Step (2) **Project Specifications**

Entities and Attributes:

Researcher: Attributes include researcher_id (primary key), first_name, last_name, email_address, and organization.

GrantCompetition: Attributes include competition_id (primary key), title, application_deadline, description, area, and status.

GrantProposal: Attributes include proposal_id (primary key), requested_amount, competition_id (foreign key referencing GrantCompetition), principal_investigator_id (foreign key referencing Researcher), status, awarded_amount, and submission_date.

Collaborators: This is implicit in the GrantProposals table, with the combination of proposal_id and researcher_id.

Reviewer: Attributes include reviewer_id (primary key), researcher_id (foreign key referencing Researcher), and conflicts_of_interest (not explicitly modeled, but assumed to be stored as a list of researcher_ids within the Reviewer entity).

ReviewerAssignment: Attributes include assignment_id (primary key), proposal_id (foreign key referencing GrantProposals), reviewer_id (foreign key referencing Reviewer), review_deadline, and review_submitted.

CommitteeMeetings: Attributes include meeting_id (primary key) and meeting_date.

MeetingDiscussions: This is implicit in the schema, with the combination of meeting_id and proposal_id.

schemas are outlined below as:

Researcher={_researcherID_, firstName, lastName, emailAddress, organization}

GrantCompetition={_competitionID_, title, applicationDeadline, Description, area, status}

GrantProposals={_proposal_id_,requested_amount,competition_id(fkGrantCompetitions),principal_investigator_id(fkResearchers),status,awarded_amount,submission_date}

Collaborators={proposal_id(fkGrantproposals),researcher_id(fkResearchers),PRIMARY KEY (proposal_id, researcher_id),}

Reviewers={_reviewer_id_,researcher_id(fkResearchers)}

ReviewerAssignments={_assignment_id_,proposal_id(fkGrantProposals),reviewer_id(fkReviewers),review_deadline,review_submitted}

ConflictsOfInterest={_conflict_id_, reviewer_id(fkReviewers), researcher_id(fkResearchers)}

CommitteeMeetings={_meeting_id_, meeting_date}

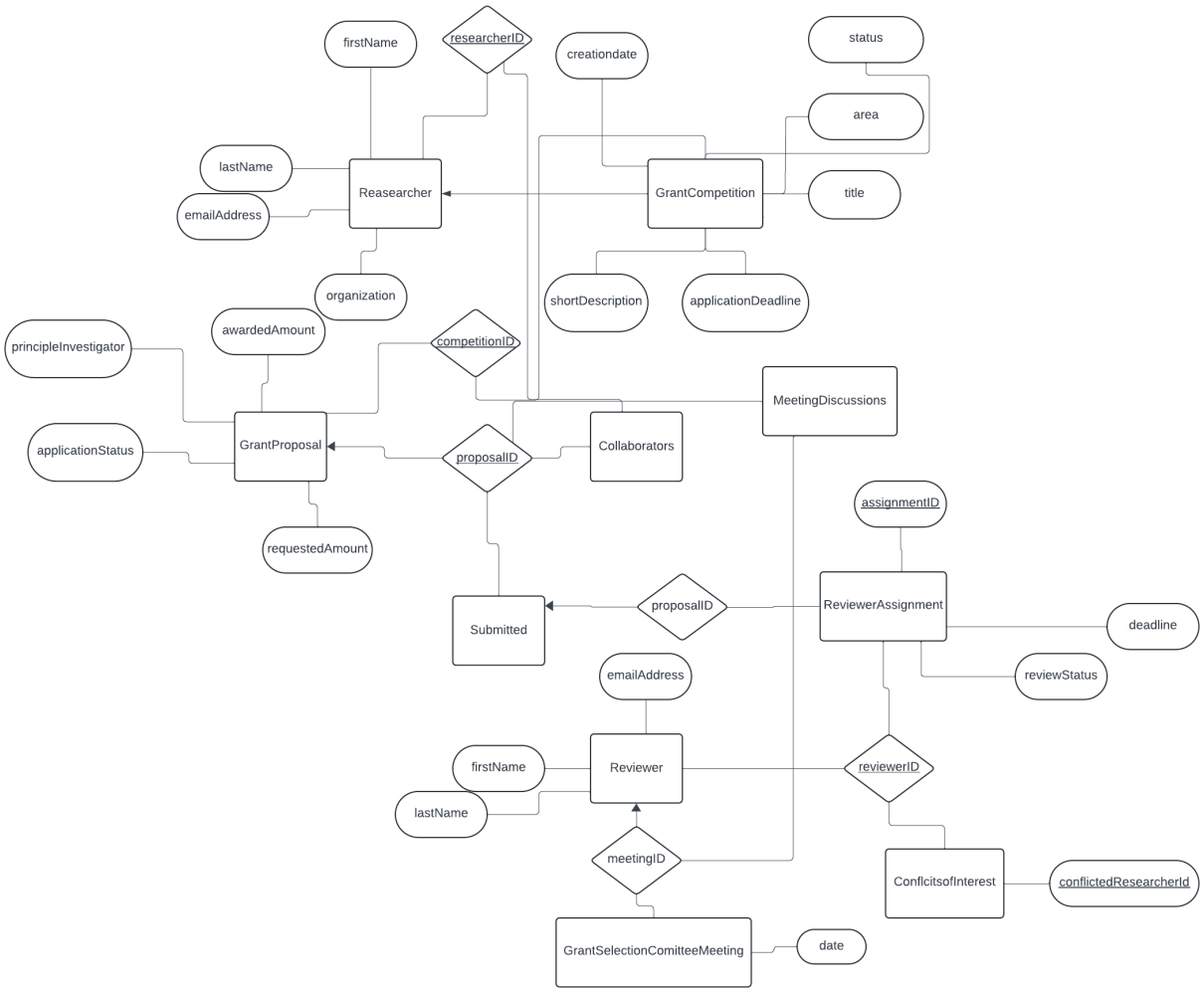
MeetingDiscussions={meeting_id(fkCommitteeMeetings), proposal_id(fkGrantProposals), PRIMARY KEY (meeting_id, proposal_id)}

Relationships:

We manage the many to many relationship between collaborators using the collaborators table

reviewers table assumes that reviewer is a role a researcher can have.

Step (3): E/R Diagrams



Step (4): Does your design allow anomalies?

Researchers Table:

researcher_id → {first_name, last_name, email_address, organization}

GrantCompetitions Table:

competition_id → {title, application_deadline,creationdate, description, area, status}

GrantProposals Table:

proposal_id → {requested_amount, competition_id, principal_investigator_id, status, awarded_amount, submission_date,competitionID}

Collaborators Table:

{proposal_id, researcher_id} → {proposal_id, researcher_id}

Reviewers Table:

reviewer_id → {researcher_id}

ReviewerAssignments Table:

assignment_id → {proposal_id, reviewer_id, review_deadline, review_submitted}

ConflictsOfInterest Table:

conflict_id → {reviewer_id, researcher_id}

CommitteeMeetings Table:

meeting_id → {meeting_date}

MeetingDiscussions Table:

{meeting_id, proposal_id} → {meeting_id, proposal_id}

For each of these relations the primary key is a candidate key so the schema is bcnf and does not allow for anomalies