

<b>EVALUATION CRITERIA</b>	<b>GRADE</b>
Project Evaluation By Supervisor	5%
<ul style="list-style-type: none"> <li>• Project Assessment</li> </ul>	
DB Evaluation by Committee Members	15%
<ul style="list-style-type: none"> <li>• Logical ERD <ul style="list-style-type: none"> <li>○ Notation/Style</li> <li>○ Completeness</li> <li>○ Correctness against Reqs</li> </ul> </li> <li>• Documentation <ul style="list-style-type: none"> <li>○ Completeness</li> <li>○ Correctness Against ERD</li> </ul> </li> <li>• Relational Schema <ul style="list-style-type: none"> <li>○ Relations</li> <li>○ Constraints</li> </ul> </li> <li>• Build Script : create table statements <ul style="list-style-type: none"> <li>○ Completeness</li> <li>○ Correctness</li> </ul> </li> <li>• Script to populate the tables with three rows <ul style="list-style-type: none"> <li>○ Completeness</li> <li>○ Correctness</li> </ul> </li> <li>• Use of procedural programs to solve complicated tasks such as constraints/business rules (procedure, trigger, function, package –min two) <ul style="list-style-type: none"> <li>○ Completeness</li> <li>○ Correctness</li> </ul> </li> <li>• Views to support forms and reports (min two) <ul style="list-style-type: none"> <li>○ Completeness</li> <li>○ Correctness</li> </ul> </li> <li>• Security/Privacy: password encryption, account profiles regarding number of login attempts etc. <ul style="list-style-type: none"> <li>○ Completeness</li> <li>○ Correctness</li> </ul> </li> <li>• Backup/Recover strategy and scripts, schedules <ul style="list-style-type: none"> <li>○ Completeness</li> <li>○ Correctness</li> </ul> </li> </ul>	
Peer Evaluation	5%
<ul style="list-style-type: none"> <li>• Did reasonable share of work</li> <li>• Was cooperative/positive and helpful, did agree upon tasks</li> <li>• Contributed to ideas/planning</li> <li>• Was available for communication</li> <li>• Contributed to overall project success</li> </ul>	
Project and GUI Evaluation by Jury Members	15%
<ul style="list-style-type: none"> <li>• Project Demonstration <ul style="list-style-type: none"> <li>○ Overall Presentation was well organized</li> <li>○ Overall Demonstration was well organized</li> <li>○ Help facility is easy to follow and enlighten the readers</li> <li>○ Application is fully functional (no error at runtime)</li> <li>○ Simplicity of use/User friendliness of application</li> <li>○ Effective in conveying marketing information</li> <li>○ GUI</li> </ul> </li> </ul>	
Report Evaluation By Committee Members	10%
<ul style="list-style-type: none"> <li>• Report Organization and Layout</li> </ul>	

- Cover Page & Table of Content
- Font & Size
- Page Number & Figure Number
- Overall Consistency/Aesthetics
- Report Content
  - Language Effectiveness
  - Completeness
  - Reference/ User's Guide is usable

---

Student Evaluation By Jury Members 10%

- Self-Confidence
- Knowledge of speaker
- Responses to the Questions
- Organization of Content/slides
- Clarity of Content
- Quality of content/Slides (appropriate amount of material)
- Presentation Skills/Professionalism
- Support main points
- Language usage (appropriate and professional, no slang)
- Dress/Appearance
- Eye contact, expression and Body composure (poise, posture, gestures)
- Effective use of time given for presentation

---

Student Evaluation By Supervisor 5%

- Technical ability/Competence: Capable of completing his/her part of the project, developed the necessary skills throughout the project lifecycle
- Team participation/Interpersonal Skills
- Responsibility/Promptness: Completed all assignments in a timely manner
- Initiative: Leadership, effective decision making
- Quality of Work: The quality of work done is up to standards
- Attitude/Enthusiasm: Projected positive attitude throughout project
- Contributions: Reflected an understanding of the core IT curriculum
- Participation: The level of participation in all phases of the project
- Attendance: How often did the team member attend a group meeting

---

UML Evaluation By Committee Members 20%

- Use Case Diagram
  - General Use Case
  - Partial Use Case
  - System Boundary
  - Includes Relationship
  - Extends Relationship
  - Generalization Relationship
- Design Class Diagram
  - Methods
  - Attribute Types
  - Visibility
  - Interface Class
  - Controller Class
- Use Case Templates
  - Completeness
  - Normal Flow
  - Alternative Flow

- Numbering
- Sequence Diagrams
  - Conditional Statements
  - Interface Objects
  - Controller Objects
  - Data Object
- Package Diagram
  - Completeness

---

Source Code Evaluation By Committee Members

15%

- Straightforward to understand the implementation architecture.
- Straightforward to understand individual source code files and how they fit into the implementation architecture?
- Source code is structured into modules or packages.
- Source code structure relates clearly to the architecture or design.
- Structure of the source code repository and how this maps to the software's components is documented.
- Source code is commented.
- Source code is laid out and indented well.
- Source code uses sensible class, package and variable names.
- There are no old source code files/depreciated commands that should be handled by version control.
- There is no commented out code.
- There are no TODOs in the code.
- Codes written have been separated from the auto generated codes.
- How to regenerate the auto-generated source code is documented.
- Coding standards used are the ones recommended by the project proposal.
- Project-specific coding standards are consistent with community or generic coding standards.