

Syllabus

Virginia Programming Semester B

Course Overview

This one-semester course is intended as a practical, hands-on guide to help you understand various phases of the software development life cycle (SDLC). This course has four Units with 14 lessons and four Unit Activities. Each lesson contains one or more Lesson Activities.

This course will cover various phases of SDLC such as analysis, design, development, testing, and implementation. This course describes software development methodologies, how client requirements are gathered and analyzed, various types of project plans, design using unified modeling language (UML), coding, types of testing, quality control, and maintenance of software systems. This course also covers various security threats and risks and the methods to mitigate them.

You will submit the Unit Activity documents to your teacher, and you will grade your work in the Lesson Activities by comparing them with 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:

- Describe various phases of the SDLC.
- Describe and compare different methods of software development.
- Discuss how client requirements are gathered and analyzed.
- Describe the different types of plans created during software development.
- Explore different methodologies for creating software designs.
- Create a design using unified modeling language (UML).
- Describe the importance of the coding phase in the SDLC.
- Describe various types of testing.
- Describe the importance of a test plan and test scripts.
- Describe how quality control helps in improving the processes and the quality of software.
- Discuss how the system is implemented in a production environment.
- Discuss how software companies provide product maintenance.
- Explore the different types of documentation created during the SDLC.

- Explore different types of security risks and threats to computer systems.
- Discuss different methods to recover from security risks and threats.

Prerequisite Skills

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

- basic math knowledge
- ability to visualize and apply creativity and innovation
- general 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, and use word processing, spreadsheet, and presentation software.
- Perform online research using various search engines and library databases.
- Communicate through email and participate in discussion boards.

For a complete list of general skills that are 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

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.

Course Structure

Unit 1: Software Development Life Cycle and Initial Phases

Summary

In this unit, you will learn about the software development life cycle (SDLC). You will also learn about the various software development methodologies, such as the waterfall and spiral models. You will also compare various software development methodologies based on their advantages and disadvantages. Additionally, in this unit, you will study the first two phases of the SDLC, namely the requirements of gathering and planning.

Day	Activity/Objective	Type
1 day: 1	Syllabus and Plato Student Orientation <i>Review the Plato Student Orientation and Course Syllabus at the beginning of this course.</i>	Course Orientation
4 days: 2–5	The Software Development Life Cycle <i>Explore the different phases of the software development life cycle (SDLC).</i>	Lesson
4 days: 6–9	Software Development Methodologies <i>Describe and compare different methods of software development, such as the waterfall method and the rapid prototyping method.</i>	Lesson
4 days: 10–13	Requirements Gathering and Analysis <i>Describe and analyze client and project requirements.</i>	Lesson
1 day: 14	Space Jumble	Game
4 days: 15–18	Planning <i>Describe various tasks in the planning phase of the SDLC.</i>	Lesson
5 days: 19–23	Unit Activity/Threaded Discussion—Unit 1	Unit Activity
1 day: 24	Posttest—Unit 1	Assessment

Unit 2: Development Phases of the SDLC

Summary

In this unit, you will study the development phases of the SDLC, namely design and coding. You will learn about various design methodologies and tools that you can use to design your software application. Design methods include object-oriented and structured methods, while tools include flowcharts and unified modeling language (UML). You will also learn to create a software design using UML. In the last lesson of this unit, you will learn about the different tasks involved in the coding phase of the SDLC.

Day	Activity/Objective	Type
4 days: 25–28	Design Methodologies <i>Explore different methodologies and tools for developing a software design.</i>	Lesson
4 days: 29–32	Unified Modeling Language <i>Create a design document using Unified Modeling Language (UML).</i>	Lesson
1 day: 33	Para Jumble	Game
4 days: 34–37	Coding <i>Describe various tasks in the coding phase of the SDLC.</i>	Lesson
5 days: 38–42	Unit Activity/Threaded Discussion —Unit 2	Unit Activity
1 day: 43	Posttest—Unit 2	Assessment

Unit 3: Testing and Quality Control

Summary

In this unit, you will learn to maintain quality in a software application. The unit begins with a lesson that describes the different types of testing. You will identify the different programming errors, such as syntax errors and logical errors. You will then learn about the testing done in the different phases of the SDLC. Further in the unit, you will learn what a test plan is and how to use it along with the test scripts to carry out the testing of a software application.

Day	Activity/Objective	Type
4 days: 44–47	Testing <i>Describe different types of testing.</i>	Lesson
4 days: 48–51	Test Plan and Test Scripts <i>Describe how to use a test plan and test scripts for testing.</i>	Lesson
1 day: 52	Space Jumble	Game
4 days: 53–56	Quality Control <i>Describe how quality control helps in improving the processes and the quality of the software.</i>	Lesson
5 days: 57–61	Unit Activity/Threaded Discussion —Unit 3	Unit Activity
1 day: 62	Post test- Unit 3	

Unit 4: Successful and Safe Project Implementation

Summary

In this unit, you will learn about the safe and successful implementation and working of a software application. The unit starts with teaching you how a system is implemented and the activities carried out to maintain the system. It further describes the different types of documentation used during the SDLC. In the latter part of the unit, you will learn about the different types of security risks and threats to your computer system and ways to mitigate and recover from them.

Day	Activity/Objective	Type
4 days: 63–66	Implementation and Maintenance <i>Explore how a system is implemented in a production environment and how companies provide product maintenance for customers.</i>	Lesson
3 days: 67–69	Types of Documentation <i>Explore different types of documentation used in the software development life cycle.</i>	Lesson
1 day: 70	Para Jumble	Game
3 days: 71–73	Need for Computer Applications	Course Activity
4 days: 74–77	Information Security Risks and Threats <i>Explore different types of security risks and threats to computer systems.</i>	Lesson
4 days: 78–81	Disaster Recovery <i>Explore different methods to recover from security risks and threats.</i>	Lesson
5 days: 82–86	Unit Activity/Threaded Discussion —Unit 4	Unit Activity
1 day: 87	Space Jumble	Game
1 day: 88	Posttest—Unit 4	Assessment
1 day: 89	Semester Review	
1 day: 90	End-of-Semester Test	Assessment