Project Phase 3 Deliverables: Fall 2017

Revised: 10/15/2017

Congratulations! You've made it to Project Phase 3, the Implementation phase. Your task now is to implement the system you've been designing to this point.

IMPORTANT NOTE: This is a standard document used semester after semester, but please read to the end, there is an important addition!

Team Projects are designed to be collaborative and allow those teams with extensive background the freedom to expand on their skills. We have suggested the use of the various *AMP or *APP stacks, implementing in MySQL or PostgreSQL and PHP v7, but you are by no means limited to those, you are welcome to implement your system in any language or platform you wish (Python, Java, Ruby), subject to the caveat that you must **write your own SQL** and implement your own logic on the database. Use of tools such as hibernate and others which handle your database interaction (and even your schema design) are strictly prohibited. **Note**: **SQLite is not allowed** for this course, if a submission includes the use of a SQLite DB, no points for the SQL portion of project will be awarded. A full featured *relational* DBMS is required for the project: teams are free to use PostgreSQL or MySQL if desired. (Non-relational noSQL: Hadoop, Cassandra, MongoDB, etc. are not allowed)

We have also said that you have great latitude in your UI, as this is not a course in user interface design, as long as you have a functional UI the details do not matter. You do, however, have to have one, do not make the mistake a team made a few semesters ago of trying to claim that their "UI" was just post requests using the Postman tool and JSON or XML payloads. If you are unsure whether your choice of UI will be acceptable, please ask us in a private message to your Team+Instructors group on Piazza. This shouldn't be needed for most of you, but if you think you're being too clever by half, just check with us first.

How You Should Work:
We require teams to perform all of source code sharing on the assigned GaTech GitHub Enterpirise account. This will ensure that you have checked in everything necessary, because if anyone fails to submit something important, the rest of the team will pick up on it and make sure it's corrected. This way, when you submit your final commit on T-Square, we will not have to figure out how to handle the files you intended to submit but didn't realize Git had not properly committed.

The expectation is to have a fully functional finished product (no page crashes, all buttons work, data validation and error handling correctly handled, all queries successfully commit to the backend, reports are accurate, etc.) In addition, business logic constraints must be ensured and your app must prevent users from performing those actions which are prohibited (grayed out/hidden buttons, notifications, etc.).

Students have a wide variety of backgrounds and skill sets which means completing each phase may be easy for some and time consuming for others. The amount of time spent on the project is not a reflection of grade performance. Always ask yourself if the deliverables meet the assignment expectations, if unsure of expectations, ask prior to the deadline for clarification. When grading your submissions, we are only looking at the deliverables themselves, not on how much effort it took to generate those deliverables.

All team members should participate in the actual coding of the source files (. php, .sql, .html, .rb). Keep in mind, phase 3 is the most time consuming of the three project phases. The best strategy is to rank the source files by difficulty then assign the source file to each member based on individual coding background. Individual coding style will not be considered during the Phase 3 grading. Do NOT wait until the last two weeks to begin coding as this will reflect poorly on your demo!

All team members are expected to pull equal weight toward the project. There are many online resources available to learn enough to be successful with the project in the time allowed. Again, all team members are expected to code their source files individually and **merge those into the master branch via pull requests**. If a specific team member is not coding their files as the deadline for phase 3 approaches, reach out to the Instructional Team for guidance.

There will be **two deliverables** for this phase of the project. The **first major deliverable** is your implementation must be done and checked in to **GitHub by the deadline posted on T-square**. You will indicate this by submitting just as you have for the first two phases. If you don't remember how that was done, please refer back to the deliverables documents for Phase 1 and 2.

Please include a **TeamInfo.txt** file in the root directory of your project indicating which file/files contains the SQL implemented by your project. The grading TA will want to look at your SQL, and it will be much easier for us to find if you tell us which file(s) that's located in! MAKE SURE you have pushed your changes to GitHub, not merely committed locally. We need to be able to check out your change. If you are unsure how to get the commit ID, one way is to go into your project directory and type "git log".

This will give you a long list of output, and each of your commits will begin with a line like this:
"`commit: 40da96c63a0a82d71e531cf137b6c95e282f9289`" where the commit ID is the part after "commit".

The **second major deliverable** for this phase of the project will be a **BlueJeans demo** for one of the TAs. Sometime in the week leading up to the project deadline, we will put up a sign-up sheet on Piazza for teams to choose a time to demonstrate their project to an assigned TA. Each slot will be a *maximum* of **25 minutes**. Your specific Phase 3 grading TA will reach out to you on Piazza after we process the signup sheets to notify the TA/Team number pairing.

One team member will act as "Presenter" for the Phase 3 demo and need to screen share. The Presenter will run your team's project code local to his/her device using a **good internet connection (>15 Mbps**). Again, do not publically share your code as this violates the GT Honor Code.

The goal of the demo is to ensure your app works and the interaction between the UI (buttons/forms) and DB (insert/delete/ select) is correct for all queries. Sequentially demonstrate the functional requirements are met. Note: any functionality which is not covered (ran out of time) or does not work (server crashes, db connection issues, Error 500, etc.) during the demo will count as 'missing' with applicable point deductions. For this reason, we recommend teams NOT use a **web hosting/cloud service** for the demo, instead your Presenter will run your team's project code local to his/her device. Practice prior to the demo to ensure your Presenter does not run out of time. Each team has one shot of performing the demo as we do not have the TA resources to redo student demos. In short, **if we did not see it during the demo, that functionality or requirement was not met**, therefore regardless of it works at the code level, point deductions will apply.

Details on **sample/seed data guidelines** (INSERT INTO table) will be posted on Piazza at a later date prior to the demo. Teams should be prepared to add this data for the demo. We recommend using a scripting language (Python/bash) to INSERT seed data efficiently. We will also be asking your Presenter to demonstrate some specific test cases to ensure your app works as expected, details are to follow on Piazza.

Team members who are able to attend should be prompt, most of the time there will be another team following you, and so if you are late, it will cut into your time and you may not be able to demonstrate your system, which will likely negatively impact your grade! We suggest that you have a "backup presenter" ready to also run the code if something happens to your main Presenters computer during the demo.

Demo Timeline: 25 minutes total per team

~0-1 min TA gives the green light and begins recording the demo session
~1-5 min Team introductions, Presenter performs git clone/checkout, then runs 'team0##_p3_complete_v7.sql'
~5-20 min Presenter performs demo (backup presenter on standby)
~20-25 min Answer TA questions

Regrade Requests: For any project regrade requests, please offer specific evidence as to why points should be returned. Because the grade is "lower than we expected" is not sufficient reason for honoring a grade change. Be sure to include "p# *grade change request*" in the title of your Piazza post and itemized list, so the instructional team do not confuse your regrade request with a question seeking clarification only. **Regrade requests will only be considered for 1 week after feedback is returned**. As phase 3 is at the end of semester, any requests need to be addressed quickly prior to the term finishing.

Feel free to post a private question: (Team### + Instructors) if you need further clarification for your team project.

Thank you in advance!

CS6400 Instructional Team

The important note specific to Fall 2017: Our project this time has turned out to be much larger than we had anticipated, which worked quite well for Phase 1 and Phase 2, but probably focuses too heavily on the UI aspect of things rather than the

database for Phase 3. We will therefore be paring down the list of requirements that need to be implemented for Phase 3. Unfortunately, this decision came out of some very recent comments by some students, and so we have not yet had time to pare down the implementation details. Please be watching on Piazza on or before Thursday morning, we will post details as to which specific requirements need to be met.