

Vote	Innovation	Voter	Team
<pre>-voteld : long -legit : boolean -innovation : Innovation -voter : Voter +getInnovation() : Innovation +setInnovation(Innovation) : void +getVoteld() : long +setVoteld(long) : void +getVoter() : Voter +setVoter(Voter) : void +isLegit() : boolean +setLegit(boolean) : void +toString() : String</pre>	<pre>-innold : long -innoName : String -team : Team -innoDesc : String -innoOwner : String -voteCount : long +getInnold() : long +setInnold(long) : void +getInnoName() : String +setInnoName(String) : void +getTeam() : Team +setTeam(Team) : void +getInnoDesc() : String +setInnoDesc(String) : void +getInnoOwner() : String +setInnoOwner(String) : void +getVoteCount() : long +setVoteCount(long) : void +toString() : String</pre>	<pre>-voterId : long -firstName : String -lastName : String -team : Team -type : String -voted : boolean +getVoteld() : long +setVoteld(long) : void +getFirstName() : String +setFirstName(String) : void +getLastName() : String +setLastName(String) : void +getTeam() : Team +setTeam(Team) : void +getType() : String +setType(String) : void +isVoted() : boolean +setVoted(boolean) : void +toString() : String</pre>	<pre>-teamId : long -teamName : String +getTeamId() : long +setTeamId(long) : void +getTeamName : String +setTeamName(String) : void +equals(Object) : boolean +hashCode() : int +toString() : String</pre>

```

classDiagram
    class AdminCtrl {
        -innodao
        -teamdao
        -votedao
        -voterdao
        +adminAuth() String
        +showAdmin(Model, InnovationForm, VoterForm, TeamForm, BindingResult, HttpServletRequest) String
        +showTeamAdmin(Model, HttpSession, InnovationForm, VoterForm, BindingResult, HttpServletRequest) String
        +showVotes(Model) String
    }
    class InnovationCtrl {
        -dao
        -teamdao
        -voterdao
        +addNew(InnovationForm) String
        +getDao() InnoDAOJdbcImpl
        +setDao(InnoDAOJdbcImpl) void
        +viewInnovations(InnovationForm, Model, HttpSession) String
    }
    class MainCtrl {
        -dao
        +showLogin(HttpSession, Model) String
    }
    class LoginCtrl {
        -dao
        -log
        -voterdao
        +handleLogin(LoginForm, BindingResult, HttpSession) String
        +viewLogin(LoginForm, Model) String
    }
    class TeamCtrl {
        -teamdao
        +create(String, TeamForm, BindingResult) String
        +getDao() VoterDAOJdbcImpl
        +setDao(VoterDAOJdbcImpl) void
    }
    class VoteCtrl {
        -innovationdao
        -log
        -votedao
        -voterdao
        +handleVote(InnovationForm, BindingResult, Model, long, HttpSession) String
        +getDao() InnovationDAOJdbcImpl
        +setDao(InnovationDAOJdbcImpl) void
    }
    class VoterCtrl {
        -dao
        -teamdao
        +addNewVoter(VoterForm, BindingResult, HttpSession, Strng) String
        +getDao() VoterDAOJdbcImpl
        +setDao(VoterDAOJdbcImpl) void
    }

```

```

classDiagram
    class InnovationDAO {
        +addNew(Innovation) : Object
        +find(long) : Innovation
        +findAll() : List<Innovation>
    }
    class TeamDAO {
        +addNew(Team) : Team
        +find(long) : Team
        +findAll() : List<Team>
        +findByTeamName(String) : Team
    }
    class VoteDAO {
        +add(Vote) : void
        +findByInnovation(Innovation) : List<Vote>
        +findAllVotes() : List<Vote>
    }
    class VoterDAO {
        +addVoter(Voter) : Voter
        +find(long) : Voter
        +findByVoterName(String, String) : Voter
        +updateVoted(long) : void
        +updateTeam(Voter, Team) : void
        +findByVoterTeam(String, String, String) : Voter
        +findAll() : List<Voter>
    }
    class InnoDBJdbcImpl {
        -jdbcTemplate
        -memberName
        +addNew(Innovation) : Innovation
        +find(long) : Innovation
        +findAll() : List<Innovation>
        +getJdbcTemplate() : JdbcTemplate
        +setJdbcTemplate(JdbcTemplate) : void
    }
    class TeamDAOJdbcImpl {
        -memberName
        -jdbcTemplate
        +addNew(Team) : Team
        +find(long) : Team
        +findAll() : List<Team>
        +findByTeamName(String) : Team
        +getJdbcTemplate() : JdbcTemplate
        +setJdbcTemplate(JdbcTemplate) : void
        -memberName
    }
    class VoteDAOJdbcImpl {
        -innovationdao
        -jdbc
        -log
        -voterdao
        +add(Vote) : void
        +findByInnovation(Innovation) : List<Vote>
        +findAllVotes() : List<Vote>
        +setJdbcTemplate(JdbcTemplate) : void
        +getJdbcTemplate() : JdbcTemplate
    }
    class VoterDAOJdbcImpl {
        -teamdao
        -jdbc
        -log
        +addVoter(Voter) : Voter
        +find(long) : Voter
        +findAll() : List<Voter>
        +findByVoterName(String, String) : Voter
        +findByVoterTeam(String, String, String) : Voter
        +updateVoted(long) : void
        +updateTeam(Voter, Team) : void
        +getJdbcTemplate() : JdbcTemplate
        +setJdbcTemplate(JdbcTemplate) : void
    }
    InnovationDAO <|-- InnoDBJdbcImpl
    TeamDAO <|-- TeamDAOJdbcImpl
    VoteDAO <|-- VoteDAOJdbcImpl
    VoterDAO <|-- VoterDAOJdbcImpl
  
```

The diagram illustrates the structure of DAOs in a Spring application. It shows four interfaces and their corresponding implementations. Each implementation class inherits from its respective interface and implements all its methods. The interfaces define the methods for adding, finding, and updating data, while the implementations provide the actual logic for interacting with the database using JDBC.

InnovationForm	VoterForm	LoginForm
<pre>+innold : long +innoOwner : String +innoName : String +innoDesc : String +teamName : String +error : String ----- +getInnold() : long +getXX : String +setInnold(long) : void +setXX(String) : void</pre>	<pre>+fName : String +sName : String +tName : String +vType : String ----- +getXX() : String +setXX(String) : void</pre>	<pre>+voterFirstName : String +voterSirName : String +voterTeam : String ----- +getXX() : String +setXX(String) : void</pre>
<pre>-name : String ----- +getName() : String +setName(String) : void</pre>		

```
graph TD; subgraph resources; direction TB; subgraph Row1; admin.html; innovations.html; login.html; head.html; end; subgraph Row2; results.html; teamadmin.html; main.css; end; end;
```



