

## INTRODUCTION TO SOFTWARE ENGINEERING

Version: 2022.02.14

## 1. THÔNG TIN CHUNG

## GENERAL INFORMATION

<b>Tên học phần:</b>	Nhập môn Công nghệ phần mềm
<b>Course name:</b>	<i>(Introduction to Software Engineering)</i>
<b>Mã học phần</b>	IT3180E
<b>Code:</b>	IT3180E
<b>Khối lượng</b>	3(2-2-0-6)
<b>Credit:</b>	<ul style="list-style-type: none"><li>- Lý thuyết - Lecture: 30 hours</li><li>- Bài tập - Exercise: 30 hours (Capstone project is used)</li><li>- Thí nghiệm - Experiments: 0 hours</li></ul>
<b>Học phần tiên quyết</b>	Không
<b>Prerequisite:</b>	No
<b>Học phần học trước</b>	<ul style="list-style-type: none"><li>- IT1110E: Introduction to Programming</li></ul>
<b>Prior courses:</b>	<ul style="list-style-type: none"><li>- IT3100E: Object-oriented Programming</li></ul>
<b>Học phần song hành</b>	Không
<b>Parallel course:</b>	No

## 2. MÔ TẢ HỌC PHẦN - COURSE DESCRIPTION

Học phần cung cấp cho sinh viên các kiến thức về các hoạt động chính trong vòng đời phần mềm, bắt đầu từ một ý tưởng hoặc nhu cầu từ khách hàng, được thỏa mãn một phần hoặc toàn bộ bởi phần mềm và kết thúc khi phần mềm không được sử dụng nữa. Các hoạt động chính bao gồm: các quy trình phát triển phần mềm (bao gồm kỹ nghệ yêu cầu phần mềm, thiết kế và xây dựng phần mềm, tích hợp phần mềm, kiểm thử chất lượng phần mềm), triển khai, vận hành và bảo trì phần mềm trong thực tiễn. Trong đó, học phần tập trung vào các kiến thức về mô hình phát triển phần mềm hiện đại (thác nước, mẫu thử, xoắn ốc, lặp, linh hoạt), quản lý dự án phần mềm, quản lý cấu hình, phiên bản và đảm bảo chất lượng phần mềm. Sinh viên được trải nghiệm phát triển một phần mềm theo quy trình trong thực tiễn từ xác định yêu cầu, phân tích thiết kế, lập trình, kiểm thử, và triển khai phần mềm qua bài tập/bài tập lớn. Ngoài ra, học phần cũng cung cấp cho sinh viên các kỹ năng làm việc nhóm, thuyết trình và thái độ cần thiết để làm việc và ứng dụng các kỹ thuật trong các công ty, dự án phần mềm sau này.

The course provides students with the main activities of the software life cycle processes beginning with an idea or a need that can be satisfied wholly or partly by software and ends with the retirement of the software. The main activities include software development processes (i.e. software requirement engineering, software design and construction, software integration, software qualification testing), software delivery, operation and maintenance. The course introduces modern software models (waterfall, prototype, iterative, agile), basic software project management, configuration and version management, and software quality assurance.

Students will experience the software development process in practice from defining requirements, analysis and design, programming, testing, and software deployment through exercises and capstone projects. In addition, this course also equips students with teamwork and presentation skills as well as attitudes needed for future work in software development companies.

### 3. MỤC TIÊU VÀ CHUẨN ĐẦU RA CỦA HỌC PHẦN - OBJECTIVE AND OUTPUT REQUIREMENTS

Sinh viên hoàn thành học phần này có khả năng

After this course the student will obtain the followings:

<b>Mục tiêu/CD R Goal</b>	<b>Mô tả mục tiêu/Chuẩn đầu ra của học phần Description of the goal or output requirement</b>	<b>CDR được phân bổ cho HP/ Mức độ (I/T/U) Output division/ Level (I/T/U)</b>
<b>[1]</b>	<b>[2]</b>	<b>[3]</b>
<b>M1</b>	<b>Có kiến thức cơ sở chuyên môn vững chắc về công nghệ phần mềm để thích ứng tốt với những công việc khác nhau trong lĩnh vực rộng của ngành khoa học máy tính, kỹ thuật máy tính</b> <b>Understand background knowledge in software engineering to adapt various jobs in the wide fields of Computer Science</b>	1.1.4; 1.2.1; 1.2.3; 1.2.6; 1.2.7; 1.3.1;
M1.1	Nắm vững kiến thức cơ bản về công nghệ phần mềm, biết áp dụng trong thực tế nghề nghiệp. Understand basic knowledge of software engineering, and application in practice.	[1.1.4] (U)
M1.2	Nhận diện, so sánh và phân loại được các quy trình, vòng đời phần mềm và khả năng áp dụng trong phát triển phần mềm. Identify, compare, and categorize software development processes in software life cycles, and applicability.	[1.2.1] (U) [1.2.3] (U) [1.2.6] (IU) [1.2.7] (I)
M1.3	Hiểu và so sánh một số phương pháp và công cụ trong thực tế để phân tích và đặc tả yêu cầu, thiết kế, xây dựng, kiểm thử, thẩm định và kiểm chứng, quản lý phần mềm một cách chuyên nghiệp, theo chuẩn quốc tế. Understand and compare methods and tools in practice to perform requirements specification and analysis, software designing and building, testing (validation and verification) and software management, according to international standards.	[1.3.1] (T)
M1.4	Hiểu và so sánh một số phương pháp và công cụ trong việc phát triển phần mềm có chất lượng tốt. Understand and compare methods and tools in software quality.	[1.3.1] (I)
<b>M2</b>	<b>Nhận diện và làm chủ được các kỹ năng chuyên nghiệp và phẩm chất cá nhân cần thiết để thành công trong nghề nghiệp</b>	2.1.1; 2.1.2; 2.4.1; 2.5.1; 2.5.4;

	<b>Identify and understand the professional skills and personal qualities needed to be successful in a career</b>	
M2.1	Có khả năng xác định và hình thành vấn đề Be able to identify and state problems	[2.1.1] (IU)
M2.2	Có khả năng mô hình hóa vấn đề Be able to model the problem	[2.1.2] (IU)
M2.3	Độc lập, chủ động, kiên trì và linh hoạt trong công việc Be able to work independently, proactively and flexibly	[2.4.1] (I)
M2.4	Thể hiện tính trung thực, có trách nhiệm và tin cậy trong công việc. Có động cơ, mục tiêu trong học tập và trong sự nghiệp. Perform work honestly, responsibly and reliably. Be motivated in studies and career.	[2.5.1] (I); [2.5.4] (I)
<b>M3</b>	<b>Có kỹ năng xã hội cần thiết để làm việc hiệu quả trong nhóm đa ngành và trong môi trường quốc tế</b> <b>Identify the the social skills needed to teamwork in international environment</b>	3.1.1; 3.1.2; 3.1.3; 3.1.4
M3.1	Chủ động tham gia cũng như có khả năng thành lập nhóm phù hợp với công việc Have abilities of organization and management of software development group	[3.1.1] (IU)
M3.2	Tổ chức các hoạt động nhóm Organize group activities	[3.1.2] (IU)
M3.3	Quản lý tiến trình phát triển phần mềm của nhóm Manage the software development process	[3.1.3] (IU)
M3.4	Có khả năng hợp tác, phối hợp với các thành viên khác trong nhóm, giải quyết vấn đề Be able to cooperate with other team members, solve problems	[3.1.4] (IU)

#### 4. TÀI LIỆU HỌC TẬP - LEARNING MATERIALS

##### Giáo trình – Textbook

- [1] R. Pressman, Software Engineering: A Practitioner’s Approach. 8th Ed., McGraw-Hill, 2016.

##### Sách tham khảo - Reference book

- [1] I. Sommerville, Software Engineering. 10th Ed., AddisonWesley, 2017.
- [2] Pankaj Jalote, An Integrated Approach to Software Engineering, Third Edition, Springer.
- [3] Shari Lawrence Pleegeer, Joanne M. Atlee, Software Engineering theory and practice. 4th Edition.
- [4] Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides (1994). *A Design Patterns: Elements of Reusable Object*. Addison-Wesley Professional; 1<sup>st</sup> edition

#### 5. CÁCH ĐÁNH GIÁ HỌC PHẦN - COURSE EVALUATION

Điểm thành phần Module	Phương pháp đánh giá cụ thể Evaluation method	Mô tả Detail	CĐR được đánh giá Output	Tỷ trọng Percent
[1]	[2]	[3]	[4]	[5]
<b>A1. Điểm quá trình</b> <b>Mid-term score (*)</b>	<b>Đánh giá quá trình</b> <b>Progress evaluation</b>			<b>40%</b>
	A1.1. Thảo luận Discussion	Tự luận Written	M1.1; M1.2; M2.2; M3.2	10%
	A1.2. Luyện tập trắc nghiệm trên LMS Multiple choice test on Learning Management System	Bài tập trắc nghiệm Multiple choice test	M1.2; M2.1; M3.1	10%
	A1.3. Bài tập lớn Capstone project	Báo cáo Demonstration , Presentation	M1.2	20%
<b>A2. Điểm cuối kỳ</b> <b>Final score</b>	<b>A2.1. Thi cuối kỳ</b> <b>Final exam</b>	Thi viết Written exam	M2.1÷M2. 2 M3.2÷M3. 6	<b>60%</b>

\* Điểm quá trình sẽ được điều chỉnh bằng cách cộng thêm điểm chuyên cần. Điểm chuyên cần có giá trị từ -2 đến +1, theo Quy chế Đào tạo đại học hệ chính quy của Trường ĐH Bách khoa Hà Nội.

The evaluation about the progress can be adjusted with some bonus. The bonus should belong to [-2, +1], according to the policy of Hanoi University of Science and Technology.

## 6. KẾ HOẠCH GIẢNG DẠY – SCHEDULE

Tuần Week	Nội dung Content	CĐR học phần Output	Hoạt động dạy và học Activities	Bài đánh giá Evaluation
[1]	[2]	[3]	[4]	[5]
1	<b>Giới thiệu môn học</b> <b>Giới thiệu danh sách bài tập lớn môn học</b> (các scenario cụ thể để sinh viên rèn luyện, thực hành các kiến thức phát triển phần mềm trong khóa học) <b>Chương 1: Tổng quan về Công nghệ phần mềm</b> 1.1 Phần mềm là gì? 1.2 Phân loại phần mềm 1.3 Công nghệ phần mềm là gì?	M1.1 M3.1 M3.2	Note reading; Teaching; Introduction to Case Studies  Students sign up groups to do capstone project	A2.1

	<p>1.4 Các vấn đề trong Công nghệ phần mềm</p> <p>Introduction to the course</p> <p>Introduction to the list of capstone projects</p> <p><b>Chapter 1: Introduction to Software engineering</b></p> <p>1.1. What is software?</p> <p>1.2. Software categories</p> <p>1.3. What is software engineering?</p> <p>1.4. Problems in software engineering</p>			
2	<p><b>Chương 2: Vòng đời phần mềm</b></p> <p>2.1 Hệ thống và Phần mềm</p> <p>2.2 Vòng đời hệ thống/phần mềm</p> <p>2.3 Quy trình phát triển phần mềm</p> <p>2.4 Các mô hình quy trình phần mềm: Thác nước, mẫu thử, tăng dần, nhanh, xoắn ốc</p> <p>2.5. So sánh các mô hình quy trình phần mềm</p> <p>2.6. Thảo luận nhóm và lựa chọn mô hình quy trình phù hợp</p> <p><i><b>Ví dụ và bài tập</b></i></p> <p><b>Chapter 2: Software life cycle</b></p> <p>2.1. System vs Software</p> <p>2.2. System/Software life cycle</p> <p>2.3. Software development process</p> <p>2.4. Software process models: Waterfall, prototype, incremental, rapid, spiral</p> <p>2.5. Compare software process models</p> <p>2.6. Discuss in groups and select an appropriate process model</p> <p>Examples and exercises</p>	<p>M1.1</p> <p>M1.2</p>	<p>Note reading;</p> <p>Teaching;</p> <p><i>Exercise:</i> Compare software development processes</p>	<p>A1.2</p> <p>A2.1</p>
3	<p><b>Chương 3: Phương pháp Agile</b></p> <p>3.1 Khái niệm</p> <p>3.2 Các nguyên lý cơ bản</p> <p>3.2 Ưu, nhược điểm của phương pháp Agile</p> <p>3.3 Extreme Programming</p> <p>3.4 Scrum</p> <p>3.5 Các phương pháp Agile khác</p> <p><i><b>Ví dụ và bài tập</b></i></p> <p><b>Chapter 3: Agile Software</b></p>	<p>M1.1</p> <p>M1.2</p> <p>M3.2</p>	<p>Note reading;</p> <p>Teaching;</p> <p><i>Exercise:</i> Compare agile method and other methods</p>	<p>A1.2</p> <p>A2.1</p>

	<b>Development</b> 3.1 Concepts 3.2 Basic Principles 3.2 Advantages and disadvantages of Agile method 3.3 Extreme Programming 3.4 Scrum 3.5 Other Agile Methods Examples and exercises			
4	<b>Chương 4: Quản lý dự án phần mềm</b> 4.1 Khái niệm về dự án phần mềm <ul style="list-style-type: none"> <li>Yếu tố con người: Stakeholder/ Team Leader/ Software Team/ Communication issue</li> <li>Yếu tố Sản phẩm: Software scope/ Process/ Project</li> </ul> 4.2 Quy trình quản lý dự án phần mềm <ul style="list-style-type: none"> <li>Ước lượng dự án</li> <li>Lập kế hoạch dự án</li> <li>Quản lý rủi ro dự án</li> </ul> <b>Chapter 4: Software project management</b> 4.1 Software project concepts <ul style="list-style-type: none"> <li>Human factors: Stakeholder / TeamLeader / Software Team / Communication issue</li> <li>Product factors: Software scope / Process / Project</li> </ul> 4.2 Software project management process <ul style="list-style-type: none"> <li>Project estimate</li> <li>Project planning</li> <li>Project risk management</li> </ul>	M1.1 M1.2	Note reading; Teaching;  <i>Exercise:</i> + Software project management tools; + Develop a simple software project plan for the capstone project	A1.2 A2.1
5	<b>Chương 5: Quản lý cấu hình phần mềm</b> 5.1 Khái niệm quản lý cấu hình phần mềm 5.2 Quy trình cấu hình phần mềm 5.3 Quản lý phiên bản 5.4 Quản lý thay đổi <i>Ví dụ và bài tập</i>  <b>Chapter 5: Software configuration management</b> 5.1 Software configuration management concepts 5.2 Software configuration process 5.3 Version management	M1.2 M1.3  M1.4 M2.1 M2.2	Note reading; Teaching;  <i>Exercise:</i> + GIT configuration management tool; + Set up a configuration management repository for the capstone project	A1.2 A2.1

	5.4 Change management Examples and exercises			
6	<b>Chương 6: Kỹ nghệ yêu cầu phần mềm (Requirement Engineering)</b> 6.1 Khái niệm 6.2 Tầm quan trọng của yêu cầu phần mềm 6.3 Yêu cầu chức năng và yêu cầu phi chức năng 6.4 Các hoạt động chính trong kỹ nghệ yêu cầu phần mềm: Thu thập, Phát hiện, Phân tích, Đặc tả, Thẩm định, Quản lý <i>Ví dụ và bài tập</i>  <b>Chapter 6: Requirement Engineering</b> 6.1 Concepts 6.2 The Importance of Software Requirements 6.3 Functional requirements and non-functional requirements 6.4 Key activities in the software requirement engineering: Inception, Elicitation, Analysis, Specification, Verification and Validation, Management Examples and exercises	M1.2; M1.3; M2.1; M2.2; M3.2; M3.3; M3.4	Note reading; Teaching;  <i>Exercise:</i> Define business processes. Detecting requirements and listing software features in capstone project	A1.2
7	6.5. Quy trình kỹ nghệ yêu cầu phần mềm <i>Ví dụ và bài tập</i>  6.5. Requirement engineering process Examples and exercises	M1.2; M1.3; M2.1; M2.2; M3.2; M3.3; M3.4	Note reading; Teaching;  <i>Exercise:</i> + Students present software requirement specification for capstone project	A1.1 A1.2 A2.1
8	<b>Chương 7: Thiết kế phần mềm</b> 7.1 Tổng quan về thiết kế phần mềm 7.2 Các khái niệm trong thiết kế phần mềm 7.3 Tính móc nối (Coupling) và tính kết dính (Cohesion) 7.4 Thiết kế kiến trúc 7.5 Thiết kế chi tiết <i>Ví dụ và bài tập</i>	M1.2; M1.3 M3.2	Note reading; Teaching;  <i>Exercise:</i> software design for capstone project	A1.1 A1.2 A2.1

	<b>Chapter 7: Software design</b> 7.1 Software Design Overview 7.2 Software Design Concepts 7.3 Coupling and cohesion 7.4 Architectural design 7.5 Detailed design Examples and exercises			
9	7.6 Thiết kế giao diện người dùng <ul style="list-style-type: none"> <li>- Các vấn đề thiết kế</li> <li>- Quy trình thiết kế UI</li> <li>- Phân tích người dùng</li> <li>- Tạo mẫu thử giao diện, mẫu thử tương tác</li> <li>- Đánh giá UI</li> <li>- Các công cụ thiết kế UI</li> </ul> <i><b>Ví dụ và bài tập</b></i>  7.6 User Interface Design <ul style="list-style-type: none"> <li>- Design issues</li> <li>- UI design process</li> <li>- User analysis</li> <li>- Create interface prototypes, interactive prototypes</li> <li>- UI evaluation</li> <li>- UI design tools</li> </ul> Examples and exercises	M1.2; M1.3	Note reading; Teaching;  <i>Exercise:</i> User interface design for capstone project.	A1.1 A1.2 A2.1
10	<b>Chương 8: Xây dựng phần mềm</b> 8.1 Khái niệm 8.2 Quy trình xây dựng phần mềm 8.3 Quy ước viết mã nguồn 8.4 Tái cấu trúc mã nguồn 8.5 Rà soát mã nguồn <i><b>Ví dụ và bài tập</b></i>  <b>Chapter 8: Software construction</b> 8.1 Concepts 8.2 Software Building Process 8.3 Code conventions 8.4 Refactoring the source code 8.5 Reviewing the source code Examples and exercises	M1.2; M2.2	Note reading; Teaching;  <i>Exercise:</i> coding conventions for teamwork. Build the source code for capstone project	A2.1 A1.2
11	<b>Chương 9: Đảm bảo chất lượng phần mềm</b> 9.1 Mô hình V&V 9.2 Các thuật ngữ về kiểm thử 9.3 Phương pháp kiểm thử hộp trắng - Khái niệm	M1.2; M2.2  M3.2; M3.3; M3.4	Note reading; Teaching;  <i>Exercise:</i> Design test cases for capstone	A1.1 A1.2 A2.1



	<ul style="list-style-type: none"> <li>- Vai trò</li> <li>- Kỹ thuật bao phủ luồng điều khiển</li> </ul> <p>9.4 Phương pháp kiểm thử hộp đen</p> <ul style="list-style-type: none"> <li>- Khái niệm</li> <li>- Vai trò</li> <li>- Kỹ thuật phân vùng tương đương</li> </ul> <p>9.5 Quản lý chất lượng phần mềm</p> <p><b>Chapter 9: Software quality assurance</b></p> <p>9.1 V&amp;V model</p> <p>9.2 Testing terms</p> <p>9.3 White box test method</p> <ul style="list-style-type: none"> <li>- Concept</li> <li>- Role</li> <li>- Control flow technique</li> </ul> <p>9.4 Black box test method</p> <ul style="list-style-type: none"> <li>- Concept</li> <li>- Role</li> <li>- Equivalent technique</li> </ul> <p>9.5 Software quality management</p>		project (white box / black box test method)	
12	<p>9.6 Bảo trì</p> <ul style="list-style-type: none"> <li>- Khái niệm</li> <li>- Quy trình nghiệp vụ</li> <li>- Các vấn đề còn tồn tại</li> <li>- Bảo trì trong các phương pháp phát triển phần mềm</li> <li>- Kiểm thử và bảo trì</li> </ul> <p>9.6 Maintenance</p> <ul style="list-style-type: none"> <li>- Concept</li> <li>- Business process</li> <li>- Outstanding issues</li> <li>- Maintenance in software development methods</li> <li>- Test and maintenance</li> </ul>	M1.2 M2.1	Note reading; Teaching;	A1.1 A1.2 A2.1
13	<p><b>Báo cáo bài tập nhóm</b></p> <p>Capstone project presentation</p>	M2.3 M2.4 M3.1; M3.2 M3.3; M3.4	Demonstration, Presentation; Discussion	A1.3
14	<p><b>Báo cáo bài tập nhóm</b></p>	M2.3	Demonstration,	

	Capstone project presentation	M2.4 M3.1; M3.2 M3.3; M3.4	Presentation; Discussion	A1.3
15	<b><i>Tổng kết và ôn tập</i></b> <b><i>Summary</i></b>		Trao đổi; Discussion;	

## 7. QUY ĐỊNH CỦA HỌC PHẦN - COURSE REQUIREMENT

- Chủ động đọc trước tài liệu giáo trình, in bài giảng (\*.pdf), chuẩn bị sẵn các câu hỏi.
- Dự lớp đầy đủ, theo dõi ghi chú vào tập bài giảng, chủ động đặt câu hỏi, tích cực tham gia phần thảo luận trên lớp.
- Làm bài tập về nhà đầy đủ theo yêu cầu của giảng viên.
- Thực hành cài đặt và sử dụng các công cụ theo hướng dẫn của giảng viên.
- Hoàn thành đầy đủ các nội dung của bài tập lớn (làm bài tập lớn và thảo luận theo nhóm (3-5 người)), có báo cáo và bảo vệ tại lớp.

### - Nếu môn học được giảng dạy theo hình thức blended learning:

- SV tự học online ở nhà qua hệ thống LMS trước khi đến học tại giảng đường; nội dung tự học online gồm đọc tài liệu, xem bài giảng video, làm bài trắc nghiệm.
- Buổi học trên lớp sinh viên sẽ thảo luận và thực hiện các bài tập giảng viên giao cho.

- Students should read textbook and lectures, print lectures (\*.pdf)

- Students should be required to attend classes.

- Students need to complete exercise and homeworks.

- Complete the capstone project (in groups (3-5 members))

### - If the subject is in the form of blended learning:

• Students self-study online through LMS; Online self-study content includes reading documents, watching video lectures, and taking quizzes.

• In class, students will perform assignments and discussion.

## 8. NGÀY PHÊ DUYỆT - DATE: .....

**Chủ tịch hội đồng**

**Committee chair**

**Nhóm xây dựng đề cương**

**Course preparation group**

## 9. QUÁ TRÌNH CẬP NHẬT - UPDATE INFORMATION

<b>ST T No</b>	<b>Nội dung điều chỉnh Content of the update</b>	<b>Ngày tháng được phê duyet Date accepted</b>	<b>Áp dụng từ kỳ/ khóa A pplicable from</b>	<b>Ghi chú Note</b>
1	<ul style="list-style-type: none"> <li>- Thay đổi học phần học trước: <ul style="list-style-type: none"> <li>○ Loại bỏ học phần học trước IT3010E: Data Structures and Algorithms</li> <li>○ Bổ sung học phần học trước IT3100E: Object-oriented Programming</li> </ul> </li> </ul>	14.02.2022		
2	.....			