**DESIGN DECISIONS – PHASE 2 THRU AT LEAST 9/30/21**

1. **User related:**
   1. Only two kinds of people register:
      1. SFF volunteers with assigned roles.
      2. People associated with ideas and/or projects.
      3. Note: Allowing other types of users to register may be more trouble than it is initially worth but it might possibly be a future enhancement if people ask for it.
   2. If a an Idea or Project user becomes a volunteer
      1. They need do nothing.
      2. SFF will add them to userrole
      3. Next time they login their role entry points will be added to their login destination choices.
   3. If a volunteer wants to become an Idea or Project user:
      1. The user will click the Share a New Idea button.
      2. The system
         1. Logs them in
         2. Sees no ideas or projects
         3. Displays the full registration form but with known data already populated.
         4. Displays links – but the links displayed are only for their SFF volunteer roles.
            1. See above in the Register module, under the ’Forms & Process(es)’ section, under the ‘if so’ paragraph, the last two steps which describe slight differences in the processes depending on where the process was initiated.
   4. Early designs included users who send messages on subjects of research interest on table users.
      1. These users were to be identified as those having sent message(s) on subject(s) listed on table ‘messagesubj’ where column ‘includeinusers’ = TRUE.
      2. It is possible, however, that some such users might not want their messaging mixed with their application information which would happen if their project emailaddress overwrites their message emailaddress. On the other hand, maintaining two separate email addresses for such users seemed to be excessively complex.
      3. Therefore, no message users will be added to table Users. This loses nothing since the system can still always search the message table for messages from these users, if desired for any report.
   5. As of 7/21/21, finally understood that different types of users (requiring different sets of info) require different tables. As a result, redefined users as: those on table User (those associated with ideas and/or projects) and sff volunteers who reside on table User\_sff.
2. **user\_sff and user\_sff\_role**
   1. Issue: User\_sff\_role must be populated after user\_sff because the foreign key constraint on user\_sff\_role references back to table user\_sff. As a result, when a user is first set up administratively, both the user\_sff and user\_sff\_role tables are populated.
   2. Solution: the user is set up administratively first. This adds the user to user\_sff & allows creating one or more roles for the user at that time. When the user registers:
      1. The user completes the additional required user data on registercpt.html and that together with the data from user\_prelim are inserted into user.
      2. At the same time, the approved roles from user\_role\_prelim are inserted into userrole.
      3. Finally the old entry for the user on user\_prelim is deleted.
      4. Because user\_role\_prelim has a cascade delete foreign key to user\_prelim, when the user is delete from user\_prelim, all user roles are also deleted from user\_role\_prelim.
3. **Ideas and Projects split into two tables**
   1. Once the SFF website is well received, it might have 20 applications for every grant. In this case, it would not make sense to mix applications with data about the conduct of actual projects. **Therefore there is an Idea (grant application) table with Ideanum and a separate Project table with projects identified by ProjectNr** and conducted by grantees. Also, the data elements for each is likely to be different.
4. **Hiistorical data saved at Idea Submission.**
   1. The identical timestamp should be used for all records timestamped for a single specific submit process. This timestamp should be the first timestamp in the process. Since database timestamps are accurate within one millionth of a second, it is perhaps conceivable that automatic database timestamps for the first step in this transaction might have a different timestamp that the last step. Therefore, the first timestamp should be saved and explicitly applied to all steps in this process. For this reason the MySQL “current\_value” function is not used for timestamps in the Ideahist, ideafinalhist or userhist table creates or updates and all individual updates that are part of the same transaction must have their timestamp set and maintained by the coded processes.
      1. At initial submission and at final submission the idea, useridea and related user data will serve as the basis for SFF grants decisions.
      2. As a result, the idea rows are timestamped in the S1 and S2 timestamp columns (see table ideastatus for the meaning of these codes) and copied to table Ideahist. Then and additional timestamp is saved to ‘savedtime’ in the row’s primary key.
      3. The table ‘useridea’ and ‘user’ rows are copied to tables ‘userideahist’ and ‘userhist’ respectively and timestamped in the ‘savedtime’ timestamp column of the receiving tables.
      4. In addition, the idea rows are effectively locked by the change in Ideastatus as described in Ref 1.a. above.
      5. User and User\_sff rows are never locked to their owners as identified by the email address on the record and user\_sff rows are never copied for this purpose.
5. **Milestones:**
   1. Initially, only projects will have milestones. Ideas will not have milestones in the database although a milestones document will be submitted listing the milestones for the idea.
   2. Milestones for projects will have their own milestones table. PK: projnr, milestonenr.
   3. The initial milestone will be the contract signed date and the final milestone could be something like the project completion date or the final report received date.
6. **Ideas**
   1. Ideas are grant requests prepared by researchers to obtain funding for their research.
   2. Ideas are relatively long and complex and may take researchers substantial time to prepare. Thus anything that can be done to help ease this burden should be done.
   3. As a result, the Idea creation process in this system is a bit complex and involves multiple tables.
   4. The life cycle of an Idea’s tables is as follows:
      1. ‘**ideafirst**’. Ideas are initially created on table ‘ideafirst’ which reserves a unique ‘ideanum’ to permanently identify the idea.
      2. ‘**ideatemp**’. As a researcher is creating an idea the early versions may contain data entry errors. Some of these errors are initially allowed to facilitate the work. As a result, this table omits most ‘not null’ constraints.
      3. ‘**idea**’. Ideas are created in two stages. Preliminary ideas are created using the “Share My Idea” input form which saves them initially to table ‘Ideatemp’ and later, as they become more mature, to table ‘idea’. Table ‘Idea’ contains all the data constraints to insure data integrity and accuracy to the extent possible.
      4. ‘**ideafinal**’. The best ideas become finalists. Much new data is collected for finalists and submitted on form “Complete My Idea”. This additional data for finalists is saved on table ‘ideafinal’.
      5. ‘**ideahist’** and ‘**ideafinalhist**’ correspond to tables ‘idea’ and ‘ideafinal’ and save snapshots of these tables at notable milestones during their development for long term retention.
   5. For more about these tables see the current version of “Tables yymmdd.docx”.
   6. Successful Ideas receive grants. Grants will be tracked by our website as Projects. That section of the website is to be designed.