

# EPICS: Engineering Projects in Community Service

## PROJECT EVALUATION CRITERIA AND DELIVERABLES

***DUE DATE: No later than the last day of classes (only an e-copy need to be uploaded on our website)***

As stated in the syllabus, the following criteria will be used to grade your overall project:

- **Project completion and presentation:** Demonstration of your completed work to your instructor, class and client(s).
- **Quality of final report:** This includes completeness, correctness, writing style, clarity and effectiveness of communication. A recommended layout of the content of your final report is described below.
- **Project poster or webpage.** Display important highlights of your project (include pictures, images, etc.). Work with our webmaster to ensure the presence of your project on our main EPICS webpage.
- **Effective project planning and professionalism.** You will be judged on such aspects as regularity in delivering project milestones throughout the semester and overall professional attitude.

Your project's final report will be comprehensive and it will include all the artifacts and related documentation that you have created and maintained through-out the semester chronologically. A recommended organization of your report is as follows (Note: Depending on the type and status of your own project some of the sections below may not apply to you. So, you have some freedom to adjust the format of this report, as you see appropriate.)

**Cover page** (course name, team name, team members, semester, project name, etc.)

**Table of contents**

**Summary/Abstract**

**Chapter 1: Introduction**

- Problem statement and objectives
- Motivation and rationale
- Description of the customer and developers
- Your overall approach and process model
- Glossary and terminology
- Organization of the report (briefly describe what is included in this report)

**Chapter 2: Requirements Specifications**

- Description of functional (e.g. features) and non-functional requirements (e.g. performance)
- Assumptions and constraints

**Chapter 3: Architecture**

- Overview of high-level system architecture to include:
  - System services (features)
  - System structure (components and their relationships)

- System communication (interaction between components)
- System functionality (responsibilities of each component)
- Architectural decomposition and style (client/server, web-based, three-tier, etc.)
- System platforms (hardware & software)

#### **Chapter 4: Design**

- Description of the user interface (include screen print-outs)
  - Features/operations
  - Layout and aesthetics
  - Organization of window displays, dialog boxes and menus
  - Printout report formats
  - Navigation and browsing options
  - Error handling dialogs

#### **Chapter 5: Implementation**

- Selection of implementation language(s)
- Coding standards and comments used
- Implementation process and distribution of work
- Organization of the code base (e.g. directories, files, packages, classes etc.)

#### **Chapter 6: Quality Assurance & Testing**

- Describe the selection of your testing objectives
- Explain your basic testing approach and method(s) used
- Describe related checklists and/or templates used for testing purposes
- Explain your defect detection, reporting and management process
- Include some documented sample runs of the code (using several test data cases, errors detected and how they were fixed)

#### **Chapter 7: Project Organization & Management**

- Describe your team's organizational structure
- Explain the role and detailed contribution of each member in the team
- Clearly describe how the overall work was divided and carried out by different members
- Describe your project's management process (e.g. conducting meetings, scheduling, communication, planning, reporting, conflict resolution, etc.)
- Describe any related technologies and tools selection and use
- Include all Weekly Status Reports (WSR)
- Provide a detailed user's manual (e.g. instructions on how to operate your system)

#### **Chapter 8: Future Work**

- Explain work that needs be accomplished in the future
- Include any additional functionality or requirements that can be implemented by the next team

**References/bibliography** (include all references and/or websites you used)

**Appendices** (complete source code, presentation slides, customer and peer evaluation forms, etc.)