Ethan Booker

Dr. Mao Zheng

CS 743 Software Verification and Validation

Homework 3

Research Grant Use Case

**Notes**

* Name: *Research Grant Application**System*
* Any faculty → research grant [provost office] → register within sys → successful then account → login and create application → only 1 app → delete/modify before deadline → after only view → emailed [outside of scope]
* Committee [faculty] review apps → decide which funded → view all, create report [comments and scores] → can edit but not delete report
  + Must login → view apps → prepare reports
* Admin [not committee] only add/delete accounts for committee members

**Consistency Check**

* The primary actors: applicant, committee member, and administrator are consistent
* Every use case narrative has unique case ID.
* All sections of narrative are not empty
* There are no extends or includes.
* **Inconsistent** with name of the system: “RAG system” vs “RGA system”
  + Located in RGA-F0
* More inconsistencies: Naming conventions for Use Case ID where started a suffix of F0 while C1 and A1 are the other starts.
* Small **Inconsistent** Case Sensitive Concern in
  + “Add Committee **M**ember” where the M is capitalized while the in the diagram “Committee **m**ember” user is not capitalized.
  + “Delete Committee **m**ember” in diagram while in narrative: “Delete Committee **M**ember”
* **Inconsistent** with “Login” for administrator to be included as a primary actor. The narrative and the diagram does not indicate that the admin can login.
* **Inconsistent** with “Logout” for administrator to be included as a primary actor. The diagram does indicate that admin can logout.
* No parameter appears in both input and output parameter in “Logout”
* **Inconsistent** with “Search and View Application” and “Select and View Application” name.
* Preconditions should have “Users must have logged in” for most of the narratives. But most do not so that is inconsistent.
* Postconditions should have “if failed” section and describe what to expect such as a “display result message/cause of error”
* Login could’ve been reduced to one use case in the diagram.

**Completeness check**

* I think a key functionality that is missing is for the Committee member to actually select which applications pass. Even though they leave scores they must still select the research grant applications which is the last big functionality of a committee..
* “Initial password” when an account is created indicates that the user will need to update their password. If that is the case then “Update Password” function may need to be added.
* Login’s success is a bit ambiguous as “ The result of matching is sent back to the user.” does not make sense, while the output parameter is none. Result of matching is exactly what?
* ‘Save’ option was brought up multiple times and then they said they would ‘discard’ whatever. The save function was not described well enough and ‘discarding’ doesn’t tell us how they will deal with the information well enough.
  + For example, create application has the discard function but will it also delete that unique application ID?
* Modify Application has a side effect where if there is no application, it will create an application for them. Perhaps modify application shouldn’t be available unless they had created an app in the first place?
  + Modify application also uses 4.1, 4.2 , 4.3 notation which is the only one of it’s kind. This perhaps could have been written better so that 4.1, 4.2, 4.3 was not needed and could be extra bullet points like: 4, 5, 6, 7...
* All the post-conditions have “if successful” but no “if failed”/”else” part as there are cases where they are not always successful.
* The design is not complete as the Admin having to login is not included in the diagram but mented in narrative.
  + Need to add the administrator “Login” and “Logout” action to the diagram.
* There are also some similar functionalities that could’ve been added such as “search for application ID”.
  + Thus I would consider adding an “include” between those similar functionalities that involve looking up by ID.
* The input and output for the most part seems appropriate to me.
* I don’t think they are missing anything important for their input and output parameters for these narratives.
* The expected constraints are specified in the exception section
  + Though some of them are redundant and can be reduced through a better design.
* I would question why you need to check if they are logged in if you can just give them the option to do those actions only if they are logged in. There should be no reason why someone can invoke these actions without being logged in.
  + Being logged in should be a precondition.
* Also there should be an “award grant” functionality as we still don’t know which faculty members have been awarded the grant.
* The successful scenarios seem good to me.
  + Perhaps a typo, but “ View Application ” has successful scenario: 3 → 5, I think they meant to type 4.
* Delete Committee Member’s remark section says that the system may perform a ‘soft delete’. This feature was not supported well since this could cause memory issues if there is no hard delete. Also, there is no information about how they will choose the ‘soft’ or ‘hard’ delete, so does that mean it’s always a ‘soft’ delete?
  + Does that mean they are still in the database when you update?
  + This does not sound complete for deleting.
* I think for completeness-wise, this design is somewhat solid but could use some refactoring to reduce the repetition, add the missing functionalities, change the name of some of those use case names that had conflicts, and lastly add some includes in the diagram as many of these actions require the same type of steps. But in terms of functionality based on the description of the domain, almost all functionality is there along with the correct actors.