The lab project should be assembled in a format like a class paper. The main difference is that the subject is the student’s own lab project and may not have many or even any references. You name should be on the cover, pages should be number and the title of the project (and your name) should be on each page. It should be logically organized and professionally done (no spelling errors, clear writing). The lab project must be submitted as ONE DOCUMENT in either WORD OR PDF FORMAT.

It is intended that this is an extension of your project for DBST 651 so most of the sections should already be done except for the security section. You might have to alter the tables a bit, create a new table or add rows that can be used to illustrate that your security works. If you don’t want to reuse an old project you have the option of starting from scratch (from a blank sheet of paper).

The comments below cover everything I can think of that could be in your project. There are options, e.g., if you can fully implement your security with roles, views, grants, procedures, and triggers you would not have to have any security labels or a VPD. If you implement your security with VPD you might not need any roles, views, or triggers. If you implemented your security with the Oracle Security Labels you might not need any VPD etc. Not every example I give below applies to every project. The lab project report should be as long as it needs to be. I have had reports from 20-30 pages to 160 pages.

Long listings of big tables to not add value, but do inflate page length! You probably need less than 10-15 rows in most of the tables. Example of security implementation should be targeted (an update against 1-3 rows not the whole table so that you then demonstrate that the whole table is changed).

It should have sections such as

Introduction
Timeline
Conceptual & Logical Data Model
Physical Design
Implementation
Security
Verification of Database Security Implementation

Introduction:

What is the project about? (A pet store, a small corner supermarket, a doctor’s office, etc). What are the key users and what are their functions? (A pet store has an owner, clerks, and maybe an office manger that does the banking, payroll)

Timeline:

the logical series of steps that must be done and when they must be done and what order

Conceptual Model:

The functional roles of the various users and database and user requirements, design assumptions, entity types and relationships, ER diagram(s), constraints/dependencies,

Logical Model:

Rules Used to Map ER Diagram to Relations, ER-to-Relational Mapping, Map of ERD to relations (tables), functional dependencies, normalized relations (should all be 3NF or BCNF) Notional Data Elements for each Table

Physical Model:

DDL to create basic relations (tables, indexes if any, non security related triggers, procedures or functions). Security DDL/DML should be in the security section

Implementation:

Insert statements into Tables, queries to show table populations (Select *) – you should only have a small number of rows in each table, enough not to be trivial, but no need to have more that one-five record of each type (one owner, 2-3 clerks, one office manager, three different types of pet, etc); examples of the most frequent queries for the business (e.g., what pets do we have in inventory, what is the cheapest or most expensive pet (could list by price) what was the payroll last week, month, year)

Security:

The most important section: What is your security plan? What are your security policies that support that plan? What security procedures will support the plan and policies? What is your security model? And the security implementation (users created with passwords, roles created, privileges granted, views created for security reasons, security triggers, procedures, functions, labels, etc). Evidence of the implementation (scripts that show the DDL to create the users, grant the privileges, etc and the results (able created, grants affected, etc). Evidence that the security works as it is supposed to (users can do what they are supposed to be able and can not do things that they are prohibited from doing or that they are not explicitly supposed to do

References: If any