

**University of North Florida
School of Computing**

CIS 4594 – Gaming and Mobile Apps Capstone II (3 Semester Credits)

Instructor: Karthikeyan Umapathy, Assistant Professor

Office location: Mathews (15/ 3214), 1 UNF Drive, Jacksonville, FL 32224

Phone: 904-620-1329 (office)

Email: k.umapathy@unf.edu

Catalog Description:

This is the second of a sequence of two courses where students will have the opportunity to analyze, implement, and deploy complex software systems as enterprise mobile applications, as computer games, and as an appropriate combination of both.

Prerequisite: CIS 4593 - Gaming and Mobile Apps Capstone I

Learning outcomes for CIS 4593/CIS 4594:

Upon completion of the two course sequence, students should be able to:

- Elicitation of requirements
- Network and software system design, including testing and redesign
- Periodic customer demonstrations of the system evolution (before the customers)
- Refine functional and non-functional requirements.
- Develop and deploy functional system.
- Test the application on a variety of mobile devices.
- Demonstrate the application to a panel of stakeholders.

Method of Teaching:

Lecture, in-class activities, group projects, and presentations

Reference books:

There is no prescribed textbook for this course. However, listed books are good reference materials for different topics that will be covered in this course.

1. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development (3rd Edition)

Author: Craig Larman

Publisher: Prentice Hall

ISBN-13: 978-0131489066 / ISBN-10: 0131489062

2. Agile Principles, Patterns, and Practices in C# (1st Edition)

Author: Robert C. Martin and Micah Martin

Publisher: Prentice Hall

ISBN-13: 9780131857254

3. Systems Analysis and Design (7th Edition)

Authors: Kenneth E. Kendall and Julie E. Kendall
Publisher: Prentice Hall
ISBN-13: 978-0-13-224085-7

4. Rational Unified Process, The: An Introduction (3rd Edition)

Author: Philippe Kruchten
Publisher: Addison-Wesley Professional
ISBN-13: 9780321197702

5. Agile Game Development with Scrum (1st Edition)

Author: Clinton Keith
Publisher: Addison-Wesley Professional
ISBN-13: 9780321618528 / ISBN-10: 0321618521

Deliverables

Team experience

Students are expected to form a team of 4-5 students. Selected project must have sufficient functionalities to ensure that each team member can make major development contributions. All students must experience various aspects of the software-development lifecycle discussed in the class and will be held accountable for their portion of the project deliverables.

Please note, project deliverables 1, 2, and 3 were produced and delivered as part of the first sequence of the course – CIS 4593 Gaming and Mobile Apps Capstone I.

Project Deliverable 4

1. Data Flow Diagram – Context Diagram

Data Flow Diagram (DFD) is a technique that depicts the processes and the data flows within a system. As the name implies, the context diagram shows the entire system in context with its environment. Create a context diagram that shows the entire system as one higher-level process and data flows to and from external entities.

2. Data Flow Diagram – Level 0 Diagram

Level 0 diagram show all the major system processes, data stores, external entities, and data flows among them. Create a level 0 diagram that decomposes the context diagram and shows more detail about the processes, how they are interrelated and data flows inside the system.

3. Architecture Model

The selection of appropriate software architectures is indispensable for a team-based software development. Software applications must be designed and implemented in parts (“modularized”) and then assembled. Architecture selection provides this modularization. Deciding on a best architecture depends upon non-functional requirements and functional requirements captured in the Use-Case Model Specifications. Select and justify the selection of appropriate software architectures for your project.

4. Detailed Design

For the detailed design, you will need to provide a complete specification of the entire software: every package, sub system, and class. Overall structure of the design model can be understood easily, by grouping design classes into packages, preferably according to adopted architectural style, then showing how these packages related to one another. Create a UML package diagram that provides an overall structure of the design model for your application. Create UML class diagrams for each package identified in the package diagram. Create other optional UML diagrams depending upon your project context.

5. Sprint backlog

Create a sprint backlog containing a list of tasks (drawn from the Product Backlog and other software development tasks) that the team completed in the current iteration and will work on for the next iteration.

6. Fourth iteration product release

As part of this deliverable, your team has to construct major portions of the software functionality for your project. While none of the functionalities are expected to be fully constructed, it is expected that necessary barebones for each use case functionality and important non-functional requirements are in place. For this deliverable, each team will demonstrate their software application as part of the fourth deliverable presentations.

7. Also include updated product backlog, user stories, sprint backlog, software development plan, and use case artifacts.

Project Deliverable 5

1. Database Design

A data model is a formal way of representing the data that are used and created by an information system. A data model shows the logical organization of data free of any implementation details using graphical drawing technique such as an entity relationship diagram. Create an entity relationship diagram shows the information that is created, stored, and used by an information system.

2. User Interface Design

User interface design is the process of defining how the system will interact with external entities (e.g., customers, suppliers, other systems). In many ways, user interface design is an art. Apply the interface design principles to create an interface pleasing to the eye and simple to users, while minimizing the effort users need to accomplish the work.

3. Hardware and Software Specification

In today's environment, the most information system is spread across two or more computers. Therefore, it is important to plan for how the system will be distributed across the computers and what hardware and software will be used for each computer. Develop a hardware and software specification that provides a list of hardware that is needed to support the information system developed by your team and the software components that will run on each hardware device.

4. Fifth iteration product release

Your team is expected to continue working on the functionalities developed from previous releases. At minimum, one or two functionalities should be fully developed and integrated with the rest of the system as per the stated requirements. For this deliverable, each student will schedule an appointment with instructor and individually demonstrate the functionality developed.

5. Also include updated product backlog, user stories, sprint backlog artifacts, software development plan, use case specifications, data-flow diagrams, architecture model document, package diagram, and class diagrams.

Project Deliverable 6

1. Test cases

Pivotal to software testing activity are the test cases. Develop test case documents that specify set of test inputs, execution conditions, and expected results developed for a particular objective, such as to exercise a particular program path or to verify compliance with a specific requirement.

2. Sixth iteration product release

Your team is expected to continue working on the functionalities developed from previous releases. At minimum, three or four functionalities should be fully developed and integrated with the rest of the system as per the stated requirements. For this deliverable, each student will schedule an appointment with instructor and individually demonstrate the functionality developed.

3. Also include updated product backlog, user stories, sprint backlog artifacts, software development plan, use case specifications, data-flow diagrams, architecture model document, package diagram, class diagrams, database design, user interface design, and hardware/software specifications.

Final Capstone Project Report

The final report for your project must include details of the system that will be delivered to the community partner. The report should include the system's context, requirements, design, implementation, testing, and installation instructions. Submit your project's source code files along with the final report document.

Bloggng as a means to Manage and Share Knowledge (Individual Activity, *Due: Daily or Weekly basis*)

In a collaborative software development, team members spend a considerable amount of time seeking information sources to help with their software development tasks. Team members quite often face problems they encountered earlier; however, they won't be able to remember information sources used or solutions adopted to solve the problem. Managing information sources and knowledge gained (i.e., lessons learned) from the sources helps the team improve their performance. Therefore, maintaining and sharing useful information sources, problems faced, and solutions adopted are critical for successful on time and on-budget completion of the project. For more information on knowledge management in the context of software development, read the research article in the below given link:

http://ase.cpsc.ucalgary.ca/uploads/Publications/holz_maurer.pdf

As part of this deliverable, student should blog about their activities using Posterous, microblogging site. The purposes of this blogging assignment are:

- To engage students in critical reflection of their software development tasks
- To impact students with skills to document their knowledge gained
- To provide alternative means for students to communicate with the instructor and team members about their efforts towards the project
- To improve collaborative software development by making team members aware of work performed by others – what team members are working, where they are working, what they are looking for, and what did they find.

Students are expected to blog about activities related to software development tasks, information needed to perform project tasks, sources considered and utilized to seek information, problems faced, solution utilized to solve the problem, and any other relevant issues faced during the project.

As a guideline, you may use following questions to structure for your blog. Please personalize your blog as per your lessons learned. Remember these questions are provided to give you some clue what to blog about. DO NOT approach this as Q&A; you will not receive any points for it.

- What tasks did you work on? Did you complete it?
- Did you encounter any problems? Describe the problem? If you did not face any problem, from where did you learn how to perform the task?
- What information sources did you use to search for solutions to your problems? Provide a list of sources used and your analysis of how the source was useful or useless?
- Describe solution adopted to solve the problem.

Students are expected to blog on a weekly or daily basis, i.e., at minimum one blog entry per week. If you worked more than one task, you can make additional posts for each task.

We will use Posterous.com, a microblogging site to blog about our activities. Students should create an account with Posterous.com. After creating Posterous account, provide your email address used to create the account to the instructor. Instructor will add you as a contributor to the UNFSocSeniorProject space (unfsocseniorproject.posterous.com), after which you will be able to post your blog this.

Students must post their activities blog entries to the UNFSocSeniorProject space (unfsocseniorproject.posterous.com).

Any content posted to the UNFSocSeniorProject will be viewable by instructor and other students in the class. Since your postings will be visible to entire class, I suggest refraining from posting any controversial, personal or conflicts to the UNFSocSeniorProject space. Instead, discuss those issues either individually or as a team with the instructor.

Weekly Sprint Review Meeting Reports (Individual Activity, *Due by every Thursday*)

During every Thursday class, student teams are expected to conduct weekly sprint review meeting to reflect on their project progress. Students are expected to report their weekly progress via Posterous microblogging site. Students are expected to use the Posterous microblogging site for sharing information and their project work status with their team members. For example, student can share their progress with assigned tasks and share relevant information they find that may be useful for their team members.

Students must post their weekly sprint report blog entries to the UNFSeniorProjectSprintReports space (unfseniorprojectsprintreports.posterous.com).

Students are expected to report the project status by answering following questions:

- What have you done in the past week? List of tasks performed and whether it complete or incomplete.
- What tasks are you planning to work on for next week?
- Do you have any problems preventing you from accomplishing your goal? Lack of skill set, lack of direction, team communication issues, community partner issues, etc.
- Do you have any useful information to may be relevant to your team member's task or help improve their performance?

Capstone Project Reflective Essay (Individual Activity)

At the end of the semester, students are expected to write an essay reflecting on their project accomplishments as well as on their strategies to become a successful information technology professional. There is no specific length or formatting requirements for this essay. Following guidelines should be used to write this reflective essay:

- Capstone Project Reflection
 - Describe your understanding of the nature of problems (or opportunity) addressed by the project.
 - Discuss about your interactions with team members. Describe both positive and negative aspects of the interaction. Describe importance of the collaboration to success of the project.
 - Discuss about your abilities to work collaboratively within a software development team. Describe roles you assumed within your team and provide an analysis of your job performance for each role assumed. Describe how your team collaboratively worked to achieve project goals. Describe how you addressed conflicts within the team.
 - Describe how this project work influences to be a better professional and citizen. Describe how you will apply computing knowledge gained at UNF to promote and improve quality of life within Jacksonville community.
- Information Technology Professional Reflective
 - Discuss your personal career goals. Describe short-term goals (5 or fewer years and long-term goals).
 - Describe how the capstone project activities increased your interest with the computing discipline.
 - Describe how the capstone project activities helped you to be better prepared for professional career. Describe how capstone project activities helped you to identify and develop professional qualities and skills.

- Identify the qualities or skills that you need to further develop to become a successful professional. Describe how you plan to develop those skills and qualities.
- Discuss your professional development plans. Describe how you plan to keep up with latest technological developments as a professional.
- Describe any recommendations for improving this course that could help you achieve your professional goals.

Method of Evaluation:

Method of Evaluation	Team Assessment	Individual Assessment
Project Deliverable 4	5%	10%
Project Deliverable 5	5%	10%
Project Deliverable 6	5%	10%
Final Report	5%	10%
Weekly Sprint Review Meeting		5%
Software Development Blogs		5%
Capstone Project Reflective Essay		5%
Project Presentation	10%	5%
In-Class Assignments		5%
Class Participation		5%
Sub Total	30%	70%
Total	100%	

Letter grades will be based on:

- 94 – 100 = A
- 90 – 93.99 = A-
- 87 – 89.99 = B+
- 84 – 86.99 = B
- 80 – 83.99 = B-
- 77 – 79.99 = C+
- 70 – 76.99 = C
- 60 – 69.99 = D
- less than 60=F

The penalty for cheating or plagiarizing on assignments will be F grade in the course. Work which is similar beyond coincidence will automatically be considered cheating by all parties.

Class participation and In-Class Assignments:

Regular class attendance and participation in the in-class activities are a necessity to excel in this class. Students are expected to maintain good student standing by regularly attending the class (more than 90% of the scheduled class) and participating in the different topic discussions. If a student cannot attend a class, please inform your team mates and instructor.

Late Assignments:

There will be a penalty of 10 % per day for late submission of assignments (including weekends and holidays).

Academic dishonesty:

No type of academic dishonesty will be tolerated. If you are caught cheating or plagiarizing on the assignments, the punishment will be the most severe penalty allowed by the university policy. The policy on academic integrity and misuse of computer equipment and computer accounts found at http://www.unf.edu/ccec/computing/Policies_Guidelines.aspx applies to this course.

School of Computing Student Symposium

CIS 4594 students are expected to attend the School of Computing Student Symposium this spring. Final project presentations will be held as part of the student symposium.

Other remarks:

- A grade of incomplete will not be given except for catastrophic illness or calamity.
- All university rules regarding classroom behavior and attendance apply.
- If a student misses a class, the student is still responsible for the material that is covered and for completing any assignments by the due date that may have been handed out by the professor in class.

Course Topics

It is expected that the student will read the chapter assigned prior to the class meetings and will have questions for the instructor on any topics the student is not sure of, or does not understand. The student is responsible for all topics presented in the text regardless of their coverage. In addition, the students will be responsible for all lecture material that is not included in the text.

Please note that below listing of chapters does not mean that all text in those chapters would be covered in this course. Only that material that very closely pertains to course would be covered. Throughout the course, Instructor would provide other supplementary materials to provide targeted guidance to team project deliverables.

Week	Topics	Chapters	Due Dates
1	Introduction and syllabus		
2	Data Flow Diagram	TB2 – Chapter 7	
3	Design Principles	TB1 – Chapter 16, 33, and 39	
4	Design Patterns	TB1 – Chapter 17, 18, 23, 24, 25, and 26	Deliverable 4 Due
5	Database Design	TB2 – Chapter 13	
6	Interface Design	TB 2 – Chapter 14 and 15	
7	Security	TB2 – Chapter 16	

Week	Topics	Chapters	Due Dates
8	Implementation	TB1 – Chapter 20	Deliverable 5 Due
9			
10			
11	Spring Break		
12	Implementation	TB1 – Chapter 20	
13			Deliverable 6 Due
14	Testing	TB1 – Chapter 21	
15	Deployment and Delivery	TB1 – Chapter 38	
16	Project Presentations		Final Report Due

Please Note

Instructor reserves the right to modify course to meet the student's needs.

Legends

TB1 – Applying UML and Patterns

TB2 – Systems Analysis and Design

Students with Disabilities

Students with disabilities who seek reasonable accommodations in the classroom or other aspects of performing their coursework must first register with the UNF Disability Resource Center (DRC) located in Building 57, Room 1500. DRC staff members work with students to obtain required documentation of disability and to identify appropriate accommodations as required by applicable disability laws including the Americans with Disabilities Act (ADA). After receiving all necessary documentation, the DRC staff determines whether a student qualifies for services with the DRC and if so, the accommodations the student requires will be provided. DRC staff then prepares a letter for the student to provide faculty advising them of approved accommodations. For further information, contact the DRC by phone (904) 620-2769, e-mail drcexams@unf.edu, or visit the DRC website www.unf.edu/drc

Military and veteran students may need both physical and academic accommodations and may contact the DRC to find further information. Military and veteran students who return from combat exposure may be utilizing the post 9/11 GI bill to continue postsecondary education goals. Contact Military and Veterans Resource Center by phone (904) 620-2655 or e-mail mvrc@unf.edu

Satisfactory Progress Policy

The School of Computing enforces the "one repeat" rule for all prerequisite and core courses offered by the School for its major programs. Students who do not successfully complete a

prerequisite or core requirement for a School of Computing course on the first attempt (i.e., earn a grade of D, F, W, WP or WF) will be granted one chance to repeat the course. Students who do not successfully complete a prerequisite or core requirement within two attempts will not be permitted to register for courses offered by the School in future semesters. This stipulation applies whether or not the student has declared a major in a School of Computing program.
http://www.unf.edu/ccec/computing/PoliciesGuidelines/Satisfactory_Progress_Policy.aspx

Community-Based Transformational Learning

Community-Based Transformational Learning is about providing students with first-hand experiences that take them outside the walls of the classroom and into the community. By engaging in these activities, UNF students learn how to translate theory into practice, strengthen their sense of civic and ethical responsibility, and gain from professional and career development opportunities. In many cases, these experiences transform the lives of students.
http://www.unf.edu/ccbl/What_is_Community-Based_Transformational_Learning.aspx