최종 class diagram::최종 class diagram

회원시스템

«boundary»
User Interface::SignUpUI
-signUp: SignUp
+putUserDetails(member: Member&)

«control»
Control::SignUp
+putUserDetails(member: vector<Memver>&, devidedMemberType: int, name: string, number: int, id: string, password: string)

«boundary»
User Interface::WithdrawalUI
-withDrawal: WithDrawal
+clickWithdrawal(currentMember: Member, members: vector<Member>&)

«control»
Control::Withdrawal
+clickWithdrawal(currentMember: Member, members: vector<Member>): string

«boundary»
User Interface::SignInUI
-signIn: SignIn
+putUserDetails(member: vector<Member>&, currentMember: Member&)

«control»
Control::SignIn
+putUserDetails(members: vector<Member>&, currentMember: Member&, id: string, password: string): string

«boundary»
User Interface::SignOutUI
+signOutUI(member: Member&)

«control»
Control::SignOut
+signOut(member: Member&): string

지원정보관리 시스템

«boundary»
User Interface::ShowApplyInformationUI
-showApplyInformation: ShowApplyInformation
+inquireApply(memberID: string, apply: vector<Apply>&)
+cancelApply(applierID: string, apply: vector<Apply>&)
+showWorkApply(member: Member, id: string, companyName: string, apply: vector<Apply>&, recruitment: vector<Recruitment>&)

«control»
Control::ShowApplyInformation
-applies: vector<Apply>
-recruitments: vector<Recruitment>
+showApply(memberID: string, apply: vector<Apply>&): vector<tuple>
+cancelApply(companyNumber: int, applierID: string, apply: vector<Apply>&): string
+showWorkApply(member: Member, id: string, companyName: string, apply: vector<Apply>&, recruitment: vector<Recruitment>&): map<string, int>

«entity»
Apply
-companyName:string
-companyNumber:int
-work:string
-TO:int
-deadlineDate:string
-applierID:string
-writerID:string
+getCompanyName(): string
+getWork(): string
+getCompanyNumber(): int
+getApplierID(): string
+getWriterID(): string
+setApply(companyName:string, companyNumber:int, work:string, TO:int, deadlindDate:string, applierID:string)(companyName: string, companyNumber: int, work: string, TO: int, deadlindDate: string, applierID: string, writerID: string)
+getApply(): tuple
+cancleApply()

0..*
+1
0..*
+1..*

«entity»
Member
-name: string
-number: int
-id: string
-password: string
-type: string
-status: bool
+getName(): string
+getNumber(): int
+getID(): string
+getPassword(): string
+setStatus(status: bool)
+validate(member: vector<Member>&, inputId: string&, inpurPassword: string&)
+saveMember(): int
+getType(): string
+getID(): bool
+deleteMember()

+1
+1

«entity»
Recruitment
-companyName: string
-companyNumber: int
-work: string
-numOfPeople: int
-dueDate: string
+getCompanyName(): string
+getWork(): string
+getNumOfPeople(): int
+getDueDate(): string
+getRecruitment(): tuple

«entity»
CompanyMember
+getType(): string

«entity»
GeneralMember
+getType(): string

채용지원정보 관리 시스템

«boundary»
User Interface::SearchRecruitmentUI
-searchRecruitment: SearchRecruitment
+searchByCompanyName(recruitment: vector<Recruitment>&)
+applyRecruitment(applierID: string, recruitment: vector<Recruitment>&, apply: vector<Apply>&)

«control»
Control::SearchRecruitment
-apply: Apply
+showRecruitmentList(companyName: string, recruitment: vector<Recruitment>&): vector<tuple>
+addNewApply(companyNumber: int, applierID: string, recruitment: vector<Recruitment>&): tuple

채용등록 시스템

«boundary»
UserInterface::ShowRecruitmentUI
+startInterface(recruitment: vector<Recruitment>&, name: string)

«control»
Control::ManageRecruitment
+addNewRecruitment(recruitments: vector<Recruitment>&, companyName: string&, work: string&, numOfPeople: int&, dueDate: string&): string
+showRecruitmentDetail(recruitments: vector<Recruitment>&, name: string): vector<Recruitment>

«boundary»
UserInterface::AddRecruitmentUI
+createNewRecruitment(recruitments: vector<Recruitment>&, company: Member&)