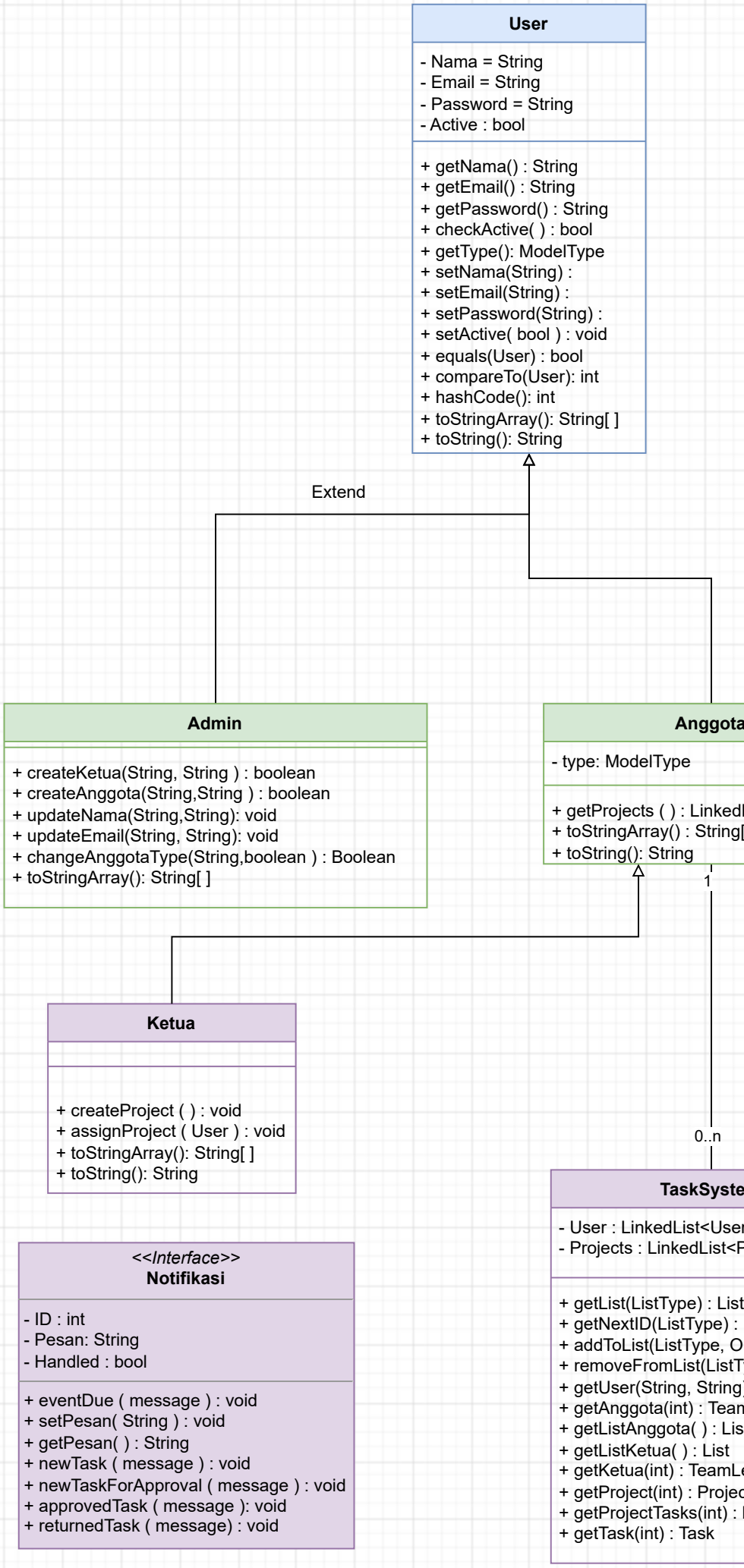
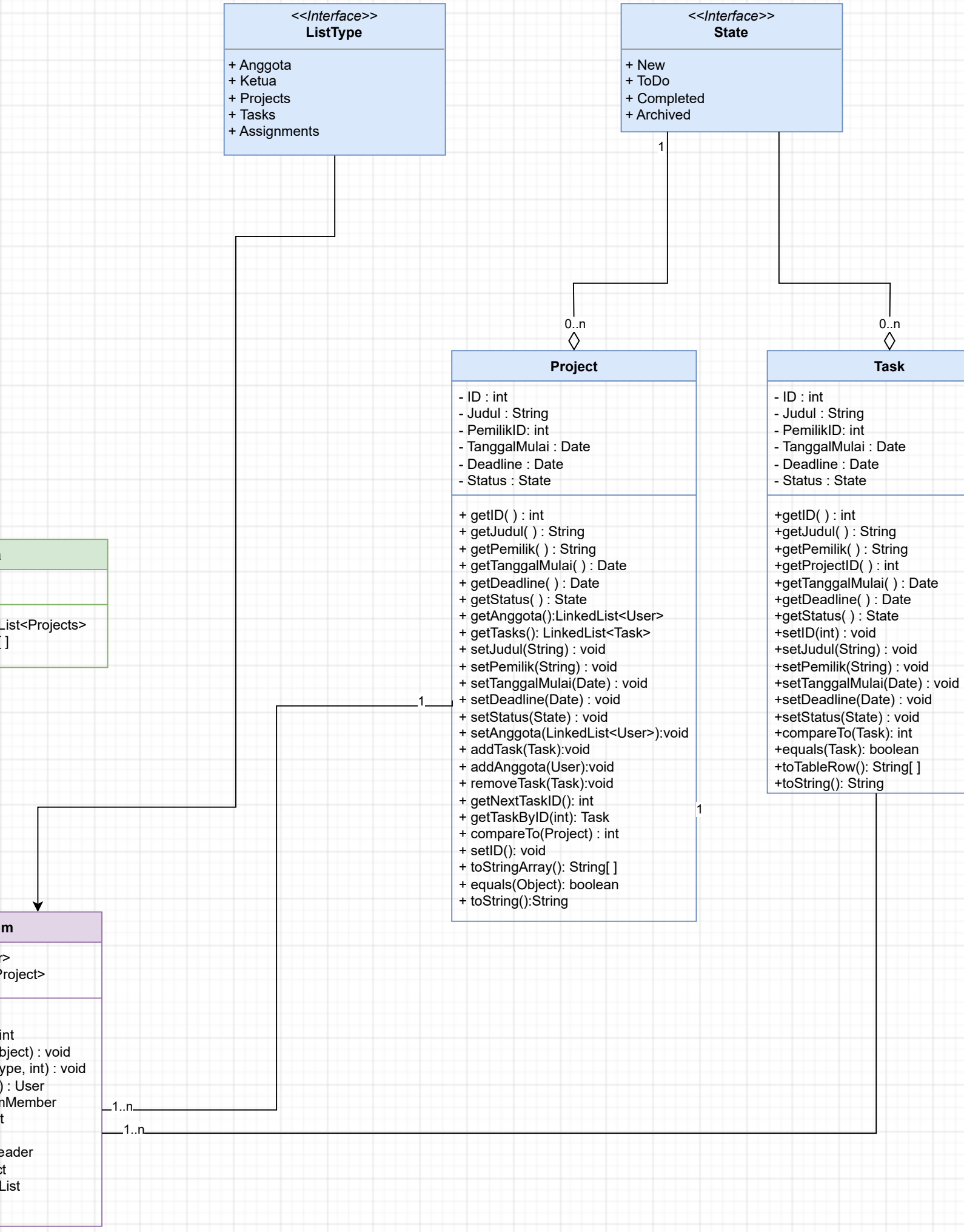

User
- Nama = String
- Email = String
- Password = String
- Active : bool
+ getNama() : String
+ getEmail() : String
+ getPassword() : String
+ checkActive( ) : bool
+ getType(): ModelType
+ setNama(String) :
+ setEmail(String) :
+ setPassword(String) :
+ setActive(bool) : void
+ equals(User) : bool
+ compareTo(User): int
+ hashCode(): int
+ toStringArray(): String[ ]
+ toString(): String
Extend
Admin
+ createKetua(String, String) : boolean
+ createAnggota(String,String) : boolean
+ updateNama(String,String): void
+ updateEmail(String, String): void
+ changeAnggotaType(String,boolean) : Boolean
+ toStringArray(): String[ ]
Anggota
- type: ModelType
+ getProjects ( ) : Linked
+ toStringArray() : String[
+ toString(): String
1
0..n
Ketua
+ createProject ( ) : void
+ assignProject (User) : void
+ toStringArray(): String[ ]
+ toString(): String
<<Interface>>
Notifikasi
- ID : int
- Pesan: String
- Handled : bool
+ eventDue (message) : void
+ setPesan(String) : void
+ getPesan( ) : String
+ newTask (message) : void
+ newTaskForApproval (message) : void
+ approvedTask (message): void
+ returnedTask (message) : void
TaskSyste
- User : LinkedList<User
- Projects : LinkedList<P
+ getList(ListType) : List
+ getNextID(ListType) :
+ addToList(ListType, O
+ removeFromList(ListT
+ getUser(String, String
+ getAnggota(int) : Tean
+ getListAnggota( ) : Lis
+ getListKetua( ) : List
+ getKetua(int) : TeamLe
+ getProject(int) : Projec
+ getProjectTasks(int) :
+ getTask(int) : Task


<<Interface>>
ListType
+ Anggota
+ Ketua
+ Projects
+ Tasks
+ Assignments
<<Interface>>
State
+ New
+ ToDo
+ Completed
+ Archived
1
0..n
0..n
Project
- ID : int
- Judul : String
- PemilikID: int
- TanggalMulai : Date
- Deadline : Date
- Status : State
+ getID( ) : int
+ getJudul( ) : String
+ getPemilik( ) : String
+ getTanggalMulai( ) : Date
+ getDeadline( ) : Date
+ getStatus( ) : State
+ getAnggota():LinkedList<User>
+ getTasks(): LinkedList<Task>
+ setJudul(String) : void
+ setPemilik(String) : void
+ setTanggalMulai(Date) : void
+ setDeadline(Date) : void
+ setStatus(State) : void
+ setAnggota(LinkedList<User>):void
+ addTask(Task):void
+ addAnggota(User):void
+ removeTask(Task):void
+ getNextTaskID(): int
+ getTaskByID(int): Task
+ compareTo(Project) : int
+ setID(): void
+ toStringArray(): String[ ]
+ equals(Object): boolean
+ toString():String
Task
- ID : int
- Judul : String
- PemilikID: int
- TanggalMulai : Date
- Deadline : Date
- Status : State
+getID( ) : int
+getJudul( ) : String
+getPemilik( ) : String
+getProjectID( ) : int
+getTanggalMulai( ) : Date
+getDeadline( ) : Date
+getStatus( ) : State
+setID(int) : void
+setJudul(String) : void
+setPemilik(String) : void
+setTanggalMulai(Date) : void
+setDeadline(Date) : void
+setStatus(State) : void
+compareTo(Task): int
+equals(Task): boolean
+toTableRow(): String[ ]
+toString(): String
List<Projects>
]
m
r>
Project>
int
bject) : void
ype, int) : void
) : User
nMember
t
eader
t
List
1
1
1..n
1..n