CSI 3370: Project Description

Fall 2022

Group Project (30 points)

Due: 12/1/2022 (Thursday)

In this project, you will develop a software system as a practice of applying the methods and techniques discussed in this course. The project is teamwork of 5 people. Each team will select their own project. Teams will be announced on 9/16 (Friday) after the last day of drop-off. If you have a team formed, it should be informed to the instructor no later than 9/15 (Thursday). No request after 9/15 will be taken. The following shows examples of topics.

Examples

Standalone applications

- Text editors
- Email clients
- Drawing tools
- Messengers
- Music/video clip players
- Music players
- Image editor/steganography
- Security surveillance apps
- Software visualization

Mobile applications

- Mobile security
- Parking lot tracking
- Smart home
- Restaurant reservation
- Park facility reservation
- Multi-level/graphical authentication
- Games
- Privacy protection
- Smart GPS (e.g., information sharing such as road construction, traffic accidents)
- GPS-based attendance taker
- Privacy control
- Distributed computation
- Healthcare
- Biometric authentication and identification
- Virtual reality
- Reading pictures

Web applications

- Online healthcare system
- Skill/knowledge/expertise/service sharing (e.g., languages, technical skills)

- Data analysis
- Web browser plugins (e.g., ads blocking, malicious link detection)
- Flight gate/baggage claim finder
- Image searching
- Converting text to cartoon
- Online voting system

System applications
- Energy saving
- Bluetooth connection
- CAN bus security in the automotive domain
- Network traffic monitoring
- Task manager
- Performance booster
- Workload balancing
- Image encryption/decryption

The project has five phases – Phase 1: Project Development, Phase 2: Requirements Analysis, Phase 3: Design Modeling, Phase 4: Testing, Phase 5: Final Submission. Each phase has required deliverables and deadline. There will be a submission link created for each phase.

Phase 1: Project Development

In this phase, the team will focus on developing the project and analyzing requirements. The following are deliverables.

- Project Description
  - Project title
  - Team members & roles
  - Problem Statement
    - Describe the problem to be addressed by the development.
  - Objectives and Impact
    - Describe the goal of the development and its potential impact on individuals, organizations, and society.
  - Functional requirements
  - Non-functional requirements
  - Target Environment
  - Technologies & tools
    - Describe technologies and tools being used for the project. The following are example categories.
      - Development platform
      - Programming languages
      - Database
      - Network protocols
      - Web technologies
      - Algorithms
      - COTS/Open source components

        - Security techniques
        - User interface
        - Hardware
        - Simulation

- Process model
  - Describe the process model (e.g., Scrum, UP, XP, Spiral, Waterfall, or any combination) being used for the project and a justification for choosing the model.

- Project schedule and estimation
  - The project schedule for the activities in the process model.
  - Effort estimation (e.g., person hours, line of code)

- Risk analysis
  - List of potential risks (e.g., learning curve for new technologies, a team member falling behind or dropping the course) with priority during the project and contingent plans for the risks.

- References
  - List of references (e.g., books, papers, web sites) used in the project description.
  - Examples
    - Journal paper
      - D. Ferraiolo, R. Sandhu, S. Gavrila, D. Kuhn, and R. Chandramouli, "Proposed NIST Standard for Role-Based Access Control," ACM Transactions on Information and System Security, vol. 4, no. 3, pp. 224–274, 2001.
    - Conference paper
      - Z.Idrus,S.Abidin,andN.Omar,"ManagingCSCW'susersinscripting language," in Proceedings of the 4th International Conference on Future Generation Communication Technology, Luton, UK, July 2015.
    - Book
      - R. C. Martin, Agile Software Development, Principles, Patterns and Practices, NJ: Prentice Hall, 2002.
    - Website
      - T. Peham, "The Challenges of Legacy Software", April 2019. [online], Available: https://dzone.com/articles/the-challenges-of-legacy- software. [Accessed October, 2019].

Submit the deliverables via Moodle. **Only one submission is needed per group. Due: 10/3 (Monday)**. Name the file as "**Team#-Phase1.pdf**". This submission is for feedback only and will not be graded. The feedback should be reflected faithfully in the final report. Otherwise, points will be deducted.

Phase 2: Requirements Analysis

- Use case diagram. The diagram should contain a minimum of five use cases.
- Fully dressed use case specification for each use case in the use case diagram

- Domain class diagram based on use case specifications

Submit the deliverables via Moodle. **Only one submission is needed per group. Due: 10/12 (Wednesday)**. Name the file as "**Team#-Phase2.pdf**". This submission is for feedback only and will not be graded. The feedback should be reflected faithfully in the final report. Otherwise, points will be deducted.

Phase 3: Design Modeling

- Design model
  - Design sequence diagram (DSD) for the success scenario of each use case
  - Design class diagram (DCD) based on sequence diagrams. Cass diagrams and sequence diagrams should be consistent.

Submit the deliverables via Moodle. **Only one submission is needed per group. Due: 11/2 (Wednesday)**. Name the file as "**Team#-Phase3.pdf**". This submission is for feedback only and will not be graded. The feedback should be reflected faithfully in the final report. Otherwise, points will be deducted.

Phase 4: Testing

Choose one of the following per your language use.

- OOP Languages – Unit tests for two classes (of the team's choice) that have at least 3 methods. Test cases may be JUnit cases for Java and other tools (e.g., Visual Student for C++) may be used for other OOP languages. Build tools such as Maven and Gradle may also be used. Submit the test cases and screenshots of pass for each test.
- Non-OOP Languages - Independent path tests for 3 functions of the team's choice. Submit the test cases which include input data, expected output, and actual output, and screenshots of test execution.

Submit the deliverables via Moodle. **Only one submission is needed per group. Due: 11/16 (Wednesday)**. Name the file as "**Team#-Phase4.pdf**". This submission is for feedback only and will not be graded. The feedback should be reflected faithfully in the final report. Otherwise, points will be deducted.

Phase 5: Final submission

- Project Development (3 points)
- Requirements Analysis (7 points)
- Design Modeling (7 points)
- Unit Testing (3 points)
- Implementation (10 points)
  - Source code including the folder structure. The system may be implemented in any OO programming language. If the adopted language is not OO, there will be a discrepancy between the design and the implementation. In such a case, provide a separate description on how the implementation should be interpreted to the design.

- Screenshots demonstrating the use cases in Phase 2.

Submit the deliverables via Moodle. **Due: 12/1 (Tnursday)**. The project will be graded based on the final submission.

- Group submission
  - The project report containing the deliverables as described in the above and the presentation slides. **Only one submission is needed per group.** Submit only softcopy through Moodle. Zip the project report and presentation into one single file and name it as "**Team#-Final.zip**". The person who submits the group submission on behalf of the team will also include the individual deliverables in the submission.

- Individual Submission
  - Everyone must submit the individual submission which include **peer review** and **project diary**.
  - **Peer review**: Each team member will review the contributions of other members in the team. The review form will be available in Moodle. Depending on the review, grading of individual members will be different. Missing the peer review will cause a 1-point deduction.
    - Example

      Project total = 27/30
      Peer review scores = 19/25 (a score ranging from 1 (poor) to 5 (excellent) will be received from each group member. For example, for a group of six people, up to 25 score can be received from the other five members in the group)
      Contribution-based grade = (peer review scores*project total)/max peer review = (19*27)/25 = 20.52
      No peer review: -1
      Final grade: 20.52 - 1 = 19.52

  - **Project diary**: Describe your commitment in terms of tasks, dates, and hours. Missing the peer review will cause a 1-point deduction.
  - Submit only softcopy through Moodle.
  - **Due: 12/1/2022 (Thursday).**

Class Presentation

Each team will have a 30-minute presentation in class. Every member of the team must participate in the presentation by leading some part. Presentation is scheduled on 11/28, 11/30, and 12/5.

Extra points

- The project will be voted. Top two teams will receive 3 points and other teams will receive 2 points.