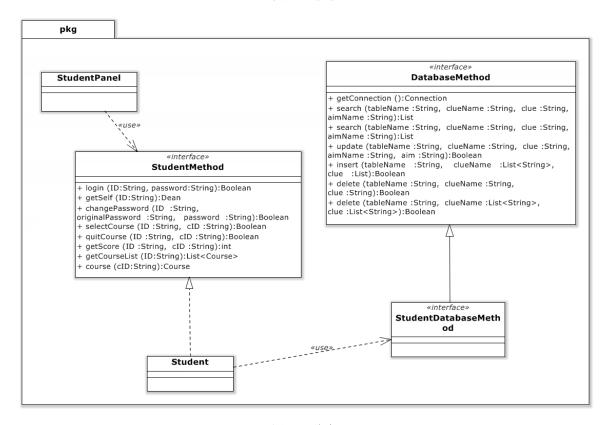
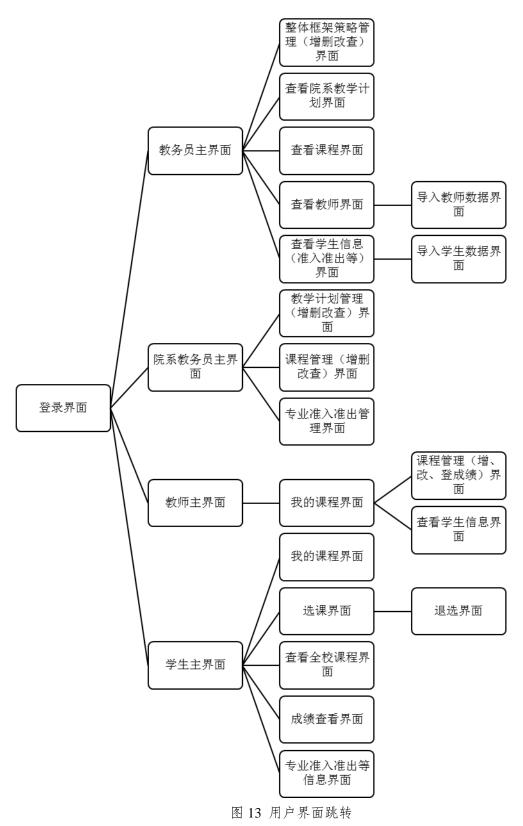


图 12 学生

5.2 用户界面层的分解



用户界面类如图 14 所示。

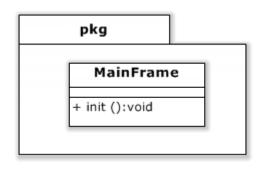


图 14 用户界面类

5.2.1 用户界面层模块的职责

表 5

	7,00	
模块	职责	
MainFrame	界面 Frame,负责界面的显示和界面的跳转	

5.2.2 用户界面层模块的接口规范

表 6

	语法	init(args:String[])
MainFrame	前置条件	无
	后置条件	显示 Frame 和 LoginPanel

表 7

	•= -
服务名	服务
logicService.DeanMethod	教务员页面的业务逻辑接口
logicService.FacultyDeanMethod	院系教务员页面的业务逻辑接口
logicService.TeacherMethod	教师页面的业务逻辑接口
logicService.StudentMethod	学生页面的业务逻辑接口

5.2.3 用户界面模块设计原理

用户界面利用 Java 的 Swing 和 AWT 库来实现。

5.3 业务逻辑层的分解

业务逻辑层包括多个针对界面的业务逻辑处理对象。

5.3.1 业务逻辑层模块的职责

业务逻辑层模块的职责如表8所示。

表 8 业务逻辑层模块的职责

模块	职责
logicService.DeanMethod	负责教务员任务界面所需的服务
logicService.FacultyDeanMethod	负责院系教务员任务界面所需的服务
logicService.TeacherMethod	负责老师任务界面所需的服务
logicService.StudentMethod	负责学生员任务界面所需的服务

5.3.2 业务逻辑层模块接口规范

表 9 DeanMethod 模块接口规范

提供的服务(供接口)				
	语法	public interface DeanMethod extends Remote		
Dean.login	前置条件	用户输入有效的账号和密码。		
	后置条件	根据账号查找是否存在相应的用户,根据账号验证		

		密码是否正确,如果通过验证,则登录成功,返回			
		True 否则,返回 False			
	语法	public Dean getSelf(String ID) throws			
Dean.getSelf		RemoteException;			
Ü	前置条件	学校教务员已经成功登录			
	后置条件	返回该院系教务员信息			
		public boolean changePassword(String ID, String			
	语法	originalPassword,String password)throws			
		RemoteException;			
	前置条件	学校教务员已经登录。且用户输入有效的账号,原密			
Dean. changePassword	加重水川	码和现密码。			
		院系教务员修改密码。根据账号查找是否存在相应			
	后置条件 后置条件	的用户,根据账号验证旧密码是否正确,如果通过验			
	川且赤口	证,则将密码修改为现密码,且返回 True, 否则, 返			
		回 False			
	语法	public boolean formulateFrame(BasicFrame bf)throws			
	归亿	RemoteException;			
D f1-4-E		学校教务员已经登录,且用户根据内容进行了有效			
Dean. formulateFrame	前置条件	的输入			
	己四万加	建立整体框架,如果创建成功,则返回 True, 否则			
	后置条件	返回 False			
	语法	public boolean modifyFrame(BasicFrame bf)throws			
		RemoteException;			
D 116 E	24 W 47 M.	学校教务员已经登录,且用户根据内容进行了有效			
Dean. modifyFrame	前置条件	的输入			
	己四石加	学校教务员修改整体框架,如果修改成功,则返回			
	后置条件	True,否则返回 False			
-) = \2	public BasicFrame lookUpBasicFrame()throws			
	语法	RemoteException;			
Dean.	前置条件	学校教务员已经登录			
lookUpBasicFrame		学校教务员请求查看整体框架,如果整体框架已创			
	后置条件	建,则返回该框架。反之,则系统提示错误			
		public Frame lookUpFrame(String facultyID)throws			
	语法	RemoteException;			
Dean.lookUpFrame	前置条件	学校教务员已经登录,且输入了有效的院系 ID			
_ tumosnopi tumo		教务员查看院系教学计划,根据 ID 找到对应的院			
	后置条件	系,并返回该院系计划,否则系统提示 ID 输入有误			
		public List <teacher> getAllTeacher()throws</teacher>			
	语法	RemoteException;			
Dean.getAllTeacher	前置条件	学校教务员已经成功登录			
Dean.gen III Teachel	后置条件	学校教务员查看全校老师列表,返回一个装有信息			
		子仪叙分贝旦有主仪名师列表, 这回			
Dean.getFacultyTeacher	语法	public List <teacher> getFacultyTeacher(String</teacher>			
Dean.gen acuity reacher	口 亿	public List< reacher> getFaculty reacher(String			

		6 1 TD) 1 D	. P		
24 III 14 M		facultyID)throws Remo	<u> </u>		
前置条			,且输入了有效的院系 ID		
			所有老师,根据 ID 找到对应的		
			行有老师列表,否则系统提示 ID		
		输入有误			
	语法	public List <cou< td=""><td>rse> getAllCourse()throws</td></cou<>	rse> getAllCourse()throws		
		RemoteException;			
Dean.getAllCourse	前置条件	‡ 学校教务员已经成功	登录		
	后置条件	学校教务员查看全校 的列表	课程列表,返回一个装有信息		
	语法	public List <cours< td=""><td>e> getFacultyCourse(String</td></cours<>	e> getFacultyCourse(String		
	后 伝	facultyID)throws Remo	oteException;		
D 4F 14 C	前置条件	‡ 学校教务员已经登录。	, 且输入了有效的院系 ID		
Dean.getFacultyCours	se	学校教务员查看院系	所有老师, 根据 ID 找到对应的		
	后置条件		f有老师列表,否则系统提示 ID		
		输入有误			
) <u></u>	public boolean pu	ublishCourse(Course c)throws		
	语法	RemoteException;	,		
Dean.publishCourse	前置条件	-	学校教务员已登录,正确输入课程信息		
	C = 4 /	发布课程成功,系统返回 true, 否则提示发布失败,			
	后置条件	F 返回 false			
	/王 /十	public boolean m	odifyCourse(Course c)throws		
D 110 G	语法	RemoteException;			
Dean.modifyCourse	前置条件	‡ 学校教务员已登录,	正确查找已发布课程		
	后置条件	井 修改成功,系统返回	修改成功,系统返回 true,否则返回 false		
		需要的服务(需接口)			
	服务名		服务		
DatabaseMethod.getC			得到数据库的服务引用		
		Name,List clueName,List			
clue,List aimName)	` "	,	在数据库中的数据进行搜索		
DatabaseMethod.upda	te(String table	eName.String			
clueName,String clue,		•	更新数据库中信息		
DatabaseMethod.inser					
clueName,List clue)		······································	向数据库中插入条目		
DatabaseMethod.delete(String tableName.List <string></string>					
clueName,List <string< td=""><td></td><td>, ,</td><td>删除数据库中的条目</td></string<>		, ,	删除数据库中的条目		
	表 10 Fa	acultyDeanMethod 模块的	接口规范		
)モバナ	public boolean login(Stri	ng ID,String password) throws		
	语法	RemoteException;	· · · · · · · · · · · · · · · · · · ·		
FacultyDean.login	前置条件	ID,password 符合输入规则			
·	后置条件	系统判断,当 ID 和 password 匹配 return true,反之 return alse;			

	2五八十	public FacultyDean getSelf(String ID) throws		
Engulty Doon got Colf	语法	RemoteException;		
FacultyDean.getSelf	前置条件	一位院系教务员已经登录		
	后置条件	返回该院系教务员信息		
		public boolean changePassword(String ID,String		
	语法	originalPassword,String password) throws		
FacultyDean.		RemoteException;		
changePassword	前置条件	一位院系教务员已经登录		
	后置条件	院系教务员修改密码。传入的		
	川旦小川	原密码若正确 return true,若原密码错误 return false。		
	语法	public boolean publishClass(String ID,Course c) throws		
FacultyDean.	ИПА	RemoteException;		
publishClass	前置条件	一位院系教务员已经登录		
puonsnetass	 后置条件	发布课程,若课程已存在,则发布失败 return false,反		
	川且ボロ	之 return true。		
	语法	public List <course> getClassList(String ID) throws</course>		
FacultyDean.	И1Д	RemoteException;		
getClassList	前置条件	一位院系教务员已经登录		
	后置条件	院系教务员查看本院系课程列表 return		
	语法	public Course getClass(String cID) throws		
FacultyDean.	日	RemoteException;		
getClass	前置条件	一位院系教务员已经登录		
	后置条件	查看任何课程信息		
	语法	Public Boolean modifyClass(Course c) throws		
FacultyDean.		RemoteException;		
modifyClass	前置条件	一位院系教务员已经登录		
	后置条件	修改课程信息		
		需要的服务(需接口)		
	服务名	服务		
DatabaseMethod.getC	Connection	得到数据库的服务引用		
DatabaseMethod.searc	ch(String tab	leName,List clueName,List 在数据库中的数据进行搜索		
clue,List aimName)		1. 数加井中的数加边行设系		
DatabaseMethod.upda	ate(String	tableName,String 更新数据库中信息		
clueName,String clue,String aimName,String aim) 更新致据库中信息				
DatabaseMethod.insert(String tableName,List <string> 向数据库中插入条目</string>				
clueName,List clue) 阿剱加岸中加八宋日				
DatabaseMethod.delete(String tableName,List <string> 删除数据库中的条目</string>				
clueName,List <string< td=""><td>> clue)</td><td>加州赤 致加</td></string<>	> clue)	加州赤 致加		
表 11 TeacherMethod 模块的接口规范				
	,=	public boolean login(String ID,String password)		
TeacherMethod.lo	gin	throws RemoteException;		
	前置	置条件 无		

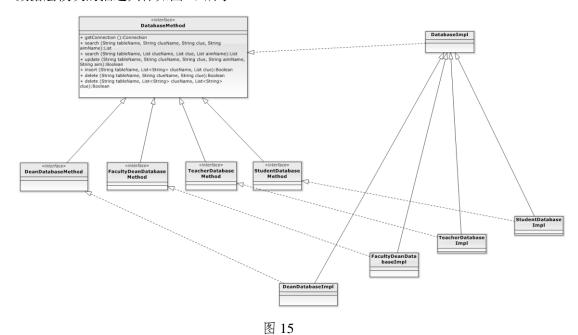
	ı			
	后置条件	系统判断,当 ID 和 password 匹配 return true,反		
	/I <u> </u>	之 return false;		
	语法	public Teacher getSelf(String ID) throws		
TeacherMethod.getSelf		RemoteException;		
reachemicanou.getsen	前置条件	一位教师已经登录		
	后置条件	返回该院系教务员信息		
		public boolean changePassword(String ID,String		
	语法	originalPassword,String password) throws		
TeacherMethod.		RemoteException;		
changePassword	前置条件	一位教师已经登录		
changer assword		教师修改密码。传入的		
	后置条件	原密码若正确 return true, 若原密码错误 return		
		false。		
	2五八十	public boolean filnCourseInfor(String cID,String		
T 1 M 1 1	语法	text) throws RemoteException;		
TeacherMethod.	前置条件	一位老师已经登录		
filnCourseInfor	己四夕加	老师填写课程信息, 若成功 return true, 反之 return		
	后置条件	false		
)T.).	public List <student> getCourseStudent(String cID)</student>		
	语法	throws RemoteException;		
TeacherMethod.	前置条件	一位老师已经登录		
getCourseStudent		老师查看选课学生,返回选择该课程的学生列		
	后置条件	表。		
) T) I.	public boolean recordScore(String cID,Map score)		
	语法	throws RemoteException;		
TeacherMethod.recordScore	前置条件	一位老师已经登录		
	后置条件	老师录入成绩		
	 需要的	 		
		服务		
DatabaseMethod.getConnecti	on	得到数据库的服务引用		
DatabaseMethod.search(Strin		ne List		
clueName,List clue,List aimN	Jame)	在数据库中的数据进行搜索		
DatabaseMethod.update(Strin				
tableName,String clueName,		String 更新数据库中信息		
aimName,String aim)				
DatabaseMethod.insert(String	7			
tableName,List <string> cluel</string>		ue) 向数据库中插入条目		
DatabaseMethod.delete(String		,		
tableName,List <string></string>		删除数据库中的条目		
clueName,List <string> clue)</string>				
DatabaseMethod.getConnecti	on	得到数据库的服务引用		
		tudentMethod 模块接口		
		加务(供接口)		
)近 六日	の以の 、「内」女山 /		

	语法	public boolean login(String ID,String password)
	-,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	throws RemoteException;
StudentMethod.login	前置条件	用户输入有效的账号和密码。
Studentivictiod.10gm		根据账号查找是否存在相应的用户,根据账号
	后置条件	验证密码是否正确,如果通过验证,则登录成
		功,返回 True 否则,返回 False
	\ \.	public Student getSelf(String ID) throws
	语法	RemoteException;
StudentMethod.getSelf	前置条件	一位学生已经登录
	后置条件	返回该学生信息
		public boolean changePassword(String ID,String
	语法	originalPassword,String password) throws
		RemoteException;
		一位学生已经登录,且用户输入有效的账号,
StudentMethod.	前置条件	原密码和现密码。
changePassword		学生修改密码。根据账号查找是否存在相应的
		用户,根据账号验证旧密码是否正确,如果通
	后置条件	过验证,则将密码修改为现密码,且返回 True,
		否则,返回 False
	语法	public boolean selectCourse(String ID,String
		cID) throws RemoteException;
	前置条件	一位学生已经登录,且用户输入有效地学生
StudentMethod. selectCourse		ID 和课程 ID
		学生选择课程,根据学生 ID 查找是否存在相
	后置条件	应的用户,根据课程 ID 查找是否存在相应的
	77 11274111	课程,如果通过验证,则学生选课成功,且返
		回 True,否则,返回 False
	语法	public boolean quitCourse(String ID,String cID)
	ИТА	throws RemoteException;
		 一位学生已经登录,且用户输入有效地学生
	前置条件	ID 和课程 ID
		मि निस्ति है मि
StudentMethod. quitCourse		学生退选课程,根据学生 ID 查找是否存在相
	二里夕 孙	应的用户,根据课程 ID 查找是否存在相应的
		课程,如果该学生已选择该课程,则退课成功,
	后置条件	返回 True, 如果该学生未选择该课程, 则系统
		提示该学生未选择该课程,返回 False,其余
		情况均返回 False
	ハエ ハ J .	public int getScore(String ID,String cID) throws
	语法	RemoteException;
StudentMethod.getScore	V. III. 5- 53	一位学生已经登录,且用户输入有效地学生
	前置条件	ID 和课程 ID
	后置条件	学生查看课程成绩,根据学生 ID 查找是否存
	/P = 47 1	

		在相应的用户,根据课程 ID 查找是否存在相
		应的课程,如果该学生已选择该课程,则显示
		学生成绩,返回 True,如果该该课程未登记成
		绩,则系统提示未登记成绩,返回 False,如果
		该学生未选择该课程,则系统提示该学生未选
		择该课程,返回 False,其余情况均返回 False
		public List <course> getCourseList(String ID)</course>
	语法	throws RemoteException;
StudentMethod.getCourseList		一位学生已经登录,且用户输入有效地学生
	前置条件	ID
	己四石川	学生查看自己的选课列表,根据账号查找是否
	后置条件	存在相应的用户,通过验证,则返回选课列表
	があり仕	public Course course(String cID) throws
	语法	RemoteException;
	公里 夕 (4)	一位学生已经登录,且用户输入有效地课程
StudentMethod.getCourseList	前置条件	ID
		学生查看自己的选课列表,根据课程 ID 查找
	后置条件	是否存在相应的课程,通过验证,则返回课程
		信息
		1H 10.

5.4 数据层的分解

数据层主要给业务逻辑层提供数据访问服务,包括对持久画数据的增、删、改查。 Dean 业务逻辑需要的服务由 DeanDatabaseMethod 接口提供,FacultyDean 业务逻辑需要的服务由 FacultyDeanDatabaseMethod 接口提供,Teacher 业务逻辑需要的服务由 TeacherDatabaseMethod 提供,Student 业务逻辑需要的服务由 StudentDatabaseMethod 提供。数据层模块的描述具体如图 15 所示。



5.4.1 数据层模块的职责 数据层模块的职责如表 13 所示

表 13 数据层模块的职责

模块	职责
DatabaseMethod	持久化数据库的接口,提供集体载入、集体保存、增、
Databasewiethod	删、改、查服务
DeanDatabaseMethod	持久化数据库的接口,提供集体载入、集体保存、增、
DeamPatabaseWethod	删、改、查服务
FacultyDeanDatabaseMethod	持久化数据库的接口,提供集体载入、集体保存、增、
- Taculty Beam Batta ouserviethou	删、改、查服务
TeacherDatabaseMethod	持久化数据库的接口,提供集体载入、集体保存、增、
- Teacher Battabasety Tetriou	删、改、查服务
StudentDatabaseMethod	持久化数据库的接口,提供集体载入、集体保存、增、
Statembatabaservicaso	删、改、查服务
DatabaseImpl	基于 MySql 数据库的持久化数据库的接口,提供集体
	载入、集体保存、增、删、改、查服务
DeanDatabaseImpl	基于 MySql 数据库的持久化数据库的接口,提供集体
	载入、集体保存、增、删、改、查服务
FacultyDeanDatabaseImpl	基于 MySql 数据库的持久化数据库的接口,提供集体
	载入、集体保存、增、删、改、查服务
TeacherDatabaseImpl	基于 MySql 数据库的持久化数据库的接口,提供集体
	载入、集体保存、增、删、改、查服务
StudentDatabaseImpl	基于 MySql 数据库的持久化数据库的接口,提供集体
z to z tilization of the tiliz	载入、集体保存、增、删、改、查服务

5.4.2 数据层模块的接口规范

数据层模块的接口规范如表 14 所示。

表 14 数据层模块的接口规范

	是供的服务 (供接口)
	语法	public Connection getConnection()
DatabaseMethod.getConnection	前置条件	无
	后置条件	初始化持久数据库
		public List search(String
	语法	tableName,String clueName,String
DatabaseMethod.search		clue,String aimName)
	前置条件	无
	后置条件	按搜索条件返回相应的结果
	语法	public List search(String tableName,List
Datahasa Mathadasanah	占 伝	clueName,List clue,List aimName)
DatabaseMethod.search	前置条件	无
	后置条件	按搜索条件返回相应的结果
		public boolean update(String
	语法	tableName,String clueName,String
DatabaseMethod.update		clue,String aimName,String aim)
	 	数据库存在 aimName 属性上与 aim 相同
	前置条件	的数据

	后置条件	更新在 aimName 属性上与 aim 相同的数据
	语法	public boolean insert(String tableName,List <string> clueName,List</string>
DatabaseMethod.insert	旧亿	clue)
	前置条件	数据库里不存在相同的数据
	后置条件	在数据库中插入一个纪录
	语法	public boolean delete(String
DatabaseMethod.delete	占 伝	tableName,String clueName,String clue)
Databaseivietilod.delete	前置条件	无
	后置条件	数据库删除与条件匹配的纪录
		public boolean delete(String
	语法	tableName,List <string></string>
DatabaseMethod.delete		clueName,List <string> clue)</string>
	前置条件	无
	后置条件	数据库删除与条件匹配的纪录

6. 信息视角

6.1 数据持久化对象

Dean 类包含教务员的 ID、姓名。

FacultyDean 类包含院系教务员的 ID、姓名、院系。

Teacher 类包含老师的 ID、姓名、院系。

Student 类包含学生的 ID、姓名、院系、入学年份。

BasicFrame 类包含 A、B、C、D、E、F、L 的学分范围和时间范围。

Frame 类包含 A、B、C、D、E、F、L 的学分。

持久化用户对象的定义参见 com.data.po 包。

6.2 数据库表

basicFrame course courseScore courseStudent courseTeacher courseTime dean deanPassword faculty facultyDean facultyPassword frame
courseScore courseStudent courseTeacher courseTime dean deanPassword faculty facultyDean facultyPassword frame
courseStudent courseTeacher courseTime dean deanPassword faculty faculty facultyPassword frame
courseTeacher courseTime dean deanPassword faculty facultyDean facultyPassword frame
courseTime dean deanPassword faculty faculty facultyPassword frame
dean deanPassword faculty facultyDean facultyPassword frame
deanPassword faculty facultyDean facultyPassword frame
 ▶ faculty ▶ facultyDean ▶ facultyPassword ▶ frame
▶ facultyDean▶ facultyPassword▶ frame
► facultyPassword Frame
▶ frame
b ctudont
▶
▶
▶ i teacher
▶ teacherPassword