

## Front End of BaKaPlan

Generated by Doxygen 1.8.1.2

Fri Apr 26 2013 13:14:19



# Contents

<b>1</b>	<b>Class Index</b>	<b>1</b>
1.1	Class Hierarchy . . . . .	1
<b>2</b>	<b>Class Index</b>	<b>3</b>
2.1	Class List . . . . .	3
<b>3</b>	<b>File Index</b>	<b>5</b>
3.1	File List . . . . .	5
<b>4</b>	<b>Class Documentation</b>	<b>9</b>
4.1	ArrangeRollNo Class Reference . . . . .	9
4.1.1	Detailed Description . . . . .	10
4.1.2	Constructor & Destructor Documentation . . . . .	10
4.1.2.1	ArrangeRollNo . . . . .	10
4.1.2.2	~ArrangeRollNo . . . . .	10
4.1.3	Member Function Documentation . . . . .	10
4.1.3.1	AddPrefixWithRollNo . . . . .	10
4.1.3.2	Main . . . . .	10
4.1.3.3	RemoveNonEligibleRollNo . . . . .	11
4.1.3.4	RemoveRedundancy . . . . .	11
4.1.3.5	SortRollNo . . . . .	11
4.1.3.6	WriteArrangedRollNo . . . . .	11
4.1.4	Member Data Documentation . . . . .	11
4.1.4.1	expandRNo . . . . .	11
4.1.4.2	notIncludedRNo . . . . .	12
4.1.4.3	prefix . . . . .	12
4.1.4.4	prefixRollNo . . . . .	12
4.1.4.5	rollNo . . . . .	12
4.2	ClassDetail Class Reference . . . . .	12
4.2.1	Detailed Description . . . . .	13
4.2.2	Constructor & Destructor Documentation . . . . .	13
4.2.2.1	ClassDetail . . . . .	13

4.2.2.2	~ClassDetail	13
4.2.3	Member Function Documentation	13
4.2.3.1	ClassDetailPage	13
4.2.3.2	NewProject	13
4.2.3.3	OldProject	14
4.2.3.4	ProjectType	14
4.2.3.5	SetDefaultValue	14
4.3	Database Class Reference	14
4.3.1	Detailed Description	15
4.3.2	Constructor & Destructor Documentation	15
4.3.2.1	Database	15
4.3.2.2	~Database	15
4.3.3	Member Function Documentation	16
4.3.3.1	DeleteQuery	16
4.3.3.2	InsertClassDetail	16
4.3.3.3	InsertDateSheet	16
4.3.3.4	InsertExamDetail	16
4.3.3.5	InsertProjectDetail	16
4.3.3.6	InsertQuery	16
4.3.3.7	InsertRegistrationDetail	17
4.3.3.8	InsertRollNoDetail	17
4.3.3.9	InsertRoomDetail	17
4.3.3.10	InsertSessionDetail	17
4.3.3.11	InsertStrategyDetail	17
4.3.3.12	InsertUserDetail	17
4.3.3.13	Query	17
4.3.3.14	SelectColumn	18
4.3.3.15	SelectProjectID	18
4.3.3.16	SelectQuery	18
4.3.3.17	SelectSum	18
4.3.4	Member Data Documentation	18
4.3.4.1	connect	18
4.3.4.2	numrows	18
4.4	Databaso Class Reference	19
4.5	DataSheet Class Reference	19
4.5.1	Detailed Description	19
4.6	DateSheet Class Reference	19
4.6.1	Detailed Description	20
4.6.2	Constructor & Destructor Documentation	20
4.6.2.1	DateSheet	20

4.6.2.2	~DateSheet	20
4.6.2.3	DateSheet	20
4.6.2.4	~DateSheet	20
4.6.3	Member Function Documentation	20
4.6.3.1	AddRollNoWithDateSheet	20
4.6.3.2	DateSheetPage	21
4.6.3.3	Main	21
4.6.3.4	ReadRollNoDetail	21
4.6.3.5	SetDefaultValue	21
4.6.3.6	WriteDateSheet	21
4.6.3.7	WriteRollNoDetail	21
4.7	ExamDetail Class Reference	22
4.7.1	Detailed Description	22
4.7.2	Constructor & Destructor Documentation	22
4.7.2.1	ExamDetail	22
4.7.2.2	~ExamDetail	22
4.7.3	Member Function Documentation	23
4.7.3.1	ExamDetailPage	23
4.7.3.2	ReadRoomDetail	23
4.7.3.3	SetDefaultValue	23
4.7.3.4	WriteRoomDetail	23
4.8	ExpandRollNo Class Reference	23
4.8.1	Detailed Description	23
4.9	ExpandRollNo Class Reference	24
4.9.1	Detailed Description	24
4.9.2	Constructor & Destructor Documentation	24
4.9.2.1	ExpandRollNo	24
4.9.2.2	~ExpandRollNo	24
4.9.3	Member Function Documentation	25
4.9.3.1	ExpandRollNoList	25
4.9.3.2	ExpandRollNos	25
4.9.3.3	WriteExpandRollNo	25
4.10	InputDetail Class Reference	25
4.10.1	Detailed Description	27
4.10.2	Constructor & Destructor Documentation	28
4.10.2.1	InputDetail	28
4.10.3	Member Function Documentation	28
4.10.3.1	ErrorMessage	28
4.10.3.2	FileName	28
4.10.3.3	Footer	28

4.10.3.4	Header	29
4.10.3.5	IntToString	29
4.10.3.6	SplitString	29
4.10.3.7	StringToInt	29
4.10.3.8	Time	29
4.10.4	Member Data Documentation	29
4.10.4.1	centreName	29
4.10.4.2	classID	30
4.10.4.3	className	30
4.10.4.4	columns	30
4.10.4.5	database	30
4.10.4.6	date	30
4.10.4.7	emailID	30
4.10.4.8	endRollNo	30
4.10.4.9	examCode	30
4.10.4.10	examName	30
4.10.4.11	examSession	31
4.10.4.12	examTime	31
4.10.4.13	examVenue	31
4.10.4.14	fieldName	31
4.10.4.15	fileName	31
4.10.4.16	inFile	31
4.10.4.17	l	31
4.10.4.18	msg	31
4.10.4.19	notIncluded	31
4.10.4.20	oldProject	32
4.10.4.21	outFile	32
4.10.4.22	page	32
4.10.4.23	prefix	32
4.10.4.24	projectID	32
4.10.4.25	projectName	32
4.10.4.26	projectType	32
4.10.4.27	readField	32
4.10.4.28	roomNo	32
4.10.4.29	rows	33
4.10.4.30	sameDetail	33
4.10.4.31	sessionID	33
4.10.4.32	splitString	33
4.10.4.33	ss	33
4.10.4.34	startRollNo	33

4.10.4.35 strategyChoice . . . . .	33
4.10.4.36 subjectCode . . . . .	33
4.10.4.37 subjectName . . . . .	33
4.10.4.38 tableHeading . . . . .	34
4.10.4.39 temp1 . . . . .	34
4.10.4.40 totalCentres . . . . .	34
4.10.4.41 totalClasses . . . . .	34
4.10.4.42 totalCols . . . . .	34
4.10.4.43 totalDays . . . . .	34
4.10.4.44 totalRows . . . . .	34
4.10.4.45 vecTemp . . . . .	34
4.10.4.46 where . . . . .	34
4.11 InputFieldName Class Reference . . . . .	35
4.11.1 Detailed Description . . . . .	35
4.11.2 Constructor & Destructor Documentation . . . . .	36
4.11.2.1 InputFieldName . . . . .	36
4.11.3 Member Function Documentation . . . . .	36
4.11.3.1 SetFieldNames . . . . .	36
4.11.4 Member Data Documentation . . . . .	36
4.11.4.1 centreName . . . . .	36
4.11.4.2 classID . . . . .	36
4.11.4.3 className . . . . .	36
4.11.4.4 columns . . . . .	37
4.11.4.5 date . . . . .	37
4.11.4.6 emailID . . . . .	37
4.11.4.7 endRollNo . . . . .	37
4.11.4.8 examCode . . . . .	37
4.11.4.9 examDate . . . . .	37
4.11.4.10 examName . . . . .	37
4.11.4.11 examSession . . . . .	37
4.11.4.12 examTime . . . . .	37
4.11.4.13 examVenue . . . . .	38
4.11.4.14 key . . . . .	38
4.11.4.15 notIncluded . . . . .	38
4.11.4.16 password . . . . .	38
4.11.4.17 prefix . . . . .	38
4.11.4.18 projectID . . . . .	38
4.11.4.19 projectName . . . . .	38
4.11.4.20 projectType . . . . .	38
4.11.4.21 retypePassword . . . . .	38

4.11.4.22 roomNo . . . . .	39
4.11.4.23 rows . . . . .	39
4.11.4.24 sameDetail . . . . .	39
4.11.4.25 startRollNo . . . . .	39
4.11.4.26 strategyChoice . . . . .	39
4.11.4.27 subjectCode . . . . .	39
4.11.4.28 subjectName . . . . .	39
4.11.4.29 totalCentres . . . . .	39
4.11.4.30 totalClasses . . . . .	39
4.11.4.31 totalDays . . . . .	40
4.11.4.32 totalRooms . . . . .	40
4.11.4.33 totalSubjects . . . . .	40
4.12 JavaScript Class Reference . . . . .	40
4.12.1 Detailed Description . . . . .	40
4.12.2 Constructor & Destructor Documentation . . . . .	40
4.12.2.1 JavaScript . . . . .	40
4.12.3 Member Function Documentation . . . . .	41
4.12.3.1 OnChangeEvent . . . . .	41
4.12.3.2 OnFocusEvent . . . . .	41
4.13 Login Class Reference . . . . .	41
4.13.1 Detailed Description . . . . .	42
4.13.2 Constructor & Destructor Documentation . . . . .	42
4.13.2.1 Login . . . . .	42
4.13.2.2 ~Login . . . . .	42
4.13.3 Member Function Documentation . . . . .	42
4.13.3.1 ConfirmPage . . . . .	42
4.13.3.2 LoginPage . . . . .	43
4.13.3.3 LogoutPage . . . . .	43
4.13.3.4 NewUser . . . . .	43
4.13.3.5 ReadLoginDetail . . . . .	43
4.13.3.6 RegistrationPage . . . . .	43
4.13.3.7 SelectLoginDetail . . . . .	43
4.13.4 Member Data Documentation . . . . .	43
4.13.4.1 currentTime . . . . .	44
4.13.4.2 emailID . . . . .	44
4.13.4.3 password . . . . .	44
4.13.4.4 regKey . . . . .	44
4.13.4.5 retypePassword . . . . .	44
4.13.4.6 userEmailID . . . . .	44
4.13.4.7 userPassword . . . . .	44



4.14	jwsmtplib::mailer Class Reference	44
4.14.1	Detailed Description	45
4.15	MD5 Class Reference	46
4.15.1	Detailed Description	46
4.16	PageLayout Class Reference	46
4.16.1	Detailed Description	47
4.16.2	Constructor & Destructor Documentation	47
4.16.2.1	PageLayout	47
4.16.3	Member Function Documentation	47
4.16.3.1	ContentType	47
4.16.3.2	Footer	47
4.16.3.3	Head	48
4.16.3.4	Header	48
4.16.3.5	Logo	48
4.16.3.6	LogoutLink	48
4.16.3.7	Menu	48
4.16.3.8	SetCookies	49
4.17	PageStructureMaker Class Reference	49
4.17.1	Detailed Description	50
4.17.2	Constructor & Destructor Documentation	50
4.17.2.1	PageStructureMaker	50
4.17.3	Member Function Documentation	51
4.17.3.1	Anchor	51
4.17.3.2	BodyEnd	51
4.17.3.3	BodyStart	51
4.17.3.4	Button	51
4.17.3.5	CSS	51
4.17.3.6	DivEnd	52
4.17.3.7	DivStart	52
4.17.3.8	FormEnd	52
4.17.3.9	FormStart	52
4.17.3.10	HeadEnd	52
4.17.3.11	HeadStart	53
4.17.3.12	HTMLEnd	53
4.17.3.13	HTMLStart	53
4.17.3.14	InputField	53
4.17.3.15	InputField	53
4.17.3.16	Javascript	54
4.17.3.17	Label	54
4.17.3.18	ListEnd	54

4.17.3.19 ListItem . . . . .	54
4.17.3.20 ListStart . . . . .	54
4.17.3.21 SelectFieldEnd . . . . .	55
4.17.3.22 SelectFieldStart . . . . .	55
4.17.3.23 SelectOptionEnd . . . . .	55
4.17.3.24 SelectOptionStart . . . . .	55
4.17.3.25 SetHTMLVariables . . . . .	55
4.17.3.26 TableEnd . . . . .	56
4.17.3.27 TableStart . . . . .	56
4.17.3.28 Title . . . . .	56
4.17.4 Member Data Documentation . . . . .	56
4.17.4.1 startH1 . . . . .	56
4.18 ProjectDetail Class Reference . . . . .	56
4.18.1 Detailed Description . . . . .	57
4.18.2 Constructor & Destructor Documentation . . . . .	57
4.18.2.1 ProjectDetail . . . . .	57
4.18.2.2 ~ProjectDetail . . . . .	57
4.18.3 Member Function Documentation . . . . .	58
4.18.3.1 AuthorizeUser . . . . .	58
4.18.3.2 HomePage . . . . .	58
4.18.3.3 OldProject . . . . .	58
4.18.3.4 ProjectDetailPage . . . . .	58
4.19 ReadInput Class Reference . . . . .	59
4.19.1 Detailed Description . . . . .	60
4.19.2 Constructor & Destructor Documentation . . . . .	60
4.19.2.1 ReadInput . . . . .	60
4.19.2.2 ~ReadInput . . . . .	60
4.19.3 Member Function Documentation . . . . .	61
4.19.3.1 FileName . . . . .	61
4.19.3.2 ReadClassDetail . . . . .	61
4.19.3.3 ReadDateSheet . . . . .	61
4.19.3.4 ReadDateSheetOut . . . . .	61
4.19.3.5 ReadExamDetail . . . . .	62
4.19.3.6 ReadRollNoDetail . . . . .	62
4.19.3.7 ReadRoomDetail . . . . .	62
4.19.3.8 ReadStrategy . . . . .	62
4.19.4 Member Data Documentation . . . . .	62
4.19.4.1 centreName . . . . .	62
4.19.4.2 className . . . . .	62
4.19.4.3 cols . . . . .	63

4.19.4.4	date	63
4.19.4.5	dateSheetRNo	63
4.19.4.6	dateSheetRNoSize	63
4.19.4.7	examSubCode	63
4.19.4.8	examSubName	63
4.19.4.9	fileName	63
4.19.4.10	inFile	63
4.19.4.11	n	63
4.19.4.12	notIncluded	64
4.19.4.13	outFile	64
4.19.4.14	prefix	64
4.19.4.15	projectID	64
4.19.4.16	rollNo	64
4.19.4.17	roomNo	64
4.19.4.18	rows	64
4.19.4.19	strategyOption	64
4.19.4.20	subjectCode	64
4.19.4.21	subjectName	65
4.19.4.22	temp	65
4.19.4.23	totalCentres	65
4.19.4.24	totalClasses	65
4.19.4.25	totalDays	65
4.19.4.26	totalExams	65
4.19.4.27	totalRooms	65
4.19.4.28	totalSubjects	65
4.20	ReadInputField Class Reference	65
4.20.1	Detailed Description	66
4.20.2	Constructor & Destructor Documentation	66
4.20.2.1	ReadInputField	66
4.20.3	Member Function Documentation	66
4.20.3.1	ReadFieldValue	66
4.20.3.2	ReadFieldValue	66
4.20.4	Member Data Documentation	67
4.20.4.1	fi	67
4.20.4.2	formData	67
4.21	Report Class Reference	67
4.21.1	Detailed Description	67
4.21.2	Constructor & Destructor Documentation	68
4.21.2.1	Report	68
4.21.2.2	~Report	68

4.21.3	Member Function Documentation	68
4.21.3.1	ReadValidationDetail	68
4.21.3.2	ReportPage	68
4.22	RollNoDetail Class Reference	68
4.22.1	Detailed Description	69
4.22.2	Constructor & Destructor Documentation	69
4.22.2.1	RollNoDetail	69
4.22.2.2	~RollNoDetail	69
4.22.3	Member Function Documentation	69
4.22.3.1	ReadClassDetail	69
4.22.3.2	RollNoDetailPage	69
4.22.3.3	SetDefaultValue	70
4.22.3.4	WriteClassDetail	70
4.23	RoomDetail Class Reference	70
4.23.1	Detailed Description	71
4.23.2	Constructor & Destructor Documentation	71
4.23.2.1	RoomDetail	71
4.23.2.2	~RoomDetail	71
4.23.3	Member Function Documentation	71
4.23.3.1	ReadDateSheet	71
4.23.3.2	RoomDetailPage	71
4.23.3.3	SetDefaultValue	71
4.23.3.4	WriteDateSheet	71
4.24	SeatPlan Class Reference	72
4.24.1	Detailed Description	73
4.24.2	Constructor & Destructor Documentation	73
4.24.2.1	~RoomReport	73
4.24.2.2	SeatPlan	73
4.24.2.3	~SeatPlan	73
4.24.3	Member Function Documentation	73
4.24.3.1	ReadExamDetail	73
4.24.3.2	ReadSeatPlan	74
4.24.3.3	RoomReport	74
4.24.3.4	WriteHTMLFile	74
4.24.3.5	WriteHTMLFile	74
4.25	SendMail Class Reference	74
4.25.1	Detailed Description	75
4.25.2	Constructor & Destructor Documentation	75
4.25.2.1	SendMail	75
4.25.2.2	~SendMail	75

4.25.3	Member Function Documentation	75
4.25.3.1	RegistrationMail	75
4.25.3.2	SetHTMLMessage	75
4.25.3.3	SetMailData	76
4.26	jwsmtplib::SOCKADDR_IN Struct Reference	76
4.26.1	Detailed Description	76
4.27	Strategy Class Reference	76
4.27.1	Detailed Description	77
4.27.2	Constructor & Destructor Documentation	77
4.27.2.1	Strategy	77
4.27.2.2	~Strategy	78
4.27.2.3	Strategy	78
4.27.2.4	~Strategy	78
4.27.3	Member Function Documentation	78
4.27.3.1	Main	78
4.27.3.2	ReadExamDetail	78
4.27.3.3	SetDefaultValue	78
4.27.3.4	StrategyPage	78
4.27.3.5	WriteExamDetail	78
4.27.4	Member Data Documentation	79
4.27.4.1	strategyName	79
4.27.4.2	totalStrategy	79
4.28	Validation Class Reference	79
4.28.1	Detailed Description	80
4.28.2	Constructor & Destructor Documentation	80
4.28.2.1	Validation	80
4.28.2.2	~Validation	80
4.28.3	Member Function Documentation	80
4.28.3.1	ReadStrategyDetail	80
4.28.3.2	ReadValidation	80
4.28.3.3	ValidationPage	80
4.28.3.4	WriteStrategyDetail	80
<b>5</b>	<b>File Documentation</b>	<b>83</b>
5.1	src/adduser-main.cpp File Reference	83
5.1.1	Detailed Description	83
5.1.2	Function Documentation	83
5.1.2.1	main	84
5.2	src/backend/arrangerollno.cc File Reference	84
5.2.1	Detailed Description	84

5.3	<a href="#">src/backend/expandrollno.cc File Reference</a>	85
5.3.1	<a href="#">Detailed Description</a>	85
5.4	<a href="#">src/backend/header/arrangerollno.h File Reference</a>	85
5.4.1	<a href="#">Detailed Description</a>	85
5.5	<a href="#">src/backend/header/expandrollno.h File Reference</a>	86
5.5.1	<a href="#">Detailed Description</a>	86
5.6	<a href="#">src/backend/header/readinput.h File Reference</a>	86
5.6.1	<a href="#">Detailed Description</a>	87
5.7	<a href="#">src/backend/header/seatplan.h File Reference</a>	87
5.7.1	<a href="#">Detailed Description</a>	87
5.8	<a href="#">src/backend/main.cpp File Reference</a>	88
5.8.1	<a href="#">Detailed Description</a>	88
5.9	<a href="#">src/backend/readinput.cc File Reference</a>	88
5.9.1	<a href="#">Detailed Description</a>	88
5.10	<a href="#">src/backend/seatplan.cc File Reference</a>	89
5.10.1	<a href="#">Detailed Description</a>	89
5.11	<a href="#">src/class-main.cpp File Reference</a>	89
5.11.1	<a href="#">Detailed Description</a>	90
5.12	<a href="#">src/class.cc File Reference</a>	90
5.12.1	<a href="#">Detailed Description</a>	90
5.13	<a href="#">src/confirm-main.cpp File Reference</a>	91
5.13.1	<a href="#">Detailed Description</a>	91
5.14	<a href="#">src/datesheet-main.cpp File Reference</a>	91
5.14.1	<a href="#">Detailed Description</a>	91
5.15	<a href="#">src/exam-main.cpp File Reference</a>	92
5.15.1	<a href="#">Detailed Description</a>	92
5.16	<a href="#">src/exam.cc File Reference</a>	92
5.16.1	<a href="#">Detailed Description</a>	93
5.17	<a href="#">src/header/class.h File Reference</a>	93
5.17.1	<a href="#">Detailed Description</a>	93
5.18	<a href="#">src/header/exam.h File Reference</a>	94
5.18.1	<a href="#">Detailed Description</a>	94
5.19	<a href="#">src/header/rollno.h File Reference</a>	94
5.19.1	<a href="#">Detailed Description</a>	94
5.20	<a href="#">src/header/room.h File Reference</a>	95
5.20.1	<a href="#">Detailed Description</a>	95
5.21	<a href="#">src/header/sendmail-detail.h File Reference</a>	96
5.21.1	<a href="#">Detailed Description</a>	96
5.22	<a href="#">src/header/sendmail.h File Reference</a>	96
5.22.1	<a href="#">Detailed Description</a>	97

5.23	src/header/validation.h File Reference	97
5.23.1	Detailed Description	97
5.24	src/javascript.cc File Reference	98
5.24.1	Detailed Description	98
5.25	src/login-main.cpp File Reference	98
5.25.1	Detailed Description	98
5.26	src/logout-main.cpp File Reference	99
5.26.1	Detailed Description	99
5.27	src/register-main.cpp File Reference	99
5.27.1	Detailed Description	100
5.27.2	Function Documentation	100
5.27.2.1	main	100
5.28	src/report-main.cpp File Reference	100
5.28.1	Detailed Description	100
5.29	src/rollno-main.cpp File Reference	101
5.29.1	Detailed Description	101
5.30	src/rollno.cc File Reference	101
5.30.1	Detailed Description	102
5.31	src/room-main.cpp File Reference	102
5.31.1	Detailed Description	102
5.32	src/room.cc File Reference	103
5.32.1	Detailed Description	103
5.33	src/sendmail.cc File Reference	103
5.33.1	Detailed Description	103
5.34	src/strategy-main.cpp File Reference	104
5.34.1	Detailed Description	104
5.35	src/validation-main.cpp File Reference	104
5.35.1	Detailed Description	104
5.36	src/validation.cc File Reference	105
5.36.1	Detailed Description	105





# Chapter 1

## Class Index

### 1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Database . . . . .	14
Databaso . . . . .	19
DataSheet . . . . .	19
ExapandRollNo . . . . .	23
InputDetail . . . . .	25
ClassDetail . . . . .	12
DateSheet . . . . .	19
ExamDetail . . . . .	22
Login . . . . .	41
ProjectDetail . . . . .	56
Report . . . . .	67
RollNoDetail . . . . .	68
RoomDetail . . . . .	70
Strategy . . . . .	76
Validation . . . . .	79
InputFieldName . . . . .	35
JavaScript . . . . .	40
jwsmtplib::mailer . . . . .	44
MD5 . . . . .	46
PageStructureMaker . . . . .	49
PageLayout . . . . .	46
ReadInput . . . . .	59
ArrangeRollNo . . . . .	9
DateSheet . . . . .	19
ExpandRollNo . . . . .	24
SeatPlan . . . . .	72
Strategy . . . . .	76
ReadInputField . . . . .	65
SendMail . . . . .	74
jwsmtplib::SOCKADDR_IN . . . . .	76



## Chapter 2

# Class Index

### 2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">ArrangeRollNo</a>	Arrange roll no class for sorting, removing redundant roll nos and also excludeing roll nos that are not for seating plan . . . . .	9
<a href="#">ClassDetail</a>	For getting class details nd reading projectdetail . . . . .	12
<a href="#">Database</a>	. . . . .	14
<a href="#">Databaso</a>	. . . . .	19
<a href="#">DataSheet</a>	. . . . .	19
<a href="#">DateSheet</a>	<a href="#">DateSheet</a> class for reading datesheet detail, class detail and create datesheet.out file for seating plan . . . . .	19
<a href="#">ExamDetail</a>	<a href="#">ExamDetail</a> class for getting exam from user . . . . .	22
<a href="#">ExapandRollNo</a>	Class for expanding roll nos . . . . .	23
<a href="#">ExpandRollNo</a>	. . . . .	24
<a href="#">InputDetail</a>	Declare common variables, functions that are used by derived class . . . . .	25
<a href="#">InputFieldName</a>	. . . . .	35
<a href="#">JavaScript</a>	. . . . .	40
<a href="#">Login</a>	. . . . .	41
<a href="#">jwsmtplib::mailer</a>	. . . . .	44
<a href="#">MD5</a>	. . . . .	46
<a href="#">PageLayout</a>	. . . . .	46
<a href="#">PageStructureMaker</a>	. . . . .	49
<a href="#">ProjectDetail</a>	<a href="#">ProjectDetail</a> class for creating new project for user . . . . .	56
<a href="#">ReadInput</a>	Class for reading input files ie reading roll no details, class detail, room detail, etc . . . . .	59
<a href="#">ReadInputField</a>	. . . . .	65
<a href="#">Report</a>	. . . . .	67
<a href="#">RollNoDetail</a>	<a href="#">RollNoDetail</a> class for taking roll no detail from user . . . . .	68
<a href="#">RoomDetail</a>	. . . . .	70
<a href="#">SeatPlan</a>	<a href="#">SeatPlan</a> CClass for generating seating plan . . . . .	72
<a href="#">SendMail</a>	Class for sending mail to user for registration . . . . .	74

jwsmtplib::SOCKADDR_IN . . . . .	76
Strategy	
Strategy class to check selected strategy is valid or not for seating plan . . . . .	76
Validation . . . . .	79

## Chapter 3

# File Index

### 3.1 File List

Here is a list of all documented files with brief descriptions:

src/adduser-main.cpp		
Adding new user and setting its password	83	
src/class-main.cpp		
Main method	89	
src/class.cc		
ClassDetail function definition	90	
src/confirm-main.cpp		
Main method to call ConfirmPage of Login class and creating confirm.html page	91	
src/database.cc	??	
src/datesheet-main.cpp		
Main method	91	
src/datesheet.cc	??	
src/exam-main.cpp		
Main method	92	
src/exam.cc		
ExamDetail Class func. definition	92	
src/inputdetail.cc	??	
src/inputfieldname.cc	??	
src/javascript.cc		
Function definition of javascript class	98	
src/login-main.cpp		
Main Function for create login.html page	98	
src/login.cc	??	
src/logout-main.cpp		
Log out user account	99	
src/md5.cc	??	
src/newuser-main.cpp	??	
src/pagelayout.cc	??	
src/pagestructure.cc	??	
src/project-main.cpp	??	
src/project.cc	??	
src/readinputfield.cc	??	
src/register-main.cpp		
Main method for register.html	99	
src/report-main.cpp		
Main method	100	
src/report.cc	??	

src/rollno-main.cpp	
Main method	101
src/rollno.cc	
RollNo detail Class definition	101
src/room-main.cpp	
Main method	102
src/room.cc	
RoomDetail class definition	103
src/sendmail.cc	
Function Definitons of SendMail Class	103
src/strategy-main.cpp	
Main method	104
src/strategy.cc	??
src/validation-main.cpp	
Main method	104
src/validation.cc	
Validation func. definition	105
src/backend/arrangerollno.cc	
Function definition of ArrangeRollNo class	84
src/backend/datesheet.cc	??
src/backend/expandrollno.cc	
Function definition of ExpandRollNo class defintion	85
src/backend/main.cpp	
Main Method	88
src/backend/readinput.cc	
Function Definition of ReadInput Class	88
src/backend/report.cc	??
src/backend/seatplan.cc	
Function definition of SeatPlan Class	89
src/backend/strategy.cc	??
src/backend/header/arrangerollno.h	
ArrangeRollNo class for arranging roll no, sorting, and removing roll nos that are not required for seating plan	85
src/backend/header/datesheet.h	??
src/backend/header/expandrollno.h	
ExpandRollNo class declaration for expanding roll nos	86
src/backend/header/filename.h	??
src/backend/header/header.h	??
src/backend/header/readinput.h	
Declaration of ReadInput class for reading input details from file	86
src/backend/header/report.h	??
src/backend/header/seatplan.h	
Declaration of SeatPlan class for generating seating plan	87
src/backend/header/strategy.h	??
src/header/cgicc.h	??
src/header/class.h	
ClassDetail Class declaration	93
src/header/constant.h	??
src/header/database-detail.h	??
src/header/database.h	??
src/header/datesheet.h	??
src/header/exam.h	
ExamDetail class for exam detail	94
src/header/filename.h	??
src/header/header.h	??
src/header/inputdetail.h	??
src/header/inputfieldname.h	??
src/header/javascript.h	??

src/header/ <b>login.h</b>	??
src/header/ <b>md5.h</b>	??
src/header/ <b>pagelayout.h</b>	??
src/header/ <b>pagestructure.h</b>	??
src/header/ <b>project.h</b>	??
src/header/ <b>readinputfield.h</b>	??
src/header/ <b>report.h</b>	??
src/header/ <b>rollno.h</b>	
RollNo class declaration for taking roll no detail	94
src/header/ <b>room.h</b>	
RoomDetail class	95
src/header/ <b>sendmail-detail.h</b>	
SMTP swttings detail	96
src/header/ <b>sendmail.h</b>	
Declaration of SendMail Class usinf jwSMTP library	96
src/header/ <b>strategy.h</b>	??
src/header/ <b>validation.h</b>	
Validation class	97
src/header/jwsmtp/ <b>base64.cpp</b>	??
src/header/jwsmtp/ <b>base64.h</b>	??
src/header/jwsmtp/ <b>compat.cpp</b>	??
src/header/jwsmtp/ <b>compat.h</b>	??
src/header/jwsmtp/ <b>jwsmtp.h</b>	??
src/header/jwsmtp/ <b>mailer.cpp</b>	??
src/header/jwsmtp/ <b>mailer.h</b>	??





## Chapter 4

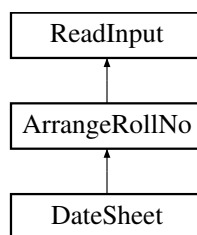
# Class Documentation

### 4.1 ArrangeRollNo Class Reference

Arrange roll no class for sorting, removing redundant roll nos and also excluding roll nos that are not for seating plan.

```
#include <arrangerollno.h>
```

Inheritance diagram for ArrangeRollNo:



#### Public Member Functions

- [ArrangeRollNo](#) ()  
*Constructor.*
- void [SortRollNo](#) (INT\_VEC &rollno)  
*Function for sorting roll no vector array.*
- void [RemoveRedundancy](#) (INT\_VEC &rollno)
- void [RemoveNonEligibleRollNo](#) (INT\_VEC &rollno, INT\_VEC &[notIncludedRNo](#))  
*Removing roll no that are not eligible.*
- void [AddPrefixWithRollNo](#) (INT\_VEC &rollno, STRING\_VEC &[prefix](#))  
*Adding prefix with roll nos.*
- void [WriteArrangedRollNo](#) ()  
*Writing arranged roll nos to file as O/p file.*
- void [Main](#) (string pID)  
*Main function to call all functions.*
- [~ArrangeRollNo](#) ()  
*Destructor.*

## Protected Attributes

- INT\_VEC [rollNo](#)
- INT\_VEC [notIncludedRNo](#)
- INT\_VEC **rollNoSize**
- INT\_VEC **notIncludedRNoSize**
- STRING\_VEC [prefix](#)
- STRING\_2DVEC [prefixRollNo](#)
- [ExpandRollNo](#) [expandRNo](#)

### 4.1.1 Detailed Description

Arrange roll no class for sorting, removing redundant roll nos and also excluding roll nos that are not for seating plan.

Include [arrangerollno.h](#) file

Include [expandrollno.h](#) file

Definition at line 30 of file [arrangerollno.h](#).

### 4.1.2 Constructor & Destructor Documentation

#### 4.1.2.1 ArrangeRollNo::ArrangeRollNo ( )

Constructor.

Constructor

Definition at line 27 of file [arrangerollno.cc](#).

#### 4.1.2.2 ArrangeRollNo::~~ArrangeRollNo ( )

Destructor.

Desrtuctor

Definition at line 212 of file [arrangerollno.cc](#).

### 4.1.3 Member Function Documentation

#### 4.1.3.1 void ArrangeRollNo::AddPrefixWithRollNo ( INT\_VEC & *rollNo*, STRING\_VEC & *prefix* )

Adding prefix with roll nos.

Adding prefix with roll nos

##### Parameters

<i>rollNo</i>	List of eligible roll nos
<i>prefix</i>	Prefix ie added with roll nos

Definition at line 107 of file [arrangerollno.cc](#).

#### 4.1.3.2 void ArrangeRollNo::Main ( string *pid* )

Main function to call all functions.

Main method to call all function in order/sequence

#### Parameters

<i>pID</i>	Project ID
------------	------------

Reimplemented in [DateSheet](#).

Definition at line 151 of file arrangerollno.cc.

#### 4.1.3.3 void ArrangeRollNo::RemoveNonEligibleRollNo ( INT\_VEC & *rollNo*, INT\_VEC & *notIncludedRNo* )

Removing roll no that are not eligible.

Removing non eligible roll nos

#### Parameters

<i>rollNo</i>	roll nos for seating plan
<i>notIncludedRNo</i>	Roll no list that is not eligible for seating plan

Definition at line 79 of file arrangerollno.cc.

#### 4.1.3.4 void ArrangeRollNo::RemoveRedundancy ( INT\_VEC & *rollNo* )

Removing redundancy from roll nos

Definition at line 51 of file arrangerollno.cc.

#### 4.1.3.5 void ArrangeRollNo::SortRollNo ( INT\_VEC & *rollNo* )

Function for sorting roll no vector array.

Sorting Roll nos

#### Parameters

<i>rollNo</i>	For sorting array
---------------	-------------------

Definition at line 39 of file arrangerollno.cc.

#### 4.1.3.6 void ArrangeRollNo::WriteArrangedRollNo ( )

Writing arranged roll nos to file as O/p file.

Writing arranged roll no's into file

Definition at line 124 of file arrangerollno.cc.

### 4.1.4 Member Data Documentation

#### 4.1.4.1 ExpandRollNo ArrangeRollNo::expandRNo [protected]

Object of [ExpandRollNo](#) class

Definition at line 45 of file arrangerollno.h.

#### 4.1.4.2 INT\_VEC ArrangeRollNo::notIncludedRNo [protected]

storing rollnos that are not included

Definition at line 34 of file arrangerollno.h.

#### 4.1.4.3 STRING\_VEC ArrangeRollNo::prefix [protected]

Storing prefix of roll nos

Definition at line 40 of file arrangerollno.h.

#### 4.1.4.4 STRING\_2DVEC ArrangeRollNo::prefixRollNo [protected]

Roll no with prefix

Definition at line 43 of file arrangerollno.h.

#### 4.1.4.5 INT\_VEC ArrangeRollNo::rollNo [protected]

Reading roll nos

Definition at line 34 of file arrangerollno.h.

The documentation for this class was generated from the following files:

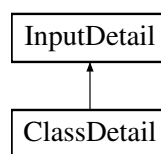
- [src/backend/header/arrangerollno.h](#)
- [src/backend/arrangerollno.cc](#)

## 4.2 ClassDetail Class Reference

For getting class details nd reading projectdetail.

```
#include <class.h>
```

Inheritance diagram for ClassDetail:



### Public Member Functions

- [ClassDetail](#) ()  
*Constructor.*
- void [SetDefaultValue](#) ()  
*Setting default values of text fields (classname, subject name, subject code)*
- void [ProjectType](#) ()  
*For reading project type ie old or new then calling OldProject and NewProject func. accord to that.*
- void [OldProject](#) ()  
*This func. is called is project type = Old.*
- void [NewProject](#) ()  
*This func tis called if ProjectType = New.*

- void [ClassDetailPage](#) (string *msg*="")  
*function for displaying html page for class details*
- [~ClassDetail](#) ()  
*Destructor.*

## Protected Attributes

- [ProjectDetail](#) **project**
- [STRING\\_VEC](#) **value**

### 4.2.1 Detailed Description

For getting class details nd reading projectdetail.

Include [class.h](#)

Include [project.h](#)

Definition at line 30 of file class.h.

### 4.2.2 Constructor & Destructor Documentation

#### 4.2.2.1 [ClassDetail::ClassDetail](#) ( )

Constructor.

Constructor

Definition at line 27 of file class.cc.

#### 4.2.2.2 [ClassDetail::~~ClassDetail](#) ( )

Destructor.

DEstructor

Definition at line 289 of file class.cc.

### 4.2.3 Member Function Documentation

#### 4.2.3.1 [void ClassDetail::ClassDetailPage](#) ( string *msg* = " " )

function for displaying html page for class details

Class detail page

#### Parameters

<i>msg</i>	For displaying error message if any
------------	-------------------------------------

Definition at line 178 of file class.cc.

#### 4.2.3.2 [void ClassDetail::NewProject](#) ( )

This func tis called if ProjectType = New.

New project < If projectName already exists

Definition at line 93 of file class.cc.

#### 4.2.3.3 void ClassDetail::OldProject ( )

This func. is called is project type = Old.

Existing project < If projectName already exists

Definition at line 129 of file class.cc.

#### 4.2.3.4 void ClassDetail::ProjectType ( )

For reading project type ie old or new then calling OldProject and NewProject func. accord to that.

Type of project

Definition at line 48 of file class.cc.

#### 4.2.3.5 void ClassDetail::SetDefaultValue ( )

Setting default values of text fields (classname, subject name, subject code)

Set default values

Definition at line 74 of file class.cc.

The documentation for this class was generated from the following files:

- src/header/class.h
- src/class.cc

## 4.3 Database Class Reference

```
#include <database.h>
```

### Public Member Functions

- [Database](#) ()
- void [Query](#) (string query)
  - Function to execute any query.*
- void [SelectQuery](#) (string query, STRING\_VEC &result)
  - Select Query for selecting single column from table.*
- void [InsertQuery](#) (string query)
  - Function for executing InsertQuery.*
- void [DeleteQuery](#) (string table, string where)
  - Delete Query.*
- void [SelectSum](#) (string column, string table, string projectID, string &result)
- void [SelectColumn](#) (STRING\_VEC &result, string column, string table, string where="")
- void [SelectProjectID](#) (string &projectID)
  - Select project ID from database.*
- void [InsertUserDetail](#) (string emailID, string password)
  - Insert User detail into User table.*
- void [InsertRegistrationDetail](#) (string emailID, string registrationKey)
  - Inserting Registration detail.*
- void [InsertProjectDetail](#) (string emailID, string projectName)

*Inserting Project Detail into database.*

- void [InsertClassDetail](#) (string projectID, string className, string subjectName, string subjectCode)
- void [InsertRollNoDetail](#) (string projectID, string classID, string prefix, string startRNo, string endRNo, string notIncluded)
- void [InsertRoomDetail](#) (string projectID, string dateSheetID, string centreName, string roomNo, string rows, string cols)
- void [InsertStrategyDetail](#) (string projectID, string dateSheetID, string strategyName)
- void [InsertDateSheet](#) (string projectID, string date, string examCode)
- void [InsertExamDetail](#) (string projectID, string dateSheetID, string examName, string examSession, string examTime, string examVenue)
- void [InsertSessionDetail](#) (string emailID, string sessionKey)
- [~Database](#) ()

## Protected Attributes

- MYSQL \* [connect](#)
- MYSQL\_RES \* [res\\_set](#)
- MYSQL\_ROW [row](#)
- unsigned int [numrows](#)
- STRING\_VEC [vecTemp](#)
- int [i](#)
- int [j](#)
- string [query](#)

### 4.3.1 Detailed Description

Include header.h, [database-detail.h](#), mysql.h

Class: [Database](#)

Description: [Database](#) class for database accessibility

Definition at line 37 of file database.h.

### 4.3.2 Constructor & Destructor Documentation

#### 4.3.2.1 Database::Database ( )

[Database](#) Constructor

Include [database.h](#) file local header file(declaration of class)

Class: [Database](#) Method: [Database](#) :: [Database](#)()

Description: Constructor to initialize database conectivity

Definition at line 33 of file database.cc.

#### 4.3.2.2 Database::~~Database ( )

Select last project Id of project name [Database](#) Destructor

Class: [Database](#) Method: [Database](#) :: [~Database](#)()

Description: Destructor for closing database connectivity

Definition at line 311 of file database.cc.

### 4.3.3 Member Function Documentation

#### 4.3.3.1 void Database::DeleteQuery ( string *table*, string *where* = " " )

Delete Query.

Delete Query

##### Parameters

--	--

Definition at line 110 of file database.cc.

#### 4.3.3.2 void Database::InsertClassDetail ( string *projectID*, string *className*, string *subjectName*, string *subjectCode* )

Insert Class Detail

Definition at line 214 of file database.cc.

#### 4.3.3.3 void Database::InsertDateSheet ( string *projectID*, string *date*, string *examCode* )

Insert [DateSheet](#)

Definition at line 262 of file database.cc.

#### 4.3.3.4 void Database::InsertExamDetail ( string *projectID*, string *dateSheetID*, string *examName*, string *examSession*, string *examTime*, string *examVenue* )

Insert Exam Detail

Definition at line 272 of file database.cc.

#### 4.3.3.5 void Database::InsertProjectDetail ( string *emailID*, string *projectName* )

Inserting Project Detail into database.

Insert Project Detail

Definition at line 204 of file database.cc.

#### 4.3.3.6 void Database::InsertQuery ( string *query* )

Function for executing InsertQuery.

Insert Query

##### Parameters

<i>query</i>	query for inserting into value
--------------	--------------------------------

Definition at line 97 of file database.cc.



4.3.3.7 void Database::InsertRegistrationDetail ( string *emailID*, string *registrationKey* )

Inserting Registration detail.

Insert Registration Detail

#### Parameters

<i>emailID</i>	emailId of user
<i>registrationKey</i>	Registration Key

Definition at line 187 of file database.cc.

4.3.3.8 void Database::InsertRollNoDetail ( string *projectID*, string *classID*, string *prefix*, string *startRNo*, string *endRNo*, string *notIncluded* )

Insert RollNo detail

Definition at line 226 of file database.cc.

4.3.3.9 void Database::InsertRoomDetail ( string *projectID*, string *dateSheetID*, string *centreName*, string *roomNo*, string *rows*, string *cols* )

Insert Room Detail

Definition at line 238 of file database.cc.

4.3.3.10 void Database::InsertSessionDetail ( string *emailID*, string *sessionKey* )

Insert Session Detail

Definition at line 285 of file database.cc.

4.3.3.11 void Database::InsertStrategyDetail ( string *projectID*, string *dateSheetID*, string *strategyName* )

Insert [Strategy](#) Detail

Definition at line 251 of file database.cc.

4.3.3.12 void Database::InsertUserDetail ( string *emailID*, string *password* )

Insert User detail into User table.

Insert User Detail into database

#### Parameters

<i>emailID</i>	Email Id
<i>password</i>	Password of user in encrypted form

Definition at line 170 of file database.cc.

4.3.3.13 void Database::Query ( string *query* )

Function to execute any query.

Function for executing query ie for insertion or deletion

Definition at line 57 of file database.cc.

**4.3.3.14** void Database::SelectColumn ( STRING\_VEC & *result*, string *column*, string *table*, string *where* = " " )

Select single column from table

Definition at line 128 of file database.cc.

**4.3.3.15** void Database::SelectProjectID ( string & *projectID* )

Select project ID from database.

Select last Project ID from table

#### Parameters

<i>projectID</i>	Unique project ID
------------------	-------------------

Definition at line 152 of file database.cc.

**4.3.3.16** Database::SelectQuery ( string *query*, STRING\_VEC & *result* )

Select Query for selecting single column from table.

Selecting column from table.

Select Query

#### Parameters

<i>query</i>	Select query
<i>result</i>	For storing selected column valued in result variable
<i>result</i>	
<i>column</i>	
<i>table</i>	
<i>where</i>	

Definition at line 75 of file database.cc.

**4.3.3.17** void Database::SelectSum ( string *column*, string *table*, string *projectID*, string & *result* )

Select sum of data entry in table of specific projectID

Definition at line 293 of file database.cc.

### 4.3.4 Member Data Documentation

**4.3.4.1** MYSQL\* Database::connect [protected]

MySQL connectivity Variables

Definition at line 41 of file database.h.

**4.3.4.2** unsigned int Database::numrows [protected]

unsigned int variable

Definition at line 46 of file database.h.

The documentation for this class was generated from the following files:

- src/header/database.h
- src/database.cc

## 4.4 Databaso Class Reference

The documentation for this class was generated from the following file:

- src/database.cc

## 4.5 DataSheet Class Reference

### 4.5.1 Detailed Description

Include datesheet.h file

The documentation for this class was generated from the following file:

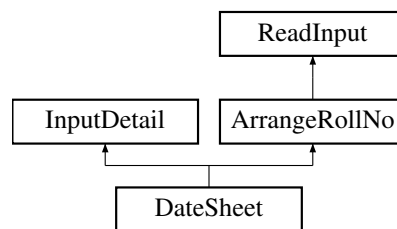
- src/backend/datesheet.cc

## 4.6 DateSheet Class Reference

[DateSheet](#) class for reading datesheet detail, class detail and create datesheet.out file for seating plan.

```
#include <datesheet.h>
```

Inheritance diagram for DateSheet:



### Public Member Functions

- [DateSheet](#) ()  
*Constructor.*
- void [AddRollNoWithDateSheet](#) ()  
*Adding roll no to with exam subject code in datesheet.*
- void [WriteDateSheet](#) (string [projectId](#))
- void [Main](#) (string pID)
- [~DateSheet](#) ()  
*DEstructor.*
- [DateSheet](#) ()
- void [ReadRollNoDetail](#) ()

- Reading roll no details filled by user.*
- void [WriteRollNoDetail](#) ()
- Writing rol no details into database.*
- void [SetDefaultValue](#) ()
- Setting default values for i/p field of datesheet.*
- void [DateSheetPage](#) ()
- [DateSheet](#) Page for taking I/P from user.*
- [~DateSheet](#) ()

## Additional Inherited Members

### 4.6.1 Detailed Description

[DateSheet](#) class for reading datesheet detail, class detail and create datesheet.out file for seating plan.

Datesheet class for taking datesheet detail from user.

Include [arrangerollno.h](#) header file

include datesheet.h

include [inputdetail.h](#)

Definition at line 27 of file datesheet.h.

### 4.6.2 Constructor & Destructor Documentation

#### 4.6.2.1 [DateSheet::DateSheet](#) ( )

Constructor.

Constructor

Definition at line 27 of file datesheet.cc.

#### 4.6.2.2 [DateSheet::~~DateSheet](#) ( )

DEstructor.

Desructor

Definition at line 161 of file datesheet.cc.

#### 4.6.2.3 [DateSheet::DateSheet](#) ( )

Constructor

#### 4.6.2.4 [DateSheet::~~DateSheet](#) ( )

Destructor

### 4.6.3 Member Function Documentation

#### 4.6.3.1 [void DateSheet::AddRollNoWithDateSheet](#) ( )

Adding roll no to with exam subject code in datesheet.

Adding roll nos with date sheet

Definition at line 38 of file datesheet.cc.

#### 4.6.3.2 void DateSheet::DateSheetPage ( )

[DateSheet](#) Page for taking I/P from user.

[DateSheet](#) page for taking I/P from user

Definition at line 181 of file datesheet.cc.

#### 4.6.3.3 void DateSheet::Main ( string pID )

Main function to call other functions

Reimplemented from [ArrangeRollNo](#).

Definition at line 143 of file datesheet.cc.

#### 4.6.3.4 void DateSheet::ReadRollNoDetail ( )

Reading roll no details filled by user.

ReadRollNo detail

Definition at line 43 of file datesheet.cc.

#### 4.6.3.5 void DateSheet::SetDefaultValue ( )

Setting default values for i/p field of datesheet.

Setting default values for datesheet page

Definition at line 165 of file datesheet.cc.

#### 4.6.3.6 void DateSheet::WriteDateSheet ( string projectID )

Writing datesheet.out file i.e. used for seating plan

Definition at line 99 of file datesheet.cc.

#### 4.6.3.7 void DateSheet::WriteRollNoDetail ( )

Writing roll no details into database.

Write roll no detail into database

Definition at line 112 of file datesheet.cc.

The documentation for this class was generated from the following files:

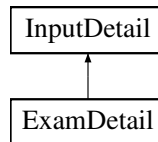
- src/backend/header/datesheet.h
- src/header/datesheet.h
- src/backend/datesheet.cc
- src/datesheet.cc

## 4.7 ExamDetail Class Reference

[ExamDetail](#) class for getting exam from user.

```
#include <exam.h>
```

Inheritance diagram for ExamDetail:



### Public Member Functions

- [ExamDetail](#) ()  
*Constructor.*
- void [ReadRoomDetail](#) ()  
*Reading room details from previous page.*
- void [WriteRoomDetail](#) ()  
*After Reading, saving details in [Database](#).*
- void [SetDefaultValue](#) ()  
*Setting Default values for exam fields if projecy is new or old project has empty detail in database.*
- void [ExamDetailPage](#) ()  
*Page for taking exam detail.*
- [~ExamDetail](#) ()  
*Destructor.*

### Additional Inherited Members

#### 4.7.1 Detailed Description

[ExamDetail](#) class for getting exam from user.

Include [exam.h](#)

Include [inputdetail.h](#)

Definition at line 29 of file exam.h.

#### 4.7.2 Constructor & Destructor Documentation

##### 4.7.2.1 ExamDetail::ExamDetail ( )

Constructor.

Constroctor

Definition at line 27 of file exam.cc.

##### 4.7.2.2 ExamDetail::~ExamDetail ( )

Destructor.

Destructor

Definition at line 380 of file exam.cc.

### 4.7.3 Member Function Documentation

#### 4.7.3.1 void ExamDetail::ExamDetailPage ( )

Page for taking exam detail.

Exam Detail Page

Definition at line 263 of file exam.cc.

#### 4.7.3.2 void ExamDetail::ReadRoomDetail ( )

Reading room details from previous page.

Reading room detail

Definition at line 105 of file exam.cc.

#### 4.7.3.3 void ExamDetail::SetDefaultValue ( )

Setting Default values for exam fields if proejcy is new or old project has empty detail in database.

Setting default values

Definition at line 47 of file exam.cc.

#### 4.7.3.4 void ExamDetail::WriteRoomDetail ( )

After Reading, saving details in [Database](#).

Writing room details into database

Definition at line 164 of file exam.cc.

The documentation for this class was generated from the following files:

- [src/header/exam.h](#)
- [src/exam.cc](#)

## 4.8 ExapandRollNo Class Reference

Class for expanding roll nos.

```
#include <expandrollno.h>
```

### 4.8.1 Detailed Description

Class for expanding roll nos.

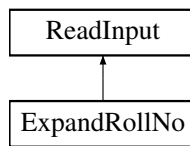
Include [readinput.h](#)

The documentation for this class was generated from the following file:

- [src/backend/header/expandrollno.h](#)

## 4.9 ExpandRollNo Class Reference

Inheritance diagram for ExpandRollNo:



### Public Member Functions

- [ExpandRollNo](#) ()
- `template<typename OutIter >`  
`bool ExpandRollNoList (istream &is, OutIter out)`  
*Template for expanding roll nos.*
- `void ExpandRollNos (string projectId)`  
*Expanding Roll nos.*
- `void WriteExpandRollNo (string projectId)`  
*Write expanded roll nos into file.*
- `~ExpandRollNo ()`  
*Destructor.*

### Protected Attributes

- `INT_VEC rollNoSize`
- `INT_VEC notIncludedRNoSize`

#### 4.9.1 Detailed Description

Definition at line 27 of file `expandrollno.h`.

#### 4.9.2 Constructor & Destructor Documentation

##### 4.9.2.1 `ExpandRollNo::ExpandRollNo ( )`

Constructor

include [expandrollno.h](#)

Definition at line 22 of file `expandrollno.cc`.

##### 4.9.2.2 `ExpandRollNo::~~ExpandRollNo ( )`

Destructor.

Destructor

Definition at line 195 of file `expandrollno.cc`.



### 4.9.3 Member Function Documentation

#### 4.9.3.1 `template<typename OutIter > bool ExpandRollNo::ExpandRollNoList ( istream & is, OutIter out )`

Template for expanding roll nos.

Template for expanding roll no

Definition at line 94 of file `expandrollno.cc`.

#### 4.9.3.2 `void ExpandRollNo::ExpandRollNos ( string projectID )`

Expanding Roll nos.

For expanding roll no and calling `ExpandRollNoList` template

##### Parameters

<i>projectID</i>	Project ID of user's project
------------------	------------------------------

Definition at line 34 of file `expandrollno.cc`.

#### 4.9.3.3 `void ExpandRollNo::WriteExpandRollNo ( string projectID )`

Write expanded roll nos into file.

Write expanded roll no's in file

##### Parameters

<i>projectID</i>	Project ID of user's project
------------------	------------------------------

Definition at line 149 of file `expandrollno.cc`.

The documentation for this class was generated from the following files:

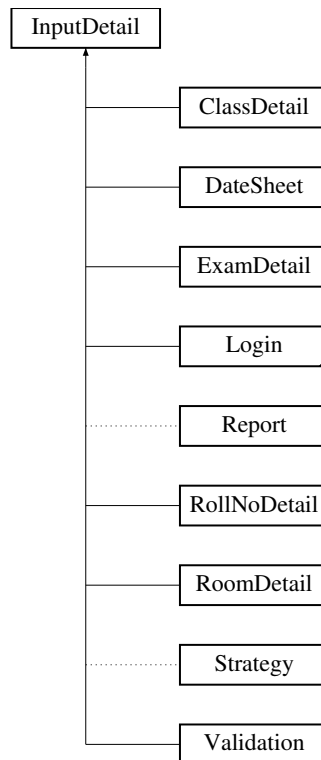
- `src/backend/header/expandrollno.h`
- `src/backend/expandrollno.cc`

## 4.10 InputDetail Class Reference

Declare common variables, functions that are used by derived class.

```
#include <inputdetail.h>
```

Inheritance diagram for `InputDetail`:



## Public Member Functions

- [InputDetail](#) ()
- void [Header](#) (string titleName)
- void [Footer](#) ()
- string [IntToString](#) (int value)
- int [StringToInt](#) (string value)
- void [ErrorMessage](#) (string msg="")
- string [Time](#) ()
- void [SplitString](#) (STRING\_VEC &output, string input, string token)  
*Split string using boost library func.*
- string [FileName](#) (string file, string [projectID](#), int fileType)  
*For creating filename w.r.t to projectID.*

## Protected Attributes

- [InputFieldName](#) fieldName
- [ReadInputField](#) readField
- [Database](#) database
- [PageLayout](#) page
- int **i**
- int **j**
- int **k**
- int **l**
- stringstream **ss**
- string **temp**
- string **temp1**
- string [projectID](#)
- string [emailID](#)

- string `projectName`
- string `fileName`
- string `where`
- string `msg`
- STRING\_VEC `vecTemp`
- STRING\_VEC `oldProject`
- STRING\_VEC `splitString`
- ifstream `inFile`
- ofstream `outFile`
- string `sessionID`
- int `totalClasses`
- int `totalCols`
- int `totalRows`
- int `totalDays`
- INT\_VEC `totalCentres`
- string `projectType`
- string `sameDetail`
- STRING\_VEC `className`
- STRING\_VEC `subjectName`
- STRING\_VEC `subjectCode`
- STRING\_VEC `classID`
- STRING\_VEC `tableHeading`
- STRING\_VEC `prefix`
- STRING\_VEC `startRollNo`
- STRING\_VEC `endRollNo`
- STRING\_VEC `notIncluded`
- STRING\_VEC `date`
- STRING\_VEC `examCode`
- STRING\_VEC `examName`
- STRING\_VEC `examSession`
- STRING\_VEC `examVenue`
- STRING\_VEC `examTime`
- STRING\_VEC `strategyChoice`
- STRING\_2DVEC `centreName`
- STRING\_2DVEC `roomNo`
- STRING\_2DVEC `rows`
- STRING\_2DVEC `columns`
- INT\_VEC **`totalSubjects`**
- INT\_2DVEC **`totalRooms`**
- STRING\_2DVEC **`subCode`**
- STRING\_2DVEC **`subName`**
- STRING\_3DVEC **`rmNo`**
- STRING\_3DVEC **`row`**
- STRING\_3DVEC **`col`**

#### 4.10.1 Detailed Description

Declare common variables, functions that are used by derived class.

Include Local header files

Definition at line 40 of file inputdetail.h.

## 4.10.2 Constructor & Destructor Documentation

### 4.10.2.1 InputDetail::InputDetail ( )

Constructor

Include detail.h file

Definition of member functions

Class: [InputDetail](#) Method: [InputDetail](#) :: [InputDetail\(\)](#)

Description: Constructor

Definition at line 37 of file inputdetail.cc.

## 4.10.3 Member Function Documentation

### 4.10.3.1 void InputDetail::ErrorMessage ( string msg = " " )

Shows message if user filled data in field

Definition at line 109 of file inputdetail.cc.

### 4.10.3.2 string InputDetail::FileName ( string file, string projectID, int fileType )

For creating filename w.r.t to projectID.

Retruning filename w.r.t. to project ID

Parameters

<i>projectID</i>	Unique ID i.e. used to read file of user
<i>file</i>	I/O file Name
<i>fileType</i>	For defining file type i.e I/P or O/P file. If fileType == 1, Input file If fileType == 0(else), Output File

Returns

fileName This function will return file name with project ID

Definition at line 171 of file inputdetail.cc.

### 4.10.3.3 void InputDetail::Footer ( )

Footer of page

Class: [InputDetail](#) Method: [InputDetail](#) :: [Footer\(\)](#)

Description: Common for all pages(end body, html)

Definition at line 68 of file inputdetail.cc.

4.10.3.4 void InputDetail::Header ( string *titleName* )

Header of page

Class: [InputDetail](#) Method: [InputDetail :: Header\(string titleName\)](#)

Description: common header for all pages(start html, head, body)

Definition at line 50 of file inputdetail.cc.

4.10.3.5 string InputDetail::IntToString ( int *value* )

Convert Integer to string

Class: [InputDetail](#) Method: [InputDetail :: IntToString\(int value\)](#)

Description: Convert Int to String using stringstream object

Definition at line 82 of file inputdetail.cc.

4.10.3.6 void InputDetail::SplitString ( [STRING\\_VEC](#) & *output*, string *input*, string *token* )

Split string using boost library func.

Splitting string

Parameters

<i>output</i>	Stores result of splitted string
<i>input</i>	string for splitting
<i>token</i>	eg , or . or - any char to split string

Definition at line 149 of file inputdetail.cc.

4.10.3.7 int InputDetail::StringToInt ( string *value* )

Convert String to Integer

Class: [ReadInputField](#) Method: [ReadInputField :: StringToInt\(string value\)](#)

Description: Converts string value to integer

Definition at line 97 of file inputdetail.cc.

## 4.10.3.8 string InputDetail::Time ( )

For locating current time on system

Definition at line 126 of file inputdetail.cc.

## 4.10.4 Member Data Documentation

4.10.4.1 [STRING\\_2DVEC](#) InputDetail::centreName [protected]

Centre Name

Definition at line 113 of file inputdetail.h.

#### 4.10.4.2 **STRING\_VEC InputDetail::classID** [protected]

Class Details ID

Definition at line 89 of file inputdetail.h.

#### 4.10.4.3 **STRING\_VEC InputDetail::className** [protected]

Class/Branch Name

Definition at line 89 of file inputdetail.h.

#### 4.10.4.4 **STRING\_2DVEC InputDetail::columns** [protected]

columns of room

Definition at line 113 of file inputdetail.h.

#### 4.10.4.5 **Database InputDetail::database** [protected]

Accessing [Database](#)

Definition at line 49 of file inputdetail.h.

#### 4.10.4.6 **STRING\_VEC InputDetail::date** [protected]

Date of examination

Definition at line 89 of file inputdetail.h.

#### 4.10.4.7 **string InputDetail::emailID** [protected]

Email id

Definition at line 57 of file inputdetail.h.

#### 4.10.4.8 **STRING\_VEC InputDetail::endRollNo** [protected]

ending roll no

Definition at line 89 of file inputdetail.h.

#### 4.10.4.9 **STRING\_VEC InputDetail::examCode** [protected]

Exam/subject code for seating plan

Definition at line 89 of file inputdetail.h.

#### 4.10.4.10 **STRING\_VEC InputDetail::examName** [protected]

Examination name

Definition at line 89 of file inputdetail.h.

**4.10.4.11** `STRING_VEC InputDetail::examSession` `[protected]`

Session Morning/Evening

Definition at line 89 of file inputdetail.h.

**4.10.4.12** `STRING_VEC InputDetail::examTime` `[protected]`

Exam. Time

Definition at line 89 of file inputdetail.h.

**4.10.4.13** `STRING_VEC InputDetail::examVenue` `[protected]`

Exam Venue

Definition at line 89 of file inputdetail.h.

**4.10.4.14** `InputFieldName InputDetail::fieldName` `[protected]`

I/P field name

Definition at line 47 of file inputdetail.h.

**4.10.4.15** `string InputDetail::fileName` `[protected]`

file name for opening file

Definition at line 61 of file inputdetail.h.

**4.10.4.16** `ifstream InputDetail::inFile` `[protected]`

For Reading file

Definition at line 72 of file inputdetail.h.

**4.10.4.17** `int InputDetail::l` `[protected]`

Looping Variables

Definition at line 52 of file inputdetail.h.

**4.10.4.18** `string InputDetail::msg` `[protected]`

For Displaying error msg

Definition at line 61 of file inputdetail.h.

**4.10.4.19** `STRING_VEC InputDetail::notIncluded` `[protected]`

roll no's that are not included in seating plan

Definition at line 89 of file inputdetail.h.

#### 4.10.4.20 **STRING\_VEC InputDetail::oldProject** [protected]

For storing old projects if any

Definition at line 66 of file inputdetail.h.

#### 4.10.4.21 **ofstream InputDetail::outFile** [protected]

For writing file

Definition at line 73 of file inputdetail.h.

#### 4.10.4.22 **PageLayout InputDetail::page** [protected]

HTML Tags

Definition at line 50 of file inputdetail.h.

#### 4.10.4.23 **STRING\_VEC InputDetail::prefix** [protected]

Prefix of roll no

Definition at line 89 of file inputdetail.h.

#### 4.10.4.24 **string InputDetail::projectID** [protected]

Project ID

Definition at line 57 of file inputdetail.h.

#### 4.10.4.25 **string InputDetail::projectName** [protected]

Project Name

Definition at line 57 of file inputdetail.h.

#### 4.10.4.26 **string InputDetail::projectType** [protected]

Project Type Old/New

Definition at line 84 of file inputdetail.h.

#### 4.10.4.27 **ReadInputField InputDetail::readField** [protected]

Read I/P Field

Definition at line 48 of file inputdetail.h.

#### 4.10.4.28 **STRING\_2DVEC InputDetail::roomNo** [protected]

Room no. in one centre

Definition at line 113 of file inputdetail.h.



**4.10.4.29** `STRING_2DVEC InputDetail::rows` `[protected]`

Rows of room

Definition at line 113 of file inputdetail.h.

**4.10.4.30** `string InputDetail::sameDetail` `[protected]`

For holding value yes/no for same detail for rest of pages

Definition at line 84 of file inputdetail.h.

**4.10.4.31** `string InputDetail::sessionID` `[protected]`

Session ID

Definition at line 74 of file inputdetail.h.

**4.10.4.32** `STRING_VEC InputDetail::splitString` `[protected]`

For storing values of splitted string

Definition at line 66 of file inputdetail.h.

**4.10.4.33** `stringstream InputDetail::ss` `[protected]`

for converting int to string

Definition at line 54 of file inputdetail.h.

**4.10.4.34** `STRING_VEC InputDetail::startRollNo` `[protected]`

starting roll no

Definition at line 89 of file inputdetail.h.

**4.10.4.35** `STRING_VEC InputDetail::strategyChoice` `[protected]`

[Strategy](#) name selected by user

Definition at line 89 of file inputdetail.h.

**4.10.4.36** `STRING_VEC InputDetail::subjectCode` `[protected]`

Subject Codes

Definition at line 89 of file inputdetail.h.

**4.10.4.37** `STRING_VEC InputDetail::subjectName` `[protected]`

Subject Names

Definition at line 89 of file inputdetail.h.

#### 4.10.4.38 `STRING_VEC InputDetail::tableHeading` [protected]

For table heading

Definition at line 89 of file inputdetail.h.

#### 4.10.4.39 `string InputDetail::temp1` [protected]

for temporary storage

Definition at line 55 of file inputdetail.h.

#### 4.10.4.40 `INT_VEC InputDetail::totalCentres` [protected]

No of centres for exam

Definition at line 82 of file inputdetail.h.

#### 4.10.4.41 `int InputDetail::totalClasses` [protected]

No. of classes for seating plan

Definition at line 77 of file inputdetail.h.

#### 4.10.4.42 `int InputDetail::totalCols` [protected]

total Columns in table

Definition at line 77 of file inputdetail.h.

#### 4.10.4.43 `int InputDetail::totalDays` [protected]

No of days in datesheet

Definition at line 77 of file inputdetail.h.

#### 4.10.4.44 `int InputDetail::totalRows` [protected]

total rows in table

Definition at line 77 of file inputdetail.h.

#### 4.10.4.45 `STRING_VEC InputDetail::vecTemp` [protected]

string Vector temporary use

Definition at line 66 of file inputdetail.h.

#### 4.10.4.46 `string InputDetail::where` [protected]

Temp variable to store where clause

Definition at line 61 of file inputdetail.h.

The documentation for this class was generated from the following files:

- `src/header/inputdetail.h`
- `src/inputdetail.cc`

## 4.11 InputFieldName Class Reference

```
#include <inputfieldname.h>
```

### Public Member Functions

- [InputFieldName](#) ()
- void [SetFieldNames](#) ()

### Public Attributes

- string [className](#)
- string [totalClasses](#)
- string [totalSubjects](#)
- string [subjectCode](#)
- string [subjectName](#)
- string [classID](#)
- string [prefix](#)
- string [startRollNo](#)
- string [endRollNo](#)
- string [notIncluded](#)
- string [projectName](#)
- string [projectID](#)
- string [projectType](#)
- string [emailID](#)
- string [password](#)
- string [retypePassword](#)
- string [key](#)
- string [totalCentres](#)
- string [totalRooms](#)
- string [centreName](#)
- string [roomNo](#)
- string [rows](#)
- string [columns](#)
- string [strategyChoice](#)
- string [examName](#)
- string [examDate](#)
- string [examSession](#)
- string [examTime](#)
- string [examVenue](#)
- string [date](#)
- string [examCode](#)
- string [totalDays](#)
- string [sameDetail](#)

### 4.11.1 Detailed Description

Include local header file that has global header files

Class: [InputFieldName](#) Description: This class has variable names for all fields that

are used to take input from user.

Definition at line 37 of file inputfieldname.h.

## 4.11.2 Constructor & Destructor Documentation

### 4.11.2.1 InputFieldName::InputFieldName ( )

Constructor

Include header file that contains [InputFieldName](#) Class's

Declaration

Definition of member functions

Class: [InputFieldName](#) Method: [InputFieldName](#) :: [InputFieldName\(\)](#)

Description: Constructor calls [SetFieldNames\(\)](#)

Definition at line 38 of file inputfieldname.cc.

## 4.11.3 Member Function Documentation

### 4.11.3.1 void InputFieldName::SetFieldNames ( )

Set Field Values

Class: [InputFieldName](#) Method: [InputFieldName](#) :: [SetFieldNames\(\)](#) Description: Assign Values of field names in its respective

variables.

Definition at line 52 of file inputfieldname.cc.

## 4.11.4 Member Data Documentation

### 4.11.4.1 string InputFieldName::centreName

Center Name

Definition at line 63 of file inputfieldname.h.

### 4.11.4.2 string InputFieldName::classID

Class id

Definition at line 44 of file inputfieldname.h.

### 4.11.4.3 string InputFieldName::className

ClassName

Definition at line 44 of file inputfieldname.h.

#### 4.11.4.4 string InputFieldName::columns

Columns in room

Definition at line 63 of file inputfieldname.h.

#### 4.11.4.5 string InputFieldName::date

Date of examination

Definition at line 63 of file inputfieldname.h.

#### 4.11.4.6 string InputFieldName::emailID

Email Field

Definition at line 63 of file inputfieldname.h.

#### 4.11.4.7 string InputFieldName::endRollNo

ending rollno

Definition at line 44 of file inputfieldname.h.

#### 4.11.4.8 string InputFieldName::examCode

Exam code accord to date sheet

Definition at line 63 of file inputfieldname.h.

#### 4.11.4.9 string InputFieldName::examDate

Examination Date

Definition at line 63 of file inputfieldname.h.

#### 4.11.4.10 string InputFieldName::examName

Examination Name

Definition at line 63 of file inputfieldname.h.

#### 4.11.4.11 string InputFieldName::examSession

Exam. Session (M/E)

Definition at line 63 of file inputfieldname.h.

#### 4.11.4.12 string InputFieldName::examTime

Examination Time

Definition at line 63 of file inputfieldname.h.

**4.11.4.13 string InputFieldName::examVenue**

Examination Venue

Definition at line 63 of file inputfieldname.h.

**4.11.4.14 string InputFieldName::key**

Registration Key

Definition at line 63 of file inputfieldname.h.

**4.11.4.15 string InputFieldName::notIncluded**

roll that are not included in seating plan

Definition at line 44 of file inputfieldname.h.

**4.11.4.16 string InputFieldName::password**

Password field

Definition at line 63 of file inputfieldname.h.

**4.11.4.17 string InputFieldName::prefix**

Prefix of rollno

Definition at line 44 of file inputfieldname.h.

**4.11.4.18 string InputFieldName::projectID**

Project ID

Definition at line 44 of file inputfieldname.h.

**4.11.4.19 string InputFieldName::projectName**

Project Name

Definition at line 44 of file inputfieldname.h.

**4.11.4.20 string InputFieldName::projectType**

ProjectType ie new or old

Definition at line 44 of file inputfieldname.h.

**4.11.4.21 string InputFieldName::retypePassword**

Retype Password

Definition at line 63 of file inputfieldname.h.

**4.11.4.22 string InputFieldName::roomNo**

Room Nos w.r.t to centre name

Definition at line 63 of file inputfieldname.h.

**4.11.4.23 string InputFieldName::rows**

Rows of room

Definition at line 63 of file inputfieldname.h.

**4.11.4.24 string InputFieldName::sameDetail**

Same detail of room, exam, strategy for seating plan

Definition at line 63 of file inputfieldname.h.

**4.11.4.25 string InputFieldName::startRollNo**

starting roll no

Definition at line 44 of file inputfieldname.h.

**4.11.4.26 string InputFieldName::strategyChoice**

[Strategy](#) option selected by user

Definition at line 63 of file inputfieldname.h.

**4.11.4.27 string InputFieldName::subjectCode**

SubjectCode

Definition at line 44 of file inputfieldname.h.

**4.11.4.28 string InputFieldName::subjectName**

SubjectName

Definition at line 44 of file inputfieldname.h.

**4.11.4.29 string InputFieldName::totalCentres**

Total centres

Definition at line 63 of file inputfieldname.h.

**4.11.4.30 string InputFieldName::totalClasses**

TotalClasses

Definition at line 44 of file inputfieldname.h.

#### 4.11.4.31 `string InputFieldName::totalDays`

Total days in datesheet

Definition at line 63 of file `inputfieldname.h`.

#### 4.11.4.32 `string InputFieldName::totalRooms`

Total Rooms w.r.t centers name

Definition at line 63 of file `inputfieldname.h`.

#### 4.11.4.33 `string InputFieldName::totalSubjects`

TotalSubjects

Definition at line 44 of file `inputfieldname.h`.

The documentation for this class was generated from the following files:

- `src/header/inputfieldname.h`
- `src/inputfieldname.cc`

## 4.12 JavaScript Class Reference

```
#include <javascript.h>
```

### Public Member Functions

- [JavaScript](#) ()
- void [OnFocusEvent](#) ()
- void [OnChangeEvent](#) ()

#### 4.12.1 Detailed Description

Include header.h file

Class: [JavaScript](#)

**Description:** For creating .js files

Definition at line 36 of file `javascript.h`.

#### 4.12.2 Constructor & Destructor Documentation

##### 4.12.2.1 `JavaScript::JavaScript ( )`

Constructor



### 4.12.3 Member Function Documentation

#### 4.12.3.1 void JavaScript::OnChangeEvent ( )

onchange func. of javascript for i/p fields

#### 4.12.3.2 void JavaScript::OnFocusEvent ( )

For input fields(onfocus action)

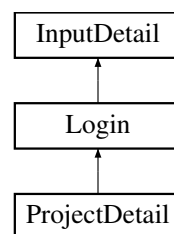
The documentation for this class was generated from the following file:

- src/header/javascript.h

## 4.13 Login Class Reference

```
#include <login.h>
```

Inheritance diagram for Login:



### Public Member Functions

- [Login](#) ()  
*Constructor.*
- void [LoginPage](#) (string [msg](#)="", string [emailID](#)="abc@you.com", string [password](#)="\*\*\*\*\*")  
*Function for creating login Page.*
- void [ReadLoginDetail](#) ()  
*Reading login detail from text fields filled by user.*
- void [RegistrationPage](#) (string [msg](#)="", string [emailID](#)="abc@you.com")  
*For registering new user.*
- void [NewUser](#) ()  
*Adding new user into database and if user already exists then move back to register page for unique email id.*
- void [SelectLoginDetail](#) ()  
*For reading email Id and password from User table in database.*
- void [ConfirmPage](#) (string [msg](#)="", string [password](#)="\*\*\*\*\*", string [retypePassword](#)="\*\*\*\*\*")  
*Page for validating email ID and setting user's password.*
- void [AddUser](#) ()  
*Add New user.*
- void [LogoutPage](#) ()  
*Page for logging out.*
- [~Login](#) ()

## Protected Attributes

- `STRING_VEC` [emailID](#)
- `STRING_VEC` [password](#)
- `STRING_VEC` [regKey](#)
- `string` [userEmailID](#)
- `string` [userPassword](#)
- `string` [retypePassword](#)
- `string` [currentTime](#)
- `string` [key](#)
- [SendMail](#) [sendMail](#)

### 4.13.1 Detailed Description

Include required header files

Class: [Login](#)

Description: [Login](#) class for user login

Include [login.h](#) header file

Definition at line 38 of file [login.h](#).

### 4.13.2 Constructor & Destructor Documentation

#### 4.13.2.1 `Login::Login ( )`

Comstructor.

Constructor

Definition at line 32 of file [login.cc](#).

#### 4.13.2.2 `Login::~~Login ( )` `[inline]`

Destructor

Definition at line 87 of file [login.h](#).

### 4.13.3 Member Function Documentation

#### 4.13.3.1 `void Login::ConfirmPage ( string msg = " ", string password = "*****", string retypePassword = "*****" )`

Page for validating email ID and setting user's password.

Parameters

<i>msg</i>	Shows error message
<i>password</i>	Password filled by user
<i>retypePassword</i>	Password filled by user

Definition at line 237 of file [login.cc](#).

**4.13.3.2** `void Login::LoginPage ( string msg = " ", string emailID = "abc@you.com", string password = "*****" )`

Function for creating login Page.

Creating login page

#### Parameters

<i>msg</i>	Show Message if emailID/ password incorrect
<i>emailID</i>	user filled email id
<i>password</i>	uer password

Definition at line 87 of file login.cc.

**4.13.3.3** `void Login::LogoutPage ( )`

Page for logging out.

Logout Page

Definition at line 348 of file login.cc.

**4.13.3.4** `void Login::NewUser ( )`

Adding new user into database and if user already exists then move back to register page for unique email id.

Add new user in database

Definition at line 170 of file login.cc.

**4.13.3.5** `void Login::ReadLoginDetail ( )`

Reading login detail from text fields filled by user.

Read [Login](#) Detail

#### Parameters

--	--

Definition at line 71 of file login.cc.

**4.13.3.6** `void Login::RegistrationPage ( string msg = " ", string emailID = "abc@you.com" )`

For registering new user.

Register user page

Definition at line 129 of file login.cc.

**4.13.3.7** `void Login::SelectLoginDetail ( )`

For reading email Id and password from User table in database.

For selecting email and password from user table

Definition at line 44 of file login.cc.

## 4.13.4 Member Data Documentation

#### 4.13.4.1 `string Login::currentTime` [protected]

Current Time

Definition at line 51 of file login.h.

#### 4.13.4.2 `STRING_VEC Login::emailID` [protected]

For storing email and password of users Email ID as vector

Definition at line 43 of file login.h.

#### 4.13.4.3 `STRING_VEC Login::password` [protected]

password as vector variable

Definition at line 43 of file login.h.

#### 4.13.4.4 `STRING_VEC Login::regKey` [protected]

Registration key

Definition at line 43 of file login.h.

#### 4.13.4.5 `string Login::retypePassword` [protected]

For reading password

Definition at line 47 of file login.h.

#### 4.13.4.6 `string Login::userEmailID` [protected]

For reading emailID in text field

Definition at line 47 of file login.h.

#### 4.13.4.7 `string Login::userPassword` [protected]

For reading password in text field

Definition at line 47 of file login.h.

The documentation for this class was generated from the following files:

- src/header/login.h
- src/login.cc

## 4.14 `jwsntp::mailer` Class Reference

### Classes

- struct **Address**

## Public Types

- enum {  
    **TO**, **Cc**, **Bcc**, **SMTP\_PORT** = 25,  
    **DNS\_PORT** = 53 }
- enum **authtype** { **LOGIN** = 1, **PLAIN** }

## Public Member Functions

- **mailer** (const char \*TOaddress, const char \*FROMaddress, const char \*Subject, const std::vector< char > &Message, const char \*server="127.0.0.1", unsigned short Port=SMTP\_PORT, bool MXLookup=true)
- **mailer** (const char \*TOaddress, const char \*FROMaddress, const char \*Subject, const char \*Message, const char \*server="127.0.0.1", unsigned short Port=SMTP\_PORT, bool MXLookup=true)
- **mailer** (bool MXLookup=false, unsigned short Port=SMTP\_PORT)
- void **operator()** ()
- void **send** ()
- bool **attach** (const std::string &filename)
- bool **removeattachment** (const std::string &filename)
- bool **setmessage** (const std::string &newmessage)
- bool **setmessage** (const std::vector< char > &newmessage)
- bool **setmessageHTML** (const std::string &newmessage)
- bool **setmessageHTML** (const std::vector< char > &newmessage)
- bool **setmessageHTMLfile** (const std::string &filename)
- bool **setsubject** (const std::string &newSubject)
- bool **setserver** (const std::string &nameserver\_or\_smtpserver)
- bool **setsender** (const std::string &newsender)
- bool **addrecipient** (const std::string &newrecipient, short recipient\_type=TO)
- bool **removerecipient** (const std::string &recipient)
- void **clearrecipients** ()
- void **clearattachments** ()
- void **reset** ()
- const std::string & **response** () const
- void **authtype** (const enum authtype Type)
- void **username** (const std::string &User)
- void **password** (const std::string &Pass)

## Public Attributes

- enum jwsmtplib::mailer::authtype **type**

## Static Public Attributes

- static const enum  
jwsmtplib::mailer:: { ... } **consts**

### 4.14.1 Detailed Description

Definition at line 33 of file mailer.h.

The documentation for this class was generated from the following files:

- src/header/jwsmtplib/mailer.h
- src/header/jwsmtplib/mailer.cpp

## 4.15 MD5 Class Reference

### Public Types

- typedef unsigned int **size\_type**

### Public Member Functions

- **MD5** (const std::string &text)
- void **update** (const unsigned char \*buf, size\_type length)
- void **update** (const char \*buf, size\_type length)
- **MD5** & **finalize** ()
- std::string **hexdigest** () const

### Friends

- std::ostream & **operator**<< (std::ostream &, **MD5** md5)

#### 4.15.1 Detailed Description

Definition at line 69 of file md5.h.

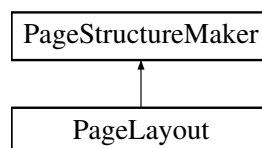
The documentation for this class was generated from the following files:

- src/header/md5.h
- src/md5.cc

## 4.16 PageLayout Class Reference

```
#include <pagelayout.h>
```

Inheritance diagram for PageLayout:



### Public Member Functions

- **PageLayout** ()
- void **Header** ()
- void **Footer** ()
- void **Menu** ()
- void **Logo** (string logoName)
- void **Head** (string titleName)
- void **SetCookies** (string emailID, string sessionID)
- void **ContentType** ()
- void **LogoutLink** ()

## Protected Attributes

- string **projectName**

## Additional Inherited Members

### 4.16.1 Detailed Description

Include required header files

Class: [PageLayout](#) : public [PageStructureMaker](#)

**Description:** For adding header and footer to page

Definition at line 36 of file pagelayout.h.

### 4.16.2 Constructor & Destructor Documentation

#### 4.16.2.1 PageLayout::PageLayout ( )

Constructor

Include local header file

Function Definitions of [PageLayout](#) Class

Class: [PageLayout](#) Method: [PageLayout](#) :: [PageLayout\(\)](#)

**Description:** Constructor

Definition at line 38 of file pagelayout.cc.

### 4.16.3 Member Function Documentation

#### 4.16.3.1 void PageLayout::ContentType ( )

Content-Type header

---

Class: [PageLayout](#)

Method: [PageLayout](#) :: [ContextType](#)

**Description:** Setting content-type header

Definition at line 166 of file pagelayout.cc.

#### 4.16.3.2 void PageLayout::Footer ( )

Footer Section of Page

Class: [PageLayout](#) Method: [PageLayout](#) :: [Footer\(\)](#)

Description: Footer of pages

Definition at line 196 of file pagelayout.cc.

#### 4.16.3.3 void PageLayout::Head ( string *titleName* )

Head Section of Page

Class: [PageLayout](#) Method: [PageLayout :: Head\(string titleName\)](#) Description: Head Section of page, titlaName pass to Title

function.

Definition at line 106 of file pagelayout.cc.

#### 4.16.3.4 void PageLayout::Header ( )

Header Section of Page

Class: [PageLayout](#) Method: [PageLayout :: Header\(\)](#)

Description: Call Menu Function

Definition at line 89 of file pagelayout.cc.

#### 4.16.3.5 void PageLayout::Logo ( string *logoName* )

Logo on Page

Class: [PageLayout](#) Method: [PageLayout :: Logo\(string logoName\)](#)

Description: Logo to project

Definition at line 74 of file pagelayout.cc.

#### 4.16.3.6 void PageLayout::LogoutLink ( )

Logout link

---

Class: [PageLayout](#)

Method: [PageLayout :: LogOutLink\(\)](#)

Description: logout link

Definition at line 179 of file pagelayout.cc.

#### 4.16.3.7 void PageLayout::Menu ( )

List/Menu Navigation()

Class: [PageLayout](#) Method: [PageLayout :: Menu\(\)](#)



Description: Add Menu on pages(for navigation)

Definition at line 51 of file pagelayout.cc.

4.16.3.8 void PageLayout::SetCookies ( string userEmailID, string sessionID )

For Cookies after login

Class: [PageLayout](#)

Method: [PageLayout](#) :: SetCookies(string sessionID)

Description: Setting session ID in cookies

Definition at line 152 of file pagelayout.cc.

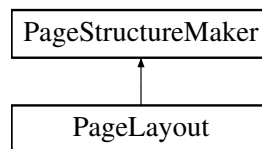
The documentation for this class was generated from the following files:

- src/header/pagelayout.h
- src/pagelayout.cc

## 4.17 PageStructureMaker Class Reference

```
#include <pagestructure.h>
```

Inheritance diagram for PageStructureMaker:



### Public Member Functions

- [PageStructureMaker](#) ()
- void [SetHTMLVariables](#) ()
- void [HTMLStart](#) ()
- void [HTMLEnd](#) ()
- void [HeadStart](#) ()
- void [HeadEnd](#) ()
- void [Title](#) (string pageTitle)
- void [CSS](#) (string href)
- void [Javascript](#) (string src)
- void [BodyStart](#) ()
- void [BodyEnd](#) ()
- void [DivStart](#) (string id, string className)
- void [DivEnd](#) ()
- void [FormStart](#) (string name, string action, string method)
- void [FormEnd](#) ()
- void [TableStart](#) (string id, string className)
- void [TableEnd](#) ()
- void [ListStart](#) (string listType)

- void [ListEnd](#) (string listType)
- void [ListItem](#) (string listItem)
- void [Anchor](#) (string href, string target)
- void [InputField](#) (string type, string name, string placeholder, string value="")
- void [InputField](#) (string type, string name, int nameNo, string placeholder, string value="")
- void [Label](#) (string forField, string value)
- void [SelectFieldStart](#) (string name)
- void [SelectFieldEnd](#) ()
- void [SelectOptionStart](#) (string value, string selected)
- void [SelectOptionEnd](#) ()
- void [Button](#) (string id, string type, string className, string value, string onClick="")

## Public Attributes

- string [startH1](#)
- string [endH1](#)
- string [startH3](#)
- string [endH3](#)
- string [startTD](#)
- string [endTD](#)
- string [startTH](#)
- string [endTH](#)
- string [startTR](#)
- string [endTR](#)
- string [startB](#)
- string [endB](#)
- string [brk](#)

### 4.17.1 Detailed Description

#### Include Header file

Class: [PageStructureMaker](#)

#### Description: Declaration

Definition at line 34 of file pagestructure.h.

### 4.17.2 Constructor & Destructor Documentation

#### 4.17.2.1 PageStructureMaker::PageStructureMaker ( )

Constructor

Include htmltags.h header files

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: [PageStructureMaker\(\)](#)

#### Description: Constructor

Definition at line 33 of file pagestructure.cc.

### 4.17.3 Member Function Documentation

#### 4.17.3.1 void PageStructureMaker::Anchor ( string href, string target )

Anchor Tag

---

Class: [PageStructureMaker](#)

Method: [PageStructureMaker](#) :: Anchor(string href)

Description: Anchor Tag

Definition at line 340 of file pagestructure.cc.

#### 4.17.3.2 void PageStructureMaker::BodyEnd ( )

Display </BODY>

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: BodyEnd()

Description: Display </BODY>

Definition at line 180 of file pagestructure.cc.

#### 4.17.3.3 void PageStructureMaker::BodyStart ( )

Display <BODY>

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: BodyStart()

Description: Display <BODY>

Definition at line 166 of file pagestructure.cc.

#### 4.17.3.4 void PageStructureMaker::Button ( string id, string type, string className, string value, string onClick = " " )

Button

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: Button(string id, string type, string className, string value)

Description: Create button(next, submit, etc)

Definition at line 497 of file pagestructure.cc.

#### 4.17.3.5 void PageStructureMaker::CSS ( string href )

Add External CSS

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: CSS(string href)

Description: Add External CSS file

Definition at line 138 of file pagestructure.cc.

4.17.3.6 void PageStructureMaker::DivEnd ( )

End div section

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: DivEnd()

Description: Close Div Section

Definition at line 225 of file pagestructure.cc.

4.17.3.7 void PageStructureMaker::DivStart ( string *id*, string *className* )

Start Div Section

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: DivStart(string id, string className)

Description: Start Div Section with id and className(for CSS)

Definition at line 211 of file pagestructure.cc.

4.17.3.8 void PageStructureMaker::FormEnd ( )

End Form

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: FormEnd()

Description: Close Form

Definition at line 257 of file pagestructure.cc.

4.17.3.9 void PageStructureMaker::FormStart ( string *name*, string *action*, string *method* )

Start Form

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: FormStart(string name, string action, string method)

Description: Start Form with name, action and method(GET/POST)

Definition at line 239 of file pagestructure.cc.

4.17.3.10 void PageStructureMaker::HeadEnd ( )

Display </HEAD>

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: HeadEnd()

Description: Display </HEAD>

Definition at line 112 of file pagestructure.cc.

## 4.17.3.11 void PageStructureMaker::HeadStart ( )

Display <HEAD>

Class: [PageStructureMaker](#) Method: [PageStructureMaker :: HeadStart\(\)](#)

Description: Display <HEAD>

Definition at line 99 of file pagestructure.cc.

## 4.17.3.12 void PageStructureMaker::HTMLEnd ( )

Display </HTML>

Class: [PageStructureMaker](#) Method: [PageStructureMaker :: HTMLEnd\(\)](#)

Description: Display </HTML>

Definition at line 86 of file pagestructure.cc.

## 4.17.3.13 void PageStructureMaker::HTMLStart ( )

Display <HTML>

Class: [PageStructureMaker](#) Method: [PageStructureMaker :: HTMLStart\(\)](#)

Description: Display <HTML> Tag

Definition at line 73 of file pagestructure.cc.

## 4.17.3.14 void PageStructureMaker::InputField ( string type, string name, string placeholder, string value = " " )

Input field with 3 arguments

---

Class: [PageStructureMaker](#)

Method: [PageStructureMaker :: InputField\(string type, string name, string value\)](#)

Description: For creating input field with 3 arguments

Definition at line 369 of file pagestructure.cc.

## 4.17.3.15 void PageStructureMaker::InputField ( string type, string name, int nameNo, string placeholder, string value = " " )

Input Field with 4 arguments

Class: [PageStructureMaker](#) Method: [PageStructureMaker :: InputField\(string type, string name, string value\)](#)

Description: Create Input fields like text field, submit

button, etc.

Definition at line 409 of file pagestructure.cc.

#### 4.17.3.16 void PageStructureMaker::Javascript ( string src )

Add Javascript File

Class: [PageStructureMaker](#) Method: [PageStructureMaker :: Javascript\(string src\)](#)

Description: Add external Javascript file

Definition at line 153 of file pagestructure.cc.

#### 4.17.3.17 void PageStructureMaker::Label ( string forField, string value )

Label tag

---

Class: [PageStructureMaker](#)

Method: [PageStructureMaker :: Label\(string forValue, string value\)](#)

Description: For adding label in html page

Definition at line 354 of file pagestructure.cc.

#### 4.17.3.18 void PageStructureMaker::ListEnd ( string listType )

End List

---

Class: [PageStructureMaker](#)

Method: [PageStructureMaker :: ListEnd\(string listType\)](#)

Description: Close List Tag

Definition at line 314 of file pagestructure.cc.

#### 4.17.3.19 void PageStructureMaker::ListItem ( string listItem )

List Item

---

Class: [PageStructureMaker](#)

Method: [PageStructureMaker :: ListItem\(string listItem\)](#)

Description: List Item

Definition at line 327 of file pagestructure.cc.

#### 4.17.3.20 void PageStructureMaker::ListStart ( string listType )

Start List

Class: [PageStructureMaker](#)

Method: [PageStructureMaker](#) :: [ListStart\(string listType\)](#)

Description: Start any list like ul, ol, etc.

Definition at line 301 of file pagestructure.cc.

#### 4.17.3.21 void PageStructureMaker::SelectFieldEnd ( )

End Select Field

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: [SelectFieldEnd\(\)](#)

Description: Close select field

Definition at line 453 of file pagestructure.cc.

#### 4.17.3.22 void PageStructureMaker::SelectFieldStart ( string name )

Select Field Start

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: [SelectFieldStart\(string name\)](#)

Description: Create Select Field

Definition at line 440 of file pagestructure.cc.

#### 4.17.3.23 void PageStructureMaker::SelectOptionEnd ( )

Select Option End

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: [SelectOptionEnd\(\)](#)

Description: Close select option

Definition at line 483 of file pagestructure.cc.

#### 4.17.3.24 void PageStructureMaker::SelectOptionStart ( string value, string selected )

Select Option Start

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: [SelectOptionStart\(string value, string selected\)](#)

Description: Options for select

Definition at line 467 of file pagestructure.cc.

#### 4.17.3.25 void PageStructureMaker::SetHTMLVariables ( )

Assign Values to variables

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: [SetHTMLVariables\(\)](#) Description: Set values of common HTML tags in respective

variables.

Definition at line 48 of file pagestructure.cc.

#### 4.17.3.26 void PageStructureMaker::TableEnd ( )

End Table

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: [TableEnd\(\)](#)

Description: Close Table tag

Definition at line 288 of file pagestructure.cc.

#### 4.17.3.27 void PageStructureMaker::TableStart ( string id, string className )

Start Table

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: [TableStart\(string id, string className\)](#)

Description: Start Table with id and className(CSS) attributes

Definition at line 274 of file pagestructure.cc.

#### 4.17.3.28 void PageStructureMaker::Title ( string pageTitle )

Display <TITLE> </TITLE>

Class: [PageStructureMaker](#) Method: [PageStructureMaker](#) :: [Title\(string pageTitle\)](#)

Description: Display Page Tile

Definition at line 125 of file pagestructure.cc.

### 4.17.4 Member Data Documentation

#### 4.17.4.1 string PageStructureMaker::startH1

HTML Tag Variables for td, th, bold, etc

Definition at line 38 of file pagestructure.h.

The documentation for this class was generated from the following files:

- [src/header/pagestructure.h](#)
- [src/pagestructure.cc](#)

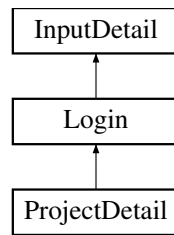
## 4.18 ProjectDetail Class Reference

[ProjectDetail](#) class for creating new project for user.

```
#include <project.h>
```

Inheritance diagram for ProjectDetail:





## Public Member Functions

- [ProjectDetail](#) ()
- void [ProjectDetailPage](#) (string [msg](#)="", string [projectName](#)="Project Name")
- void [AuthorizeUser](#) ()
- void [OldProject](#) ()
  - List Existing projects from database.*
- void [HomePage](#) ()
  - Home Page.*
- [~ProjectDetail](#) ()

## Additional Inherited Members

### 4.18.1 Detailed Description

[ProjectDetail](#) class for creating new project for user.

Include [login.h](#)

Definition at line 31 of file [project.h](#).

### 4.18.2 Constructor & Destructor Documentation

#### 4.18.2.1 [ProjectDetail::ProjectDetail](#) ( )

[ProjectDetail](#) Constructor

include [projectdetail.h](#) -----

Class: [ProjectDetail](#)

Method: [ProjectDetail](#) :: [ProjectDetail](#)()

Description: Constructor of [ProjectDetail](#)

Definition at line 29 of file [project.cc](#).

#### 4.18.2.2 [ProjectDetail::~~ProjectDetail](#) ( )

Destructor

Class: [ProjectDetail](#)

Method: [ProjectDetail](#) :: [~ProjectDetail](#)()

Description: Destuctor

Definition at line 194 of file project.cc.

### 4.18.3 Member Function Documentation

#### 4.18.3.1 void ProjectDetail::AuthorizeUser ( )

Checking user login details valid or not

---

Class: [ProjectDetail](#)

Method: [ProjectDetail](#) :: [AuthorizeUser\(\)](#)

Description: Checking user login details valid or invalid

Matching user details with values in database

< If Email ID valid

< If Password Correct

< If Password Incorrect

< If Email ID invalid

Definition at line 42 of file project.cc.

#### 4.18.3.2 void ProjectDetail::HomePage ( )

Home Page.

Home Page

Definition at line 172 of file project.cc.

#### 4.18.3.3 void ProjectDetail::OldProject ( )

List Existing projects from database.

Old projects

Definition at line 144 of file project.cc.

#### 4.18.3.4 void ProjectDetail::ProjectDetailPage ( string msg = " ", string projectName = "Project Name" )

Display Page for project details

---

Class: [ProjectDetail](#)

Method: [ProjectDetail](#) :: [ProjectDetailPage\(\)](#)

Description: Diplaying page for project details

Definition at line 93 of file project.cc.

The documentation for this class was generated from the following files:

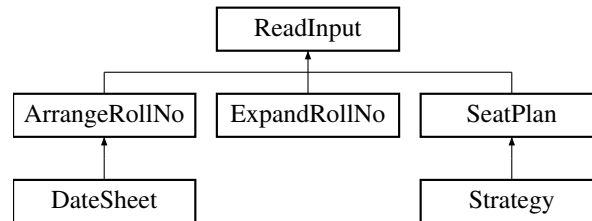
- src/header/project.h
- src/project.cc

## 4.19 ReadInput Class Reference

Class for reading input files ie reading roll no details, class detail, room detail, etc.

```
#include <readinput.h>
```

Inheritance diagram for ReadInput:



### Public Member Functions

- [ReadInput \(\)](#)  
*Constructor.*
- string [FileName](#) (string file, string [projectId](#), int fileType)  
*For creating filename w.r.t to projectId.*
- void [ReadRollNoDetail](#) (string [projectId](#))  
*Reading Roll No detail from file.*
- void [ReadClassDetail](#) (string [projectId](#))  
*Reading class details.*
- void [ReadDateSheet](#) (string [projectId](#))  
*Reading datesheet.*
- void [ReadDateSheetOut](#) (string [projectId](#))  
*Reading datesheet.out file.*
- void [ReadRoomDetail](#) (string [projectId](#))  
*Reading room details.*
- void [ReadStrategy](#) (string [projectId](#))  
*Function for reading strategy.*
- void [ReadExamDetail](#) (string [projectId](#))  
*Reading Exam DEtails.*
- [~ReadInput \(\)](#)  
*Destructor.*

### Protected Attributes

- STRING\_VEC [prefix](#)
- STRING\_VEC [rollNo](#)
- STRING\_VEC [notIncluded](#)
- STRING\_VEC [className](#)
- STRING\_VEC [date](#)
- STRING\_VEC [strategyOption](#)
- STRING\_VEC [examName](#)

- `STRING_VEC examSession`
- `STRING_VEC examTime`
- `STRING_VEC examVenue`
- `STRING_VEC examDate`
- `STRING_2DVEC subjectCode`
- `STRING_2DVEC subjectName`
- `STRING_2DVEC examSubCode`
- `STRING_2DVEC examSubName`
- `STRING_2DVEC centreName`
- `STRING_3DVEC roomNo`
- `int totalClasses`
- `int totalDays`
- `INT_VEC totalSubjects`
- `INT_VEC totalExams`
- `INT_VEC totalCentres`
- `INT_2DVEC totalRooms`
- `INT_3DVEC rows`
- `INT_3DVEC cols`
- `ifstream inFile`
- `ofstream outFile`
- `string fileName`
- `string temp`
- `int i`
- `int j`
- `int k`
- `int l`
- `int m`
- `int n`
- `string projectID`
- `INT_2DVEC dateSheetRNoSize`
- `STRING_3DVEC dateSheetRNo`

#### 4.19.1 Detailed Description

Class for reading input files ie reading roll no details, class detail, room detail, etc.

Include local header file

Include `readinput.h` file

Definition at line 31 of file `readinput.h`.

#### 4.19.2 Constructor & Destructor Documentation

##### 4.19.2.1 `ReadInput::ReadInput ( )`

Constructor.

Constructor

Definition at line 27 of file `readinput.cc`.

##### 4.19.2.2 `ReadInput::~ReadInput ( )`

Destructor.

Desrtuctor

Definition at line 378 of file `readinput.cc`.

### 4.19.3 Member Function Documentation

#### 4.19.3.1 `string ReadInput::FileName ( string file, string projectID, int fileType )`

For creating filename w.r.t to projectID.

Retrurning filename w.r.t. to project ID

##### Parameters

<i>projectID</i>	Unique ID i.e. used to read file of user
<i>file</i>	I/O file Name
<i>fileType</i>	For defining file type i.e I/P or O/P file. If fileType == 1, Input file If fileType == 0(else), Output File

##### Returns

fileName This function will return file name with project ID

Definition at line 48 of file readinput.cc.

#### 4.19.3.2 `void ReadInput::ReadClassDetail ( string fileName )`

Reading class details.

Reading Class details from I/P file

##### Parameters

<i>fileName</i>	I/P filename for class detail
-----------------	-------------------------------

Definition at line 267 of file readinput.cc.

#### 4.19.3.3 `void ReadInput::ReadDateSheet ( string fileName )`

Reading datesheet.

Reading Datesheet

##### Parameters

<i>fileName</i>	Name of I/P file
-----------------	------------------

Definition at line 312 of file readinput.cc.

#### 4.19.3.4 `void ReadInput::ReadDateSheetOut ( string projectID )`

Reading datesheet.out file.

Read date sheet with roll nos

##### Parameters

<i>projectID</i>	ProjectID
------------------	-----------

Definition at line 68 of file readinput.cc.

#### 4.19.3.5 void ReadInput::ReadExamDetail ( string *projectID* )

Reading Exam DEtails.

Reading exam details

Reimplemented in [SeatPlan](#).

Definition at line 232 of file readinput.cc.

#### 4.19.3.6 void ReadInput::ReadRollNoDetail ( string *fileName* )

Reading Roll No detail from file.

Reading Roll No details from I/P file

##### Parameters

<i>projectID</i>	Name of I/P file for roll nos.
------------------	--------------------------------

Definition at line 201 of file readinput.cc.

#### 4.19.3.7 void ReadInput::ReadRoomDetail ( string *projectID* )

Reading room details.

Read Room Details

##### Parameters

<i>projectID</i>	projectId of user's project
------------------	-----------------------------

Definition at line 134 of file readinput.cc.

#### 4.19.3.8 void ReadInput::ReadStrategy ( string *projectID* )

Function for reading strategy.

Reading strategy choice from strategy.in file

##### Parameters

<i>projectID</i>	projectId of user's project
------------------	-----------------------------

Definition at line 355 of file readinput.cc.

### 4.19.4 Member Data Documentation

#### 4.19.4.1 STRING\_2DVEC ReadInput::centreName [protected]

Centre NAmE

Definition at line 50 of file readinput.h.

#### 4.19.4.2 STRING\_VEC ReadInput::className [protected]

Branch/trade/class Name

Definition at line 35 of file readinput.h.

**4.19.4.3 INT\_3DVEC ReadInput::cols** [protected]

columns in room

Definition at line 67 of file readinput.h.

**4.19.4.4 STRING\_VEC ReadInput::date** [protected]

Date for examination

Definition at line 35 of file readinput.h.

**4.19.4.5 STRING\_3DVEC ReadInput::dateSheetRNo** [protected]

Roll Nos w.r.t datesheet

Definition at line 83 of file readinput.h.

**4.19.4.6 INT\_2DVEC ReadInput::dateSheetRNoSize** [protected]

Total roll nos for exam code

Definition at line 81 of file readinput.h.

**4.19.4.7 STRING\_2DVEC ReadInput::examSubCode** [protected]

Examination/subject code

Definition at line 50 of file readinput.h.

**4.19.4.8 STRING\_2DVEC ReadInput::examSubName** [protected]

Exam/ subejct Name

Definition at line 50 of file readinput.h.

**4.19.4.9 string ReadInput::fileName** [protected]

I/O file name

Definition at line 73 of file readinput.h.

**4.19.4.10 ifstream ReadInput::inFile** [protected]

For reading data from file

Definition at line 70 of file readinput.h.

**4.19.4.11 int ReadInput::n** [protected]

for looping

Definition at line 76 of file readinput.h.

#### 4.19.4.12 `STRING_VEC ReadInput::notIncluded` [protected]

Rollno ie not for seating plan

Definition at line 35 of file readinput.h.

#### 4.19.4.13 `ofstream ReadInput::outFile` [protected]

for writing data into file

Definition at line 71 of file readinput.h.

#### 4.19.4.14 `STRING_VEC ReadInput::prefix` [protected]

Prefix of rollno

Definition at line 35 of file readinput.h.

#### 4.19.4.15 `string ReadInput::projectID` [protected]

Project id of user's project

Definition at line 78 of file readinput.h.

#### 4.19.4.16 `STRING_VEC ReadInput::rollNo` [protected]

rollno range

Definition at line 35 of file readinput.h.

#### 4.19.4.17 `STRING_3DVEC ReadInput::roomNo` [protected]

Room No

Definition at line 56 of file readinput.h.

#### 4.19.4.18 `INT_3DVEC ReadInput::rows` [protected]

Rows in room

Definition at line 67 of file readinput.h.

#### 4.19.4.19 `STRING_VEC ReadInput::strategyOption` [protected]

[Strategy](#) option selected by user

Definition at line 35 of file readinput.h.

#### 4.19.4.20 `STRING_2DVEC ReadInput::subjectCode` [protected]

Subject Code

Definition at line 50 of file readinput.h.



**4.19.4.21** `STRING_2DVEC ReadInput::subjectName` `[protected]`

Subject Name

Definition at line 50 of file `readinput.h`.

**4.19.4.22** `string ReadInput::temp` `[protected]`

For temporary use

Definition at line 73 of file `readinput.h`.

**4.19.4.23** `INT_VEC ReadInput::totalCentres` `[protected]`

Total centres

Definition at line 61 of file `readinput.h`.

**4.19.4.24** `int ReadInput::totalClasses` `[protected]`

Total classes/branches/trades

Definition at line 58 of file `readinput.h`.

**4.19.4.25** `int ReadInput::totalDays` `[protected]`

No. of days in datesheet

Definition at line 58 of file `readinput.h`.

**4.19.4.26** `INT_VEC ReadInput::totalExams` `[protected]`

Total exams on 1 day

Definition at line 61 of file `readinput.h`.

**4.19.4.27** `INT_2DVEC ReadInput::totalRooms` `[protected]`

total rooms in centre

Definition at line 65 of file `readinput.h`.

**4.19.4.28** `INT_VEC ReadInput::totalSubjects` `[protected]`

Total subejcts of class

Definition at line 61 of file `readinput.h`.

The documentation for this class was generated from the following files:

- `src/backend/header/readinput.h`
- `src/backend/readinput.cc`

## 4.20 ReadInputField Class Reference

```
#include <readinputfield.h>
```

## Public Member Functions

- [ReadInputField](#) ()
- string [ReadFieldValue](#) (string fieldName)
- string [ReadFieldValue](#) (string fieldName, int fieldNo)

## Protected Attributes

- string **fieldValue**
- Cgicc [formData](#)
- form\_iterator [fi](#)

### 4.20.1 Detailed Description

Include required header files

Class: [ReadInputField](#) Description: For reading values of fields like textbox, select,

etc.

Definition at line 38 of file readinputfield.h.

### 4.20.2 Constructor & Destructor Documentation

#### 4.20.2.1 [ReadInputField::ReadInputField](#) ( ) [[inline](#)]

Constructor

Definition at line 53 of file readinputfield.h.

### 4.20.3 Member Function Documentation

#### 4.20.3.1 string [ReadInputField::ReadFieldValue](#) ( string *fieldName* )

Read Field's Value with one argument fieldName

Include Header file of [ReadInputField](#) class declaration

Definition of member functions of [ReadInputField](#) Class

Class: [ReadInputField](#) Method: [ReadInputField](#) :: [ReadFieldValue](#)(string fieldName)

Description: Read field's value and return it as string

Definition at line 37 of file readinputfield.cc.

#### 4.20.3.2 string [ReadInputField::ReadFieldValue](#) ( string *fieldName*, int *fieldNo* )

Read Field's value with two arguments fieldName and field no.)

Class: [ReadInputField](#) Method: [ReadInputField](#) :: [ReadFieldValue](#)(string fieldName, int fieldNo)

Description: Read field's value and return it as string

Definition at line 60 of file readinputfield.cc.

#### 4.20.4 Member Data Documentation

##### 4.20.4.1 `form_iterator` `ReadInputField::fi` `[protected]`

Form Iterator

Definition at line 46 of file readinputfield.h.

##### 4.20.4.2 `Cgicc` `ReadInputField::formData` `[protected]`

Read Form Data

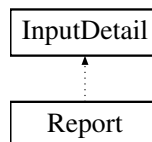
Definition at line 45 of file readinputfield.h.

The documentation for this class was generated from the following files:

- `src/header/readinputfield.h`
- `src/readinputfield.cc`

## 4.21 Report Class Reference

Inheritance diagram for Report:



### Public Member Functions

- [Report](#) ()
- void [ReadValidationDetail](#) ()  
*Reading hidden fields of validation page.*
- void [ReportPage](#) ()  
*Report Page.*
- [~Report](#) ()

### Additional Inherited Members

#### 4.21.1 Detailed Description

Definition at line 21 of file report.h.

## 4.21.2 Constructor & Destructor Documentation

### 4.21.2.1 Report::Report ( )

Constructor

Definition at line 18 of file report.cc.

### 4.21.2.2 Report::~~Report ( )

Destructor

Definition at line 122 of file report.cc.

## 4.21.3 Member Function Documentation

### 4.21.3.1 void Report::ReadValidationDetail ( )

Reading hidden fields of validation page.

Reading validation page's hidden fields

Definition at line 35 of file report.cc.

### 4.21.3.2 void Report::ReportPage ( )

[Report](#) Page.

[Report](#) Page

Definition at line 60 of file report.cc.

The documentation for this class was generated from the following files:

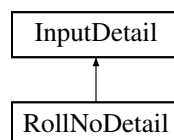
- src/header/report.h
- src/report.cc

## 4.22 RollNoDetail Class Reference

[RollNoDetail](#) class for taking roll no detail from user.

```
#include <rollno.h>
```

Inheritance diagram for RollNoDetail:



### Public Member Functions

- [RollNoDetail](#) ()  
*Constructor.*
- void [ReadClassDetail](#) ()

- Reading class details from previous page and store them into database.*
- void [WriteClassDetail](#) ()  
*Write class details into database.*
- void [SetDefaultValue](#) (int [totalClasses](#))  
*Setting default valued of i/p fields for taking i/p from user.*
- void [RollNoDetailPage](#) (string [msg](#)="")  
*RollNo detail page for taking i/p from user.*
- [~RollNoDetail](#) ()  
*Destructor.*

## Additional Inherited Members

### 4.22.1 Detailed Description

[RollNoDetail](#) class for taking roll no detail from user.

include [class.h](#)

include [rollno.h](#)

Definition at line 32 of file [rollno.h](#).

### 4.22.2 Constructor & Destructor Documentation

#### 4.22.2.1 RollNoDetail::RollNoDetail ( )

Constructor.

Roll No detail Constructor

Definition at line 27 of file [rollno.cc](#).

#### 4.22.2.2 RollNoDetail::~~RollNoDetail ( )

Destructor.

Destructor

Definition at line 358 of file [rollno.cc](#).

### 4.22.3 Member Function Documentation

#### 4.22.3.1 void RollNoDetail::ReadClassDetail ( )

Reading class details from previous page and store them into database.

Read Class Details

Definition at line 49 of file [rollno.cc](#).

#### 4.22.3.2 void RollNoDetail::RollNoDetailPage ( string *msg* = " " )

RollNo detail page for taking i/p from user.

RollNo detail page

## Parameters

<i>msg</i>	For displaying error msg if any
------------	---------------------------------

Definition at line 222 of file rollno.cc.

#### 4.22.3.3 void RollNoDetail::SetDefaultValue ( int *totalClasses* )

Setting default valued of i/p fields for taking i/p from user.

Set default values of field

## Parameters

<i>totalClasses</i>	Total no. of rows/classes
---------------------	---------------------------

Definition at line 141 of file rollno.cc.

#### 4.22.3.4 void RollNoDetail::WriteClassDetail ( )

Write class details into database.

Write class details in database

Definition at line 163 of file rollno.cc.

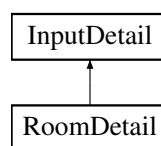
The documentation for this class was generated from the following files:

- [src/header/rollno.h](#)
- [src/rollno.cc](#)

## 4.23 RoomDetail Class Reference

```
#include <room.h>
```

Inheritance diagram for RoomDetail:



### Public Member Functions

- [RoomDetail](#) ()  
*Constructor.*
- void [SetDefaultValue](#) ()
- void [ReadDateSheet](#) ()  
*Reading Date sheet I/P given by user.*
- void [WriteDateSheet](#) ()  
*Writing Date Sheet into database.*
- void [RoomDetailPage](#) ()  
*For Taking I/P from user of room details.*
- [~RoomDetail](#) ()  
*Destructor to de-allocate objects.*

## Protected Attributes

- [DateSheet](#) `dateWiseRollNo`

### 4.23.1 Detailed Description

include inpudetail.h

Include [room.h](#)

Definition at line 26 of file room.h.

### 4.23.2 Constructor & Destructor Documentation

#### 4.23.2.1 RoomDetail::RoomDetail ( )

Constructor.

Constructor

Definition at line 27 of file room.cc.

#### 4.23.2.2 RoomDetail::~RoomDetail ( )

Destructor to de-allocate objects.

Desctructor

Definition at line 421 of file room.cc.

### 4.23.3 Member Function Documentation

#### 4.23.3.1 void RoomDetail::ReadDateSheet ( )

Reading Date sheet I/P given by user.

Reading Datesheet

Definition at line 148 of file room.cc.

#### 4.23.3.2 void RoomDetail::RoomDetailPage ( )

For Taking I/P from user of room details.

Room DEtail page for taking I/P from user

Definition at line 243 of file room.cc.

#### 4.23.3.3 void RoomDetail::SetDefaultValue ( )

Set Default Value

Definition at line 46 of file room.cc.

#### 4.23.3.4 void RoomDetail::WriteDateSheet ( )

Writing Date Sheet into database.

Writing [DateSheet](#)

Definition at line 189 of file room.cc.

The documentation for this class was generated from the following files:

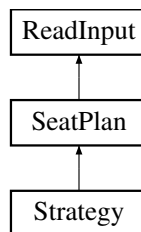
- src/header/[room.h](#)
- src/[room.cc](#)

## 4.24 SeatPlan Class Reference

[SeatPlan](#) Class for generating seating plan.

```
#include <report.h>
```

Inheritance diagram for [SeatPlan](#):



### Public Member Functions

- [RoomReport](#) ()
- void [ReadExamDetail](#) (string [projectId](#))
- void [ReadSeatPlan](#) (string [projectId](#))
- void [WriteHTMLFile](#) (string [projectId](#))
- [~RoomReport](#) ()
- [SeatPlan](#) ()  
*Constructor.*
- void [SetRollNo](#) (int strategy, int i)
- string [RollNo](#) (int s)
- void [SeatingPlan](#) (int strategy, int i)
- void [CreateFile](#) (string [projectId](#))
- void [WriteSeatPlan](#) (string [projectId](#), int i)
- void [WriteHTMLFile](#) (string [projectId](#), int i)  
*Creating HTML file.*
- [~SeatPlan](#) ()  
*Destructor.*

### Protected Attributes

- int **totalSeats**
- int **totalStudents**
- int **totalGroupSeats**
- int **day**
- INT\_VEC **iTemp**
- INT\_VEC **indexValue**
- INT\_VEC **groupStudentSize**
- INT\_VEC **seatSize**
- INT\_VEC **size**



- INT\_3DVEC **roomSize**
- STRING\_VEC **subSubCode**
- STRING\_2DVEC **seatRollNo**
- STRING\_2DVEC **subRollNo**
- STRING\_4DVEC **seat**
- int **centre**
- int **room**
- int **row**
- int **col**
- int **s**
- int **start**
- int **end**
- int **index**

#### 4.24.1 Detailed Description

[SeatPlan](#) CClass for generating seating plan.

Include local header file

include [readinput.h](#)

include [seatplan.h](#) file

Definition at line 25 of file report.h.

#### 4.24.2 Constructor & Destructor Documentation

##### 4.24.2.1 `SeatPlan::~~RoomReport ( )`

Destructor

##### 4.24.2.2 `SeatPlan::SeatPlan ( )`

Constructor.

Constructor

Definition at line 28 of file seatplan.cc.

##### 4.24.2.3 `SeatPlan::~~SeatPlan ( )`

DEstructor.

Destructor

Definition at line 313 of file seatplan.cc.

#### 4.24.3 Member Function Documentation

##### 4.24.3.1 `void SeatPlan::ReadExamDetail ( string projectID )`

Read Exam Detail

Reimplemented from [ReadInput](#).

#### 4.24.3.2 void SeatPlan::ReadSeatPlan ( string *projectID* )

Read Seat Plan

#### 4.24.3.3 SeatPlan::RoomReport ( )

Constructor

#### 4.24.3.4 void SeatPlan::WriteHTMLFile ( string *projectID* )

WriteHTMLFile

#### 4.24.3.5 void SeatPlan::WriteHTMLFile ( string *projectID*, int *i* )

Creating HTML file.

##### Parameters

<i>projectID</i>	Project Id of eting plan project
<i>i</i>	For creating file accord to datesheet

Definition at line 225 of file seatplan.cc.

The documentation for this class was generated from the following files:

- src/backend/header/report.h
- src/backend/header/[seatplan.h](#)
- src/backend/[seatplan.cc](#)

## 4.25 SendMail Class Reference

Class for sending mail to user for registration.

```
#include <sendmail.h>
```

### Public Member Functions

- [SendMail](#) ()  
*Constructor.*
- void [SetMailData](#) ()  
*Settinf variable that are used to send mail.*
- void [SetHTMLMessage](#) (string regKey)  
*Setting body of mail(message ie send to user)*
- void [RegistrationMail](#) (string setRecipient, string regKey)  
*Sending registration mail to user.*
- [~SendMail](#) ()  
*Destructor.*

### Protected Attributes

- string **setSender**
- string **setSubject**

- string **setMessage**
- string **setServer**
- string **htmlMessage**
- string **url**

#### 4.25.1 Detailed Description

Class for sending mail to user for registration.

Include [sendmail.h](#)

Definition at line 39 of file sendmail.h.

#### 4.25.2 Constructor & Destructor Documentation

##### 4.25.2.1 SendMail::SendMail ( )

Constructor.

Constructor

Definition at line 28 of file sendmail.cc.

##### 4.25.2.2 SendMail::~~SendMail ( )

Destructor.

Desrtuctor

Definition at line 100 of file sendmail.cc.

#### 4.25.3 Member Function Documentation

##### 4.25.3.1 void SendMail::RegistrationMail ( string *setRecipient*, string *regKey* )

Sending registration mail to user.

Function for sending mail to new user(registration link)

Parameters

--	--

Definition at line 75 of file sendmail.cc.

##### 4.25.3.2 void SendMail::SetHTMLMessage ( string *regKey* )

Setting body of mail(message ie send to user)

Setting message of mail

Parameters

<i>regKey</i>	Registration Key
---------------	------------------

Definition at line 54 of file sendmail.cc.

#### 4.25.3.3 void SendMail::SetMailData ( )

Settinf variable that are used to send mail.

Setting variables

Definition at line 39 of file sendmail.cc.

The documentation for this class was generated from the following files:

- src/header/sendmail.h
- src/sendmail.cc

## 4.26 jwsmtplib::SOCKADDR\_IN Struct Reference

### Public Member Functions

- **SOCKADDR\_IN** (const std::string &address, unsigned short port, short family=AF\_INET)
- **SOCKADDR\_IN** (const [SOCKADDR\\_IN](#) &addr)
- [SOCKADDR\\_IN](#) **operator=** (const [SOCKADDR\\_IN](#) &addr)
- **operator bool** () const
- **operator const sockaddr\_in** () const
- **operator const sockaddr** () const
- size\_t **get\_size** () const
- char \* **get\_sin\_addr** ()
- void **set\_port** (unsigned short newport)
- void **set\_ip** (const std::string &newip)
- void **zeroaddress** ()

### Public Attributes

- sockaddr\_in **ADDR**

#### 4.26.1 Detailed Description

Definition at line 84 of file compat.h.

The documentation for this struct was generated from the following file:

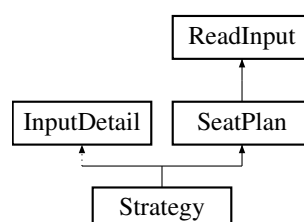
- src/header/jwsmtplib/compat.h

## 4.27 Strategy Class Reference

[Strategy](#) class to check selected strategy is valid or not for seating plan.

```
#include <strategy.h>
```

Inheritance diagram for Strategy:



## Public Member Functions

- [Strategy](#) ()  
*Constructor.*
- void [ChooseStrategy](#) ()  
*Select [Strategy](#).*
- void **TotalSeats** (int strategy, int i)
- void **TotalStudents** (int i)
- void **CheckValidation** (int strategy, int i)
- void **TotalGroupStudents** (int strategy, int i)
- void **GroupCondition** (int strategy, int i)
- void [Main](#) (string pID)  
*Main Function to call all other functions.*
- [~Strategy](#) ()  
*Destructor.*
- [Strategy](#) ()
- void [ReadExamDetail](#) ()  
*Reading exam detail from last page filled by user.*
- void [WriteExamDetail](#) ()  
*Writing exam detail into DB.*
- void [SetDefaultValue](#) ()  
*Setting default or retrieving old values from database.*
- void [StrategyPage](#) ()  
*[Strategy](#) Page.*
- [~Strategy](#) ()

## Protected Attributes

- `STRING_VEC` [strategyName](#)
- `int` [totalStrategy](#)

## Additional Inherited Members

### 4.27.1 Detailed Description

[Strategy](#) class to check selected strategy is valid or not for seating plan.

include [inputdetail.h](#)

include [seatplan.h](#) file

include [strategy.h](#)

Definition at line 29 of file [strategy.h](#).

### 4.27.2 Constructor & Destructor Documentation

#### 4.27.2.1 [Strategy::Strategy](#) ( )

Constructor.

Constructor

Definition at line 28 of file [strategy.cc](#).

#### 4.27.2.2 Strategy::~~Strategy ( )

Destructor.

Destructor

Definition at line 239 of file strategy.cc.

#### 4.27.2.3 Strategy::Strategy ( )

Constructor

#### 4.27.2.4 Strategy::~~Strategy ( )

Destructor

### 4.27.3 Member Function Documentation

#### 4.27.3.1 void Strategy::Main ( string *pID* )

Main Function to call all other functions.

Parameters

<i>pID</i>	project ID of user's project
------------	------------------------------

Definition at line 221 of file strategy.cc.

#### 4.27.3.2 void Strategy::ReadExamDetail ( )

Reading exam detail from last page filled by user.

Reading exam detail

Definition at line 54 of file strategy.cc.

#### 4.27.3.3 void Strategy::SetDefaultValue ( )

Setting default or retrieving old values from database.

Setting default value

Definition at line 96 of file strategy.cc.

#### 4.27.3.4 void Strategy::StrategyPage ( )

[Strategy](#) Page.

strategy detail page

Definition at line 180 of file strategy.cc.

#### 4.27.3.5 void Strategy::WriteExamDetail ( )

Writing exam detail into DB.

Write Exam detail into database

Definition at line 120 of file strategy.cc.

### 4.27.4 Member Data Documentation

#### 4.27.4.1 `STRING_VEC Strategy::strategyName` `[protected]`

Name of [Strategy](#)

Definition at line 32 of file `strategy.h`.

#### 4.27.4.2 `int Strategy::totalStrategy` `[protected]`

Total Strategies

Definition at line 33 of file `strategy.h`.

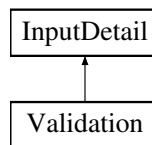
The documentation for this class was generated from the following files:

- `src/backend/header/strategy.h`
- `src/header/strategy.h`
- `src/backend/strategy.cc`
- `src/strategy.cc`

## 4.28 Validation Class Reference

```
#include <validation.h>
```

Inheritance diagram for Validation:



### Public Member Functions

- [Validation](#) ()  
*Constructor.*
- void [ReadStrategyDetail](#) ()  
*Reading strategy value from last page.*
- void [WriteStrategyDetail](#) ()  
*Write strategy detail into DB.*
- void [ReadValidation](#) ()  
*Reading validation file for checking strategy is valid or not.*
- void [ValidationPage](#) ()  
*Validation.html page.*
- [~Validation](#) ()  
*Destructor.*

### Protected Attributes

- [Strategy](#) `validStrategy`

### 4.28.1 Detailed Description

include inpudetail.h

Definition at line 25 of file validation.h.

### 4.28.2 Constructor & Destructor Documentation

#### 4.28.2.1 Validation::Validation ( )

Constructor.

Constructor

Definition at line 24 of file validation.cc.

#### 4.28.2.2 Validation::~~Validation ( )

Destructor.

desrtuctor

Definition at line 229 of file validation.cc.

### 4.28.3 Member Function Documentation

#### 4.28.3.1 void Validation::ReadStrategyDetail ( )

Reading strategy value from last page.

Read [Strategy](#) Detail

Definition at line 46 of file validation.cc.

#### 4.28.3.2 void Validation::ReadValidation ( )

Reading validation file for checking strategy is valid or not.

Read Validation.out file

Definition at line 131 of file validation.cc.

#### 4.28.3.3 void Validation::ValidationPage ( )

Validation.html page.

[Validation](#) Page

Definition at line 158 of file validation.cc.

#### 4.28.3.4 void Validation::WriteStrategyDetail ( )

Write strategy detail into DB.

Write [Strategy](#) Detail

Definition at line 80 of file validation.cc.

The documentation for this class was generated from the following files:

- src/header/[validation.h](#)



- [src/validation.cc](#)



## Chapter 5

# File Documentation

### 5.1 src/adduser-main.cpp File Reference

Adding new user and setting its password.

```
#include "header/login.h"
```

#### Functions

- int [main](#) (void)  
*Main Method.*

#### 5.1.1 Detailed Description

Adding new user and setting its password.

#### Version

0.6

#### Date

Sunday 07 April 2013 01:15:32 IST  
Compiler g++

#### Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

#### Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [adduser-main.cpp](#).

#### 5.1.2 Function Documentation

### 5.1.2.1 main ( void )

Main Method.

Main function to call functions of [ProjectDetail](#) Class.

Main method to call NewUser function of [Login](#) class.

Main function for calling functions of [Login](#) class to create login.html page.

Main method.

Main method for creating.html page.

Include [login.h](#)

include header file

Include [login.h](#)

#### Parameters

<i>void</i>	no arguments
-------------	--------------

Include [login.h](#) file

Include projectdetail.h

Definition at line 26 of file adduser-main.cpp.

## 5.2 src/backend/arrangerollno.cc File Reference

Function definition of [ArrangeRollNo](#) class.

```
#include "header/arrangerollno.h"
```

### 5.2.1 Detailed Description

Function definition of [ArrangeRollNo](#) class.

#### Version

0.6

#### Date

Sunday 31 March 2013 06:50:00 IST  
Compiler g++

#### Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

#### Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [arrangerollno.cc](#).

## 5.3 src/backend/expandrollno.cc File Reference

Function definition of [ExpandRollNo](#) class definition.

```
#include "header/expandrollno.h"
```

### 5.3.1 Detailed Description

Function definition of [ExpandRollNo](#) class definition.

#### Version

0.6

#### Date

Sunday 31 March 2013 06:49:23 IST

Compiler g++

#### Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

#### Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [expandrollno.cc](#).

## 5.4 src/backend/header/arrangerollno.h File Reference

[ArrangeRollNo](#) class for arranging roll no, sorting, and removing roll nos that are not required for seating plan.

```
#include "expandrollno.h"
```

### Classes

- class [ArrangeRollNo](#)

*Arrange roll no class for sorting, removing redundant roll nos and also excluding roll nos that are not for seating plan.*

### 5.4.1 Detailed Description

[ArrangeRollNo](#) class for arranging roll no, sorting, and removing roll nos that are not required for seating plan.

#### Version

0.6

#### Date

Sunday 31 March 2013 06:33:39 IST

Compiler g++

**Author**

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

**Copyright**

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [arrangerollno.h](#).

## 5.5 src/backend/header/expandrollno.h File Reference

[ExpandRollNo](#) class declaration for expanding roll nos.

```
#include "readinput.h"
```

**Classes**

- class [ExpandRollNo](#)

### 5.5.1 Detailed Description

[ExpandRollNo](#) class declaration for expanding roll nos.

**Version**

0.6

**Date**

Sunday 31 March 2013 05:45:55 IST  
Compiler g++

**Author**

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

**Copyright**

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [expandrollno.h](#).

## 5.6 src/backend/header/readinput.h File Reference

Declaration of [ReadInput](#) class for reading input details from file.

```
#include "header.h"
```

**Classes**

- class [ReadInput](#)

*Class for reading input files ie reading roll no details, class detail, room detail, etc.*

### 5.6.1 Detailed Description

Declaration of [ReadInput](#) class for reading input details from file.

#### Version

0.6

#### Date

Sunday 31 March 2013 02:33:49 IST  
Compiler g++

#### Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

#### Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [readinput.h](#).

## 5.7 src/backend/header/seatplan.h File Reference

Declaration of [SeatPlan](#) class for generating seating plan.

```
#include "readinput.h"
```

### Classes

- class [SeatPlan](#)  
*[SeatPlan](#) CClass for generating seating plan.*

### 5.7.1 Detailed Description

Declaration of [SeatPlan](#) class for generating seating plan.

#### Version

0.6

#### Date

Tuesday 02 April 2013 02:05:39 IST  
Compiler g++

#### Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

#### Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [seatplan.h](#).

## 5.8 src/backend/main.cpp File Reference

Main Method.

```
#include "header/datesheet.h"
#include "header/strategy.h"
```

### Functions

- int **main** (void)

### 5.8.1 Detailed Description

Main Method.

Version

0.6

Date

Sunday 31 March 2013 08:13:40 IST  
Compiler g++

Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [main.cpp](#).

## 5.9 src/backend/readinput.cc File Reference

Function Definition of [ReadInput](#) Class.

```
#include "header/readinput.h"
```

### 5.9.1 Detailed Description

Function Definition of [ReadInput](#) Class.

Version

0.6

Date

Sunday 31 March 2013 06:48:51 IST  
Compiler g++



**Author**

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

**Copyright**

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [readinput.cc](#).

## 5.10 src/backend/seatplan.cc File Reference

Function definition of [SeatPlan](#) Class.

```
#include "header/seatplan.h"
```

### 5.10.1 Detailed Description

Function definition of [SeatPlan](#) Class.

**Version**

0.6

**Date**

Tuesday 02 April 2013 02:44:21 IST  
Compiler g++

**Author**

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

**Copyright**

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [seatplan.cc](#).

## 5.11 src/class-main.cpp File Reference

main method

```
#include "header/class.h"
```

**Functions**

- int **main** (void)

### 5.11.1 Detailed Description

main method

Version

0.6

Date

Sunday 07 April 2013 09:00:37 IST  
Compiler g++

Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [class-main.cpp](#).

## 5.12 src/class.cc File Reference

[ClassDetail](#) function definition.

```
#include "header/class.h"
```

### 5.12.1 Detailed Description

[ClassDetail](#) function definition.

Version

0.6

Date

Sunday 07 April 2013 07:49:18 IST  
Compiler g++

Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [class.cc](#).

## 5.13 src/confirm-main.cpp File Reference

Main method to call ConfirmPage of [Login](#) class and creating confirm.html page.

```
#include "header/login.h"
```

### Functions

- int **main** (void)

#### 5.13.1 Detailed Description

Main method to call ConfirmPage of [Login](#) class and creating confirm.html page.

##### Version

0.6

##### Date

Sunday 07 April 2013 07:40:31 IST  
Compiler g++

##### Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

##### Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [confirm-main.cpp](#).

## 5.14 src/datesheet-main.cpp File Reference

main method

```
#include "header/datesheet.h"
```

### Functions

- int **main** (void)

#### 5.14.1 Detailed Description

main method

##### Version

0.6

**Date**

Sunday 07 April 2013 09:00:37 IST  
Compiler g++

**Author**

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

**Copyright**

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [datesheet-main.cpp](#).

## 5.15 src/exam-main.cpp File Reference

main method

```
#include "header/exam.h"
```

**Functions**

- int **main** (void)

### 5.15.1 Detailed Description

main method

**Version**

0.6

**Date**

Sunday 07 April 2013 09:00:37 IST  
Compiler g++

**Author**

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

**Copyright**

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [exam-main.cpp](#).

## 5.16 src/exam.cc File Reference

[ExamDetail](#) Class func. definition.

```
#include "header/exam.h"
```

### 5.16.1 Detailed Description

[ExamDetail](#) Class func. definition.

#### Version

0.6

#### Date

Sunday 07 April 2013 08:04:21 IST  
Compiler g++

#### Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

#### Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [exam.cc](#).

## 5.17 src/header/class.h File Reference

[ClassDetail](#) Class declaration.

```
#include "project.h"
```

### Classes

- class [ClassDetail](#)

*For getting class details nd reading projectdetail.*

### 5.17.1 Detailed Description

[ClassDetail](#) Class declaration.

#### Version

0.6

#### Date

Sunday 07 April 2013 07:45:59 IST  
Compiler g++

#### Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

#### Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [class.h](#).

## 5.18 src/header/exam.h File Reference

[ExamDetail](#) class for exam detail.

```
#include "inputdetail.h"
```

### Classes

- class [ExamDetail](#)  
*[ExamDetail](#) class for getting exam from user.*

### 5.18.1 Detailed Description

[ExamDetail](#) class for exam detail.

#### Version

0.6

#### Date

Sunday 07 April 2013 08:01:53 IST  
Compiler g++

#### Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

#### Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [exam.h](#).

## 5.19 src/header/rollno.h File Reference

RollNo class declaration for taking roll no detail.

```
#include "inputdetail.h"
```

### Classes

- class [RollNoDetail](#)  
*[RollNoDetail](#) class for taking roll no detail from user.*

### 5.19.1 Detailed Description

RollNo class declaration for taking roll no detail.

**Version**

0.6

**Date**

Sunday 07 April 2013 07:51:45 IST  
Compiler g++

**Author**

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

**Copyright**

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [rollno.h](#).

## 5.20 src/header/room.h File Reference

[RoomDetail](#) class.

```
#include "inputdetail.h"  
#include "../backend/header/datesheet.h"
```

**Classes**

- class [RoomDetail](#)

### 5.20.1 Detailed Description

[RoomDetail](#) class.

**Version**

0.6

**Date**

Sunday 07 April 2013 08:14:54 IST  
Compiler g++

**Author**

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

**Copyright**

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [room.h](#).

## 5.21 src/header/sendmail-detail.h File Reference

SMTP swttings detail.

```
#include "constant.h"
```

### Variables

- const char **SENDER\_EMAILID** [FILE\_SIZE] = "baithnekaplan@gmail.com"
- const char **SERVER\_MAIL** [FILE\_SIZE] = "localhoat"

### 5.21.1 Detailed Description

SMTP swttings detail.

#### Version

0.6

#### Date

Saturday 06 April 2013 10:46:54 IST  
Compiler g++

#### Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

#### Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [sendmail-detail.h](#).

## 5.22 src/header/sendmail.h File Reference

Declaration of [SendMail](#) Class usinf jwSMTP library.

```
#include <iostream>
#include "jwsmtp/jwsmtp.h"
#include "sendmail-detail.h"
```

### Classes

- class [SendMail](#)

*Class for sending mail to user for registration.*



### 5.22.1 Detailed Description

Declaration of [SendMail](#) Class usinf jwSMTP library.

#### Version

0.6

#### Date

Saturday 06 April 2013 07:46:27 IST  
Compiler g++

#### Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

#### Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [sendmail.h](#).

## 5.23 src/header/validation.h File Reference

[Validation](#) class.

```
#include "inputdetail.h"  
#include "../backend/header/strategy.h"
```

### Classes

- class [Validation](#)

### 5.23.1 Detailed Description

[Validation](#) class.

#### Version

0.6

#### Date

Sunday 07 April 2013 08:24:00 IST  
Compiler g++

#### Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

#### Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [validation.h](#).

## 5.24 src/javascript.cc File Reference

function definition of javascript class

```
#include "header/javascript.h"
```

### 5.24.1 Detailed Description

function definition of javascript class

#### Version

0.6

#### Date

Thursday 04 April 2013 09:15:01 IST  
Compiler g++

#### Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

#### Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [javascript.cc](#).

## 5.25 src/login-main.cpp File Reference

Main Function for create login.html page.

```
#include "header/login.h"
```

### Functions

- int **main** (void)

### 5.25.1 Detailed Description

Main Function for create login.html page.

#### Version

0.6

#### Date

Friday 05 April 2013 07:30:10 IST  
Compiler g++

**Author**

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

**Copyright**

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [login-main.cpp](#).

## 5.26 src/logout-main.cpp File Reference

Log out user account.

```
#include "header/login.h"
```

**Functions**

- int **main** (void)

### 5.26.1 Detailed Description

Log out user account.

**Version**

0.6

**Date**

Sunday 07 April 2013 04:29:47 IST  
Compiler g++

**Author**

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

**Copyright**

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [logout-main.cpp](#).

## 5.27 src/register-main.cpp File Reference

Main method for register.html.

```
#include "header/login.h"
```

**Functions**

- int **main** (void)

### 5.27.1 Detailed Description

Main method for register.html.

#### Version

0.6

#### Date

Friday 05 April 2013 10:56:56 IST  
Compiler g++

#### Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

#### Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [register-main.cpp](#).

### 5.27.2 Function Documentation

#### 5.27.2.1 int main ( void )

Include [login.h](#) Main method for creating register.html page using functions of [Login](#) class

Definition at line 26 of file register-main.cpp.

## 5.28 src/report-main.cpp File Reference

main method

```
#include "header/report.h"
```

### Functions

- int **main** (void)

### 5.28.1 Detailed Description

main method

#### Version

0.6

#### Date

Sunday 07 April 2013 09:00:37 IST  
Compiler g++

**Author**

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

**Copyright**

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [report-main.cpp](#).

## 5.29 src/rollno-main.cpp File Reference

main method

```
#include "header/rollno.h"
```

**Functions**

- int **main** (void)

### 5.29.1 Detailed Description

main method

**Version**

0.6

**Date**

Sunday 07 April 2013 09:00:37 IST  
Compiler g++

**Author**

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

**Copyright**

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [rollno-main.cpp](#).

## 5.30 src/rollno.cc File Reference

RollNo detail Class definition.

```
#include "header/rollno.h"
```

### 5.30.1 Detailed Description

RollNo detail Class definition.

Version

0.6

Date

Sunday 07 April 2013 07:53:57 IST  
Compiler g++

Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [rollno.cc](#).

## 5.31 src/room-main.cpp File Reference

main method

```
#include "header/room.h"
```

### Functions

- int **main** (void)

### 5.31.1 Detailed Description

main method

Version

0.6

Date

Sunday 07 April 2013 09:00:37 IST  
Compiler g++

Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [room-main.cpp](#).

## 5.32 src/room.cc File Reference

[RoomDetail](#) class definition.

```
#include "header/room.h"
```

### 5.32.1 Detailed Description

[RoomDetail](#) class definition.

Version

0.6

Date

Sunday 07 April 2013 08:17:36 IST  
Compiler g++

Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [room.cc](#).

## 5.33 src/sendmail.cc File Reference

Function Definitons of [SendMail](#) Class.

```
#include "header/sendmail.h"
```

### 5.33.1 Detailed Description

Function Definitons of [SendMail](#) Class.

Version

0.6

Date

Saturday 06 April 2013 07:58:56 IST  
Compiler g++

Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [sendmail.cc](#).

## 5.34 src/strategy-main.cpp File Reference

main method

```
#include "header/strategy.h"
```

### Functions

- int **main** (void)

### 5.34.1 Detailed Description

main method

Version

0.6

Date

Sunday 07 April 2013 09:00:37 IST  
Compiler g++

Author

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [strategy-main.cpp](#).

## 5.35 src/validation-main.cpp File Reference

main method

```
#include "header/validation.h"
```

### Functions

- int **main** (void)

### 5.35.1 Detailed Description

main method

Version

0.6



**Date**

Sunday 07 April 2013 09:00:37 IST  
Compiler g++

**Author**

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

**Copyright**

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [validation-main.cpp](#).

## 5.36 src/validation.cc File Reference

validation func. definition

```
#include "header/validation.h"
```

### 5.36.1 Detailed Description

validation func. definition

**Version**

0.6

**Date**

Sunday 07 April 2013 08:25:24 IST  
Compiler g++

**Author**

Mandeep Kaur, [meghasimak@gmail.com](mailto:meghasimak@gmail.com) License GNU General Public License

**Copyright**

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [validation.cc](#).

# Index

- ~ArrangeRollNo
  - ArrangeRollNo, [10](#)
- ~ClassDetail
  - ClassDetail, [13](#)
- ~Database
  - Database, [15](#)
- ~DateSheet
  - DateSheet, [20](#)
- ~ExamDetail
  - ExamDetail, [22](#)
- ~ExpandRollNo
  - ExpandRollNo, [24](#)
- ~Login
  - Login, [42](#)
- ~ProjectDetail
  - ProjectDetail, [57](#)
- ~ReadInput
  - ReadInput, [60](#)
- ~Report
  - Report, [68](#)
- ~RollNoDetail
  - RollNoDetail, [69](#)
- ~RoomDetail
  - RoomDetail, [71](#)
- ~RoomReport
  - SeatPlan, [73](#)
- ~SeatPlan
  - SeatPlan, [73](#)
- ~SendMail
  - SendMail, [75](#)
- ~Strategy
  - Strategy, [77](#), [78](#)
- ~Validation
  - Validation, [80](#)
- AddPrefixWithRollNo
  - ArrangeRollNo, [10](#)
- AddRollNoWithDateSheet
  - DateSheet, [20](#)
- adduser-main.cpp
  - main, [83](#)
- Anchor
  - PageStructureMaker, [51](#)
- ArrangeRollNo, [9](#)
  - ~ArrangeRollNo, [10](#)
  - AddPrefixWithRollNo, [10](#)
  - ArrangeRollNo, [10](#)
  - ArrangeRollNo, [10](#)
  - expandRNo, [11](#)
  - Main, [10](#)
  - notIncludedRNo, [11](#)
  - prefix, [12](#)
  - prefixRollNo, [12](#)
  - RemoveNonEligibleRollNo, [11](#)
  - RemoveRedundancy, [11](#)
  - rollNo, [12](#)
  - SortRollNo, [11](#)
  - WriteArrangedRollNo, [11](#)
- AuthorizeUser
  - ProjectDetail, [58](#)
- BodyEnd
  - PageStructureMaker, [51](#)
- BodyStart
  - PageStructureMaker, [51](#)
- Button
  - PageStructureMaker, [51](#)
- CSS
  - PageStructureMaker, [51](#)
- centreName
  - InputDetail, [29](#)
  - InputFieldName, [36](#)
  - ReadInput, [62](#)
- ClassDetail, [12](#)
  - ~ClassDetail, [13](#)
  - ClassDetail, [13](#)
  - ClassDetailPage, [13](#)
  - ClassDetail, [13](#)
  - NewProject, [13](#)
  - OldProject, [14](#)
  - ProjectType, [14](#)
  - SetDefaultValue, [14](#)
- ClassDetailPage
  - ClassDetail, [13](#)
- classID
  - InputDetail, [30](#)
  - InputFieldName, [36](#)
- className
  - InputDetail, [30](#)
  - InputFieldName, [36](#)
  - ReadInput, [62](#)
- cols
  - ReadInput, [62](#)
- columns
  - InputDetail, [30](#)
  - InputFieldName, [36](#)
- ConfirmPage
  - Login, [42](#)
- connect

- Database, 18
- ContentType
  - PageLayout, 47
- currentTime
  - Login, 43
- DataSheet, 19
- Database, 14
  - ~Database, 15
  - connect, 18
  - Database, 15
  - DeleteQuery, 16
  - InsertClassDetail, 16
  - InsertDateSheet, 16
  - InsertExamDetail, 16
  - InsertProjectDetail, 16
  - InsertQuery, 16
  - InsertRegistrationDetail, 16
  - InsertRollNoDetail, 17
  - InsertRoomDetail, 17
  - InsertSessionDetail, 17
  - InsertStrategyDetail, 17
  - InsertUserDetail, 17
  - numrows, 18
  - Query, 17
  - SelectColumn, 18
  - SelectProjectID, 18
  - SelectQuery, 18
  - SelectSum, 18
- database
  - InputDetail, 30
- Databaso, 19
- date
  - InputDetail, 30
  - InputFieldName, 37
  - ReadInput, 63
- DateSheet, 19
  - ~DateSheet, 20
  - AddRollNoWithDateSheet, 20
  - DateSheet, 20
  - DateSheetPage, 21
  - DateSheet, 20
  - Main, 21
  - ReadRollNoDetail, 21
  - SetDefaultValue, 21
  - WriteDateSheet, 21
  - WriteRollNoDetail, 21
- DateSheetPage
  - DateSheet, 21
- dateSheetRNo
  - ReadInput, 63
- dateSheetRNoSize
  - ReadInput, 63
- DeleteQuery
  - Database, 16
- DivEnd
  - PageStructureMaker, 52
- DivStart
  - PageStructureMaker, 52
- emailID
  - InputDetail, 30
  - InputFieldName, 37
  - Login, 44
- endRollNo
  - InputDetail, 30
  - InputFieldName, 37
- ErrorMessage
  - InputDetail, 28
- examCode
  - InputDetail, 30
  - InputFieldName, 37
- examDate
  - InputFieldName, 37
- ExamDetail, 22
  - ~ExamDetail, 22
  - ExamDetail, 22
  - ExamDetailPage, 23
  - ExamDetail, 22
  - ReadRoomDetail, 23
  - SetDefaultValue, 23
  - WriteRoomDetail, 23
- ExamDetailPage
  - ExamDetail, 23
- examName
  - InputDetail, 30
  - InputFieldName, 37
- examSession
  - InputDetail, 30
  - InputFieldName, 37
- examSubCode
  - ReadInput, 63
- examSubName
  - ReadInput, 63
- examTime
  - InputDetail, 31
  - InputFieldName, 37
- examVenue
  - InputDetail, 31
  - InputFieldName, 37
- ExapandRollNo, 23
- expandRNo
  - ArrangeRollNo, 11
- ExpandRollNo, 24
  - ~ExpandRollNo, 24
  - ExpandRollNo, 24
  - ExpandRollNoList, 25
  - ExpandRollNos, 25
  - ExpandRollNo, 24
  - WriteExpandRollNo, 25
- ExpandRollNoList
  - ExpandRollNo, 25
- ExpandRollNos
  - ExpandRollNo, 25
- fi
  - ReadInputField, 67
- fieldName
  - InputDetail, 31

- FileName
  - InputDetail, 28
  - ReadInput, 61
- fileName
  - InputDetail, 31
  - ReadInput, 63
- Footer
  - InputDetail, 28
  - PageLayout, 47
- formData
  - ReadInputField, 67
- FormEnd
  - PageStructureMaker, 52
- FormStart
  - PageStructureMaker, 52
- HTMLEnd
  - PageStructureMaker, 53
- HTMLStart
  - PageStructureMaker, 53
- Head
  - PageLayout, 48
- HeadEnd
  - PageStructureMaker, 52
- HeadStart
  - PageStructureMaker, 52
- Header
  - InputDetail, 28
  - PageLayout, 48
- HomePage
  - ProjectDetail, 58
- inFile
  - InputDetail, 31
  - ReadInput, 63
- InputDetail, 25
  - centreName, 29
  - classID, 30
  - className, 30
  - columns, 30
  - database, 30
  - date, 30
  - emailID, 30
  - endRollNo, 30
  - ErrorMessage, 28
  - examCode, 30
  - examName, 30
  - examSession, 30
  - examTime, 31
  - examVenue, 31
  - fieldName, 31
  - FileName, 28
  - fileName, 31
  - Footer, 28
  - Header, 28
  - inFile, 31
  - InputDetail, 28
  - InputDetail, 28
  - IntToString, 29
  - I, 31
  - msg, 31
  - notIncluded, 31
  - oldProject, 31
  - outFile, 32
  - page, 32
  - prefix, 32
  - projectID, 32
  - projectName, 32
  - projectType, 32
  - readField, 32
  - roomNo, 32
  - rows, 32
  - sameDetail, 33
  - sessionID, 33
  - SplitString, 29
  - splitString, 33
  - ss, 33
  - startRollNo, 33
  - strategyChoice, 33
  - StringToInt, 29
  - subjectCode, 33
  - subjectName, 33
  - tableHeading, 33
  - temp1, 34
  - Time, 29
  - totalCentres, 34
  - totalClasses, 34
  - totalCols, 34
  - totalDays, 34
  - totalRows, 34
  - vecTemp, 34
  - where, 34
- InputField
  - PageStructureMaker, 53
- InputFieldName, 35
  - centreName, 36
  - classID, 36
  - className, 36
  - columns, 36
  - date, 37
  - emailID, 37
  - endRollNo, 37
  - examCode, 37
  - examDate, 37
  - examName, 37
  - examSession, 37
  - examTime, 37
  - examVenue, 37
  - InputFieldName, 36
  - InputFieldName, 36
  - key, 38
  - notIncluded, 38
  - password, 38
  - prefix, 38
  - projectID, 38
  - projectName, 38
  - projectType, 38

- retryPassword, [38](#)
  - roomNo, [38](#)
  - rows, [39](#)
  - sameDetail, [39](#)
  - SetFieldNames, [36](#)
  - startRollNo, [39](#)
  - strategyChoice, [39](#)
  - subjectCode, [39](#)
  - subjectName, [39](#)
  - totalCentres, [39](#)
  - totalClasses, [39](#)
  - totalDays, [39](#)
  - totalRooms, [40](#)
  - totalSubjects, [40](#)
- InsertClassDetail
  - Database, [16](#)
- InsertDateSheet
  - Database, [16](#)
- InsertExamDetail
  - Database, [16](#)
- InsertProjectDetail
  - Database, [16](#)
- InsertQuery
  - Database, [16](#)
- InsertRegistrationDetail
  - Database, [16](#)
- InsertRollNoDetail
  - Database, [17](#)
- InsertRoomDetail
  - Database, [17](#)
- InsertSessionDetail
  - Database, [17](#)
- InsertStrategyDetail
  - Database, [17](#)
- InsertUserDetail
  - Database, [17](#)
- IntToString
  - InputDetail, [29](#)
- JavaScript, [40](#)
  - JavaScript, [40](#)
  - JavaScript, [40](#)
  - OnChangeEvent, [41](#)
  - OnFocusEvent, [41](#)
- Javascript
  - PageStructureMaker, [53](#)
- jwsmtplib::SOCKADDR\_IN, [76](#)
- jwsmtplib::mailer, [44](#)
- key
  - InputFieldName, [38](#)
- I
  - InputDetail, [31](#)
- Label
  - PageStructureMaker, [54](#)
- ListEnd
  - PageStructureMaker, [54](#)
- ListItem
  - PageStructureMaker, [54](#)
- ListStart
  - PageStructureMaker, [54](#)
- Login, [41](#)
  - ~Login, [42](#)
  - ConfirmPage, [42](#)
  - currentTime, [43](#)
  - emailID, [44](#)
  - Login, [42](#)
  - LoginPage, [42](#)
  - LogoutPage, [43](#)
  - NewUser, [43](#)
  - password, [44](#)
  - ReadLoginDetail, [43](#)
  - regKey, [44](#)
  - RegistrationPage, [43](#)
  - retryPassword, [44](#)
  - SelectLoginDetail, [43](#)
  - userEmailID, [44](#)
  - userPassword, [44](#)
- LoginPage
  - Login, [42](#)
- Logo
  - PageLayout, [48](#)
- LogoutLink
  - PageLayout, [48](#)
- LogoutPage
  - Login, [43](#)
- MD5, [46](#)
- Main
  - ArrangeRollNo, [10](#)
  - DateSheet, [21](#)
  - Strategy, [78](#)
- main
  - adduser-main.cpp, [83](#)
  - register-main.cpp, [100](#)
- Menu
  - PageLayout, [48](#)
- msg
  - InputDetail, [31](#)
- n
  - ReadInput, [63](#)
- NewProject
  - ClassDetail, [13](#)
- NewUser
  - Login, [43](#)
- notIncluded
  - InputDetail, [31](#)
  - InputFieldName, [38](#)
  - ReadInput, [63](#)
- notIncludedRNo
  - ArrangeRollNo, [11](#)
- numrows
  - Database, [18](#)
- OldProject
  - ClassDetail, [14](#)

- ProjectDetail, 58
- oldProject
  - InputDetail, 31
- OnChangeEvent
  - JavaScript, 41
- OnFocusEvent
  - JavaScript, 41
- outFile
  - InputDetail, 32
  - ReadInput, 64
- page
  - InputDetail, 32
- PageLayout, 46
  - ContentType, 47
  - Footer, 47
  - Head, 48
  - Header, 48
  - Logo, 48
  - LogoutLink, 48
  - Menu, 48
  - PageLayout, 47
  - PageLayout, 47
  - SetCookies, 49
- PageStructureMaker, 49
  - Anchor, 51
  - BodyEnd, 51
  - BodyStart, 51
  - Button, 51
  - CSS, 51
  - DivEnd, 52
  - DivStart, 52
  - FormEnd, 52
  - FormStart, 52
  - HTMLEnd, 53
  - HTMLStart, 53
  - HeadEnd, 52
  - HeadStart, 52
  - InputField, 53
  - Javascript, 53
  - Label, 54
  - ListEnd, 54
  - ListItem, 54
  - ListStart, 54
  - PageStructureMaker, 50
  - PageStructureMaker, 50
  - SelectFieldEnd, 55
  - SelectFieldStart, 55
  - SelectOptionEnd, 55
  - SelectOptionStart, 55
  - SetHTMLVariables, 55
  - startH1, 56
  - TableEnd, 56
  - TableStart, 56
  - Title, 56
- password
  - InputFieldName, 38
  - Login, 44
- prefix
  - ArrangeRollNo, 12
  - InputDetail, 32
  - InputFieldName, 38
  - ReadInput, 64
- prefixRollNo
  - ArrangeRollNo, 12
- ProjectDetail, 56
  - ~ProjectDetail, 57
  - AuthorizeUser, 58
  - HomePage, 58
  - OldProject, 58
  - ProjectDetail, 57
  - ProjectDetailPage, 58
  - ProjectDetail, 57
- ProjectDetailPage
  - ProjectDetail, 58
- projectID
  - InputDetail, 32
  - InputFieldName, 38
  - ReadInput, 64
- projectName
  - InputDetail, 32
  - InputFieldName, 38
- ProjectType
  - ClassDetail, 14
- projectType
  - InputDetail, 32
  - InputFieldName, 38
- Query
  - Database, 17
- ReadClassDetail
  - ReadInput, 61
  - RollNoDetail, 69
- ReadDateSheet
  - ReadInput, 61
  - RoomDetail, 71
- ReadDateSheetOut
  - ReadInput, 61
- ReadExamDetail
  - ReadInput, 61
  - SeatPlan, 73
  - Strategy, 78
- readField
  - InputDetail, 32
- ReadFieldValue
  - ReadInputField, 66
- ReadInput, 59
  - ~ReadInput, 60
  - centreName, 62
  - className, 62
  - cols, 62
  - date, 63
  - dateSheetRNo, 63
  - dateSheetRNoSize, 63
  - examSubCode, 63
  - examSubName, 63
  - FileName, 61

- fileName, 63
- inFile, 63
- n, 63
- notIncluded, 63
- outFile, 64
- prefix, 64
- projectId, 64
- ReadClassDetail, 61
- ReadDateSheet, 61
- ReadDateSheetOut, 61
- ReadExamDetail, 61
- ReadInput, 60
- ReadRollNoDetail, 62
- ReadRoomDetail, 62
- ReadStrategy, 62
- ReadInput, 60
- rollNo, 64
- roomNo, 64
- rows, 64
- strategyOption, 64
- subjectCode, 64
- subjectName, 64
- temp, 65
- totalCentres, 65
- totalClasses, 65
- totalDays, 65
- totalExams, 65
- totalRooms, 65
- totalSubjects, 65
- ReadInputField, 65
  - fi, 67
  - formData, 67
  - ReadFieldValue, 66
  - ReadInputField, 66
  - ReadInputField, 66
- ReadLoginDetail
  - Login, 43
- ReadRollNoDetail
  - DateSheet, 21
  - ReadInput, 62
- ReadRoomDetail
  - ExamDetail, 23
  - ReadInput, 62
- ReadSeatPlan
  - SeatPlan, 73
- ReadStrategy
  - ReadInput, 62
- ReadStrategyDetail
  - Validation, 80
- ReadValidation
  - Validation, 80
- ReadValidationDetail
  - Report, 68
- regKey
  - Login, 44
- register-main.cpp
  - main, 100
- RegistrationMail
  - SendMail, 75
- RegistrationPage
  - Login, 43
- RemoveNonEligibleRollNo
  - ArrangeRollNo, 11
- RemoveRedundancy
  - ArrangeRollNo, 11
- Report, 67
  - ~Report, 68
  - ReadValidationDetail, 68
  - Report, 68
  - ReportPage, 68
- ReportPage
  - Report, 68
- retryPassword
  - InputFieldName, 38
  - Login, 44
- rollNo
  - ArrangeRollNo, 12
  - ReadInput, 64
- RollNoDetail, 68
  - ~RollNoDetail, 69
  - ReadClassDetail, 69
  - RollNoDetail, 69
  - RollNoDetailPage, 69
  - RollNoDetail, 69
  - SetDefaultValue, 70
  - WriteClassDetail, 70
- RollNoDetailPage
  - RollNoDetail, 69
- RoomDetail, 70
  - ~RoomDetail, 71
  - ReadDateSheet, 71
  - RoomDetail, 71
  - RoomDetailPage, 71
  - RoomDetail, 71
  - SetDefaultValue, 71
  - WriteDateSheet, 71
- RoomDetailPage
  - RoomDetail, 71
- roomNo
  - InputDetail, 32
  - InputFieldName, 38
  - ReadInput, 64
- RoomReport
  - SeatPlan, 74
- rows
  - InputDetail, 32
  - InputFieldName, 39
  - ReadInput, 64
- sameDetail
  - InputDetail, 33
  - InputFieldName, 39
- SeatPlan, 72
  - ~RoomReport, 73
  - ~SeatPlan, 73
  - ReadExamDetail, 73
  - ReadSeatPlan, 73

- RoomReport, 74
- SeatPlan, 73
- SeatPlan, 73
- WriteHTMLFile, 74
- SelectColumn
  - Database, 18
- SelectFieldEnd
  - PageStructureMaker, 55
- SelectFieldStart
  - PageStructureMaker, 55
- SelectLoginDetail
  - Login, 43
- SelectOptionEnd
  - PageStructureMaker, 55
- SelectOptionStart
  - PageStructureMaker, 55
- SelectProjectID
  - Database, 18
- SelectQuery
  - Database, 18
- SelectSum
  - Database, 18
- SendMail, 74
  - ~SendMail, 75
  - RegistrationMail, 75
  - SendMail, 75
  - SendMail, 75
  - SetHTMLMessage, 75
  - SetMailData, 75
- sessionID
  - InputDetail, 33
- SetCookies
  - PageLayout, 49
- SetDefaultValue
  - ClassDetail, 14
  - DateSheet, 21
  - ExamDetail, 23
  - RollNoDetail, 70
  - RoomDetail, 71
  - Strategy, 78
- SetFieldNames
  - InputFieldName, 36
- SetHTMLMessage
  - SendMail, 75
- SetHTMLVariables
  - PageStructureMaker, 55
- SetMailData
  - SendMail, 75
- SortRollNo
  - ArrangeRollNo, 11
- SplitString
  - InputDetail, 29
- splitString
  - InputDetail, 33
- src/adduser-main.cpp, 83
- src/backend/arrangerollno.cc, 84
- src/backend/expandrollno.cc, 85
- src/backend/header/arrangerollno.h, 85
- src/backend/header/expandrollno.h, 86
- src/backend/header/readinput.h, 86
- src/backend/header/seatplan.h, 87
- src/backend/main.cpp, 88
- src/backend/readinput.cc, 88
- src/backend/seatplan.cc, 89
- src/class-main.cpp, 89
- src/class.cc, 90
- src/confirm-main.cpp, 91
- src/datesheet-main.cpp, 91
- src/exam-main.cpp, 92
- src/exam.cc, 92
- src/header/class.h, 93
- src/header/exam.h, 94
- src/header/rollno.h, 94
- src/header/room.h, 95
- src/header/sendmail-detail.h, 96
- src/header/sendmail.h, 96
- src/header/validation.h, 97
- src/javascript.cc, 98
- src/login-main.cpp, 98
- src/logout-main.cpp, 99
- src/register-main.cpp, 99
- src/report-main.cpp, 100
- src/rollno-main.cpp, 101
- src/rollno.cc, 101
- src/room-main.cpp, 102
- src/room.cc, 103
- src/sendmail.cc, 103
- src/strategy-main.cpp, 104
- src/validation-main.cpp, 104
- src/validation.cc, 105
- ss
  - InputDetail, 33
- startH1
  - PageStructureMaker, 56
- startRollNo
  - InputDetail, 33
  - InputFieldName, 39
- Strategy, 76
  - ~Strategy, 77, 78
  - Main, 78
  - ReadExamDetail, 78
  - SetDefaultValue, 78
  - Strategy, 77, 78
  - strategyName, 79
  - StrategyPage, 78
  - totalStrategy, 79
  - WriteExamDetail, 78
- strategyChoice
  - InputDetail, 33
  - InputFieldName, 39
- strategyName
  - Strategy, 79
- strategyOption
  - ReadInput, 64
- StrategyPage
  - Strategy, 78



StringToInt  
   InputDetail, 29  
 subjectCode  
   InputDetail, 33  
   InputFieldName, 39  
   ReadInput, 64  
 subjectName  
   InputDetail, 33  
   InputFieldName, 39  
   ReadInput, 64  
  
 TableEnd  
   PageStructureMaker, 56  
 tableHeading  
   InputDetail, 33  
 TableStart  
   PageStructureMaker, 56  
 temp  
   ReadInput, 65  
 temp1  
   InputDetail, 34  
 Time  
   InputDetail, 29  
 Title  
   PageStructureMaker, 56  
 totalCentres  
   InputDetail, 34  
   InputFieldName, 39  
   ReadInput, 65  
 totalClasses  
   InputDetail, 34  
   InputFieldName, 39  
   ReadInput, 65  
 totalCols  
   InputDetail, 34  
 totalDays  
   InputDetail, 34  
   InputFieldName, 39  
   ReadInput, 65  
 totalExams  
   ReadInput, 65  
 totalRooms  
   InputFieldName, 40  
   ReadInput, 65  
 totalRows  
   InputDetail, 34  
 totalStrategy  
   Strategy, 79  
 totalSubjects  
   InputFieldName, 40  
   ReadInput, 65  
  
 userEmailID  
   Login, 44  
 userPassword  
   Login, 44  
  
 Validation, 79  
   ~Validation, 80  
  
   ReadStrategyDetail, 80  
   ReadValidation, 80  
   Validation, 80  
   ValidationPage, 80  
   WriteStrategyDetail, 80  
 ValidationPage  
   Validation, 80  
 vecTemp  
   InputDetail, 34  
  
 where  
   InputDetail, 34  
 WriteArrangedRollNo  
   ArrangeRollNo, 11  
 WriteClassDetail  
   RollNoDetail, 70  
 WriteDateSheet  
   DateSheet, 21  
   RoomDetail, 71  
 WriteExamDetail  
   Strategy, 78  
 WriteExpandRollNo  
   ExpandRollNo, 25  
 WriteHTMLFile  
   SeatPlan, 74  
 WriteRollNoDetail  
   DateSheet, 21  
 WriteRoomDetail  
   ExamDetail, 23  
 WriteStrategyDetail  
   Validation, 80