

Semester Project Phase 2

CS 3200 | Summer 2 - 2025

Due: Tuesday, August 5, 2025 @ 11:59 pm

(Due Tuesday so that it isn't due the night before Exam 2!)

Overview of Phase 2

In Phase 2, using your output from Phase 1, your team will design and develop your data model and initial Database in MySQL, including some sample data. Finally, you'll return to your user stories and write SQL CRUD queries for each.

Your Tasks for Phase 2

1. For each of your app's personas, develop a localized conceptual data model as an ER diagram representing the data needs for that particular persona. Reference your personas, user stories, and wireframes (update these as needed). Be as detailed and complete as you can in these localized diagrams while keeping the following in mind:
 - a. Do not include any entities or attributes in the localized conceptual models that aren't needed by that persona.
 - *For example, if you're developing an application to manage a toy manufacturing company, the design engineering team that creates, prototypes, and tests new toys probably doesn't need to know anything about customers or salespeople. Similarly, that company's customers likely don't need a view into the quality assurance and testing done by the design engineers.*
 - b. When conceptual models of two personas refer to the same entity or attribute, use consistent naming in both diagrams. This will help when merging them into the global model
 - *For example, if two of the localized conceptual data models for the toy company mentioned above refer to the company's customers, label both entities with the same name, **such as customers**.*
2. From the localized conceptual models for each persona, develop a single conceptual global data model as an ER diagram that incorporates the data needs of all personas. Be intentional about not duplicating entities or attributes. Include as much detail as possible (cardinalities, participations, weak entities if applicable, multivalued & composite attributes, etc).
3. Create a Relational Database Diagram based on the global conceptual data model from Step 2. Use the process covered in Lecture Note Set 12 (you only need to submit the final relational db diagram, not one after each process step).
4. Based on the output of Step 3, create the 1st draft of the MySQL DDL for your project's database. This file should be executable from top to bottom without generating any error messages. It should include:
 - a. The creation of the database and setting it as the current database (the **use** command)
 - b. The creation of all tables, including appropriate columns and FK constraints consistent with your relational model. Drop tables before creating them in case they may already exist.
 - c. The creation of a small amount of sample data for each table (you'll add more later). 2 or 3 rows per table is fine. You can generate this data by hand.
5. Generate a Database Diagram in Datagrip and supply that.
6. Create a SQL CRUD query for each user story of each persona.

Deliverables for Phase 2

All of the following will be added to your project's design document. All diagrams should be created professionally. Apps like Lucid Charts, dbdiagrams.io, or Miro have functionality for creating ER diagrams.

- ☐ Local conceptual data model as an ER diagram for Persona 1.
- ☐ Local conceptual data model as an ER diagram for Persona 2.
- ☐ Local conceptual data model as an ER diagram for Persona 3.
- ☐ Local conceptual data model as an ER diagram for Persona 4+.
- ☐ Global conceptual data model as ER Diagram for the entire application (this will likely be big) and a link to a publicly available version (so we can zoom in if needed)
- ☐ Relational Database Diagram
- ☐ SQL File that creates the database, each table, all constraints (can be part of the table creation), and sample data for each table
- ☐ DataGrip Database Diagram
- ☐ SQL CRUD query for each user story of each persona

Model Complexity

You may be wondering how complex your data model needs to be. I don't want to give hard and fast minimums and maximums, which could stifle creativity. I think you can probably guess that a database with 1 table and 5 attributes won't cut it. But you probably also realize that 50 tables and 1000 attributes might be overkill for a semester project.

For a project of this scope, it would be reasonable for you to have between 10 and 15 tables (including bridge tables) with 50 to 70 attributes combined in your global ER diagram. If you want to have more, great – go for it. If you want to have less, just ping me about it. I can offer you feedback or suggest ideas for expanding your product or data model.

Remember: While your MVP UIs will be built for different user types/personas, you don't need to worry about implementing user login/authentication systems. So, in your data model, it is just fine to have users and passwords, but you don't need to have the user create accounts, add passwords, or perform login.

What you'll submit:

One person, on behalf of the team, will submit two files to GradeScope:

1. a PDF of your project's expanded Design Document based on the provided template of Items 1 - 6 from above