Syllabus

Virginia Programming Semester B

Course Overview

This one-semester course is intended as a practical, hands-on guide to help you understand the software development life cycle, information security risks, and game programming. This course has 18 lessons organized into four units, plus four Unit Activities. Each lesson contains one or more Lesson Activities.

Additionally, there is one Course Activity that you need to work on throughout the duration of the course. The activity is a long-term project spread over the length of the course. The due date for the activity is to be determined by the course instructor.

This course will cover the software development life cycle. This course also covers the development phases in the software development life cycle. In addition, the course includes project implementation and maintenance, and information risks and threats. Finally, the course covers game programming elements, such as game genres, game storyboards, and how to create a game.

You will submit the Unit Activity documents to your teacher, and you will grade your work on the Lesson Activities by comparing them with the given sample responses. The Unit Activities (submitted to the teacher) and the Lesson Activities (self-checked) are the major components of this course. There are other assessment components, namely the mastery test questions that feature along with the lesson; the pre- and post-test questions that come at the beginning and end of the unit respectively; and an end-of-semester test. All of these tests are a combination of simple multiple-choice questions and technology-enhanced (TE) questions.

Course Goals

This course will help you meet the following goals:

- Familiarize yourself with the different stages of the software development life cycle (SDLC).
- Explore different methods of software development, such as the waterfall method and the rapid prototyping method.
- Familiarize yourself with client and project requirements.
- Understand various tasks in the planning stage of the SDLC.
- Discuss different methodologies and tools for developing a software design.
- Create a design document using Unified Modeling Language (UML).
- Analyze various tasks in the coding stage of the SDLC.
- Understand how a system is implemented in a production environment and how companies provide product maintenance for customers.
- Describe different types of documentation used in the software development life cycle.

- Familiarize yourself with the different types of security risks and threats to computer systems.
- Explore different methods to recover from security risks and threats.
- Understand the early history of video games, including key influences.
- Explore different types of gaming platforms and how the new technologies are converging.
- Explore various game genres and their characteristics.
- Discuss the different demographics and how that differs over geographic markets.
- Create a storyboard.
- Describe the new elements of HTML5 and use the canvas element.
- Animate a simple object for a game.

Prerequisite Skills

Virginia Programming Semester B has a prerequisite course, Virginia Programming Semester A. Also, these fundamental skills will be helpful:

- basic math knowledge
- ability to visualize and apply creativity and innovation
- familiarity with the writing process and following guidelines

General Skills

To participate in this course, you should be able to do the following:

- Perform basic operations on a computer.
- Perform online research using various search engines and library databases.
- Communicate through email and participate in discussion boards.

For a complete list of the general skills required for participation in online courses, refer to the Prerequisites section of the Plato Student Orientation document, found at the beginning of this course.

Credit Value

Virginia Programming Semester B is a 0.5-credit course.

Course Materials

- notebook
- computer with Internet connection and speakers or headphones
- Microsoft Word or equivalent
- Microsoft Excel or equivalent
- Microsoft PowerPoint or equivalent
- online UML Tools
- scanner
- printer

Course Pacing Guide

This course description and pacing guide is intended to help you stay on schedule with your work. Note that your course instructor may modify the schedule to meet the specific needs of your class. Also, the course instructor will determine the due date for the Course Activity, which is a long-term project over the length of the course.

Unit 1: Software Development Life Cycle and Initial Phases

Summary

In this unit, you will learn the stages of software development life cycle. You will familiarize yourself with software development methodologies, and requirements gathering and analysis. In addition, you will learn how to plan a software project.

Day	**Activity/***Objective*	Type
1 day: 1	**Syllabus and Plato Student Orientation** *Review the Plato Student Orientation and Course Syllabus at the beginning of this course.*	Course Orientation
Extended Project	**Preparing for Industry Certification**	Course Activity
4 days: 2–5	**The Software Development Life Cycle** *Explore the different stages of the software development life cycle (SDLC).*	Lesson
4 days: 6–9	**Software Development Methodologies** *Describe and compare different methods of software development, such as the waterfall method and the rapid prototyping method.*	Lesson
4 days: 10–13	**Requirements Gathering and Analysis** *Learn to describe and analyze client and project requirements.*	Lesson
1 day: 14	**Space Jumble**	Game
4 days: 15–18	**Planning** *Learn about various tasks in the planning stage of the SDLC.*	Lesson
4 days: 19–22	**Unit Activity/Threaded Discussion—Unit 1**	Unit Activity
1 day: 23	**Post-test—Unit 1**	Assessment

Unit 2: Development Phases of SDLC

Summary
In this unit, you will familiarize yourself with software design methodologies. You will learn how to create a design document using Unified Modeling Language (UML). Additionally, you will learn how a design is coded using different programming languages.

Day	Activity/*Objective*	Type
4 days: 24–27	**Design Methodologies** *Explore different methodologies and tools for developing a software design.*	Lesson
4 days: 28–31	**Unified Modeling Language** *Create a design document using Unified Modeling Language (UML).*	Lesson
1 day: 32	**Para Jumble**	Game
4 days: 33–36	**Coding** *Learn about various tasks in the coding stage of the SDLC.*	Lesson
4 days: 37–40	**Unit Activity/Threaded Discussion—Unit 2**	Unit Activity
1 day: 41	**Post-test—Unit 2**	Assessment

Unit 3: Successful and Safe Project Implementation

Summary
In this unit, you will learn about system implementation and maintenance. You will familiarize yourself with types of documentation in SDLC. You will explore different types of security risks and threats to computer systems, and methods to recover from security risks and threats.

Day	Activity/*Objective*	Type
4 days: 42–45	**Implementation and Maintenance** *Explore how a system is implemented in a production environment and how companies provide product maintenance for customers.*	Lesson
3 days: 46–48	**Types of Documentation** *Explore different types of documentation used in the software development life cycle.*	Lesson
1 day: 49	**Para Jumble**	Game
3 days: 50–52	**Information Security Risks and Threats** *Explore different types of security risks and threats to computer systems.*	Lesson
3 days: 53–55	**Disaster Recovery** *Explore different methods to recover from security risks and threats.*	Lesson
1 day: 56	**Space Jumble**	Game
4 days: 57–60	**Unit Activity/Threaded Discussion—Unit 3**	Unit Activity
1 day: 61	**Post-test—Unit 3**	Assessment

Unit 4: Game Programming

Summary
In this unit, you will familiarize yourself with history of video games. You will explore game genres, gaming platforms and how the new technologies are converging. In addition, you will explore the influence of demographics and cultural factors in playing games. You will learn how to create a game storyboard and how to create an object using elements of HTML5 and the canvas element. Finally, you will learn how to create a game program.

Day	**Activity/***Objective*	Type
3 days: 62–64	**Video Game Beginnings and Arcades** *Describe the early history of video games, including key influences.*	Lesson
3 days: 65–67	**Platforms and Convergence** *Analyze the many different types of gaming platforms and how the new technologies are converging.*	Lesson
3 days: 68–70	**Game Genres** *Discuss various game genres and their characteristics.*	Lesson
3 days: 71–73	**Game and Player Demographics** *Analyze the different demographics and how that differs over geographic markets.*	Lesson
3 days: 74–76	**Storyboarding** *Create a storyboard.*	Lesson
3 days: 77–79	**HTML5** *Describe the new elements of HTML5 and use canvas element.*	Lesson
3 days: 80–82	**Creating a Game Program** *Animate a simple object for a game.*	Lesson
1 day: 83	**Thwack-A-Mole**	Game
4 days: 84–87	**Unit Activity/Threaded Discussion—Unit 4**	Unit Activity
1 day: 88	**Post-test—Unit 4**	Assessment
1 day: 89	**Semester Review**	
1 day: 90	**End-of-Semester Test**	Assessment