Final Capstone – Things to Think About or Consider
(04/2021)

1. **Read all README files** in their entirety and understand what they are telling you. Read them several times if you need to. Refer to them as you develop.

2. Ask questions of the Scrum Master and/or Product Owner if you don't understand anything. **DO NOT ASSUME.**

3. Be sure you start the Java, back-end, API server before you run your Vue project. Verify that the server port is **8080**. If it is not **8080** the starter authorization code will not work properly.

4. **You should not have to make any changes or additions to the back-end starter code provided. Doing so could render them unusable.** Any code you write for the back-end should be in new files. If you feel you must positively, absolutely modify any of the back-end starter code, check with your instructor beforehand.

5. Create new classes in the appropriate packages for your controllers, model and DAO components. Consider creating a new package for your JDBC DAO code.

6. Create a connection in dbVisualizer to the **final_capstone** database so can examine your tables.

7. Place any tables you may need in the **final_capstone** database,

8. **DO NOT** modify the **users** table created by the *setup* script.

   The **users** table is used by the security and authorization code provided. **If you change this table in any way you may render this code unusable.**

   If you need to keep information about a user of your application, create separate table for that information. If information in the **users** table is required for an application user, consider creating a relator table between the **users** table and any the other table and/or use an SQL join/sub-query to get the related information from the **users** table you need.

9. Remember each member of team needs to create the database and tables used in your project. Be sure to save the SQL needed to create and/or insert data into you project tables are stored in the **back-end/java/database** folder of your project.

10. Be careful that only one team member is editing a specific file or making changes to that file at a time. It's OK for one person to be working on one file and another person on another, but be sure each of them has pushed their changes and pulled the other's to stay in sync. **Two people editing the same file is sure fire merge conflict.**

11. If you get merge conflicts or git errors, check with your instructor if you are not sure how to fix them. Better to ask and do it right than think you know what are doing, do it and mangle your repository.

12. Come up with a way to coordinate who is working on what files to avoid problems. Comments on the Trello card for the part of the project you are implementing is a good place for this.

13. Be sure you have pushed any and all changes you have made before you go to bed or go to an interview and let your team members know what you have pushed. Meaningful comment messages go a long way to help your team understand what you have been working on and where you are.

14. **Do not make unilateral changes to any part of your project**. Involve your teammates before you start any changes.

15. Remember, **you are a team of peers**. No one person is in charge and all team members should contribute to the project solution. If you feel you are being marginalized by your team or otherwise not having a good experience, let your instructor know.